

## CaaS Xen Hypervisor

Generated by Doxygen 1.8.4

Tue Aug 13 2013 15:17:21



# Contents

<b>1</b>	<b>Data Structure Index</b>	<b>1</b>
1.1	Data Structures . . . . .	1
<b>2</b>	<b>File Index</b>	<b>5</b>
2.1	File List . . . . .	5
<b>3</b>	<b>Data Structure Documentation</b>	<b>9</b>
3.1	_SHA256_CTX Struct Reference . . . . .	9
3.1.1	Detailed Description . . . . .	9
3.1.2	Field Documentation . . . . .	9
3.1.2.1	bitcount . . . . .	9
3.1.2.2	buffer . . . . .	9
3.1.2.3	state . . . . .	9
3.2	_SHA512_CTX Struct Reference . . . . .	10
3.2.1	Field Documentation . . . . .	10
3.2.1.1	bitcount . . . . .	10
3.2.1.2	buffer . . . . .	10
3.2.1.3	state . . . . .	10
3.3	blk_buffer Struct Reference . . . . .	10
3.3.1	Field Documentation . . . . .	10
3.3.1.1	gref . . . . .	10
3.3.1.2	page . . . . .	10
3.4	blk_req Struct Reference . . . . .	10
3.4.1	Field Documentation . . . . .	11
3.4.1.1	aiocb . . . . .	11
3.4.1.2	next . . . . .	11
3.4.1.3	rand_value . . . . .	11
3.5	blkback_aiocb Struct Reference . . . . .	11
3.5.1	Detailed Description . . . . .	11
3.5.2	Field Documentation . . . . .	11
3.5.2.1	b_dev . . . . .	11
3.5.2.2	f_aiocb . . . . .	11

3.5.2.3	pages_new	11
3.5.2.4	pages_order	11
3.5.2.5	pages_orig	11
3.5.2.6	req	11
3.6	blkback_dev Struct Reference	11
3.6.1	Field Documentation	12
3.6.1.1	b_thread	12
3.6.1.2	count_read	12
3.6.1.3	count_write	12
3.6.1.4	dom	12
3.6.1.5	domc_info	12
3.6.1.6	events	12
3.6.1.7	evtchn	12
3.6.1.8	f_dev	12
3.6.1.9	f_thread	12
3.6.1.10	frontend	12
3.6.1.11	gmap	12
3.6.1.12	handle	12
3.6.1.13	info	12
3.6.1.14	local_port	12
3.6.1.15	nodename	12
3.6.1.16	outstanding	12
3.6.1.17	ring	13
3.6.1.18	ring_ref	13
3.7	blkback_info Struct Reference	13
3.7.1	Field Documentation	13
3.7.1.1	barrier	13
3.7.1.2	bootable	13
3.7.1.3	dev	13
3.7.1.4	encrypt	13
3.7.1.5	flush	13
3.7.1.6	hotplug_status	13
3.7.1.7	info	13
3.7.1.8	mode	13
3.7.1.9	online	13
3.7.1.10	sector_size	13
3.7.1.11	sectors	13
3.7.1.12	type	13
3.8	blkfront_aiocb Struct Reference	14
3.8.1	Field Documentation	14



3.8.1.1	<a href="#">aio_buf</a>	14
3.8.1.2	<a href="#">aio_cb</a>	14
3.8.1.3	<a href="#">aio_dev</a>	14
3.8.1.4	<a href="#">aio_nbytes</a>	14
3.8.1.5	<a href="#">aio_offset</a>	14
3.8.1.6	<a href="#">data</a>	14
3.8.1.7	<a href="#">gref</a>	14
3.8.1.8	<a href="#">is_write</a>	14
3.8.1.9	<a href="#">n</a>	14
3.8.1.10	<a href="#">total_bytes</a>	14
3.9	<a href="#">blkfront_dev Struct Reference</a>	14
3.9.1	<a href="#">Field Documentation</a>	15
3.9.1.1	<a href="#">backend</a>	15
3.9.1.2	<a href="#">dom</a>	15
3.9.1.3	<a href="#">events</a>	15
3.9.1.4	<a href="#">evtchn</a>	15
3.9.1.5	<a href="#">handle</a>	15
3.9.1.6	<a href="#">info</a>	15
3.9.1.7	<a href="#">nodename</a>	15
3.9.1.8	<a href="#">ring</a>	15
3.9.1.9	<a href="#">ring_ref</a>	15
3.10	<a href="#">blkfront_info Struct Reference</a>	15
3.10.1	<a href="#">Field Documentation</a>	15
3.10.1.1	<a href="#">barrier</a>	15
3.10.1.2	<a href="#">flush</a>	15
3.10.1.3	<a href="#">info</a>	15
3.10.1.4	<a href="#">mode</a>	16
3.10.1.5	<a href="#">sector_size</a>	16
3.10.1.6	<a href="#">sectors</a>	16
3.11	<a href="#">boot_video_info Struct Reference</a>	16
3.11.1	<a href="#">Field Documentation</a>	16
3.11.1.1	<a href="#">blue_pos</a>	16
3.11.1.2	<a href="#">blue_size</a>	16
3.11.1.3	<a href="#">capabilities</a>	16
3.11.1.4	<a href="#">green_pos</a>	16
3.11.1.5	<a href="#">green_size</a>	16
3.11.1.6	<a href="#">lfb_base</a>	16
3.11.1.7	<a href="#">lfb_depth</a>	17
3.11.1.8	<a href="#">lfb_height</a>	17
3.11.1.9	<a href="#">lfb_linelength</a>	17

3.11.1.10 lfb_size . . . . .	17
3.11.1.11 lfb_width . . . . .	17
3.11.1.12 orig_video_cols . . . . .	17
3.11.1.13 orig_video_isVGA . . . . .	17
3.11.1.14 orig_video_lines . . . . .	17
3.11.1.15 orig_video_mode . . . . .	17
3.11.1.16 orig_video_points . . . . .	17
3.11.1.17 orig_x . . . . .	17
3.11.1.18 orig_y . . . . .	17
3.11.1.19 red_pos . . . . .	17
3.11.1.20 red_size . . . . .	17
3.11.1.21 rsvd_pos . . . . .	17
3.11.1.22 rsvd_size . . . . .	17
3.11.1.23 vesa_attrib . . . . .	17
3.11.1.24 vesapm_off . . . . .	17
3.11.1.25 vesapm_seg . . . . .	17
3.12 buffer Struct Reference . . . . .	17
3.12.1 Field Documentation . . . . .	18
3.12.1.1 capacity . . . . .	18
3.12.1.2 consumed . . . . .	18
3.12.1.3 data . . . . .	18
3.12.1.4 max_capacity . . . . .	18
3.12.1.5 size . . . . .	18
3.13 buffer_t Struct Reference . . . . .	18
3.13.1 Field Documentation . . . . .	18
3.13.1.1 alloc_size . . . . .	18
3.13.1.2 bytes . . . . .	18
3.13.1.3 is_owner . . . . .	18
3.13.1.4 size . . . . .	18
3.14 caas_op Struct Reference . . . . .	18
3.14.1 Field Documentation . . . . .	19
3.14.1.1 buf . . . . .	19
3.14.1.2 cmd . . . . .	19
3.14.1.3 size . . . . .	19
3.15 cap_t Union Reference . . . . .	19
3.15.1 Field Documentation . . . . .	19
3.15.1.1 duration . . . . .	19
3.15.1.2 manufacturer_id . . . . .	19
3.15.1.3 num_pcrs . . . . .	19
3.15.1.4 owned . . . . .	19

3.15.1.5	perm_flags	19
3.15.1.6	stclear_flags	19
3.15.1.7	timeout	19
3.15.1.8	tpm_version	19
3.15.1.9	tpm_version_1_2	19
3.16	changed_domain Struct Reference	20
3.16.1	Field Documentation	20
3.16.1.1	domid	20
3.16.1.2	list	20
3.16.1.3	nbentry	20
3.17	changed_node Struct Reference	20
3.17.1	Field Documentation	20
3.17.1.1	list	20
3.17.1.2	node	20
3.17.1.3	recurse	20
3.18	cipherInstance Struct Reference	20
3.18.1	Field Documentation	21
3.18.1.1	IV	21
3.18.1.2	IV_old	21
3.18.1.3	mode	21
3.19	CRYPTO_INFO Struct Reference	21
3.19.1	Field Documentation	21
3.19.1.1	algorithmID	21
3.19.1.2	encScheme	21
3.19.1.3	keyInfo	21
3.19.1.4	sigScheme	21
3.20	cspipe Struct Reference	21
3.20.1	Detailed Description	21
3.20.2	Field Documentation	22
3.20.2.1	domid_domc	22
3.20.2.2	domid_domu	22
3.20.2.3	magic_nr	22
3.20.2.4	s_key	22
3.21	dom0_vga_console_info Struct Reference	22
3.21.1	Field Documentation	22
3.21.1.1	bits_per_pixel	22
3.21.1.2	blue_pos	22
3.21.1.3	blue_size	23
3.21.1.4	bytes_per_line	23
3.21.1.5	columns	23

3.21.1.6	<a href="#">cursor_x</a>	23
3.21.1.7	<a href="#">cursor_y</a>	23
3.21.1.8	<a href="#">font_height</a>	23
3.21.1.9	<a href="#">green_pos</a>	23
3.21.1.10	<a href="#">green_size</a>	23
3.21.1.11	<a href="#">height</a>	23
3.21.1.12	<a href="#">lfb_base</a>	23
3.21.1.13	<a href="#">lfb_size</a>	23
3.21.1.14	<a href="#">red_pos</a>	23
3.21.1.15	<a href="#">red_size</a>	23
3.21.1.16	<a href="#">rows</a>	23
3.21.1.17	<a href="#">rsvd_pos</a>	23
3.21.1.18	<a href="#">rsvd_size</a>	23
3.21.1.19	<a href="#">text_mode_3</a>	23
3.21.1.20	<a href="#">u</a>	23
3.21.1.21	<a href="#">vesa_lfb</a>	23
3.21.1.22	<a href="#">video_type</a>	23
3.21.1.23	<a href="#">width</a>	23
3.22	<a href="#">domain Struct Reference</a>	23
3.22.1	<a href="#">Member Enumeration Documentation</a>	25
3.22.1.1	<a href="#">anonymous enum</a>	25
3.22.2	<a href="#">Member Function Documentation</a>	26
3.22.2.1	<a href="#">DECLARE_BITMAP</a>	26
3.22.3	<a href="#">Field Documentation</a>	26
3.22.3.1	<a href="#">arch</a>	26
3.22.3.2	<a href="#">buffer</a>	26
3.22.3.3	<a href="#">conn</a>	26
3.22.3.4	<a href="#">conspath</a>	26
3.22.3.5	<a href="#">cpupool</a>	26
3.22.3.6	<a href="#">debugger_attached</a>	26
3.22.3.7	<a href="#">disable_migrate</a>	26
3.22.3.8	<a href="#">domain_dirty_cpumask</a>	26
3.22.3.9	<a href="#">domain_id</a>	26
3.22.3.10	<a href="#">domain_lock</a>	26
3.22.3.11	<a href="#">domid</a>	26
3.22.3.12	<a href="#">domid</a>	26
3.22.3.13	<a href="#">event_count</a>	26
3.22.3.14	<a href="#">event_lock</a>	26
3.22.3.15	<a href="#">evtchn</a>	26
3.22.3.16	<a href="#">grant_table</a>	26

3.22.3.17 handle . . . . .	26
3.22.3.18 hypercall_deadlock_mutex . . . . .	26
3.22.3.19 interface . . . . .	26
3.22.3.20 interface . . . . .	26
3.22.3.21 iomem_caps . . . . .	26
3.22.3.22 irq_caps . . . . .	26
3.22.3.23 is_dead . . . . .	26
3.22.3.24 is_dying . . . . .	26
3.22.3.25 is_hvm . . . . .	26
3.22.3.26 is_paused_by_controller . . . . .	27
3.22.3.27 is_pinned . . . . .	27
3.22.3.28 is_privileged . . . . .	27
3.22.3.29 is_shut_down . . . . .	27
3.22.3.30 is_shutting_down . . . . .	27
3.22.3.31 last_alloc_node . . . . .	27
3.22.3.32 list . . . . .	27
3.22.3.33 local_port . . . . .	27
3.22.3.34 log_fd . . . . .	27
3.22.3.35 master_fd . . . . .	27
3.22.3.36 max_pages . . . . .	27
3.22.3.37 max_vcpus . . . . .	27
3.22.3.38 mem_event . . . . .	27
3.22.3.39 mfn . . . . .	27
3.22.3.40 nbentry . . . . .	27
3.22.3.41 nbwatch . . . . .	27
3.22.3.42 need_iommu . . . . .	27
3.22.3.43 next . . . . .	27
3.22.3.44 next_in_hashbucket . . . . .	27
3.22.3.45 next_in_list . . . . .	27
3.22.3.46 next_period . . . . .	27
3.22.3.47 node_affinity . . . . .	27
3.22.3.48 node_affinity_lock . . . . .	27
3.22.3.49 nr_pirqs . . . . .	27
3.22.3.50 page_alloc_lock . . . . .	27
3.22.3.51 page_list . . . . .	27
3.22.3.52 path . . . . .	27
3.22.3.53 pause_count . . . . .	27
3.22.3.54 pirq_mask . . . . .	28
3.22.3.55 pirq_to_evtchn . . . . .	28
3.22.3.56 port . . . . .	28

3.22.3.57 profile_head . . . . .	28
3.22.3.58 rangesets . . . . .	28
3.22.3.59 rangesets_lock . . . . .	28
3.22.3.60 rcu . . . . .	28
3.22.3.61 refcnt . . . . .	28
3.22.3.62 remote_port . . . . .	28
3.22.3.63 remote_port . . . . .	28
3.22.3.64 ring_ref . . . . .	28
3.22.3.65 sched_priv . . . . .	28
3.22.3.66 shared_info . . . . .	28
3.22.3.67 shr_pages . . . . .	28
3.22.3.68 shutdown . . . . .	28
3.22.3.69 shutdown_code . . . . .	28
3.22.3.70 shutdown_lock . . . . .	28
3.22.3.71 slave_fd . . . . .	28
3.22.3.72 ssid . . . . .	28
3.22.3.73 suspend_evtchn . . . . .	28
3.22.3.74 target . . . . .	28
3.22.3.75 time_offset_seconds . . . . .	28
3.22.3.76 tmem . . . . .	28
3.22.3.77 tot_pages . . . . .	28
3.22.3.78 vcpu . . . . .	28
3.22.3.79 vm_assist . . . . .	28
3.22.3.80 watchdog_inuse_map . . . . .	28
3.22.3.81 watchdog_lock . . . . .	28
3.22.3.82 watchdog_timer . . . . .	29
3.22.3.83 xce_handle . . . . .	29
3.22.3.84 xenheap_pages . . . . .	29
3.22.3.85 xenoprof . . . . .	29
3.22.3.86 xenpage_list . . . . .	29
3.23 domain_create Struct Reference . . . . .	29
3.23.1 Field Documentation . . . . .	29
3.23.1.1 config_file . . . . .	29
3.23.1.2 console_autoconnect . . . . .	29
3.23.1.3 daemonize . . . . .	29
3.23.1.4 debug . . . . .	29
3.23.1.5 dryrun . . . . .	29
3.23.1.6 extra_config . . . . .	29
3.23.1.7 migrate_fd . . . . .	29
3.23.1.8 migration_domname_r . . . . .	29

3.23.1.9	paused	29
3.23.1.10	quiet	29
3.23.1.11	restore_file	30
3.24	domain_setup_info Struct Reference	30
3.24.1	Field Documentation	30
3.24.1.1	elf_paddr_offset	30
3.24.1.2	image_addr	30
3.24.1.3	image_len	30
3.24.1.4	load_symtab	30
3.24.1.5	pae_kernel	30
3.24.1.6	symtab_addr	30
3.24.1.7	symtab_len	30
3.24.1.8	v_end	30
3.24.1.9	v_kernend	30
3.24.1.10	v_kernentry	30
3.24.1.11	v_kernstart	30
3.24.1.12	v_start	30
3.25	domc_info Struct Reference	31
3.25.1	Detailed Description	31
3.25.2	Field Documentation	31
3.25.2.1	self	31
3.25.2.2	target	31
3.26	duration_t Struct Reference	31
3.26.1	Field Documentation	31
3.26.1.1	tpm_long	31
3.26.1.2	tpm_medium	31
3.26.1.3	tpm_short	31
3.27	evchn Struct Reference	31
3.27.1	Field Documentation	32
3.27.1.1	consumer_is_xen	32
3.27.1.2	interdomain	32
3.27.1.3	irq	32
3.27.1.4	next_port	32
3.27.1.5	notify_vcpu_id	32
3.27.1.6	pirq	32
3.27.1.7	prev_port	32
3.27.1.8	remote_dom	32
3.27.1.9	remote_domid	32
3.27.1.10	remote_port	32
3.27.1.11	state	32

3.27.1.12 u	32
3.27.1.13 unbound	32
3.27.1.14 virq	32
3.28 fe_gref_map Struct Reference	33
3.28.1 Detailed Description	33
3.28.2 Field Documentation	33
3.28.2.1 gmap	33
3.28.2.2 gref	33
3.28.2.3 order	33
3.28.2.4 pbuf	33
3.28.2.5 pgmap	33
3.29 keyInstance Struct Reference	33
3.29.1 Field Documentation	33
3.29.1.1 direction	33
3.29.1.2 ek	33
3.29.1.3 keyLen	33
3.29.1.4 keyMaterial	34
3.29.1.5 Nr	34
3.29.1.6 rk	34
3.30 kvec Struct Reference	34
3.30.1 Field Documentation	34
3.30.1.1 iov_base	34
3.30.1.2 iov_len	34
3.31 libxl__cpuid_policy Struct Reference	34
3.31.1 Field Documentation	34
3.31.1.1 input	34
3.31.1.2 policy	34
3.32 libxl__device Struct Reference	35
3.32.1 Field Documentation	35
3.32.1.1 backend_devid	35
3.32.1.2 backend_domid	35
3.32.1.3 backend_kind	35
3.32.1.4 devid	35
3.32.1.5 domid	35
3.32.1.6 kind	35
3.33 libxl__device_model_starting Struct Reference	35
3.33.1 Field Documentation	35
3.33.1.1 dom_path	35
3.33.1.2 domid	35
3.33.1.3 for_spawn	35



3.34 libxl__gc Struct Reference . . . . .	36
3.34.1 Field Documentation . . . . .	36
3.34.1.1 alloc_maxsize . . . . .	36
3.34.1.2 alloc_ptrs . . . . .	36
3.34.1.3 owner . . . . .	36
3.35 libxl__spawn_starting Struct Reference . . . . .	36
3.35.1 Field Documentation . . . . .	36
3.35.1.1 intermediate . . . . .	36
3.35.1.2 what . . . . .	36
3.36 libxl__xen_console_reader Struct Reference . . . . .	36
3.36.1 Field Documentation . . . . .	37
3.36.1.1 buffer . . . . .	37
3.36.1.2 clear . . . . .	37
3.36.1.3 count . . . . .	37
3.36.1.4 incremental . . . . .	37
3.36.1.5 index . . . . .	37
3.36.1.6 size . . . . .	37
3.37 libxl__cpuarray Struct Reference . . . . .	37
3.37.1 Field Documentation . . . . .	37
3.37.1.1 array . . . . .	37
3.37.1.2 entries . . . . .	37
3.38 libxl__cpumap Struct Reference . . . . .	37
3.38.1 Field Documentation . . . . .	38
3.38.1.1 map . . . . .	38
3.38.1.2 size . . . . .	38
3.39 libxl__ctx Struct Reference . . . . .	38
3.39.1 Field Documentation . . . . .	38
3.39.1.1 lg . . . . .	38
3.39.1.2 version_info . . . . .	38
3.39.1.3 waitpid_instead . . . . .	38
3.39.1.4 xch . . . . .	38
3.39.1.5 xsh . . . . .	38
3.40 libxl__domain_config Struct Reference . . . . .	38
3.40.1 Field Documentation . . . . .	39
3.40.1.1 b_info . . . . .	39
3.40.1.2 c_info . . . . .	39
3.40.1.3 disks . . . . .	39
3.40.1.4 dm_info . . . . .	39
3.40.1.5 domc_disks . . . . .	39
3.40.1.6 domc_vifs . . . . .	39

3.40.1.7	num_disks	39
3.40.1.8	num_domc_disks	39
3.40.1.9	num_domc_vifs	39
3.40.1.10	num_pcidevs	39
3.40.1.11	num_vfbs	39
3.40.1.12	num_vif2s	39
3.40.1.13	num_vifs	39
3.40.1.14	num_vkbs	39
3.40.1.15	on_crash	39
3.40.1.16	on_poweroff	39
3.40.1.17	on_reboot	39
3.40.1.18	on_watchdog	39
3.40.1.19	pcidevs	40
3.40.1.20	vfbs	40
3.40.1.21	vif2s	40
3.40.1.22	vifs	40
3.40.1.23	vkbs	40
3.41	libxl_domain_suspend_info Struct Reference	40
3.41.1	Field Documentation	40
3.41.1.1	flags	40
3.41.1.2	suspend_callback	40
3.42	libxl_event Struct Reference	40
3.42.1	Field Documentation	40
3.42.1.1	path	40
3.42.1.2	token	40
3.42.1.3	type	41
3.43	libxl_file_reference Struct Reference	41
3.43.1	Field Documentation	41
3.43.1.1	data	41
3.43.1.2	mapped	41
3.43.1.3	path	41
3.43.1.4	size	41
3.44	libxl_waiter Struct Reference	41
3.44.1	Field Documentation	41
3.44.1.1	path	41
3.44.1.2	token	41
3.45	mem_event_domain Struct Reference	42
3.45.1	Field Documentation	42
3.45.1.1	front_ring	42
3.45.1.2	ring_lock	42

3.45.1.3	ring_page	42
3.45.1.4	shared_page	42
3.45.1.5	xen_port	42
3.46	memop_args Struct Reference	42
3.46.1	Field Documentation	42
3.46.1.1	domain	42
3.47	memory_map_context Struct Reference	42
3.47.1	Field Documentation	43
3.47.1.1	map	43
3.47.1.2	n	43
3.47.1.3	s	43
3.48	migrate_info Struct Reference	43
3.48.1	Field Documentation	43
3.48.1.1	cpu	43
3.48.1.2	data	43
3.48.1.3	func	43
3.48.1.4	nest	43
3.48.1.5	vcpu	43
3.49	mmu_update Struct Reference	43
3.49.1	Field Documentation	43
3.49.1.1	ptr	43
3.49.1.2	val	44
3.50	mmuext_op Struct Reference	44
3.50.1	Field Documentation	44
3.50.1.1	arg1	44
3.50.1.2	arg2	44
3.50.1.3	cmd	44
3.50.1.4	linear_addr	44
3.50.1.5	mfn	44
3.50.1.6	nr_ents	44
3.50.1.7	src_mfn	44
3.50.1.8	vcpumask	44
3.51	multicall_entry Struct Reference	44
3.51.1	Field Documentation	45
3.51.1.1	args	45
3.51.1.2	op	45
3.51.1.3	result	45
3.52	oiap_sess Struct Reference	45
3.52.1	Field Documentation	45
3.52.1.1	enonce	45

3.52.1.2	handle	45
3.53	Opt_args Struct Reference	45
3.53.1	Detailed Description	46
3.53.2	Member Enumeration Documentation	46
3.53.2.1	EntropySource	46
3.53.2.2	Mode	46
3.53.2.3	StartUp	46
3.53.3	Field Documentation	46
3.53.3.1	entropysrc	46
3.53.3.2	loglevel	46
3.53.3.3	mode	46
3.53.3.4	startup	46
3.54	osap_sess Struct Reference	47
3.54.1	Field Documentation	47
3.54.1.1	enonce	47
3.54.1.2	handle	47
3.54.1.3	secret	47
3.55	pack_buf_t Struct Reference	47
3.55.1	Field Documentation	47
3.55.1.1	data	47
3.55.1.2	size	47
3.56	pack_constbuf_t Struct Reference	47
3.56.1	Field Documentation	48
3.56.1.1	data	48
3.56.1.2	size	48
3.57	permanent_flags_t Struct Reference	48
3.57.1	Field Documentation	48
3.57.1.1	allowMaintenance	48
3.57.1.2	CEKPUUsed	48
3.57.1.3	deactivated	48
3.57.1.4	disable	48
3.57.1.5	disableFullIDALogicInfo	48
3.57.1.6	disableOwnerClear	48
3.57.1.7	enableRevokeEK	48
3.57.1.8	FIPS	48
3.57.1.9	maintenanceDone	49
3.57.1.10	nvLocked	49
3.57.1.11	operator	49
3.57.1.12	ownership	49
3.57.1.13	physicalPresenceCMDEnable	49

3.57.1.14 physicalPresenceHWEEnable . . . . .	49
3.57.1.15 physicalPresenceLifetimeLock . . . . .	49
3.57.1.16 readPubek . . . . .	49
3.57.1.17 readSRKPub . . . . .	49
3.57.1.18 tag . . . . .	49
3.57.1.19 tpmEstablished . . . . .	49
3.57.1.20 TPMpost . . . . .	49
3.57.1.21 TPMpostLock . . . . .	49
3.58 ptwr_emulate_ctxt Struct Reference . . . . .	49
3.58.1 Field Documentation . . . . .	49
3.58.1.1 cr2 . . . . .	49
3.58.1.2 ctxt . . . . .	49
3.58.1.3 pte . . . . .	49
3.59 save_file_header Struct Reference . . . . .	49
3.59.1 Field Documentation . . . . .	50
3.59.1.1 byteorder . . . . .	50
3.59.1.2 magic . . . . .	50
3.59.1.3 mandatory_flags . . . . .	50
3.59.1.4 optional_data_len . . . . .	50
3.59.1.5 optional_flags . . . . .	50
3.60 schedid_name Struct Reference . . . . .	50
3.60.1 Field Documentation . . . . .	50
3.60.1.1 id . . . . .	50
3.60.1.2 name . . . . .	50
3.61 shared_info Struct Reference . . . . .	50
3.61.1 Field Documentation . . . . .	51
3.61.1.1 arch . . . . .	51
3.61.1.2 evtchn_mask . . . . .	51
3.61.1.3 evtchn_pending . . . . .	51
3.61.1.4 vcpu_info . . . . .	51
3.61.1.5 wc_nsec . . . . .	51
3.61.1.6 wc_sec . . . . .	51
3.61.1.7 wc_version . . . . .	51
3.62 shared_info_any_t Union Reference . . . . .	51
3.62.1 Field Documentation . . . . .	51
3.62.1.1 s . . . . .	51
3.63 shpage Struct Reference . . . . .	51
3.63.1 Field Documentation . . . . .	52
3.63.1.1 grantref . . . . .	52
3.63.1.2 page . . . . .	52

3.64	start_info Struct Reference	52
3.64.1	Field Documentation	52
3.64.1.1	caas_mfn	52
3.64.1.2	cmd_line	52
3.64.1.3	console	52
3.64.1.4	dom0	52
3.64.1.5	domU	53
3.64.1.6	evtchn	53
3.64.1.7	first_p2m_pfn	53
3.64.1.8	flags	53
3.64.1.9	info_off	53
3.64.1.10	info_size	53
3.64.1.11	magic	53
3.64.1.12	mfn	53
3.64.1.13	mfn_list	53
3.64.1.14	mod_len	53
3.64.1.15	mod_start	53
3.64.1.16	nr_p2m_frames	53
3.64.1.17	nr_pages	53
3.64.1.18	nr_pt_frames	53
3.64.1.19	pt_base	53
3.64.1.20	shared_info	53
3.64.1.21	store_evtchn	53
3.64.1.22	store_mfn	53
3.65	start_info_any_t Union Reference	53
3.65.1	Field Documentation	53
3.65.1.1	s	53
3.66	stclear_flags_t Struct Reference	54
3.66.1	Field Documentation	54
3.66.1.1	bGlobalLock	54
3.66.1.2	deactivated	54
3.66.1.3	disableForceClear	54
3.66.1.4	physicalPresence	54
3.66.1.5	physicalPresenceLock	54
3.66.1.6	tag	54
3.67	suspendinfo Struct Reference	54
3.67.1	Field Documentation	54
3.67.1.1	domid	54
3.67.1.2	flags	54
3.67.1.3	gc	54

3.67.1.4	guest_responded	55
3.67.1.5	hvm	55
3.67.1.6	suspend_eventchn	55
3.67.1.7	xce	55
3.68	tc_info Struct Reference	55
3.68.1	Detailed Description	55
3.68.2	Field Documentation	55
3.68.2.1	chip	55
3.68.2.2	hvh	55
3.69	tc_state Struct Reference	55
3.69.1	Field Documentation	56
3.69.1.1	info	56
3.69.1.2	key	56
3.69.1.3	keypw	56
3.69.1.4	loadedKeyHandle	56
3.69.1.5	oi	56
3.69.1.6	srkpw	56
3.70	TCS_AUTH Struct Reference	56
3.70.1	Field Documentation	56
3.70.1.1	AuthHandle	56
3.70.1.2	fContinueAuthSession	56
3.70.1.3	HMAC	56
3.70.1.4	NonceEven	56
3.70.1.5	NonceOdd	56
3.71	tdVTPM_GLOBALS Struct Reference	56
3.71.1	Field Documentation	57
3.71.1.1	be_thread	57
3.71.1.2	bootKey	57
3.71.1.3	bootKeyWrap	57
3.71.1.4	connected_dmis	57
3.71.1.5	dmi_map	57
3.71.1.6	dmi_thread	57
3.71.1.7	hp_thread	57
3.71.1.8	keyAuth	57
3.71.1.9	manager_tcs_handle	57
3.71.1.10	master_thread	57
3.71.1.11	mig_keys	57
3.71.1.12	owner_usage_auth	57
3.71.1.13	srk_usage_auth	57
3.71.1.14	storage_key_usage_auth	57

3.71.1.15 storageKey . . . . .	57
3.71.1.16 storageKeyHandle . . . . .	57
3.71.1.17 storageKeyWrap . . . . .	58
3.72 tdVTPM_MIGKEY_LIST Struct Reference . . . . .	58
3.72.1 Field Documentation . . . . .	58
3.72.1.1 key . . . . .	58
3.72.1.2 name . . . . .	58
3.72.1.3 name_size . . . . .	58
3.72.1.4 next . . . . .	58
3.73 timeout_t Struct Reference . . . . .	58
3.73.1 Field Documentation . . . . .	58
3.73.1.1 a . . . . .	58
3.73.1.2 b . . . . .	58
3.73.1.3 c . . . . .	58
3.73.1.4 d . . . . .	58
3.74 tmem_oid Struct Reference . . . . .	59
3.74.1 Detailed Description . . . . .	59
3.74.2 Field Documentation . . . . .	59
3.74.2.1 oid . . . . .	59
3.75 TPM_BOUND_DATA Struct Reference . . . . .	59
3.75.1 Field Documentation . . . . .	59
3.75.1.1 payload . . . . .	59
3.75.1.2 payloadData . . . . .	59
3.75.1.3 ver . . . . .	59
3.76 tpm_chip Struct Reference . . . . .	59
3.76.1 Field Documentation . . . . .	60
3.76.1.1 baseaddr . . . . .	60
3.76.1.2 data_buffer . . . . .	60
3.76.1.3 data_len . . . . .	60
3.76.1.4 did . . . . .	60
3.76.1.5 duration . . . . .	60
3.76.1.6 enabled_localities . . . . .	60
3.76.1.7 int_queue . . . . .	60
3.76.1.8 irq . . . . .	60
3.76.1.9 list . . . . .	60
3.76.1.10 locality . . . . .	60
3.76.1.11 pages . . . . .	60
3.76.1.12 read_queue . . . . .	60
3.76.1.13 rid . . . . .	60
3.76.1.14 timeout_a . . . . .	60



3.76.1.15 timeout_b	60
3.76.1.16 timeout_c	60
3.76.1.17 timeout_d	60
3.76.1.18 vid	61
3.77 tpm_cmd Struct Reference	61
3.77.1 Field Documentation	61
3.77.1.1 header	61
3.77.1.2 params	61
3.78 tpm_cmd_header Union Reference	61
3.78.1 Field Documentation	61
3.78.1.1 in	61
3.78.1.2 out	61
3.79 tpm_cmd_params Union Reference	61
3.79.1 Field Documentation	62
3.79.1.1 flushspecific_in	62
3.79.1.2 getcap_in	62
3.79.1.3 getcap_out	62
3.79.1.4 getrandom_in	62
3.79.1.5 getrandom_out	62
3.79.1.6 oiap_out	62
3.79.1.7 osap_in	62
3.79.1.8 osap_out	62
3.79.1.9 pcrextend_in	62
3.79.1.10 pcrread_in	62
3.79.1.11 pcrread_out	62
3.79.1.12 readpubek_out	62
3.79.1.13 readpubek_out_buffer	62
3.80 TPM_DIGEST Struct Reference	62
3.80.1 Field Documentation	63
3.80.1.1 digest	63
3.81 tpm_flushspecific_in Struct Reference	63
3.81.1 Field Documentation	63
3.81.1.1 handle	63
3.81.1.2 resourceType	63
3.82 tpm_getcap_params_in Struct Reference	63
3.82.1 Field Documentation	63
3.82.1.1 cap	63
3.82.1.2 subcap	63
3.82.1.3 subcap_size	63
3.83 tpm_getcap_params_out Struct Reference	64

3.83.1	Field Documentation	64
3.83.1.1	cap	64
3.83.1.2	cap_size	64
3.84	tpm_getrandom_in Struct Reference	64
3.84.1	Field Documentation	64
3.84.1.1	numbytes	64
3.85	tpm_getrandom_out Struct Reference	64
3.85.1	Field Documentation	64
3.85.1.1	numrandbytes	64
3.85.1.2	randbytes	64
3.86	tpm_hmac_ctx_t Struct Reference	65
3.86.1	Field Documentation	65
3.86.1.1	ctx	65
3.86.1.2	k_opad	65
3.87	tpm_input_header Struct Reference	65
3.87.1	Field Documentation	65
3.87.1.1	length	65
3.87.1.2	ordinal	65
3.87.1.3	tag	65
3.88	TPM_KEY Struct Reference	65
3.88.1	Field Documentation	66
3.88.1.1	algorithmParms	66
3.88.1.2	authDataUsage	66
3.88.1.3	encData	66
3.88.1.4	encDataSize	66
3.88.1.5	keyFlags	66
3.88.1.6	keyUsage	66
3.88.1.7	PCRInfo	66
3.88.1.8	PCRInfoSize	66
3.88.1.9	pubKey	66
3.88.1.10	ver	66
3.89	TPM_KEY12 Struct Reference	66
3.89.1	Field Documentation	67
3.89.1.1	algorithmParms	67
3.89.1.2	authDataUsage	67
3.89.1.3	encData	67
3.89.1.4	encDataSize	67
3.89.1.5	fill	67
3.89.1.6	keyFlags	67
3.89.1.7	keyUsage	67

3.89.1.8 PCRInfo . . . . .	67
3.89.1.9 PCRInfoSize . . . . .	67
3.89.1.10 pubKey . . . . .	67
3.89.1.11 tag . . . . .	67
3.90 TPM_KEY_PARMS Struct Reference . . . . .	67
3.90.1 Field Documentation . . . . .	68
3.90.1.1 algorithmID . . . . .	68
3.90.1.2 algorithmID . . . . .	68
3.90.1.3 encScheme . . . . .	68
3.90.1.4 encScheme . . . . .	68
3.90.1.5 parms . . . . .	68
3.90.1.6 parms . . . . .	68
3.90.1.7 parmSize . . . . .	68
3.90.1.8 parmSize . . . . .	68
3.90.1.9 sigScheme . . . . .	68
3.90.1.10 sigScheme . . . . .	68
3.91 TPM_NONCE Struct Reference . . . . .	68
3.91.1 Field Documentation . . . . .	68
3.91.1.1 nonce . . . . .	68
3.92 tpm_oiap_out Struct Reference . . . . .	68
3.92.1 Field Documentation . . . . .	69
3.92.1.1 authHandle . . . . .	69
3.92.1.2 nonceEven . . . . .	69
3.93 tpm_osap_in Struct Reference . . . . .	69
3.93.1 Field Documentation . . . . .	69
3.93.1.1 entity_type . . . . .	69
3.93.1.2 entity_value . . . . .	69
3.93.1.3 nonce_odd_osap . . . . .	69
3.94 tpm_osap_out Struct Reference . . . . .	69
3.94.1 Field Documentation . . . . .	69
3.94.1.1 authhandle . . . . .	69
3.94.1.2 nonce_even . . . . .	69
3.94.1.3 nonce_even_osap . . . . .	69
3.95 tpm_output_header Struct Reference . . . . .	70
3.95.1 Field Documentation . . . . .	70
3.95.1.1 length . . . . .	70
3.95.1.2 return_code . . . . .	70
3.95.1.3 tag . . . . .	70
3.96 TPM_PCR_COMPOSITE Struct Reference . . . . .	70
3.96.1 Field Documentation . . . . .	70

3.96.1.1	pcrValue	70
3.96.1.2	select	70
3.96.1.3	valueSize	70
3.97	TPM_PCR_INFO Struct Reference	70
3.97.1	Field Documentation	71
3.97.1.1	digestAtCreation	71
3.97.1.2	digestAtRelease	71
3.97.1.3	pcrSelection	71
3.98	TPM_PCR_SELECTION Struct Reference	71
3.98.1	Field Documentation	71
3.98.1.1	pcrSelect	71
3.98.1.2	sizeOfSelect	71
3.99	tpm_pcrextend_in Struct Reference	71
3.99.1	Field Documentation	72
3.99.1.1	hash	72
3.99.1.2	pcr_idx	72
3.100	tpm_pcrread_in Struct Reference	72
3.100.1	Field Documentation	72
3.100.1.1	pcr_idx	72
3.101	tpm_pcrread_out Struct Reference	72
3.101.1	Field Documentation	72
3.101.1.1	pcr_result	72
3.102	TPM_PUBKEY Struct Reference	72
3.102.1	Field Documentation	73
3.102.1.1	algorithmParms	73
3.102.1.2	pubKey	73
3.103	tpm_readpubek_params_out Struct Reference	73
3.103.1	Field Documentation	73
3.103.1.1	algorithm	73
3.103.1.2	checksum	73
3.103.1.3	encscheme	73
3.103.1.4	keysize	73
3.103.1.5	modulus	73
3.103.1.6	parameters	73
3.103.1.7	paramsize	73
3.103.1.8	sigscheme	73
3.104	TPM_RSA_KEY_PARMS Struct Reference	74
3.104.1	Field Documentation	74
3.104.1.1	exponent	74
3.104.1.2	exponent	74

3.104.1.3 exponentSize . . . . .	74
3.104.1.4 exponentSize . . . . .	74
3.104.1.5 keyLength . . . . .	74
3.104.1.6 keyLength . . . . .	74
3.104.1.7 numPrimes . . . . .	74
3.104.1.8 numPrimes . . . . .	74
3.105tpm_sha1_ctx_t Struct Reference . . . . .	74
3.105.1 Field Documentation . . . . .	75
3.105.1.1 buf . . . . .	75
3.105.1.2 count_hi . . . . .	75
3.105.1.3 count_lo . . . . .	75
3.105.1.4 h . . . . .	75
3.106TPM_STORE_PUBKEY Struct Reference . . . . .	75
3.106.1 Field Documentation . . . . .	75
3.106.1.1 key . . . . .	75
3.106.1.2 key . . . . .	75
3.106.1.3 keyLength . . . . .	75
3.106.1.4 keyLength . . . . .	75
3.107TPM_STORED_DATA Struct Reference . . . . .	75
3.107.1 Field Documentation . . . . .	76
3.107.1.1 encData . . . . .	76
3.107.1.2 encDataSize . . . . .	76
3.107.1.3 sealInfo . . . . .	76
3.107.1.4 sealInfoSize . . . . .	76
3.107.1.5 ver . . . . .	76
3.108TPM_VERSION Struct Reference . . . . .	76
3.108.1 Field Documentation . . . . .	76
3.108.1.1 major . . . . .	76
3.108.1.2 minor . . . . .	76
3.108.1.3 revMajor . . . . .	76
3.108.1.4 revMinor . . . . .	76
3.109tpm_version_1_2_t Struct Reference . . . . .	76
3.109.1 Field Documentation . . . . .	77
3.109.1.1 Major . . . . .	77
3.109.1.2 Minor . . . . .	77
3.109.1.3 revMajor . . . . .	77
3.109.1.4 revMinor . . . . .	77
3.109.1.5 tag . . . . .	77
3.110tpm_version_t Struct Reference . . . . .	77
3.110.1 Field Documentation . . . . .	77

3.110.1.1 Major	77
3.110.1.2 Minor	77
3.110.1.3 revMajor	77
3.110.1.4 revMinor	77
3.111tpmback_dev Struct Reference	77
3.111.1 Field Documentation	78
3.111.1.1 close_callback	78
3.111.1.2 events	78
3.111.1.3 exclusive_uuids	78
3.111.1.4 flags	78
3.111.1.5 map	78
3.111.1.6 num_alloc	78
3.111.1.7 num_tpms	78
3.111.1.8 open_callback	78
3.111.1.9 resume_callback	78
3.111.1.10suspend_callback	78
3.111.1.11tpmlist	78
3.112tpmcmd Struct Reference	78
3.112.1 Field Documentation	78
3.112.1.1 domid	78
3.112.1.2 handle	78
3.112.1.3 req	78
3.112.1.4 req_len	78
3.112.1.5 resp	78
3.112.1.6 resp_len	79
3.112.1.7 uuid	79
3.113tpmfront_dev Struct Reference	79
3.113.1 Field Documentation	79
3.113.1.1 bedomid	79
3.113.1.2 bepath	79
3.113.1.3 evtchn	79
3.113.1.4 nodename	79
3.113.1.5 pages	79
3.113.1.6 respbuf	79
3.113.1.7 resplen	79
3.113.1.8 ring_ref	79
3.113.1.9 state	79
3.113.1.10tx	79
3.113.1.11waiting	79
3.113.1.12waitq	79

3.114tpmif Struct Reference . . . . .	80
3.114.1 Member Enumeration Documentation . . . . .	80
3.114.1.1 anonymous enum . . . . .	80
3.114.2 Field Documentation . . . . .	80
3.114.2.1 domid . . . . .	80
3.114.2.2 evtchn . . . . .	80
3.114.2.3 fe_path . . . . .	80
3.114.2.4 fe_state_path . . . . .	80
3.114.2.5 flags . . . . .	80
3.114.2.6 handle . . . . .	80
3.114.2.7 pages . . . . .	80
3.114.2.8 state . . . . .	80
3.114.2.9 status . . . . .	80
3.114.2.10x . . . . .	81
3.114.2.11uuid . . . . .	81
3.115transaction Struct Reference . . . . .	81
3.115.1 Field Documentation . . . . .	81
3.115.1.1 changed_domains . . . . .	81
3.115.1.2 changes . . . . .	81
3.115.1.3 generation . . . . .	81
3.115.1.4 id . . . . .	81
3.115.1.5 list . . . . .	81
3.115.1.6 tdb . . . . .	81
3.115.1.7 tdb_name . . . . .	81
3.116vcpu Struct Reference . . . . .	81
3.116.1 Member Function Documentation . . . . .	82
3.116.1.1 XEN_GUEST_HANDLE . . . . .	82
3.116.2 Field Documentation . . . . .	82
3.116.2.1 arch . . . . .	82
3.116.2.2 continue_hypercall_tasklet . . . . .	82
3.116.2.3 cpu_affinity . . . . .	82
3.116.2.4 cpu_affinity_tmp . . . . .	82
3.116.2.5 defer_shutdown . . . . .	82
3.116.2.6 domain . . . . .	82
3.116.2.7 fpu_dirtied . . . . .	83
3.116.2.8 fpu_initialised . . . . .	83
3.116.2.9 is_initialised . . . . .	83
3.116.2.10s_running . . . . .	83
3.116.2.11is_urgent . . . . .	83
3.116.2.12ast_run_time . . . . .	83

3.116.2.13	mc_state	83
3.116.2.14	next_in_list	83
3.116.2.15	pause_count	83
3.116.2.16	pause_flags	83
3.116.2.17	paused_for_shutdown	83
3.116.2.18	periodic_last_event	83
3.116.2.19	periodic_period	83
3.116.2.20	periodic_timer	83
3.116.2.21	pirq_evtchn_head	83
3.116.2.22	poll_evtchn	83
3.116.2.23	poll_timer	83
3.116.2.24	processor	83
3.116.2.25	runstate	83
3.116.2.26	sched_priv	83
3.116.2.27	singleshot_timer	83
3.116.2.28	vcpu_dirty_cpumask	83
3.116.2.29	vcpu_id	83
3.116.2.30	vcpu_info	83
3.116.2.31	virq_lock	83
3.116.2.32	virq_to_evtchn	83
3.116.2.33	waitqueue_vcpu	83
3.117	vcpu_guest_context_any_t Union Reference	84
3.117.1	Field Documentation	84
3.117.1.1	c	84
3.118	vcpu_guest_context_u Union Reference	84
3.118.1	Field Documentation	84
3.118.1.1	cmp	84
3.118.1.2	nat	84
3.119	vcpu_info Struct Reference	84
3.119.1	Field Documentation	85
3.119.1.1	arch	85
3.119.1.2	evtchn_pending_sel	85
3.119.1.3	evtchn_upcall_mask	85
3.119.1.4	evtchn_upcall_pending	85
3.119.1.5	time	85
3.120	vcpu_time_info Struct Reference	85
3.120.1	Field Documentation	85
3.120.1.1	pad0	85
3.120.1.2	pad1	85
3.120.1.3	system_time	85



3.120.1.4 tsc_shift . . . . .	85
3.120.1.5 tsc_timestamp . . . . .	85
3.120.1.6 tsc_to_system_mul . . . . .	85
3.120.1.7 version . . . . .	85
3.121 vfs_cmd Struct Reference . . . . .	85
3.121.1 Field Documentation . . . . .	86
3.121.1.1 close . . . . .	86
3.121.1.2 init . . . . .	86
3.121.1.3 lseek . . . . .	86
3.121.1.4 majorminor . . . . .	86
3.121.1.5 open . . . . .	86
3.121.1.6 read . . . . .	86
3.121.1.7 stat . . . . .	86
3.121.1.8 type . . . . .	86
3.121.1.9 u . . . . .	86
3.121.1.10 write . . . . .	86
3.122 vfs_cmd_close Struct Reference . . . . .	86
3.122.1 Field Documentation . . . . .	87
3.122.1.1 errn . . . . .	87
3.122.1.2 fd . . . . .	87
3.122.1.3 ret . . . . .	87
3.123 vfs_cmd_init Struct Reference . . . . .	87
3.123.1 Detailed Description . . . . .	87
3.123.2 Field Documentation . . . . .	87
3.123.2.1 ret . . . . .	87
3.124 vfs_cmd_lseek Struct Reference . . . . .	87
3.124.1 Field Documentation . . . . .	88
3.124.1.1 errn . . . . .	88
3.124.1.2 fd . . . . .	88
3.124.1.3 offset . . . . .	88
3.124.1.4 ret . . . . .	88
3.124.1.5 whence . . . . .	88
3.125 vfs_cmd_majorminor Struct Reference . . . . .	88
3.125.1 Field Documentation . . . . .	88
3.125.1.1 major . . . . .	88
3.125.1.2 minor . . . . .	88
3.125.1.3 path . . . . .	88
3.126 vfs_cmd_open Struct Reference . . . . .	88
3.126.1 Field Documentation . . . . .	89
3.126.1.1 errn . . . . .	89

3.126.1.2 flags . . . . .	89
3.126.1.3 mode . . . . .	89
3.126.1.4 path . . . . .	89
3.126.1.5 ret . . . . .	89
3.127vfs_cmd_read Struct Reference . . . . .	89
3.127.1 Field Documentation . . . . .	89
3.127.1.1 errn . . . . .	89
3.127.1.2 fd . . . . .	89
3.127.1.3 gref . . . . .	89
3.127.1.4 nbyte . . . . .	89
3.127.1.5 ret . . . . .	89
3.128vfs_cmd_stat Struct Reference . . . . .	89
3.128.1 Field Documentation . . . . .	90
3.128.1.1 errn . . . . .	90
3.128.1.2 gref . . . . .	90
3.128.1.3 path . . . . .	90
3.128.1.4 ret . . . . .	90
3.129vfs_cmd_write Struct Reference . . . . .	90
3.129.1 Field Documentation . . . . .	90
3.129.1.1 errn . . . . .	90
3.129.1.2 fd . . . . .	90
3.129.1.3 gref . . . . .	90
3.129.1.4 nbyte . . . . .	90
3.129.1.5 ret . . . . .	90
3.130vfs_gref_map Struct Reference . . . . .	90
3.130.1 Field Documentation . . . . .	91
3.130.1.1 grefs . . . . .	91
3.130.1.2 num . . . . .	91
3.131vmcb Struct Reference . . . . .	91
3.131.1 Detailed Description . . . . .	91
3.131.2 Field Documentation . . . . .	91
3.131.2.1 s_key . . . . .	91
3.132VTPM_DMI_RESOURCE_T Struct Reference . . . . .	91
3.132.1 Field Documentation . . . . .	92
3.132.1.1 connected . . . . .	92
3.132.1.2 dmi_domain_id . . . . .	92
3.132.1.3 dmi_id . . . . .	92
3.132.1.4 DMI_measurement . . . . .	92
3.132.1.5 dmi_pid . . . . .	92
3.132.1.6 dmi_type . . . . .	92

3.132.1.7 NVM_measurement . . . . .	92
3.132.1.8 NVMLocation . . . . .	92
3.132.1.9 rx_tpm_ipc_h . . . . .	92
3.132.1.10rx_vtpm_ipc_h . . . . .	92
3.132.1.11TCSCContext . . . . .	92
3.132.1.12x_tpm_ipc_h . . . . .	92
3.132.1.13x_vtpm_ipc_h . . . . .	92
3.133vtpm_ipc_handle_t Struct Reference . . . . .	92
3.133.1 Field Documentation . . . . .	93
3.133.1.1 fh . . . . .	93
3.133.1.2 flags . . . . .	93
3.133.1.3 name . . . . .	93
3.134vtpm_thread_params_s Struct Reference . . . . .	93
3.134.1 Field Documentation . . . . .	93
3.134.1.1 fw_rx_ipc_h . . . . .	93
3.134.1.2 fw_tpm . . . . .	93
3.134.1.3 fw_tx_ipc_h . . . . .	93
3.134.1.4 is_priv . . . . .	93
3.134.1.5 rx_ipc_h . . . . .	93
3.134.1.6 thread_name . . . . .	93
3.134.1.7 tx_ipc_h . . . . .	93
3.135xc_core_header Struct Reference . . . . .	93
3.135.1 Field Documentation . . . . .	94
3.135.1.1 xch_ctxt_offset . . . . .	94
3.135.1.2 xch_index_offset . . . . .	94
3.135.1.3 xch_magic . . . . .	94
3.135.1.4 xch_nr_pages . . . . .	94
3.135.1.5 xch_nr_vcpus . . . . .	94
3.135.1.6 xch_pages_offset . . . . .	94
3.136xc_cpupoolinfo Struct Reference . . . . .	94
3.136.1 Field Documentation . . . . .	94
3.136.1.1 cpumap . . . . .	94
3.136.1.2 cpupool_id . . . . .	94
3.136.1.3 n_dom . . . . .	94
3.136.1.4 sched_id . . . . .	94
3.137xc_cx_stat Struct Reference . . . . .	94
3.137.1 Field Documentation . . . . .	95
3.137.1.1 cc3 . . . . .	95
3.137.1.2 cc6 . . . . .	95
3.137.1.3 idle_time . . . . .	95

3.137.1.4 last . . . . .	95
3.137.1.5 nr . . . . .	95
3.137.1.6 pc3 . . . . .	95
3.137.1.7 pc6 . . . . .	95
3.137.1.8 pc7 . . . . .	95
3.137.1.9 residencies . . . . .	95
3.137.1.10triggers . . . . .	95
3.138xc_dom_arch Struct Reference . . . . .	95
3.138.1 Field Documentation . . . . .	96
3.138.1.1 alloc_magic_pages . . . . .	96
3.138.1.2 count_pgtables . . . . .	96
3.138.1.3 guest_type . . . . .	96
3.138.1.4 native_protocol . . . . .	96
3.138.1.5 next . . . . .	96
3.138.1.6 page_shift . . . . .	96
3.138.1.7 setup_pgtables . . . . .	96
3.138.1.8 shared_info . . . . .	96
3.138.1.9 sizeof_pfn . . . . .	96
3.138.1.10start_info . . . . .	96
3.138.1.11vcpu . . . . .	96
3.139xc_dom_image Struct Reference . . . . .	96
3.139.1 Field Documentation . . . . .	97
3.139.1.1 alloc_bootstack . . . . .	97
3.139.1.2 alloc_domU_map . . . . .	97
3.139.1.3 alloc_file_map . . . . .	97
3.139.1.4 alloc_malloc . . . . .	97
3.139.1.5 alloc_mem_map . . . . .	97
3.139.1.6 allocate . . . . .	97
3.139.1.7 arch_hooks . . . . .	98
3.139.1.8 bootstack_pfn . . . . .	98
3.139.1.9 bsd_syntab_start . . . . .	98
3.139.1.10caas_pfn . . . . .	98
3.139.1.11cmdline . . . . .	98
3.139.1.12console_evtchn . . . . .	98
3.139.1.13console_pfn . . . . .	98
3.139.1.14devicetree_seg . . . . .	98
3.139.1.15extra_pages . . . . .	98
3.139.1.16active . . . . .	98
3.139.1.17requested . . . . .	98
3.139.1.18flags . . . . .	98

3.139.1.19	guest_domid	98
3.139.1.20	guest_type	98
3.139.1.21	kernel_blob	98
3.139.1.22	kernel_loader	98
3.139.1.23	kernel_seg	98
3.139.1.24	kernel_size	98
3.139.1.25	memblocks	98
3.139.1.26	p2m_guest	98
3.139.1.27	p2m_host	98
3.139.1.28	p2m_seg	98
3.139.1.29	parms	98
3.139.1.30	pg_l1	98
3.139.1.31	pg_l2	98
3.139.1.32	pg_l3	98
3.139.1.33	pg_l4	98
3.139.1.34	pgtables	98
3.139.1.35	pgtables_seg	99
3.139.1.36	phys_pages	99
3.139.1.37	private_loader	99
3.139.1.38	ramdisk_blob	99
3.139.1.39	ramdisk_seg	99
3.139.1.40	ramdisk_size	99
3.139.1.41	realmodearea_log	99
3.139.1.42	shadow_enabled	99
3.139.1.43	shared_info_mfn	99
3.139.1.44	shared_info_pfn	99
3.139.1.45	start_info_pfn	99
3.139.1.46	superpages	99
3.139.1.47	total_pages	99
3.139.1.48	hpt_size_log2	99
3.139.1.49	virt_alloc_end	99
3.139.1.50	virt_pgtab_end	99
3.139.1.51	xch	99
3.139.1.52	xen_caps	99
3.139.1.53	xen_version	99
3.139.1.54	xenstore_evtchn	99
3.139.1.55	xenstore_pfn	99
3.140	xc_dom_loader Struct Reference	99
3.140.1	Field Documentation	100
3.140.1.1	loader	100

3.140.1.2 name	100
3.140.1.3 next	100
3.140.1.4 parser	100
3.140.1.5 probe	100
3.141xc_dom_mem Struct Reference	100
3.141.1 Field Documentation	100
3.141.1.1 memory	100
3.141.1.2 mmap_len	100
3.141.1.3 mmap_ptr	100
3.141.1.4 next	100
3.142xc_dom_phys Struct Reference	100
3.142.1 Field Documentation	101
3.142.1.1 count	101
3.142.1.2 first	101
3.142.1.3 next	101
3.142.1.4 ptr	101
3.143xc_dom_seg Struct Reference	101
3.143.1 Field Documentation	101
3.143.1.1 pfn	101
3.143.1.2 vend	101
3.143.1.3 vstart	101
3.144xc_dominfo Struct Reference	101
3.144.1 Field Documentation	102
3.144.1.1 blocked	102
3.144.1.2 cpu_time	102
3.144.1.3 cpupool	102
3.144.1.4 crashed	102
3.144.1.5 debugged	102
3.144.1.6 domid	102
3.144.1.7 dying	102
3.144.1.8 handle	102
3.144.1.9 hvm	102
3.144.1.10max_memkb	102
3.144.1.11max_vcpu_id	102
3.144.1.12nr_online_vcpus	102
3.144.1.13nr_pages	102
3.144.1.14nr_shared_pages	102
3.144.1.15paused	102
3.144.1.16running	103
3.144.1.17shared_info_frame	103

3.144.1.18 shutdown	103
3.144.1.19 shutdown_reason	103
3.144.1.20 ssidref	103
3.145 xc_error Struct Reference	103
3.145.1 Field Documentation	103
3.145.1.1 code	103
3.145.1.2 message	103
3.146 xc_get_cpufreq_para Struct Reference	103
3.146.1 Field Documentation	104
3.146.1.1 affected_cpus	104
3.146.1.2 cpu_num	104
3.146.1.3 cpuinfo_cur_freq	104
3.146.1.4 cpuinfo_max_freq	104
3.146.1.5 cpuinfo_min_freq	104
3.146.1.6 freq_num	104
3.146.1.7 gov_num	104
3.146.1.8 ondemand	104
3.146.1.9 scaling_available_frequencies	104
3.146.1.10 scaling_available_governors	104
3.146.1.11 scaling_cur_freq	104
3.146.1.12 scaling_driver	104
3.146.1.13 scaling_governor	104
3.146.1.14 scaling_max_freq	104
3.146.1.15 scaling_min_freq	104
3.146.1.16 turbo_enabled	104
3.146.1.17 u	104
3.146.1.18 userspace	104
3.147 xc_hypercall_buffer Struct Reference	104
3.147.1 Field Documentation	105
3.147.1.1 dir	105
3.147.1.2 hbuf	105
3.147.1.3 param_shadow	105
3.147.1.4 sz	105
3.147.1.5 ubuf	105
3.148 xc_px_stat Struct Reference	105
3.148.1 Field Documentation	105
3.148.1.1 cur	105
3.148.1.2 last	105
3.148.1.3 pt	105
3.148.1.4 total	105

3.148.1.5 trans_pt . . . . .	105
3.148.1.6 usable . . . . .	105
3.149xc_px_val Struct Reference . . . . .	106
3.149.1 Field Documentation . . . . .	106
3.149.1.1 count . . . . .	106
3.149.1.2 freq . . . . .	106
3.149.1.3 residency . . . . .	106
3.150xen_domctl Struct Reference . . . . .	106
3.150.1 Field Documentation . . . . .	107
3.150.1.1 access_required . . . . .	107
3.150.1.2 address_size . . . . .	107
3.150.1.3 arch_setup . . . . .	107
3.150.1.4 assign_device . . . . .	107
3.150.1.5 bind_pt_irq . . . . .	107
3.150.1.6 cmd . . . . .	107
3.150.1.7 createdomain . . . . .	107
3.150.1.8 debug_op . . . . .	107
3.150.1.9 disable_migrate . . . . .	107
3.150.1.10domain . . . . .	107
3.150.1.11ext_vcpucontext . . . . .	107
3.150.1.12gdbsx_domstatus . . . . .	107
3.150.1.13gdbsx_guest_memio . . . . .	108
3.150.1.14gdbsx_pauseunp_vcpu . . . . .	108
3.150.1.15get_device_group . . . . .	108
3.150.1.16getdomaininfo . . . . .	108
3.150.1.17getmemlist . . . . .	108
3.150.1.18getpageframeinfo . . . . .	108
3.150.1.19getpageframeinfo2 . . . . .	108
3.150.1.20getpageframeinfo3 . . . . .	108
3.150.1.21getvcpuinfo . . . . .	108
3.150.1.22hvmcontext . . . . .	108
3.150.1.23hvmcontext_partial . . . . .	108
3.150.1.24hypercall_init . . . . .	108
3.150.1.25interface_version . . . . .	108
3.150.1.26iomem_permission . . . . .	108
3.150.1.27ioport_mapping . . . . .	108
3.150.1.28ioport_permission . . . . .	108
3.150.1.29irq_permission . . . . .	108
3.150.1.30max_mem . . . . .	108
3.150.1.31max_vcpus . . . . .	108



3.150.1.32	mem_event_op	108
3.150.1.33	mem_sharing_op	108
3.150.1.34	memory_mapping	108
3.150.1.35	pad	108
3.150.1.36	pin_mem_cacheattr	108
3.150.1.37	real_mode_area	108
3.150.1.38	scheduler_op	108
3.150.1.39	sendtrigger	108
3.150.1.40	set_opt_feature	108
3.150.1.41	set_target	109
3.150.1.42	setdebugging	109
3.150.1.43	setdomainhandle	109
3.150.1.44	settimeoffset	109
3.150.1.45	shadow_op	109
3.150.1.46	subscribe	109
3.150.1.47	sc_info	109
3.150.1.48	u	109
3.150.1.49	vcpuaffinity	109
3.150.1.50	vcpucontext	109
3.151	xen_domctl_address_size Struct Reference	109
3.151.1	Field Documentation	109
3.151.1.1	size	109
3.152	xen_domctl_arch_setup Struct Reference	109
3.152.1	Field Documentation	110
3.152.1.1	flags	110
3.153	xen_domctl_assign_device Struct Reference	110
3.153.1	Field Documentation	110
3.153.1.1	machine_bdf	110
3.154	xen_domctl_bind_pt_irq Struct Reference	110
3.154.1	Field Documentation	111
3.154.1.1	bus	111
3.154.1.2	device	111
3.154.1.3	gflags	111
3.154.1.4	gtable	111
3.154.1.5	gvec	111
3.154.1.6	hvm_domid	111
3.154.1.7	intx	111
3.154.1.8	irq_type	111
3.154.1.9	isa	111
3.154.1.10	isa_irq	111

3.154.1.11machine_irq . . . . .	111
3.154.1.12msi . . . . .	111
3.154.1.13pci . . . . .	111
3.154.1.14u . . . . .	111
3.155xen_domctl_createdomain Struct Reference . . . . .	111
3.155.1 Field Documentation . . . . .	111
3.155.1.1 flags . . . . .	111
3.155.1.2 handle . . . . .	111
3.155.1.3 ssidref . . . . .	111
3.156xen_domctl_debug_op Struct Reference . . . . .	112
3.156.1 Field Documentation . . . . .	112
3.156.1.1 op . . . . .	112
3.156.1.2 vcpu . . . . .	112
3.157xen_domctl_disable_migrate Struct Reference . . . . .	112
3.157.1 Field Documentation . . . . .	112
3.157.1.1 disable . . . . .	112
3.158xen_domctl_ext_vcpucontext Struct Reference . . . . .	112
3.158.1 Field Documentation . . . . .	113
3.158.1.1 size . . . . .	113
3.158.1.2 vcpu . . . . .	113
3.159xen_domctl_gdbsx_domstatus Struct Reference . . . . .	113
3.159.1 Field Documentation . . . . .	113
3.159.1.1 paused . . . . .	113
3.159.1.2 vcpu_ev . . . . .	113
3.159.1.3 vcpu_id . . . . .	113
3.160xen_domctl_gdbsx_memio Struct Reference . . . . .	113
3.160.1 Field Documentation . . . . .	113
3.160.1.1 gva . . . . .	113
3.160.1.2 gwr . . . . .	113
3.160.1.3 len . . . . .	114
3.160.1.4 pgd3val . . . . .	114
3.160.1.5 remain . . . . .	114
3.160.1.6 uva . . . . .	114
3.161xen_domctl_gdbsx_pauseunp_vcpu Struct Reference . . . . .	114
3.161.1 Field Documentation . . . . .	114
3.161.1.1 vcpu . . . . .	114
3.162xen_domctl_get_device_group Struct Reference . . . . .	114
3.162.1 Member Function Documentation . . . . .	114
3.162.1.1 XEN_GUEST_HANDLE_64 . . . . .	114
3.162.2 Field Documentation . . . . .	114

3.162.2.1 machine_bdf . . . . .	114
3.162.2.2 max_sdevs . . . . .	115
3.162.2.3 num_sdevs . . . . .	115
3.163xen_domctl_getdomaininfo Struct Reference . . . . .	115
3.163.1 Field Documentation . . . . .	115
3.163.1.1 cpu_time . . . . .	115
3.163.1.2 cpupool . . . . .	115
3.163.1.3 domain . . . . .	115
3.163.1.4 flags . . . . .	115
3.163.1.5 handle . . . . .	115
3.163.1.6 max_pages . . . . .	115
3.163.1.7 max_vcpu_id . . . . .	115
3.163.1.8 nr_online_vcpus . . . . .	115
3.163.1.9 shared_info_frame . . . . .	115
3.163.1.10shr_pages . . . . .	115
3.163.1.11ssidref . . . . .	115
3.163.1.12tot_pages . . . . .	115
3.164xen_domctl_getmemlist Struct Reference . . . . .	116
3.164.1 Member Function Documentation . . . . .	116
3.164.1.1 XEN_GUEST_HANDLE_64 . . . . .	116
3.164.2 Field Documentation . . . . .	116
3.164.2.1 max_pfns . . . . .	116
3.164.2.2 num_pfns . . . . .	116
3.164.2.3 start_pfn . . . . .	116
3.165xen_domctl_getpageframeinfo Struct Reference . . . . .	116
3.165.1 Field Documentation . . . . .	116
3.165.1.1 gmfn . . . . .	116
3.165.1.2 type . . . . .	116
3.166xen_domctl_getpageframeinfo2 Struct Reference . . . . .	117
3.166.1 Member Function Documentation . . . . .	117
3.166.1.1 XEN_GUEST_HANDLE_64 . . . . .	117
3.166.2 Field Documentation . . . . .	117
3.166.2.1 num . . . . .	117
3.167xen_domctl_getpageframeinfo3 Struct Reference . . . . .	117
3.167.1 Member Function Documentation . . . . .	117
3.167.1.1 XEN_GUEST_HANDLE_64 . . . . .	117
3.167.2 Field Documentation . . . . .	117
3.167.2.1 num . . . . .	117
3.168xen_domctl_getvcpuinfo Struct Reference . . . . .	118
3.168.1 Field Documentation . . . . .	118

3.168.1.1 blocked . . . . .	118
3.168.1.2 cpu . . . . .	118
3.168.1.3 cpu_time . . . . .	118
3.168.1.4 online . . . . .	118
3.168.1.5 running . . . . .	118
3.168.1.6 vcpu . . . . .	118
3.169xen_domctl_hvmcontext Struct Reference . . . . .	118
3.169.1 Member Function Documentation . . . . .	118
3.169.1.1 XEN_GUEST_HANDLE_64 . . . . .	118
3.169.2 Field Documentation . . . . .	118
3.169.2.1 size . . . . .	118
3.170xen_domctl_hvmcontext_partial Struct Reference . . . . .	119
3.170.1 Member Function Documentation . . . . .	119
3.170.1.1 XEN_GUEST_HANDLE_64 . . . . .	119
3.170.2 Field Documentation . . . . .	119
3.170.2.1 instance . . . . .	119
3.170.2.2 type . . . . .	119
3.171xen_domctl_hypercall_init Struct Reference . . . . .	119
3.171.1 Field Documentation . . . . .	119
3.171.1.1 gmfn . . . . .	119
3.172xen_domctl_iomem_permission Struct Reference . . . . .	119
3.172.1 Field Documentation . . . . .	120
3.172.1.1 allow_access . . . . .	120
3.172.1.2 first_mfn . . . . .	120
3.172.1.3 nr_mfns . . . . .	120
3.173xen_domctl_ioport_mapping Struct Reference . . . . .	120
3.173.1 Field Documentation . . . . .	120
3.173.1.1 add_mapping . . . . .	120
3.173.1.2 first_gport . . . . .	120
3.173.1.3 first_mport . . . . .	120
3.173.1.4 nr_ports . . . . .	120
3.174xen_domctl_ioport_permission Struct Reference . . . . .	120
3.174.1 Field Documentation . . . . .	121
3.174.1.1 allow_access . . . . .	121
3.174.1.2 first_port . . . . .	121
3.174.1.3 nr_ports . . . . .	121
3.175xen_domctl_irq_permission Struct Reference . . . . .	121
3.175.1 Field Documentation . . . . .	121
3.175.1.1 allow_access . . . . .	121
3.175.1.2 pirq . . . . .	121

3.176xen_domctl_max_mem Struct Reference . . . . .	121
3.176.1 Field Documentation . . . . .	121
3.176.1.1 max_memkb . . . . .	121
3.177xen_domctl_max_vcpus Struct Reference . . . . .	121
3.177.1 Field Documentation . . . . .	122
3.177.1.1 max . . . . .	122
3.178xen_domctl_mem_event_op Struct Reference . . . . .	122
3.178.1 Field Documentation . . . . .	122
3.178.1.1 gfn . . . . .	122
3.178.1.2 mode . . . . .	122
3.178.1.3 op . . . . .	122
3.178.1.4 ring_addr . . . . .	122
3.178.1.5 shared_addr . . . . .	122
3.179xen_domctl_mem_sharing_op Struct Reference . . . . .	122
3.179.1 Field Documentation . . . . .	123
3.179.1.1 client_handle . . . . .	123
3.179.1.2 debug . . . . .	123
3.179.1.3 enable . . . . .	123
3.179.1.4 gfn . . . . .	123
3.179.1.5 grant_ref . . . . .	123
3.179.1.6 gref . . . . .	123
3.179.1.7 handle . . . . .	123
3.179.1.8 mfn . . . . .	123
3.179.1.9 nominate . . . . .	123
3.179.1.10op . . . . .	123
3.179.1.11share . . . . .	123
3.179.1.12source_handle . . . . .	123
3.179.1.13u . . . . .	123
3.179.1.14u . . . . .	123
3.179.1.15u . . . . .	123
3.180xen_domctl_memory_mapping Struct Reference . . . . .	124
3.180.1 Field Documentation . . . . .	124
3.180.1.1 add_mapping . . . . .	124
3.180.1.2 first_gfn . . . . .	124
3.180.1.3 first_mfn . . . . .	124
3.180.1.4 nr_mfns . . . . .	124
3.180.1.5 padding . . . . .	124
3.181xen_domctl_pin_mem_cacheattr Struct Reference . . . . .	124
3.181.1 Field Documentation . . . . .	124
3.181.1.1 end . . . . .	124

3.181.1.2 start . . . . .	124
3.181.1.3 type . . . . .	124
3.182xen_domctl_real_mode_area Struct Reference . . . . .	125
3.182.1 Field Documentation . . . . .	125
3.182.1.1 log . . . . .	125
3.183xen_domctl_scheduler_op Struct Reference . . . . .	125
3.183.1 Field Documentation . . . . .	125
3.183.1.1 cap . . . . .	125
3.183.1.2 cmd . . . . .	125
3.183.1.3 credit . . . . .	125
3.183.1.4 credit2 . . . . .	125
3.183.1.5 extratime . . . . .	126
3.183.1.6 latency . . . . .	126
3.183.1.7 period . . . . .	126
3.183.1.8 sched_id . . . . .	126
3.183.1.9 sedf . . . . .	126
3.183.1.10slice . . . . .	126
3.183.1.11u . . . . .	126
3.183.1.12weight . . . . .	126
3.183.1.13weight . . . . .	126
3.184xen_domctl_sendtrigger Struct Reference . . . . .	126
3.184.1 Field Documentation . . . . .	126
3.184.1.1 trigger . . . . .	126
3.184.1.2 vcpu . . . . .	126
3.185xen_domctl_set_access_required Struct Reference . . . . .	126
3.185.1 Field Documentation . . . . .	127
3.185.1.1 access_required . . . . .	127
3.186xen_domctl_set_opt_feature Struct Reference . . . . .	127
3.186.1 Field Documentation . . . . .	127
3.186.1.1 dummy . . . . .	127
3.187xen_domctl_set_target Struct Reference . . . . .	127
3.187.1 Field Documentation . . . . .	127
3.187.1.1 target . . . . .	127
3.188xen_domctl_setdebugging Struct Reference . . . . .	127
3.188.1 Field Documentation . . . . .	128
3.188.1.1 enable . . . . .	128
3.189xen_domctl_setdomainhandle Struct Reference . . . . .	128
3.189.1 Field Documentation . . . . .	128
3.189.1.1 handle . . . . .	128
3.190xen_domctl_settimeoffset Struct Reference . . . . .	128

3.190.1 Field Documentation . . . . .	128
3.190.1.1 time_offset_seconds . . . . .	128
3.191xen_domctl_shadow_op Struct Reference . . . . .	128
3.191.1 Member Function Documentation . . . . .	129
3.191.1.1 XEN_GUEST_HANDLE_64 . . . . .	129
3.191.2 Field Documentation . . . . .	129
3.191.2.1 mb . . . . .	129
3.191.2.2 mode . . . . .	129
3.191.2.3 op . . . . .	129
3.191.2.4 pages . . . . .	129
3.191.2.5 stats . . . . .	129
3.192xen_domctl_shadow_op_stats Struct Reference . . . . .	129
3.192.1 Field Documentation . . . . .	129
3.192.1.1 dirty_count . . . . .	129
3.192.1.2 fault_count . . . . .	129
3.193xen_domctl_subscribe Struct Reference . . . . .	129
3.193.1 Field Documentation . . . . .	130
3.193.1.1 port . . . . .	130
3.194xen_domctl_tsc_info Struct Reference . . . . .	130
3.194.1 Member Function Documentation . . . . .	130
3.194.1.1 XEN_GUEST_HANDLE_64 . . . . .	130
3.194.2 Field Documentation . . . . .	130
3.194.2.1 info . . . . .	130
3.195xen_domctl_vcpuaffinity Struct Reference . . . . .	130
3.195.1 Field Documentation . . . . .	131
3.195.1.1 cpumap . . . . .	131
3.195.1.2 vcpu . . . . .	131
3.196xen_domctl_vcpucontext Struct Reference . . . . .	131
3.196.1 Member Function Documentation . . . . .	131
3.196.1.1 XEN_GUEST_HANDLE_64 . . . . .	131
3.196.2 Field Documentation . . . . .	131
3.196.2.1 vcpu . . . . .	131
3.197xen_guest_tsc_info Struct Reference . . . . .	131
3.197.1 Field Documentation . . . . .	132
3.197.1.1 elapsed_nsec . . . . .	132
3.197.1.2 gtsc_khz . . . . .	132
3.197.1.3 incarnation . . . . .	132
3.197.1.4 pad . . . . .	132
3.197.1.5 tsc_mode . . . . .	132
3.198xen_multiboot_mod_list Struct Reference . . . . .	132

3.198.1 Field Documentation . . . . .	132
3.198.1.1 cmdline . . . . .	132
3.198.1.2 mod_end . . . . .	132
3.198.1.3 mod_start . . . . .	132
3.198.1.4 pad . . . . .	132
3.199xlc_cmd Struct Reference . . . . .	132
3.199.1 Field Documentation . . . . .	133
3.199.1.1 dombuild . . . . .	133
3.199.1.2 domcreate . . . . .	133
3.199.1.3 dommake . . . . .	133
3.199.1.4 getpage . . . . .	133
3.199.1.5 ping . . . . .	133
3.199.1.6 retpage . . . . .	133
3.199.1.7 type . . . . .	133
3.199.1.8 u . . . . .	133
3.200xlc_cmd_dombuild Struct Reference . . . . .	133
3.200.1 Field Documentation . . . . .	133
3.200.1.1 domid . . . . .	133
3.200.1.2 in_gref . . . . .	133
3.200.1.3 out_gref . . . . .	133
3.200.1.4 ret . . . . .	133
3.201xlc_cmd_domcreate Struct Reference . . . . .	134
3.201.1 Field Documentation . . . . .	134
3.201.1.1 domid . . . . .	134
3.201.1.2 domid_dc . . . . .	134
3.201.1.3 in_gref . . . . .	134
3.201.1.4 out_gref . . . . .	134
3.201.1.5 restore_fd . . . . .	134
3.201.1.6 ret . . . . .	134
3.202xlc_cmd_dommake Struct Reference . . . . .	134
3.202.1 Field Documentation . . . . .	134
3.202.1.1 domid . . . . .	134
3.202.1.2 in_gref . . . . .	134
3.202.1.3 out_gref . . . . .	134
3.202.1.4 ret . . . . .	135
3.203xlc_cmd_getpage Struct Reference . . . . .	135
3.203.1 Detailed Description . . . . .	135
3.203.2 Field Documentation . . . . .	135
3.203.2.1 in_gref . . . . .	135
3.203.2.2 out_gref . . . . .	135



3.204	<a href="#">xlc_cmd_ping Struct Reference</a>	135
3.204.1	<a href="#">Detailed Description</a>	135
3.204.2	<a href="#">Field Documentation</a>	135
3.204.2.1	<a href="#">ret</a>	135
3.205	<a href="#">xlc_cmd_retpage Struct Reference</a>	136
3.205.1	<a href="#">Field Documentation</a>	136
3.205.1.1	<a href="#">out_gref</a>	136
3.206	<a href="#">XIOObject Struct Reference</a>	136
3.206.1	<a href="#">Field Documentation</a>	136
3.206.1.1	<a href="#">ctx</a>	136
3.206.1.2	<a href="#">logger</a>	136
3.206.1.3	<a href="#">minmsglevel</a>	136
3.206.1.4	<a href="#">PyObject_HEAD</a>	136
3.207	<a href="#">yyalloc Union Reference</a>	136
3.207.1	<a href="#">Field Documentation</a>	137
3.207.1.1	<a href="#">yyls_alloc</a>	137
3.207.1.2	<a href="#">yyss_alloc</a>	137
3.207.1.3	<a href="#">yyvs_alloc</a>	137
3.208	<a href="#">YYLTYPE Struct Reference</a>	137
3.208.1	<a href="#">Field Documentation</a>	137
3.208.1.1	<a href="#">first_column</a>	137
3.208.1.2	<a href="#">first_line</a>	137
3.208.1.3	<a href="#">last_column</a>	137
3.208.1.4	<a href="#">last_line</a>	137
3.209	<a href="#">YYSTYPE Union Reference</a>	137
3.209.1	<a href="#">Field Documentation</a>	138
3.209.1.1	<a href="#">setting</a>	138
3.209.1.2	<a href="#">string</a>	138
<b>4</b>	<b><a href="#">File Documentation</a></b>	<b>139</b>
4.1	<a href="#">xen/extras/mini-os/arch/ia64/iow.c File Reference</a>	139
4.1.1	<a href="#">Function Documentation</a>	139
4.1.1.1	<a href="#">ioread32</a>	139
4.1.1.2	<a href="#">ioread8</a>	139
4.1.1.3	<a href="#">iowrite32</a>	139
4.1.1.4	<a href="#">iowrite8</a>	139
4.2	<a href="#">xen/extras/mini-os/arch/x86/iow.c File Reference</a>	139
4.2.1	<a href="#">Function Documentation</a>	140
4.2.1.1	<a href="#">ioread32</a>	140
4.2.1.2	<a href="#">ioread8</a>	140

4.2.1.3	<a href="#">iowrite32</a>	140
4.2.1.4	<a href="#">iowrite8</a>	140
4.3	<a href="#">xen/extras/mini-os/arch/x86/mm.c File Reference</a>	140
4.3.1	<a href="#">Macro Definition Documentation</a>	141
4.3.1.1	<a href="#">DEBUG</a>	141
4.3.1.2	<a href="#">DEMAND_MAP_PAGES</a>	141
4.3.1.3	<a href="#">HEAP_PAGES</a>	141
4.3.1.4	<a href="#">L1_P2M_ENTRIES</a>	141
4.3.1.5	<a href="#">L1_P2M_MASK</a>	141
4.3.1.6	<a href="#">L1_P2M_SHIFT</a>	141
4.3.1.7	<a href="#">L2_P2M_ENTRIES</a>	141
4.3.1.8	<a href="#">L2_P2M_MASK</a>	141
4.3.1.9	<a href="#">L2_P2M_SHIFT</a>	141
4.3.1.10	<a href="#">L3_P2M_ENTRIES</a>	141
4.3.1.11	<a href="#">L3_P2M_MASK</a>	141
4.3.1.12	<a href="#">L3_P2M_SHIFT</a>	141
4.3.1.13	<a href="#">MAP_BATCH</a>	141
4.3.1.14	<a href="#">MAX_CONTIG_ORDER</a>	141
4.3.1.15	<a href="#">MEM_TEST_MAX_ERRORS</a>	141
4.3.1.16	<a href="#">UNMAP_BATCH</a>	142
4.3.2	<a href="#">Function Documentation</a>	142
4.3.2.1	<a href="#">alloc_contig_pages</a>	142
4.3.2.2	<a href="#">allocate_ondemand</a>	142
4.3.2.3	<a href="#">arch_init_demand_mapping_area</a>	142
4.3.2.4	<a href="#">arch_init_mm</a>	142
4.3.2.5	<a href="#">arch_init_p2m</a>	142
4.3.2.6	<a href="#">do_map_frames</a>	142
4.3.2.7	<a href="#">map_frames_ex</a>	142
4.3.2.8	<a href="#">mem_test</a>	142
4.3.2.9	<a href="#">mfn_is_ram</a>	142
4.3.2.10	<a href="#">need_pgt</a>	142
4.3.2.11	<a href="#">page_walk</a>	142
4.3.2.12	<a href="#">unmap_frames</a>	142
4.3.3	<a href="#">Variable Documentation</a>	142
4.3.3.1	<a href="#">mfn_zero</a>	142
4.3.3.2	<a href="#">phys_to_machine_mapping</a>	142
4.3.3.3	<a href="#">shared_info</a>	142
4.3.3.4	<a href="#">stack</a>	142
4.4	<a href="#">xen/xen/arch/x86/mm.c File Reference</a>	143
4.4.1	<a href="#">Macro Definition Documentation</a>	145

4.4.1.1	<a href="#">adjust_guest_l1e</a>	145
4.4.1.2	<a href="#">adjust_guest_l2e</a>	145
4.4.1.3	<a href="#">adjust_guest_l3e</a>	145
4.4.1.4	<a href="#">alloc_l4_table</a>	145
4.4.1.5	<a href="#">define_get_linear_pagetable</a>	145
4.4.1.6	<a href="#">fixmap_domain_page</a>	145
4.4.1.7	<a href="#">fixunmap_domain_page</a>	145
4.4.1.8	<a href="#">flush_area</a>	145
4.4.1.9	<a href="#">free_l4_table</a>	145
4.4.1.10	<a href="#">l1_disallow_mask</a>	145
4.4.1.11	<a href="#">l1f_to_l1f</a>	146
4.4.1.12	<a href="#">l3_disallow_mask</a>	146
4.4.1.13	<a href="#">l1f_to_l1f</a>	146
4.4.1.14	<a href="#">MEM_LOG</a>	146
4.4.1.15	<a href="#">pae_flush_pgd</a>	146
4.4.1.16	<a href="#">PAGE_CACHE_ATTRS</a>	146
4.4.1.17	<a href="#">PTE_UPDATE_WITH_CMPXCHG</a>	146
4.4.1.18	<a href="#">unadjust_guest_l3e</a>	146
4.4.1.19	<a href="#">UPDATE_ENTRY</a>	146
4.4.2	<a href="#">Typedef Documentation</a>	146
4.4.2.1	<a href="#">e820entry_t</a>	146
4.4.3	<a href="#">Function Documentation</a>	146
4.4.3.1	<a href="#">__attribute__</a>	146
4.4.3.2	<a href="#">__set_fixmap</a>	146
4.4.3.3	<a href="#">arch_init_memory</a>	146
4.4.3.4	<a href="#">arch_memory_op</a>	146
4.4.3.5	<a href="#">boolean_param</a>	146
4.4.3.6	<a href="#">clear_superpage_mark</a>	146
4.4.3.7	<a href="#">create_grant_host_mapping</a>	146
4.4.3.8	<a href="#">define_get_linear_pagetable</a>	146
4.4.3.9	<a href="#">define_get_linear_pagetable</a>	146
4.4.3.10	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	146
4.4.3.11	<a href="#">destroy_gdt</a>	146
4.4.3.12	<a href="#">destroy_xen_mappings</a>	146
4.4.3.13	<a href="#">do_mmu_update</a>	147
4.4.3.14	<a href="#">do_mmuext_op</a>	147
4.4.3.15	<a href="#">do_set_gdt</a>	147
4.4.3.16	<a href="#">do_update_descriptor</a>	147
4.4.3.17	<a href="#">do_update_va_mapping</a>	147
4.4.3.18	<a href="#">do_update_va_mapping_otherdomain</a>	147

4.4.3.19	<a href="#">domain_get_maximum_gpfn</a>	147
4.4.3.20	<a href="#">donate_page</a>	147
4.4.3.21	<a href="#">free_page_type</a>	147
4.4.3.22	<a href="#">free_xen_pagetable</a>	147
4.4.3.23	<a href="#">get_page</a>	147
4.4.3.24	<a href="#">get_page_from_l1e</a>	147
4.4.3.25	<a href="#">get_page_type</a>	147
4.4.3.26	<a href="#">get_page_type_preemptible</a>	147
4.4.3.27	<a href="#">init_frametable</a>	147
4.4.3.28	<a href="#">is_iomem_page</a>	147
4.4.3.29	<a href="#">make_cr3</a>	147
4.4.3.30	<a href="#">map_ldt_shadow_page</a>	147
4.4.3.31	<a href="#">map_pages_to_xen</a>	147
4.4.3.32	<a href="#">memguard_guard_stack</a>	147
4.4.3.33	<a href="#">memguard_unguard_stack</a>	147
4.4.3.34	<a href="#">new_guest_cr3</a>	147
4.4.3.35	<a href="#">page_get_owner_and_reference</a>	147
4.4.3.36	<a href="#">page_is_ram_type</a>	147
4.4.3.37	<a href="#">page_make_private</a>	147
4.4.3.38	<a href="#">page_make_sharable</a>	147
4.4.3.39	<a href="#">ptwr_do_page_fault</a>	148
4.4.3.40	<a href="#">put_page</a>	148
4.4.3.41	<a href="#">put_page_from_l1e</a>	148
4.4.3.42	<a href="#">put_page_type</a>	148
4.4.3.43	<a href="#">put_page_type_preemptible</a>	148
4.4.3.44	<a href="#">replace_grant_host_mapping</a>	148
4.4.3.45	<a href="#">set_gdt</a>	148
4.4.3.46	<a href="#">share_xen_page_with_guest</a>	148
4.4.3.47	<a href="#">share_xen_page_with_privileged_guests</a>	148
4.4.3.48	<a href="#">steal_page</a>	148
4.4.3.49	<a href="#">update_cr3</a>	148
4.4.3.50	<a href="#">write_ptbase</a>	148
4.4.4	<a href="#">Variable Documentation</a>	148
4.4.4.1	<a href="#">opt_allow_superpage</a>	148
4.5	<a href="#">xen/extras/mini-os/blkfront.c File Reference</a>	148
4.5.1	<a href="#">Macro Definition Documentation</a>	149
4.5.1.1	<a href="#">BLK_RING_SIZE</a>	149
4.5.1.2	<a href="#">GRANT_INVALID_REF</a>	149
4.5.1.3	<a href="#">strtoul</a>	149
4.5.2	<a href="#">Function Documentation</a>	149

4.5.2.1	<a href="#">blkfront_aio</a>	149
4.5.2.2	<a href="#">blkfront_aio_poll</a>	149
4.5.2.3	<a href="#">blkfront_aio_push_operation</a>	149
4.5.2.4	<a href="#">blkfront_handler</a>	149
4.5.2.5	<a href="#">blkfront_io</a>	149
4.5.2.6	<a href="#">blkfront_sync</a>	149
4.5.2.7	<a href="#">DECLARE_WAIT_QUEUE_HEAD</a>	149
4.5.2.8	<a href="#">init_blkfront</a>	149
4.5.2.9	<a href="#">shutdown_blkfront</a>	149
4.6	<a href="#">xen/extras/mini-os/console/console.c File Reference</a>	150
4.6.1	<a href="#">Macro Definition Documentation</a>	150
4.6.1.1	<a href="#">USE_XEN_CONSOLE</a>	150
4.6.2	<a href="#">Function Documentation</a>	150
4.6.2.1	<a href="#">console_print</a>	150
4.6.2.2	<a href="#">fini_console</a>	150
4.6.2.3	<a href="#">init_console</a>	150
4.6.2.4	<a href="#">print</a>	150
4.6.2.5	<a href="#">printk</a>	150
4.6.2.6	<a href="#">xencons_rx</a>	150
4.6.2.7	<a href="#">xencons_tx</a>	150
4.6.2.8	<a href="#">xprintk</a>	150
4.7	<a href="#">xen/extras/mini-os/discovery.c File Reference</a>	150
4.8	<a href="#">xen/extras/mini-os/include/blkfront.h File Reference</a>	151
4.8.1	<a href="#">Macro Definition Documentation</a>	151
4.8.1.1	<a href="#">blkfront_aio_read</a>	151
4.8.1.2	<a href="#">blkfront_aio_write</a>	151
4.8.1.3	<a href="#">blkfront_read</a>	151
4.8.1.4	<a href="#">blkfront_write</a>	151
4.8.2	<a href="#">Function Documentation</a>	151
4.8.2.1	<a href="#">blkfront_aio</a>	151
4.8.2.2	<a href="#">blkfront_aio_poll</a>	152
4.8.2.3	<a href="#">blkfront_aio_push_operation</a>	152
4.8.2.4	<a href="#">blkfront_io</a>	152
4.8.2.5	<a href="#">blkfront_sync</a>	152
4.8.2.6	<a href="#">init_blkfront</a>	152
4.8.2.7	<a href="#">shutdown_blkfront</a>	152
4.8.3	<a href="#">Variable Documentation</a>	152
4.8.3.1	<a href="#">blkfront_queue</a>	152
4.9	<a href="#">xen/extras/mini-os/include/byteorder.h File Reference</a>	152
4.9.1	<a href="#">Macro Definition Documentation</a>	152

4.9.1.1	<a href="#">be16_to_cpu</a>	152
4.9.1.2	<a href="#">be32_to_cpu</a>	152
4.9.1.3	<a href="#">be64_to_cpu</a>	152
4.9.1.4	<a href="#">cpu_to_be16</a>	152
4.9.1.5	<a href="#">cpu_to_be32</a>	152
4.9.1.6	<a href="#">cpu_to_be64</a>	152
4.9.1.7	<a href="#">cpu_to_le16</a>	152
4.9.1.8	<a href="#">cpu_to_le32</a>	153
4.9.1.9	<a href="#">cpu_to_le64</a>	153
4.9.1.10	<a href="#">le16_to_cpu</a>	153
4.9.1.11	<a href="#">le32_to_cpu</a>	153
4.9.1.12	<a href="#">le64_to_cpu</a>	153
4.10	<a href="#">xen/extras/mini-os/include/byteswap.h File Reference</a>	153
4.10.1	<a href="#">Macro Definition Documentation</a>	153
4.10.1.1	<a href="#">bswap_16</a>	153
4.10.1.2	<a href="#">bswap_32</a>	153
4.10.1.3	<a href="#">bswap_64</a>	153
4.11	<a href="#">xen/extras/mini-os/include/discovery.h File Reference</a>	154
4.11.1	<a href="#">Function Documentation</a>	154
4.11.1.1	<a href="#">discovery</a>	154
4.11.1.2	<a href="#">discovery_start_kernel</a>	154
4.12	<a href="#">xen/extras/mini-os/include/endian.h File Reference</a>	154
4.12.1	<a href="#">Macro Definition Documentation</a>	154
4.12.1.1	<a href="#">__BIG_ENDIAN</a>	154
4.12.1.2	<a href="#">__FLOAT_WORD_ORDER</a>	154
4.12.1.3	<a href="#">__LITTLE_ENDIAN</a>	154
4.12.1.4	<a href="#">__PDP_ENDIAN</a>	154
4.12.1.5	<a href="#">ARCH_ENDIAN_H</a>	154
4.13	<a href="#">xen/extras/mini-os/include/ia64/arch_endian.h File Reference</a>	154
4.14	<a href="#">xen/extras/mini-os/include/x86/arch_endian.h File Reference</a>	154
4.15	<a href="#">xen/extras/mini-os/include/ia64/arch_wordsize.h File Reference</a>	154
4.15.1	<a href="#">Macro Definition Documentation</a>	155
4.15.1.1	<a href="#">__WORDSIZE</a>	155
4.16	<a href="#">xen/extras/mini-os/include/x86/x86_32/arch_wordsize.h File Reference</a>	155
4.16.1	<a href="#">Macro Definition Documentation</a>	155
4.16.1.1	<a href="#">__WORDSIZE</a>	155
4.17	<a href="#">xen/extras/mini-os/include/x86/x86_64/arch_wordsize.h File Reference</a>	155
4.17.1	<a href="#">Macro Definition Documentation</a>	155
4.17.1.1	<a href="#">__WORDSIZE</a>	155
4.17.1.2	<a href="#">__WORDSIZE_COMPAT32</a>	155

4.18 xen/extras/mini-os/include/iow.h File Reference . . . . .	155
4.18.1 Function Documentation . . . . .	155
4.18.1.1 ioread32 . . . . .	155
4.18.1.2 ioread8 . . . . .	155
4.18.1.3 iowrite32 . . . . .	155
4.18.1.4 iowrite8 . . . . .	155
4.19 xen/extras/mini-os/include/lib.h File Reference . . . . .	156
4.19.1 Macro Definition Documentation . . . . .	157
4.19.1.1 _p . . . . .	157
4.19.1.2 ARRAY_SIZE . . . . .	157
4.19.1.3 ASSERT . . . . .	157
4.19.1.4 BUG_ON . . . . .	157
4.19.1.5 RAND_MIX . . . . .	157
4.19.2 Function Documentation . . . . .	157
4.19.2.1 memcmp . . . . .	157
4.19.2.2 memcpy . . . . .	157
4.19.2.3 memset . . . . .	157
4.19.2.4 rand . . . . .	157
4.19.2.5 sanity_check . . . . .	157
4.19.2.6 scnprintf . . . . .	157
4.19.2.7 simple_strtol . . . . .	157
4.19.2.8 simple_strtoll . . . . .	157
4.19.2.9 simple_strtoul . . . . .	157
4.19.2.10 simple_strtoull . . . . .	157
4.19.2.11 snprintf . . . . .	157
4.19.2.12 sprintf . . . . .	157
4.19.2.13 sscanf . . . . .	157
4.19.2.14 strcat . . . . .	157
4.19.2.15 strchr . . . . .	157
4.19.2.16 strcmp . . . . .	157
4.19.2.17 strcpy . . . . .	158
4.19.2.18 strdup . . . . .	158
4.19.2.19 strlen . . . . .	158
4.19.2.20 strncmp . . . . .	158
4.19.2.21 strncpy . . . . .	158
4.19.2.22 strnlen . . . . .	158
4.19.2.23 strrchr . . . . .	158
4.19.2.24 strstr . . . . .	158
4.19.2.25 vsnprintf . . . . .	158
4.19.2.26 vsnprintf . . . . .	158

4.19.2.27 vsprintf . . . . .	158
4.19.2.28 vsscanf . . . . .	158
4.20 xen/extras/mini-os/include/minios_macros.h File Reference . . . . .	158
4.21 xen/stubdom/domt/integrated/minios_macros.h File Reference . . . . .	158
4.21.1 Macro Definition Documentation . . . . .	158
4.21.1.1 DEBUG . . . . .	158
4.21.1.2 DEBUG_ON . . . . .	159
4.21.1.3 DEBUG_TEST . . . . .	159
4.21.1.4 ERROR . . . . .	159
4.21.1.5 PANIC . . . . .	159
4.21.1.6 XENBUS_READ2 . . . . .	159
4.21.1.7 XENBUS_XPRINTF . . . . .	159
4.21.1.8 XENBUS_XREAD . . . . .	159
4.21.1.9 XENBUS_XREAD2 . . . . .	159
4.21.1.10 XENBUS_XREAD_INTEGER2 . . . . .	160
4.21.1.11 XENBUS_XSET_PERMS . . . . .	160
4.21.1.12 XENBUS_XWRITE . . . . .	160
4.21.1.13 XENBUS_XWRITE_INT . . . . .	160
4.22 xen/extras/mini-os/include/tpm_tis.h File Reference . . . . .	160
4.22.1 Macro Definition Documentation . . . . .	161
4.22.1.1 TPM_BASEADDR . . . . .	161
4.22.1.2 TPM_PROBE_IRQ . . . . .	161
4.22.1.3 TPM_TIS_EN_LOCL0 . . . . .	161
4.22.1.4 TPM_TIS_EN_LOCL1 . . . . .	161
4.22.1.5 TPM_TIS_EN_LOCL2 . . . . .	161
4.22.1.6 TPM_TIS_EN_LOCL3 . . . . .	161
4.22.1.7 TPM_TIS_EN_LOCL4 . . . . .	161
4.22.1.8 TPM_TIS_EN_LOCLALL . . . . .	161
4.22.1.9 TPM_TIS_LOCL_INT_TO_FLAG . . . . .	161
4.22.2 Function Documentation . . . . .	161
4.22.2.1 init_tpm_tis . . . . .	161
4.22.2.2 shutdown_tpm_tis . . . . .	161
4.22.2.3 tpm_tis_cmd . . . . .	161
4.22.2.4 tpm_tis_request_locality . . . . .	161
4.22.2.5 tpm_tis_transmit . . . . .	161
4.23 xen/extras/mini-os/include/tpmback.h File Reference . . . . .	161
4.23.1 Typedef Documentation . . . . .	162
4.23.1.1 tpmcmd_t . . . . .	162
4.23.2 Function Documentation . . . . .	162
4.23.2.1 init_tpmback . . . . .	162



4.23.2.2	<a href="#">shutdown_tpmback</a>	162
4.23.2.3	<a href="#">tpmback_get_uuid</a>	162
4.23.2.4	<a href="#">tpmback_num_frontends</a>	162
4.23.2.5	<a href="#">tpmback_req</a>	162
4.23.2.6	<a href="#">tpmback_req_any</a>	163
4.23.2.7	<a href="#">tpmback_resp</a>	163
4.23.2.8	<a href="#">tpmback_set_close_callback</a>	163
4.23.2.9	<a href="#">tpmback_set_open_callback</a>	163
4.23.2.10	<a href="#">tpmback_set_resume_callback</a>	163
4.23.2.11	<a href="#">tpmback_set_suspend_callback</a>	163
4.23.2.12	<a href="#">tpmback_wait_for_frontend_connect</a>	163
4.24	<a href="#">xen/extras/mini-os/include/tpmfront.h File Reference</a>	163
4.24.1	<a href="#">Function Documentation</a>	164
4.24.1.1	<a href="#">init_tpmfront</a>	164
4.24.1.2	<a href="#">shutdown_tpmfront</a>	164
4.24.1.3	<a href="#">tpmfront_cmd</a>	164
4.25	<a href="#">xen/extras/mini-os/include/uuid.h File Reference</a>	164
4.25.1	<a href="#">Typedef Documentation</a>	164
4.25.1.1	<a href="#">uuid_t</a>	164
4.25.2	<a href="#">Function Documentation</a>	164
4.25.2.1	<a href="#">uuid_create</a>	164
4.25.2.2	<a href="#">uuid_is_nil</a>	164
4.25.2.3	<a href="#">uuid_to_string</a>	164
4.26	<a href="#">xen/extras/mini-os/include/vfs.h File Reference</a>	164
4.27	<a href="#">xen/stubdom/domt/integrated/vfs.h File Reference</a>	164
4.27.1	<a href="#">Function Documentation</a>	165
4.27.1.1	<a href="#">vfs_command</a>	165
4.27.1.2	<a href="#">vfs_init</a>	165
4.28	<a href="#">xen/extras/mini-os/include/vfs_hooks.h File Reference</a>	165
4.29	<a href="#">xen/stubdom/domt/integrated/vfs_hooks.h File Reference</a>	165
4.29.1	<a href="#">Function Documentation</a>	165
4.29.1.1	<a href="#">vfs_close</a>	165
4.29.1.2	<a href="#">vfs_lseek</a>	165
4.29.1.3	<a href="#">vfs_major_minor</a>	165
4.29.1.4	<a href="#">vfs_mmap</a>	165
4.29.1.5	<a href="#">vfs_open</a>	165
4.29.1.6	<a href="#">vfs_read</a>	166
4.29.1.7	<a href="#">vfs_stat</a>	166
4.29.1.8	<a href="#">vfs_write</a>	166
4.30	<a href="#">xen/extras/mini-os/include/vfs_intf.h File Reference</a>	166

4.31	<a href="#">xen/stubdom/domt/integrated/vfs_intf.h File Reference</a>	166
4.31.1	Macro Definition Documentation	166
4.31.1.1	<a href="#">GREFS_PER_GREFMAP</a>	166
4.31.1.2	<a href="#">PAGE_SIZE</a>	166
4.31.1.3	<a href="#">VFS_O_CREAT</a>	166
4.31.1.4	<a href="#">VFS_O_RDONLY</a>	167
4.31.1.5	<a href="#">VFS_O_RDWR</a>	167
4.31.1.6	<a href="#">VFS_O_TRUNC</a>	167
4.31.1.7	<a href="#">VFS_O_WRONLY</a>	167
4.31.1.8	<a href="#">VFS_PATH</a>	167
4.31.2	Enumeration Type Documentation	167
4.31.2.1	<a href="#">vfs_cmdtype</a>	167
4.32	<a href="#">xen/tools/xenvfsd/include/vfs_intf.h File Reference</a>	167
4.33	<a href="#">xen/extras/mini-os/include/x86/arch_limits.h File Reference</a>	167
4.33.1	Macro Definition Documentation	167
4.33.1.1	<a href="#">__PAGE_SHIFT</a>	167
4.33.1.2	<a href="#">__PAGE_SIZE</a>	167
4.33.1.3	<a href="#">__STACK_SIZE</a>	167
4.33.1.4	<a href="#">__STACK_SIZE_PAGE_ORDER</a>	167
4.34	<a href="#">xen/extras/mini-os/kernel.c File Reference</a>	168
4.34.1	Macro Definition Documentation	168
4.34.1.1	<a href="#">DEPTH</a>	168
4.34.1.2	<a href="#">HEIGHT</a>	168
4.34.1.3	<a href="#">WIDTH</a>	168
4.34.2	Function Documentation	169
4.34.2.1	<a href="#">__attribute__</a>	169
4.34.2.2	<a href="#">do_exit</a>	169
4.34.2.3	<a href="#">setup_xen_features</a>	169
4.34.2.4	<a href="#">start_kernel</a>	169
4.34.2.5	<a href="#">stop_kernel</a>	169
4.34.2.6	<a href="#">test_xenbus</a>	169
4.34.3	Variable Documentation	169
4.34.3.1	<a href="#">xen_features</a>	169
4.35	<a href="#">xen/extras/mini-os/lib/sys.c File Reference</a>	169
4.35.1	Macro Definition Documentation	169
4.35.1.1	<a href="#">DEBUG</a>	169
4.36	<a href="#">xen/extras/mini-os/lib/xs.c File Reference</a>	169
4.37	<a href="#">xen/extras/mini-os/main.c File Reference</a>	169
4.38	<a href="#">xen/stubdom/domc/main.c File Reference</a>	170
4.38.1	Macro Definition Documentation	170

4.38.1.1	DIM	170
4.38.2	Function Documentation	170
4.38.2.1	main	170
4.38.2.2	wrap_notify_remote_via_evchn	170
4.38.3	Variable Documentation	170
4.38.3.1	grant_table	170
4.38.3.2	grant_table2	170
4.38.3.3	setup_op	171
4.39	xen/stubdom/domt/main.c File Reference	171
4.39.1	Function Documentation	171
4.39.1.1	main	171
4.40	xen/extras/mini-os/tpm_tis.c File Reference	171
4.40.1	Macro Definition Documentation	175
4.40.1.1	CAP_VERSION_1_1	175
4.40.1.2	CAP_VERSION_1_2	175
4.40.1.3	min	175
4.40.1.4	TPM_ACCESS	175
4.40.1.5	TPM_BUFSIZE	175
4.40.1.6	TPM_CAP_FLAG	175
4.40.1.7	TPM_CAP_FLAG_PERM	175
4.40.1.8	TPM_CAP_FLAG_VOL	175
4.40.1.9	TPM_CAP_PROP	175
4.40.1.10	TPM_CAP_PROP_MANUFACTURER	175
4.40.1.11	TPM_CAP_PROP_OWNER	175
4.40.1.12	TPM_CAP_PROP_PCR	175
4.40.1.13	TPM_CAP_PROP_TIS_DURATION	175
4.40.1.14	TPM_CAP_PROP_TIS_TIMEOUT	175
4.40.1.15	TPM_DATA_FIFO	175
4.40.1.16	TPM_DID_VID	175
4.40.1.17	TPM_DIGEST_SIZE	175
4.40.1.18	TPM_DIGEST_SIZE	175
4.40.1.19	TPM_ERROR_SIZE	175
4.40.1.20	TPM_HEADER_SIZE	175
4.40.1.21	TPM_INT_ENABLE	175
4.40.1.22	TPM_INT_STATUS	175
4.40.1.23	TPM_INT_VECTOR	175
4.40.1.24	TPM_INTERNAL_RESULT_SIZE	175
4.40.1.25	TPM_INTF_CAPS	175
4.40.1.26	TPM_MAX_ORDINAL	175
4.40.1.27	TPM_MAX_PROTECTED_ORDINAL	175

4.40.1.28	TPM_ORD_GET_CAP	176
4.40.1.29	TPM_PROTECTED_ORDINAL_MASK	176
4.40.1.30	TPM_RET_CODE_IDX	176
4.40.1.31	TPM_RID	176
4.40.1.32	TPM_STS	176
4.40.1.33	TPM_TAG_RQU_COMMAND	176
4.40.1.34	TPM_TIMEOUT	176
4.40.1.35	tpm_tis_cancel_cmd	176
4.40.2	Enumeration Type Documentation	176
4.40.2.1	tis_access	176
4.40.2.2	tis_defaults	176
4.40.2.3	tis_int_flags	176
4.40.2.4	tis_status	177
4.40.2.5	tpm_duration	177
4.40.3	Function Documentation	177
4.40.3.1	__attribute__	177
4.40.3.2	init_tpm_tis	177
4.40.3.3	MINIOS_LIST_HEAD	177
4.40.3.4	release_locality	177
4.40.3.5	shutdown_tpm_tis	177
4.40.3.6	tpm_calc_ordinal_duration	177
4.40.3.7	tpm_continue_selftest	177
4.40.3.8	tpm_get_timeouts	177
4.40.3.9	tpm_getcap	177
4.40.3.10	tpm_tis_cmd	177
4.40.3.11	tpm_tis_recv	177
4.40.3.12	tpm_tis_request_locality	177
4.40.3.13	tpm_tis_send	177
4.40.3.14	tpm_tis_transmit	178
4.40.4	Variable Documentation	178
4.40.4.1	__attribute__	178
4.40.4.2	a	178
4.40.4.3	algorithm	178
4.40.4.4	allowMaintenance	178
4.40.4.5	b	178
4.40.4.6	bGlobalLock	178
4.40.4.7	c	178
4.40.4.8	cap	178
4.40.4.9	cap_size	178
4.40.4.10	CEKPUse	178

4.40.4.11 checksum . . . . .	178
4.40.4.12 d . . . . .	178
4.40.4.13 deactivated . . . . .	178
4.40.4.14 disable . . . . .	178
4.40.4.15 disableForceClear . . . . .	178
4.40.4.16 disableFullDALogicInfo . . . . .	178
4.40.4.17 disableOwnerClear . . . . .	178
4.40.4.18 enableRevokeEK . . . . .	178
4.40.4.19 encscheme . . . . .	178
4.40.4.20 FIPS . . . . .	178
4.40.4.21 hash . . . . .	178
4.40.4.22 header . . . . .	178
4.40.4.23 keysize . . . . .	178
4.40.4.24 length . . . . .	178
4.40.4.25 maintenanceDone . . . . .	178
4.40.4.26 Major . . . . .	178
4.40.4.27 Minor . . . . .	179
4.40.4.28 modulus . . . . .	179
4.40.4.29 nvLocked . . . . .	179
4.40.4.30 operator . . . . .	179
4.40.4.31 ordinal . . . . .	179
4.40.4.32 ownership . . . . .	179
4.40.4.33 parameters . . . . .	179
4.40.4.34 params . . . . .	179
4.40.4.35 paramsize . . . . .	179
4.40.4.36 pcr_idx . . . . .	179
4.40.4.37 pcr_result . . . . .	179
4.40.4.38 physicalPresence . . . . .	179
4.40.4.39 physicalPresenceCMDEnable . . . . .	179
4.40.4.40 physicalPresenceHWEEnable . . . . .	179
4.40.4.41 physicalPresenceLifetimeLock . . . . .	179
4.40.4.42 physicalPresenceLock . . . . .	179
4.40.4.43 readPubek . . . . .	179
4.40.4.44 readSRKPub . . . . .	179
4.40.4.45 return_code . . . . .	179
4.40.4.46 revMajor . . . . .	179
4.40.4.47 revMinor . . . . .	179
4.40.4.48 sigscheme . . . . .	179
4.40.4.49 subcap . . . . .	179
4.40.4.50 subcap_size . . . . .	179

4.40.4.51 tag . . . . .	179
4.40.4.52 tpm_getcap_header . . . . .	179
4.40.4.53 tpm_long . . . . .	180
4.40.4.54 tpm_medium . . . . .	180
4.40.4.55 tpm_ordinal_duration . . . . .	180
4.40.4.56 tpm_protected_ordinal_duration . . . . .	180
4.40.4.57 tpm_short . . . . .	180
4.40.4.58 tpmEstablished . . . . .	180
4.40.4.59 TPMpost . . . . .	180
4.40.4.60 TPMpostLock . . . . .	180
4.41 xen/extras/mini-os/tpmback.c File Reference . . . . .	180
4.41.1 Macro Definition Documentation . . . . .	182
4.41.1.1 DEF_ARRAY_SIZE . . . . .	182
4.41.1.2 min . . . . .	182
4.41.1.3 strtoul . . . . .	182
4.41.1.4 TPMBACK_DEBUG . . . . .	182
4.41.1.5 TPMBACK_ERR . . . . .	182
4.41.1.6 TPMBACK_LOG . . . . .	182
4.41.1.7 TPMIF_CLOSED . . . . .	182
4.41.1.8 TPMIF_REQ_READY . . . . .	182
4.41.2 Typedef Documentation . . . . .	182
4.41.2.1 tpmback_dev_t . . . . .	182
4.41.2.2 tpmif_t . . . . .	182
4.41.3 Enumeration Type Documentation . . . . .	182
4.41.3.1 anonymous enum . . . . .	182
4.41.4 Function Documentation . . . . .	182
4.41.4.1 __free_tpmif . . . . .	182
4.41.4.2 __get_tpmif_index . . . . .	182
4.41.4.3 __init_tpmif . . . . .	182
4.41.4.4 connect_fe . . . . .	182
4.41.4.5 event_thread . . . . .	183
4.41.4.6 free_tpmif . . . . .	183
4.41.4.7 get_request . . . . .	183
4.41.4.8 get_tpmif . . . . .	183
4.41.4.9 get_tpmif_index . . . . .	183
4.41.4.10 handle_backend_event . . . . .	183
4.41.4.11 init_tpmback . . . . .	183
4.41.4.12 init_tpmcmd . . . . .	183
4.41.4.13 insert_tpmif . . . . .	183
4.41.4.14 new_tpmif . . . . .	183

4.41.4.15	<a href="#">remove_tpmif</a>	183
4.41.4.16	<a href="#">send_response</a>	183
4.41.4.17	<a href="#">shutdown_tpmback</a>	183
4.41.4.18	<a href="#">tpmback_get_uuid</a>	183
4.41.4.19	<a href="#">tpmback_handler</a>	184
4.41.4.20	<a href="#">tpmback_num_frontends</a>	184
4.41.4.21	<a href="#">tpmback_req</a>	184
4.41.4.22	<a href="#">tpmback_req_any</a>	184
4.41.4.23	<a href="#">tpmback_resp</a>	184
4.41.4.24	<a href="#">tpmback_set_close_callback</a>	184
4.41.4.25	<a href="#">tpmback_set_open_callback</a>	184
4.41.4.26	<a href="#">tpmback_set_resume_callback</a>	184
4.41.4.27	<a href="#">tpmback_set_suspend_callback</a>	184
4.41.4.28	<a href="#">tpmback_wait_for_frontend_connect</a>	184
4.41.4.29	<a href="#">tpmdev_check_req</a>	184
4.41.4.30	<a href="#">tpmif_change_state</a>	184
4.41.4.31	<a href="#">tpmif_req_finished</a>	185
4.41.4.32	<a href="#">tpmif_req_ready</a>	185
4.41.5	<a href="#">Variable Documentation</a>	185
4.41.5.1	<a href="#">globalinit</a>	185
4.41.5.2	<a href="#">tpm_ev_enum</a>	185
4.41.5.3	<a href="#">waitq</a>	185
4.42	<a href="#">xen/extras/mini-os/tpmfront.c File Reference</a>	185
4.42.1	<a href="#">Macro Definition Documentation</a>	185
4.42.1.1	<a href="#">min</a>	185
4.42.1.2	<a href="#">TPMFRONT_DEBUG</a>	185
4.42.1.3	<a href="#">TPMFRONT_ERR</a>	185
4.42.1.4	<a href="#">TPMFRONT_LOG</a>	185
4.42.2	<a href="#">Function Documentation</a>	185
4.42.2.1	<a href="#">init_tpmfront</a>	186
4.42.2.2	<a href="#">shutdown_tpmfront</a>	186
4.42.2.3	<a href="#">tpmfront_cmd</a>	186
4.42.2.4	<a href="#">tpmfront_handler</a>	186
4.42.2.5	<a href="#">tpmfront_recv</a>	186
4.42.2.6	<a href="#">tpmfront_send</a>	186
4.43	<a href="#">xen/extras/mini-os/uuid.c File Reference</a>	186
4.43.1	<a href="#">Function Documentation</a>	186
4.43.1.1	<a href="#">uuid_create</a>	186
4.43.1.2	<a href="#">uuid_is_nil</a>	186
4.43.1.3	<a href="#">uuid_to_string</a>	186

4.44	<a href="#">xen/extras/mini-os/vfs.c File Reference</a>	186
4.45	<a href="#">xen/stubdom/domt/integrated/vfs.c File Reference</a>	187
4.45.1	<a href="#">Macro Definition Documentation</a>	187
4.45.1.1	<a href="#">INVALID_GREF</a>	187
4.45.1.2	<a href="#">MIN</a>	187
4.45.1.3	<a href="#">NR_GRANT_ENTRIES</a>	188
4.45.1.4	<a href="#">NR_GRANT_FRAMES</a>	188
4.45.1.5	<a href="#">NR_RESERVED_ENTRIES</a>	188
4.45.2	<a href="#">Function Documentation</a>	188
4.45.2.1	<a href="#">vfs_close</a>	188
4.45.2.2	<a href="#">vfs_command</a>	188
4.45.2.3	<a href="#">vfs_init</a>	188
4.45.2.4	<a href="#">vfs_lseek</a>	188
4.45.2.5	<a href="#">vfs_major_minor</a>	188
4.45.2.6	<a href="#">vfs_mmap</a>	188
4.45.2.7	<a href="#">vfs_open</a>	188
4.45.2.8	<a href="#">vfs_read</a>	188
4.45.2.9	<a href="#">vfs_stat</a>	188
4.45.2.10	<a href="#">vfs_write</a>	188
4.45.3	<a href="#">Variable Documentation</a>	188
4.45.3.1	<a href="#">__attribute__</a>	188
4.46	<a href="#">xen/stubdom/domc/blkback.c File Reference</a>	189
4.46.1	<a href="#">Macro Definition Documentation</a>	189
4.46.1.1	<a href="#">BLK_RING_SIZE</a>	189
4.46.1.2	<a href="#">MAX_GNTMAP</a>	189
4.46.1.3	<a href="#">SECTOR_SIZE</a>	190
4.46.1.4	<a href="#">SECTS_IN_SEG</a>	190
4.46.1.5	<a href="#">SECTS_PER_PAGE</a>	190
4.46.2	<a href="#">Enumeration Type Documentation</a>	190
4.46.2.1	<a href="#">bridging</a>	190
4.46.3	<a href="#">Function Documentation</a>	190
4.46.3.1	<a href="#">blkback_aio_poll</a>	190
4.46.3.2	<a href="#">blkback_build_fwd</a>	190
4.46.3.3	<a href="#">blkback_cb_back</a>	190
4.46.3.4	<a href="#">blkback_monitor</a>	190
4.46.3.5	<a href="#">derive_blkback</a>	190
4.46.3.6	<a href="#">init_blkback</a>	190
4.46.3.7	<a href="#">shutdown_blkback</a>	190
4.47	<a href="#">xen/stubdom/domc/blkback.h File Reference</a>	190
4.47.1	<a href="#">Function Documentation</a>	191



4.47.1.1	<a href="#">blkback_aio_poll</a>	191
4.47.1.2	<a href="#">blkback_monitor</a>	191
4.47.1.3	<a href="#">derive_blkback</a>	191
4.47.1.4	<a href="#">init_blkback</a>	191
4.47.1.5	<a href="#">shutdown_blkback</a>	191
4.48	<a href="#">xen/stubdom/dmcc/crypt.c File Reference</a>	191
4.48.1	<a href="#">Macro Definition Documentation</a>	192
4.48.1.1	<a href="#">SECTOR_SIZE</a>	192
4.48.2	<a href="#">Enumeration Type Documentation</a>	192
4.48.2.1	<a href="#">crypt_dir</a>	192
4.48.3	<a href="#">Function Documentation</a>	192
4.48.3.1	<a href="#">decrypt_sectors</a>	192
4.48.3.2	<a href="#">encrypt_sectors</a>	192
4.48.3.3	<a href="#">shutdown_crypt</a>	192
4.48.3.4	<a href="#">startup_crypt</a>	192
4.48.3.5	<a href="#">test_crypt</a>	192
4.48.4	<a href="#">Variable Documentation</a>	192
4.48.4.1	<a href="#">cipherInst_iv</a>	192
4.48.4.2	<a href="#">keyInst_dec</a>	192
4.48.4.3	<a href="#">keyInst_enc</a>	192
4.48.4.4	<a href="#">keyInst_iv</a>	192
4.49	<a href="#">xen/stubdom/dmcc/crypt.h File Reference</a>	192
4.49.1	<a href="#">Function Documentation</a>	193
4.49.1.1	<a href="#">decrypt_sectors</a>	193
4.49.1.2	<a href="#">encrypt_sectors</a>	193
4.49.1.3	<a href="#">shutdown_crypt</a>	193
4.49.1.4	<a href="#">startup_crypt</a>	193
4.49.1.5	<a href="#">test_crypt</a>	193
4.50	<a href="#">xen/stubdom/dmcc/crypto/rijndael-alg-fst.c File Reference</a>	193
4.50.1	<a href="#">Macro Definition Documentation</a>	193
4.50.1.1	<a href="#">FULL_UNROLL</a>	193
4.50.1.2	<a href="#">GETU32</a>	194
4.50.1.3	<a href="#">PUTU32</a>	194
4.50.1.4	<a href="#">SWAP</a>	194
4.50.2	<a href="#">Function Documentation</a>	194
4.50.2.1	<a href="#">rijndaelDecrypt</a>	194
4.50.2.2	<a href="#">rijndaelEncrypt</a>	194
4.50.2.3	<a href="#">rijndaelKeySetupDec</a>	194
4.50.2.4	<a href="#">rijndaelKeySetupEnc</a>	194
4.51	<a href="#">xen/stubdom/dmcc/crypto/rijndael-alg-fst.h File Reference</a>	194

4.51.1	Macro Definition Documentation	195
4.51.1.1	MAXKB	195
4.51.1.2	MAXKC	195
4.51.1.3	MAXNR	195
4.51.2	Typedef Documentation	195
4.51.2.1	u16	195
4.51.2.2	u32	195
4.51.2.3	u8	195
4.51.3	Function Documentation	196
4.51.3.1	rijndaelDecrypt	196
4.51.3.2	rijndaelEncrypt	196
4.51.3.3	rijndaelKeySetupDec	196
4.51.3.4	rijndaelKeySetupEnc	196
4.52	xen/stubdom/domc/crypto/rijndael-api-fst.c File Reference	196
4.52.1	Function Documentation	196
4.52.1.1	blockDecrypt	196
4.52.1.2	blockEncrypt	196
4.52.1.3	cipherInit	196
4.52.1.4	makeKey	196
4.52.1.5	padDecrypt	197
4.52.1.6	padEncrypt	197
4.53	xen/stubdom/domc/crypto/rijndael-api-fst.h File Reference	197
4.53.1	Macro Definition Documentation	198
4.53.1.1	BAD_BLOCK_LENGTH	198
4.53.1.2	BAD_CIPHER_INSTANCE	198
4.53.1.3	BAD_CIPHER_MODE	198
4.53.1.4	BAD_CIPHER_STATE	198
4.53.1.5	BAD_DATA	198
4.53.1.6	BAD_KEY_DIR	198
4.53.1.7	BAD_KEY_INSTANCE	198
4.53.1.8	BAD_KEY_MAT	198
4.53.1.9	BAD_OTHER	199
4.53.1.10	BITSPERBLOCK	199
4.53.1.11	DIR_DECRYPT	199
4.53.1.12	DIR_ENCRYPT	199
4.53.1.13	FALSE	199
4.53.1.14	MAX_IV_SIZE	199
4.53.1.15	MAX_KEY_SIZE	199
4.53.1.16	MODE_CBC	199
4.53.1.17	MODE_CFB1	199

4.53.1.18	MODE_ECB	199
4.53.1.19	TRUE	199
4.53.2	Typedef Documentation	199
4.53.2.1	BYTE	199
4.53.3	Function Documentation	200
4.53.3.1	blockDecrypt	200
4.53.3.2	blockEncrypt	200
4.53.3.3	cipherInit	200
4.53.3.4	makeKey	200
4.53.3.5	padDecrypt	200
4.53.3.6	padEncrypt	200
4.54	xen/stubdom/domc/crypto/sha2.c File Reference	201
4.54.1	Macro Definition Documentation	202
4.54.1.1	ADDINC128	202
4.54.1.2	Ch	202
4.54.1.3	Maj	202
4.54.1.4	MEMCPY_BCOPY	202
4.54.1.5	MEMSET_BZERO	202
4.54.1.6	R	202
4.54.1.7	REVERSE32	202
4.54.1.8	REVERSE64	202
4.54.1.9	S32	203
4.54.1.10	S64	203
4.54.1.11	SHA256_SHORT_BLOCK_LENGTH	203
4.54.1.12	SHA2_USE_MEMSET_MEMCPY	203
4.54.1.13	SHA384_SHORT_BLOCK_LENGTH	203
4.54.1.14	SHA512_SHORT_BLOCK_LENGTH	203
4.54.1.15	Sigma0_256	203
4.54.1.16	sigma0_256	203
4.54.1.17	Sigma0_512	203
4.54.1.18	sigma0_512	203
4.54.1.19	Sigma1_256	203
4.54.1.20	sigma1_256	203
4.54.1.21	Sigma1_512	203
4.54.1.22	sigma1_512	203
4.54.2	Typedef Documentation	203
4.54.2.1	sha2_byte	203
4.54.2.2	sha2_word32	203
4.54.2.3	sha2_word64	203
4.54.3	Function Documentation	203

4.54.3.1	SHA256_Data . . . . .	203
4.54.3.2	SHA256_End . . . . .	203
4.54.3.3	SHA256_Final . . . . .	203
4.54.3.4	SHA256_Init . . . . .	203
4.54.3.5	SHA256_Transform . . . . .	203
4.54.3.6	SHA256_Update . . . . .	203
4.54.3.7	SHA384_Data . . . . .	203
4.54.3.8	SHA384_End . . . . .	203
4.54.3.9	SHA384_Final . . . . .	203
4.54.3.10	SHA384_Init . . . . .	204
4.54.3.11	SHA384_Update . . . . .	204
4.54.3.12	SHA512_Data . . . . .	204
4.54.3.13	SHA512_End . . . . .	204
4.54.3.14	SHA512_Final . . . . .	204
4.54.3.15	SHA512_Init . . . . .	204
4.54.3.16	SHA512_Last . . . . .	204
4.54.3.17	SHA512_Transform . . . . .	204
4.54.3.18	SHA512_Update . . . . .	204
4.55	xen/stubdom/domc/crypto/sha2.h File Reference . . . . .	204
4.55.1	Macro Definition Documentation . . . . .	205
4.55.1.1	BYTE_ORDER . . . . .	205
4.55.1.2	LITTLE_ENDIAN . . . . .	205
4.55.1.3	SHA256_BLOCK_LENGTH . . . . .	205
4.55.1.4	SHA256_DIGEST_LENGTH . . . . .	205
4.55.1.5	SHA256_DIGEST_STRING_LENGTH . . . . .	205
4.55.1.6	SHA2_USE_INTTYPES_H . . . . .	205
4.55.1.7	SHA384_BLOCK_LENGTH . . . . .	205
4.55.1.8	SHA384_DIGEST_LENGTH . . . . .	205
4.55.1.9	SHA384_DIGEST_STRING_LENGTH . . . . .	205
4.55.1.10	SHA512_BLOCK_LENGTH . . . . .	205
4.55.1.11	SHA512_DIGEST_LENGTH . . . . .	205
4.55.1.12	SHA512_DIGEST_STRING_LENGTH . . . . .	205
4.55.2	Typedef Documentation . . . . .	205
4.55.2.1	SHA256_CTX . . . . .	205
4.55.2.2	SHA384_CTX . . . . .	206
4.55.2.3	SHA512_CTX . . . . .	206
4.55.3	Function Documentation . . . . .	206
4.55.3.1	SHA256_Data . . . . .	206
4.55.3.2	SHA256_End . . . . .	206
4.55.3.3	SHA256_Final . . . . .	206

4.55.3.4	SHA256_Init	206
4.55.3.5	SHA256_Update	206
4.55.3.6	SHA384_Data	206
4.55.3.7	SHA384_End	206
4.55.3.8	SHA384_Final	206
4.55.3.9	SHA384_Init	206
4.55.3.10	SHA384_Update	206
4.55.3.11	SHA512_Data	206
4.55.3.12	SHA512_End	206
4.55.3.13	SHA512_Final	206
4.55.3.14	SHA512_Init	206
4.55.3.15	SHA512_Update	206
4.56	xen/stubdom/domc/cspipe_intf.h File Reference	206
4.57	xen/stubdom/domt/cspipe_intf.h File Reference	206
4.57.1	Function Documentation	207
4.57.1.1	__attribute__	207
4.57.2	Variable Documentation	207
4.57.2.1	domid_domc	207
4.57.2.2	domid_domu	207
4.57.2.3	magic_nr	207
4.57.2.4	s_key	207
4.58	xen/stubdom/domc/main.h File Reference	207
4.58.1	Typedef Documentation	207
4.58.1.1	domc_info_t	207
4.58.2	Function Documentation	207
4.58.2.1	wrap_notify_remote_via_evtchn	207
4.59	xen/stubdom/domt/main.h File Reference	208
4.60	xen/stubdom/domt/caas.c File Reference	208
4.60.1	Function Documentation	208
4.60.1.1	caas_domain_create	208
4.60.1.2	caas_init	208
4.60.1.3	caas_setup_cspipe	208
4.60.2	Variable Documentation	209
4.60.2.1	tc	209
4.61	xen/stubdom/domt/caas.h File Reference	209
4.61.1	Function Documentation	209
4.61.1.1	__attribute__	209
4.61.1.2	caas_domain_create	209
4.61.1.3	caas_init	209
4.61.1.4	caas_setup_cspipe	209

4.61.2	Variable Documentation	210
4.61.2.1	s_key	210
4.62	xen/stubdom/domt/crypto/hmac.c File Reference	210
4.62.1	Function Documentation	210
4.62.1.1	tpm_hmac_final	210
4.62.1.2	tpm_hmac_init	210
4.62.1.3	tpm_hmac_update	210
4.63	xen/stubdom/domt/crypto/hmac.h File Reference	210
4.63.1	Macro Definition Documentation	211
4.63.1.1	HMAC_PAD_LENGTH	211
4.63.2	Function Documentation	211
4.63.2.1	tpm_hmac_final	211
4.63.2.2	tpm_hmac_init	211
4.63.2.3	tpm_hmac_update	211
4.63.3	Variable Documentation	211
4.63.3.1	__attribute__	211
4.64	xen/stubdom/domt/crypto/sha1.c File Reference	211
4.64.1	Macro Definition Documentation	211
4.64.1.1	B0	211
4.64.1.2	B1	211
4.64.1.3	F0	212
4.64.1.4	F1	212
4.64.1.5	F2	212
4.64.1.6	R0	212
4.64.1.7	R1	212
4.64.1.8	R2	212
4.64.1.9	R3	212
4.64.1.10	R4	212
4.64.1.11	rol	212
4.64.2	Function Documentation	212
4.64.2.1	tpm_sha1_final	212
4.64.2.2	tpm_sha1_init	212
4.64.2.3	tpm_sha1_update	212
4.65	xen/stubdom/domt/crypto/sha1.h File Reference	212
4.65.1	Macro Definition Documentation	213
4.65.1.1	CPU_TO_BE32	213
4.65.1.2	SHA1_DIGEST_LENGTH	213
4.65.2	Function Documentation	213
4.65.2.1	tpm_sha1_final	213
4.65.2.2	tpm_sha1_init	213

4.65.2.3	tpm_sha1_update	213
4.66	xen/stubdom/domt/tc.c File Reference	213
4.66.1	Macro Definition Documentation	214
4.66.1.1	FILL_CONVERT	214
4.66.1.2	TC_CHECK_BUF_OVERFLOW	214
4.66.1.3	TC_CHECK_DIGESTS	214
4.66.1.4	TC_CHECK_IF_ALL_READ	215
4.66.1.5	TC_CHECK_RC_ERROR	215
4.66.1.6	TC_CHECK_TPM_TIS_CMD	215
4.66.1.7	TC_OPT_INVALIDATE_OIAP	215
4.66.1.8	TC_OPT_INVALIDATE_OSAP	215
4.66.1.9	tc_place16	216
4.66.1.10	tc_place16_h	216
4.66.1.11	tc_place16_s	216
4.66.1.12	tc_place32	216
4.66.1.13	tc_place32_h	216
4.66.1.14	tc_place32_s	216
4.66.1.15	tc_place	216
4.66.1.16	tc_place_h	216
4.66.1.17	tc_place_s	216
4.66.1.18	TC_PLACE_SZ	216
4.66.1.19	TPM_DEFAULT_ENCSIZE	216
4.66.1.20	TPM_DEFAULT_PKEYLEN	216
4.66.2	Function Documentation	216
4.66.2.1	init_oiap	216
4.66.2.2	init_osap	216
4.66.2.3	init_tc	217
4.66.2.4	shutdown_tc	217
4.66.2.5	tc_chk_loadkey2	217
4.66.2.6	tc_create_key	217
4.66.2.7	tc_createwrapkey	217
4.66.2.8	tc_flush_all_keys	217
4.66.2.9	tc_flushspecific	217
4.66.2.10	tc_free_key	217
4.66.2.11	tc_hexdump	217
4.66.2.12	tc_listkeys	217
4.66.2.13	tc_loadkey2	217
4.66.2.14	tc_print_pcrs	217
4.66.2.15	tc_reset	217
4.66.2.16	tc_test	217

4.66.2.17	<a href="#">tc_transmit_cmd</a>	217
4.66.2.18	<a href="#">tc_unbind</a>	217
4.66.2.19	<a href="#">tpm_get_random</a>	217
4.66.2.20	<a href="#">tpm_pcr_extend</a>	218
4.67	<a href="#">xen/stubdom/domt/tc.h File Reference</a>	218
4.67.1	<a href="#">Macro Definition Documentation</a>	219
4.67.1.1	<a href="#">TC_PATH_KEYDATA</a>	219
4.67.1.2	<a href="#">TC_PATH_PREFIX</a>	219
4.67.1.3	<a href="#">TC_PATH_PUBKEY</a>	219
4.67.1.4	<a href="#">TPM_LEN</a>	219
4.67.1.5	<a href="#">TPM_LOCS</a>	219
4.67.1.6	<a href="#">TPM_MFN</a>	219
4.67.1.7	<a href="#">TPM_NUM_PCR</a>	219
4.67.2	<a href="#">Function Documentation</a>	219
4.67.2.1	<a href="#">init_oiap</a>	219
4.67.2.2	<a href="#">init_osap</a>	219
4.67.2.3	<a href="#">init_tc</a>	219
4.67.2.4	<a href="#">shutdown_tc</a>	219
4.67.2.5	<a href="#">tc_chk_loadkey2</a>	219
4.67.2.6	<a href="#">tc_create_key</a>	219
4.67.2.7	<a href="#">tc_createwrapkey</a>	219
4.67.2.8	<a href="#">tc_flush_all_keys</a>	219
4.67.2.9	<a href="#">tc_flushspecific</a>	219
4.67.2.10	<a href="#">tc_free_key</a>	219
4.67.2.11	<a href="#">tc_hexdump</a>	219
4.67.2.12	<a href="#">tc_listkeys</a>	219
4.67.2.13	<a href="#">tc_loadkey2</a>	220
4.67.2.14	<a href="#">tc_print_pcrs</a>	220
4.67.2.15	<a href="#">tc_reset</a>	220
4.67.2.16	<a href="#">tc_test</a>	220
4.67.2.17	<a href="#">tc_transmit_cmd</a>	220
4.67.2.18	<a href="#">tc_unbind</a>	220
4.67.2.19	<a href="#">tpm_get_random</a>	220
4.67.2.20	<a href="#">tpm_pcr_extend</a>	220
4.68	<a href="#">xen/stubdom/domt/tpm_defs.c File Reference</a>	220
4.68.1	<a href="#">Variable Documentation</a>	220
4.68.1.1	<a href="#">flushspecific_header</a>	220
4.68.1.2	<a href="#">getrandom_header</a>	221
4.68.1.3	<a href="#">oiap_header</a>	221
4.68.1.4	<a href="#">osap_header</a>	221



4.68.1.5	pcextend_header	221
4.68.1.6	pcrread_header	221
4.68.1.7	reset_header	221
4.68.1.8	tpm_readpubek_header	222
4.69	xen/stubdom/domt/tpm_defs.h File Reference	222
4.69.1	Macro Definition Documentation	225
4.69.1.1	_TPM_ALG_RSA	225
4.69.1.2	_TPM_ES_RSAESOAEP_SHA1_MGF1	225
4.69.1.3	_TPM_ET_SRK	225
4.69.1.4	_TPM_KEY_BIND	225
4.69.1.5	_TPM_KH_SRK	225
4.69.1.6	_TPM_ORD_FLUSHSPECIFIC	225
4.69.1.7	_TPM_ORD_GET_CAP	225
4.69.1.8	_TPM_ORD_GETRANDOM	225
4.69.1.9	_TPM_ORD_OIAP	225
4.69.1.10	_TPM_ORD_OSAP	225
4.69.1.11	_TPM_ORD_PCR_EXTEND	225
4.69.1.12	_TPM_ORD_READPUBEK	225
4.69.1.13	_TPM_ORD_RESET	225
4.69.1.14	_TPM_ORDINAL_PCRREAD	225
4.69.1.15	_TPM_SS_NONE	225
4.69.1.16	_TPM_TAG_KEY12	225
4.69.1.17	_TPM_TAG_RQU_AUTH1_COMMAND	225
4.69.1.18	_TPM_TAG_RQU_AUTH2_COMMAND	225
4.69.1.19	_TPM_TAG_RQU_COMMAND	225
4.69.1.20	MAX_RAND_SIZE	225
4.69.1.21	TPM_ALG_RSA	225
4.69.1.22	TPM_AUTH_ALWAYS	225
4.69.1.23	TPM_BUFSIZE	225
4.69.1.24	TPM_CAP_HANDLE	225
4.69.1.25	TPM_CAP_KEY_HANDLE	225
4.69.1.26	TPM_DIGEST_SIZE	225
4.69.1.27	TPM_ERROR_SIZE	225
4.69.1.28	TPM_ES_RSAESOAEP_SHA1_MGF1	226
4.69.1.29	TPM_ET_SRK	226
4.69.1.30	TPM_HEADER_SIZE	226
4.69.1.31	TPM_INTERNAL_RESULT_SIZE	226
4.69.1.32	TPM_KEY12_EMPTYSIZE	226
4.69.1.33	TPM_KEY_BIND	226
4.69.1.34	TPM_KEY_PARMS_EMPTYSIZE	226

4.69.1.35	TPM_KEY_STORAGE	226
4.69.1.36	TPM_KH_SRK	226
4.69.1.37	TPM_NOSPACE	226
4.69.1.38	TPM_ORD_CREATEWRAPKEY	226
4.69.1.39	TPM_ORD_GETCAPABILITY	226
4.69.1.40	TPM_ORD_LOADKEY2	226
4.69.1.41	TPM_ORD_UNBIND	226
4.69.1.42	TPM_RET_CODE_IDX	226
4.69.1.43	TPM_RSA_KEY_EMPTYSIZE	226
4.69.1.44	TPM_RT_AUTH	226
4.69.1.45	TPM_RT_CONTEXT	226
4.69.1.46	TPM_RT_COUNTER	226
4.69.1.47	TPM_RT_DAA_TPM	226
4.69.1.48	TPM_RT_DAA_V0	226
4.69.1.49	TPM_RT_DAA_V1	226
4.69.1.50	TPM_RT_DELEGATE	226
4.69.1.51	TPM_RT_HASH	226
4.69.1.52	TPM_RT_KEY	226
4.69.1.53	TPM_RT_TRANS	226
4.69.1.54	TPM_SS_NONE	226
4.69.1.55	TPM_STORE_PUBKEY_EMPTYSIZE	226
4.69.1.56	TPM_TAG_KEY12	227
4.69.1.57	TPM_TAG_RQU_AUTH1_COMMAND	227
4.69.1.58	TPM_TAG_RQU_AUTH2_COMMAND	227
4.69.1.59	TPM_TAG_RQU_COMMAND	227
4.69.1.60	TPM_TAG_RSP_AUTH1_COMMAND	227
4.69.1.61	TPM_TAG_RSP_AUTH2_COMMAND	227
4.69.1.62	TPM_TAG_RSP_COMMAND	227
4.69.2	Enumeration Type Documentation	227
4.69.2.1	tpm_duration	227
4.69.3	Function Documentation	227
4.69.3.1	__attribute__	227
4.69.4	Variable Documentation	227
4.69.4.1	__attribute__	227
4.69.4.2	algorithm	227
4.69.4.3	authHandle	227
4.69.4.4	authhandle	227
4.69.4.5	checksum	227
4.69.4.6	encscheme	227
4.69.4.7	entity_type	227

4.69.4.8	entity_value	227
4.69.4.9	flushspecific_header	227
4.69.4.10	getrandom_header	227
4.69.4.11	handle	228
4.69.4.12	hash	228
4.69.4.13	header	228
4.69.4.14	keysize	228
4.69.4.15	length	228
4.69.4.16	modulus	228
4.69.4.17	nonce_even	228
4.69.4.18	nonce_even_osap	228
4.69.4.19	nonce_odd_osap	228
4.69.4.20	nonceEven	228
4.69.4.21	numbytes	228
4.69.4.22	numrandbytes	228
4.69.4.23	oiap_header	228
4.69.4.24	ordinal	228
4.69.4.25	osap_header	228
4.69.4.26	parameters	228
4.69.4.27	params	228
4.69.4.28	paramsize	228
4.69.4.29	pcr_idx	228
4.69.4.30	pcr_result	228
4.69.4.31	pcrextend_header	228
4.69.4.32	pcrread_header	228
4.69.4.33	randbytes	228
4.69.4.34	reset_header	228
4.69.4.35	resourceType	228
4.69.4.36	return_code	228
4.69.4.37	sigscheme	228
4.69.4.38	tag	228
4.69.4.39	tpm_readpubek_header	229
4.70	xen/stubdom/domt/xlc.c File Reference	229
4.70.1	Function Documentation	229
4.70.1.1	do_create	229
4.70.1.2	xlc_init	229
4.70.1.3	xlc_listen	229
4.71	xen/tools/libxlc/xlc.c File Reference	229
4.71.1	Macro Definition Documentation	230
4.71.1.1	DOMID_T	230

4.71.1.2	TIMEOUT_SEC	230
4.71.1.3	TIMEOUT_SEC_INIT	230
4.71.2	Function Documentation	230
4.71.2.1	print_libxl_domain_config	230
4.71.2.2	read_dyn_domid	230
4.71.2.3	xl_command	230
4.71.2.4	xl_do_domain_create	230
4.71.2.5	xl_domain_build	230
4.71.2.6	xl_domain_make	230
4.72	xen/stubdom/domt/xlc.h File Reference	230
4.72.1	Function Documentation	231
4.72.1.1	do_create	231
4.72.1.2	xl_init	231
4.72.1.3	xl_listen	231
4.73	xen/tools/libxlc/xlc.h File Reference	231
4.73.1	Macro Definition Documentation	231
4.73.1.1	XLC_FAILED	231
4.73.2	Function Documentation	231
4.73.2.1	xl_do_domain_create	231
4.73.2.2	xl_domain_build	231
4.73.2.3	xl_domain_make	231
4.74	xen/stubdom/domt/xlc_intf.h File Reference	232
4.74.1	Macro Definition Documentation	232
4.74.1.1	XLC_PATH	232
4.74.2	Enumeration Type Documentation	232
4.74.2.1	xl_cmdtype	232
4.75	xen/tools/libxlc/include/xlc_intf.h File Reference	232
4.76	xen/stubdom/domt/xlc_util.c File Reference	232
4.76.1	Macro Definition Documentation	233
4.76.1.1	CHK_MEMCMP	233
4.76.1.2	CHK_STRCMP	233
4.76.1.3	CHK_XLC_DARRAY_IN	233
4.76.1.4	CHK_XLC_DARRAY_OUT	233
4.76.1.5	CHK_XLC_DESERIALIZE_DISKS	233
4.76.1.6	CHK_XLC_DESERIALIZE_VIFS	233
4.76.1.7	CHK_XLC_MEMCPY_IN	233
4.76.1.8	CHK_XLC_MEMCPY_OUT	233
4.76.1.9	CHK_XLC_SERIALIZE_DISKS	233
4.76.1.10	CHK_XLC_SERIALIZE_VIFS	233
4.76.1.11	CHK_XLC_STRACPY_IN	233

4.76.1.12	CHK_XLC_STRCPY_OUT . . . . .	233
4.76.2	Function Documentation . . . . .	233
4.76.2.1	deserialize_domain_build_info . . . . .	234
4.76.2.2	deserialize_domain_config . . . . .	234
4.76.2.3	deserialize_domain_create_info . . . . .	234
4.76.2.4	serialize_domain_build_info . . . . .	234
4.76.2.5	serialize_domain_config . . . . .	234
4.76.2.6	serialize_domain_create_info . . . . .	234
4.77	xen/tools/libxlc/xlc_util.c File Reference . . . . .	234
4.78	xen/stubdom/domt/xlc_util.h File Reference . . . . .	234
4.78.1	Function Documentation . . . . .	234
4.78.1.1	deserialize_domain_build_info . . . . .	234
4.78.1.2	deserialize_domain_config . . . . .	234
4.78.1.3	deserialize_domain_create_info . . . . .	234
4.78.1.4	serialize_domain_build_info . . . . .	235
4.78.1.5	serialize_domain_config . . . . .	235
4.78.1.6	serialize_domain_create_info . . . . .	235
4.79	xen/tools/libxlc/xlc_util.h File Reference . . . . .	235
4.80	xen/stubdom/vtpm/compat/big_endian.h File Reference . . . . .	235
4.80.1	Macro Definition Documentation . . . . .	235
4.80.1.1	__be16_to_cpu . . . . .	235
4.80.1.2	__be32_to_cpu . . . . .	235
4.80.1.3	__be64_to_cpu . . . . .	235
4.80.1.4	__cpu_to_be16 . . . . .	235
4.80.1.5	__cpu_to_be32 . . . . .	235
4.80.1.6	__cpu_to_be64 . . . . .	235
4.80.1.7	__cpu_to_le16 . . . . .	235
4.80.1.8	__cpu_to_le32 . . . . .	235
4.80.1.9	__cpu_to_le64 . . . . .	235
4.80.1.10	__le16_to_cpu . . . . .	235
4.80.1.11	__le32_to_cpu . . . . .	235
4.80.1.12	__le64_to_cpu . . . . .	235
4.81	xen/stubdom/vtpm/compat/endian_test.c File Reference . . . . .	236
4.81.1	Function Documentation . . . . .	236
4.81.1.1	main . . . . .	236
4.82	xen/stubdom/vtpm/compat/little_endian.h File Reference . . . . .	236
4.82.1	Macro Definition Documentation . . . . .	236
4.82.1.1	__be16_to_cpu . . . . .	236
4.82.1.2	__be32_to_cpu . . . . .	236
4.82.1.3	__be64_to_cpu . . . . .	236

4.82.1.4	<a href="#">__cpu_to_be16</a>	236
4.82.1.5	<a href="#">__cpu_to_be32</a>	236
4.82.1.6	<a href="#">__cpu_to_be64</a>	236
4.82.1.7	<a href="#">__cpu_to_le16</a>	236
4.82.1.8	<a href="#">__cpu_to_le32</a>	236
4.82.1.9	<a href="#">__cpu_to_le64</a>	236
4.82.1.10	<a href="#">__le16_to_cpu</a>	236
4.82.1.11	<a href="#">__le32_to_cpu</a>	237
4.82.1.12	<a href="#">__le64_to_cpu</a>	237
4.83	<a href="#">xen/stubdom/vtpm/compat/swab.h File Reference</a>	237
4.83.1	Macro Definition Documentation	237
4.83.1.1	<a href="#">__swab16</a>	237
4.83.1.2	<a href="#">__swab32</a>	237
4.83.1.3	<a href="#">__swab64</a>	237
4.84	<a href="#">xen/stubdom/vtpm/vtpm.c File Reference</a>	238
4.84.1	Function Documentation	238
4.84.1.1	<a href="#">get_rand_bytes_rand</a>	238
4.84.1.2	<a href="#">init_random</a>	238
4.84.1.3	<a href="#">main</a>	238
4.84.1.4	<a href="#">parse_cmd_line</a>	238
4.84.1.5	<a href="#">tpm_get_extern_random_bytes</a>	238
4.84.1.6	<a href="#">tpm_get_ticks</a>	238
4.84.1.7	<a href="#">tpm_log</a>	238
4.84.1.8	<a href="#">tpm_read_from_file</a>	238
4.84.1.9	<a href="#">tpm_write_to_file</a>	239
4.84.2	Variable Documentation	239
4.84.2.1	<a href="#">opt_args</a>	239
4.84.2.2	<a href="#">tpmfront_dev</a>	239
4.85	<a href="#">xen/stubdom/vtpm/vtpm.h File Reference</a>	239
4.85.1	Macro Definition Documentation	239
4.85.1.1	<a href="#">VERS_CMD</a>	239
4.85.1.2	<a href="#">VERS_CMD_LEN</a>	239
4.85.2	Variable Documentation	239
4.85.2.1	<a href="#">opt_args</a>	239
4.86	<a href="#">xen/stubdom/vtpm/vtpm_cmd.c File Reference</a>	240
4.86.1	Macro Definition Documentation	240
4.86.1.1	<a href="#">CHECKSTATUSGOTO</a>	240
4.86.1.2	<a href="#">ERR_MALFORMED</a>	240
4.86.1.3	<a href="#">ERR_TPMFRONT</a>	240
4.86.1.4	<a href="#">TRYFAILGOTO</a>	240

4.86.1.5	TRYFAILGOTOMSG . . . . .	241
4.86.2	Typedef Documentation . . . . .	241
4.86.2.1	shpage_t . . . . .	241
4.86.3	Function Documentation . . . . .	241
4.86.3.1	VTPM_GetRandom . . . . .	241
4.86.3.2	VTPM_LoadKey . . . . .	241
4.86.3.3	VTPM_SaveKey . . . . .	241
4.87	xen/stubdom/vtpm/vtpm_cmd.h File Reference . . . . .	241
4.87.1	Function Documentation . . . . .	242
4.87.1.1	VTPM_GetNVMSize . . . . .	242
4.87.1.2	VTPM_GetRandom . . . . .	242
4.87.1.3	VTPM_LoadKey . . . . .	242
4.87.1.4	VTPM_LoadNVM . . . . .	242
4.87.1.5	VTPM_SaveKey . . . . .	242
4.87.1.6	VTPM_SaveNVM . . . . .	242
4.88	xen/stubdom/vtpm/vtpmblk.c File Reference . . . . .	242
4.88.1	Macro Definition Documentation . . . . .	243
4.88.1.1	BLKSZ . . . . .	243
4.88.1.2	KEYSZ . . . . .	243
4.88.1.3	read_data . . . . .	243
4.88.1.4	read_finish . . . . .	243
4.88.1.5	SECTORS_PER_BLOCK . . . . .	243
4.88.1.6	write_data . . . . .	243
4.88.1.7	write_finish . . . . .	243
4.88.2	Function Documentation . . . . .	243
4.88.2.1	decrypt_vtpmblk . . . . .	243
4.88.2.2	encrypt_vtpmblk . . . . .	243
4.88.2.3	init_vtpmblk . . . . .	243
4.88.2.4	read_vtpmblk . . . . .	243
4.88.2.5	read_vtpmblk_raw . . . . .	244
4.88.2.6	shutdown_vtpmblk . . . . .	244
4.88.2.7	write_vtpmblk . . . . .	244
4.88.2.8	write_vtpmblk_raw . . . . .	244
4.88.3	Variable Documentation . . . . .	244
4.88.3.1	aiocb . . . . .	244
4.88.3.2	blkdev . . . . .	244
4.88.3.3	blkinfo . . . . .	244
4.88.3.4	blkoff . . . . .	244
4.88.3.5	blocksize . . . . .	244
4.88.3.6	disksize . . . . .	244

4.89	xen/stubdom/vtpm/vtpmbk.h File Reference	244
4.89.1	Function Documentation	244
4.89.1.1	init_vtpmbk	244
4.89.1.2	read_vtpmbk	244
4.89.1.3	shutdown_vtpmbk	244
4.89.1.4	write_vtpmbk	244
4.90	xen/tools/console/daemon/io.c File Reference	245
4.90.1	Macro Definition Documentation	246
4.90.1.1	_GNU_SOURCE	246
4.90.1.2	DOMID_DOMT	246
4.90.1.3	MAX	246
4.90.1.4	MAX_STRLEN	246
4.90.1.5	MIN	246
4.90.1.6	RATE_LIMIT_ALLOWANCE	246
4.90.1.7	RATE_LIMIT_PERIOD	246
4.90.2	Function Documentation	246
4.90.2.1	enum_domains	246
4.90.2.2	handle_io	246
4.90.3	Variable Documentation	246
4.90.3.1	discard_overflowed_data	246
4.90.3.2	log_dir	246
4.90.3.3	log_guest	246
4.90.3.4	log_hv	246
4.90.3.5	log_reload	246
4.90.3.6	log_time_guest	246
4.90.3.7	log_time_hv	246
4.91	xen/tools/debugger/kdd/.kdd-xen.o.d File Reference	246
4.92	xen/tools/debugger/kdd/.kdd.o.d File Reference	246
4.93	xen/tools/ioemu-dir/alpha-linux-user/config.h File Reference	246
4.93.1	Macro Definition Documentation	247
4.93.1.1	CONFIG_LINUX_USER	247
4.93.1.2	CONFIG_QEMU_PREFIX	247
4.93.1.3	CONFIG_USER_ONLY	247
4.93.1.4	TARGET_ALPHA	247
4.93.1.5	TARGET_ARCH	247
4.94	xen/tools/ioemu-dir/arm-linux-user/config.h File Reference	247
4.94.1	Macro Definition Documentation	247
4.94.1.1	CONFIG_LINUX_USER	247
4.94.1.2	CONFIG_QEMU_PREFIX	247
4.94.1.3	CONFIG_SOFTFLOAT	247



4.94.1.4	CONFIG_USER_ONLY	247
4.94.1.5	TARGET_ARCH	247
4.94.1.6	TARGET_ARM	247
4.94.1.7	TARGET_HAS_BFLT	247
4.95	xen/tools/ioemu-dir/arm-softmmu/config.h File Reference	247
4.95.1	Macro Definition Documentation	248
4.95.1.1	CONFIG_QEMU_PREFIX	248
4.95.1.2	CONFIG_SOFTFLOAT	248
4.95.1.3	CONFIG_SOFTMMU	248
4.95.1.4	TARGET_ARCH	248
4.95.1.5	TARGET_ARM	248
4.96	xen/tools/ioemu-dir/armeb-linux-user/config.h File Reference	248
4.96.1	Macro Definition Documentation	248
4.96.1.1	CONFIG_LINUX_USER	248
4.96.1.2	CONFIG_QEMU_PREFIX	248
4.96.1.3	CONFIG_SOFTFLOAT	248
4.96.1.4	CONFIG_USER_ONLY	248
4.96.1.5	TARGET_ARCH	248
4.96.1.6	TARGET_ARM	248
4.96.1.7	TARGET_HAS_BFLT	248
4.96.1.8	TARGET_WORDS_BIGENDIAN	248
4.97	xen/tools/ioemu-dir/cris-linux-user/config.h File Reference	249
4.97.1	Macro Definition Documentation	249
4.97.1.1	CONFIG_LINUX_USER	249
4.97.1.2	CONFIG_QEMU_PREFIX	249
4.97.1.3	CONFIG_USER_ONLY	249
4.97.1.4	TARGET_ARCH	249
4.97.1.5	TARGET_CRIS	249
4.98	xen/tools/ioemu-dir/cris-softmmu/config.h File Reference	249
4.98.1	Macro Definition Documentation	249
4.98.1.1	CONFIG_QEMU_PREFIX	249
4.98.1.2	CONFIG_SOFTMMU	249
4.98.1.3	TARGET_ARCH	249
4.98.1.4	TARGET_CRIS	249
4.99	xen/tools/ioemu-dir/i386-linux-user/config.h File Reference	249
4.99.1	Macro Definition Documentation	250
4.99.1.1	CONFIG_LINUX_USER	250
4.99.1.2	CONFIG_QEMU_PREFIX	250
4.99.1.3	CONFIG_USER_ONLY	250
4.99.1.4	TARGET_ARCH	250

4.99.1.5 TARGET_I386 . . . . .	250
4.100xen/tools/ioemu-dir/i386-softmmu/config.h File Reference . . . . .	250
4.100.1 Macro Definition Documentation . . . . .	250
4.100.1.1 CONFIG_KVM . . . . .	250
4.100.1.2 CONFIG_QEMU_PREFIX . . . . .	250
4.100.1.3 CONFIG_SOFTMMU . . . . .	250
4.100.1.4 TARGET_ARCH . . . . .	250
4.100.1.5 TARGET_I386 . . . . .	250
4.100.1.6 USE_KQEMU . . . . .	250
4.101xen/tools/ioemu-dir/m68k-linux-user/config.h File Reference . . . . .	250
4.101.1 Macro Definition Documentation . . . . .	251
4.101.1.1 CONFIG_LINUX_USER . . . . .	251
4.101.1.2 CONFIG_QEMU_PREFIX . . . . .	251
4.101.1.3 CONFIG_SOFTFLOAT . . . . .	251
4.101.1.4 CONFIG_USER_ONLY . . . . .	251
4.101.1.5 TARGET_ARCH . . . . .	251
4.101.1.6 TARGET_HAS_BFLT . . . . .	251
4.101.1.7 TARGET_M68K . . . . .	251
4.101.1.8 TARGET_WORDS_BIGENDIAN . . . . .	251
4.102xen/tools/ioemu-dir/m68k-softmmu/config.h File Reference . . . . .	251
4.102.1 Macro Definition Documentation . . . . .	251
4.102.1.1 CONFIG_QEMU_PREFIX . . . . .	251
4.102.1.2 CONFIG_SOFTFLOAT . . . . .	251
4.102.1.3 CONFIG_SOFTMMU . . . . .	251
4.102.1.4 TARGET_ARCH . . . . .	251
4.102.1.5 TARGET_M68K . . . . .	252
4.102.1.6 TARGET_WORDS_BIGENDIAN . . . . .	252
4.103xen/tools/ioemu-dir/mips-linux-user/config.h File Reference . . . . .	252
4.103.1 Macro Definition Documentation . . . . .	252
4.103.1.1 CONFIG_LINUX_USER . . . . .	252
4.103.1.2 CONFIG_QEMU_PREFIX . . . . .	252
4.103.1.3 CONFIG_SOFTFLOAT . . . . .	252
4.103.1.4 CONFIG_USER_ONLY . . . . .	252
4.103.1.5 TARGET_ABI_MIPSO32 . . . . .	252
4.103.1.6 TARGET_ARCH . . . . .	252
4.103.1.7 TARGET_MIPS . . . . .	252
4.103.1.8 TARGET_WORDS_BIGENDIAN . . . . .	252
4.104xen/tools/ioemu-dir/mips-softmmu/config.h File Reference . . . . .	252
4.104.1 Macro Definition Documentation . . . . .	253
4.104.1.1 CONFIG_QEMU_PREFIX . . . . .	253

4.104.1.2 CONFIG_SOFTFLOAT . . . . .	253
4.104.1.3 CONFIG_SOFTMMU . . . . .	253
4.104.1.4 TARGET_ABI_MIPSO32 . . . . .	253
4.104.1.5 TARGET_ARCH . . . . .	253
4.104.1.6 TARGET_MIPS . . . . .	253
4.104.1.7 TARGET_WORDS_BIGENDIAN . . . . .	253
4.105xen/tools/ioemu-dir/mips64-softmmu/config.h File Reference . . . . .	253
4.105.1 Macro Definition Documentation . . . . .	253
4.105.1.1 CONFIG_QEMU_PREFIX . . . . .	253
4.105.1.2 CONFIG_SOFTFLOAT . . . . .	253
4.105.1.3 CONFIG_SOFTMMU . . . . .	253
4.105.1.4 TARGET_ABI_MIPSN64 . . . . .	253
4.105.1.5 TARGET_ARCH . . . . .	253
4.105.1.6 TARGET_MIPS . . . . .	253
4.105.1.7 TARGET_MIPS64 . . . . .	253
4.105.1.8 TARGET_WORDS_BIGENDIAN . . . . .	253
4.106xen/tools/ioemu-dir/mips64el-softmmu/config.h File Reference . . . . .	253
4.106.1 Macro Definition Documentation . . . . .	254
4.106.1.1 CONFIG_QEMU_PREFIX . . . . .	254
4.106.1.2 CONFIG_SOFTFLOAT . . . . .	254
4.106.1.3 CONFIG_SOFTMMU . . . . .	254
4.106.1.4 TARGET_ABI_MIPSN64 . . . . .	254
4.106.1.5 TARGET_ARCH . . . . .	254
4.106.1.6 TARGET_MIPS . . . . .	254
4.106.1.7 TARGET_MIPS64 . . . . .	254
4.107xen/tools/ioemu-dir/mipsel-linux-user/config.h File Reference . . . . .	254
4.107.1 Macro Definition Documentation . . . . .	254
4.107.1.1 CONFIG_LINUX_USER . . . . .	254
4.107.1.2 CONFIG_QEMU_PREFIX . . . . .	254
4.107.1.3 CONFIG_SOFTFLOAT . . . . .	254
4.107.1.4 CONFIG_USER_ONLY . . . . .	254
4.107.1.5 TARGET_ABI_MIPSO32 . . . . .	254
4.107.1.6 TARGET_ARCH . . . . .	255
4.107.1.7 TARGET_MIPS . . . . .	255
4.108xen/tools/ioemu-dir/mipsel-softmmu/config.h File Reference . . . . .	255
4.108.1 Macro Definition Documentation . . . . .	255
4.108.1.1 CONFIG_QEMU_PREFIX . . . . .	255
4.108.1.2 CONFIG_SOFTFLOAT . . . . .	255
4.108.1.3 CONFIG_SOFTMMU . . . . .	255
4.108.1.4 TARGET_ABI_MIPSO32 . . . . .	255

4.108.1.5 TARGET_ARCH . . . . .	255
4.108.1.6 TARGET_MIPS . . . . .	255
4.109xen/tools/ioemu-dir/ppc-linux-user/config.h File Reference . . . . .	255
4.109.1 Macro Definition Documentation . . . . .	255
4.109.1.1 CONFIG_LINUX_USER . . . . .	255
4.109.1.2 CONFIG_QEMU_PREFIX . . . . .	256
4.109.1.3 CONFIG_SOFTFLOAT . . . . .	256
4.109.1.4 CONFIG_USER_ONLY . . . . .	256
4.109.1.5 TARGET_ARCH . . . . .	256
4.109.1.6 TARGET_PPC . . . . .	256
4.109.1.7 TARGET_WORDS_BIGENDIAN . . . . .	256
4.110xen/tools/ioemu-dir/ppc-softmmu/config.h File Reference . . . . .	256
4.110.1 Macro Definition Documentation . . . . .	256
4.110.1.1 CONFIG_QEMU_PREFIX . . . . .	256
4.110.1.2 CONFIG_SOFTFLOAT . . . . .	256
4.110.1.3 CONFIG_SOFTMMU . . . . .	256
4.110.1.4 TARGET_ARCH . . . . .	256
4.110.1.5 TARGET_PPC . . . . .	256
4.110.1.6 TARGET_WORDS_BIGENDIAN . . . . .	256
4.111xen/tools/ioemu-dir/ppc64-linux-user/config.h File Reference . . . . .	256
4.111.1 Macro Definition Documentation . . . . .	257
4.111.1.1 CONFIG_LINUX_USER . . . . .	257
4.111.1.2 CONFIG_QEMU_PREFIX . . . . .	257
4.111.1.3 CONFIG_SOFTFLOAT . . . . .	257
4.111.1.4 CONFIG_USER_ONLY . . . . .	257
4.111.1.5 TARGET_ARCH . . . . .	257
4.111.1.6 TARGET_PPC . . . . .	257
4.111.1.7 TARGET_PPC64 . . . . .	257
4.111.1.8 TARGET_WORDS_BIGENDIAN . . . . .	257
4.112xen/tools/ioemu-dir/ppc64-softmmu/config.h File Reference . . . . .	257
4.112.1 Macro Definition Documentation . . . . .	257
4.112.1.1 CONFIG_QEMU_PREFIX . . . . .	257
4.112.1.2 CONFIG_SOFTFLOAT . . . . .	257
4.112.1.3 CONFIG_SOFTMMU . . . . .	257
4.112.1.4 TARGET_ARCH . . . . .	257
4.112.1.5 TARGET_PPC . . . . .	257
4.112.1.6 TARGET_PPC64 . . . . .	257
4.112.1.7 TARGET_WORDS_BIGENDIAN . . . . .	257
4.113xen/tools/ioemu-dir/ppc64abi32-linux-user/config.h File Reference . . . . .	258
4.113.1 Macro Definition Documentation . . . . .	258

4.113.1.1 CONFIG_LINUX_USER . . . . .	258
4.113.1.2 CONFIG_QEMU_PREFIX . . . . .	258
4.113.1.3 CONFIG_SOFTFLOAT . . . . .	258
4.113.1.4 CONFIG_USER_ONLY . . . . .	258
4.113.1.5 TARGET_ABI32 . . . . .	258
4.113.1.6 TARGET_ARCH . . . . .	258
4.113.1.7 TARGET_PPC . . . . .	258
4.113.1.8 TARGET_PPC64 . . . . .	258
4.113.1.9 TARGET_WORDS_BIGENDIAN . . . . .	258
4.114xen/tools/ioemu-dir/ppcemb-softmmu/config.h File Reference . . . . .	258
4.114.1 Macro Definition Documentation . . . . .	259
4.114.1.1 CONFIG_QEMU_PREFIX . . . . .	259
4.114.1.2 CONFIG_SOFTFLOAT . . . . .	259
4.114.1.3 CONFIG_SOFTMMU . . . . .	259
4.114.1.4 TARGET_ARCH . . . . .	259
4.114.1.5 TARGET_PPC . . . . .	259
4.114.1.6 TARGET_PPCEMB . . . . .	259
4.114.1.7 TARGET_WORDS_BIGENDIAN . . . . .	259
4.115xen/tools/ioemu-dir/sh4-linux-user/config.h File Reference . . . . .	259
4.115.1 Macro Definition Documentation . . . . .	259
4.115.1.1 CONFIG_LINUX_USER . . . . .	259
4.115.1.2 CONFIG_QEMU_PREFIX . . . . .	259
4.115.1.3 CONFIG_USER_ONLY . . . . .	259
4.115.1.4 TARGET_ARCH . . . . .	259
4.115.1.5 TARGET_HAS_BFLT . . . . .	259
4.115.1.6 TARGET_SH4 . . . . .	259
4.116xen/tools/ioemu-dir/sh4-softmmu/config.h File Reference . . . . .	259
4.116.1 Macro Definition Documentation . . . . .	260
4.116.1.1 CONFIG_QEMU_PREFIX . . . . .	260
4.116.1.2 CONFIG_SOFTMMU . . . . .	260
4.116.1.3 TARGET_ARCH . . . . .	260
4.116.1.4 TARGET_SH4 . . . . .	260
4.117xen/tools/ioemu-dir/sh4eb-linux-user/config.h File Reference . . . . .	260
4.117.1 Macro Definition Documentation . . . . .	260
4.117.1.1 CONFIG_LINUX_USER . . . . .	260
4.117.1.2 CONFIG_QEMU_PREFIX . . . . .	260
4.117.1.3 CONFIG_USER_ONLY . . . . .	260
4.117.1.4 TARGET_ARCH . . . . .	260
4.117.1.5 TARGET_HAS_BFLT . . . . .	260
4.117.1.6 TARGET_SH4 . . . . .	260

4.117.1.7 TARGET_WORDS_BIGENDIAN . . . . .	260
4.118xen/tools/ioemu-dir/sh4eb-softmmu/config.h File Reference . . . . .	260
4.118.1 Macro Definition Documentation . . . . .	261
4.118.1.1 CONFIG_QEMU_PREFIX . . . . .	261
4.118.1.2 CONFIG_SOFTMMU . . . . .	261
4.118.1.3 TARGET_ARCH . . . . .	261
4.118.1.4 TARGET_SH4 . . . . .	261
4.118.1.5 TARGET_WORDS_BIGENDIAN . . . . .	261
4.119xen/tools/ioemu-dir/sparc-linux-user/config.h File Reference . . . . .	261
4.119.1 Macro Definition Documentation . . . . .	261
4.119.1.1 CONFIG_LINUX_USER . . . . .	261
4.119.1.2 CONFIG_QEMU_PREFIX . . . . .	261
4.119.1.3 CONFIG_SOFTFLOAT . . . . .	261
4.119.1.4 CONFIG_USER_ONLY . . . . .	261
4.119.1.5 TARGET_ARCH . . . . .	261
4.119.1.6 TARGET_SPARC . . . . .	261
4.119.1.7 TARGET_WORDS_BIGENDIAN . . . . .	261
4.120xen/tools/ioemu-dir/sparc-softmmu/config.h File Reference . . . . .	262
4.120.1 Macro Definition Documentation . . . . .	262
4.120.1.1 CONFIG_QEMU_PREFIX . . . . .	262
4.120.1.2 CONFIG_SOFTFLOAT . . . . .	262
4.120.1.3 CONFIG_SOFTMMU . . . . .	262
4.120.1.4 TARGET_ARCH . . . . .	262
4.120.1.5 TARGET_SPARC . . . . .	262
4.120.1.6 TARGET_WORDS_BIGENDIAN . . . . .	262
4.121xen/tools/ioemu-dir/sparc32plus-linux-user/config.h File Reference . . . . .	262
4.121.1 Macro Definition Documentation . . . . .	262
4.121.1.1 CONFIG_LINUX_USER . . . . .	262
4.121.1.2 CONFIG_QEMU_PREFIX . . . . .	262
4.121.1.3 CONFIG_SOFTFLOAT . . . . .	262
4.121.1.4 CONFIG_USER_ONLY . . . . .	263
4.121.1.5 TARGET_ABI32 . . . . .	263
4.121.1.6 TARGET_ARCH . . . . .	263
4.121.1.7 TARGET_SPARC . . . . .	263
4.121.1.8 TARGET_SPARC64 . . . . .	263
4.121.1.9 TARGET_WORDS_BIGENDIAN . . . . .	263
4.122xen/tools/ioemu-dir/sparc64-linux-user/config.h File Reference . . . . .	263
4.122.1 Macro Definition Documentation . . . . .	263
4.122.1.1 CONFIG_LINUX_USER . . . . .	263
4.122.1.2 CONFIG_QEMU_PREFIX . . . . .	263

4.122.1.3 CONFIG_SOFTFLOAT . . . . .	263
4.122.1.4 CONFIG_USER_ONLY . . . . .	263
4.122.1.5 TARGET_ARCH . . . . .	263
4.122.1.6 TARGET_HAS_ELFLOAD32 . . . . .	263
4.122.1.7 TARGET_SPARC . . . . .	263
4.122.1.8 TARGET_SPARC64 . . . . .	263
4.122.1.9 TARGET_WORDS_BIGENDIAN . . . . .	263
4.123xen/tools/ioemu-dir/x86_64-linux-user/config.h File Reference . . . . .	263
4.123.1 Macro Definition Documentation . . . . .	264
4.123.1.1 CONFIG_LINUX_USER . . . . .	264
4.123.1.2 CONFIG_QEMU_PREFIX . . . . .	264
4.123.1.3 CONFIG_USER_ONLY . . . . .	264
4.123.1.4 TARGET_ARCH . . . . .	264
4.123.1.5 TARGET_I386 . . . . .	264
4.123.1.6 TARGET_X86_64 . . . . .	264
4.124xen/tools/ioemu-dir/x86_64-softmmu/config.h File Reference . . . . .	264
4.124.1 Macro Definition Documentation . . . . .	264
4.124.1.1 CONFIG_KVM . . . . .	264
4.124.1.2 CONFIG_QEMU_PREFIX . . . . .	264
4.124.1.3 CONFIG_SOFTMMU . . . . .	264
4.124.1.4 TARGET_ARCH . . . . .	264
4.124.1.5 TARGET_I386 . . . . .	264
4.124.1.6 TARGET_X86_64 . . . . .	264
4.124.1.7 USE_KQEMU . . . . .	264
4.125xen/tools/ioemu-dir/config-host.h File Reference . . . . .	265
4.125.1 Macro Definition Documentation . . . . .	265
4.125.1.1 AUDIO_DRIVERS . . . . .	265
4.125.1.2 CONFIG_AC97 . . . . .	265
4.125.1.3 CONFIG_AIO . . . . .	265
4.125.1.4 CONFIG_ES1370 . . . . .	265
4.125.1.5 CONFIG_OPENGL . . . . .	265
4.125.1.6 CONFIG_QEMU_SHARED_DIR . . . . .	265
4.125.1.7 CONFIG_SB16 . . . . .	265
4.125.1.8 CONFIG_SDL . . . . .	265
4.125.1.9 CONFIG_UNAME_RELEASE . . . . .	265
4.125.1.10CONFIG_VNC_TLS . . . . .	265
4.125.1.11DEFAULT_NETWORK_DOWN_SCRIPT . . . . .	265
4.125.1.12DEFAULT_NETWORK_SCRIPT . . . . .	265
4.125.1.13HAVE_BYTESWAP_H . . . . .	265
4.125.1.14HAVE_IOVEC . . . . .	265

4.125.1.15	HOST_LONG_BITS	266
4.125.1.16	HOST_X86_64	266
4.125.1.17	QEMU_VERSION	266
4.126	xen/tools/ioemu-dir/tests/test-mmap.c File Reference	266
4.127	xen/tools/libxc/xc_caas.c File Reference	266
4.127.1	Function Documentation	266
4.127.1.1	xc_caas_exempt	266
4.127.1.2	xc_caas_op	266
4.128	xen/tools/libxc/xc_dom.h File Reference	266
4.128.1	Macro Definition Documentation	268
4.128.1.1	__init	268
4.128.1.2	INVALID_P2M_ENTRY	268
4.128.1.3	PRIPfn	268
4.128.1.4	XC_DOM_PAGE_SHIFT	268
4.128.1.5	XC_DOM_PAGE_SIZE	268
4.128.1.6	xc_dom_panic	268
4.128.1.7	xc_dom_trace	268
4.128.2	Typedef Documentation	268
4.128.2.1	xen_paddr_t	268
4.128.2.2	xen_vaddr_t	268
4.128.3	Function Documentation	268
4.128.3.1	arch_setup_bootearly	268
4.128.3.2	arch_setup_bootlate	268
4.128.3.3	arch_setup_meminit	268
4.128.3.4	xc_dom_alloc_page	268
4.128.3.5	xc_dom_alloc_segment	268
4.128.3.6	xc_dom_allocate	268
4.128.3.7	xc_dom_boot_domU_map	268
4.128.3.8	xc_dom_boot_image	268
4.128.3.9	xc_dom_boot_mem_init	268
4.128.3.10	xc_dom_boot_xen_init	268
4.128.3.11	xc_dom_build_image	268
4.128.3.12	xc_dom_check_gzip	268
4.128.3.13	xc_dom_compat_check	268
4.128.3.14	xc_dom_do_gunzip	268
4.128.3.15	xc_dom_find_arch_hooks	268
4.128.3.16	xc_dom_kernel_file	269
4.128.3.17	xc_dom_kernel_mem	269
4.128.3.18	xc_dom_log_memory_footprint	269
4.128.3.19	xc_dom_loginit	269



4.128.3.20xc_dom_malloc . . . . .	269
4.128.3.21xc_dom_malloc_filemap . . . . .	269
4.128.3.22xc_dom_malloc_page_aligned . . . . .	269
4.128.3.23xc_dom_mem_init . . . . .	269
4.128.3.24xc_dom_panic_func . . . . .	269
4.128.3.25xc_dom_parse_image . . . . .	269
4.128.3.26xc_dom_pfn_to_ptr . . . . .	269
4.128.3.27xc_dom_printf . . . . .	269
4.128.3.28xc_dom_ramdisk_file . . . . .	269
4.128.3.29xc_dom_ramdisk_mem . . . . .	269
4.128.3.30xc_dom_register_arch_hooks . . . . .	269
4.128.3.31xc_dom_register_loader . . . . .	269
4.128.3.32xc_dom_release . . . . .	269
4.128.3.33xc_dom_release_phys . . . . .	269
4.128.3.34xc_dom_strdup . . . . .	269
4.128.3.35xc_dom_try_gunzip . . . . .	269
4.128.3.36xc_dom_unmap_all . . . . .	269
4.128.3.37xc_dom_unmap_one . . . . .	269
4.128.3.38xc_dom_update_guest_p2m . . . . .	269
4.129xen/tools/libxc/xc_dom_core.c File Reference . . . . .	270
4.129.1 Function Documentation . . . . .	270
4.129.1.1 xc_dom_alloc_page . . . . .	270
4.129.1.2 xc_dom_alloc_segment . . . . .	270
4.129.1.3 xc_dom_allocate . . . . .	271
4.129.1.4 xc_dom_build_image . . . . .	271
4.129.1.5 xc_dom_check_gzip . . . . .	271
4.129.1.6 xc_dom_do_gunzip . . . . .	271
4.129.1.7 xc_dom_find_arch_hooks . . . . .	271
4.129.1.8 xc_dom_kernel_file . . . . .	271
4.129.1.9 xc_dom_kernel_mem . . . . .	271
4.129.1.10xc_dom_log_memory_footprint . . . . .	271
4.129.1.11xc_dom_loginit . . . . .	271
4.129.1.12xc_dom_malloc . . . . .	271
4.129.1.13xc_dom_malloc_filemap . . . . .	271
4.129.1.14xc_dom_malloc_page_aligned . . . . .	271
4.129.1.15xc_dom_mem_init . . . . .	271
4.129.1.16xc_dom_panic_func . . . . .	271
4.129.1.17xc_dom_parse_image . . . . .	271
4.129.1.18xc_dom_pfn_to_ptr . . . . .	271
4.129.1.19xc_dom_printf . . . . .	271

4.129.1.20xc_dom_ramdisk_file . . . . .	271
4.129.1.21xc_dom_ramdisk_mem . . . . .	271
4.129.1.22xc_dom_register_arch_hooks . . . . .	271
4.129.1.23xc_dom_register_loader . . . . .	271
4.129.1.24xc_dom_release . . . . .	271
4.129.1.25xc_dom_strdup . . . . .	271
4.129.1.26xc_dom_try_gunzip . . . . .	271
4.129.1.27xc_dom_unmap_all . . . . .	271
4.129.1.28xc_dom_unmap_one . . . . .	271
4.129.1.29xc_dom_update_guest_p2m . . . . .	271
4.130xen/tools/libxc/xc_dom_x86.c File Reference . . . . .	272
4.130.1 Macro Definition Documentation . . . . .	272
4.130.1.1 bits_to_mask . . . . .	272
4.130.1.2 L1_PROT . . . . .	272
4.130.1.3 L1_PROT . . . . .	272
4.130.1.4 L2_PROT . . . . .	272
4.130.1.5 L2_PROT . . . . .	272
4.130.1.6 L3_PROT . . . . .	272
4.130.1.7 L3_PROT . . . . .	272
4.130.1.8 L4_PROT . . . . .	273
4.130.1.9 pfn_to_paddr . . . . .	273
4.130.1.10round_down . . . . .	273
4.130.1.11round_up . . . . .	273
4.130.1.12SUPERPAGE_NR_PFNS . . . . .	273
4.130.1.13SUPERPAGE_PFN_SHIFT . . . . .	273
4.130.2 Function Documentation . . . . .	273
4.130.2.1 arch_setup_bootearly . . . . .	273
4.130.2.2 arch_setup_bootlate . . . . .	273
4.130.2.3 arch_setup_meminit . . . . .	273
4.131xen/tools/libxc/xenctrl.h File Reference . . . . .	273
4.131.1 Macro Definition Documentation . . . . .	280
4.131.1.1 __XEN_TOOLS__ . . . . .	280
4.131.1.2 DECLARE_HYPERCALL_BUFFER . . . . .	280
4.131.1.3 DECLARE_HYPERCALL_BUFFER_ARGUMENT . . . . .	281
4.131.1.4 HYPERCALL_BUFFER . . . . .	281
4.131.1.5 HYPERCALL_BUFFER_AS_ARG . . . . .	281
4.131.1.6 HYPERCALL_BUFFER_INIT_NO_BOUNCE . . . . .	281
4.131.1.7 INVALID_MFN . . . . .	281
4.131.1.8 set_xen_guest_handle . . . . .	281
4.131.1.9 XC__HYPERCALL_BUFFER_NAME . . . . .	281

4.131.1.10XC_CORE_MAGIC . . . . .	281
4.131.1.11XC_CORE_MAGIC_HVM . . . . .	281
4.131.1.12xc_hypercall_buffer_alloc . . . . .	281
4.131.1.13xc_hypercall_buffer_alloc_pages . . . . .	282
4.131.1.14xc_hypercall_buffer_free . . . . .	282
4.131.1.15xc_hypercall_buffer_free_pages . . . . .	282
4.131.1.16XC_MAX_ERROR_MSG_LEN . . . . .	282
4.131.1.17XC_PAGE_MASK . . . . .	282
4.131.1.18XC_PAGE_SHIFT . . . . .	282
4.131.1.19XC_PAGE_SIZE . . . . .	282
4.131.1.20XENCTRL_HAS_XC_INTERFACE . . . . .	282
4.131.2 Typedef Documentation . . . . .	282
4.131.2.1 dumpcore_rtn_t . . . . .	282
4.131.2.2 evtchn_port_or_error_t . . . . .	282
4.131.2.3 xc_core_header_t . . . . .	282
4.131.2.4 xc_cpu_to_core_t . . . . .	282
4.131.2.5 xc_cpu_to_node_t . . . . .	282
4.131.2.6 xc_cpu_to_socket_t . . . . .	282
4.131.2.7 xc_cpuidinfo_t . . . . .	282
4.131.2.8 xc_cpumap_t . . . . .	282
4.131.2.9 xc_cpupoolinfo_t . . . . .	282
4.131.2.10xc_cx_stat_t . . . . .	282
4.131.2.11xc_domaininfo_t . . . . .	282
4.131.2.12xc_dominfo_t . . . . .	282
4.131.2.13xc_error . . . . .	282
4.131.2.14xc_error_code . . . . .	282
4.131.2.15xc_evtchn . . . . .	282
4.131.2.16xc_evtchn_status_t . . . . .	282
4.131.2.17xc_gnttab . . . . .	282
4.131.2.18xc_hypercall_buffer_t . . . . .	282
4.131.2.19xc_interface . . . . .	283
4.131.2.20xc_lockprof_data_t . . . . .	283
4.131.2.21xc_node_to_memfree_t . . . . .	283
4.131.2.22xc_node_to_memsizes_t . . . . .	283
4.131.2.23xc_node_to_node_dist_t . . . . .	283
4.131.2.24xc_numainfo_t . . . . .	283
4.131.2.25xc_ondemand_t . . . . .	283
4.131.2.26xc_perfc_desc_t . . . . .	283
4.131.2.27xc_perfc_val_t . . . . .	283
4.131.2.28xc_physinfo_t . . . . .	283

4.131.2.29xc_shadow_op_stats_t . . . . .	283
4.131.2.30xc_topologyinfo_t . . . . .	283
4.131.2.31xc_userspace_t . . . . .	283
4.131.2.32xc_vcpuinfo_t . . . . .	283
4.131.3 Enumeration Type Documentation . . . . .	283
4.131.3.1 xc_error_code . . . . .	283
4.131.3.2 xc_open_flags . . . . .	283
4.131.4 Function Documentation . . . . .	283
4.131.4.1 xc__hypercall_buffer_alloc . . . . .	283
4.131.4.2 xc__hypercall_buffer_alloc_pages . . . . .	283
4.131.4.3 xc__hypercall_buffer_free . . . . .	283
4.131.4.4 xc__hypercall_buffer_free_pages . . . . .	284
4.131.4.5 XC__HYPERCALL_BUFFER_NAME . . . . .	284
4.131.4.6 xc_acm_op . . . . .	284
4.131.4.7 xc_assign_device . . . . .	284
4.131.4.8 xc_availheap . . . . .	284
4.131.4.9 xc_caas_exempt . . . . .	284
4.131.4.10xc_caas_op . . . . .	284
4.131.4.11xc_clear_domain_page . . . . .	284
4.131.4.12xc_clear_last_error . . . . .	284
4.131.4.13xc_copy_to_domain_page . . . . .	284
4.131.4.14xc_cpu_offline . . . . .	284
4.131.4.15xc_cpu_online . . . . .	284
4.131.4.16xc_cpumap_alloc . . . . .	284
4.131.4.17xc_cpupool_addcpu . . . . .	284
4.131.4.18xc_cpupool_create . . . . .	284
4.131.4.19xc_cpupool_destroy . . . . .	284
4.131.4.20xc_cpupool_freeinfo . . . . .	285
4.131.4.21xc_cpupool_getinfo . . . . .	285
4.131.4.22xc_cpupool_infofree . . . . .	285
4.131.4.23xc_cpupool_movedomain . . . . .	285
4.131.4.24xc_cpupool_removecpu . . . . .	285
4.131.4.25xc_deassign_device . . . . .	285
4.131.4.26xc_disable_turbo . . . . .	285
4.131.4.27xc_domain_add_to_physmap . . . . .	285
4.131.4.28xc_domain_bind_pt_irq . . . . .	285
4.131.4.29xc_domain_bind_pt_isa_irq . . . . .	285
4.131.4.30xc_domain_bind_pt_pci_irq . . . . .	285
4.131.4.31xc_domain_create . . . . .	285
4.131.4.32xc_domain_debug_control . . . . .	285

4.131.4.33xc_domain_decrease_reservation . . . . .	285
4.131.4.34xc_domain_decrease_reservation_exact . . . . .	286
4.131.4.35xc_domain_destroy . . . . .	286
4.131.4.36xc_domain_disable_migrate . . . . .	286
4.131.4.37xc_domain_dumpcore . . . . .	286
4.131.4.38xc_domain_dumpcore_via_callback . . . . .	286
4.131.4.39xc_domain_get_cpu_usage . . . . .	286
4.131.4.40xc_domain_get_machine_address_size . . . . .	286
4.131.4.41xc_domain_get_pod_target . . . . .	286
4.131.4.42xc_domain_get_tsc_info . . . . .	286
4.131.4.43xc_domain_getinfo . . . . .	286
4.131.4.44xc_domain_getinfo . . . . .	286
4.131.4.45xc_domain_hvm_getcontext . . . . .	287
4.131.4.46xc_domain_hvm_getcontext_partial . . . . .	287
4.131.4.47xc_domain_hvm_setcontext . . . . .	287
4.131.4.48xc_domain_increase_reservation . . . . .	287
4.131.4.49xc_domain_increase_reservation_exact . . . . .	287
4.131.4.50xc_domain_iomem_permission . . . . .	287
4.131.4.51xc_domain_ioport_mapping . . . . .	287
4.131.4.52xc_domain_ioport_permission . . . . .	287
4.131.4.53xc_domain_irq_permission . . . . .	287
4.131.4.54xc_domain_max_vcpus . . . . .	287
4.131.4.55xc_domain_maximum_gpfn . . . . .	287
4.131.4.56xc_domain_memory_exchange_pages . . . . .	288
4.131.4.57xc_domain_memory_mapping . . . . .	288
4.131.4.58xc_domain_pause . . . . .	288
4.131.4.59xc_domain_pin_memory_cacheattr . . . . .	288
4.131.4.60xc_domain_populate_physmap . . . . .	288
4.131.4.61xc_domain_populate_physmap_exact . . . . .	288
4.131.4.62xc_domain_resume . . . . .	288
4.131.4.63xc_domain_send_trigger . . . . .	288
4.131.4.64xc_domain_set_access_required . . . . .	288
4.131.4.65xc_domain_set_machine_address_size . . . . .	288
4.131.4.66xc_domain_set_memmap_limit . . . . .	288
4.131.4.67xc_domain_set_pod_target . . . . .	288
4.131.4.68xc_domain_set_target . . . . .	288
4.131.4.69xc_domain_set_time_offset . . . . .	289
4.131.4.70xc_domain_set_tsc_info . . . . .	289
4.131.4.71xc_domain_setdebugging . . . . .	289
4.131.4.72xc_domain_sethandle . . . . .	289

4.131.4.73xc_domain_setmaxmem . . . . .	289
4.131.4.74xc_domain_shutdown . . . . .	289
4.131.4.75xc_domain_subscribe_for_suspend . . . . .	289
4.131.4.76xc_domain_suppress_spurious_page_faults . . . . .	289
4.131.4.77xc_domain_unbind_msi_irq . . . . .	289
4.131.4.78xc_domain_unbind_pt_irq . . . . .	289
4.131.4.79xc_domain_unpause . . . . .	289
4.131.4.80xc_domain_update_msi_irq . . . . .	289
4.131.4.81xc_domctl . . . . .	289
4.131.4.82xc_elf_set_logfile . . . . .	289
4.131.4.83xc_enable_turbo . . . . .	289
4.131.4.84xc_error_code_to_desc . . . . .	289
4.131.4.85xc_evtchn_alloc_unbound . . . . .	290
4.131.4.86xc_evtchn_bind_interdomain . . . . .	290
4.131.4.87xc_evtchn_bind_unbound_port . . . . .	290
4.131.4.88xc_evtchn_bind_virq . . . . .	290
4.131.4.89xc_evtchn_close . . . . .	290
4.131.4.90xc_evtchn_fd . . . . .	290
4.131.4.91xc_evtchn_notify . . . . .	290
4.131.4.92xc_evtchn_open . . . . .	290
4.131.4.93xc_evtchn_pending . . . . .	290
4.131.4.94xc_evtchn_reset . . . . .	290
4.131.4.95xc_evtchn_status . . . . .	290
4.131.4.96xc_evtchn_unbind . . . . .	290
4.131.4.97xc_evtchn_unmask . . . . .	290
4.131.4.98xc_flask_access . . . . .	290
4.131.4.99xc_flask_add_device . . . . .	290
4.131.4.100xc_flask_add_iomem . . . . .	290
4.131.4.101xc_flask_add_ioport . . . . .	290
4.131.4.102xc_flask_add_pirq . . . . .	290
4.131.4.103xc_flask_avc_cachestats . . . . .	290
4.131.4.104xc_flask_avc_hashstats . . . . .	290
4.131.4.105xc_flask_context_to_sid . . . . .	290
4.131.4.106xc_flask_del_device . . . . .	291
4.131.4.107xc_flask_del_iomem . . . . .	291
4.131.4.108xc_flask_del_ioport . . . . .	291
4.131.4.109xc_flask_del_pirq . . . . .	291
4.131.4.110xc_flask_getavc_threshold . . . . .	291
4.131.4.111xc_flask_getenforce . . . . .	291
4.131.4.112xc_flask_load . . . . .	291

4.131.4.118c_flask_op	291
4.131.4.11xc_flask_policyvers	291
4.131.4.115c_flask_setavc_threshold	291
4.131.4.116c_flask_setenforce	291
4.131.4.117c_flask_sid_to_context	291
4.131.4.118c_get_cpufreq_avgfreq	291
4.131.4.118c_get_cpufreq_para	291
4.131.4.120c_get_cpuidle_max_cstate	291
4.131.4.121c_get_cpumap_size	291
4.131.4.122c_get_device_group	291
4.131.4.123c_get_hvm_param	291
4.131.4.124c_get_last_error	291
4.131.4.125c_get_max_cpus	291
4.131.4.126c_get_pfn_list	291
4.131.4.127c_get_tot_pages	291
4.131.4.128c_get_vcpu_migration_delay	291
4.131.4.128c_getcpuinfo	291
4.131.4.130c_gnttab_close	291
4.131.4.131c_gnttab_get_version	291
4.131.4.132c_gnttab_map_domain_grant_refs	292
4.131.4.133c_gnttab_map_grant_ref	292
4.131.4.134c_gnttab_map_grant_refs	292
4.131.4.135c_gnttab_map_table_v1	292
4.131.4.136c_gnttab_map_table_v2	292
4.131.4.137c_gnttab_munmap	292
4.131.4.138c_gnttab_op	292
4.131.4.138c_gnttab_open	292
4.131.4.140c_gnttab_set_max_grants	292
4.131.4.141c_hvm_get_mem_access	292
4.131.4.142c_hvm_inject_trap	292
4.131.4.143c_hvm_modified_memory	292
4.131.4.144c_hvm_set_isa_irq_level	292
4.131.4.145c_hvm_set_mem_access	292
4.131.4.146c_hvm_set_mem_type	292
4.131.4.147c_hvm_set_pci_intx_level	292
4.131.4.148c_hvm_set_pci_link_route	292
4.131.4.148c_hvm_track_dirty_vram	292
4.131.4.150c_ia64_fpsr_default	293
4.131.4.151c_ia64_nvram_init	293
4.131.4.152c_ia64_save_to_nvram	293

4.131.4.153c_ia64_set_os_type . . . . .	293
4.131.4.154c_interface_close . . . . .	293
4.131.4.155c_interface_is_fake . . . . .	293
4.131.4.156c_interface_open . . . . .	293
4.131.4.157c_lockprof_query . . . . .	293
4.131.4.158c_lockprof_query_number . . . . .	293
4.131.4.159c_lockprof_reset . . . . .	293
4.131.4.160c_machphys_mfn_list . . . . .	293
4.131.4.161c_make_page_below_4G . . . . .	293
4.131.4.162c_map_foreign_batch . . . . .	294
4.131.4.163c_map_foreign_bulk . . . . .	294
4.131.4.164c_map_foreign_pages . . . . .	294
4.131.4.165c_map_foreign_range . . . . .	294
4.131.4.166c_maximum_ram_page . . . . .	294
4.131.4.167c_mem_access_resume . . . . .	294
4.131.4.168c_mem_event_control . . . . .	294
4.131.4.169c_mem_event_disable . . . . .	294
4.131.4.170c_mem_event_enable . . . . .	294
4.131.4.171c_mem_paging_evict . . . . .	294
4.131.4.172c_mem_paging_nominate . . . . .	294
4.131.4.173c_mem_paging_prep . . . . .	294
4.131.4.174c_mem_paging_resume . . . . .	294
4.131.4.175c_memshr_control . . . . .	294
4.131.4.176c_memshr_debug_gfn . . . . .	294
4.131.4.177c_memshr_debug_gref . . . . .	294
4.131.4.178c_memshr_debug_mfn . . . . .	294
4.131.4.179c_memshr_domain_resume . . . . .	295
4.131.4.180c_memshr_nominate_gfn . . . . .	295
4.131.4.181c_memshr_nominate_gref . . . . .	295
4.131.4.182c_memshr_share . . . . .	295
4.131.4.183c_mmuext_op . . . . .	295
4.131.4.184c_numainfo . . . . .	295
4.131.4.185c_perfc_query . . . . .	295
4.131.4.186c_perfc_query_number . . . . .	295
4.131.4.187c_perfc_reset . . . . .	295
4.131.4.188c_physdev_map_pirq . . . . .	295
4.131.4.189c_physdev_map_pirq_msi . . . . .	295
4.131.4.190c_physdev_pci_access_modify . . . . .	295
4.131.4.191c_physdev_unmap_pirq . . . . .	295
4.131.4.192c_physinfo . . . . .	295



4.131.4.198	<a href="#">pm_get_cxstat</a>	295
4.131.4.199	<a href="#">pm_get_max_cx</a>	295
4.131.4.199	<a href="#">pm_get_max_px</a>	295
4.131.4.199	<a href="#">pm_get_pxstat</a>	295
4.131.4.199	<a href="#">pm_reset_cxstat</a>	295
4.131.4.199	<a href="#">pm_reset_pxstat</a>	295
4.131.4.199	<a href="#">readconsolering</a>	295
4.131.4.200	<a href="#">sched_arinc653_schedule_get</a>	295
4.131.4.200	<a href="#">sched_arinc653_schedule_set</a>	295
4.131.4.200	<a href="#">sched_credit2_domain_get</a>	295
4.131.4.200	<a href="#">sched_credit2_domain_set</a>	295
4.131.4.200	<a href="#">sched_credit_domain_get</a>	295
4.131.4.200	<a href="#">sched_credit_domain_set</a>	296
4.131.4.200	<a href="#">sched_id</a>	296
4.131.4.200	<a href="#">sedf_domain_get</a>	296
4.131.4.200	<a href="#">sedf_domain_set</a>	296
4.131.4.200	<a href="#">send_debug_keys</a>	296
4.131.4.210	<a href="#">set_cpufreq_gov</a>	296
4.131.4.210	<a href="#">set_cpufreq_para</a>	296
4.131.4.210	<a href="#">set_cpuidle_max_cstate</a>	296
4.131.4.210	<a href="#">set_hvm_param</a>	296
4.131.4.210	<a href="#">set_sched_opt_smt</a>	296
4.131.4.210	<a href="#">set_vcpu_migration_delay</a>	296
4.131.4.210	<a href="#">shadow_control</a>	296
4.131.4.210	<a href="#">strerror</a>	296
4.131.4.210	<a href="#">sysctl</a>	296
4.131.4.210	<a href="#">tbuf_disable</a>	296
4.131.4.220	<a href="#">tbuf_enable</a>	296
4.131.4.220	<a href="#">tbuf_get_size</a>	296
4.131.4.220	<a href="#">tbuf_set_cpu_mask</a>	296
4.131.4.220	<a href="#">tbuf_set_evt_mask</a>	296
4.131.4.220	<a href="#">tbuf_set_size</a>	297
4.131.4.220	<a href="#">test_assign_device</a>	297
4.131.4.220	<a href="#">tmem_auth</a>	297
4.131.4.220	<a href="#">tmem_control</a>	297
4.131.4.220	<a href="#">tmem_control_oid</a>	297
4.131.4.220	<a href="#">tmem_restore</a>	297
4.131.4.230	<a href="#">tmem_restore_extra</a>	297
4.131.4.230	<a href="#">tmem_save</a>	297
4.131.4.230	<a href="#">tmem_save_done</a>	297

4.131.4.238	<a href="#">tmem_save_extra</a>	297
4.131.4.239	<a href="#">topologyinfo</a>	297
4.131.4.240	<a href="#">translate_foreign_address</a>	297
4.131.4.241	<a href="#">vcpu_getaffinity</a>	297
4.131.4.242	<a href="#">vcpu_getcontext</a>	297
4.131.4.243	<a href="#">vcpu_getinfo</a>	298
4.131.4.244	<a href="#">vcpu_setaffinity</a>	298
4.131.4.245	<a href="#">vcpu_setcontext</a>	298
4.131.4.246	<a href="#">version</a>	298
4.131.4.247	<a href="#">watchdog</a>	298
4.132	<a href="#">xen/tools/libxl/libxl.c File Reference</a>	298
4.132.1	<a href="#">Macro Definition Documentation</a>	300
4.132.1.1	<a href="#">BACKEND_STRING_SIZE</a>	300
4.132.1.2	<a href="#">PAGE_TO_MEMKB</a>	300
4.132.1.3	<a href="#">STRINGIFY</a>	300
4.132.1.4	<a href="#">TOSTRING</a>	300
4.132.2	<a href="#">Function Documentation</a>	300
4.132.2.1	<a href="#">libxl_button_press</a>	301
4.132.2.2	<a href="#">libxl_cdrom_insert</a>	301
4.132.2.3	<a href="#">libxl_console_exec</a>	301
4.132.2.4	<a href="#">libxl_cpupool_cpuadd</a>	301
4.132.2.5	<a href="#">libxl_cpupool_cpuadd_node</a>	301
4.132.2.6	<a href="#">libxl_cpupool_cpuremove</a>	301
4.132.2.7	<a href="#">libxl_cpupool_cpuremove_node</a>	301
4.132.2.8	<a href="#">libxl_cpupool_movedomain</a>	301
4.132.2.9	<a href="#">libxl_cpupool_rename</a>	301
4.132.2.10	<a href="#">libxl_create_cpupool</a>	301
4.132.2.11	<a href="#">libxl_ctx_free</a>	301
4.132.2.12	<a href="#">libxl_ctx_init</a>	301
4.132.2.13	<a href="#">libxl_destroy_cpupool</a>	301
4.132.2.14	<a href="#">libxl_device_console_add</a>	301
4.132.2.15	<a href="#">libxl_device_disk_add</a>	301
4.132.2.16	<a href="#">libxl_device_disk_del</a>	301
4.132.2.17	<a href="#">libxl_device_disk_getinfo</a>	301
4.132.2.18	<a href="#">libxl_device_disk_list</a>	301
4.132.2.19	<a href="#">libxl_device_disk_local_attach</a>	301
4.132.2.20	<a href="#">libxl_device_disk_local_detach</a>	301
4.132.2.21	<a href="#">libxl_device_net2_add</a>	301
4.132.2.22	<a href="#">libxl_device_net2_del</a>	301
4.132.2.23	<a href="#">libxl_device_net2_init</a>	301

4.132.2.24libxl_device_net2_list . . . . .	301
4.132.2.25libxl_device_nic_add . . . . .	301
4.132.2.26libxl_device_nic_del . . . . .	301
4.132.2.27libxl_device_nic_init . . . . .	302
4.132.2.28libxl_device_vfb_add . . . . .	302
4.132.2.29libxl_device_vfb_clean_shutdown . . . . .	302
4.132.2.30libxl_device_vfb_hard_shutdown . . . . .	302
4.132.2.31libxl_device_vfb_init . . . . .	302
4.132.2.32libxl_device_vkb_add . . . . .	302
4.132.2.33libxl_device_vkb_clean_shutdown . . . . .	302
4.132.2.34libxl_device_vkb_hard_shutdown . . . . .	302
4.132.2.35libxl_device_vkb_init . . . . .	302
4.132.2.36libxl_domain_core_dump . . . . .	302
4.132.2.37libxl_domain_destroy . . . . .	302
4.132.2.38libxl_domain_info . . . . .	302
4.132.2.39libxl_domain_need_memory . . . . .	302
4.132.2.40libxl_domain_pause . . . . .	302
4.132.2.41libxl_domain_preserve . . . . .	302
4.132.2.42libxl_domain_rename . . . . .	302
4.132.2.43libxl_domain_resume . . . . .	302
4.132.2.44libxl_domain_setmaxmem . . . . .	302
4.132.2.45libxl_domain_shutdown . . . . .	302
4.132.2.46libxl_domain_suspend . . . . .	302
4.132.2.47libxl_domain_unpause . . . . .	302
4.132.2.48libxl_event_get_disk_eject_info . . . . .	302
4.132.2.49libxl_event_get_domain_death_info . . . . .	303
4.132.2.50libxl_file_reference_destroy . . . . .	303
4.132.2.51libxl_free_event . . . . .	303
4.132.2.52libxl_free_waiter . . . . .	303
4.132.2.53libxl_get_event . . . . .	303
4.132.2.54libxl_get_free_memory . . . . .	303
4.132.2.55libxl_get_freecpus . . . . .	303
4.132.2.56libxl_get_memory_target . . . . .	303
4.132.2.57libxl_get_physinfo . . . . .	303
4.132.2.58libxl_get_sched_id . . . . .	303
4.132.2.59libxl_get_topologyinfo . . . . .	303
4.132.2.60libxl_get_version_info . . . . .	303
4.132.2.61libxl_get_wait_fd . . . . .	303
4.132.2.62libxl_key_value_list_destroy . . . . .	303
4.132.2.63libxl_list_cpupool . . . . .	303

4.132.2.64libxl_list_domain . . . . .	303
4.132.2.65libxl_list_nics . . . . .	303
4.132.2.66libxl_list_vcpu . . . . .	303
4.132.2.67libxl_list_vm . . . . .	303
4.132.2.68libxl_primary_console_exec . . . . .	304
4.132.2.69libxl_sched_credit_domain_get . . . . .	304
4.132.2.70libxl_sched_credit_domain_set . . . . .	304
4.132.2.71libxl_send_debug_keys . . . . .	304
4.132.2.72libxl_send_sysrq . . . . .	304
4.132.2.73libxl_send_trigger . . . . .	304
4.132.2.74libxl_set_memory_target . . . . .	304
4.132.2.75libxl_set_vcpuaffinity . . . . .	304
4.132.2.76libxl_set_vcpuonline . . . . .	304
4.132.2.77libxl_stop_waiting . . . . .	304
4.132.2.78libxl_string_list_destroy . . . . .	304
4.132.2.79libxl_tmem_destroy . . . . .	304
4.132.2.80libxl_tmem_freeable . . . . .	304
4.132.2.81libxl_tmem_freeze . . . . .	304
4.132.2.82libxl_tmem_list . . . . .	304
4.132.2.83libxl_tmem_set . . . . .	304
4.132.2.84libxl_tmem_shared_auth . . . . .	304
4.132.2.85libxl_tmem_thaw . . . . .	304
4.132.2.86libxl_vm_get_start_time . . . . .	304
4.132.2.87libxl_vncviewer_exec . . . . .	304
4.132.2.88libxl_wait_for_disk_ejects . . . . .	304
4.132.2.89libxl_wait_for_domain_death . . . . .	304
4.132.2.90libxl_wait_for_free_memory . . . . .	304
4.132.2.91libxl_wait_for_memory_target . . . . .	305
4.132.2.92libxl_xen_console_read_finish . . . . .	305
4.132.2.93libxl_xen_console_read_line . . . . .	305
4.132.2.94libxl_xen_console_read_start . . . . .	305
4.133xen/tools/libxl/libxl.h File Reference . . . . .	305
4.133.1 Macro Definition Documentation . . . . .	309
4.133.1.1 LIBXL_CPUARRAY_INVALID_ENTRY . . . . .	309
4.133.1.2 LIBXL_PCI_FUNC_ALL . . . . .	309
4.133.1.3 LIBXL_VERSION . . . . .	309
4.133.1.4 XL_SUSPEND_DEBUG . . . . .	309
4.133.1.5 XL_SUSPEND_LIVE . . . . .	309
4.133.2 Typedef Documentation . . . . .	309
4.133.2.1 libxl_console_ready . . . . .	309

4.133.2.2 libxl_cpuid_policy . . . . .	309
4.133.2.3 libxl_cpuid_policy_list . . . . .	309
4.133.2.4 libxl_hwcap . . . . .	309
4.133.2.5 libxl_key_value_list . . . . .	309
4.133.2.6 libxl_mac . . . . .	309
4.133.2.7 libxl_string_list . . . . .	309
4.133.2.8 libxl_xen_console_reader . . . . .	309
4.133.3 Enumeration Type Documentation . . . . .	309
4.133.3.1 anonymous enum . . . . .	309
4.133.3.2 libxl_action_on_shutdown . . . . .	310
4.133.3.3 libxl_button . . . . .	310
4.133.3.4 libxl_console_consback . . . . .	310
4.133.3.5 libxl_console_constype . . . . .	310
4.133.3.6 libxl_disk_backend . . . . .	310
4.133.3.7 libxl_disk_format . . . . .	310
4.133.3.8 libxl_event_type . . . . .	311
4.133.3.9 libxl_nic_type . . . . .	311
4.133.3.10 libxl_qemu_machine_type . . . . .	311
4.133.4 Function Documentation . . . . .	311
4.133.4.1 init_console_info . . . . .	311
4.133.4.2 libxl_bindir_path . . . . .	311
4.133.4.3 libxl_button_press . . . . .	311
4.133.4.4 libxl_cdrom_insert . . . . .	311
4.133.4.5 libxl_console_exec . . . . .	311
4.133.4.6 libxl_cpumap_destroy . . . . .	311
4.133.4.7 libxl_cpuid_apply_policy . . . . .	311
4.133.4.8 libxl_cpuid_destroy . . . . .	311
4.133.4.9 libxl_cpuid_parse_config . . . . .	311
4.133.4.10 libxl_cpuid_parse_config_xend . . . . .	311
4.133.4.11 libxl_cpuid_set . . . . .	311
4.133.4.12 libxl_cpumap_destroy . . . . .	311
4.133.4.13 libxl_cpupool_cpuadd . . . . .	311
4.133.4.14 libxl_cpupool_cpuadd_node . . . . .	311
4.133.4.15 libxl_cpupool_cpuremove . . . . .	311
4.133.4.16 libxl_cpupool_cpuremove_node . . . . .	311
4.133.4.17 libxl_cpupool_movedomain . . . . .	312
4.133.4.18 libxl_cpupool_rename . . . . .	312
4.133.4.19 libxl_create_cpupool . . . . .	312
4.133.4.20 libxl_ctx_free . . . . .	312
4.133.4.21 libxl_ctx_init . . . . .	312

4.133.4.22libxl_ctx_postfork . . . . .	312
4.133.4.23libxl_ctx_set_log . . . . .	312
4.133.4.24libxl_destroy_cpupool . . . . .	312
4.133.4.25libxl_device_console_add . . . . .	312
4.133.4.26libxl_device_disk_add . . . . .	312
4.133.4.27libxl_device_disk_del . . . . .	312
4.133.4.28libxl_device_disk_getinfo . . . . .	312
4.133.4.29libxl_device_disk_list . . . . .	312
4.133.4.30libxl_device_disk_local_attach . . . . .	312
4.133.4.31libxl_device_disk_local_detach . . . . .	312
4.133.4.32libxl_device_net2_add . . . . .	312
4.133.4.33libxl_device_net2_del . . . . .	312
4.133.4.34libxl_device_net2_init . . . . .	312
4.133.4.35libxl_device_net2_list . . . . .	312
4.133.4.36libxl_device_nic_add . . . . .	312
4.133.4.37libxl_device_nic_del . . . . .	312
4.133.4.38libxl_device_nic_init . . . . .	312
4.133.4.39libxl_device_pci_add . . . . .	312
4.133.4.40libxl_device_pci_list_assignable . . . . .	312
4.133.4.41libxl_device_pci_list_assigned . . . . .	312
4.133.4.42libxl_device_pci_parse_bdf . . . . .	312
4.133.4.43libxl_device_pci_remove . . . . .	313
4.133.4.44libxl_device_pci_shutdown . . . . .	313
4.133.4.45libxl_device_vfb_add . . . . .	313
4.133.4.46libxl_device_vfb_clean_shutdown . . . . .	313
4.133.4.47libxl_device_vfb_hard_shutdown . . . . .	313
4.133.4.48libxl_device_vfb_init . . . . .	313
4.133.4.49libxl_device_vkb_add . . . . .	313
4.133.4.50libxl_device_vkb_clean_shutdown . . . . .	313
4.133.4.51libxl_device_vkb_hard_shutdown . . . . .	313
4.133.4.52libxl_device_vkb_init . . . . .	313
4.133.4.53libxl_domain_config_destroy . . . . .	313
4.133.4.54libxl_domain_core_dump . . . . .	313
4.133.4.55libxl_domain_create_new . . . . .	313
4.133.4.56libxl_domain_create_restore . . . . .	313
4.133.4.57libxl_domain_destroy . . . . .	313
4.133.4.58libxl_domain_info . . . . .	313
4.133.4.59libxl_domain_need_memory . . . . .	313
4.133.4.60libxl_domain_pause . . . . .	313
4.133.4.61libxl_domain_preserve . . . . .	313

4.133.4.62libxl_domain_rename . . . . .	313
4.133.4.63libxl_domain_resume . . . . .	313
4.133.4.64libxl_domain_setmaxmem . . . . .	313
4.133.4.65libxl_domain_shutdown . . . . .	314
4.133.4.66libxl_domain_suspend . . . . .	314
4.133.4.67libxl_domain_unpause . . . . .	314
4.133.4.68libxl_event_get_disk_eject_info . . . . .	314
4.133.4.69libxl_event_get_domain_death_info . . . . .	314
4.133.4.70libxl_file_reference_destroy . . . . .	314
4.133.4.71libxl_free_event . . . . .	314
4.133.4.72libxl_free_waiter . . . . .	314
4.133.4.73libxl_get_event . . . . .	314
4.133.4.74libxl_get_free_memory . . . . .	314
4.133.4.75libxl_get_freecpus . . . . .	314
4.133.4.76libxl_get_max_cpus . . . . .	314
4.133.4.77libxl_get_memory_target . . . . .	314
4.133.4.78libxl_get_physinfo . . . . .	314
4.133.4.79libxl_get_sched_id . . . . .	314
4.133.4.80libxl_get_topologyinfo . . . . .	314
4.133.4.81libxl_get_version_info . . . . .	314
4.133.4.82libxl_get_wait_fd . . . . .	314
4.133.4.83libxl_init_build_info . . . . .	315
4.133.4.84libxl_init_create_info . . . . .	315
4.133.4.85libxl_init_dm_info . . . . .	315
4.133.4.86libxl_key_value_list_destroy . . . . .	315
4.133.4.87libxl_libdir_path . . . . .	315
4.133.4.88libxl_libexec_path . . . . .	315
4.133.4.89libxl_list_cpupool . . . . .	315
4.133.4.90libxl_list_domain . . . . .	315
4.133.4.91libxl_list_nics . . . . .	315
4.133.4.92libxl_list_vcpu . . . . .	315
4.133.4.93libxl_list_vm . . . . .	315
4.133.4.94libxl_lock_dir_path . . . . .	315
4.133.4.95libxl_primary_console_exec . . . . .	315
4.133.4.96libxl_private_bindir_path . . . . .	315
4.133.4.97libxl_run_bootloader . . . . .	315
4.133.4.98libxl_sbindir_path . . . . .	315
4.133.4.99libxl_sched_credit_domain_get . . . . .	315
4.133.4.100libxl_sched_credit_domain_set . . . . .	315
4.133.4.101libxl_send_debug_keys . . . . .	315

4.133.4.101	<a href="#">libxl_send_sysrq</a>	316
4.133.4.102	<a href="#">libxl_send_trigger</a>	316
4.133.4.103	<a href="#">libxl_set_memory_target</a>	316
4.133.4.104	<a href="#">libxl_set_vcpuaffinity</a>	316
4.133.4.105	<a href="#">libxl_set_vcpuonline</a>	316
4.133.4.106	<a href="#">libxl_sharedir_path</a>	316
4.133.4.107	<a href="#">libxl_stop_waiting</a>	316
4.133.4.108	<a href="#">libxl_string_list_destroy</a>	316
4.133.4.109	<a href="#">libxl_tmem_destroy</a>	316
4.133.4.110	<a href="#">libxl_tmem_freeable</a>	316
4.133.4.111	<a href="#">libxl_tmem_freeze</a>	316
4.133.4.112	<a href="#">libxl_tmem_list</a>	316
4.133.4.113	<a href="#">libxl_tmem_set</a>	316
4.133.4.114	<a href="#">libxl_tmem_shared_auth</a>	316
4.133.4.115	<a href="#">libxl_tmem_thaw</a>	316
4.133.4.116	<a href="#">libxl_userdata_retrieve</a>	316
4.133.4.117	<a href="#">libxl_userdata_store</a>	316
4.133.4.118	<a href="#">libxl_vm_get_start_time</a>	316
4.133.4.119	<a href="#">libxl_vncviewer_exec</a>	316
4.133.4.120	<a href="#">libxl_wait_for_disk_ejects</a>	316
4.133.4.121	<a href="#">libxl_wait_for_domain_death</a>	316
4.133.4.122	<a href="#">libxl_wait_for_free_memory</a>	316
4.133.4.123	<a href="#">libxl_wait_for_memory_target</a>	316
4.133.4.124	<a href="#">libxl_xen_config_dir_path</a>	316
4.133.4.125	<a href="#">libxl_xen_console_read_finish</a>	317
4.133.4.126	<a href="#">libxl_xen_console_read_line</a>	317
4.133.4.127	<a href="#">libxl_xen_console_read_start</a>	317
4.133.4.128	<a href="#">libxl_xen_script_dir_path</a>	317
4.133.4.129	<a href="#">libxl_xenfirmware_dir_path</a>	317
4.134	<a href="#">xen/tools/libxl/libxl.idl File Reference</a>	317
4.135	<a href="#">xen/tools/libxl/libxl_blkmap2.c File Reference</a>	317
4.135.1	<a href="#">Function Documentation</a>	317
4.135.1.1	<a href="#">libxl__blkmap_devpath</a>	317
4.135.1.2	<a href="#">libxl__blkmap_enabled</a>	317
4.136	<a href="#">xen/tools/libxl/libxl_create.c File Reference</a>	317
4.136.1	<a href="#">Function Documentation</a>	318
4.136.1.1	<a href="#">init_console_info</a>	318
4.136.1.2	<a href="#">libxl__domain_build</a>	318
4.136.1.3	<a href="#">libxl__domain_make</a>	318
4.136.1.4	<a href="#">libxl__domain_config_destroy</a>	318



4.136.1.5 libxl_domain_create_new . . . . .	318
4.136.1.6 libxl_domain_create_restore . . . . .	318
4.136.1.7 libxl_init_build_info . . . . .	318
4.136.1.8 libxl_init_create_info . . . . .	318
4.136.1.9 libxl_init_dm_info . . . . .	318
4.137 xen/tools/libxl/libxl_device.c File Reference . . . . .	318
4.137.1 Function Documentation . . . . .	319
4.137.1.1 libxl__device_backend_path . . . . .	319
4.137.1.2 libxl__device_del . . . . .	319
4.137.1.3 libxl__device_destroy . . . . .	319
4.137.1.4 libxl__device_disk_dev_number . . . . .	319
4.137.1.5 libxl__device_disk_string_of_backend . . . . .	319
4.137.1.6 libxl__device_disk_string_of_format . . . . .	319
4.137.1.7 libxl__device_frontend_path . . . . .	319
4.137.1.8 libxl__device_generic_add . . . . .	319
4.137.1.9 libxl__device_physdisk_major_minor . . . . .	319
4.137.1.10 libxl__devices_destroy . . . . .	319
4.137.1.11 libxl__wait_for_backend . . . . .	319
4.137.1.12 libxl__wait_for_device_model . . . . .	319
4.138 xen/tools/libxl/libxl_dm.c File Reference . . . . .	320
4.138.1 Function Documentation . . . . .	320
4.138.1.1 libxl__confirm_device_model_startup . . . . .	320
4.138.1.2 libxl__create_device_model . . . . .	320
4.138.1.3 libxl__create_xenpv_qemu . . . . .	320
4.138.1.4 libxl__destroy_device_model . . . . .	320
4.138.1.5 libxl__need_xenpv_qemu . . . . .	320
4.139 xen/tools/libxl/libxl_dom.c File Reference . . . . .	321
4.139.1 Function Documentation . . . . .	321
4.139.1.1 libxl__build_hvm . . . . .	321
4.139.1.2 libxl__build_post . . . . .	322
4.139.1.3 libxl__build_pre . . . . .	322
4.139.1.4 libxl__build_pv . . . . .	322
4.139.1.5 libxl__domain_is_hvm . . . . .	322
4.139.1.6 libxl__domain_restore_common . . . . .	322
4.139.1.7 libxl__domain_save_device_model . . . . .	322
4.139.1.8 libxl__domain_shutdown_reason . . . . .	322
4.139.1.9 libxl__domain_suspend_common . . . . .	322
4.139.1.10 libxl__userdata_destroyall . . . . .	322
4.139.1.11 libxl__uuid2string . . . . .	322
4.139.1.12 libxl__userdata_retrieve . . . . .	322

4.139.1.13libxl_userdata_store . . . . .	322
4.140xen/tools/libxl/libxl_internal.c File Reference . . . . .	322
4.140.1 Function Documentation . . . . .	323
4.140.1.1 libxl__abs_path . . . . .	323
4.140.1.2 libxl__calloc . . . . .	323
4.140.1.3 libxl__dirname . . . . .	323
4.140.1.4 libxl__error_set . . . . .	323
4.140.1.5 libxl__file_reference_map . . . . .	323
4.140.1.6 libxl__file_reference_unmap . . . . .	323
4.140.1.7 libxl__free_all . . . . .	323
4.140.1.8 libxl__log . . . . .	323
4.140.1.9 libxl__logv . . . . .	323
4.140.1.10libxl__ptr_add . . . . .	323
4.140.1.11libxl__sprintf . . . . .	323
4.140.1.12libxl__strdup . . . . .	323
4.140.1.13libxl__zalloc . . . . .	323
4.141xen/tools/libxl/libxl_internal.h File Reference . . . . .	323
4.141.1 Macro Definition Documentation . . . . .	326
4.141.1.1 _hidden . . . . .	326
4.141.1.2 _protected . . . . .	326
4.141.1.3 ARRAY_SIZE . . . . .	326
4.141.1.4 AUTO_PHP_SLOT . . . . .	326
4.141.1.5 is_valid_device_kind . . . . .	326
4.141.1.6 LIBXL__LOG . . . . .	326
4.141.1.7 LIBXL__LOG_DEBUG . . . . .	326
4.141.1.8 LIBXL__LOG_ERRNO . . . . .	326
4.141.1.9 LIBXL__LOG_ERRNOVAL . . . . .	326
4.141.1.10LIBXL__LOG_ERROR . . . . .	326
4.141.1.11LIBXL__LOG_INFO . . . . .	326
4.141.1.12LIBXL__LOG_WARNING . . . . .	326
4.141.1.13LIBXL__LOGGING_ENABLED . . . . .	326
4.141.1.14LIBXL_DESTROY_TIMEOUT . . . . .	326
4.141.1.15LIBXL_DEVICE_MODEL_START_TIMEOUT . . . . .	326
4.141.1.16LIBXL_HVM_EXTRA_MEMORY . . . . .	327
4.141.1.17LIBXL_INIT_GC . . . . .	327
4.141.1.18LIBXL_MAXMEM_CONSTANT . . . . .	327
4.141.1.19LIBXL_MIN_DOM0_MEM . . . . .	327
4.141.1.20LIBXL_PV_EXTRA_MEMORY . . . . .	327
4.141.1.21LIBXL_XENCONSOLE_LIMIT . . . . .	327
4.141.1.22LIBXL_XENCONSOLE_PROTOCOL . . . . .	327

4.141.1.23	PCI_BAR_IO . . . . .	327
4.141.1.24	PRINTF_ATTRIBUTE . . . . .	327
4.141.1.25	PROC_PCI_NUM_RESOURCES . . . . .	327
4.141.1.26	QEMU_SIGNATURE . . . . .	327
4.141.1.27	SAVEFILE . . . . .	327
4.141.1.28	STUBDOM_CONSOLE_LOGGING . . . . .	327
4.141.1.29	STUBDOM_CONSOLE_RESTORE . . . . .	327
4.141.1.30	STUBDOM_CONSOLE_SAVE . . . . .	327
4.141.1.31	STUBDOM_CONSOLE_SERIAL . . . . .	327
4.141.1.32	STUBDOM_SPECIAL_CONSOLES . . . . .	327
4.141.1.33	SYSFS_PCI_DEV . . . . .	327
4.141.1.34	SYSFS_PCIBACK_DRIVER . . . . .	327
4.141.1.35	XC_PCI_BDF . . . . .	327
4.141.1.36	XENSTORE_PID_FILE . . . . .	327
4.141.2	Enumeration Type Documentation . . . . .	327
4.141.2.1	libxl__device_kinds . . . . .	327
4.141.3	Function Documentation . . . . .	328
4.141.3.1	libxl__abs_path . . . . .	328
4.141.3.2	libxl__blktap_devpath . . . . .	328
4.141.3.3	libxl__blktap_enabled . . . . .	328
4.141.3.4	libxl__build_hvm . . . . .	328
4.141.3.5	libxl__build_post . . . . .	328
4.141.3.6	libxl__build_pre . . . . .	328
4.141.3.7	libxl__build_pv . . . . .	328
4.141.3.8	libxl__calloc . . . . .	328
4.141.3.9	libxl__confirm_device_model_startup . . . . .	328
4.141.3.10	libxl__cpupoolid_to_name . . . . .	328
4.141.3.11	libxl__create_device_model . . . . .	328
4.141.3.12	libxl__create_xenpv_qemu . . . . .	328
4.141.3.13	libxl__destroy_device_model . . . . .	328
4.141.3.14	libxl__device_backend_path . . . . .	328
4.141.3.15	libxl__device_del . . . . .	328
4.141.3.16	libxl__device_destroy . . . . .	328
4.141.3.17	libxl__device_disk_dev_number . . . . .	328
4.141.3.18	libxl__device_disk_string_of_backend . . . . .	328
4.141.3.19	libxl__device_disk_string_of_format . . . . .	328
4.141.3.20	libxl__device_frontend_path . . . . .	328
4.141.3.21	libxl__device_generic_add . . . . .	328
4.141.3.22	libxl__device_pci_add . . . . .	328
4.141.3.23	libxl__device_physdisk_major_minor . . . . .	329

4.141.3.24libxl__devices_destroy . . . . .	329
4.141.3.25libxl__dirname . . . . .	329
4.141.3.26libxl__domain_build . . . . .	329
4.141.3.27libxl__domain_is_hvm . . . . .	329
4.141.3.28libxl__domain_make . . . . .	329
4.141.3.29libxl__domain_restore_common . . . . .	329
4.141.3.30libxl__domain_save_device_model . . . . .	329
4.141.3.31libxl__domain_shutdown_reason . . . . .	329
4.141.3.32libxl__domain_suspend_common . . . . .	329
4.141.3.33libxl__domid_to_name . . . . .	329
4.141.3.34libxl__error_set . . . . .	329
4.141.3.35libxl__exec . . . . .	329
4.141.3.36libxl__file_reference_map . . . . .	329
4.141.3.37libxl__file_reference_unmap . . . . .	329
4.141.3.38libxl__free_all . . . . .	329
4.141.3.39libxl__log . . . . .	329
4.141.3.40libxl__log_child_exitstatus . . . . .	329
4.141.3.41libxl__logv . . . . .	329
4.141.3.42libxl__need_xenpv_qemu . . . . .	329
4.141.3.43libxl__ptr_add . . . . .	329
4.141.3.44libxl__spawn_check . . . . .	329
4.141.3.45libxl__spawn_detach . . . . .	329
4.141.3.46libxl__spawn_spawn . . . . .	329
4.141.3.47libxl__sprintf . . . . .	329
4.141.3.48libxl__strdup . . . . .	330
4.141.3.49libxl__userdata_destroyall . . . . .	330
4.141.3.50libxl__uuid2string . . . . .	330
4.141.3.51libxl__wait_for_backend . . . . .	330
4.141.3.52libxl__wait_for_device_model . . . . .	330
4.141.3.53libxl__xs_directory . . . . .	330
4.141.3.54libxl__xs_get_dompath . . . . .	330
4.141.3.55libxl__xs_kvs_of_flexarray . . . . .	330
4.141.3.56libxl__xs_read . . . . .	330
4.141.3.57libxl__xs_write . . . . .	330
4.141.3.58libxl__xs_writev . . . . .	330
4.141.3.59libxl__zalloc . . . . .	330
4.142xen/tools/libxl/libxl_utils.c File Reference . . . . .	330
4.142.1 Macro Definition Documentation . . . . .	331
4.142.1.1 QEMU_VERSION_STR . . . . .	331
4.142.1.2 READ_WRITE_EXACTLY . . . . .	331

4.142.2 Function Documentation . . . . .	332
4.142.2.1 libxl__cpupoolid_to_name . . . . .	332
4.142.2.2 libxl__domid_to_name . . . . .	332
4.142.2.3 libxl_basename . . . . .	332
4.142.2.4 libxl_check_device_model_version . . . . .	332
4.142.2.5 libxl_cpumap_alloc . . . . .	332
4.142.2.6 libxl_cpumap_destroy . . . . .	332
4.142.2.7 libxl_cpumap_alloc . . . . .	332
4.142.2.8 libxl_cpumap_destroy . . . . .	332
4.142.2.9 libxl_cpumap_reset . . . . .	332
4.142.2.10 libxl_cpumap_set . . . . .	332
4.142.2.11 libxl_cpumap_test . . . . .	332
4.142.2.12 libxl_cpupoolid_to_name . . . . .	332
4.142.2.13 libxl_create_logfile . . . . .	332
4.142.2.14 libxl_ctx_postfork . . . . .	332
4.142.2.15 libxl_devid_to_device_disk . . . . .	332
4.142.2.16 libxl_devid_to_device_net2 . . . . .	332
4.142.2.17 libxl_devid_to_device_nic . . . . .	332
4.142.2.18 libxl_domid_to_name . . . . .	332
4.142.2.19 libxl_fork . . . . .	332
4.142.2.20 libxl_get_max_cpus . . . . .	332
4.142.2.21 libxl_get_required_shadow_memory . . . . .	332
4.142.2.22 libxl_get_stubdom_id . . . . .	332
4.142.2.23 libxl_is_stubdom . . . . .	333
4.142.2.24 libxl_mac_to_device_nic . . . . .	333
4.142.2.25 libxl_name_to_cpupoolid . . . . .	333
4.142.2.26 libxl_name_to_domid . . . . .	333
4.142.2.27 libxl_name_to_schedid . . . . .	333
4.142.2.28 libxl_pipe . . . . .	333
4.142.2.29 libxl_read_file_contents . . . . .	333
4.142.2.30 libxl_schedid_to_name . . . . .	333
4.142.2.31 libxl_string_to_backend . . . . .	333
4.142.2.32 libxl_strtomac . . . . .	333
4.143 xen/tools/libxl/libxl_uuid.h File Reference . . . . .	333
4.143.1 Macro Definition Documentation . . . . .	333
4.143.1.1 LIBXL__UUID_BYTES . . . . .	333
4.143.1.2 LIBXL__UUID_FMT . . . . .	333
4.144 xen/tools/libxl/libxlu_cfg_y.c File Reference . . . . .	333
4.144.1 Macro Definition Documentation . . . . .	336
4.144.1.1 YY_ . . . . .	336

4.144.1.2 YY_LOCATION_PRINT . . . . .	336
4.144.1.3 YY_REDUCE_PRINT . . . . .	336
4.144.1.4 YY_STACK_PRINT . . . . .	336
4.144.1.5 YY_SYMBOL_PRINT . . . . .	336
4.144.1.6 YYABORT . . . . .	336
4.144.1.7 YYACCEPT . . . . .	336
4.144.1.8 YYBACKUP . . . . .	336
4.144.1.9 YYBISON . . . . .	336
4.144.1.10 YYBISON_VERSION . . . . .	336
4.144.1.11 YYCASE_ . . . . .	336
4.144.1.12 ychar . . . . .	336
4.144.1.13 yclearin . . . . .	336
4.144.1.14 YYCOPY . . . . .	336
4.144.1.15 YYCOPY_NEEDED . . . . .	337
4.144.1.16 ydebug . . . . .	337
4.144.1.17 YYDEBUG . . . . .	337
4.144.1.18 YYDPRINTF . . . . .	337
4.144.1.19 YYEMPTY . . . . .	337
4.144.1.20 YYEOF . . . . .	337
4.144.1.21 YYERRCODE . . . . .	337
4.144.1.22 yerrok . . . . .	337
4.144.1.23 yerror . . . . .	337
4.144.1.24 YERROR . . . . .	337
4.144.1.25 YERROR_VERBOSE . . . . .	337
4.144.1.26 YFAIL . . . . .	337
4.144.1.27 YYFINAL . . . . .	337
4.144.1.28 YFREE . . . . .	337
4.144.1.29 YYID . . . . .	337
4.144.1.30 YYINITDEPTH . . . . .	337
4.144.1.31 YYLAST . . . . .	337
4.144.1.32 ylex . . . . .	337
4.144.1.33 YYLEX . . . . .	337
4.144.1.34 YYLEX_PARAM . . . . .	337
4.144.1.35 yllloc . . . . .	337
4.144.1.36 YYLLOC_DEFAULT . . . . .	337
4.144.1.37 YYLSP_NEEDED . . . . .	338
4.144.1.38 ylttype . . . . .	338
4.144.1.39 YYLTYPE_IS_DECLARED . . . . .	338
4.144.1.40 YYLTYPE_IS_TRIVIAL . . . . .	338
4.144.1.41 yylval . . . . .	338

4.144.1.42YYMALLOC . . . . .	338
4.144.1.43YYMAXDEPTH . . . . .	338
4.144.1.44YYMAXUTOK . . . . .	338
4.144.1.45yynerrs . . . . .	338
4.144.1.46YYNNTS . . . . .	338
4.144.1.47YYNRULES . . . . .	338
4.144.1.48YYNSTATES . . . . .	338
4.144.1.49YYNTOKENS . . . . .	338
4.144.1.50YYPACT_NINF . . . . .	338
4.144.1.51yypact_value_is_default . . . . .	338
4.144.1.52yparse . . . . .	338
4.144.1.53YYPOPSTACK . . . . .	338
4.144.1.54YYPULL . . . . .	338
4.144.1.55YYPURE . . . . .	338
4.144.1.56YYPUSH . . . . .	338
4.144.1.57YYRECOVERING . . . . .	338
4.144.1.58YYRHSLOC . . . . .	338
4.144.1.59YYSIZE_MAXIMUM . . . . .	338
4.144.1.60YYSIZE_T . . . . .	339
4.144.1.61YYSKELETON_NAME . . . . .	339
4.144.1.62YYSTACK_ALLOC . . . . .	339
4.144.1.63YYSTACK_ALLOC_MAXIMUM . . . . .	339
4.144.1.64YYSTACK_BYTES . . . . .	339
4.144.1.65YYSTACK_FREE . . . . .	339
4.144.1.66YYSTACK_GAP_MAXIMUM . . . . .	339
4.144.1.67YYSTACK_RELOCATE . . . . .	339
4.144.1.68yystype . . . . .	339
4.144.1.69YYSTYPE_IS_DECLARED . . . . .	339
4.144.1.70YYSTYPE_IS_TRIVIAL . . . . .	339
4.144.1.71YYSYNTAX_ERROR . . . . .	339
4.144.1.72YYTABLE_NINF . . . . .	339
4.144.1.73ytable_value_is_error . . . . .	339
4.144.1.74YYTERROR . . . . .	339
4.144.1.75YYTOKEN_TABLE . . . . .	339
4.144.1.76YYTOKENTYPE . . . . .	339
4.144.1.77YYTRANSLATE . . . . .	340
4.144.1.78YYUNDEFTOK . . . . .	340
4.144.1.79YYUSE . . . . .	340
4.144.2 Typedef Documentation . . . . .	340
4.144.2.1 YYLTYPE . . . . .	340

4.144.2.2 YYSTYPE . . . . .	340
4.144.2.3 yytype_int16 . . . . .	340
4.144.2.4 yytype_int8 . . . . .	340
4.144.2.5 yytype_uint16 . . . . .	340
4.144.2.6 yytype_uint8 . . . . .	340
4.144.3 Enumeration Type Documentation . . . . .	340
4.144.3.1 yytokentype . . . . .	340
4.144.4 Function Documentation . . . . .	340
4.144.4.1 yyparse . . . . .	340
4.144.4.2 yyparse . . . . .	340
4.145xen/tools/libxl/libxl_cfg_y.h File Reference . . . . .	340
4.145.1 Macro Definition Documentation . . . . .	341
4.145.1.1 yyltype . . . . .	341
4.145.1.2 YYLTYPE_IS_DECLARED . . . . .	341
4.145.1.3 YYLTYPE_IS_TRIVIAL . . . . .	341
4.145.1.4 yystype . . . . .	341
4.145.1.5 YYSTYPE_IS_DECLARED . . . . .	341
4.145.1.6 YYSTYPE_IS_TRIVIAL . . . . .	341
4.145.2 Typedef Documentation . . . . .	341
4.145.2.1 YYLTYPE . . . . .	341
4.145.2.2 YYSTYPE . . . . .	341
4.145.3 Enumeration Type Documentation . . . . .	341
4.145.3.1 yytokentype . . . . .	341
4.146xen/tools/libxl/xl_cmdimpl.c File Reference . . . . .	342
4.146.1 Macro Definition Documentation . . . . .	344
4.146.1.1 ADD_OPTDATA . . . . .	344
4.146.1.2 CHK_ERRNO . . . . .	344
4.146.1.3 DSTATE_INITIAL . . . . .	344
4.146.1.4 DSTATE_PHYSPATH . . . . .	344
4.146.1.5 DSTATE_RW . . . . .	344
4.146.1.6 DSTATE_TAP . . . . .	344
4.146.1.7 DSTATE_TERMINAL . . . . .	344
4.146.1.8 DSTATE_VIRTPATH . . . . .	344
4.146.1.9 DSTATE_VIRTTYPE . . . . .	344
4.146.1.10 LOG . . . . .	344
4.146.1.11 MUST . . . . .	344
4.146.1.12 OPTDATA_LEFT . . . . .	345
4.146.1.13 SAVEFILE_BYTEORDER_VALUE . . . . .	345
4.146.1.14 WITH_OPTDATA . . . . .	345
4.146.2 Function Documentation . . . . .	345



4.146.2.1 help . . . . .	345
4.146.2.2 main_blockattach . . . . .	345
4.146.2.3 main_blockdetach . . . . .	345
4.146.2.4 main_blocklist . . . . .	345
4.146.2.5 main_button_press . . . . .	345
4.146.2.6 main_cd_eject . . . . .	345
4.146.2.7 main_cd_insert . . . . .	345
4.146.2.8 main_console . . . . .	345
4.146.2.9 main_cpupoolcpuadd . . . . .	345
4.146.2.10main_cpupoolcpuremove . . . . .	345
4.146.2.11main_cpupoolcreate . . . . .	345
4.146.2.12main_cpupooldestroy . . . . .	345
4.146.2.13main_cpupoollist . . . . .	345
4.146.2.14main_cpupoolmigrate . . . . .	345
4.146.2.15main_cpupoolnumasplit . . . . .	345
4.146.2.16main_cpupoolrename . . . . .	345
4.146.2.17main_create . . . . .	346
4.146.2.18main_debug_keys . . . . .	346
4.146.2.19main_destroy . . . . .	346
4.146.2.20main_dmesg . . . . .	346
4.146.2.21main_domid . . . . .	346
4.146.2.22main_domname . . . . .	346
4.146.2.23main_dump_core . . . . .	346
4.146.2.24main_info . . . . .	346
4.146.2.25main_list . . . . .	346
4.146.2.26main_list_vm . . . . .	346
4.146.2.27main_memmax . . . . .	346
4.146.2.28main_memset . . . . .	346
4.146.2.29main_migrate . . . . .	346
4.146.2.30main_migrate_receive . . . . .	346
4.146.2.31main_network2attach . . . . .	346
4.146.2.32main_network2detach . . . . .	346
4.146.2.33main_network2list . . . . .	346
4.146.2.34main_networkattach . . . . .	346
4.146.2.35main_networkdetach . . . . .	346
4.146.2.36main_networklist . . . . .	346
4.146.2.37main_pause . . . . .	346
4.146.2.38main_pciattach . . . . .	346
4.146.2.39main_pcidetach . . . . .	346
4.146.2.40main_pcilist . . . . .	346

4.146.2.41	<a href="#">main_pclist_assignable</a>	346
4.146.2.42	<a href="#">main_reboot</a>	346
4.146.2.43	<a href="#">main_rename</a>	346
4.146.2.44	<a href="#">main_restore</a>	346
4.146.2.45	<a href="#">main_save</a>	347
4.146.2.46	<a href="#">main_sched_credit</a>	347
4.146.2.47	<a href="#">main_shutdown</a>	347
4.146.2.48	<a href="#">main_sysrq</a>	347
4.146.2.49	<a href="#">main_tmem_destroy</a>	347
4.146.2.50	<a href="#">main_tmem_freeable</a>	347
4.146.2.51	<a href="#">main_tmem_freeze</a>	347
4.146.2.52	<a href="#">main_tmem_list</a>	347
4.146.2.53	<a href="#">main_tmem_set</a>	347
4.146.2.54	<a href="#">main_tmem_shared_auth</a>	347
4.146.2.55	<a href="#">main_tmem_thaw</a>	347
4.146.2.56	<a href="#">main_top</a>	347
4.146.2.57	<a href="#">main_trigger</a>	347
4.146.2.58	<a href="#">main_unpause</a>	347
4.146.2.59	<a href="#">main_uptime</a>	347
4.146.2.60	<a href="#">main_vcpulist</a>	347
4.146.2.61	<a href="#">main_vcpupin</a>	347
4.146.2.62	<a href="#">main_vcpu_set</a>	347
4.146.2.63	<a href="#">main_vncviewer</a>	347
4.146.3	Variable Documentation	347
4.146.3.1	<a href="#">ctx</a>	347
4.146.3.2	<a href="#">logfile</a>	347
4.147	<a href="#">xen/tools/python/xen/lowlevel/xl/xl.c</a> File Reference	348
4.147.1	Macro Definition Documentation	349
4.147.1.1	<a href="#">_INT_CONST</a>	349
4.147.1.2	<a href="#">_INT_CONST_LIBXL</a>	349
4.147.1.3	<a href="#">ARRAY_SIZE</a>	349
4.147.1.4	<a href="#">CLS</a>	349
4.147.1.5	<a href="#">Py_ssize_t</a>	349
4.147.1.6	<a href="#">PyMODINIT_FUNC</a>	349
4.147.2	Function Documentation	349
4.147.2.1	<a href="#">attrib__libxl_cpumap_get</a>	349
4.147.2.2	<a href="#">attrib__libxl_cpumap_set</a>	349
4.147.2.3	<a href="#">attrib__libxl_cpuid_policy_list_get</a>	349
4.147.2.4	<a href="#">attrib__libxl_cpuid_policy_list_set</a>	349
4.147.2.5	<a href="#">attrib__libxl_cpumap_get</a>	349

4.147.2.6 attrib__libxl_cpumap_set . . . . .	349
4.147.2.7 attrib__libxl_domain_build_state_ptr_get . . . . .	349
4.147.2.8 attrib__libxl_domain_build_state_ptr_set . . . . .	349
4.147.2.9 attrib__libxl_file_reference_get . . . . .	349
4.147.2.10attrib__libxl_file_reference_set . . . . .	350
4.147.2.11attrib__libxl_hwcap_get . . . . .	350
4.147.2.12attrib__libxl_hwcap_set . . . . .	350
4.147.2.13attrib__libxl_key_value_list_get . . . . .	350
4.147.2.14attrib__libxl_key_value_list_set . . . . .	350
4.147.2.15attrib__libxl_mac_get . . . . .	350
4.147.2.16attrib__libxl_mac_set . . . . .	350
4.147.2.17attrib__libxl_string_list_get . . . . .	350
4.147.2.18attrib__libxl_string_list_set . . . . .	350
4.147.2.19attrib__libxl_uuid_get . . . . .	350
4.147.2.20attrib__libxl_uuid_set . . . . .	350
4.147.2.21attrib__struct_in_addr_get . . . . .	350
4.147.2.22attrib__struct_in_addr_set . . . . .	350
4.147.2.23attrib__void_ptr_get . . . . .	350
4.147.2.24attrib__void_ptr_set . . . . .	350
4.147.2.25genwrap__ll_get . . . . .	350
4.147.2.26genwrap__ll_set . . . . .	350
4.147.2.27genwrap__obj_init . . . . .	350
4.147.2.28genwrap__string_get . . . . .	350
4.147.2.29genwrap__string_set . . . . .	350
4.147.2.30genwrap__ull_get . . . . .	350
4.147.2.31genwrap__ull_set . . . . .	350
4.147.2.32nitxl . . . . .	350
4.148xen/tools/vtpm_manager/crypto/crypto.h File Reference . . . . .	350
4.148.1 Macro Definition Documentation . . . . .	352
4.148.1.1 CRYPTO_ALGORITHM_RSA . . . . .	352
4.148.1.2 CRYPTO_ES_NONE . . . . .	352
4.148.1.3 CRYPTO_ES_RSAESOAEP_SHA1_MGF1 . . . . .	352
4.148.1.4 CRYPTO_ES_RSAESPKCSv15 . . . . .	352
4.148.1.5 CRYPTO_MAX_RSA_KEY_SIZE . . . . .	352
4.148.1.6 CRYPTO_MAX_SIG_SIZE . . . . .	352
4.148.1.7 CRYPTO_SS_NONE . . . . .	352
4.148.1.8 CRYPTO_SS_RSASSAPKCS1v15_DER . . . . .	352
4.148.1.9 CRYPTO_SS_RSASSAPKCS1v15_SHA1 . . . . .	352
4.148.1.10OAEP_P . . . . .	352
4.148.1.11OAEP_P_SIZE . . . . .	352

4.148.2 Typedef Documentation . . . . .	352
4.148.2.1 CRYPTO_INFO . . . . .	352
4.148.3 Function Documentation . . . . .	352
4.148.3.1 Crypto_Exit . . . . .	352
4.148.3.2 Crypto_GetRandom . . . . .	352
4.148.3.3 Crypto_HMAC . . . . .	352
4.148.3.4 Crypto_HMAC_buf . . . . .	352
4.148.3.5 Crypto_Init . . . . .	352
4.148.3.6 Crypto_RSABuildCryptoInfo . . . . .	352
4.148.3.7 Crypto_RSABuildCryptoInfoPublic . . . . .	352
4.148.3.8 Crypto_RSACreateKey . . . . .	352
4.148.3.9 Crypto_RSACryptoInfoFree . . . . .	352
4.148.3.10 Crypto_RSADec . . . . .	353
4.148.3.11 Crypto_RSAAnc . . . . .	353
4.148.3.12 Crypto_RSAPackCryptoInfo . . . . .	353
4.148.3.13 Crypto_RSASign . . . . .	353
4.148.3.14 Crypto_RSAUnpackCryptoInfo . . . . .	353
4.148.3.15 Crypto_RSASignVerify . . . . .	353
4.148.3.16 Crypto_SHA1Complete . . . . .	353
4.148.3.17 Crypto_SHA1Full . . . . .	353
4.148.3.18 Crypto_SHA1Full_buf . . . . .	353
4.148.3.19 Crypto_SHA1Start . . . . .	353
4.148.3.20 Crypto_SHA1Update . . . . .	353
4.148.3.21 RSA_sign_DER . . . . .	353
4.148.3.22 RSA_verify_DER . . . . .	353
4.149 xen/tools/vtpm_manager/crypto/rsa.c File Reference . . . . .	353
4.149.1 Function Documentation . . . . .	354
4.149.1.1 Crypto_RSABuildCryptoInfo . . . . .	354
4.149.1.2 Crypto_RSABuildCryptoInfoPublic . . . . .	354
4.149.1.3 Crypto_RSACreateKey . . . . .	354
4.149.1.4 Crypto_RSACryptoInfoFree . . . . .	354
4.149.1.5 Crypto_RSADec . . . . .	354
4.149.1.6 Crypto_RSAAnc . . . . .	354
4.149.1.7 Crypto_RSAPackCryptoInfo . . . . .	354
4.149.1.8 Crypto_RSASign . . . . .	354
4.149.1.9 Crypto_RSAUnpackCryptoInfo . . . . .	354
4.149.1.10 Crypto_RSASignVerify . . . . .	354
4.150 xen/tools/vtpm_manager/crypto/sym_crypto.c File Reference . . . . .	354
4.150.1 Typedef Documentation . . . . .	355
4.150.1.1 crypt_op_type_t . . . . .	355

4.150.2 Enumeration Type Documentation . . . . .	355
4.150.2.1 crypt_op_type_t . . . . .	355
4.150.3 Function Documentation . . . . .	355
4.150.3.1 Crypto_symcrypto_decrypt . . . . .	355
4.150.3.2 Crypto_symcrypto_encrypt . . . . .	355
4.150.3.3 Crypto_symcrypto_freekey . . . . .	355
4.150.3.4 Crypto_symcrypto_genkey . . . . .	355
4.150.3.5 Crypto_symcrypto_initkey . . . . .	355
4.150.3.6 openssl_symcrypto_op . . . . .	355
4.150.4 Variable Documentation . . . . .	356
4.150.4.1 SYM_CIPHER . . . . .	356
4.150.4.2 ZERO_IV . . . . .	356
4.151 xen/tools/vtpm_manager/manager/dmictl.c File Reference . . . . .	356
4.151.1 Macro Definition Documentation . . . . .	356
4.151.1.1 TPM_EMULATOR_PATH . . . . .	356
4.151.2 Function Documentation . . . . .	356
4.151.2.1 close_dmi . . . . .	356
4.151.2.2 free_dmi . . . . .	356
4.151.2.3 init_dmi . . . . .	356
4.151.2.4 VTPM_Handle_Close_DMI . . . . .	356
4.151.2.5 VTPM_Handle_Delete_DMI . . . . .	356
4.151.2.6 VTPM_Handle_New_DMI . . . . .	356
4.152 xen/tools/vtpm_manager/manager/securestorage.c File Reference . . . . .	357
4.152.1 Function Documentation . . . . .	357
4.152.1.1 envelope_decrypt . . . . .	357
4.152.1.2 envelope_encrypt . . . . .	357
4.152.1.3 symkey_decrypt . . . . .	357
4.152.1.4 symkey_encrypt . . . . .	357
4.152.1.5 VTPM_Handle_Get_NVM_Size . . . . .	357
4.152.1.6 VTPM_Handle_Load_Key . . . . .	358
4.152.1.7 VTPM_Handle_Load_NVM . . . . .	358
4.152.1.8 VTPM_Handle_Save_Key . . . . .	358
4.152.1.9 VTPM_Handle_Save_NVM . . . . .	358
4.152.1.10 VTPM_LoadManagerData . . . . .	358
4.152.1.11 VTPM_SaveManagerData . . . . .	358
4.153 xen/tools/vtpm_manager/manager/vtpm_ipc.c File Reference . . . . .	358
4.153.1 Function Documentation . . . . .	358
4.153.1.1 vtpm_ipc_close . . . . .	358
4.153.1.2 vtpm_ipc_create . . . . .	358
4.153.1.3 vtpm_ipc_init . . . . .	358

4.153.1.4 vtpm_ipc_read . . . . .	358
4.153.1.5 vtpm_ipc_write . . . . .	358
4.153.2 Variable Documentation . . . . .	358
4.153.2.1 IPC_QUIT_FLAG . . . . .	358
4.154xen/tools/vtpm_manager/manager/vtpm_ipc.h File Reference . . . . .	359
4.154.1 Macro Definition Documentation . . . . .	359
4.154.1.1 VTPM_IPC_CLOSED . . . . .	359
4.154.2 Typedef Documentation . . . . .	359
4.154.2.1 vtpm_ipc_handle_t . . . . .	359
4.154.3 Function Documentation . . . . .	359
4.154.3.1 vtpm_ipc_close . . . . .	359
4.154.3.2 vtpm_ipc_create . . . . .	359
4.154.3.3 vtpm_ipc_init . . . . .	359
4.154.3.4 vtpm_ipc_read . . . . .	359
4.154.3.5 vtpm_ipc_write . . . . .	359
4.154.4 Variable Documentation . . . . .	359
4.154.4.1 IPC_QUIT_FLAG . . . . .	360
4.155xen/tools/vtpm_manager/manager/vtpm_manager.c File Reference . . . . .	360
4.155.1 Macro Definition Documentation . . . . .	360
4.155.1.1 RLISTSZ . . . . .	360
4.155.2 Function Documentation . . . . .	360
4.155.2.1 VTPM_Create_Manager . . . . .	360
4.155.2.2 VTPM_FlushResources . . . . .	360
4.155.2.3 VTPM_Init_Manager . . . . .	360
4.155.2.4 VTPM_Stop_Manager . . . . .	360
4.155.3 Variable Documentation . . . . .	360
4.155.3.1 SRK_AUTH . . . . .	360
4.155.3.2 vtpm_globals . . . . .	361
4.156xen/tools/vtpm_manager/manager/vtpm_manager.h File Reference . . . . .	361
4.156.1 Macro Definition Documentation . . . . .	362
4.156.1.1 COMMAND_BUFFER_SIZE . . . . .	362
4.156.1.2 TPM_EMULATOR_PATH . . . . .	362
4.156.1.3 VTPM_BE_FNAME . . . . .	362
4.156.1.4 VTPM_COMMAND_HEADER_SIZE_CLT . . . . .	362
4.156.1.5 VTPM_COMMAND_HEADER_SIZE_SRV . . . . .	362
4.156.1.6 VTPM_DUMMY_RX_BE_FNAME . . . . .	362
4.156.1.7 VTPM_DUMMY_TX_BE_FNAME . . . . .	362
4.156.1.8 VTPM_FAIL . . . . .	362
4.156.1.9 VTPM_FORBIDDEN . . . . .	362
4.156.1.10VTPM_INVALID_REQUEST . . . . .	362

4.156.1.11	VTPM_ORD_BASE	362
4.156.1.12	VTPM_ORD_CLOSE	362
4.156.1.13	VTPM_ORD_DELETE	362
4.156.1.14	VTPM_ORD_GET_MIG_KEY	362
4.156.1.15	VTPM_ORD_GETNVMSIZE	362
4.156.1.16	VTPM_ORD_LOAD_MIG_KEY	362
4.156.1.17	VTPM_ORD_LOADKEY	362
4.156.1.18	VTPM_ORD_LOADNVM	362
4.156.1.19	VTPM_ORD_MIGRATE_IN	362
4.156.1.20	VTPM_ORD_MIGRATE_OUT	362
4.156.1.21	VTPM_ORD_OPEN	362
4.156.1.22	VTPM_ORD_SAVEKEY	362
4.156.1.23	VTPM_ORD_SAVENVM	362
4.156.1.24	VTPM_ORD_TPMCOMMAND	362
4.156.1.25	VTPM_PRIV_BASE	362
4.156.1.26	VTPM_PRIV_MASK	362
4.156.1.27	VTPM_RESTORE_CONTEXT_FAILED	362
4.156.1.28	VTPM_RX_HP_FNAME	363
4.156.1.29	VTPM_RX_TPM_FNAME	363
4.156.1.30	VTPM_RX_VTPM_FNAME	363
4.156.1.31	VTPM_SUCCESS	363
4.156.1.32	VTPM_TAG_REQ	363
4.156.1.33	VTPM_TAG_RSP	363
4.156.1.34	VTPM_TX_HP_FNAME	363
4.156.1.35	VTPM_TX_TPM_FNAME	363
4.156.1.36	VTPM_TX_VTPM_FNAME	363
4.156.1.37	VTPM_TYPE_HVM	363
4.156.1.38	VTPM_TYPE_HVM_STRING	363
4.156.1.39	VTPM_TYPE_MIGRATABLE	363
4.156.1.40	VTPM_TYPE_MIGRATED	363
4.156.1.41	VTPM_TYPE_NON_MIGRATABLE	363
4.156.1.42	VTPM_TYPE_PVM	363
4.156.1.43	VTPM_TYPE_PVM_STRING	363
4.156.1.44	VTPM_UNSUPPORTED	363
4.157	xen/tools/vtpm_manager/manager/vtpm_manager_handler.c File Reference	363
4.157.1	Macro Definition Documentation	364
4.157.1.1	vtpmhandlerlogerror	364
4.157.1.2	vtpmhandlerloginfo	364
4.157.1.3	vtpmhandlerloginfomore	364
4.157.2	Function Documentation	364

4.157.2.1 vtpm_manager_handle_tpm_cmd . . . . .	364
4.157.2.2 vtpm_manager_handle_vtpm_cmd . . . . .	364
4.157.2.3 VTPM_Manager_Handler . . . . .	364
4.157.3 Variable Documentation . . . . .	364
4.157.3.1 HANDLER_QUIT_FLAG . . . . .	364
4.158xen/tools/vtpm_manager/manager/vtpmd.c File Reference . . . . .	365
4.158.1 Function Documentation . . . . .	365
4.158.1.1 main . . . . .	365
4.158.1.2 signal_handler . . . . .	365
4.158.1.3 VTPM_Close_DMI_Extra . . . . .	365
4.158.1.4 vtpm_manager_thread . . . . .	365
4.158.1.5 VTPM_New_DMI_Extra . . . . .	365
4.158.2 Variable Documentation . . . . .	365
4.158.2.1 ctl_c_handler . . . . .	365
4.159xen/tools/vtpm_manager/manager/vtpmpriv.h File Reference . . . . .	366
4.159.1 Macro Definition Documentation . . . . .	367
4.159.1.1 DMI_NVM_FILE . . . . .	367
4.159.1.2 STATE_FILE . . . . .	367
4.159.1.3 VTPM_CTL_DM . . . . .	367
4.159.1.4 VTPM_MANAGER_GEN . . . . .	367
4.159.2 Typedef Documentation . . . . .	367
4.159.2.1 VTPM_DMI_RESOURCE . . . . .	367
4.159.2.2 VTPM_GLOBALS . . . . .	367
4.159.2.3 VTPM_MIGKEY_LIST . . . . .	367
4.159.3 Function Documentation . . . . .	367
4.159.3.1 close_dmi . . . . .	367
4.159.3.2 envelope_decrypt . . . . .	367
4.159.3.3 envelope_encrypt . . . . .	367
4.159.3.4 free_dmi . . . . .	367
4.159.3.5 init_dmi . . . . .	367
4.159.3.6 VTPM_Close_DMI_Extra . . . . .	367
4.159.3.7 VTPM_Handle_Close_DMI . . . . .	367
4.159.3.8 VTPM_Handle_Delete_DMI . . . . .	367
4.159.3.9 VTPM_Handle_Get_Migration_key . . . . .	368
4.159.3.10VTPM_Handle_Get_NVM_Size . . . . .	368
4.159.3.11VTPM_Handle_Load_NVM . . . . .	368
4.159.3.12VTPM_Handle_Migrate_In . . . . .	368
4.159.3.13VTPM_Handle_Migrate_Out . . . . .	368
4.159.3.14VTPM_Handle_New_DMI . . . . .	368
4.159.3.15VTPM_Handle_Save_NVM . . . . .	368



4.159.3.16/TPM_Handle_TPM_Command . . . . .	368
4.159.3.17/TPM_Init_Manager . . . . .	368
4.159.3.18/TPM_LoadManagerData . . . . .	368
4.159.3.19/TPM_Manager_Handler . . . . .	368
4.159.3.20/TPM_New_DMI_Extra . . . . .	368
4.159.3.21/TPM_SaveManagerData . . . . .	368
4.159.3.22/TPM_Stop_Manager . . . . .	368
4.159.4 Variable Documentation . . . . .	368
4.159.4.1 HANDLER_QUIT_FLAG . . . . .	368
4.159.4.2 SRK_AUTH . . . . .	368
4.159.4.3 vtpm_globals . . . . .	368
4.160xen/tools/vtpm_manager/manager/vtsp.c File Reference . . . . .	368
4.160.1 Macro Definition Documentation . . . . .	369
4.160.1.1 RSA_KEY_SIZE . . . . .	369
4.160.2 Function Documentation . . . . .	369
4.160.2.1 GenerateAuth . . . . .	369
4.160.2.2 VerifyAuth . . . . .	369
4.160.2.3 VTSP_Bind . . . . .	369
4.160.2.4 VTSP_CreateWrapKey . . . . .	370
4.160.2.5 VTSP_DisablePubekRead . . . . .	370
4.160.2.6 VTSP_LoadKey . . . . .	370
4.160.2.7 VTSP_OIAP . . . . .	370
4.160.2.8 VTSP_OSAP . . . . .	370
4.160.2.9 VTSP_RawTransmit . . . . .	370
4.160.2.10VTSP_ReadPubek . . . . .	370
4.160.2.11VTSP_SaveState . . . . .	370
4.160.2.12VTSP_Seal . . . . .	370
4.160.2.13VTSP_TakeOwnership . . . . .	370
4.160.2.14VTSP_TerminateHandle . . . . .	370
4.160.2.15VTSP_Unbind . . . . .	370
4.160.2.16VTSP_Unseal . . . . .	370
4.161xen/tools/vtpm_manager/migration/vtpm_manager_if.c File Reference . . . . .	371
4.161.1 Function Documentation . . . . .	371
4.161.1.1 vtpm_manager_close . . . . .	371
4.161.1.2 vtpm_manager_command . . . . .	371
4.161.1.3 vtpm_manager_open . . . . .	371
4.162xen/tools/vtpm_manager/tcs/contextmgr.c File Reference . . . . .	371
4.162.1 Function Documentation . . . . .	372
4.162.1.1 AddHandleToList . . . . .	372
4.162.1.2 AddMemBlock . . . . .	372

4.162.1.3 DeleteHandleFromList . . . . .	372
4.162.1.4 DeleteMemBlock . . . . .	372
4.162.1.5 FreeHandleList . . . . .	372
4.163xen/tools/vtpm_manager/tcs/tcs.c File Reference . . . . .	372
4.163.1 Macro Definition Documentation . . . . .	373
4.163.1.1 TCPA_MAX_BUFFER_LENGTH . . . . .	373
4.163.2 Function Documentation . . . . .	373
4.163.2.1 LookupContext . . . . .	373
4.163.2.2 packAuth . . . . .	373
4.163.2.3 TCS_CloseContext . . . . .	373
4.163.2.4 TCS_create . . . . .	373
4.163.2.5 TCS_destroy . . . . .	373
4.163.2.6 TCS_FreeMemory . . . . .	373
4.163.2.7 TCS_Malloc . . . . .	373
4.163.2.8 TCS_OpenContext . . . . .	373
4.163.2.9 TCSP_CreateWrapKey . . . . .	373
4.163.2.10TCSP_DisablePubekRead . . . . .	373
4.163.2.11TCSP_EvictKey . . . . .	374
4.163.2.12TCSP_Extend . . . . .	374
4.163.2.13TCSP_GetRandom . . . . .	374
4.163.2.14TCSP_LoadKeyByBlob . . . . .	374
4.163.2.15TCSP_OIAP . . . . .	374
4.163.2.16TCSP_OSAP . . . . .	374
4.163.2.17TCSP_RawTransmitData . . . . .	374
4.163.2.18TCSP_ReadPubek . . . . .	374
4.163.2.19TCSP_SaveState . . . . .	374
4.163.2.20TCSP_Seal . . . . .	374
4.163.2.21TCSP_TakeOwnership . . . . .	374
4.163.2.22TCSP_TerminateHandle . . . . .	374
4.163.2.23TCSP_UnBind . . . . .	374
4.163.2.24TCSP_Unseal . . . . .	374
4.163.2.25unpackAuth . . . . .	374
4.163.3 Variable Documentation . . . . .	374
4.163.3.1 context_ht . . . . .	374
4.164xen/tools/vtpm_manager/tcs/tpmddl.c File Reference . . . . .	375
4.164.1 Macro Definition Documentation . . . . .	375
4.164.1.1 MIN . . . . .	375
4.164.2 Function Documentation . . . . .	375
4.164.2.1 TDDL_FlushSpecific . . . . .	375
4.164.2.2 TDDL_GetCapability . . . . .	375

4.165xen/tools/vtpm_manager/tcs/tpmddl.h File Reference . . . . .	375
4.165.1 Macro Definition Documentation . . . . .	376
4.165.1.1 TDDL_CAP_PROP_MANUFACTURER . . . . .	376
4.165.1.2 TDDL_E_FAIL . . . . .	376
4.165.1.3 TDDL_E_SUCCESS . . . . .	376
4.165.1.4 TDDL_SUCCESS . . . . .	376
4.165.2 Typedef Documentation . . . . .	376
4.165.2.1 TDDL_BYTE . . . . .	376
4.165.2.2 TDDL_RESULT . . . . .	376
4.165.2.3 TDDL_UINT32 . . . . .	376
4.165.3 Function Documentation . . . . .	376
4.165.3.1 TDDL_Close . . . . .	376
4.165.3.2 TDDL_FlushSpecific . . . . .	376
4.165.3.3 TDDL_GetCapability . . . . .	376
4.165.3.4 TDDL_GetStatus . . . . .	376
4.165.3.5 TDDL_Open . . . . .	376
4.165.3.6 TDDL_SetCapability . . . . .	376
4.165.3.7 TDDL_TransmitData . . . . .	376
4.166xen/tools/vtpm_manager/tcs/transmit.c File Reference . . . . .	376
4.166.1 Macro Definition Documentation . . . . .	377
4.166.1.1 TPM_TX_FNAME . . . . .	377
4.166.2 Function Documentation . . . . .	377
4.166.2.1 TDDL_Close . . . . .	377
4.166.2.2 TDDL_Open . . . . .	377
4.166.2.3 TDDL_TransmitData . . . . .	377
4.167xen/tools/vtpm_manager/util/buffer.c File Reference . . . . .	377
4.167.1 Function Documentation . . . . .	378
4.167.1.1 buffer_append_raw . . . . .	378
4.167.1.2 buffer_copy . . . . .	378
4.167.1.3 buffer_eq . . . . .	378
4.167.1.4 buffer_free . . . . .	378
4.167.1.5 buffer_init . . . . .	378
4.167.1.6 buffer_init_alias . . . . .	378
4.167.1.7 buffer_init_alias_convert . . . . .	378
4.167.1.8 buffer_init_const . . . . .	378
4.167.1.9 buffer_init_convert . . . . .	378
4.167.1.10buffer_init_copy . . . . .	378
4.167.1.11buffer_len . . . . .	378
4.167.1.12buffer_memset . . . . .	378
4.167.1.13buffer_prepend_raw . . . . .	378

4.168	xen/tools/vtpm_manager/util/buffer.h File Reference	378
4.168.1	Macro Definition Documentation	379
4.168.1.1	NULL_BUF	379
4.168.2	Typedef Documentation	379
4.168.2.1	tpm_size_t	379
4.168.3	Function Documentation	379
4.168.3.1	buffer_append_raw	379
4.168.3.2	buffer_copy	379
4.168.3.3	buffer_eq	379
4.168.3.4	buffer_free	379
4.168.3.5	buffer_init	379
4.168.3.6	buffer_init_alias	379
4.168.3.7	buffer_init_alias_convert	379
4.168.3.8	buffer_init_const	379
4.168.3.9	buffer_init_convert	379
4.168.3.10	buffer_init_copy	379
4.168.3.11	buffer_len	379
4.168.3.12	buffer_memset	379
4.168.3.13	buffer_prepend_raw	379
4.169	xen/tools/vtpm_manager/util/tcg.h File Reference	379
4.169.1	Macro Definition Documentation	387
4.169.1.1	ERRORDIE	387
4.169.1.2	FALSE	387
4.169.1.3	NULL_PACK_BUF	387
4.169.1.4	TCPA_ALG_AES	387
4.169.1.5	TCPA_MAX_BUFFER_LENGTH	387
4.169.1.6	TPM_ALG_3DES	387
4.169.1.7	TPM_ALG_DES	387
4.169.1.8	TPM_ALG_HMAC	387
4.169.1.9	TPM_ALG_RSA	387
4.169.1.10	TPM_ALG_SHA	387
4.169.1.11	TPM_AREA_LOCKED	387
4.169.1.12	TPM_AUDITFAIL_SUCCESSFUL	387
4.169.1.13	TPM_AUDITFAIL_UNSUCCESSFUL	387
4.169.1.14	TPM_AUDITFAILURE	387
4.169.1.15	TPM_AUTH2FAIL	387
4.169.1.16	TPM_AUTH_ALWAYS	387
4.169.1.17	TPM_AUTH_CONFLICT	387
4.169.1.18	TPM_AUTH_NEVER	387
4.169.1.19	TPM_AUTHFAIL	387

4.169.1.20TPM_BAD_ATTRIBUTES . . . . .	387
4.169.1.21TPM_BAD_COUNTER . . . . .	388
4.169.1.22TPM_BAD_DATASIZE . . . . .	388
4.169.1.23TPM_BAD_KEY_PROPERTY . . . . .	388
4.169.1.24TPM_BAD_LOCALITY . . . . .	388
4.169.1.25TPM_BAD_MIGRATION . . . . .	388
4.169.1.26TPM_BAD_MODE . . . . .	388
4.169.1.27TPM_BAD_ORDINAL . . . . .	388
4.169.1.28TPM_BAD_PARAM_SIZE . . . . .	388
4.169.1.29TPM_BAD_PARAMETER . . . . .	388
4.169.1.30TPM_BAD_PRESENCE . . . . .	388
4.169.1.31TPM_BAD_SCHEME . . . . .	388
4.169.1.32TPM_BAD_TYPE . . . . .	388
4.169.1.33TPM_BAD_VERSION . . . . .	388
4.169.1.34TPM_BADINDEX . . . . .	388
4.169.1.35TPM_BADTAG . . . . .	388
4.169.1.36TPM_BASE . . . . .	388
4.169.1.37TPM_CAP_ALG . . . . .	388
4.169.1.38TPM_CAP_AUTH_ENCRYPT . . . . .	388
4.169.1.39TPM_CAP_CHECK_LOADED . . . . .	388
4.169.1.40TPM_CAP_DA_LOGIC . . . . .	388
4.169.1.41TPM_CAP_FLAG . . . . .	388
4.169.1.42TPM_CAP_FLAG_PERMANENT . . . . .	388
4.169.1.43TPM_CAP_FLAG_VOLATILE . . . . .	388
4.169.1.44TPM_CAP_HANDLE . . . . .	388
4.169.1.45TPM_CAP_KEY_HANDLE . . . . .	388
4.169.1.46TPM_CAP_KEY_STATUS . . . . .	388
4.169.1.47TPM_CAP_MFR . . . . .	388
4.169.1.48TPM_CAP_NV_INDEX . . . . .	388
4.169.1.49TPM_CAP_NV_LIST . . . . .	389
4.169.1.50TPM_CAP_ORD . . . . .	389
4.169.1.51TPM_CAP_PID . . . . .	389
4.169.1.52TPM_CAP_PROP_ACTIVE_COUNTER . . . . .	389
4.169.1.53TPM_CAP_PROP_AUTHSESS . . . . .	389
4.169.1.54TPM_CAP_PROP_CMK_RESTRICTION . . . . .	389
4.169.1.55TPM_CAP_PROP_CONTEXT . . . . .	389
4.169.1.56TPM_CAP_PROP_CONTEXT_DIST . . . . .	389
4.169.1.57TPM_CAP_PROP_COUNTERS . . . . .	389
4.169.1.58TPM_CAP_PROP_DAA_INTERRUPT . . . . .	389
4.169.1.59TPM_CAP_PROP_DAASESS . . . . .	389

4.169.1.60TPM_CAP_PROP_DELEGATE_ROW . . . . .	389
4.169.1.61TPM_CAP_PROP_DIR . . . . .	389
4.169.1.62TPM_CAP_PROP_DURATION . . . . .	389
4.169.1.63TPM_CAP_PROP_FAMILYROWS . . . . .	389
4.169.1.64TPM_CAP_PROP_INPUT_BUFFER . . . . .	389
4.169.1.65TPM_CAP_PROP_KEYS . . . . .	389
4.169.1.66TPM_CAP_PROP_MANUFACTURER . . . . .	389
4.169.1.67TPM_CAP_PROP_MAX_AUTHSESS . . . . .	389
4.169.1.68TPM_CAP_PROP_MAX_CONTEXT . . . . .	389
4.169.1.69TPM_CAP_PROP_MAX_COUNTERS . . . . .	389
4.169.1.70TPM_CAP_PROP_MAX_DAASESS . . . . .	389
4.169.1.71TPM_CAP_PROP_MAX_KEYS . . . . .	389
4.169.1.72TPM_CAP_PROP_MAX_NV_AVAILABLE . . . . .	389
4.169.1.73TPM_CAP_PROP_MAX_SESSIONS . . . . .	389
4.169.1.74TPM_CAP_PROP_MAX_TRANSESS . . . . .	389
4.169.1.75TPM_CAP_PROP_MIN_COUNTER . . . . .	389
4.169.1.76TPM_CAP_PROP_OWNER . . . . .	389
4.169.1.77TPM_CAP_PROP_PCR . . . . .	390
4.169.1.78TPM_CAP_PROP_SESSIONS . . . . .	390
4.169.1.79TPM_CAP_PROP_STARTUP_EFFECT . . . . .	390
4.169.1.80TPM_CAP_PROP_TIS_TIMEOUT . . . . .	390
4.169.1.81TPM_CAP_PROP_TRANSESS . . . . .	390
4.169.1.82TPM_CAP_PROPERTY . . . . .	390
4.169.1.83TPM_CAP_SELECT_SIZE . . . . .	390
4.169.1.84TPM_CAP_SYM_MODE . . . . .	390
4.169.1.85TPM_CAP_TRANS_ALG . . . . .	390
4.169.1.86TPM_CAP_TRANS_ES . . . . .	390
4.169.1.87TPM_CAP_VERSION . . . . .	390
4.169.1.88TPM_CAP_VERSION_VAL . . . . .	390
4.169.1.89TPM_CLEAR_DISABLED . . . . .	390
4.169.1.90TPM_CONNECTION_ORDINAL . . . . .	390
4.169.1.91TPM_CONTEXT_GAP . . . . .	390
4.169.1.92TPM_DEACTIVATED . . . . .	390
4.169.1.93TPM_DECRYPT_ERROR . . . . .	390
4.169.1.94TPM_DELEGATE_ADMIN . . . . .	390
4.169.1.95TPM_DELEGATE_FAMILY . . . . .	390
4.169.1.96TPM_DELEGATE_LOCK . . . . .	390
4.169.1.97TPM_DIGEST_SIZE . . . . .	390
4.169.1.98TPM_DISABLED . . . . .	390
4.169.1.99TPM_DISABLED_CMD . . . . .	390

4.169.1.100TPM_ENCRYPT_ERROR . . . . .	390
4.169.1.101TPM_ES_NONE . . . . .	390
4.169.1.102TPM_ES_RSAESOAEP_SHA1_MGF1 . . . . .	390
4.169.1.103TPM_ES_RSAESPKCSv15 . . . . .	390
4.169.1.104TPM_ET_DATA . . . . .	390
4.169.1.105TPM_ET_KEY . . . . .	391
4.169.1.106TPM_ET_KEYHANDLE . . . . .	391
4.169.1.107TPM_ET_OWNER . . . . .	391
4.169.1.108TPM_ET_SRK . . . . .	391
4.169.1.109TPM_FAIL . . . . .	391
4.169.1.110TPM_FAILEDSELFTEST . . . . .	391
4.169.1.111TPM_FAMILYCOUNT . . . . .	391
4.169.1.112TPM_INAPPROPRIATE_ENC . . . . .	391
4.169.1.113TPM_INAPPROPRIATE_SIG . . . . .	391
4.169.1.114TPM_INSTALL_DISABLED . . . . .	391
4.169.1.115TPM_INVALID_AUTHHANDLE . . . . .	391
4.169.1.116TPM_INVALID_FAMILY . . . . .	391
4.169.1.117TPM_INVALID_KEYHANDLE . . . . .	391
4.169.1.118TPM_INVALID_KEYUSAGE . . . . .	391
4.169.1.119TPM_INVALID_PCR_INFO . . . . .	391
4.169.1.120TPM_INVALID_POSTINIT . . . . .	391
4.169.1.121TPM_INVALID_RESOURCE . . . . .	391
4.169.1.122TPM_INVALID_STRUCTURE . . . . .	391
4.169.1.123TPM_IOERROR . . . . .	391
4.169.1.124TPM_KEY_AUTHCHANGE . . . . .	391
4.169.1.125TPM_KEY_BIND . . . . .	391
4.169.1.126TPM_KEY_EK . . . . .	391
4.169.1.127TPM_KEY_IDENTITY . . . . .	391
4.169.1.128TPM_KEY_LEGACY . . . . .	391
4.169.1.129TPM_KEY_NOTSUPPORTED . . . . .	391
4.169.1.130TPM_KEY_OWNER_CONTROL . . . . .	391
4.169.1.131TPM_KEY_SIGNING . . . . .	391
4.169.1.132TPM_KEY_STORAGE . . . . .	391
4.169.1.133TPM_KEYNOTFOUND . . . . .	392
4.169.1.134TPM_MAXNVWRITES . . . . .	392
4.169.1.135TPM_MIGRATEFAIL . . . . .	392
4.169.1.136TPM_NO_ENDORSEMENT . . . . .	392
4.169.1.137TPM_NO_NV_PERMISSION . . . . .	392
4.169.1.138TPM_NO_WRAP_TRANSPORT . . . . .	392
4.169.1.139TPM_NON_FATAL . . . . .	392

4.169.1.140PM_NOOPERATOR . . . . .	392
4.169.1.141PM_NOSPACE . . . . .	392
4.169.1.142PM_NOSRK . . . . .	392
4.169.1.143PM_NOT_FULLWRITE . . . . .	392
4.169.1.144PM_NOTFIPS . . . . .	392
4.169.1.145PM_NOTLOCAL . . . . .	392
4.169.1.146PM_NOTRESETABLE . . . . .	392
4.169.1.147PM_NOTSEALED_BLOB . . . . .	392
4.169.1.148PM_ORD_ActivateIdentity . . . . .	392
4.169.1.149PM_ORD_AuthorizeMigrationKey . . . . .	392
4.169.1.150PM_ORD_CertifyKey . . . . .	392
4.169.1.151PM_ORD_CertifySelfTest . . . . .	392
4.169.1.152PM_ORD_ChangeAuth . . . . .	392
4.169.1.153PM_ORD_ChangeAuthAsymFinish . . . . .	392
4.169.1.154PM_ORD_ChangeAuthAsymStart . . . . .	392
4.169.1.155PM_ORD_ChangeAuthOwner . . . . .	392
4.169.1.156PM_ORD_ContinueSelfTest . . . . .	392
4.169.1.157PM_ORD_ConvertMigrationBlob . . . . .	392
4.169.1.158PM_ORD_CreateCounter . . . . .	392
4.169.1.159PM_ORD_CreateEndorsementKeyPair . . . . .	392
4.169.1.160PM_ORD_CreateMaintenanceArchive . . . . .	392
4.169.1.161PM_ORD_CreateMigrationBlob . . . . .	393
4.169.1.162PM_ORD_CreateWrapKey . . . . .	393
4.169.1.163PM_ORD_Delegate_CreateKeyDelegation . . . . .	393
4.169.1.164PM_ORD_Delegate_CreateOwnerDelegation . . . . .	393
4.169.1.165PM_ORD_Delegate_LoadOwnerDelegation . . . . .	393
4.169.1.166PM_ORD_Delegate_Manage . . . . .	393
4.169.1.167PM_ORD_Delegate_ReadAuth . . . . .	393
4.169.1.168PM_ORD_Delegate_ReadTable . . . . .	393
4.169.1.169PM_ORD_Delegate_UpdateVerification . . . . .	393
4.169.1.170PM_ORD_Delegate_VerifyDelegation . . . . .	393
4.169.1.171PM_ORD_DirRead . . . . .	393
4.169.1.172PM_ORD_DirWriteAuth . . . . .	393
4.169.1.173PM_ORD_DisableForceClear . . . . .	393
4.169.1.174PM_ORD_DisableOwnerClear . . . . .	393
4.169.1.175PM_ORD_DisablePubekRead . . . . .	393
4.169.1.176PM_ORD_EstablishTransport . . . . .	393
4.169.1.177PM_ORD_EvictKey . . . . .	393
4.169.1.178PM_ORD_ExecuteTransport . . . . .	393
4.169.1.179PM_ORD_Extend . . . . .	393



4.169.1.180PM_ORD_FieldUpgrade . . . . .	393
4.169.1.181PM_ORD_FlushSpecific . . . . .	393
4.169.1.182PM_ORD_ForceClear . . . . .	393
4.169.1.183PM_ORD_GetAuditEvent . . . . .	393
4.169.1.184PM_ORD_GetAuditEventSigned . . . . .	393
4.169.1.185PM_ORD_GetCapability . . . . .	393
4.169.1.186PM_ORD_GetCapabilityOwner . . . . .	393
4.169.1.187PM_ORD_GetCapabilitySigned . . . . .	393
4.169.1.188PM_ORD_GetOrdinalAuditStatus . . . . .	393
4.169.1.189PM_ORD_GetPubKey . . . . .	394
4.169.1.190PM_ORD_GetRandom . . . . .	394
4.169.1.191PM_ORD_GetTestResult . . . . .	394
4.169.1.192PM_ORD_GetTicks . . . . .	394
4.169.1.193PM_ORD_IncrementCounter . . . . .	394
4.169.1.194PM_ORD_Init . . . . .	394
4.169.1.195PM_ORD_KillMaintenanceFeature . . . . .	394
4.169.1.196PM_ORD_LoadAuthContext . . . . .	394
4.169.1.197PM_ORD_LoadContext . . . . .	394
4.169.1.198PM_ORD_LoadKey . . . . .	394
4.169.1.199PM_ORD_LoadKeyContext . . . . .	394
4.169.1.200PM_ORD_LoadMaintenanceArchive . . . . .	394
4.169.1.201PM_ORD_LoadManuMaintPub . . . . .	394
4.169.1.202PM_ORD_MakeIdentity . . . . .	394
4.169.1.203PM_ORD_MAX . . . . .	394
4.169.1.204PM_ORD_NV_DefineSpace . . . . .	394
4.169.1.205PM_ORD_NV_ReadValue . . . . .	394
4.169.1.206PM_ORD_NV_ReadValueAuth . . . . .	394
4.169.1.207PM_ORD_NV_WriteValue . . . . .	394
4.169.1.208PM_ORD_NV_WriteValueAuth . . . . .	394
4.169.1.209PM_ORD_OIAP . . . . .	394
4.169.1.210PM_ORD_OSAP . . . . .	394
4.169.1.211PM_ORD_OwnerClear . . . . .	394
4.169.1.212PM_ORD_OwnerReadPubek . . . . .	394
4.169.1.213PM_ORD_OwnerSetDisable . . . . .	394
4.169.1.214PM_ORD_PCR_Reset . . . . .	394
4.169.1.215PM_ORD_PcrRead . . . . .	394
4.169.1.216PM_ORD_PhysicalDisable . . . . .	394
4.169.1.217PM_ORD_PhysicalEnable . . . . .	395
4.169.1.218PM_ORD_PhysicalSetDeactivated . . . . .	395
4.169.1.219PM_ORD_Quote . . . . .	395

4.169.1.220PM_ORD_ReadCounter . . . . .	395
4.169.1.221PM_ORD_ReadManuMaintPub . . . . .	395
4.169.1.222PM_ORD_ReadPubek . . . . .	395
4.169.1.223PM_ORD_ReleaseCounter . . . . .	395
4.169.1.224PM_ORD_ReleaseCounterOwner . . . . .	395
4.169.1.225PM_ORD_ReleaseTransportSigned . . . . .	395
4.169.1.226PM_ORD_Reset . . . . .	395
4.169.1.227PM_ORD_ReWrapKey . . . . .	395
4.169.1.228PM_ORD_SaveAuthContext . . . . .	395
4.169.1.229PM_ORD_SaveContext . . . . .	395
4.169.1.230PM_ORD_SaveKeyContext . . . . .	395
4.169.1.231PM_ORD_SaveState . . . . .	395
4.169.1.232PM_ORD_Seal . . . . .	395
4.169.1.233PM_ORD_SelfTestFull . . . . .	395
4.169.1.234PM_ORD_SelfTestStartup . . . . .	395
4.169.1.235PM_ORD_SetOrdinalAuditStatus . . . . .	395
4.169.1.236PM_ORD_SetOwnerInstall . . . . .	395
4.169.1.237PM_ORD_SetRedirection . . . . .	395
4.169.1.238PM_ORD_SetTempDeactivated . . . . .	395
4.169.1.239PM_ORD_SHA1Complete . . . . .	395
4.169.1.240PM_ORD_SHA1CompleteExtend . . . . .	395
4.169.1.241PM_ORD_SHA1Start . . . . .	395
4.169.1.242PM_ORD_SHA1Update . . . . .	395
4.169.1.243PM_ORD_Sign . . . . .	395
4.169.1.244PM_ORD_Startup . . . . .	395
4.169.1.245PM_ORD_StirRandom . . . . .	396
4.169.1.246PM_ORD_TakeOwnership . . . . .	396
4.169.1.247PM_ORD_Terminate_Handle . . . . .	396
4.169.1.248PM_ORD_TickStampBlob . . . . .	396
4.169.1.249PM_ORD_UnBind . . . . .	396
4.169.1.250PM_ORD_Unseal . . . . .	396
4.169.1.251PM_OWNER_KEYHANDLE . . . . .	396
4.169.1.252PM_OWNER_SET . . . . .	396
4.169.1.253PM_PER_NOWRITE . . . . .	396
4.169.1.254PM_PID_ADCP . . . . .	396
4.169.1.255PM_PID_ADIP . . . . .	396
4.169.1.256PM_PID_OIAP . . . . .	396
4.169.1.257PM_PID_OSAP . . . . .	396
4.169.1.258PM_PID_OWNER . . . . .	396
4.169.1.259PM_PROTECTED_ORDINAL . . . . .	396

4.169.1.260TPM_PT_ASYM . . . . .	396
4.169.1.261TPM_PT_BIND . . . . .	396
4.169.1.262TPM_PT_MAINT . . . . .	396
4.169.1.263TPM_PT_MIGRATE . . . . .	396
4.169.1.264TPM_PT_SEAL . . . . .	396
4.169.1.265TPM_READ_ONLY . . . . .	396
4.169.1.266TPM_REQUIRES_SIGN . . . . .	396
4.169.1.267TPM_RESOURCEMISSING . . . . .	396
4.169.1.268TPM_RESOURCES . . . . .	396
4.169.1.269TPM_RT_AUTH . . . . .	396
4.169.1.270TPM_RT_CONTEXT . . . . .	396
4.169.1.271TPM_RT_COUNTER . . . . .	396
4.169.1.272TPM_RT_DAA_TPM . . . . .	396
4.169.1.273TPM_RT_DAA_V0 . . . . .	397
4.169.1.274TPM_RT_DAA_V1 . . . . .	397
4.169.1.275TPM_RT_DELEGATE . . . . .	397
4.169.1.276TPM_RT_HASH . . . . .	397
4.169.1.277TPM_RT_KEY . . . . .	397
4.169.1.278TPM_RT_TRANS . . . . .	397
4.169.1.279TPM_SHA_ERROR . . . . .	397
4.169.1.280TPM_SHA_THREAD . . . . .	397
4.169.1.281TPM_SHORTRANDOM . . . . .	397
4.169.1.282TPM_SIZE . . . . .	397
4.169.1.283TPM_SRK_KEYHANDLE . . . . .	397
4.169.1.284TPM_SS_NONE . . . . .	397
4.169.1.285TPM_SS_RSASSAPKCS1v15_DER . . . . .	397
4.169.1.286TPM_SS_RSASSAPKCS1v15_SHA1 . . . . .	397
4.169.1.287TPM_ST_CLEAR . . . . .	397
4.169.1.288TPM_ST_DEACTIVATED . . . . .	397
4.169.1.289TPM_ST_STATE . . . . .	397
4.169.1.290TPM_SUCCESS . . . . .	397
4.169.1.291TPM_TAG_RQU_AUTH1_COMMAND . . . . .	397
4.169.1.292TPM_TAG_RQU_AUTH2_COMMAND . . . . .	397
4.169.1.293TPM_TAG_RQU_COMMAND . . . . .	397
4.169.1.294TPM_TAG_RSP_AUTH1_COMMAND . . . . .	397
4.169.1.295TPM_TAG_RSP_AUTH2_COMMAND . . . . .	397
4.169.1.296TPM_TAG_RSP_COMMAND . . . . .	397
4.169.1.297TPM_TRANSPORT_EXCLUSIVE . . . . .	397
4.169.1.298TPM_UNPROTECTED_ORDINAL . . . . .	397
4.169.1.299TPM_VENDOR_ERROR . . . . .	397

4.169.1.300TPM_VENDOR_ORDINAL . . . . .	398
4.169.1.301TPM_WRITE_LOCKED . . . . .	398
4.169.1.302TPM_WRONG_ENTITYTYPE . . . . .	398
4.169.1.303TPM_WRONGPCRVAL . . . . .	398
4.169.1.304TPMTRY . . . . .	398
4.169.1.305TPMTRYRETURN . . . . .	398
4.169.1.306TRUE . . . . .	398
4.169.1.307SC_ORD_PhysicalPresence . . . . .	398
4.169.2 Typedef Documentation . . . . .	398
4.169.2.1 BOOL . . . . .	398
4.169.2.2 BYTE . . . . .	398
4.169.2.3 pack_buf_t . . . . .	398
4.169.2.4 pack_constbuf_t . . . . .	398
4.169.2.5 T CPA_ENCHANDLE . . . . .	398
4.169.2.6 T CPA_ENTITYHANDLE . . . . .	398
4.169.2.7 T CPA_HASHHANDLE . . . . .	398
4.169.2.8 T CPA_HMACHANDLE . . . . .	398
4.169.2.9 T CS_AUTH . . . . .	398
4.169.2.10TCS_AUTHHANDLE . . . . .	398
4.169.2.11TCS_CONTEXT_HANDLE . . . . .	399
4.169.2.12TCS_KEY_HANDLE . . . . .	399
4.169.2.13TPM_ALGORITHM_ID . . . . .	399
4.169.2.14TPM_AUTH_DATA_USAGE . . . . .	399
4.169.2.15TPM_AUTHDATA . . . . .	399
4.169.2.16TPM_AUTHHANDLE . . . . .	399
4.169.2.17TPM_BOUND_DATA . . . . .	399
4.169.2.18TPM_CAPABILITY_AREA . . . . .	399
4.169.2.19TPM_CHOSENID_HASH . . . . .	399
4.169.2.20TPM_COMMAND_CODE . . . . .	399
4.169.2.21TPM_COMPOSITE_HASH . . . . .	399
4.169.2.22TPM_DIGEST . . . . .	399
4.169.2.23TPM_DIRINDEX . . . . .	399
4.169.2.24TPM_DIRVALUE . . . . .	399
4.169.2.25TPM_ENC_SCHEME . . . . .	399
4.169.2.26TPM_ENCAUTH . . . . .	399
4.169.2.27TPM_ENTITY_TYPE . . . . .	399
4.169.2.28TPM_HANDLE . . . . .	399
4.169.2.29TPM_HMAC . . . . .	399
4.169.2.30TPM_KEY . . . . .	399
4.169.2.31TPM_KEY_FLAGS . . . . .	399

4.169.2.32TPM_KEY_HANDLE . . . . .	399
4.169.2.33TPM_KEY_PARMS . . . . .	399
4.169.2.34TPM_KEY_USAGE . . . . .	399
4.169.2.35TPM_MIGRATE_SCHEME . . . . .	399
4.169.2.36TPM_NONCE . . . . .	399
4.169.2.37TPM_PAYLOAD_TYPE . . . . .	399
4.169.2.38TPM_PCR_COMPOSITE . . . . .	399
4.169.2.39TPM_PCR_INFO . . . . .	400
4.169.2.40TPM_PCR_SELECTION . . . . .	400
4.169.2.41TPM_PCRINDEX . . . . .	400
4.169.2.42TPM_PCRVALUE . . . . .	400
4.169.2.43TPM_PHYSICAL_PRESENCE . . . . .	400
4.169.2.44TPM_PROTOCOL_ID . . . . .	400
4.169.2.45TPM_PUBKEY . . . . .	400
4.169.2.46TPM_RESOURCE_TYPE . . . . .	400
4.169.2.47TPM_RESULT . . . . .	400
4.169.2.48TPM_RSA_KEY_PARMS . . . . .	400
4.169.2.49TPM_SECRET . . . . .	400
4.169.2.50TPM_SIG_SCHEME . . . . .	400
4.169.2.51TPM_STARTUP_TYPE . . . . .	400
4.169.2.52TPM_STORE_PUBKEY . . . . .	400
4.169.2.53TPM_STORED_DATA . . . . .	400
4.169.2.54TPM_TAG . . . . .	400
4.169.2.55TPM_VERSION . . . . .	400
4.169.2.56UINT16 . . . . .	400
4.169.2.57UINT32 . . . . .	400
4.169.2.58UINT64 . . . . .	400
4.170xen/tools/vtpm_manager/vtpmmgrtalk/vtpmmgrtalk.c File Reference . . . . .	400
4.170.1 Function Documentation . . . . .	401
4.170.1.1 main . . . . .	401
4.171xen/tools/xenstore/xenstored_core.c File Reference . . . . .	401
4.171.1 Macro Definition Documentation . . . . .	402
4.171.1.1 log . . . . .	402
4.171.1.2 TDB_FLAGS . . . . .	402
4.171.2 Function Documentation . . . . .	402
4.171.2.1 canonicalize . . . . .	402
4.171.2.2 check_event_node . . . . .	402
4.171.2.3 dump_conn . . . . .	402
4.171.2.4 get_node . . . . .	402
4.171.2.5 get_strings . . . . .	402

4.171.2.6 is_child . . . . .	402
4.171.2.7 is_valid_nodename . . . . .	403
4.171.2.8 LIST_HEAD . . . . .	403
4.171.2.9 main . . . . .	403
4.171.2.10new_connection . . . . .	403
4.171.2.11replace_tdb . . . . .	403
4.171.2.12send_ack . . . . .	403
4.171.2.13send_error . . . . .	403
4.171.2.14send_reply . . . . .	403
4.171.2.15db_context . . . . .	403
4.171.2.16trace . . . . .	403
4.171.2.17trace_create . . . . .	403
4.171.2.18trace_destroy . . . . .	403
4.171.3 Variable Documentation . . . . .	403
4.171.3.1 quota_max_entry_size . . . . .	403
4.171.3.2 quota_max_transaction . . . . .	403
4.171.3.3 quota_nb_entry_per_domain . . . . .	403
4.171.3.4 quota_nb_watch_per_domain . . . . .	403
4.171.3.5 xce_handle . . . . .	403
4.172xen/tools/xenstore/xenstored_domain.c File Reference . . . . .	403
4.172.1 Macro Definition Documentation . . . . .	404
4.172.1.1 DOMID_DOMT . . . . .	404
4.172.1.2 DOMID_DOMT . . . . .	404
4.172.2 Function Documentation . . . . .	404
4.172.2.1 do_get_domain_path . . . . .	404
4.172.2.2 do_introduce . . . . .	404
4.172.2.3 do_is_domain_introduced . . . . .	405
4.172.2.4 do_release . . . . .	405
4.172.2.5 do_resume . . . . .	405
4.172.2.6 do_set_target . . . . .	405
4.172.2.7 domain_can_read . . . . .	405
4.172.2.8 domain_can_write . . . . .	405
4.172.2.9 domain_entry . . . . .	405
4.172.2.10domain_entry_dec . . . . .	405
4.172.2.11domain_entry_fix . . . . .	405
4.172.2.12domain_entry_inc . . . . .	405
4.172.2.13domain_init . . . . .	405
4.172.2.14domain_is_unprivileged . . . . .	405
4.172.2.15domain_watch . . . . .	405
4.172.2.16domain_watch_dec . . . . .	405

4.172.2.17domain_watch_inc . . . . .	405
4.172.2.18get_implicit_path . . . . .	405
4.172.2.19handle_event . . . . .	405
4.172.2.20restore_existing_connections . . . . .	405
4.172.3 Variable Documentation . . . . .	405
4.172.3.1 xce_handle . . . . .	405
4.173xen/tools/xenstore/xenstored_transaction.c File Reference . . . . .	406
4.173.1 Function Documentation . . . . .	406
4.173.1.1 add_change_node . . . . .	406
4.173.1.2 conn_delete_all_transactions . . . . .	407
4.173.1.3 do_transaction_end . . . . .	407
4.173.1.4 do_transaction_start . . . . .	407
4.173.1.5 tdb_transaction_context . . . . .	407
4.173.1.6 transaction_entry_dec . . . . .	407
4.173.1.7 transaction_entry_inc . . . . .	407
4.173.1.8 transaction_lookup . . . . .	407
4.173.2 Variable Documentation . . . . .	407
4.173.2.1 quota_max_transaction . . . . .	407
4.174xen/tools/xenvfsd/xenvfsd.c File Reference . . . . .	407
4.174.1 Macro Definition Documentation . . . . .	408
4.174.1.1 _GNU_SOURCE . . . . .	408
4.174.1.2 DOMID_T . . . . .	408
4.174.1.3 DOMID_Z . . . . .	408
4.174.1.4 SET_OTHEREND . . . . .	408
4.174.2 Function Documentation . . . . .	408
4.174.2.1 chk_map_gref_map . . . . .	408
4.174.2.2 main . . . . .	408
4.174.2.3 map_gref_map . . . . .	408
4.174.2.4 read_arg . . . . .	408
4.174.2.5 read_arg_int . . . . .	408
4.174.2.6 read_dyn_domid . . . . .	408
4.174.2.7 read_to_gntref . . . . .	409
4.174.2.8 stat_to_gntref . . . . .	409
4.174.2.9 write_from_gntref . . . . .	409
4.174.2.10xs_write_int . . . . .	409
4.174.3 Variable Documentation . . . . .	409
4.174.3.1 dyn_file . . . . .	409
4.174.3.2 dynamic . . . . .	409
4.174.3.3 path_arg1 . . . . .	409
4.174.3.4 path_arg2 . . . . .	409

4.174.3.5 path_arg3 . . . . .	409
4.174.3.6 path_arg4 . . . . .	409
4.174.3.7 path_req . . . . .	409
4.174.3.8 path_rsp . . . . .	409
4.175xen/xen/arch/x86/domain_build.c File Reference . . . . .	410
4.175.1 Macro Definition Documentation . . . . .	411
4.175.1.1 CAAS_DEVELOP . . . . .	411
4.175.1.2 pfn . . . . .	411
4.175.1.3 REVERSE_START . . . . .	411
4.175.1.4 round_pgdown . . . . .	411
4.175.1.5 round_pgup . . . . .	411
4.175.1.6 TPM_LOC0 . . . . .	411
4.175.1.7 TPM_LOCALL . . . . .	411
4.175.2 Function Documentation . . . . .	411
4.175.2.1 alloc_dom0_vcpu0 . . . . .	411
4.175.2.2 alloc_domt_vcpu0 . . . . .	411
4.175.2.3 boolean_param . . . . .	411
4.175.2.4 construct_dom . . . . .	411
4.175.2.5 custom_param . . . . .	411
4.175.2.6 integer_param . . . . .	411
4.175.2.7 integer_param . . . . .	411
4.175.2.8 integer_param . . . . .	411
4.175.2.9 string_param . . . . .	411
4.176xen/xen/arch/x86/setup.c File Reference . . . . .	412
4.176.1 Macro Definition Documentation . . . . .	413
4.176.1.1 BOOTSTRAP_MAP_BASE . . . . .	413
4.176.1.2 BOOTSTRAP_MAP_LIMIT . . . . .	413
4.176.1.3 EARLY_FAIL . . . . .	413
4.176.2 Function Documentation . . . . .	414
4.176.2.1 __attribute__ . . . . .	414
4.176.2.2 __start_xen . . . . .	414
4.176.2.3 arch_get_xen_caps . . . . .	414
4.176.2.4 boolean_param . . . . .	414
4.176.2.5 boolean_param . . . . .	414
4.176.2.6 boolean_param . . . . .	414
4.176.2.7 boolean_param . . . . .	414
4.176.2.8 boolean_param . . . . .	414
4.176.2.9 custom_param . . . . .	414
4.176.2.10DEFINE_PER_CPU . . . . .	414
4.176.2.11discard_initial_images . . . . .	414



4.176.2.12	nit_done	414
4.176.2.13	initial_images_nrpages	414
4.176.2.14	integer_param	414
4.176.2.15	nvbool_param	414
4.176.2.16	set_pdx_range	414
4.176.2.17	srat_detect_node	414
4.176.2.18	xen_in_range	414
4.176.3	Variable Documentation	414
4.176.3.1	__bss_start	414
4.176.3.2	__init_begin	414
4.176.3.3	__init_end	414
4.176.3.4	acpi_disabled	414
4.176.3.5	acpi_force	414
4.176.3.6	boot_edid_caps	414
4.176.3.7	boot_edid_info	414
4.176.3.8	boot_vid_info	414
4.176.3.9	cpu_present_map	415
4.176.3.10	early_boot	415
4.176.3.11	mmu_cr4_features	415
4.176.3.12	xen_cpuidle	415
4.176.3.13	xen_phys_start	415
4.177	xen/xen/common/domain.c File Reference	415
4.177.1	Macro Definition Documentation	417
4.177.1.1	DOMAIN_HASH	417
4.177.1.2	DOMAIN_HASH_SIZE	417
4.177.2	Function Documentation	417
4.177.2.1	__domain_crash	417
4.177.2.2	__domain_crash_synchronous	417
4.177.2.3	alloc_vcpu	417
4.177.2.4	boolean_param	417
4.177.2.5	boot_vcpu	417
4.177.2.6	continue_hypercall_on_cpu	417
4.177.2.7	current_domain_id	417
4.177.2.8	custom_param	417
4.177.2.9	custom_param	417
4.177.2.10	DEFINE_RCU_READ_LOCK	417
4.177.2.11	DEFINE_SPINLOCK	417
4.177.2.12	do_vcpu_op	417
4.177.2.13	domain_create	417
4.177.2.14	domain_destroy	417

4.177.2.15	domain_kill . . . . .	417
4.177.2.16	domain_pause . . . . .	417
4.177.2.17	domain_pause_by_systemcontroller . . . . .	417
4.177.2.18	domain_pause_for_debugger . . . . .	418
4.177.2.19	domain_resume . . . . .	418
4.177.2.20	domain_shutdown . . . . .	418
4.177.2.21	domain_unpause . . . . .	418
4.177.2.22	domain_unpause_by_systemcontroller . . . . .	418
4.177.2.23	domain_update_node_affinity . . . . .	418
4.177.2.24	get_domain_by_id . . . . .	418
4.177.2.25	rcu_lock_domain_by_id . . . . .	418
4.177.2.26	rcu_lock_remote_target_domain_by_id . . . . .	418
4.177.2.27	rcu_lock_target_domain_by_id . . . . .	418
4.177.2.28	cpu_end_shutdown_deferral . . . . .	418
4.177.2.29	cpu_pause . . . . .	418
4.177.2.30	cpu_pause_nosync . . . . .	418
4.177.2.31	cpu_reset . . . . .	418
4.177.2.32	cpu_start_shutdown_deferral . . . . .	418
4.177.2.33	cpu_unpause . . . . .	418
4.177.2.34	vm_assist . . . . .	418
4.177.3	Variable Documentation . . . . .	418
4.177.3.1	__read_mostly . . . . .	418
4.177.3.2	cpufreq_controller . . . . .	418
4.177.3.3	dom0 . . . . .	418
4.177.3.4	domain_list . . . . .	419
4.177.3.5	domt . . . . .	419
4.177.3.6	dummy_vcpu_info . . . . .	419
4.177.3.7	xen_processor_pmbits . . . . .	419
4.178	xen/xen/common/domctl.c File Reference . . . . .	419
4.178.1	Function Documentation . . . . .	419
4.178.1.1	arch_do_domctl . . . . .	419
4.178.1.2	cpumask_to_xenctl_cpumap . . . . .	419
4.178.1.3	do_domctl . . . . .	419
4.178.1.4	domctl_lock_acquire . . . . .	419
4.178.1.5	domctl_lock_release . . . . .	419
4.178.1.6	getdomaininfo . . . . .	419
4.178.1.7	xenctl_cpumap_to_cpumask . . . . .	420
4.179	xen/xen/common/memory.c File Reference . . . . .	420
4.179.1	Function Documentation . . . . .	420
4.179.1.1	do_memory_op . . . . .	420

4.179.1.2 <code>guest_remove_page</code> . . . . .	420
4.180 <code>xen/xen/include/asm-x86/setup.h</code> File Reference . . . . .	420
4.180.1 Enumeration Type Documentation . . . . .	421
4.180.1.1 <code>domtype</code> . . . . .	421
4.180.2 Function Documentation . . . . .	421
4.180.2.1 <code>amd_init_cpu</code> . . . . .	421
4.180.2.2 <code>arch_init_memory</code> . . . . .	421
4.180.2.3 <code>centaur_init_cpu</code> . . . . .	421
4.180.2.4 <code>construct_dom</code> . . . . .	421
4.180.2.5 <code>cyrix_init_cpu</code> . . . . .	421
4.180.2.6 <code>discard_initial_images</code> . . . . .	421
4.180.2.7 <code>early_cpu_init</code> . . . . .	421
4.180.2.8 <code>early_page_fault</code> . . . . .	422
4.180.2.9 <code>early_time_init</code> . . . . .	422
4.180.2.10 <code>init_done</code> . . . . .	422
4.180.2.11 <code>init_IRQ</code> . . . . .	422
4.180.2.12 <code>initial_images_nrpages</code> . . . . .	422
4.180.2.13 <code>intel_cpu_init</code> . . . . .	422
4.180.2.14 <code>nsc_init_cpu</code> . . . . .	422
4.180.2.15 <code>numa_initmem_init</code> . . . . .	422
4.180.2.16 <code>subarch_init_memory</code> . . . . .	422
4.180.2.17 <code>transmeta_init_cpu</code> . . . . .	422
4.180.2.18 <code>vesa_init</code> . . . . .	422
4.180.2.19 <code>vesa_mtrr_init</code> . . . . .	422
4.180.3 Variable Documentation . . . . .	422
4.180.3.1 <code>early_boot</code> . . . . .	422
4.180.3.2 <code>xenheap_initial_phys_start</code> . . . . .	422
4.181 <code>xen/xen/include/public/domctl.h</code> File Reference . . . . .	422
4.181.1 Macro Definition Documentation . . . . .	429
4.181.1.1 <code>_XEN_DOMAINSETUP_hvm_guest</code> . . . . .	429
4.181.1.2 <code>_XEN_DOMAINSETUP_query</code> . . . . .	429
4.181.1.3 <code>_XEN_DOMAINSETUP_sioemu_guest</code> . . . . .	429
4.181.1.4 <code>_XEN_DOMCTL_CDF_hap</code> . . . . .	429
4.181.1.5 <code>_XEN_DOMCTL_CDF_hvm_guest</code> . . . . .	429
4.181.1.6 <code>_XEN_DOMCTL_CDF_oos_off</code> . . . . .	429
4.181.1.7 <code>_XEN_DOMCTL_CDF_priv</code> . . . . .	429
4.181.1.8 <code>_XEN_DOMCTL_CDF_s3_integrity</code> . . . . .	429
4.181.1.9 <code>_XEN_DOMINF_blocked</code> . . . . .	429
4.181.1.10 <code>_XEN_DOMINF_debugged</code> . . . . .	429
4.181.1.11 <code>_XEN_DOMINF_dying</code> . . . . .	429

4.181.1.12	XEN_DOMINF_hvm_guest . . . . .	429
4.181.1.13	XEN_DOMINF_paused . . . . .	429
4.181.1.14	XEN_DOMINF_running . . . . .	429
4.181.1.15	XEN_DOMINF_shutdown . . . . .	429
4.181.1.16	XEN_DOMCTL_ADD_MAPPING . . . . .	429
4.181.1.17	XEN_DOMCTL_REMOVE_MAPPING . . . . .	429
4.181.1.18	XEN_DOMAINSETUP_hvm_guest . . . . .	429
4.181.1.19	XEN_DOMAINSETUP_query . . . . .	429
4.181.1.20	XEN_DOMAINSETUP_sioemu_guest . . . . .	429
4.181.1.21	XEN_DOMCTL_arch_setup . . . . .	429
4.181.1.22	XEN_DOMCTL_assign_device . . . . .	429
4.181.1.23	XEN_DOMCTL_bind_pt_irq . . . . .	429
4.181.1.24	XEN_DOMCTL_CDF_hap . . . . .	429
4.181.1.25	XEN_DOMCTL_CDF_hvm_guest . . . . .	429
4.181.1.26	XEN_DOMCTL_CDF_oos_off . . . . .	429
4.181.1.27	XEN_DOMCTL_CDF_priv . . . . .	429
4.181.1.28	XEN_DOMCTL_CDF_s3_integrity . . . . .	430
4.181.1.29	XEN_DOMCTL_createdomain . . . . .	430
4.181.1.30	XEN_DOMCTL_deassign_device . . . . .	430
4.181.1.31	XEN_DOMCTL_debug_op . . . . .	430
4.181.1.32	XEN_DOMCTL_DEBUG_OP_SINGLE_STEP_OFF . . . . .	430
4.181.1.33	XEN_DOMCTL_DEBUG_OP_SINGLE_STEP_ON . . . . .	430
4.181.1.34	XEN_DOMCTL_destroydomain . . . . .	430
4.181.1.35	XEN_DOMCTL_disable_migrate . . . . .	430
4.181.1.36	XEN_DOMCTL_gdbsx_domstatus . . . . .	430
4.181.1.37	XEN_DOMCTL_gdbsx_guestmemio . . . . .	430
4.181.1.38	XEN_DOMCTL_gdbsx_pausevcpu . . . . .	430
4.181.1.39	XEN_DOMCTL_gdbsx_unpausevcpu . . . . .	430
4.181.1.40	XEN_DOMCTL_get_address_size . . . . .	430
4.181.1.41	XEN_DOMCTL_get_device_group . . . . .	430
4.181.1.42	XEN_DOMCTL_get_ext_vcpucontext . . . . .	430
4.181.1.43	XEN_DOMCTL_get_machine_address_size . . . . .	430
4.181.1.44	XEN_DOMCTL_getdomaininfo . . . . .	430
4.181.1.45	XEN_DOMCTL_gethvmcontext . . . . .	430
4.181.1.46	XEN_DOMCTL_gethvmcontext_partial . . . . .	430
4.181.1.47	XEN_DOMCTL_getmemlist . . . . .	430
4.181.1.48	XEN_DOMCTL_getpageframeinfo . . . . .	430
4.181.1.49	XEN_DOMCTL_getpageframeinfo2 . . . . .	430
4.181.1.50	XEN_DOMCTL_getpageframeinfo3 . . . . .	430
4.181.1.51	XEN_DOMCTL_gettscinfo . . . . .	430

4.181.1.52	XEN_DOMCTL_getvcpuaffinity	430
4.181.1.53	XEN_DOMCTL_getvcpucontext	430
4.181.1.54	XEN_DOMCTL_getvcpuextstate	430
4.181.1.55	XEN_DOMCTL_getvcpuinfo	430
4.181.1.56	XEN_DOMCTL_hypercall_init	431
4.181.1.57	XEN_DOMCTL_INTERFACE_VERSION	431
4.181.1.58	XEN_DOMCTL_iomem_permission	431
4.181.1.59	XEN_DOMCTL_ioport_mapping	431
4.181.1.60	XEN_DOMCTL_ioport_permission	431
4.181.1.61	XEN_DOMCTL_irq_permission	431
4.181.1.62	XEN_DOMCTL_max_mem	431
4.181.1.63	XEN_DOMCTL_max_vcpus	431
4.181.1.64	XEN_DOMCTL_MEM_CACHEATTR_UC	431
4.181.1.65	XEN_DOMCTL_MEM_CACHEATTR_UCM	431
4.181.1.66	XEN_DOMCTL_MEM_CACHEATTR_WB	431
4.181.1.67	XEN_DOMCTL_MEM_CACHEATTR_WC	431
4.181.1.68	XEN_DOMCTL_MEM_CACHEATTR_WP	431
4.181.1.69	XEN_DOMCTL_MEM_CACHEATTR_WT	431
4.181.1.70	XEN_DOMCTL_mem_event_op	431
4.181.1.71	XEN_DOMCTL_MEM_EVENT_OP_ACCESS	431
4.181.1.72	XEN_DOMCTL_MEM_EVENT_OP_ACCESS_RESUME	431
4.181.1.73	XEN_DOMCTL_MEM_EVENT_OP_DISABLE	431
4.181.1.74	XEN_DOMCTL_MEM_EVENT_OP_ENABLE	431
4.181.1.75	XEN_DOMCTL_MEM_EVENT_OP_PAGING	431
4.181.1.76	XEN_DOMCTL_MEM_EVENT_OP_PAGING_EVICT	431
4.181.1.77	XEN_DOMCTL_MEM_EVENT_OP_PAGING_NOMINATE	431
4.181.1.78	XEN_DOMCTL_MEM_EVENT_OP_PAGING_PREP	431
4.181.1.79	XEN_DOMCTL_MEM_EVENT_OP_PAGING_RESUME	431
4.181.1.80	XEN_DOMCTL_MEM_SHARING_C_HANDLE_INVALID	431
4.181.1.81	XEN_DOMCTL_mem_sharing_op	431
4.181.1.82	XEN_DOMCTL_MEM_SHARING_OP_CONTROL	431
4.181.1.83	XEN_DOMCTL_MEM_SHARING_OP_DEBUG_GFN	431
4.181.1.84	XEN_DOMCTL_MEM_SHARING_OP_DEBUG_GREF	432
4.181.1.85	XEN_DOMCTL_MEM_SHARING_OP_DEBUG_MFN	432
4.181.1.86	XEN_DOMCTL_MEM_SHARING_OP_NOMINATE_GFN	432
4.181.1.87	XEN_DOMCTL_MEM_SHARING_OP_NOMINATE_GREF	432
4.181.1.88	XEN_DOMCTL_MEM_SHARING_OP_RESUME	432
4.181.1.89	XEN_DOMCTL_MEM_SHARING_OP_SHARE	432
4.181.1.90	XEN_DOMCTL_MEM_SHARING_S_HANDLE_INVALID	432
4.181.1.91	XEN_DOMCTL_memory_mapping	432

4.181.1.92	XEN_DOMCTL_pausedomain . . . . .	432
4.181.1.93	XEN_DOMCTL_PFINFO_L1TAB . . . . .	432
4.181.1.94	XEN_DOMCTL_PFINFO_L2TAB . . . . .	432
4.181.1.95	XEN_DOMCTL_PFINFO_L3TAB . . . . .	432
4.181.1.96	XEN_DOMCTL_PFINFO_L4TAB . . . . .	432
4.181.1.97	XEN_DOMCTL_PFINFO_LPINTAB . . . . .	432
4.181.1.98	XEN_DOMCTL_PFINFO_LTAB_MASK . . . . .	432
4.181.1.99	XEN_DOMCTL_PFINFO_LTAB_SHIFT . . . . .	432
4.181.1.100	XEN_DOMCTL_PFINFO_LTABTYPE_MASK . . . . .	432
4.181.1.101	XEN_DOMCTL_PFINFO_NOTAB . . . . .	432
4.181.1.102	XEN_DOMCTL_PFINFO_PAGEDTAB . . . . .	432
4.181.1.103	XEN_DOMCTL_PFINFO_XTAB . . . . .	432
4.181.1.104	XEN_DOMCTL_pin_mem_cacheattr . . . . .	432
4.181.1.105	XEN_DOMCTL_real_mode_area . . . . .	432
4.181.1.106	XEN_DOMCTL_resumedomain . . . . .	432
4.181.1.107	XEN_DOMCTL_SCHEDOP_getinfo . . . . .	432
4.181.1.108	XEN_DOMCTL_SCHEDOP_putinfo . . . . .	432
4.181.1.109	XEN_DOMCTL_scheduler_op . . . . .	432
4.181.1.110	XEN_DOMCTL_sendtrigger . . . . .	432
4.181.1.111	XEN_DOMCTL_SENDTRIGGER_INIT . . . . .	432
4.181.1.112	XEN_DOMCTL_SENDTRIGGER_NMI . . . . .	433
4.181.1.113	XEN_DOMCTL_SENDTRIGGER_POWER . . . . .	433
4.181.1.114	XEN_DOMCTL_SENDTRIGGER_RESET . . . . .	433
4.181.1.115	XEN_DOMCTL_SENDTRIGGER_SLEEP . . . . .	433
4.181.1.116	XEN_DOMCTL_set_access_required . . . . .	433
4.181.1.117	XEN_DOMCTL_set_address_size . . . . .	433
4.181.1.118	XEN_DOMCTL_set_cpuid . . . . .	433
4.181.1.119	XEN_DOMCTL_set_ext_vcpucontext . . . . .	433
4.181.1.120	XEN_DOMCTL_set_machine_address_size . . . . .	433
4.181.1.121	XEN_DOMCTL_set_opt_feature . . . . .	433
4.181.1.122	XEN_DOMCTL_set_target . . . . .	433
4.181.1.123	XEN_DOMCTL_setdebugging . . . . .	433
4.181.1.124	XEN_DOMCTL_setdomainhandle . . . . .	433
4.181.1.125	XEN_DOMCTL_sethvmcontext . . . . .	433
4.181.1.126	XEN_DOMCTL_settimeoffset . . . . .	433
4.181.1.127	XEN_DOMCTL_settscinfo . . . . .	433
4.181.1.128	XEN_DOMCTL_setvcpuaffinity . . . . .	433
4.181.1.129	XEN_DOMCTL_setvcpucontext . . . . .	433
4.181.1.130	XEN_DOMCTL_setvcpuextstate . . . . .	433
4.181.1.131	XEN_DOMCTL_SHADOW_ENABLE_EXTERNAL . . . . .	433

4.181.1.132	XEN_DOMCTL_SHADOW_ENABLE_LOG_DIRTY . . . . .	433
4.181.1.133	XEN_DOMCTL_SHADOW_ENABLE_REFCOUNT . . . . .	433
4.181.1.134	XEN_DOMCTL_SHADOW_ENABLE_TRANSLATE . . . . .	433
4.181.1.135	XEN_DOMCTL_shadow_op . . . . .	433
4.181.1.136	XEN_DOMCTL_SHADOW_OP_CLEAN . . . . .	433
4.181.1.137	XEN_DOMCTL_SHADOW_OP_ENABLE . . . . .	433
4.181.1.138	XEN_DOMCTL_SHADOW_OP_ENABLE_LOGDIRTY . . . . .	433
4.181.1.139	XEN_DOMCTL_SHADOW_OP_ENABLE_TEST . . . . .	433
4.181.1.140	XEN_DOMCTL_SHADOW_OP_ENABLE_TRANSLATE . . . . .	434
4.181.1.141	XEN_DOMCTL_SHADOW_OP_GET_ALLOCATION . . . . .	434
4.181.1.142	XEN_DOMCTL_SHADOW_OP_OFF . . . . .	434
4.181.1.143	XEN_DOMCTL_SHADOW_OP_PEEK . . . . .	434
4.181.1.144	XEN_DOMCTL_SHADOW_OP_SET_ALLOCATION . . . . .	434
4.181.1.145	XEN_DOMCTL_subscribe . . . . .	434
4.181.1.146	XEN_DOMCTL_suppress_spurious_page_faults . . . . .	434
4.181.1.147	XEN_DOMCTL_test_assign_device . . . . .	434
4.181.1.148	XEN_DOMCTL_unbind_pt_irq . . . . .	434
4.181.1.149	XEN_DOMCTL_unpausedomain . . . . .	434
4.181.1.150	XEN_DOMINF_blocked . . . . .	434
4.181.1.151	XEN_DOMINF_debugged . . . . .	434
4.181.1.152	XEN_DOMINF_dying . . . . .	434
4.181.1.153	XEN_DOMINF_hvm_guest . . . . .	434
4.181.1.154	XEN_DOMINF_paused . . . . .	434
4.181.1.155	XEN_DOMINF_running . . . . .	434
4.181.1.156	XEN_DOMINF_shutdown . . . . .	434
4.181.1.157	XEN_DOMINF_shutdownmask . . . . .	434
4.181.1.158	XEN_DOMINF_shutdownshift . . . . .	434
4.181.1.159	XEN_SCHEDULER_ARINC653 . . . . .	434
4.181.1.160	XEN_SCHEDULER_CREDIT . . . . .	434
4.181.1.161	XEN_SCHEDULER_CREDIT2 . . . . .	434
4.181.1.162	XEN_SCHEDULER_SEDF . . . . .	434
4.181.2	Typedef Documentation . . . . .	434
4.181.2.1	pt_irq_type_t . . . . .	434
4.181.2.2	xen_domctl_address_size_t . . . . .	434
4.181.2.3	xen_domctl_arch_setup_t . . . . .	434
4.181.2.4	xen_domctl_assign_device_t . . . . .	434
4.181.2.5	xen_domctl_bind_pt_irq_t . . . . .	435
4.181.2.6	xen_domctl_createdomain_t . . . . .	435
4.181.2.7	xen_domctl_debug_op_t . . . . .	435
4.181.2.8	xen_domctl_disable_migrate_t . . . . .	435

4.181.2.9	<a href="#">xen_domctl_ext_vcpucontext_t</a>	435
4.181.2.10	<a href="#">xen_domctl_get_device_group_t</a>	435
4.181.2.11	<a href="#">xen_domctl_getdomaininfo_t</a>	435
4.181.2.12	<a href="#">xen_domctl_getmemlist_t</a>	435
4.181.2.13	<a href="#">xen_domctl_getpageframeinfo2_t</a>	435
4.181.2.14	<a href="#">xen_domctl_getpageframeinfo_t</a>	435
4.181.2.15	<a href="#">xen_domctl_getvcpuinfo_t</a>	435
4.181.2.16	<a href="#">xen_domctl_hvmcontext_partial_t</a>	435
4.181.2.17	<a href="#">xen_domctl_hvmcontext_t</a>	435
4.181.2.18	<a href="#">xen_domctl_hypercall_init_t</a>	435
4.181.2.19	<a href="#">xen_domctl_iomem_permission_t</a>	435
4.181.2.20	<a href="#">xen_domctl_ioport_mapping_t</a>	435
4.181.2.21	<a href="#">xen_domctl_ioport_permission_t</a>	435
4.181.2.22	<a href="#">xen_domctl_irq_permission_t</a>	435
4.181.2.23	<a href="#">xen_domctl_max_mem_t</a>	435
4.181.2.24	<a href="#">xen_domctl_max_vcpus_t</a>	435
4.181.2.25	<a href="#">xen_domctl_mem_event_op_t</a>	435
4.181.2.26	<a href="#">xen_domctl_mem_sharing_op_t</a>	435
4.181.2.27	<a href="#">xen_domctl_memory_mapping_t</a>	435
4.181.2.28	<a href="#">xen_domctl_pin_mem_cacheattr_t</a>	435
4.181.2.29	<a href="#">xen_domctl_real_mode_area_t</a>	435
4.181.2.30	<a href="#">xen_domctl_scheduler_op_t</a>	435
4.181.2.31	<a href="#">xen_domctl_sendtrigger_t</a>	435
4.181.2.32	<a href="#">xen_domctl_set_access_required_t</a>	435
4.181.2.33	<a href="#">xen_domctl_set_opt_feature_t</a>	436
4.181.2.34	<a href="#">xen_domctl_set_target_t</a>	436
4.181.2.35	<a href="#">xen_domctl_setdebugging_t</a>	436
4.181.2.36	<a href="#">xen_domctl_setdomainhandle_t</a>	436
4.181.2.37	<a href="#">xen_domctl_settimeoffset_t</a>	436
4.181.2.38	<a href="#">xen_domctl_shadow_op_stats_t</a>	436
4.181.2.39	<a href="#">xen_domctl_shadow_op_t</a>	436
4.181.2.40	<a href="#">xen_domctl_subscribe_t</a>	436
4.181.2.41	<a href="#">xen_domctl_t</a>	436
4.181.2.42	<a href="#">xen_domctl_tsc_info_t</a>	436
4.181.2.43	<a href="#">xen_domctl_vcpuaffinity_t</a>	436
4.181.2.44	<a href="#">xen_domctl_vcpucontext_t</a>	436
4.181.2.45	<a href="#">xen_guest_tsc_info_t</a>	436
4.181.3	Enumeration Type Documentation	436
4.181.3.1	<a href="#">pt_irq_type_e</a>	436
4.181.4	Function Documentation	436



4.181.4.1	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.2	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.3	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.4	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.5	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.6	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.7	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.8	DEFINE_XEN_GUEST_HANDLE	436
4.181.4.9	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.10	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.11	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.12	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.13	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.14	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.15	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.16	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.17	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.18	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.19	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.20	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.21	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.22	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.23	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.24	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.25	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.26	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.27	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.28	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.29	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.30	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.31	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.32	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.33	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.34	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.35	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.36	DEFINE_XEN_GUEST_HANDLE	437
4.181.4.37	DEFINE_XEN_GUEST_HANDLE	438
4.181.4.38	DEFINE_XEN_GUEST_HANDLE	438
4.181.4.39	DEFINE_XEN_GUEST_HANDLE	438
4.181.4.40	DEFINE_XEN_GUEST_HANDLE	438

4.181.4.41	DEFINE_XEN_GUEST_HANDLE	438
4.181.4.42	DEFINE_XEN_GUEST_HANDLE	438
4.182	xen/xen/include/public/xen.h File Reference	438
4.182.1	Macro Definition Documentation	441
4.182.1.1	__HYPERVISOR_arch_0	441
4.182.1.2	__HYPERVISOR_arch_1	441
4.182.1.3	__HYPERVISOR_arch_2	441
4.182.1.4	__HYPERVISOR_arch_3	441
4.182.1.5	__HYPERVISOR_arch_4	441
4.182.1.6	__HYPERVISOR_arch_5	441
4.182.1.7	__HYPERVISOR_arch_6	441
4.182.1.8	__HYPERVISOR_arch_7	441
4.182.1.9	__HYPERVISOR_callback_op	441
4.182.1.10	__HYPERVISOR_console_io	441
4.182.1.11	__HYPERVISOR_dom0_op	442
4.182.1.12	__HYPERVISOR_domctl	442
4.182.1.13	__HYPERVISOR_event_channel_op	442
4.182.1.14	__HYPERVISOR_event_channel_op	442
4.182.1.15	__HYPERVISOR_event_channel_op_compat	442
4.182.1.16	__HYPERVISOR_fpu_taskswitch	442
4.182.1.17	__HYPERVISOR_get_debugreg	442
4.182.1.18	__HYPERVISOR_grant_table_op	442
4.182.1.19	__HYPERVISOR_hvm_op	442
4.182.1.20	__HYPERVISOR_iret	442
4.182.1.21	__HYPERVISOR_kexec_op	442
4.182.1.22	__HYPERVISOR_memory_op	442
4.182.1.23	__HYPERVISOR_mmu_update	442
4.182.1.24	__HYPERVISOR_mmuext_op	442
4.182.1.25	__HYPERVISOR_multicall	442
4.182.1.26	__HYPERVISOR_nmi_op	442
4.182.1.27	__HYPERVISOR_physdev_op	442
4.182.1.28	__HYPERVISOR_physdev_op	442
4.182.1.29	__HYPERVISOR_physdev_op_compat	442
4.182.1.30	__HYPERVISOR_platform_op	442
4.182.1.31	__HYPERVISOR_sched_op	442
4.182.1.32	__HYPERVISOR_sched_op	442
4.182.1.33	__HYPERVISOR_sched_op_compat	442
4.182.1.34	__HYPERVISOR_set_callbacks	442
4.182.1.35	__HYPERVISOR_set_debugreg	442
4.182.1.36	__HYPERVISOR_set_gdt	442

4.182.1.37__HYPERVISOR_set_segment_base . . . . .	442
4.182.1.38__HYPERVISOR_set_timer_op . . . . .	442
4.182.1.39__HYPERVISOR_set_trap_table . . . . .	443
4.182.1.40__HYPERVISOR_stack_switch . . . . .	443
4.182.1.41__HYPERVISOR_sysctl . . . . .	443
4.182.1.42__HYPERVISOR_tmem_op . . . . .	443
4.182.1.43__HYPERVISOR_update_descriptor . . . . .	443
4.182.1.44__HYPERVISOR_update_va_mapping . . . . .	443
4.182.1.45__HYPERVISOR_update_va_mapping_otherdomain . . . . .	443
4.182.1.46__HYPERVISOR_vcpu_op . . . . .	443
4.182.1.47__HYPERVISOR_vm_assist . . . . .	443
4.182.1.48__HYPERVISOR_xen_version . . . . .	443
4.182.1.49__HYPERVISOR_xenoprof_op . . . . .	443
4.182.1.50__HYPERVISOR_xsm_op . . . . .	443
4.182.1.51__mk_unsigned_long . . . . .	443
4.182.1.52console_evtchn . . . . .	443
4.182.1.53console_mfn . . . . .	443
4.182.1.54CONSOLEIO_read . . . . .	443
4.182.1.55CONSOLEIO_write . . . . .	443
4.182.1.56DOMID_COW . . . . .	443
4.182.1.57DOMID_FIRST_RESERVED . . . . .	443
4.182.1.58DOMID_IDLE . . . . .	443
4.182.1.59DOMID_INVALID . . . . .	443
4.182.1.60DOMID_IO . . . . .	443
4.182.1.61DOMID_SELF . . . . .	443
4.182.1.62DOMID_XEN . . . . .	443
4.182.1.63MAX_GUEST_CMDLINE . . . . .	443
4.182.1.64MAX_VMASST_TYPE . . . . .	443
4.182.1.65mk_unsigned_long . . . . .	443
4.182.1.66MMU_MACHPHYS_UPDATE . . . . .	443
4.182.1.67MMU_NORMAL_PT_UPDATE . . . . .	444
4.182.1.68MMU_PT_UPDATE_PRESERVE_AD . . . . .	444
4.182.1.69MMUEXT_CLEAR_PAGE . . . . .	444
4.182.1.70MMUEXT_COPY_PAGE . . . . .	444
4.182.1.71MMUEXT_FLUSH_CACHE . . . . .	444
4.182.1.72MMUEXT_FLUSH_CACHE_GLOBAL . . . . .	444
4.182.1.73MMUEXT_INVLPG_ALL . . . . .	444
4.182.1.74MMUEXT_INVLPG_LOCAL . . . . .	444
4.182.1.75MMUEXT_INVLPG_MULTI . . . . .	444
4.182.1.76MMUEXT_MARK_SUPER . . . . .	444

4.182.1.77MMUEXT_NEW_BASEPTR . . . . .	444
4.182.1.78MMUEXT_NEW_USER_BASEPTR . . . . .	444
4.182.1.79MMUEXT_PIN_L1_TABLE . . . . .	444
4.182.1.80MMUEXT_PIN_L2_TABLE . . . . .	444
4.182.1.81MMUEXT_PIN_L3_TABLE . . . . .	444
4.182.1.82MMUEXT_PIN_L4_TABLE . . . . .	444
4.182.1.83MMUEXT_SET_LDT . . . . .	444
4.182.1.84MMUEXT_TLB_FLUSH_ALL . . . . .	444
4.182.1.85MMUEXT_TLB_FLUSH_LOCAL . . . . .	444
4.182.1.86MMUEXT_TLB_FLUSH_MULTI . . . . .	444
4.182.1.87MMUEXT_UNMARK_SUPER . . . . .	444
4.182.1.88MMUEXT_UNPIN_TABLE . . . . .	444
4.182.1.89NR_EVENT_CHANNELS . . . . .	444
4.182.1.90NR_VIRQS . . . . .	444
4.182.1.91SIF_INITDOMAIN . . . . .	444
4.182.1.92SIF_MOD_START_PFN . . . . .	444
4.182.1.93SIF_MULTIBOOT_MOD . . . . .	444
4.182.1.94SIF_PM_MASK . . . . .	444
4.182.1.95SIF_PRIVILEGED . . . . .	445
4.182.1.96JVMF_ALL . . . . .	445
4.182.1.97JVMF_FLUSHTYPE_MASK . . . . .	445
4.182.1.98JVMF_INVLPG . . . . .	445
4.182.1.99JVMF_LOCAL . . . . .	445
4.182.1.100JVMF_MULTI . . . . .	445
4.182.1.101JVMF_NONE . . . . .	445
4.182.1.102JVMF_TLB_FLUSH . . . . .	445
4.182.1.103IRQ_ARCH_0 . . . . .	445
4.182.1.104IRQ_ARCH_1 . . . . .	445
4.182.1.105IRQ_ARCH_2 . . . . .	445
4.182.1.106IRQ_ARCH_3 . . . . .	445
4.182.1.107IRQ_ARCH_4 . . . . .	445
4.182.1.108IRQ_ARCH_5 . . . . .	445
4.182.1.109IRQ_ARCH_6 . . . . .	445
4.182.1.110IRQ_ARCH_7 . . . . .	445
4.182.1.111IRQ_CON_RING . . . . .	445
4.182.1.112IRQ_CONSOLE . . . . .	445
4.182.1.113IRQ_DEBUG . . . . .	445
4.182.1.114IRQ_DEBUGGER . . . . .	445
4.182.1.115IRQ_DOM_EXC . . . . .	445
4.182.1.116IRQ_MEM_EVENT . . . . .	445

4.182.1.11	<a href="#">VIRQ_PCPU_STATE</a>	445
4.182.1.11	<a href="#">VIRQ_TBUF</a>	445
4.182.1.11	<a href="#">VIRQ_TIMER</a>	445
4.182.1.12	<a href="#">VIRQ_XENOPROF</a>	445
4.182.1.12	<a href="#">VMASST_CMD_disable</a>	445
4.182.1.12	<a href="#">VMASST_CMD_enable</a>	445
4.182.1.12	<a href="#">VMASST_TYPE_4gb_segments</a>	446
4.182.1.12	<a href="#">VMASST_TYPE_4gb_segments_notify</a>	446
4.182.1.12	<a href="#">VMASST_TYPE_pae_extended_cr3</a>	446
4.182.1.12	<a href="#">VMASST_TYPE_writable_pagetables</a>	446
4.182.1.12	<a href="#">xen_vga_console_info</a>	446
4.182.1.12	<a href="#">xen_vga_console_info_t</a>	446
4.182.1.12	<a href="#">XEN_VGATYPE_TEXT_MODE_3</a>	446
4.182.1.13	<a href="#">XEN_VGATYPE_VESA_LFB</a>	446
4.182.2	<a href="#">Typedef Documentation</a>	446
4.182.2.1	<a href="#">dom0_vga_console_info_t</a>	446
4.182.2.2	<a href="#">domid_t</a>	446
4.182.2.3	<a href="#">mmu_update_t</a>	446
4.182.2.4	<a href="#">mmuext_op_t</a>	446
4.182.2.5	<a href="#">multicall_entry_t</a>	446
4.182.2.6	<a href="#">shared_info_t</a>	446
4.182.2.7	<a href="#">start_info_t</a>	446
4.182.2.8	<a href="#">vcpu_info_t</a>	446
4.182.2.9	<a href="#">vcpu_time_info_t</a>	446
4.182.2.10	<a href="#">xen_domain_handle_t</a>	446
4.182.3	<a href="#">Function Documentation</a>	446
4.182.3.1	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.2	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.3	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.4	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.5	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.6	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.7	<a href="#">__DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.8	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	446
4.182.3.9	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447
4.182.3.10	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447
4.182.3.11	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447
4.182.3.12	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447
4.182.3.13	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447
4.182.3.14	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	447

4.182.3.15	DEFINE_XEN_GUEST_HANDLE	447
4.182.3.16	DEFINE_XEN_GUEST_HANDLE	447
4.183	xen/xen/include/public/xsm/caas_op.h File Reference	447
4.183.1	Macro Definition Documentation	447
4.183.1.1	CAAS_EXEMPT	447
4.183.1.2	CAAS_LAST	447
4.183.2	Typedef Documentation	447
4.183.2.1	caas_op_t	447
4.183.3	Function Documentation	447
4.183.3.1	DEFINE_XEN_GUEST_HANDLE	447
4.184	xen/xen/include/xen/domain.h File Reference	448
4.184.1	Function Documentation	448
4.184.1.1	alloc_dom0_vcpu0	448
4.184.1.2	alloc_domain_struct	448
4.184.1.3	alloc_domt_vcpu0	448
4.184.1.4	alloc_vcpu	448
4.184.1.5	alloc_vcpu_struct	448
4.184.1.6	arch_domain_create	449
4.184.1.7	arch_domain_destroy	449
4.184.1.8	arch_dump_domain_info	449
4.184.1.9	arch_dump_vcpu_info	449
4.184.1.10	arch_get_info_guest	449
4.184.1.11	arch_set_info_guest	449
4.184.1.12	arch_vcpu_reset	449
4.184.1.13	boot_vcpu	449
4.184.1.14	continue_hypercall_on_cpu	449
4.184.1.15	domain_relinquish_resources	449
4.184.1.16	domctl_lock_acquire	449
4.184.1.17	domctl_lock_release	449
4.184.1.18	dump_pageframe_info	449
4.184.1.19	free_domain_struct	449
4.184.1.20	free_vcpu_struct	449
4.184.1.21	getdomaininfo	449
4.184.1.22	vcpu_destroy	449
4.184.1.23	vcpu_initialise	449
4.184.1.24	vcpu_reset	449
4.184.2	Variable Documentation	449
4.184.2.1	xen_processor_pmbits	449
4.185	xen/xen/include/xen/sched.h File Reference	450
4.185.1	Macro Definition Documentation	453

4.185.1.1	_DOMCRF_dummy	453
4.185.1.2	_DOMCRF_hap	453
4.185.1.3	_DOMCRF_hvm	453
4.185.1.4	_DOMCRF_oos_off	453
4.185.1.5	_DOMCRF_priv	453
4.185.1.6	_DOMCRF_s3_integrity	453
4.185.1.7	_VPF_blocked	453
4.185.1.8	_VPF_blocked_in_xen	453
4.185.1.9	_VPF_down	453
4.185.1.10	_VPF_mem_event	453
4.185.1.11	_VPF_migrating	453
4.185.1.12	BITS_PER_EVTCHN_WORD	453
4.185.1.13	cpu_is_halttable	453
4.185.1.14	CPUPOOLID_NONE	454
4.185.1.15	domain_crash	454
4.185.1.16	domain_crash_synchronous	454
4.185.1.17	DOMAIN_DESTROYED	454
4.185.1.18	domain_is_locked	454
4.185.1.19	domain_lock	454
4.185.1.20	domain_unlock	454
4.185.1.21	DOMCRF_dummy	454
4.185.1.22	DOMCRF_hap	454
4.185.1.23	DOMCRF_hvm	454
4.185.1.24	DOMCRF_oos_off	454
4.185.1.25	DOMCRF_priv	454
4.185.1.26	DOMCRF_s3_integrity	454
4.185.1.27	ECS_FREE	454
4.185.1.28	ECS_INTERDOMAIN	454
4.185.1.29	ECS_IPI	454
4.185.1.30	ECS_PIRQ	454
4.185.1.31	ECS_RESERVED	454
4.185.1.32	ECS_UNBOUND	454
4.185.1.33	ECS_VIRQ	455
4.185.1.34	EVTCHNS_PER_BUCKET	455
4.185.1.35	for_each_domain	455
4.185.1.36	for_each_vcpu	455
4.185.1.37	hypercall_preempt_check	455
4.185.1.38	hvm_domain	455
4.185.1.39	hvm_vcpu	455
4.185.1.40	s_idle_domain	455

4.185.1.41	is_idle_vcpu	455
4.185.1.42	s_pinned_vcpu	455
4.185.1.43	S_PRIV	455
4.185.1.44	S_PRIV_FOR	455
4.185.1.45	MAX_EVTCHNS	455
4.185.1.46	need_iommu	455
4.185.1.47	NR_DOMAIN_WATCHDOG_TIMERS	455
4.185.1.48	NR_EVTCHN_BUCKETS	456
4.185.1.49	num_cpupool_cpus	456
4.185.1.50	PAEKERN_bimodal	456
4.185.1.51	PAEKERN_extended_cr3	456
4.185.1.52	PAEKERN_no	456
4.185.1.53	PAEKERN_yes	456
4.185.1.54	put_domain	456
4.185.1.55	runstate_guest	456
4.185.1.56	set_current_state	456
4.185.1.57	VM_ASSIST	456
4.185.1.58	VPF_blocked	456
4.185.1.59	VPF_blocked_in_xen	456
4.185.1.60	VPF_down	456
4.185.1.61	VPF_mem_event	456
4.185.1.62	VPF_migrating	456
4.185.2	Enumeration Type Documentation	456
4.185.2.1	cpufreq_controller	456
4.185.3	Function Documentation	456
4.185.3.1	__domain_crash	456
4.185.3.2	__domain_crash_synchronous	456
4.185.3.3	context_saved	457
4.185.3.4	context_switch	457
4.185.3.5	continue_running	457
4.185.3.6	cpu_disable_scheduler	457
4.185.3.7	cpu_init	457
4.185.3.8	cpupool_add_domain	457
4.185.3.9	cpupool_do_sysctl	457
4.185.3.10	cpupool_get_by_id	457
4.185.3.11	cpupool_put	457
4.185.3.12	cpupool_rm_domain	457
4.185.3.13	domain_create	457
4.185.3.14	domain_destroy	457
4.185.3.15	domain_kill	457



4.185.3.16	domain_pause	457
4.185.3.17	domain_pause_by_systemcontroller	457
4.185.3.18	domain_pause_for_debugger	457
4.185.3.19	domain_resume	457
4.185.3.20	domain_shutdown	457
4.185.3.21	domain_unpause	457
4.185.3.22	domain_unpause_by_systemcontroller	457
4.185.3.23	domain_update_node_affinity	457
4.185.3.24	evtchn_destroy	458
4.185.3.25	evtchn_destroy_final	458
4.185.3.26	evtchn_init	458
4.185.3.27	get_cpu_idle_time	458
4.185.3.28	get_domain_by_id	458
4.185.3.29	get_vcpu_migration_delay	458
4.185.3.30	hypercall_create_continuation	458
4.185.3.31	rcu_lock_domain_by_id	458
4.185.3.32	rcu_lock_remote_target_domain_by_id	458
4.185.3.33	rcu_lock_target_domain_by_id	458
4.185.3.34	sched_adjust	458
4.185.3.35	sched_adjust_global	458
4.185.3.36	sched_destroy_domain	458
4.185.3.37	sched_destroy_vcpu	458
4.185.3.38	sched_id	458
4.185.3.39	sched_init_domain	458
4.185.3.40	sched_init_vcpu	458
4.185.3.41	sched_move_domain	458
4.185.3.42	sched_tick_resume	459
4.185.3.43	sched_tick_suspend	459
4.185.3.44	schedule_cpu_switch	459
4.185.3.45	scheduler_alloc	459
4.185.3.46	scheduler_free	459
4.185.3.47	scheduler_get_default	459
4.185.3.48	scheduler_init	459
4.185.3.49	set_vcpu_migration_delay	459
4.185.3.50	startup_cpu_idle_loop	459
4.185.3.51	sync_local_execstate	459
4.185.3.52	sync_vcpu_execstate	459
4.185.3.53	vcpu_end_shutdown_deferral	459
4.185.3.54	vcpu_force_reschedule	459
4.185.3.55	vcpu_pause	459

4.185.3.56	<a href="#">cpu_pause_nosync</a>	459
4.185.3.57	<a href="#">cpu_runstate_get</a>	459
4.185.3.58	<a href="#">cpu_set_affinity</a>	459
4.185.3.59	<a href="#">cpu_sleep_nosync</a>	459
4.185.3.60	<a href="#">cpu_sleep_sync</a>	459
4.185.3.61	<a href="#">cpu_start_shutdown_deferral</a>	459
4.185.3.62	<a href="#">cpu_unblock</a>	459
4.185.3.63	<a href="#">cpu_unpause</a>	459
4.185.3.64	<a href="#">cpu_wake</a>	459
4.185.3.65	<a href="#">watchdog_domain_destroy</a>	459
4.185.3.66	<a href="#">watchdog_domain_init</a>	459
4.185.4	<a href="#">Variable Documentation</a>	460
4.185.4.1	<a href="#">cpufreq_controller</a>	460
4.185.4.2	<a href="#">dom0</a>	460
4.185.4.3	<a href="#">domain_list</a>	460
4.185.4.4	<a href="#">domlist_read_lock</a>	460
4.185.4.5	<a href="#">domlist_update_lock</a>	460
4.185.4.6	<a href="#">domt</a>	460
4.185.4.7	<a href="#">idle_vcpu</a>	460
4.185.4.8	<a href="#">sched_smt_power_savings</a>	460
4.186	<a href="#">xen/xen/include/xsm/xsm.h File Reference</a>	460
4.186.1	<a href="#">Macro Definition Documentation</a>	460
4.186.1.1	<a href="#">xsm_call</a>	460
4.186.1.2	<a href="#">XSM_MAGIC</a>	460
4.186.2	<a href="#">Typedef Documentation</a>	460
4.186.2.1	<a href="#">xsm_magic_t</a>	460
4.186.2.2	<a href="#">xsm_op_t</a>	461
4.186.3	<a href="#">Function Documentation</a>	461
4.186.3.1	<a href="#">DEFINE_XEN_GUEST_HANDLE</a>	461
4.186.4	<a href="#">Variable Documentation</a>	461
4.186.4.1	<a href="#">xsm_ops</a>	461
4.187	<a href="#">xen/xen/xsm/caas/hooks.c File Reference</a>	461
4.187.1	<a href="#">Macro Definition Documentation</a>	462
4.187.1.1	<a href="#">CAAS_ALLOWED_NONE</a>	462
4.187.1.2	<a href="#">CAAS_ALLOWED_T</a>	462
4.187.1.3	<a href="#">CAAS_ALLOWED_T_DUO</a>	462
4.187.1.4	<a href="#">CAAS_ALLOWED_Z</a>	462
4.187.1.5	<a href="#">CAAS_ALLOWED_Z_DUO</a>	462
4.187.1.6	<a href="#">CAAS_COPY_IN</a>	462
4.187.1.7	<a href="#">CAAS_COPY_OUT</a>	463

4.187.1.8 CAAS_DENY . . . . .	463
4.187.1.9 CAAS_DEVELOP . . . . .	463
4.187.1.10 CAAS_OK . . . . .	463
4.187.1.11 CAAS_PASS . . . . .	463
4.187.1.12 DOMID_DT . . . . .	463
4.187.1.13 DOMID_DZ . . . . .	463
4.187.1.14 EXBUF_N . . . . .	463
4.187.2 Function Documentation . . . . .	463
4.187.2.1 integer_param . . . . .	463
4.187.2.2 xsm_initcall . . . . .	463
4.187.3 Variable Documentation . . . . .	463
4.187.3.1 exbuf . . . . .	463
4.187.3.2 exbuf_i . . . . .	463
4.187.3.3 original_ops . . . . .	463
4.188 xen/xen/xsm/dummy.c File Reference . . . . .	463
4.188.1 Macro Definition Documentation . . . . .	464
4.188.1.1 set_to_dummy_if_null . . . . .	464
4.188.2 Function Documentation . . . . .	464
4.188.2.1 xsm_fixup_ops . . . . .	464
4.188.3 Variable Documentation . . . . .	464
4.188.3.1 dummy_xsm_ops . . . . .	464
4.189 xen/xen/xsm/xsm_core.c File Reference . . . . .	464
4.189.1 Function Documentation . . . . .	464
4.189.1.1 do_xsm_op . . . . .	464



# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">_SHA256_CTX</a>	9
<a href="#">_SHA512_CTX</a>	10
<a href="#">blk_buffer</a>	10
<a href="#">blk_req</a>	10
<a href="#">blkback_aiocb</a>	11
<a href="#">blkback_dev</a>	11
<a href="#">blkback_info</a>	13
<a href="#">blkfront_aiocb</a>	14
<a href="#">blkfront_dev</a>	14
<a href="#">blkfront_info</a>	15
<a href="#">boot_video_info</a>	16
<a href="#">buffer</a>	17
<a href="#">buffer_t</a>	18
<a href="#">caas_op</a>	18
<a href="#">cap_t</a>	19
<a href="#">changed_domain</a>	20
<a href="#">changed_node</a>	20
<a href="#">cipherInstance</a>	20
<a href="#">CRYPTO_INFO</a>	21
<a href="#">cspipe</a>	21
<a href="#">dom0_vga_console_info</a>	22
<a href="#">domain</a>	23
<a href="#">domain_create</a>	29
<a href="#">domain_setup_info</a>	30
<a href="#">domc_info</a>	31
<a href="#">duration_t</a>	31
<a href="#">evchn</a>	31
<a href="#">fe_gref_map</a>	33
<a href="#">keyInstance</a>	33
<a href="#">kvec</a>	34
<a href="#">libxl__cpuid_policy</a>	34
<a href="#">libxl__device</a>	35
<a href="#">libxl__device_model_starting</a>	35
<a href="#">libxl__gc</a>	36
<a href="#">libxl__spawn_starting</a>	36
<a href="#">libxl__xen_console_reader</a>	36
<a href="#">libxl_cpuarray</a>	37
<a href="#">libxl_cpumap</a>	37

libxl_ctx	38
libxl_domain_config	38
libxl_domain_suspend_info	40
libxl_event	40
libxl_file_reference	41
libxl_waiter	41
mem_event_domain	42
memop_args	42
memory_map_context	42
migrate_info	43
mmu_update	43
mmuext_op	44
multicall_entry	44
oiap_sess	45
Opt_args	45
osap_sess	47
pack_buf_t	47
pack_constbuf_t	47
permanent_flags_t	48
ptwr_emulate_ctxt	49
save_file_header	49
schedid_name	50
shared_info	50
shared_info_any_t	51
shpage	51
start_info	52
start_info_any_t	53
stclear_flags_t	54
suspendinfo	54
tc_info	55
tc_state	55
TCS_AUTH	56
tdVTPM_GLOBALS	56
tdVTPM_MIGKEY_LIST	58
timeout_t	58
tnmem_oid	59
TPM_BOUND_DATA	59
tpm_chip	59
tpm_cmd	61
tpm_cmd_header	61
tpm_cmd_params	61
TPM_DIGEST	62
tpm_flushspecific_in	63
tpm_getcap_params_in	63
tpm_getcap_params_out	64
tpm_getrandom_in	64
tpm_getrandom_out	64
tpm_hmac_ctx_t	65
tpm_input_header	65
TPM_KEY	65
TPM_KEY12	66
TPM_KEY_PARAMS	67
TPM_NONCE	68
tpm_oiap_out	68
tpm_osap_in	69
tpm_osap_out	69
tpm_output_header	70
TPM_PCR_COMPOSITE	70

TPM_PCR_INFO	70
TPM_PCR_SELECTION	71
tpm_pcrextend_in	71
tpm_pcrread_in	72
tpm_pcrread_out	72
TPM_PUBKEY	72
tpm_readpubek_params_out	73
TPM_RSA_KEY_PARMS	74
tpm_sha1_ctx_t	74
TPM_STORE_PUBKEY	75
TPM_STORED_DATA	75
TPM_VERSION	76
tpm_version_1_2_t	76
tpm_version_t	77
tpmback_dev	77
tpmcmd	78
tpmfront_dev	79
tpmif	80
transaction	81
vcpu	81
vcpu_guest_context_any_t	84
vcpu_guest_context_u	84
vcpu_info	84
vcpu_time_info	85
vfs_cmd	85
vfs_cmd_close	86
vfs_cmd_init	87
vfs_cmd_lseek	87
vfs_cmd_majorminor	88
vfs_cmd_open	88
vfs_cmd_read	89
vfs_cmd_stat	89
vfs_cmd_write	90
vfs_gref_map	90
vmcb	91
VTPM_DMI_RESOURCE_T	91
vtpm_ipc_handle_t	92
vtpm_thread_params_s	93
xc_core_header	93
xc_cpupoolinfo	94
xc_cx_stat	94
xc_dom_arch	95
xc_dom_image	96
xc_dom_loader	99
xc_dom_mem	100
xc_dom_phys	100
xc_dom_seg	101
xc_dominfo	101
xc_error	103
xc_get_cpufreq_para	103
xc_hypercall_buffer	104
xc_px_stat	105
xc_px_val	106
xen_domctl	106
xen_domctl_address_size	109
xen_domctl_arch_setup	109
xen_domctl_assign_device	110
xen_domctl_bind_pt_irq	110

xen_domctl_createdomain	111
xen_domctl_debug_op	112
xen_domctl_disable_migrate	112
xen_domctl_ext_vcpucontext	112
xen_domctl_gdbsx_domstatus	113
xen_domctl_gdbsx_memio	113
xen_domctl_gdbsx_pauseunp_vcpu	114
xen_domctl_get_device_group	114
xen_domctl_getdomaininfo	115
xen_domctl_getmemlist	116
xen_domctl_getpageframeinfo	116
xen_domctl_getpageframeinfo2	117
xen_domctl_getpageframeinfo3	117
xen_domctl_getvcpuinfo	118
xen_domctl_hvmcontext	118
xen_domctl_hvmcontext_partial	119
xen_domctl_hypercall_init	119
xen_domctl_iomem_permission	119
xen_domctl_ioport_mapping	120
xen_domctl_ioport_permission	120
xen_domctl_irq_permission	121
xen_domctl_max_mem	121
xen_domctl_max_vcpus	121
xen_domctl_mem_event_op	122
xen_domctl_mem_sharing_op	122
xen_domctl_memory_mapping	124
xen_domctl_pin_mem_cacheattr	124
xen_domctl_real_mode_area	125
xen_domctl_scheduler_op	125
xen_domctl_sendtrigger	126
xen_domctl_set_access_required	126
xen_domctl_set_opt_feature	127
xen_domctl_set_target	127
xen_domctl_setdebugging	127
xen_domctl_setdomainhandle	128
xen_domctl_settimeoffset	128
xen_domctl_shadow_op	128
xen_domctl_shadow_op_stats	129
xen_domctl_subscribe	129
xen_domctl_tsc_info	130
xen_domctl_vcpuaffinity	130
xen_domctl_vcpucontext	131
xen_guest_tsc_info	131
xen_multiboot_mod_list	132
xl_cmd	132
xl_cmd_dombuild	133
xl_cmd_domcreate	134
xl_cmd_dommake	134
xl_cmd_getpage	135
xl_cmd_ping	135
xl_cmd_retpage	136
XIOObject	136
yyalloc	136
YYLTYPE	137
YYSTYPE	137



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

xen/extras/mini-os/blkfront.c	148
xen/extras/mini-os/discovery.c	150
xen/extras/mini-os/kernel.c	168
xen/extras/mini-os/main.c	169
xen/extras/mini-os/tpm_tis.c	171
xen/extras/mini-os/tpmback.c	180
xen/extras/mini-os/tpmfront.c	185
xen/extras/mini-os/uuid.c	186
xen/extras/mini-os/vfs.c	186
xen/extras/mini-os/arch/ia64/iorw.c	139
xen/extras/mini-os/arch/x86/iorw.c	139
xen/extras/mini-os/arch/x86/mm.c	140
xen/extras/mini-os/console/console.c	150
xen/extras/mini-os/include/blkfront.h	151
xen/extras/mini-os/include/byteorder.h	152
xen/extras/mini-os/include/byteswap.h	153
xen/extras/mini-os/include/discovery.h	154
xen/extras/mini-os/include/endian.h	154
xen/extras/mini-os/include/iorw.h	155
xen/extras/mini-os/include/lib.h	156
xen/extras/mini-os/include/minios_macros.h	158
xen/extras/mini-os/include/tpm_tis.h	160
xen/extras/mini-os/include/tpmback.h	161
xen/extras/mini-os/include/tpmfront.h	163
xen/extras/mini-os/include/uuid.h	164
xen/extras/mini-os/include/vfs.h	164
xen/extras/mini-os/include/vfs_hooks.h	165
xen/extras/mini-os/include/vfs_intf.h	166
xen/extras/mini-os/include/ia64/arch_endian.h	154
xen/extras/mini-os/include/ia64/arch_wordsize.h	154
xen/extras/mini-os/include/x86/arch_endian.h	154
xen/extras/mini-os/include/x86/arch_limits.h	167
xen/extras/mini-os/include/x86/x86_32/arch_wordsize.h	155
xen/extras/mini-os/include/x86/x86_64/arch_wordsize.h	155
xen/extras/mini-os/lib/sys.c	169
xen/extras/mini-os/lib/xs.c	169
xen/stubdom/dmcc/blkback.c	189
xen/stubdom/dmcc/blkback.h	190

xen/stubdom/domc/crypt.c	191
xen/stubdom/domc/crypt.h	192
xen/stubdom/domc/cspipe_intf.h	206
xen/stubdom/domc/main.c	170
xen/stubdom/domc/main.h	207
xen/stubdom/domc/crypto/rijndael-alg-fst.c	193
xen/stubdom/domc/crypto/rijndael-alg-fst.h	194
xen/stubdom/domc/crypto/rijndael-api-fst.c	196
xen/stubdom/domc/crypto/rijndael-api-fst.h	197
xen/stubdom/domc/crypto/sha2.c	201
xen/stubdom/domc/crypto/sha2.h	204
xen/stubdom/domt/caas.c	208
xen/stubdom/domt/caas.h	209
xen/stubdom/domt/cspipe_intf.h	206
xen/stubdom/domt/main.c	171
xen/stubdom/domt/main.h	208
xen/stubdom/domt/tc.c	213
xen/stubdom/domt/tc.h	218
xen/stubdom/domt/tpm_defs.c	220
xen/stubdom/domt/tpm_defs.h	222
xen/stubdom/domt/xlc.c	229
xen/stubdom/domt/xlc.h	230
xen/stubdom/domt/xlc_intf.h	232
xen/stubdom/domt/xlc_util.c	232
xen/stubdom/domt/xlc_util.h	234
xen/stubdom/domt/crypto/hmac.c	210
xen/stubdom/domt/crypto/hmac.h	210
xen/stubdom/domt/crypto/sha1.c	211
xen/stubdom/domt/crypto/sha1.h	212
xen/stubdom/domt/integrated/minios_macros.h	158
xen/stubdom/domt/integrated/vfs.c	187
xen/stubdom/domt/integrated/vfs.h	164
xen/stubdom/domt/integrated/vfs_hooks.h	165
xen/stubdom/domt/integrated/vfs_intf.h	166
xen/stubdom/vtpm/vtpm.c	238
xen/stubdom/vtpm/vtpm.h	239
xen/stubdom/vtpm/vtpm_cmd.c	240
xen/stubdom/vtpm/vtpm_cmd.h	241
xen/stubdom/vtpm/vtpmblk.c	242
xen/stubdom/vtpm/vtpmblk.h	244
xen/stubdom/vtpm/compat/big_endian.h	235
xen/stubdom/vtpm/compat/edn_test.c	236
xen/stubdom/vtpm/compat/little_endian.h	236
xen/stubdom/vtpm/compat/swab.h	237
xen/tools/console/daemon/io.c	245
xen/tools/debugger/kdd/.kdd-xen.o.d	246
xen/tools/debugger/kdd/.kdd.o.d	246
xen/tools/ioemu-dir/config-host.h	265
xen/tools/ioemu-dir/alpha-linux-user/config.h	246
xen/tools/ioemu-dir/arm-linux-user/config.h	247
xen/tools/ioemu-dir/arm-softmmu/config.h	247
xen/tools/ioemu-dir/armeb-linux-user/config.h	248
xen/tools/ioemu-dir/cris-linux-user/config.h	249
xen/tools/ioemu-dir/cris-softmmu/config.h	249
xen/tools/ioemu-dir/i386-linux-user/config.h	249
xen/tools/ioemu-dir/i386-softmmu/config.h	250
xen/tools/ioemu-dir/m68k-linux-user/config.h	250
xen/tools/ioemu-dir/m68k-softmmu/config.h	251

xen/tools/ioemu-dir/mips-linux-user/config.h	252
xen/tools/ioemu-dir/mips-softmmu/config.h	252
xen/tools/ioemu-dir/mips64-softmmu/config.h	253
xen/tools/ioemu-dir/mips64el-softmmu/config.h	253
xen/tools/ioemu-dir/mipsel-linux-user/config.h	254
xen/tools/ioemu-dir/mipsel-softmmu/config.h	255
xen/tools/ioemu-dir/ppc-linux-user/config.h	255
xen/tools/ioemu-dir/ppc-softmmu/config.h	256
xen/tools/ioemu-dir/ppc64-linux-user/config.h	256
xen/tools/ioemu-dir/ppc64-softmmu/config.h	257
xen/tools/ioemu-dir/ppc64abi32-linux-user/config.h	258
xen/tools/ioemu-dir/ppcemb-softmmu/config.h	258
xen/tools/ioemu-dir/sh4-linux-user/config.h	259
xen/tools/ioemu-dir/sh4-softmmu/config.h	259
xen/tools/ioemu-dir/sh4eb-linux-user/config.h	260
xen/tools/ioemu-dir/sh4eb-softmmu/config.h	260
xen/tools/ioemu-dir/sparc-linux-user/config.h	261
xen/tools/ioemu-dir/sparc-softmmu/config.h	262
xen/tools/ioemu-dir/sparc32plus-linux-user/config.h	262
xen/tools/ioemu-dir/sparc64-linux-user/config.h	263
xen/tools/ioemu-dir/tests/test-mmap.c	266
xen/tools/ioemu-dir/x86_64-linux-user/config.h	263
xen/tools/ioemu-dir/x86_64-softmmu/config.h	264
xen/tools/libxc/xc_caas.c	266
xen/tools/libxc/xc_dom.h	266
xen/tools/libxc/xc_dom_core.c	270
xen/tools/libxc/xc_dom_x86.c	272
xen/tools/libxc/xenctrl.h	273
xen/tools/libxl/libxl.c	298
xen/tools/libxl/libxl.h	305
xen/tools/libxl/libxl.idl	317
xen/tools/libxl/libxl_blktap2.c	317
xen/tools/libxl/libxl_create.c	317
xen/tools/libxl/libxl_device.c	318
xen/tools/libxl/libxl_dm.c	320
xen/tools/libxl/libxl_dom.c	321
xen/tools/libxl/libxl_internal.c	322
xen/tools/libxl/libxl_internal.h	323
xen/tools/libxl/libxl_utils.c	330
xen/tools/libxl/libxl_uuid.h	333
xen/tools/libxl/libxlu_cfg_y.c	333
xen/tools/libxl/libxlu_cfg_y.h	340
xen/tools/libxl/xl_cmdimpl.c	342
xen/tools/libxlc/xlc.c	229
xen/tools/libxlc/xlc.h	231
xen/tools/libxlc/xlc_util.c	234
xen/tools/libxlc/xlc_util.h	235
xen/tools/libxlc/include/xlc_intf.h	232
xen/tools/python/xen/lowlevel/xl/xl.c	348
xen/tools/vtpm_manager/crypto/crypto.h	350
xen/tools/vtpm_manager/crypto/rsa.c	353
xen/tools/vtpm_manager/crypto/sym_crypto.c	354
xen/tools/vtpm_manager/manager/dmictl.c	356
xen/tools/vtpm_manager/manager/securestorage.c	357
xen/tools/vtpm_manager/manager/vtpm_ipc.c	358
xen/tools/vtpm_manager/manager/vtpm_ipc.h	359
xen/tools/vtpm_manager/manager/vtpm_manager.c	360
xen/tools/vtpm_manager/manager/vtpm_manager.h	361

xen/tools/vtpm_manager/manager/vtpm_manager_handler.c	363
xen/tools/vtpm_manager/manager/vtpmd.c	365
xen/tools/vtpm_manager/manager/vtpmpriv.h	366
xen/tools/vtpm_manager/manager/vtsp.c	368
xen/tools/vtpm_manager/migration/vtpm_manager_if.c	371
xen/tools/vtpm_manager/tcs/contextmgr.c	371
xen/tools/vtpm_manager/tcs/tcs.c	372
xen/tools/vtpm_manager/tcs/tpmddl.c	375
xen/tools/vtpm_manager/tcs/tpmddl.h	375
xen/tools/vtpm_manager/tcs/transmit.c	376
xen/tools/vtpm_manager/util/buffer.c	377
xen/tools/vtpm_manager/util/buffer.h	378
xen/tools/vtpm_manager/util/tcg.h	379
xen/tools/vtpm_manager/vtpmmgrtalk/vtpmmgrtalk.c	400
xen/tools/xenstore/xenstored_core.c	401
xen/tools/xenstore/xenstored_domain.c	403
xen/tools/xenstore/xenstored_transaction.c	406
xen/tools/xenvfsd/xenvfsd.c	407
xen/tools/xenvfsd/include/vfs_intf.h	167
xen/xen/arch/x86/domain_build.c	410
xen/xen/arch/x86/mm.c	143
xen/xen/arch/x86/setup.c	412
xen/xen/common/domain.c	415
xen/xen/common/domctl.c	419
xen/xen/common/memory.c	420
xen/xen/include/asm-x86/setup.h	420
xen/xen/include/public/domctl.h	422
xen/xen/include/public/xen.h	438
xen/xen/include/public/xsm/caas_op.h	447
xen/xen/include/xen/domain.h	448
xen/xen/include/xen/sched.h	450
xen/xen/include/xsm/xsm.h	460
xen/xen/xsm/dummy.c	463
xen/xen/xsm/xsm_core.c	464
xen/xen/xsm/caas/hooks.c	461

## Chapter 3

# Data Structure Documentation

### 3.1 `_SHA256_CTX` Struct Reference

```
#include <sha2.h>
```

#### Data Fields

- `uint32_t state` [8]
- `uint64_t bitcount`
- `uint8_t buffer` [`SHA256_BLOCK_LENGTH`]

#### 3.1.1 Detailed Description

NOTE: If your architecture does not define either `u_intXX_t` types or `uintXX_t` (from `inttypes.h`), you may need to define things by hand for your system: Most BSD systems already define `u_intXX_t` types, as does Linux. Some systems, however, like Compaq's Tru64 Unix instead can use `uintXX_t` types defined by very recent ANSI C standards and included in the file:

```
#include <inttypes.h>
```

If you choose to use `<inttypes.h>` then please define:

```
#define SHA2_USE_INTTYPES_H
```

Or on the command line during compile:

```
cc -DSHA2_USE_INTTYPES_H ...
```

#### 3.1.2 Field Documentation

##### 3.1.2.1 `uint64_t bitcount`

##### 3.1.2.2 `uint8_t buffer[SHA256_BLOCK_LENGTH]`

##### 3.1.2.3 `uint32_t state[8]`

The documentation for this struct was generated from the following file:

- `xen/stubdom/domc/crypto/sha2.h`

## 3.2 `_SHA512_CTX` Struct Reference

```
#include <sha2.h>
```

### Data Fields

- `uint64_t state` [8]
- `uint64_t bitcount` [2]
- `uint8_t buffer` [`SHA512_BLOCK_LENGTH`]

### 3.2.1 Field Documentation

3.2.1.1 `uint64_t bitcount`[2]

3.2.1.2 `uint8_t buffer`[`SHA512_BLOCK_LENGTH`]

3.2.1.3 `uint64_t state`[8]

The documentation for this struct was generated from the following file:

- `xen/stubdom/domc/crypto/sha2.h`

## 3.3 `blk_buffer` Struct Reference

### Data Fields

- `void * page`
- `grant_ref_t gref`

### 3.3.1 Field Documentation

3.3.1.1 `grant_ref_t gref`

3.3.1.2 `void* page`

The documentation for this struct was generated from the following file:

- `xen/extras/mini-os/blkfront.c`

## 3.4 `blk_req` Struct Reference

### Data Fields

- `struct blkfront_aiocb aiocb`
- `int rand_value`
- `struct blk_req * next`

### 3.4.1 Field Documentation

3.4.1.1 struct blkfront\_aiocb aiocb

3.4.1.2 struct blk\_req\* next

3.4.1.3 int rand\_value

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/kernel.c](#)

## 3.5 blkback\_aiocb Struct Reference

```
#include <blkback.h>
```

### Data Fields

- struct [blkback\\_dev](#) \* [b\\_dev](#)
- [blkif\\_request\\_t](#) \* [req](#)
- struct [blkfront\\_aiocb](#) \* [f\\_aiocb](#)
- [uint8\\_t](#) \* [pages\\_orig](#)
- [uint8\\_t](#) \* [pages\\_new](#)
- int [pages\\_order](#)

### 3.5.1 Detailed Description

Completely different that frontend aiocb

### 3.5.2 Field Documentation

3.5.2.1 struct blkback\_dev\* b\_dev

3.5.2.2 struct blkfront\_aiocb\* f\_aiocb

3.5.2.3 [uint8\\_t](#)\* [pages\\_new](#)

3.5.2.4 int [pages\\_order](#)

3.5.2.5 [uint8\\_t](#)\* [pages\\_orig](#)

3.5.2.6 [blkif\\_request\\_t](#)\* [req](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/blkback.h](#)

## 3.6 blkback\_dev Struct Reference

```
#include <blkback.h>
```

## Data Fields

- [domid\\_t dom](#)
- struct blkif\_back\_ring [ring](#)
- grant\_ref\_t [ring\\_ref](#)
- evtchn\_port\_t [evtchn](#)
- blkif\_vdev\_t [handle](#)
- char \* [nodename](#)
- char \* [frontend](#)
- struct blkback\_info [info](#)
- xenbus\_event\_queue [events](#)
- struct thread \* [b\\_thread](#)
- struct thread \* [f\\_thread](#)
- struct blkfront\_dev \* [f\\_dev](#)
- struct gntmap [gmap](#)
- [domc\\_info\\_t](#) \* [domc\\_info](#)
- evtchn\_port\_t [local\\_port](#)
- uint64\_t [count\\_read](#)
- uint64\_t [count\\_write](#)
- uint32\_t [outstanding](#)

### 3.6.1 Field Documentation

3.6.1.1 struct thread\* [b\\_thread](#)

3.6.1.2 uint64\_t [count\\_read](#)

3.6.1.3 uint64\_t [count\\_write](#)

3.6.1.4 [domid\\_t dom](#)

3.6.1.5 [domc\\_info\\_t](#)\* [domc\\_info](#)

3.6.1.6 xenbus\_event\_queue [events](#)

3.6.1.7 evtchn\_port\_t [evtchn](#)

3.6.1.8 struct blkfront\_dev\* [f\\_dev](#)

3.6.1.9 struct thread\* [f\\_thread](#)

3.6.1.10 char\* [frontend](#)

3.6.1.11 struct gntmap [gmap](#)

3.6.1.12 blkif\_vdev\_t [handle](#)

3.6.1.13 struct blkback\_info [info](#)

3.6.1.14 evtchn\_port\_t [local\\_port](#)

3.6.1.15 char\* [nodename](#)

3.6.1.16 uint32\_t [outstanding](#)



3.6.1.17 struct blkif\_back\_ring ring

3.6.1.18 grant\_ref\_t ring\_ref

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/blkback.h](#)

## 3.7 blkback\_info Struct Reference

```
#include <blkback.h>
```

### Data Fields

- uint64\_t [sectors](#)
- unsigned [sector\\_size](#)
- int [mode](#)
- int [info](#)
- int [barrier](#)
- int [flush](#)
- char \* [dev](#)
- char \* [type](#)
- char \* [hotplug\\_status](#)
- int [online](#)
- int [bootable](#)
- int [encrypt](#)

### 3.7.1 Field Documentation

3.7.1.1 int barrier

3.7.1.2 int bootable

3.7.1.3 char\* dev

3.7.1.4 int encrypt

3.7.1.5 int flush

3.7.1.6 char\* hotplug\_status

3.7.1.7 int info

3.7.1.8 int mode

3.7.1.9 int online

3.7.1.10 unsigned sector\_size

3.7.1.11 uint64\_t sectors

3.7.1.12 char\* type

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/blkback.h](#)

## 3.8 blkfront\_aiocb Struct Reference

```
#include <blkfront.h>
```

### Data Fields

- struct [blkfront\\_dev](#) \* [aio\\_dev](#)
- uint8\_t \* [aio\\_buf](#)
- size\_t [aio\\_nbytes](#)
- off\_t [aio\\_offset](#)
- size\_t [total\\_bytes](#)
- uint8\_t [is\\_write](#)
- void \* [data](#)
- grant\_ref\_t [gref](#) [BLKIF\_MAX\_SEGMENTS\_PER\_REQUEST]
- int [n](#)
- void(\* [aio\\_cb](#) )(struct [blkfront\\_aiocb](#) \*[aiocb](#), int ret)

### 3.8.1 Field Documentation

3.8.1.1 uint8\_t\* [aio\\_buf](#)

3.8.1.2 void(\* [aio\\_cb](#) )(struct [blkfront\\_aiocb](#) \*[aiocb](#), int ret)

3.8.1.3 struct [blkfront\\_dev](#)\* [aio\\_dev](#)

3.8.1.4 size\_t [aio\\_nbytes](#)

3.8.1.5 off\_t [aio\\_offset](#)

3.8.1.6 void\* [data](#)

3.8.1.7 grant\_ref\_t [gref](#)[BLKIF\_MAX\_SEGMENTS\_PER\_REQUEST]

3.8.1.8 uint8\_t [is\\_write](#)

3.8.1.9 int [n](#)

3.8.1.10 size\_t [total\\_bytes](#)

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/include/blkfront.h](#)

## 3.9 blkfront\_dev Struct Reference

### Data Fields

- domid\_t [dom](#)
- struct [blkif\\_front\\_ring](#) [ring](#)

- `grant_ref_t` [ring\\_ref](#)
- `evtchn_port_t` [evtchn](#)
- `blkif_vdev_t` [handle](#)
- `char *` [nodename](#)
- `char *` [backend](#)
- `struct blkfront_info` [info](#)
- `xenbus_event_queue` [events](#)

### 3.9.1 Field Documentation

3.9.1.1 `char*` [backend](#)

3.9.1.2 `domid_t` [dom](#)

3.9.1.3 `xenbus_event_queue` [events](#)

3.9.1.4 `evtchn_port_t` [evtchn](#)

3.9.1.5 `blkif_vdev_t` [handle](#)

3.9.1.6 `struct blkfront_info` [info](#)

3.9.1.7 `char*` [nodename](#)

3.9.1.8 `struct blkif_front_ring` [ring](#)

3.9.1.9 `grant_ref_t` [ring\\_ref](#)

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/blkfront.c](#)

## 3.10 blkfront\_info Struct Reference

```
#include <blkfront.h>
```

### Data Fields

- `uint64_t` [sectors](#)
- `unsigned` [sector\\_size](#)
- `int` [mode](#)
- `int` [info](#)
- `int` [barrier](#)
- `int` [flush](#)

### 3.10.1 Field Documentation

3.10.1.1 `int` [barrier](#)

3.10.1.2 `int` [flush](#)

3.10.1.3 `int` [info](#)

3.10.1.4 int mode

3.10.1.5 unsigned sector\_size

3.10.1.6 uint64\_t sectors

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/include/blkfront.h](#)

## 3.11 boot\_video\_info Struct Reference

### Data Fields

- [u8 orig\\_x](#)
- [u8 orig\\_y](#)
- [u8 orig\\_video\\_mode](#)
- [u8 orig\\_video\\_cols](#)
- [u8 orig\\_video\\_lines](#)
- [u8 orig\\_video\\_isVGA](#)
- [u16 orig\\_video\\_points](#)
- [u32 capabilities](#)
- [u16 lfb\\_linelength](#)
- [u16 lfb\\_width](#)
- [u16 lfb\\_height](#)
- [u16 lfb\\_depth](#)
- [u32 lfb\\_base](#)
- [u32 lfb\\_size](#)
- [u8 red\\_size](#)
- [u8 red\\_pos](#)
- [u8 green\\_size](#)
- [u8 green\\_pos](#)
- [u8 blue\\_size](#)
- [u8 blue\\_pos](#)
- [u8 rsvd\\_size](#)
- [u8 rsvd\\_pos](#)
- [u16 vesapm\\_seg](#)
- [u16 vesapm\\_off](#)
- [u16 vesa\\_attrib](#)

### 3.11.1 Field Documentation

3.11.1.1 u8 blue\_pos

3.11.1.2 u8 blue\_size

3.11.1.3 u32 capabilities

3.11.1.4 u8 green\_pos

3.11.1.5 u8 green\_size

3.11.1.6 u32 lfb\_base

- 3.11.1.7 u16 lfb\_depth
- 3.11.1.8 u16 lfb\_height
- 3.11.1.9 u16 lfb\_linelength
- 3.11.1.10 u32 lfb\_size
- 3.11.1.11 u16 lfb\_width
- 3.11.1.12 u8 orig\_video\_cols
- 3.11.1.13 u8 orig\_video\_isVGA
- 3.11.1.14 u8 orig\_video\_lines
- 3.11.1.15 u8 orig\_video\_mode
- 3.11.1.16 u16 orig\_video\_points
- 3.11.1.17 u8 orig\_x
- 3.11.1.18 u8 orig\_y
- 3.11.1.19 u8 red\_pos
- 3.11.1.20 u8 red\_size
- 3.11.1.21 u8 rsvd\_pos
- 3.11.1.22 u8 rsvd\_size
- 3.11.1.23 u16 vesa\_attrib
- 3.11.1.24 u16 vesapm\_off
- 3.11.1.25 u16 vesapm\_seg

The documentation for this struct was generated from the following file:

- [xen/xen/arch/x86/setup.c](#)

## 3.12 buffer Struct Reference

### Data Fields

- char \* [data](#)
- size\_t [consumed](#)
- size\_t [size](#)
- size\_t [capacity](#)
- size\_t [max\\_capacity](#)

### 3.12.1 Field Documentation

3.12.1.1 `size_t` capacity

3.12.1.2 `size_t` consumed

3.12.1.3 `char*` data

3.12.1.4 `size_t` max\_capacity

3.12.1.5 `size_t` size

The documentation for this struct was generated from the following file:

- [xen/tools/console/daemon/io.c](#)

## 3.13 `buffer_t` Struct Reference

```
#include <buffer.h>
```

### Data Fields

- [tpm\\_size\\_t](#) size
- [tpm\\_size\\_t](#) alloc\_size
- `BYTE *` bytes
- `BOOL` is\_owner

### 3.13.1 Field Documentation

3.13.1.1 `tpm_size_t` alloc\_size

3.13.1.2 `BYTE*` bytes

3.13.1.3 `BOOL` is\_owner

3.13.1.4 `tpm_size_t` size

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/buffer.h](#)

## 3.14 `caas_op` Struct Reference

```
#include <caas_op.h>
```

### Data Fields

- `uint32_t` cmd
- `uint32_t` size
- `char *` buf

### 3.14.1 Field Documentation

3.14.1.1 char\* buf

3.14.1.2 uint32\_t cmd

3.14.1.3 uint32\_t size

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xsm/caas\\_op.h](#)

## 3.15 cap\_t Union Reference

### Data Fields

- struct [permanent\\_flags\\_t](#) perm\_flags
- struct [stclear\\_flags\\_t](#) stclear\_flags
- bool owned
- uint32\_t num\_pcrs
- struct [tpm\\_version\\_t](#) tpm\_version
- struct [tpm\\_version\\_1\\_2\\_t](#) tpm\_version\_1\_2
- uint32\_t manufacturer\_id
- struct [timeout\\_t](#) timeout
- struct [duration\\_t](#) duration

### 3.15.1 Field Documentation

3.15.1.1 struct [duration\\_t](#) duration

3.15.1.2 uint32\_t manufacturer\_id

3.15.1.3 uint32\_t num\_pcrs

3.15.1.4 bool owned

3.15.1.5 struct [permanent\\_flags\\_t](#) perm\_flags

3.15.1.6 struct [stclear\\_flags\\_t](#) stclear\_flags

3.15.1.7 struct [timeout\\_t](#) timeout

3.15.1.8 struct [tpm\\_version\\_t](#) tpm\_version

3.15.1.9 struct [tpm\\_version\\_1\\_2\\_t](#) tpm\_version\_1\_2

The documentation for this union was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.16 changed\_domain Struct Reference

### Data Fields

- struct list\_head [list](#)
- unsigned int [domid](#)
- int [nbentry](#)

### 3.16.1 Field Documentation

3.16.1.1 unsigned int domid

3.16.1.2 struct list\_head list

3.16.1.3 int nbentry

The documentation for this struct was generated from the following file:

- [xen/tools/xenstore/xenstored\\_transaction.c](#)

## 3.17 changed\_node Struct Reference

### Data Fields

- struct list\_head [list](#)
- char \* [node](#)
- bool [recurse](#)

### 3.17.1 Field Documentation

3.17.1.1 struct list\_head list

3.17.1.2 char\* node

3.17.1.3 bool recurse

The documentation for this struct was generated from the following file:

- [xen/tools/xenstore/xenstored\\_transaction.c](#)

## 3.18 cipherInstance Struct Reference

```
#include <rijndael-api-fst.h>
```

### Data Fields

- [BYTE mode](#)
- [BYTE IV\\_old \[MAX\\_IV\\_SIZE\]](#)
- [BYTE \\* IV](#)



### 3.18.1 Field Documentation

#### 3.18.1.1 BYTE\* IV

#### 3.18.1.2 BYTE IV\_old[MAX\_IV\_SIZE]

#### 3.18.1.3 BYTE mode

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/crypto/rijndael-api-fst.h](#)

## 3.19 CRYPTO\_INFO Struct Reference

```
#include <crypto.h>
```

### Data Fields

- void \* [keyInfo](#)
- [UINT32](#) [algorithmID](#)
- [UINT32](#) [encScheme](#)
- [UINT32](#) [sigScheme](#)

### 3.19.1 Field Documentation

#### 3.19.1.1 UINT32 algorithmID

#### 3.19.1.2 UINT32 encScheme

#### 3.19.1.3 void\* keyInfo

#### 3.19.1.4 UINT32 sigScheme

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/crypto/crypto.h](#)

## 3.20 cspipe Struct Reference

```
#include <cspipe_intf.h>
```

### Data Fields

- [uint32\\_t](#) [magic\\_nr](#)
- [uint32\\_t](#) [domid\\_domu](#)
- [uint32\\_t](#) [domid\\_domc](#)
- char [s\\_key](#) [128/8]

### 3.20.1 Detailed Description

CaaS pipe info, placed in the DomC memory by the domain builder.

### 3.20.2 Field Documentation

3.20.2.1 `uint32_t domid_domc`

3.20.2.2 `uint32_t domid_domu`

3.20.2.3 `uint32_t magic_nr`

3.20.2.4 `char s_key[128/8]`

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/cspipe\\_intf.h](#)

## 3.21 `dom0_vga_console_info` Struct Reference

```
#include <xen.h>
```

### Data Fields

- `uint8_t video_type`
- `union {`
  - `struct {`
    - `uint16_t font_height`
    - `uint16_t cursor_x`
    - `uint16_t cursor_y`
    - `uint16_t rows`
    - `uint16_t columns`
  - `text_mode_3`
  - `struct {`
    - `uint16_t width`
    - `uint16_t height`
    - `uint16_t bytes_per_line`
    - `uint16_t bits_per_pixel`
    - `uint32_t lfb_base`
    - `uint32_t lfb_size`
    - `uint8_t red_pos`
    - `uint8_t red_size`
    - `uint8_t green_pos`
    - `uint8_t green_size`
    - `uint8_t blue_pos`
    - `uint8_t blue_size`
    - `uint8_t rsvd_pos`
    - `uint8_t rsvd_size`
  - `vesa_lfb`
- `} u`

### 3.21.1 Field Documentation

3.21.1.1 `uint16_t bits_per_pixel`

3.21.1.2 `uint8_t blue_pos`

- 3.21.1.3 `uint8_t blue_size`
- 3.21.1.4 `uint16_t bytes_per_line`
- 3.21.1.5 `uint16_t columns`
- 3.21.1.6 `uint16_t cursor_x`
- 3.21.1.7 `uint16_t cursor_y`
- 3.21.1.8 `uint16_t font_height`
- 3.21.1.9 `uint8_t green_pos`
- 3.21.1.10 `uint8_t green_size`
- 3.21.1.11 `uint16_t height`
- 3.21.1.12 `uint32_t lfb_base`
- 3.21.1.13 `uint32_t lfb_size`
- 3.21.1.14 `uint8_t red_pos`
- 3.21.1.15 `uint8_t red_size`
- 3.21.1.16 `uint16_t rows`
- 3.21.1.17 `uint8_t rsvd_pos`
- 3.21.1.18 `uint8_t rsvd_size`
- 3.21.1.19 `struct { ... } text_mode_3`
- 3.21.1.20 `union { ... } u`
- 3.21.1.21 `struct { ... } vesa_lfb`
- 3.21.1.22 `uint8_t video_type`
- 3.21.1.23 `uint16_t width`

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.22 domain Struct Reference

```
#include <sched.h>
```

### Public Types

- `enum { DOMDYING\_alive, DOMDYING\_dying, DOMDYING\_dead }`

## Public Member Functions

- [DECLARE\\_BITMAP](#) (poll\_mask, MAX\_VIRT\_CPUS)

## Data Fields

- int [domid](#)
- int [master\\_fd](#)
- int [slave\\_fd](#)
- int [log\\_fd](#)
- bool [is\\_dead](#)
- struct [buffer](#) [buffer](#)
- struct [domain](#) \* [next](#)
- char \* [conspath](#)
- int [ring\\_ref](#)
- [evtchn\\_port\\_or\\_error\\_t](#) [local\\_port](#)
- [evtchn\\_port\\_or\\_error\\_t](#) [remote\\_port](#)
- [xc\\_evtchn](#) \* [xce\\_handle](#)
- struct [xencons\\_interface](#) \* [interface](#)
- int [event\\_count](#)
- long long [next\\_period](#)
- struct [list\\_head](#) [list](#)
- unsigned int [domid](#)
- [evtchn\\_port\\_t](#) [port](#)
- [evtchn\\_port\\_t](#) [remote\\_port](#)
- unsigned long [mfn](#)
- char \* [path](#)
- struct [xenstore\\_domain\\_interface](#) \* [interface](#)
- struct [connection](#) \* [conn](#)
- int [shutdown](#)
- int [nbentry](#)
- int [nbwatch](#)
- [domid\\_t](#) [domain\\_id](#)
- [shared\\_info\\_t](#) \* [shared\\_info](#)
- [spinlock\\_t](#) [domain\\_lock](#)
- [spinlock\\_t](#) [page\\_alloc\\_lock](#)
- struct [page\\_list\\_head](#) [page\\_list](#)
- struct [page\\_list\\_head](#) [xenpage\\_list](#)
- unsigned int [tot\\_pages](#)
- unsigned int [max\\_pages](#)
- [atomic\\_t](#) [shr\\_pages](#)
- unsigned int [xenheap\\_pages](#)
- unsigned int [max\\_vcpus](#)
- void \* [sched\\_priv](#)
- struct [cpupool](#) \* [cpupool](#)
- struct [domain](#) \* [next\\_in\\_list](#)
- struct [domain](#) \* [next\\_in\\_hashbucket](#)
- struct [list\\_head](#) [rangesets](#)
- [spinlock\\_t](#) [rangesets\\_lock](#)
- struct [evtchn](#) \* [evtchn](#) [[NR\\_EVTCHN\\_BUCKETS](#)]
- [spinlock\\_t](#) [event\\_lock](#)
- struct [grant\\_table](#) \* [grant\\_table](#)
- unsigned int [nr\\_pirqs](#)
- [u16](#) \* [pirq\\_to\\_evtchn](#)

- unsigned long \* [pirq\\_mask](#)
- struct rangeset \* [iomem\\_caps](#)
- struct rangeset \* [irq\\_caps](#)
- bool\_t [is\\_hvm](#)
- bool\_t [need\\_iommu](#)
- bool\_t [is\\_privileged](#)
- struct domain \* [target](#)
- bool\_t [debugger\\_attached](#)
- enum domain:: { ... } [is\\_dying](#)
- bool\_t [is\\_paused\\_by\\_controller](#)
- bool\_t [is\\_pinned](#)
- spinlock\_t [shutdown\\_lock](#)
- bool\_t [is\\_shutting\\_down](#)
- bool\_t [is\\_shut\\_down](#)
- int [shutdown\\_code](#)
- int [suspend\\_evtchn](#)
- atomic\_t [pause\\_count](#)
- unsigned long [vm\\_assist](#)
- atomic\_t [refcnt](#)
- struct vcpu \*\* [vcpu](#)
- cpumask\_t [domain\\_dirty\\_cpumask](#)
- struct arch\_domain [arch](#)
- void \* [ssid](#)
- [xen\\_domain\\_handle\\_t](#) [handle](#)
- struct xenoprof \* [xenoprof](#)
- int32\_t [time\\_offset\\_seconds](#)
- spinlock\_t [watchdog\\_lock](#)
- uint32\_t [watchdog\\_inuse\\_map](#)
- struct timer [watchdog\\_timer](#) [NR\_DOMAIN\_WATCHDOG\_TIMERS]
- struct rcu\_head [rcu](#)
- spinlock\_t [hypercall\\_deadlock\\_mutex](#)
- void \* [tmem](#)
- struct lock\_profile\_qhead [profile\\_head](#)
- bool\_t [disable\\_migrate](#)
- struct mem\_event\_domain [mem\\_event](#)
- nodemask\_t [node\\_affinity](#)
- unsigned int [last\\_alloc\\_node](#)
- spinlock\_t [node\\_affinity\\_lock](#)

### 3.22.1 Member Enumeration Documentation

#### 3.22.1.1 anonymous enum

Enumerator

***DOMDYING\_alive***

***DOMDYING\_dying***

***DOMDYING\_dead***

### 3.22.2 Member Function Documentation

3.22.2.1 `DECLARE_BITMAP ( poll_mask , MAX_VIRT_CPUS )`

### 3.22.3 Field Documentation

3.22.3.1 `struct arch_domain arch`

3.22.3.2 `struct buffer buffer`

3.22.3.3 `struct connection* conn`

3.22.3.4 `char* conspath`

3.22.3.5 `struct cpupool* cpupool`

3.22.3.6 `bool_t debugger_attached`

3.22.3.7 `bool_t disable_migrate`

3.22.3.8 `cpumask_t domain_dirty_cpumask`

3.22.3.9 `domid_t domain_id`

3.22.3.10 `spinlock_t domain_lock`

3.22.3.11 `unsigned int domid`

3.22.3.12 `int domid`

3.22.3.13 `int event_count`

3.22.3.14 `spinlock_t event_lock`

3.22.3.15 `struct evtchn* evtchn[NR_EVTCHN_BUCKETS]`

3.22.3.16 `struct grant_table* grant_table`

3.22.3.17 `xen_domain_handle_t handle`

3.22.3.18 `spinlock_t hypercall_deadlock_mutex`

3.22.3.19 `struct xenstore_domain_interface* interface`

3.22.3.20 `struct xencons_interface* interface`

3.22.3.21 `struct rangeset* iomem_caps`

3.22.3.22 `struct rangeset* irq_caps`

3.22.3.23 `bool is_dead`

3.22.3.24 `enum { ... } is_dying`

3.22.3.25 `bool_t is_hvm`

- 3.22.3.26 `bool_t is_paused_by_controller`
- 3.22.3.27 `bool_t is_pinned`
- 3.22.3.28 `bool_t is_privileged`
- 3.22.3.29 `bool_t is_shut_down`
- 3.22.3.30 `bool_t is_shutting_down`
- 3.22.3.31 `unsigned int last_alloc_node`
- 3.22.3.32 `struct list_head list`
- 3.22.3.33 `evtchn_port_or_error_t local_port`
- 3.22.3.34 `int log_fd`
- 3.22.3.35 `int master_fd`
- 3.22.3.36 `unsigned int max_pages`
- 3.22.3.37 `unsigned int max_vcpus`
- 3.22.3.38 `struct mem_event_domain mem_event`
- 3.22.3.39 `unsigned long mfn`
- 3.22.3.40 `int nbentry`
- 3.22.3.41 `int nbwatch`
- 3.22.3.42 `bool_t need_iommu`
- 3.22.3.43 `struct domain* next`
- 3.22.3.44 `struct domain* next_in_hashbucket`
- 3.22.3.45 `struct domain* next_in_list`
- 3.22.3.46 `long long next_period`
- 3.22.3.47 `nodemask_t node_affinity`
- 3.22.3.48 `spinlock_t node_affinity_lock`
- 3.22.3.49 `unsigned int nr_pirqs`
- 3.22.3.50 `spinlock_t page_alloc_lock`
- 3.22.3.51 `struct page_list_head page_list`
- 3.22.3.52 `char* path`
- 3.22.3.53 `atomic_t pause_count`

- 3.22.3.54 unsigned long\* pirq\_mask
- 3.22.3.55 u16\* pirq\_to\_evtchn
- 3.22.3.56 evtchn\_port\_t port
- 3.22.3.57 struct lock\_profile\_qhead profile\_head
- 3.22.3.58 struct list\_head rangesets
- 3.22.3.59 spinlock\_t rangesets\_lock
- 3.22.3.60 struct rcu\_head rcu
- 3.22.3.61 atomic\_t refcnt
- 3.22.3.62 evtchn\_port\_t remote\_port
- 3.22.3.63 evtchn\_port\_or\_error\_t remote\_port
- 3.22.3.64 int ring\_ref
- 3.22.3.65 void\* sched\_priv
- 3.22.3.66 shared\_info\_t\* shared\_info
- 3.22.3.67 atomic\_t shr\_pages
- 3.22.3.68 int shutdown
- 3.22.3.69 int shutdown\_code
- 3.22.3.70 spinlock\_t shutdown\_lock
- 3.22.3.71 int slave\_fd
- 3.22.3.72 void\* ssid
- 3.22.3.73 int suspend\_evtchn
- 3.22.3.74 struct domain\* target
- 3.22.3.75 int32\_t time\_offset\_seconds
- 3.22.3.76 void\* tmem
- 3.22.3.77 unsigned int tot\_pages
- 3.22.3.78 struct vcpu\*\* vcpu
- 3.22.3.79 unsigned long vm\_assist
- 3.22.3.80 uint32\_t watchdog\_inuse\_map
- 3.22.3.81 spinlock\_t watchdog\_lock



3.22.3.82 struct timer watchdog\_timer[NR\_DOMAIN\_WATCHDOG\_TIMERS]

3.22.3.83 xc\_evtchn\* xce\_handle

3.22.3.84 unsigned int xenheap\_pages

3.22.3.85 struct xenoprof\* xenoprof

3.22.3.86 struct page\_list\_head xenpage\_list

The documentation for this struct was generated from the following files:

- [xen/tools/console/daemon/io.c](#)
- [xen/xen/include/xen/sched.h](#)
- [xen/tools/xenstore/xenstored\\_domain.c](#)

## 3.23 domain\_create Struct Reference

### Data Fields

- int [debug](#)
- int [daemonize](#)
- int [paused](#)
- int [dryrun](#)
- int [quiet](#)
- int [console\\_autoconnect](#)
- const char \* [config\\_file](#)
- const char \* [extra\\_config](#)
- const char \* [restore\\_file](#)
- int [migrate\\_fd](#)
- char \*\* [migration\\_domname\\_r](#)

### 3.23.1 Field Documentation

3.23.1.1 const char\* config\_file

3.23.1.2 int console\_autoconnect

3.23.1.3 int daemonize

3.23.1.4 int debug

3.23.1.5 int dryrun

3.23.1.6 const char\* extra\_config

3.23.1.7 int migrate\_fd

3.23.1.8 char\*\* migration\_domname\_r

3.23.1.9 int paused

3.23.1.10 int quiet

### 3.23.1.11 `const char* restore_file`

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/xl\\_cmdimpl.c](#)

## 3.24 `domain_setup_info` Struct Reference

```
#include <sched.h>
```

### Data Fields

- unsigned long [image\\_addr](#)
- unsigned long [image\\_len](#)
- unsigned long [v\\_start](#)
- unsigned long [v\\_end](#)
- unsigned long [v\\_kernstart](#)
- unsigned long [v\\_kernend](#)
- unsigned long [v\\_kernentry](#)
- unsigned int [pae\\_kernel](#)
- unsigned long [elf\\_paddr\\_offset](#)
- unsigned int [load\\_symtab](#)
- unsigned long [symtab\\_addr](#)
- unsigned long [symtab\\_len](#)

### 3.24.1 Field Documentation

3.24.1.1 unsigned long [elf\\_paddr\\_offset](#)

3.24.1.2 unsigned long [image\\_addr](#)

3.24.1.3 unsigned long [image\\_len](#)

3.24.1.4 unsigned int [load\\_symtab](#)

3.24.1.5 unsigned int [pae\\_kernel](#)

3.24.1.6 unsigned long [symtab\\_addr](#)

3.24.1.7 unsigned long [symtab\\_len](#)

3.24.1.8 unsigned long [v\\_end](#)

3.24.1.9 unsigned long [v\\_kernend](#)

3.24.1.10 unsigned long [v\\_kernentry](#)

3.24.1.11 unsigned long [v\\_kernstart](#)

3.24.1.12 unsigned long [v\\_start](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/xen/sched.h](#)

## 3.25 domc\_info Struct Reference

```
#include <main.h>
```

### Data Fields

- [domid\\_t self](#)
- [domid\\_t target](#)

### 3.25.1 Detailed Description

Custom types. Use domc\_ prefixes to avoid clashes.

### 3.25.2 Field Documentation

#### 3.25.2.1 domid\_t self

#### 3.25.2.2 domid\_t target

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/main.h](#)

## 3.26 duration\_t Struct Reference

### Data Fields

- uint32\_t [tpm\\_short](#)
- uint32\_t [tpm\\_medium](#)
- uint32\_t [tpm\\_long](#)

### 3.26.1 Field Documentation

#### 3.26.1.1 uint32\_t tpm\_long

#### 3.26.1.2 uint32\_t tpm\_medium

#### 3.26.1.3 uint32\_t tpm\_short

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.27 evtchn Struct Reference

```
#include <sched.h>
```

## Data Fields

- [u8 state](#)
- [u8 consumer\\_is\\_xen](#)
- [u16 notify\\_vcpu\\_id](#)
- union {
  - struct {
    - [domid\\_t remote\\_domid](#)
  - } [unbound](#)
  - struct {
    - [u16 remote\\_port](#)
    - struct [domain](#) \* [remote\\_dom](#)
  - } [interdomain](#)
  - struct {
    - [u16 irq](#)
    - [u16 next\\_port](#)
    - [u16 prev\\_port](#)
  - } [pirq](#)
  - [u16 virq](#)
- } [u](#)

### 3.27.1 Field Documentation

3.27.1.1 [u8 consumer\\_is\\_xen](#)

3.27.1.2 [struct { ... } interdomain](#)

3.27.1.3 [u16 irq](#)

3.27.1.4 [u16 next\\_port](#)

3.27.1.5 [u16 notify\\_vcpu\\_id](#)

3.27.1.6 [struct { ... } pirq](#)

3.27.1.7 [u16 prev\\_port](#)

3.27.1.8 [struct domain\\*](#) [remote\\_dom](#)

3.27.1.9 [domid\\_t](#) [remote\\_domid](#)

3.27.1.10 [u16 remote\\_port](#)

3.27.1.11 [u8 state](#)

3.27.1.12 [union { ... } u](#)

3.27.1.13 [struct { ... } unbound](#)

3.27.1.14 [u16 virq](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/xen/sched.h](#)

## 3.28 fe\_gref\_map Struct Reference

### Data Fields

- struct [vfs\\_gref\\_map](#) \* [gmap](#)
- int [gref](#)
- int [order](#)
- void \* [pgmap](#)
- void \* [pbuf](#)

### 3.28.1 Detailed Description

tracks some administration for the frontend for read/write

### 3.28.2 Field Documentation

3.28.2.1 struct [vfs\\_gref\\_map](#)\* [gmap](#)

3.28.2.2 int [gref](#)

3.28.2.3 int [order](#)

3.28.2.4 void \* [pbuf](#)

3.28.2.5 void\* [pgmap](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs.c](#)

## 3.29 keyInstance Struct Reference

```
#include <rijndael-api-fst.h>
```

### Data Fields

- [BYTE](#) [direction](#)
- int [keyLen](#)
- char [keyMaterial](#) [[MAX\\_KEY\\_SIZE](#)+1]
- int [Nr](#)
- [u32](#) [rk](#) [4 \* ([MAXNR](#)+1)]
- [u32](#) [ek](#) [4 \* ([MAXNR](#)+1)]

### 3.29.1 Field Documentation

3.29.1.1 [BYTE](#) [direction](#)

3.29.1.2 [u32](#) [ek](#)[4 \* ([MAXNR](#)+1)]

3.29.1.3 int [keyLen](#)

3.29.1.4 char keyMaterial[MAX\_KEY\_SIZE+1]

3.29.1.5 int Nr

3.29.1.6 u32 rk[4\*(MAXNR+1)]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domc/crypto/rijndael-api-fst.h](#)

## 3.30 kvec Struct Reference

```
#include <lib.h>
```

### Data Fields

- void \* [iov\\_base](#)
- size\_t [iov\\_len](#)

### 3.30.1 Field Documentation

3.30.1.1 void\* iov\_base

3.30.1.2 size\_t iov\_len

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/include/lib.h](#)

## 3.31 libxl\_\_cpuid\_policy Struct Reference

```
#include <libxl_internal.h>
```

### Data Fields

- uint32\_t [input](#) [2]
- char \* [policy](#) [4]

### 3.31.1 Field Documentation

3.31.1.1 uint32\_t input[2]

3.31.1.2 char\* policy[4]

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

## 3.32 libxl\_\_device Struct Reference

```
#include <libxl_internal.h>
```

### Data Fields

- uint32\_t [backend\\_devid](#)
- uint32\_t [backend\\_domid](#)
- uint32\_t [devid](#)
- uint32\_t [domid](#)
- [libxl\\_\\_device\\_kinds](#) [backend\\_kind](#)
- [libxl\\_\\_device\\_kinds](#) [kind](#)

### 3.32.1 Field Documentation

3.32.1.1 uint32\_t [backend\\_devid](#)

3.32.1.2 uint32\_t [backend\\_domid](#)

3.32.1.3 [libxl\\_\\_device\\_kinds](#) [backend\\_kind](#)

3.32.1.4 uint32\_t [devid](#)

3.32.1.5 uint32\_t [domid](#)

3.32.1.6 [libxl\\_\\_device\\_kinds](#) [kind](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

## 3.33 libxl\_\_device\_model\_starting Struct Reference

```
#include <libxl_internal.h>
```

### Data Fields

- [libxl\\_\\_spawn\\_starting](#) \* [for\\_spawn](#)
- char \* [dom\\_path](#)
- int [domid](#)

### 3.33.1 Field Documentation

3.33.1.1 char\* [dom\\_path](#)

3.33.1.2 int [domid](#)

3.33.1.3 [libxl\\_\\_spawn\\_starting](#)\* [for\\_spawn](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

### 3.34 libxl\_\_gc Struct Reference

```
#include <libxl_internal.h>
```

#### Data Fields

- int [alloc\\_maxsize](#)
- void \*\* [alloc\\_ptrs](#)
- [libxl\\_ctx](#) \* [owner](#)

#### 3.34.1 Field Documentation

3.34.1.1 int [alloc\\_maxsize](#)

3.34.1.2 void\*\* [alloc\\_ptrs](#)

3.34.1.3 [libxl\\_ctx](#)\* [owner](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

### 3.35 libxl\_\_spawn\_starting Struct Reference

```
#include <libxl_internal.h>
```

#### Data Fields

- pid\_t [intermediate](#)
- char \* [what](#)

#### 3.35.1 Field Documentation

3.35.1.1 pid\_t [intermediate](#)

3.35.1.2 char\* [what](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

### 3.36 libxl\_\_xen\_console\_reader Struct Reference

```
#include <libxl_internal.h>
```

#### Data Fields

- char \* [buffer](#)
- unsigned int [size](#)



- unsigned int [count](#)
- unsigned int [clear](#)
- unsigned int [incremental](#)
- unsigned int [index](#)

### 3.36.1 Field Documentation

#### 3.36.1.1 char\* [buffer](#)

#### 3.36.1.2 unsigned int [clear](#)

#### 3.36.1.3 unsigned int [count](#)

#### 3.36.1.4 unsigned int [incremental](#)

#### 3.36.1.5 unsigned int [index](#)

#### 3.36.1.6 unsigned int [size](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_internal.h](#)

## 3.37 libxl\_cpumap Struct Reference

```
#include <libxl.h>
```

### Data Fields

- uint32\_t [entries](#)
- uint32\_t \* [array](#)

### 3.37.1 Field Documentation

#### 3.37.1.1 uint32\_t\* [array](#)

#### 3.37.1.2 uint32\_t [entries](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.38 libxl\_cpumap Struct Reference

```
#include <libxl.h>
```

### Data Fields

- uint32\_t [size](#)
- uint8\_t \* [map](#)

### 3.38.1 Field Documentation

#### 3.38.1.1 uint8\_t\* map

#### 3.38.1.2 uint32\_t size

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.39 libxl\_ctx Struct Reference

```
#include <libxl.h>
```

### Data Fields

- xentoollog\_logger \* [lg](#)
- [xc\\_interface](#) \* [xch](#)
- struct xs\_handle \* [xsh](#)
- int(\* [waitpid\\_instead](#))(pid\_t pid, int \*status, int flags)
- libxl\_version\_info [version\\_info](#)

### 3.39.1 Field Documentation

#### 3.39.1.1 xentoollog\_logger\* lg

#### 3.39.1.2 libxl\_version\_info version\_info

#### 3.39.1.3 int(\* waitpid\_instead)(pid\_t pid, int \*status, int flags)

#### 3.39.1.4 xc\_interface\* xch

#### 3.39.1.5 struct xs\_handle\* xsh

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.40 libxl\_domain\_config Struct Reference

```
#include <libxl.h>
```

### Data Fields

- libxl\_domain\_create\_info [c\\_info](#)
- libxl\_domain\_build\_info [b\\_info](#)
- libxl\_device\_model\_info [dm\\_info](#)
- int [num\\_disks](#)
- int [num\\_vifs](#)
- int [num\\_vif2s](#)
- int [num\\_pcidevs](#)

- int [num\\_vfbs](#)
- int [num\\_vkbs](#)
- int [num\\_domc\\_disks](#)
- int [num\\_domc\\_vifs](#)
- libxl\_device\_disk \* [disks](#)
- libxl\_device\_nic \* [vifs](#)
- libxl\_device\_net2 \* [vif2s](#)
- libxl\_device\_pci \* [pcidevs](#)
- libxl\_device\_vfb \* [vfbs](#)
- libxl\_device\_vkb \* [vkbs](#)
- libxl\_device\_disk \* [domc\\_disks](#)
- libxl\_device\_nic \* [domc\\_vifs](#)
- enum [libxl\\_action\\_on\\_shutdown](#) [on\\_poweroff](#)
- enum [libxl\\_action\\_on\\_shutdown](#) [on\\_reboot](#)
- enum [libxl\\_action\\_on\\_shutdown](#) [on\\_watchdog](#)
- enum [libxl\\_action\\_on\\_shutdown](#) [on\\_crash](#)

### 3.40.1 Field Documentation

3.40.1.1 `libxl_domain_build_info b_info`

3.40.1.2 `libxl_domain_create_info c_info`

3.40.1.3 `libxl_device_disk* disks`

3.40.1.4 `libxl_device_model_info dm_info`

3.40.1.5 `libxl_device_disk* domc_disks`

3.40.1.6 `libxl_device_nic* domc_vifs`

3.40.1.7 `int num_disks`

3.40.1.8 `int num_domc_disks`

3.40.1.9 `int num_domc_vifs`

3.40.1.10 `int num_pcidevs`

3.40.1.11 `int num_vfbs`

3.40.1.12 `int num_vif2s`

3.40.1.13 `int num_vifs`

3.40.1.14 `int num_vkbs`

3.40.1.15 `enum libxl_action_on_shutdown on_crash`

3.40.1.16 `enum libxl_action_on_shutdown on_poweroff`

3.40.1.17 `enum libxl_action_on_shutdown on_reboot`

3.40.1.18 `enum libxl_action_on_shutdown on_watchdog`

3.40.1.19 `libxl_device_pci*` `pcidevs`

3.40.1.20 `libxl_device_vfb*` `vfbs`

3.40.1.21 `libxl_device_net2*` `vif2s`

3.40.1.22 `libxl_device_nic*` `vifs`

3.40.1.23 `libxl_device_vkb*` `vkbs`

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.41 `libxl_domain_suspend_info` Struct Reference

```
#include <libxl.h>
```

### Data Fields

- `int` [flags](#)
- `int(* suspend_callback)(void *, int)`

### 3.41.1 Field Documentation

3.41.1.1 `int` `flags`

3.41.1.2 `int(* suspend_callback)(void *, int)`

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.42 `libxl_event` Struct Reference

```
#include <libxl.h>
```

### Data Fields

- `libxl_event_type` `type`
- `char *` `path`
- `char *` `token`

### 3.42.1 Field Documentation

3.42.1.1 `char*` `path`

3.42.1.2 `char*` `token`

### 3.42.1.3 libxl\_event\_type type

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.43 libxl\_file\_reference Struct Reference

```
#include <libxl.h>
```

### Data Fields

- char \* [path](#)
- int [mapped](#)
- void \* [data](#)
- size\_t [size](#)

### 3.43.1 Field Documentation

3.43.1.1 void\* data

3.43.1.2 int mapped

3.43.1.3 char\* path

3.43.1.4 size\_t size

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.44 libxl\_waiter Struct Reference

```
#include <libxl.h>
```

### Data Fields

- char \* [path](#)
- char \* [token](#)

### 3.44.1 Field Documentation

3.44.1.1 char\* path

3.44.1.2 char\* token

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl.h](#)

## 3.45 mem\_event\_domain Struct Reference

```
#include <sched.h>
```

### Data Fields

- spinlock\_t [ring\\_lock](#)
- mem\_event\_shared\_page\_t \* [shared\\_page](#)
- void \* [ring\\_page](#)
- mem\_event\_front\_ring\_t [front\\_ring](#)
- int [xen\\_port](#)

### 3.45.1 Field Documentation

3.45.1.1 mem\_event\_front\_ring\_t [front\\_ring](#)

3.45.1.2 spinlock\_t [ring\\_lock](#)

3.45.1.3 void\* [ring\\_page](#)

3.45.1.4 mem\_event\_shared\_page\_t\* [shared\\_page](#)

3.45.1.5 int [xen\\_port](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/xen/sched.h](#)

## 3.46 memop\_args Struct Reference

### Data Fields

- struct [domain](#) \* [domain](#)

### 3.46.1 Field Documentation

3.46.1.1 struct [domain](#)\* [domain](#)

The documentation for this struct was generated from the following file:

- [xen/xen/common/memory.c](#)

## 3.47 memory\_map\_context Struct Reference

### Data Fields

- unsigned int [n](#)
- unsigned long [s](#)
- struct [xen\\_memory\\_map](#) [map](#)

### 3.47.1 Field Documentation

3.47.1.1 struct xen\_memory\_map map

3.47.1.2 unsigned int n

3.47.1.3 unsigned long s

The documentation for this struct was generated from the following file:

- [xen/xen/arch/x86/mm.c](#)

## 3.48 migrate\_info Struct Reference

### Data Fields

- long(\* [func](#))(void \*[data](#))
- void \* [data](#)
- struct [vcpu](#) \* [vcpu](#)
- unsigned int [cpu](#)
- unsigned int [nest](#)

### 3.48.1 Field Documentation

3.48.1.1 unsigned int cpu

3.48.1.2 void\* data

3.48.1.3 long(\* [func](#))(void \*[data](#))

3.48.1.4 unsigned int nest

3.48.1.5 struct [vcpu](#)\* [vcpu](#)

The documentation for this struct was generated from the following file:

- [xen/xen/common/domain.c](#)

## 3.49 mmu\_update Struct Reference

```
#include <xen.h>
```

### Data Fields

- uint64\_t [ptr](#)
- uint64\_t [val](#)

### 3.49.1 Field Documentation

3.49.1.1 uint64\_t ptr

### 3.49.1.2 uint64\_t val

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.50 mmuext\_op Struct Reference

```
#include <xen.h>
```

### Data Fields

- unsigned int [cmd](#)
- union {
  - [xen\\_pfn\\_t](#) [mfn](#)
  - unsigned long [linear\\_addr](#)
- } [arg1](#)
- union {
  - unsigned int [nr\\_ents](#)
  - const void \* [vcpumask](#)
  - [xen\\_pfn\\_t](#) [src\\_mfn](#)
- } [arg2](#)

### 3.50.1 Field Documentation

3.50.1.1 union { ... } [arg1](#)

3.50.1.2 union { ... } [arg2](#)

3.50.1.3 unsigned int [cmd](#)

3.50.1.4 unsigned long [linear\\_addr](#)

3.50.1.5 [xen\\_pfn\\_t](#) [mfn](#)

3.50.1.6 unsigned int [nr\\_ents](#)

3.50.1.7 [xen\\_pfn\\_t](#) [src\\_mfn](#)

3.50.1.8 const void\* [vcpumask](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.51 multicall\_entry Struct Reference

```
#include <xen.h>
```



## Data Fields

- unsigned long [op](#)
- unsigned long [result](#)
- unsigned long [args](#) [6]

### 3.51.1 Field Documentation

3.51.1.1 unsigned long args[6]

3.51.1.2 unsigned long op

3.51.1.3 unsigned long result

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.52 oiap\_sess Struct Reference

```
#include <tc.h>
```

## Data Fields

- uint32\_t [handle](#)
- unsigned char [enonce](#) [TPM\_DIGEST\_SIZE]

### 3.52.1 Field Documentation

3.52.1.1 unsigned char enonce[TPM\_DIGEST\_SIZE]

3.52.1.2 uint32\_t handle

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tc.h](#)

## 3.53 Opt\_args Struct Reference

```
#include <vtpm.h>
```

## Public Types

- enum [Mode](#) { [MODE\\_INVALID](#), [MODE\\_PVM](#), [MODE\\_HVM](#) }
- enum [StartUp](#) { [ST\\_CLEAR](#) = 1, [ST\\_SAVE](#) = 2, [ST\\_DEACTIVATED](#) = 3 }
- enum [EntropySource](#) { [ENT\\_TPM](#), [ENT\\_RAND](#) }

## Data Fields

- enum [Opt\\_args::Mode mode](#)
- enum [Opt\\_args::StartUp startup](#)
- enum [Opt\\_args::EntropySource entropysrc](#)
- int [loglevel](#)

### 3.53.1 Detailed Description

Global commandline options

### 3.53.2 Member Enumeration Documentation

#### 3.53.2.1 enum EntropySource

Enumerator

***ENT\_TPM***

***ENT\_RAND***

#### 3.53.2.2 enum Mode

Enumerator

***MODE\_INVALID***

***MODE\_PVM***

***MODE\_HVM***

#### 3.53.2.3 enum StartUp

Enumerator

***ST\_CLEAR***

***ST\_SAVE***

***ST\_DEACTIVATED***

### 3.53.3 Field Documentation

#### 3.53.3.1 enum Opt\_args::EntropySource entropysrc

#### 3.53.3.2 int loglevel

#### 3.53.3.3 enum Opt\_args::Mode mode

#### 3.53.3.4 enum Opt\_args::StartUp startup

The documentation for this struct was generated from the following file:

- [xen/stubdom/vtpm/vtpm.h](#)

## 3.54 osap\_sess Struct Reference

```
#include <tc.h>
```

### Data Fields

- uint32\_t [handle](#)
- unsigned char [secret](#) [TPM\_DIGEST\_SIZE]
- unsigned char [enonce](#) [TPM\_DIGEST\_SIZE]

### 3.54.1 Field Documentation

3.54.1.1 unsigned char [enonce](#)[TPM\_DIGEST\_SIZE]

3.54.1.2 uint32\_t [handle](#)

3.54.1.3 unsigned char [secret](#)[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tc.h](#)

## 3.55 pack\_buf\_t Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [UINT32](#) size
- [BYTE](#) \* [data](#)

### 3.55.1 Field Documentation

3.55.1.1 [BYTE](#)\* [data](#)

3.55.1.2 [UINT32](#) size

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.56 pack\_constbuf\_t Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [UINT32](#) size
- const [BYTE](#) \* [data](#)

### 3.56.1 Field Documentation

3.56.1.1 `const BYTE*` data

3.56.1.2 `UINT32` size

The documentation for this struct was generated from the following file:

- `xen/tools/vtpm_manager/util/tcg.h`

## 3.57 `permanent_flags_t` Struct Reference

### Data Fields

- `uint16_t` tag
- `uint8_t` disable
- `uint8_t` ownership
- `uint8_t` deactivated
- `uint8_t` readPubek
- `uint8_t` disableOwnerClear
- `uint8_t` allowMaintenance
- `uint8_t` physicalPresenceLifetimeLock
- `uint8_t` physicalPresenceHWEEnable
- `uint8_t` physicalPresenceCMDEnable
- `uint8_t` CEKPUSED
- `uint8_t` TPMpost
- `uint8_t` TPMpostLock
- `uint8_t` FIPS
- `uint8_t` operator
- `uint8_t` enableRevokeEK
- `uint8_t` nvLocked
- `uint8_t` readSRKPub
- `uint8_t` tpmEstablished
- `uint8_t` maintenanceDone
- `uint8_t` disableFullIDALogicInfo

### 3.57.1 Field Documentation

3.57.1.1 `uint8_t` allowMaintenance

3.57.1.2 `uint8_t` CEKPUSED

3.57.1.3 `uint8_t` deactivated

3.57.1.4 `uint8_t` disable

3.57.1.5 `uint8_t` disableFullIDALogicInfo

3.57.1.6 `uint8_t` disableOwnerClear

3.57.1.7 `uint8_t` enableRevokeEK

3.57.1.8 `uint8_t` FIPS

- 3.57.1.9    uint8\_t maintenanceDone
- 3.57.1.10    uint8\_t nvLocked
- 3.57.1.11    uint8\_t operator
- 3.57.1.12    uint8\_t ownership
- 3.57.1.13    uint8\_t physicalPresenceCMDEnable
- 3.57.1.14    uint8\_t physicalPresenceHWEEnable
- 3.57.1.15    uint8\_t physicalPresenceLifetimeLock
- 3.57.1.16    uint8\_t readPubek
- 3.57.1.17    uint8\_t readSRKPub
- 3.57.1.18    uint16\_t tag
- 3.57.1.19    uint8\_t tpmEstablished
- 3.57.1.20    uint8\_t TPMpost
- 3.57.1.21    uint8\_t TPMpostLock

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.58 ptwr\_emulate\_ctxt Struct Reference

### Data Fields

- struct x86\_emulate\_ctxt [ctxt](#)
- unsigned long [cr2](#)
- l1\_pgentry\_t [pte](#)

### 3.58.1 Field Documentation

- 3.58.1.1    unsigned long cr2
- 3.58.1.2    struct x86\_emulate\_ctxt ctxt
- 3.58.1.3    l1\_pgentry\_t pte

The documentation for this struct was generated from the following file:

- [xen/xen/arch/x86/mm.c](#)

## 3.59 save\_file\_header Struct Reference

## Data Fields

- char [magic](#) [32]
- uint32\_t [byteorder](#)
- uint32\_t [mandatory\\_flags](#)
- uint32\_t [optional\\_flags](#)
- uint32\_t [optional\\_data\\_len](#)

### 3.59.1 Field Documentation

3.59.1.1 [uint32\\_t byteorder](#)

3.59.1.2 [char magic\[32\]](#)

3.59.1.3 [uint32\\_t mandatory\\_flags](#)

3.59.1.4 [uint32\\_t optional\\_data\\_len](#)

3.59.1.5 [uint32\\_t optional\\_flags](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/xl\\_cmdimpl.c](#)

## 3.60 schedid\_name Struct Reference

### Data Fields

- char \* [name](#)
- int [id](#)

### 3.60.1 Field Documentation

3.60.1.1 [int id](#)

3.60.1.2 [char\\* name](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_utils.c](#)

## 3.61 shared\_info Struct Reference

```
#include <xen.h>
```

### Data Fields

- struct [vcpu\\_info vcpu\\_info](#) [XEN\_LEGACY\_MAX\_VCPUS]
- unsigned long [evtchn\\_pending](#) [sizeof(unsigned long)\*8]
- unsigned long [evtchn\\_mask](#) [sizeof(unsigned long)\*8]
- uint32\_t [wc\\_version](#)

- uint32\_t [wc\\_sec](#)
- uint32\_t [wc\\_nsec](#)
- struct arch\_shared\_info [arch](#)

### 3.61.1 Field Documentation

3.61.1.1 struct arch\_shared\_info arch

3.61.1.2 unsigned long evtchn\_mask[sizeof(unsigned long)\*8]

3.61.1.3 unsigned long evtchn\_pending[sizeof(unsigned long)\*8]

3.61.1.4 struct vcpu\_info vcpu\_info[XEN\_LEGACY\_MAX\_VCPUS]

3.61.1.5 uint32\_t wc\_nsec

3.61.1.6 uint32\_t wc\_sec

3.61.1.7 uint32\_t wc\_version

The documentation for this struct was generated from the following file:

- xen/xen/include/public/[xen.h](#)

## 3.62 shared\_info\_any\_t Union Reference

```
#include <xenctrl.h>
```

### Data Fields

- [shared\\_info\\_t s](#)

### 3.62.1 Field Documentation

3.62.1.1 shared\_info\_t s

The documentation for this union was generated from the following file:

- xen/tools/libxc/[xenctrl.h](#)

## 3.63 shpage Struct Reference

### Data Fields

- void \* [page](#)
- grant\_ref\_t [grantref](#)

### 3.63.1 Field Documentation

3.63.1.1 `grant_ref_t` grantref

3.63.1.2 `void*` page

The documentation for this struct was generated from the following file:

- [xen/stubdom/vtpm/vtpm\\_cmd.c](#)

## 3.64 `start_info` Struct Reference

```
#include <xen.h>
```

### Data Fields

- `char` [magic](#) [32]
- `unsigned long` [nr\\_pages](#)
- `unsigned long` [shared\\_info](#)
- `uint32_t` [flags](#)
- `xen_pfn_t` [store\\_mfn](#)
- `uint32_t` [store\\_evtchn](#)
- `union` {
  - `struct` {
    - `xen_pfn_t` [mfn](#)
    - `uint32_t` [evtchn](#)
  - `domU`
  - `struct` {
    - `uint32_t` [info\\_off](#)
    - `uint32_t` [info\\_size](#)
  - `dom0`
- `console`
- `unsigned long` [pt\\_base](#)
- `unsigned long` [nr\\_pt\\_frames](#)
- `unsigned long` [mfn\\_list](#)
- `unsigned long` [mod\\_start](#)
- `unsigned long` [mod\\_len](#)
- `int8_t` [cmd\\_line](#) [MAX\_GUEST\_CMDLINE]
- `unsigned long` [first\\_p2m\\_pfn](#)
- `unsigned long` [nr\\_p2m\\_frames](#)
- `unsigned long` [caas\\_mfn](#)

### 3.64.1 Field Documentation

3.64.1.1 `unsigned long` caas\_mfn

3.64.1.2 `int8_t` cmd\_line[MAX\_GUEST\_CMDLINE]

3.64.1.3 `union` { ... } console

3.64.1.4 `struct` { ... } dom0



- 3.64.1.5 struct { ... } domU
- 3.64.1.6 uint32\_t evtchn
- 3.64.1.7 unsigned long first\_p2m\_pfn
- 3.64.1.8 uint32\_t flags
- 3.64.1.9 uint32\_t info\_off
- 3.64.1.10 uint32\_t info\_size
- 3.64.1.11 char magic[32]
- 3.64.1.12 xen\_pfn\_t mfn
- 3.64.1.13 unsigned long mfn\_list
- 3.64.1.14 unsigned long mod\_len
- 3.64.1.15 unsigned long mod\_start
- 3.64.1.16 unsigned long nr\_p2m\_frames
- 3.64.1.17 unsigned long nr\_pages
- 3.64.1.18 unsigned long nr\_pt\_frames
- 3.64.1.19 unsigned long pt\_base
- 3.64.1.20 unsigned long shared\_info
- 3.64.1.21 uint32\_t store\_evtchn
- 3.64.1.22 xen\_pfn\_t store\_mfn

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.65 start\_info\_any\_t Union Reference

```
#include <xenctrl.h>
```

### Data Fields

- [start\\_info\\_t s](#)

### 3.65.1 Field Documentation

#### 3.65.1.1 start\_info\_t s

The documentation for this union was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.66 stclear\_flags\_t Struct Reference

### Data Fields

- uint16\_t [tag](#)
- uint8\_t [deactivated](#)
- uint8\_t [disableForceClear](#)
- uint8\_t [physicalPresence](#)
- uint8\_t [physicalPresenceLock](#)
- uint8\_t [bGlobalLock](#)

### 3.66.1 Field Documentation

3.66.1.1 uint8\_t bGlobalLock

3.66.1.2 uint8\_t deactivated

3.66.1.3 uint8\_t disableForceClear

3.66.1.4 uint8\_t physicalPresence

3.66.1.5 uint8\_t physicalPresenceLock

3.66.1.6 uint16\_t tag

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.67 suspendinfo Struct Reference

### Data Fields

- libxl\_\_gc \* [gc](#)
- xc\_evtchn \* [xce](#)
- int [suspend\\_eventchn](#)
- int [domid](#)
- int [hvm](#)
- unsigned int [flags](#)
- int [guest\\_responded](#)

### 3.67.1 Field Documentation

3.67.1.1 int domid

3.67.1.2 unsigned int flags

3.67.1.3 libxl\_\_gc\* gc

3.67.1.4 int guest\_responded

3.67.1.5 int hvm

3.67.1.6 int suspend\_eventchn

3.67.1.7 xc\_evtchn\* xce

The documentation for this struct was generated from the following file:

- [xen/tools/libxl/libxl\\_dom.c](#)

## 3.68 tc\_info Struct Reference

```
#include <tc.h>
```

### Data Fields

- struct [tpm\\_chip](#) \* chip
- [tpm\\_hmac\\_ctx\\_t](#) \* hmh

### 3.68.1 Detailed Description

build information used in nearly all TC funcs

### 3.68.2 Field Documentation

3.68.2.1 struct [tpm\\_chip](#)\* chip

3.68.2.2 [tpm\\_hmac\\_ctx\\_t](#)\* hmh

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tc.h](#)

## 3.69 tc\_state Struct Reference

```
#include <tc.h>
```

### Data Fields

- struct [tc\\_info](#) info
- struct [oiap\\_sess](#) \* oi
- unsigned char [srkpw](#) [TPM\_DIGEST\_SIZE]
- unsigned char [keypw](#) [TPM\_DIGEST\_SIZE]
- struct [TPM\\_KEY12](#) \* key
- uint32\_t [loadedKeyHandle](#)

### 3.69.1 Field Documentation

3.69.1.1 struct tc\_info info

3.69.1.2 struct TPM\_KEY12\* key

3.69.1.3 unsigned char keypw[TPM\_DIGEST\_SIZE]

3.69.1.4 uint32\_t loadedKeyHandle

3.69.1.5 struct oiap\_sess\* oi

3.69.1.6 unsigned char srkpw[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tc.h](#)

## 3.70 TCS\_AUTH Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [TCS\\_AUTHHANDLE AuthHandle](#)
- [TPM\\_NONCE NonceOdd](#)
- [TPM\\_NONCE NonceEven](#)
- [BOOL fContinueAuthSession](#)
- [TPM\\_AUTHDATA HMAC](#)

### 3.70.1 Field Documentation

3.70.1.1 TCS\_AUTHHANDLE AuthHandle

3.70.1.2 BOOL fContinueAuthSession

3.70.1.3 TPM\_AUTHDATA HMAC

3.70.1.4 TPM\_NONCE NonceEven

3.70.1.5 TPM\_NONCE NonceOdd

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.71 tdVTPM\_GLOBALS Struct Reference

```
#include <vtpmpriv.h>
```

## Data Fields

- pthread\_t [master\\_thread](#)
- pthread\_t [be\\_thread](#)
- pthread\_t [hp\\_thread](#)
- pthread\_t [dmi\\_thread](#)
- int [connected\\_dmis](#)
- struct hashtable \* [dmi\\_map](#)
- VTPM\_MIGKEY\_LIST \* [mig\\_keys](#)
- TCS\_CONTEXT\_HANDLE [manager\\_tcs\\_handle](#)
- TPM\_HANDLE [storageKeyHandle](#)
- CRYPTO\_INFO [storageKey](#)
- CRYPTO\_INFO [bootKey](#)
- TCS\_AUTH [keyAuth](#)
- TPM\_AUTHDATA [owner\\_usage\\_auth](#)
- buffer\_t [storageKeyWrap](#)
- TPM\_AUTHDATA [srk\\_usage\\_auth](#)
- TPM\_AUTHDATA [storage\\_key\\_usage\\_auth](#)
- buffer\_t [bootKeyWrap](#)

### 3.71.1 Field Documentation

- 3.71.1.1 pthread\_t [be\\_thread](#)
- 3.71.1.2 CRYPTO\_INFO [bootKey](#)
- 3.71.1.3 buffer\_t [bootKeyWrap](#)
- 3.71.1.4 int [connected\\_dmis](#)
- 3.71.1.5 struct hashtable\* [dmi\\_map](#)
- 3.71.1.6 pthread\_t [dmi\\_thread](#)
- 3.71.1.7 pthread\_t [hp\\_thread](#)
- 3.71.1.8 TCS\_AUTH [keyAuth](#)
- 3.71.1.9 TCS\_CONTEXT\_HANDLE [manager\\_tcs\\_handle](#)
- 3.71.1.10 pthread\_t [master\\_thread](#)
- 3.71.1.11 VTPM\_MIGKEY\_LIST\* [mig\\_keys](#)
- 3.71.1.12 TPM\_AUTHDATA [owner\\_usage\\_auth](#)
- 3.71.1.13 TPM\_AUTHDATA [srk\\_usage\\_auth](#)
- 3.71.1.14 TPM\_AUTHDATA [storage\\_key\\_usage\\_auth](#)
- 3.71.1.15 CRYPTO\_INFO [storageKey](#)
- 3.71.1.16 TPM\_HANDLE [storageKeyHandle](#)

#### 3.71.1.17 `buffer_t` storageKeyWrap

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/manager/vtpmpriv.h](#)

### 3.72 `tdVTPM_MIGKEY_LIST` Struct Reference

```
#include <vtpmpriv.h>
```

#### Data Fields

- [UINT32](#) `name_size`
- [BYTE](#) \* `name`
- [CRYPTO\\_INFO](#) `key`
- struct [tdVTPM\\_MIGKEY\\_LIST](#) \* `next`

#### 3.72.1 Field Documentation

##### 3.72.1.1 `CRYPTO_INFO` `key`

##### 3.72.1.2 `BYTE`\* `name`

##### 3.72.1.3 `UINT32` `name_size`

##### 3.72.1.4 struct `tdVTPM_MIGKEY_LIST`\* `next`

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/manager/vtpmpriv.h](#)

### 3.73 `timeout_t` Struct Reference

#### Data Fields

- `uint32_t` `a`
- `uint32_t` `b`
- `uint32_t` `c`
- `uint32_t` `d`

#### 3.73.1 Field Documentation

##### 3.73.1.1 `uint32_t` `a`

##### 3.73.1.2 `uint32_t` `b`

##### 3.73.1.3 `uint32_t` `c`

##### 3.73.1.4 `uint32_t` `d`

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.74 tmem\_oid Struct Reference

```
#include <xenctrl.h>
```

### Data Fields

- uint64\_t [oid](#) [3]

### 3.74.1 Detailed Description

tmem operations

### 3.74.2 Field Documentation

#### 3.74.2.1 uint64\_t oid[3]

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.75 TPM\_BOUND\_DATA Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [TPM\\_VERSION](#) ver
- [TPM\\_PAYLOAD\\_TYPE](#) payload
- [BYTE](#) \* [payloadData](#)

### 3.75.1 Field Documentation

#### 3.75.1.1 TPM\_PAYLOAD\_TYPE payload

#### 3.75.1.2 BYTE\* payloadData

#### 3.75.1.3 TPM\_VERSION ver

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.76 tpm\_chip Struct Reference

### Data Fields

- struct minios\_list\_head [list](#)

- int [enabled\\_localities](#)
- int [locality](#)
- unsigned long [baseaddr](#)
- uint8\_t \* [pages](#) [5]
- int [did](#)
- int [vid](#)
- int [rid](#)
- uint8\_t [data\\_buffer](#) [TPM\_BUFSIZE]
- int [data\\_len](#)
- s\_time\_t [timeout\\_a](#)
- s\_time\_t [timeout\\_b](#)
- s\_time\_t [timeout\\_c](#)
- s\_time\_t [timeout\\_d](#)
- s\_time\_t [duration](#) [3]
- unsigned int [irq](#)
- struct wait\_queue\_head [read\\_queue](#)
- struct wait\_queue\_head [int\\_queue](#)

### 3.76.1 Field Documentation

3.76.1.1 unsigned long [baseaddr](#)

3.76.1.2 uint8\_t [data\\_buffer](#)[TPM\_BUFSIZE]

3.76.1.3 int [data\\_len](#)

3.76.1.4 int [did](#)

3.76.1.5 s\_time\_t [duration](#)[3]

3.76.1.6 int [enabled\\_localities](#)

3.76.1.7 struct wait\_queue\_head [int\\_queue](#)

3.76.1.8 unsigned int [irq](#)

3.76.1.9 struct minios\_list\_head [list](#)

3.76.1.10 int [locality](#)

3.76.1.11 uint8\_t\* [pages](#)[5]

3.76.1.12 struct wait\_queue\_head [read\\_queue](#)

3.76.1.13 int [rid](#)

3.76.1.14 s\_time\_t [timeout\\_a](#)

3.76.1.15 s\_time\_t [timeout\\_b](#)

3.76.1.16 s\_time\_t [timeout\\_c](#)

3.76.1.17 s\_time\_t [timeout\\_d](#)



## 3.76.1.18 int vid

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.77 tpm\_cmd Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- [tpm\\_cmd\\_header header](#)
- [tpm\\_cmd\\_params params](#)

#### 3.77.1 Field Documentation

##### 3.77.1.1 tpm\_cmd\_header header

##### 3.77.1.2 tpm\_cmd\_params params

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.78 tpm\_cmd\_header Union Reference

```
#include <tpm_defs.h>
```

### Data Fields

- struct [tpm\\_input\\_header](#) in
- struct [tpm\\_output\\_header](#) out

#### 3.78.1 Field Documentation

##### 3.78.1.1 struct tpm\_input\_header in

##### 3.78.1.2 struct tpm\_output\_header out

The documentation for this union was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.79 tpm\_cmd\_params Union Reference

```
#include <tpm_defs.h>
```

## Data Fields

- struct [tpm\\_getcap\\_params\\_out](#) getcap\_out
- struct [tpm\\_readpubek\\_params\\_out](#) readpubek\_out
- uint8\_t [readpubek\\_out\\_buffer](#) [sizeof(struct [tpm\\_readpubek\\_params\\_out](#))]
- struct [tpm\\_getcap\\_params\\_in](#) getcap\_in
- struct [tpm\\_pcrread\\_in](#) pcrread\_in
- struct [tpm\\_pcrread\\_out](#) pcrread\_out
- struct [tpm\\_pcrextend\\_in](#) pcrextend\_in
- struct [tpm\\_oiap\\_out](#) oiap\_out
- struct [tpm\\_osap\\_in](#) osap\_in
- struct [tpm\\_osap\\_out](#) osap\_out
- struct [tpm\\_getrandom\\_in](#) getrandom\_in
- struct [tpm\\_getrandom\\_out](#) getrandom\_out
- struct [tpm\\_flushspecific\\_in](#) flushspecific\_in

### 3.79.1 Field Documentation

3.79.1.1 struct [tpm\\_flushspecific\\_in](#) flushspecific\_in

3.79.1.2 struct [tpm\\_getcap\\_params\\_in](#) getcap\_in

3.79.1.3 struct [tpm\\_getcap\\_params\\_out](#) getcap\_out

3.79.1.4 struct [tpm\\_getrandom\\_in](#) getrandom\_in

3.79.1.5 struct [tpm\\_getrandom\\_out](#) getrandom\_out

3.79.1.6 struct [tpm\\_oiap\\_out](#) oiap\_out

3.79.1.7 struct [tpm\\_osap\\_in](#) osap\_in

3.79.1.8 struct [tpm\\_osap\\_out](#) osap\_out

3.79.1.9 struct [tpm\\_pcrextend\\_in](#) pcrextend\_in

3.79.1.10 struct [tpm\\_pcrread\\_in](#) pcrread\_in

3.79.1.11 struct [tpm\\_pcrread\\_out](#) pcrread\_out

3.79.1.12 struct [tpm\\_readpubek\\_params\\_out](#) readpubek\_out

3.79.1.13 uint8\_t [readpubek\\_out\\_buffer](#)

The documentation for this union was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.80 TPM\_DIGEST Struct Reference

```
#include <tcg.h>
```

## Data Fields

- [BYTE digest](#) [[TPM\\_DIGEST\\_SIZE](#)]

### 3.80.1 Field Documentation

#### 3.80.1.1 [BYTE digest](#)[[TPM\\_DIGEST\\_SIZE](#)]

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.81 tpm\_flushspecific\_in Struct Reference

```
#include <tpm_defs.h>
```

## Data Fields

- [uint32\\_t handle](#)
- [uint32\\_t resourceType](#)

### 3.81.1 Field Documentation

#### 3.81.1.1 [uint32\\_t handle](#)

#### 3.81.1.2 [uint32\\_t resourceType](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.82 tpm\_getcap\_params\_in Struct Reference

## Data Fields

- [uint32\\_t cap](#)
- [uint32\\_t subcap\\_size](#)
- [uint32\\_t subcap](#)

### 3.82.1 Field Documentation

#### 3.82.1.1 [uint32\\_t cap](#)

#### 3.82.1.2 [uint32\\_t subcap](#)

#### 3.82.1.3 [uint32\\_t subcap\\_size](#)

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

### 3.83 tpm\_getcap\_params\_out Struct Reference

#### Data Fields

- uint32\_t [cap\\_size](#)
- [cap\\_t](#) cap

#### 3.83.1 Field Documentation

##### 3.83.1.1 [cap\\_t](#) cap

##### 3.83.1.2 uint32\_t cap\_size

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

### 3.84 tpm\_getrandom\_in Struct Reference

```
#include <tpm_defs.h>
```

#### Data Fields

- uint32\_t [numbytes](#)

#### 3.84.1 Field Documentation

##### 3.84.1.1 uint32\_t numbytes

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

### 3.85 tpm\_getrandom\_out Struct Reference

```
#include <tpm_defs.h>
```

#### Data Fields

- uint32\_t [numrandbytes](#)
- uint8\_t [randbytes](#) [MAX\_RAND\_SIZE]

#### 3.85.1 Field Documentation

##### 3.85.1.1 uint32\_t numrandbytes

##### 3.85.1.2 uint8\_t randbytes[MAX\_RAND\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.86 tpm\_hmac\_ctx\_t Struct Reference

```
#include <hmac.h>
```

### Data Fields

- [tpm\\_sha1\\_ctx\\_t ctx](#)
- [uint8\\_t k\\_opad \[HMAC\\_PAD\\_LENGTH\]](#)

### 3.86.1 Field Documentation

3.86.1.1 [tpm\\_sha1\\_ctx\\_t ctx](#)

3.86.1.2 [uint8\\_t k\\_opad\[HMAC\\_PAD\\_LENGTH\]](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/crypto/hmac.h](#)

## 3.87 tpm\_input\_header Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- [uint16\\_t tag](#)
- [uint32\\_t length](#)
- [uint32\\_t ordinal](#)

### 3.87.1 Field Documentation

3.87.1.1 [uint32\\_t length](#)

3.87.1.2 [uint32\\_t ordinal](#)

3.87.1.3 [uint16\\_t tag](#)

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.88 TPM\_KEY Struct Reference

```
#include <tcg.h>
```

## Data Fields

- [TPM\\_VERSION](#) `ver`
- [TPM\\_KEY\\_USAGE](#) `keyUsage`
- [TPM\\_KEY\\_FLAGS](#) `keyFlags`
- [TPM\\_AUTH\\_DATA\\_USAGE](#) `authDataUsage`
- [TPM\\_KEY\\_PARMS](#) `algorithmParms`
- [UINT32](#) `PCRInfoSize`
- [BYTE](#) \* `PCRInfo`
- [TPM\\_STORE\\_PUBKEY](#) `pubKey`
- [UINT32](#) `encDataSize`
- [BYTE](#) \* `encData`

### 3.88.1 Field Documentation

3.88.1.1 [TPM\\_KEY\\_PARMS](#) `algorithmParms`

3.88.1.2 [TPM\\_AUTH\\_DATA\\_USAGE](#) `authDataUsage`

3.88.1.3 [BYTE](#)\* `encData`

3.88.1.4 [UINT32](#) `encDataSize`

3.88.1.5 [TPM\\_KEY\\_FLAGS](#) `keyFlags`

3.88.1.6 [TPM\\_KEY\\_USAGE](#) `keyUsage`

3.88.1.7 [BYTE](#)\* `PCRInfo`

3.88.1.8 [UINT32](#) `PCRInfoSize`

3.88.1.9 [TPM\\_STORE\\_PUBKEY](#) `pubKey`

3.88.1.10 [TPM\\_VERSION](#) `ver`

The documentation for this struct was generated from the following file:

- `xen/tools/vtpm_manager/util/tcg.h`

## 3.89 TPM\_KEY12 Struct Reference

```
#include <tpm_defs.h>
```

## Data Fields

- [uint16\\_t](#) `tag`
- [uint16\\_t](#) `fill`
- [uint16\\_t](#) `keyUsage`
- [uint32\\_t](#) `keyFlags`
- [uint8\\_t](#) `authDataUsage`
- [struct](#) [TPM\\_KEY\\_PARMS](#) `algorithmParms`
- [uint32\\_t](#) `PCRInfoSize`
- [uint8\\_t](#) \* `PCRInfo`

- struct [TPM\\_STORE\\_PUBKEY](#) pubKey
- uint32\_t [encDataSize](#)
- uint8\_t \* [encData](#)

### 3.89.1 Field Documentation

3.89.1.1 struct **TPM\_KEY\_PARMS** algorithmParms

3.89.1.2 uint8\_t authDataUsage

3.89.1.3 uint8\_t\* encData

3.89.1.4 uint32\_t encDataSize

3.89.1.5 uint16\_t fill

3.89.1.6 uint32\_t keyFlags

3.89.1.7 uint16\_t keyUsage

3.89.1.8 uint8\_t\* PCRInfo

3.89.1.9 uint32\_t PCRInfoSize

3.89.1.10 struct **TPM\_STORE\_PUBKEY** pubKey

3.89.1.11 uint16\_t tag

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.90 TPM\_KEY\_PARMS Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- uint32\_t [algorithmID](#)
- uint16\_t [encScheme](#)
- uint16\_t [sigScheme](#)
- uint32\_t [parmSize](#)
- struct [TPM\\_RSA\\_KEY\\_PARMS](#) \* [parms](#)
- [TPM\\_ALGORITHM\\_ID](#) [algorithmID](#)
- [TPM\\_ENC\\_SCHEME](#) [encScheme](#)
- [TPM\\_SIG\\_SCHEME](#) [sigScheme](#)
- [UINT32](#) [parmSize](#)
- [BYTE](#) \* [parms](#)

### 3.90.1 Field Documentation

3.90.1.1 **TPM\_ALGORITHM\_ID** algorithmID

3.90.1.2 **uint32\_t** algorithmID

3.90.1.3 **TPM\_ENC\_SCHEME** encScheme

3.90.1.4 **uint16\_t** encScheme

3.90.1.5 **BYTE\*** parms

3.90.1.6 **struct TPM\_RSA\_KEY\_PARMS\*** parms

3.90.1.7 **UINT32** parmSize

3.90.1.8 **uint32\_t** parmSize

3.90.1.9 **TPM\_SIG\_SCHEME** sigScheme

3.90.1.10 **uint16\_t** sigScheme

The documentation for this struct was generated from the following files:

- [xen/stubdom/domt/tpm\\_defs.h](#)
- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.91 TPM\_NONCE Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [BYTE](#) nonce [TPM\_DIGEST\_SIZE]

### 3.91.1 Field Documentation

3.91.1.1 **BYTE** nonce[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.92 tpm\_oiap\_out Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- **uint32\_t** authHandle
- **uint8\_t** nonceEven [TPM\_DIGEST\_SIZE]



### 3.92.1 Field Documentation

3.92.1.1 uint32\_t authHandle

3.92.1.2 uint8\_t nonceEven[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.93 tpm\_osap\_in Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- uint16\_t [entity\\_type](#)
- uint32\_t [entity\\_value](#)
- uint8\_t [nonce\\_odd\\_osap](#) [TPM\_DIGEST\_SIZE]

### 3.93.1 Field Documentation

3.93.1.1 uint16\_t entity\_type

3.93.1.2 uint32\_t entity\_value

3.93.1.3 uint8\_t nonce\_odd\_osap[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.94 tpm\_osap\_out Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- uint32\_t [authhandle](#)
- uint8\_t [nonce\\_even](#) [TPM\_DIGEST\_SIZE]
- uint8\_t [nonce\\_even\\_osap](#) [TPM\_DIGEST\_SIZE]

### 3.94.1 Field Documentation

3.94.1.1 uint32\_t authhandle

3.94.1.2 uint8\_t nonce\_even[TPM\_DIGEST\_SIZE]

3.94.1.3 uint8\_t nonce\_even\_osap[TPM\_DIGEST\_SIZE]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/tpm\\_defs.h](#)

### 3.95 tpm\_output\_header Struct Reference

```
#include <tpm_defs.h>
```

#### Data Fields

- [uint16\\_t tag](#)
- [uint32\\_t length](#)
- [uint32\\_t return\\_code](#)

#### 3.95.1 Field Documentation

3.95.1.1 [uint32\\_t length](#)

3.95.1.2 [uint32\\_t return\\_code](#)

3.95.1.3 [uint16\\_t tag](#)

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

### 3.96 TPM\_PCR\_COMPOSITE Struct Reference

```
#include <tcg.h>
```

#### Data Fields

- [TPM\\_PCR\\_SELECTION select](#)
- [UINT32 valueSize](#)
- [TPM\\_PCRVALUE \\* pcrValue](#)

#### 3.96.1 Field Documentation

3.96.1.1 [TPM\\_PCRVALUE\\* pcrValue](#)

3.96.1.2 [TPM\\_PCR\\_SELECTION select](#)

3.96.1.3 [UINT32 valueSize](#)

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

### 3.97 TPM\_PCR\_INFO Struct Reference

```
#include <tcg.h>
```

## Data Fields

- [TPM\\_PCR\\_SELECTION pcrSelection](#)
- [TPM\\_COMPOSITE\\_HASH digestAtRelease](#)
- [TPM\\_COMPOSITE\\_HASH digestAtCreation](#)

### 3.97.1 Field Documentation

3.97.1.1 TPM\_COMPOSITE\_HASH digestAtCreation

3.97.1.2 TPM\_COMPOSITE\_HASH digestAtRelease

3.97.1.3 TPM\_PCR\_SELECTION pcrSelection

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.98 TPM\_PCR\_SELECTION Struct Reference

```
#include <tcg.h>
```

## Data Fields

- [UINT16 sizeOfSelect](#)
- [BYTE \\* pcrSelect](#)  
*in bytes*

### 3.98.1 Field Documentation

3.98.1.1 BYTE\* pcrSelect

in bytes

3.98.1.2 UINT16 sizeOfSelect

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.99 tpm\_pcrextend\_in Struct Reference

```
#include <tpm_defs.h>
```

## Data Fields

- [uint32\\_t pcr\\_idx](#)
- [uint8\\_t hash \[TPM\\_DIGEST\\_SIZE\]](#)

### 3.99.1 Field Documentation

#### 3.99.1.1 uint8\_t hash

#### 3.99.1.2 uint32\_t pcr\_idx

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.100 tpm\_pcrread\_in Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- uint32\_t [pcr\\_idx](#)

### 3.100.1 Field Documentation

#### 3.100.1.1 uint32\_t pcr\_idx

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.101 tpm\_pcrread\_out Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- uint8\_t [pcr\\_result](#) [TPM\_DIGEST\_SIZE]

### 3.101.1 Field Documentation

#### 3.101.1.1 uint8\_t pcr\_result

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

## 3.102 TPM\_PUBKEY Struct Reference

```
#include <tcg.h>
```

## Data Fields

- [TPM\\_KEY\\_PARMS](#) algorithmParms
- [TPM\\_STORE\\_PUBKEY](#) pubKey

### 3.102.1 Field Documentation

3.102.1.1 [TPM\\_KEY\\_PARMS](#) algorithmParms

3.102.1.2 [TPM\\_STORE\\_PUBKEY](#) pubKey

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.103 tpm\_readpubek\_params\_out Struct Reference

```
#include <tpm_defs.h>
```

## Data Fields

- [uint8\\_t](#) [algorithm](#) [4]
- [uint8\\_t](#) [encscheme](#) [2]
- [uint8\\_t](#) [sigscheme](#) [2]
- [uint32\\_t](#) [paramsize](#)
- [uint8\\_t](#) [parameters](#) [12]
- [uint32\\_t](#) [keysize](#)
- [uint8\\_t](#) [modulus](#) [256]
- [uint8\\_t](#) [checksum](#) [20]

### 3.103.1 Field Documentation

3.103.1.1 [uint8\\_t](#) [algorithm](#)

3.103.1.2 [uint8\\_t](#) [checksum](#)

3.103.1.3 [uint8\\_t](#) [encscheme](#)

3.103.1.4 [uint32\\_t](#) [keysize](#)

3.103.1.5 [uint8\\_t](#) [modulus](#)

3.103.1.6 [uint8\\_t](#) [parameters](#)

3.103.1.7 [uint32\\_t](#) [paramsize](#)

3.103.1.8 [uint8\\_t](#) [sigscheme](#)

The documentation for this struct was generated from the following files:

- [xen/extras/mini-os/tpm\\_tis.c](#)
- [xen/stubdom/domt/tpm\\_defs.h](#)

### 3.104 TPM\_RSA\_KEY\_PARMS Struct Reference

```
#include <tpm_defs.h>
```

#### Data Fields

- uint32\_t [keyLength](#)
- uint32\_t [numPrimes](#)
- uint32\_t [exponentSize](#)
- uint8\_t \* [exponent](#)
- UINT32 [keyLength](#)
- UINT32 [numPrimes](#)
- UINT32 [exponentSize](#)
- BYTE \* [exponent](#)

#### 3.104.1 Field Documentation

3.104.1.1 **BYTE\*** [exponent](#)

3.104.1.2 **uint8\_t\*** [exponent](#)

3.104.1.3 **UINT32** [exponentSize](#)

3.104.1.4 **uint32\_t** [exponentSize](#)

3.104.1.5 **UINT32** [keyLength](#)

3.104.1.6 **uint32\_t** [keyLength](#)

3.104.1.7 **UINT32** [numPrimes](#)

3.104.1.8 **uint32\_t** [numPrimes](#)

The documentation for this struct was generated from the following files:

- [xen/stubdom/domt/tpm\\_defs.h](#)
- [xen/tools/vtpm\\_manager/util/tcg.h](#)

### 3.105 tpm\_sha1\_ctx\_t Struct Reference

```
#include <sha1.h>
```

#### Data Fields

- uint32\_t [h](#) [5]
- uint32\_t [count\\_lo](#)
- uint32\_t [count\\_hi](#)
- uint8\_t [buf](#) [64]

### 3.105.1 Field Documentation

3.105.1.1 `uint8_t` `buf[64]`

3.105.1.2 `uint32_t` `count_hi`

3.105.1.3 `uint32_t` `count_lo`

3.105.1.4 `uint32_t` `h[5]`

The documentation for this struct was generated from the following file:

- `xen/stubdom/domt/crypto/sha1.h`

## 3.106 TPM\_STORE\_PUBKEY Struct Reference

```
#include <tpm_defs.h>
```

### Data Fields

- `uint32_t` `keyLength`
- `uint8_t *` `key`
- `UINT32` `keyLength`
- `BYTE *` `key`

### 3.106.1 Field Documentation

3.106.1.1 `BYTE*` `key`

3.106.1.2 `uint8_t*` `key`

3.106.1.3 `UINT32` `keyLength`

3.106.1.4 `uint32_t` `keyLength`

The documentation for this struct was generated from the following files:

- `xen/stubdom/domt/tpm_defs.h`
- `xen/tools/vtpm_manager/util/tcg.h`

## 3.107 TPM\_STORED\_DATA Struct Reference

```
#include <tcg.h>
```

### Data Fields

- `TPM_VERSION` `ver`
- `UINT32` `sealInfoSize`
- `BYTE *` `sealInfo`
- `UINT32` `encDataSize`
- `BYTE *` `encData`

### 3.107.1 Field Documentation

3.107.1.1 **BYTE\*** encData

3.107.1.2 **UINT32** encDataSize

3.107.1.3 **BYTE\*** sealInfo

3.107.1.4 **UINT32** sealInfoSize

3.107.1.5 **TPM\_VERSION** ver

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.108 TPM\_VERSION Struct Reference

```
#include <tcg.h>
```

### Data Fields

- [BYTE](#) major
- [BYTE](#) minor
- [BYTE](#) revMajor
- [BYTE](#) revMinor

### 3.108.1 Field Documentation

3.108.1.1 **BYTE** major

3.108.1.2 **BYTE** minor

3.108.1.3 **BYTE** revMajor

3.108.1.4 **BYTE** revMinor

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/util/tcg.h](#)

## 3.109 tpm\_version\_1\_2\_t Struct Reference

### Data Fields

- [uint16\\_t](#) tag
- [uint8\\_t](#) Major
- [uint8\\_t](#) Minor
- [uint8\\_t](#) revMajor
- [uint8\\_t](#) revMinor



### 3.109.1 Field Documentation

3.109.1.1 uint8\_t Major

3.109.1.2 uint8\_t Minor

3.109.1.3 uint8\_t revMajor

3.109.1.4 uint8\_t revMinor

3.109.1.5 uint16\_t tag

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.110 tpm\_version\_t Struct Reference

### Data Fields

- uint8\_t [Major](#)
- uint8\_t [Minor](#)
- uint8\_t [revMajor](#)
- uint8\_t [revMinor](#)

### 3.110.1 Field Documentation

3.110.1.1 uint8\_t Major

3.110.1.2 uint8\_t Minor

3.110.1.3 uint8\_t revMajor

3.110.1.4 uint8\_t revMinor

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpm\\_tis.c](#)

## 3.111 tpmback\_dev Struct Reference

### Data Fields

- [tpmif\\_t](#) \*\* [tpmlist](#)
- unsigned long [num\\_tpms](#)
- unsigned long [num\\_alloc](#)
- struct gntmap [map](#)
- int [flags](#)
- char \*\* [exclusive\\_uuids](#)
- xenbus\_event\_queue [events](#)
- void(\* [open\\_callback](#) )(domid\_t, unsigned int)
- void(\* [close\\_callback](#) )(domid\_t, unsigned int)
- void(\* [suspend\\_callback](#) )(domid\_t, unsigned int)
- void(\* [resume\\_callback](#) )(domid\_t, unsigned int)

### 3.111.1 Field Documentation

- 3.111.1.1 `void(* close_callback)(domid_t, unsigned int)`
- 3.111.1.2 `xenbus_event_queue events`
- 3.111.1.3 `char** exclusive_uuids`
- 3.111.1.4 `int flags`
- 3.111.1.5 `struct gntmap map`
- 3.111.1.6 `unsigned long num_alloc`
- 3.111.1.7 `unsigned long num_tpms`
- 3.111.1.8 `void(* open_callback)(domid_t, unsigned int)`
- 3.111.1.9 `void(* resume_callback)(domid_t, unsigned int)`
- 3.111.1.10 `void(* suspend_callback)(domid_t, unsigned int)`
- 3.111.1.11 `tpmif_t** tpmlist`

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpmback.c](#)

## 3.112 tpmcmd Struct Reference

```
#include <tpmback.h>
```

### Data Fields

- `domid_t domid`
- `unsigned int handle`
- `char * uuid`
- `unsigned int req_len`
- `uint8_t * req`
- `unsigned int resp_len`
- `uint8_t * resp`

### 3.112.1 Field Documentation

- 3.112.1.1 `domid_t domid`
- 3.112.1.2 `unsigned int handle`
- 3.112.1.3 `uint8_t* req`
- 3.112.1.4 `unsigned int req_len`
- 3.112.1.5 `uint8_t* resp`

3.112.1.6 unsigned int resp\_len

3.112.1.7 char\* uuid

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/include/tpmback.h](#)

## 3.113 tpmfront\_dev Struct Reference

```
#include <tpmfront.h>
```

### Data Fields

- grant\_ref\_t [ring\\_ref](#)
- evtchn\_port\_t [evtchn](#)
- tpmif\_tx\_interface\_t \* [tx](#)
- void \*\* [pages](#)
- domid\_t [bedomid](#)
- char \* [nodename](#)
- char \* [bepath](#)
- XenbusState [state](#)
- uint8\_t [waiting](#)
- struct wait\_queue\_head [waitq](#)
- uint8\_t \* [respbuf](#)
- size\_t [resplen](#)

### 3.113.1 Field Documentation

3.113.1.1 domid\_t [bedomid](#)

3.113.1.2 char\* [bepath](#)

3.113.1.3 evtchn\_port\_t [evtchn](#)

3.113.1.4 char\* [nodename](#)

3.113.1.5 void\*\* [pages](#)

3.113.1.6 uint8\_t\* [respbuf](#)

3.113.1.7 size\_t [resplen](#)

3.113.1.8 grant\_ref\_t [ring\\_ref](#)

3.113.1.9 XenbusState [state](#)

3.113.1.10 tpmif\_tx\_interface\_t\* [tx](#)

3.113.1.11 uint8\_t [waiting](#)

3.113.1.12 struct wait\_queue\_head [waitq](#)

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/include/tpmfront.h](#)

## 3.114 tpmif Struct Reference

### Public Types

- enum { [DISCONNECTED](#), [DISCONNECTING](#), [CONNECTED](#) }

### Data Fields

- [domid\\_t](#) domid
- unsigned int [handle](#)
- char \* [fe\\_path](#)
- char \* [fe\\_state\\_path](#)
- [evtchn\\_port\\_t](#) [evtchn](#)
- [tpmif\\_tx\\_interface\\_t](#) \* [tx](#)
- void \*\* [pages](#)
- enum [xenbus\\_state](#) [state](#)
- enum [tpmif::](#) { ... } [status](#)
- char \* [uuid](#)
- int [flags](#)

### 3.114.1 Member Enumeration Documentation

#### 3.114.1.1 anonymous enum

##### Enumerator

***DISCONNECTED***  
***DISCONNECTING***  
***CONNECTED***

### 3.114.2 Field Documentation

#### 3.114.2.1 [domid\\_t](#) domid

#### 3.114.2.2 [evtchn\\_port\\_t](#) [evtchn](#)

#### 3.114.2.3 [char\\*](#) [fe\\_path](#)

#### 3.114.2.4 [char\\*](#) [fe\\_state\\_path](#)

#### 3.114.2.5 [int](#) [flags](#)

#### 3.114.2.6 [unsigned int](#) [handle](#)

#### 3.114.2.7 [void\\*\\*](#) [pages](#)

#### 3.114.2.8 [enum](#) [xenbus\\_state](#) [state](#)

#### 3.114.2.9 [enum](#) { ... } [status](#)

3.114.2.10 `tpmif_tx_interface_t*` tx

3.114.2.11 `char*` uuid

The documentation for this struct was generated from the following file:

- [xen/extras/mini-os/tpmback.c](#)

## 3.115 transaction Struct Reference

### Data Fields

- struct list\_head [list](#)
- uint32\_t [id](#)
- unsigned int [generation](#)
- TDB\_CONTEXT \* [tdb](#)
- char \* [tdb\\_name](#)
- struct list\_head [changes](#)
- struct list\_head [changed\\_domains](#)

### 3.115.1 Field Documentation

3.115.1.1 struct list\_head [changed\\_domains](#)

3.115.1.2 struct list\_head [changes](#)

3.115.1.3 unsigned int [generation](#)

3.115.1.4 uint32\_t [id](#)

3.115.1.5 struct list\_head [list](#)

3.115.1.6 TDB\_CONTEXT\* [tdb](#)

3.115.1.7 char\* [tdb\\_name](#)

The documentation for this struct was generated from the following file:

- [xen/tools/xenstore/xenstored\\_transaction.c](#)

## 3.116 vcpu Struct Reference

```
#include <sched.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE](#) (vcpu\_runstate\_info\_t) [runstate\\_guest](#)

## Data Fields

- int `vcpu_id`
- int `processor`
- `vcpu_info_t` \* `vcpu_info`
- struct `domain` \* `domain`
- struct `vcpu` \* `next_in_list`
- `s_time_t` `periodic_period`
- `s_time_t` `periodic_last_event`
- struct timer `periodic_timer`
- struct timer `singleshot_timer`
- struct timer `poll_timer`
- void \* `sched_priv`
- struct `vcpu_runstate_info` `runstate`
- `uint64_t` `last_run_time`
- `bool_t` `fpu_initialised`
- `bool_t` `fpu_dirtied`
- `bool_t` `is_initialised`
- `bool_t` `is_running`
- `bool_t` `is_urgent`
- `bool_t` `defer_shutdown`
- `bool_t` `paused_for_shutdown`
- int `poll_evtchn`
- int `pirq_evtchn_head`
- unsigned long `pause_flags`
- `atomic_t` `pause_count`
- `u16` `virq_to_evtchn` [NR\_VIRQS]
- `spinlock_t` `virq_lock`
- `cpumask_t` `cpu_affinity`
- `cpumask_t` `cpu_affinity_tmp`
- `cpumask_t` `vcpu_dirty_cpumask`
- struct tasklet `continue_hypcall_tasklet`
- struct `mc_state` `mc_state`
- struct `waitqueue_vcpu` \* `waitqueue_vcpu`
- struct `arch_vcpu` `arch`

### 3.116.1 Member Function Documentation

3.116.1.1 `XEN_GUEST_HANDLE ( vcpu_runstate_info_t )`

### 3.116.2 Field Documentation

3.116.2.1 struct `arch_vcpu` `arch`

3.116.2.2 struct tasklet `continue_hypcall_tasklet`

3.116.2.3 `cpumask_t` `cpu_affinity`

3.116.2.4 `cpumask_t` `cpu_affinity_tmp`

3.116.2.5 `bool_t` `defer_shutdown`

3.116.2.6 struct `domain`\* `domain`

- 3.116.2.7 `bool_t fpu_dirtied`
- 3.116.2.8 `bool_t fpu_initialised`
- 3.116.2.9 `bool_t is_initialised`
- 3.116.2.10 `bool_t is_running`
- 3.116.2.11 `bool_t is_urgent`
- 3.116.2.12 `uint64_t last_run_time`
- 3.116.2.13 `struct mc_state mc_state`
- 3.116.2.14 `struct vcpu* next_in_list`
- 3.116.2.15 `atomic_t pause_count`
- 3.116.2.16 `unsigned long pause_flags`
- 3.116.2.17 `bool_t paused_for_shutdown`
- 3.116.2.18 `s_time_t periodic_last_event`
- 3.116.2.19 `s_time_t periodic_period`
- 3.116.2.20 `struct timer periodic_timer`
- 3.116.2.21 `int pirq_evtchn_head`
- 3.116.2.22 `int poll_evtchn`
- 3.116.2.23 `struct timer poll_timer`
- 3.116.2.24 `int processor`
- 3.116.2.25 `struct vcpu_runstate_info runstate`
- 3.116.2.26 `void* sched_priv`
- 3.116.2.27 `struct timer singleshoot_timer`
- 3.116.2.28 `cpumask_t vcpu_dirty_cpumask`
- 3.116.2.29 `int vcpu_id`
- 3.116.2.30 `vcpu_info_t* vcpu_info`
- 3.116.2.31 `spinlock_t virq_lock`
- 3.116.2.32 `u16 virq_to_evtchn[NR_VIRQS]`
- 3.116.2.33 `struct waitqueue_vcpu* waitqueue_vcpu`

The documentation for this struct was generated from the following file:

- `xen/xen/include/xen/sched.h`

### 3.117 `vcpu_guest_context_any_t` Union Reference

```
#include <xenctrl.h>
```

#### Data Fields

- `vcpu_guest_context_t c`

#### 3.117.1 Field Documentation

##### 3.117.1.1 `vcpu_guest_context_t c`

The documentation for this union was generated from the following file:

- `xen/tools/libxc/xenctrl.h`

### 3.118 `vcpu_guest_context_u` Union Reference

```
#include <domain.h>
```

#### Data Fields

- `struct vcpu_guest_context * nat`
- `struct compat_vcpu_guest_context * cmp`

#### 3.118.1 Field Documentation

##### 3.118.1.1 `struct compat_vcpu_guest_context* cmp`

##### 3.118.1.2 `struct vcpu_guest_context* nat`

The documentation for this union was generated from the following file:

- `xen/xen/include/xen/domain.h`

### 3.119 `vcpu_info` Struct Reference

```
#include <xen.h>
```

#### Data Fields

- `uint8_t evtchn_upcall_pending`
- `uint8_t evtchn_upcall_mask`
- `unsigned long evtchn_pending_sel`
- `struct arch_vcpu_info arch`
- `struct vcpu_time_info time`



### 3.119.1 Field Documentation

3.119.1.1 struct arch\_vcpu\_info arch

3.119.1.2 unsigned long evtchn\_pending\_sel

3.119.1.3 uint8\_t evtchn\_upcall\_mask

3.119.1.4 uint8\_t evtchn\_upcall\_pending

3.119.1.5 struct vcpu\_time\_info time

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.120 vcpu\_time\_info Struct Reference

```
#include <xen.h>
```

### Data Fields

- uint32\_t [version](#)
- uint32\_t [pad0](#)
- uint64\_t [tsc\\_timestamp](#)
- uint64\_t [system\\_time](#)
- uint32\_t [tsc\\_to\\_system\\_mul](#)
- int8\_t [tsc\\_shift](#)
- int8\_t [pad1](#) [3]

### 3.120.1 Field Documentation

3.120.1.1 uint32\_t pad0

3.120.1.2 int8\_t pad1[3]

3.120.1.3 uint64\_t system\_time

3.120.1.4 int8\_t tsc\_shift

3.120.1.5 uint64\_t tsc\_timestamp

3.120.1.6 uint32\_t tsc\_to\_system\_mul

3.120.1.7 uint32\_t version

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/xen.h](#)

## 3.121 vfs\_cmd Struct Reference

```
#include <vfs_intf.h>
```

## Data Fields

- enum [vfs\\_cmdtype](#) type
- union {
  - struct [vfs\\_cmd\\_init](#) init
  - struct [vfs\\_cmd\\_open](#) open
  - struct [vfs\\_cmd\\_close](#) close
  - struct [vfs\\_cmd\\_stat](#) stat
  - struct [vfs\\_cmd\\_lseek](#) lseek
  - struct [vfs\\_cmd\\_read](#) read
  - struct [vfs\\_cmd\\_write](#) write
  - struct [vfs\\_cmd\\_majorminor](#) majorminor
- } u

### 3.121.1 Field Documentation

3.121.1.1 struct [vfs\\_cmd\\_close](#) close

3.121.1.2 struct [vfs\\_cmd\\_init](#) init

3.121.1.3 struct [vfs\\_cmd\\_lseek](#) lseek

3.121.1.4 struct [vfs\\_cmd\\_majorminor](#) majorminor

3.121.1.5 struct [vfs\\_cmd\\_open](#) open

3.121.1.6 struct [vfs\\_cmd\\_read](#) read

3.121.1.7 struct [vfs\\_cmd\\_stat](#) stat

3.121.1.8 enum [vfs\\_cmdtype](#) type

3.121.1.9 union { ... } u

3.121.1.10 struct [vfs\\_cmd\\_write](#) write

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.122 [vfs\\_cmd\\_close](#) Struct Reference

```
#include <vfs_intf.h>
```

## Data Fields

- int [fd](#)
- int [ret](#)
- int [errn](#)

### 3.122.1 Field Documentation

3.122.1.1 int [errno](#)

3.122.1.2 int [fd](#)

3.122.1.3 int [ret](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.123 vfs\_cmd\_init Struct Reference

```
#include <vfs_intf.h>
```

### Data Fields

- int [ret](#)

### 3.123.1 Detailed Description

A) no fixed-width definitions are used for types going through the XenStore. They are used, however, for types placed in a shared page buffer.

B) The value of `errno` will only be used by the frontend if the return field indicates an error.

### 3.123.2 Field Documentation

3.123.2.1 int [ret](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.124 vfs\_cmd\_lseek Struct Reference

```
#include <vfs_intf.h>
```

### Data Fields

- int [fd](#)
- int [offset](#)
- int [whence](#)
- int [ret](#)
- int [errno](#)

### 3.124.1 Field Documentation

3.124.1.1 int [errn](#)

3.124.1.2 int [fd](#)

3.124.1.3 int [offset](#)

3.124.1.4 int [ret](#)

3.124.1.5 int [whence](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.125 vfs\_cmd\_majorminor Struct Reference

```
#include <vfs_intf.h>
```

### Data Fields

- char \* [path](#)
- int [major](#)
- int [minor](#)

### 3.125.1 Field Documentation

3.125.1.1 int [major](#)

3.125.1.2 int [minor](#)

3.125.1.3 char\* [path](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.126 vfs\_cmd\_open Struct Reference

```
#include <vfs_intf.h>
```

### Data Fields

- char \* [path](#)
- int [flags](#)
- int [mode](#)
- int [ret](#)
- int [errn](#)

### 3.126.1 Field Documentation

3.126.1.1 int `errn`

3.126.1.2 int `flags`

3.126.1.3 int `mode`

3.126.1.4 char\* `path`

3.126.1.5 int `ret`

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.127 vfs\_cmd\_read Struct Reference

```
#include <vfs_intf.h>
```

### Data Fields

- int `fd`
- int `gref`
- int `nbyte`
- int `ret`
- int `errn`

### 3.127.1 Field Documentation

3.127.1.1 int `errn`

3.127.1.2 int `fd`

3.127.1.3 int `gref`

3.127.1.4 int `nbyte`

3.127.1.5 int `ret`

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.128 vfs\_cmd\_stat Struct Reference

```
#include <vfs_intf.h>
```

## Data Fields

- char \* [path](#)
- int [gref](#)
- int [ret](#)
- int [errn](#)

### 3.128.1 Field Documentation

3.128.1.1 int [errn](#)

3.128.1.2 int [gref](#)

3.128.1.3 char\* [path](#)

3.128.1.4 int [ret](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.129 vfs\_cmd\_write Struct Reference

```
#include <vfs_intf.h>
```

## Data Fields

- int [fd](#)
- int [gref](#)
- int [nbyte](#)
- int [ret](#)
- int [errn](#)

### 3.129.1 Field Documentation

3.129.1.1 int [errn](#)

3.129.1.2 int [fd](#)

3.129.1.3 int [gref](#)

3.129.1.4 int [nbyte](#)

3.129.1.5 int [ret](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.130 vfs\_gref\_map Struct Reference

```
#include <vfs_intf.h>
```

## Data Fields

- uint32\_t [num](#)
- uint32\_t [grefs](#) [[GREFS\\_PER\\_GREFMAP](#)]

### 3.130.1 Field Documentation

3.130.1.1 uint32\_t [grefs](#) [[GREFS\\_PER\\_GREFMAP](#)]

3.130.1.2 uint32\_t [num](#)

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/integrated/vfs\\_intf.h](#)

## 3.131 vmcb Struct Reference

```
#include <caas.h>
```

## Data Fields

- char [s\\_key](#) [128/8]

### 3.131.1 Detailed Description

VM Control Blob. The vmcb and cspipe are strongly related, and before all functionality including the HMACs is fully implemented, they will look the same.

### 3.131.2 Field Documentation

3.131.2.1 char [s\\_key](#) [128/8]

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/caas.h](#)

## 3.132 VTPM\_DMI\_RESOURCE\_T Struct Reference

```
#include <vtprmppriv.h>
```

## Data Fields

- [vtpm\\_ipc\\_handle\\_t](#) \* [tx\\_vtpm\\_ipc\\_h](#)
- [vtpm\\_ipc\\_handle\\_t](#) \* [rx\\_vtpm\\_ipc\\_h](#)
- [vtpm\\_ipc\\_handle\\_t](#) \* [tx\\_tpm\\_ipc\\_h](#)
- [vtpm\\_ipc\\_handle\\_t](#) \* [rx\\_tpm\\_ipc\\_h](#)
- [pid\\_t](#) [dmi\\_pid](#)
- bool [connected](#)
- [UINT32](#) [dmi\\_domain\\_id](#)

- [TCS\\_CONTEXT\\_HANDLE](#) TCSTContext
- char \* [NVMLocation](#)
- [UINT32](#) dmi\_id
- [BYTE](#) dmi\_type
- [TPM\\_DIGEST](#) NVM\_measurement
- [TPM\\_DIGEST](#) DMI\_measurement

### 3.132.1 Field Documentation

3.132.1.1 bool connected

3.132.1.2 [UINT32](#) dmi\_domain\_id

3.132.1.3 [UINT32](#) dmi\_id

3.132.1.4 [TPM\\_DIGEST](#) DMI\_measurement

3.132.1.5 pid\_t dmi\_pid

3.132.1.6 [BYTE](#) dmi\_type

3.132.1.7 [TPM\\_DIGEST](#) NVM\_measurement

3.132.1.8 char\* [NVMLocation](#)

3.132.1.9 vtpm\_ipc\_handle\_t\* rx\_tpm\_ipc\_h

3.132.1.10 vtpm\_ipc\_handle\_t\* rx\_vtpm\_ipc\_h

3.132.1.11 [TCS\\_CONTEXT\\_HANDLE](#) TCSTContext

3.132.1.12 vtpm\_ipc\_handle\_t\* tx\_tpm\_ipc\_h

3.132.1.13 vtpm\_ipc\_handle\_t\* tx\_vtpm\_ipc\_h

The documentation for this struct was generated from the following file:

- [xen/tools/vtpm\\_manager/manager/vtpmpriv.h](#)

## 3.133 vtpm\_ipc\_handle\_t Struct Reference

```
#include <vtpm_ipc.h>
```

### Data Fields

- int [fh](#)
- int [flags](#)
- char \* [name](#)



### 3.133.1 Field Documentation

3.133.1.1 int fh

3.133.1.2 int flags

3.133.1.3 char\* name

The documentation for this struct was generated from the following file:

- xen/tools/vtpm\_manager/manager/[vtpm\\_ipc.h](#)

## 3.134 vtpm\_thread\_params\_s Struct Reference

### Data Fields

- [vtpm\\_ipc\\_handle\\_t](#) \* tx\_ipc\_h
- [vtpm\\_ipc\\_handle\\_t](#) \* rx\_ipc\_h
- [BOOL](#) fw\_tpm
- [vtpm\\_ipc\\_handle\\_t](#) \* fw\_tx\_ipc\_h
- [vtpm\\_ipc\\_handle\\_t](#) \* fw\_rx\_ipc\_h
- [BOOL](#) is\_priv
- char \* [thread\\_name](#)

### 3.134.1 Field Documentation

3.134.1.1 [vtpm\\_ipc\\_handle\\_t](#)\* fw\_rx\_ipc\_h

3.134.1.2 [BOOL](#) fw\_tpm

3.134.1.3 [vtpm\\_ipc\\_handle\\_t](#)\* fw\_tx\_ipc\_h

3.134.1.4 [BOOL](#) is\_priv

3.134.1.5 [vtpm\\_ipc\\_handle\\_t](#)\* rx\_ipc\_h

3.134.1.6 char\* thread\_name

3.134.1.7 [vtpm\\_ipc\\_handle\\_t](#)\* tx\_ipc\_h

The documentation for this struct was generated from the following file:

- xen/tools/vtpm\_manager/manager/[vtpmd.c](#)

## 3.135 xc\_core\_header Struct Reference

```
#include <xenctrl.h>
```

### Data Fields

- unsigned int [xch\\_magic](#)
- unsigned int [xch\\_nr\\_vcpus](#)

- unsigned int [xch\\_nr\\_pages](#)
- unsigned int [xch\\_ctxt\\_offset](#)
- unsigned int [xch\\_index\\_offset](#)
- unsigned int [xch\\_pages\\_offset](#)

### 3.135.1 Field Documentation

3.135.1.1 unsigned int [xch\\_ctxt\\_offset](#)

3.135.1.2 unsigned int [xch\\_index\\_offset](#)

3.135.1.3 unsigned int [xch\\_magic](#)

3.135.1.4 unsigned int [xch\\_nr\\_pages](#)

3.135.1.5 unsigned int [xch\\_nr\\_vcpus](#)

3.135.1.6 unsigned int [xch\\_pages\\_offset](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.136 xc\_cpupoolinfo Struct Reference

```
#include <xenctrl.h>
```

### Data Fields

- uint32\_t [cpupool\\_id](#)
- uint32\_t [sched\\_id](#)
- uint32\_t [n\\_dom](#)
- [xc\\_cpumap\\_t](#) [cpumap](#)

### 3.136.1 Field Documentation

3.136.1.1 [xc\\_cpumap\\_t](#) [cpumap](#)

3.136.1.2 uint32\_t [cpupool\\_id](#)

3.136.1.3 uint32\_t [n\\_dom](#)

3.136.1.4 uint32\_t [sched\\_id](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.137 xc\_cx\_stat Struct Reference

```
#include <xenctrl.h>
```

## Data Fields

- [uint32\\_t nr](#)
- [uint32\\_t last](#)
- [uint64\\_t idle\\_time](#)
- [uint64\\_t \\* triggers](#)
- [uint64\\_t \\* residencies](#)
- [uint64\\_t pc3](#)
- [uint64\\_t pc6](#)
- [uint64\\_t pc7](#)
- [uint64\\_t cc3](#)
- [uint64\\_t cc6](#)

### 3.137.1 Field Documentation

3.137.1.1 [uint64\\_t cc3](#)

3.137.1.2 [uint64\\_t cc6](#)

3.137.1.3 [uint64\\_t idle\\_time](#)

3.137.1.4 [uint32\\_t last](#)

3.137.1.5 [uint32\\_t nr](#)

3.137.1.6 [uint64\\_t pc3](#)

3.137.1.7 [uint64\\_t pc6](#)

3.137.1.8 [uint64\\_t pc7](#)

3.137.1.9 [uint64\\_t\\* residencies](#)

3.137.1.10 [uint64\\_t\\* triggers](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.138 xc\_dom\_arch Struct Reference

```
#include <xc_dom.h>
```

## Data Fields

- [int\(\\* alloc\\_magic\\_pages \)\(struct xc\\_dom\\_image \\*dom\)](#)
- [int\(\\* count\\_pgtables \)\(struct xc\\_dom\\_image \\*dom\)](#)
- [int\(\\* setup\\_pgtables \)\(struct xc\\_dom\\_image \\*dom\)](#)
- [int\(\\* start\\_info \)\(struct xc\\_dom\\_image \\*dom\)](#)
- [int\(\\* shared\\_info \)\(struct xc\\_dom\\_image \\*dom, void \\*shared\\_info\)](#)
- [int\(\\* vcpu \)\(struct xc\\_dom\\_image \\*dom, void \\*vcpu\\_ctxt\)](#)
- [char \\* guest\\_type](#)
- [char \\* native\\_protocol](#)

- int [page\\_shift](#)
- int [sizeof\\_pfn](#)
- struct [xc\\_dom\\_arch](#) \* [next](#)

### 3.138.1 Field Documentation

3.138.1.1 int(\* [alloc\\_magic\\_pages](#))(struct [xc\\_dom\\_image](#) \*dom)

3.138.1.2 int(\* [count\\_pgtables](#))(struct [xc\\_dom\\_image](#) \*dom)

3.138.1.3 char\* [guest\\_type](#)

3.138.1.4 char\* [native\\_protocol](#)

3.138.1.5 struct [xc\\_dom\\_arch](#)\* [next](#)

3.138.1.6 int [page\\_shift](#)

3.138.1.7 int(\* [setup\\_pgtables](#))(struct [xc\\_dom\\_image](#) \*dom)

3.138.1.8 int(\* [shared\\_info](#))(struct [xc\\_dom\\_image](#) \*dom, void \*[shared\\_info](#))

3.138.1.9 int [sizeof\\_pfn](#)

3.138.1.10 int(\* [start\\_info](#))(struct [xc\\_dom\\_image](#) \*dom)

3.138.1.11 int(\* [vcpu](#))(struct [xc\\_dom\\_image](#) \*dom, void \*[vcpu\\_ctxt](#))

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xc\\_dom.h](#)

## 3.139 xc\_dom\_image Struct Reference

```
#include <xc_dom.h>
```

### Data Fields

- void \* [kernel\\_blob](#)
- size\_t [kernel\\_size](#)
- void \* [ramdisk\\_blob](#)
- size\_t [ramdisk\\_size](#)
- char \* [cmdline](#)
- uint32\_t [f\\_requested](#) [XENFEAT\_NR\_SUBMAPS]
- struct [elf\\_dom\\_parms](#) [parms](#)
- char \* [guest\\_type](#)
- struct [xc\\_dom\\_seg](#) [kernel\\_seg](#)
- struct [xc\\_dom\\_seg](#) [ramdisk\\_seg](#)
- struct [xc\\_dom\\_seg](#) [p2m\\_seg](#)
- struct [xc\\_dom\\_seg](#) [pgtables\\_seg](#)
- struct [xc\\_dom\\_seg](#) [devicetree\\_seg](#)
- xen\_pfn\_t [start\\_info\\_pfn](#)
- xen\_pfn\_t [console\\_pfn](#)

- `xen_pfn_t` `xenstore_pfn`
- `xen_pfn_t` `shared_info_pfn`
- `xen_pfn_t` `bootstack_pfn`
- `xen_pfn_t` `caas_pfn`
- `xen_vaddr_t` `virt_alloc_end`
- `xen_vaddr_t` `bsd_syntab_start`
- unsigned int `pgtables`
- unsigned int `pg_l4`
- unsigned int `pg_l3`
- unsigned int `pg_l2`
- unsigned int `pg_l1`
- unsigned int `alloc_bootstack`
- unsigned int `extra_pages`
- `xen_vaddr_t` `virt_pgtab_end`
- `uint32_t` `f_active` [XENFEAT\_NR\_SUBMAPS]
- `xen_pfn_t` \* `p2m_host`
- void \* `p2m_guest`
- `xen_pfn_t` `total_pages`
- struct `xc_dom_phys` \* `phys_pages`
- int `realmodearea_log`
- struct `xc_dom_mem` \* `memblocks`
- `size_t` `alloc_malloc`
- `size_t` `alloc_mem_map`
- `size_t` `alloc_file_map`
- `size_t` `alloc_domU_map`
- unsigned long `flags`
- unsigned int `console_evtchn`
- unsigned int `xenstore_evtchn`
- `xen_pfn_t` `shared_info_mfn`
- `xc_interface` \* `xch`
- `domid_t` `guest_domid`
- `int8_t` `vhpt_size_log2`
- `int8_t` `superpages`
- int `shadow_enabled`
- int `xen_version`
- `xen_capabilities_info_t` `xen_caps`
- struct `xc_dom_loader` \* `kernel_loader`
- void \* `private_loader`
- struct `xc_dom_arch` \* `arch_hooks`
- int(\* `allocate`)(struct `xc_dom_image` \*`dom`, `xen_vaddr_t` `up_to`)

### 3.139.1 Field Documentation

3.139.1.1 unsigned int `alloc_bootstack`

3.139.1.2 `size_t` `alloc_domU_map`

3.139.1.3 `size_t` `alloc_file_map`

3.139.1.4 `size_t` `alloc_malloc`

3.139.1.5 `size_t` `alloc_mem_map`

3.139.1.6 int(\* `allocate`)(struct `xc_dom_image` \*`dom`, `xen_vaddr_t` `up_to`)

- 3.139.1.7 struct xc\_dom\_arch\* arch\_hooks
- 3.139.1.8 xen\_pfn\_t bootstack\_pfn
- 3.139.1.9 xen\_vaddr\_t bsd\_symtab\_start
- 3.139.1.10 xen\_pfn\_t caas\_pfn
- 3.139.1.11 char\* cmdline
- 3.139.1.12 unsigned int console\_evtchn
- 3.139.1.13 xen\_pfn\_t console\_pfn
- 3.139.1.14 struct xc\_dom\_seg devicetree\_seg
- 3.139.1.15 unsigned int extra\_pages
- 3.139.1.16 uint32\_t f\_active[XENFEAT\_NR\_SUBMAPS]
- 3.139.1.17 uint32\_t f\_requested[XENFEAT\_NR\_SUBMAPS]
- 3.139.1.18 unsigned long flags
- 3.139.1.19 domid\_t guest\_domid
- 3.139.1.20 char\* guest\_type
- 3.139.1.21 void\* kernel\_blob
- 3.139.1.22 struct xc\_dom\_loader\* kernel\_loader
- 3.139.1.23 struct xc\_dom\_seg kernel\_seg
- 3.139.1.24 size\_t kernel\_size
- 3.139.1.25 struct xc\_dom\_mem\* memblocks
- 3.139.1.26 void\* p2m\_guest
- 3.139.1.27 xen\_pfn\_t\* p2m\_host
- 3.139.1.28 struct xc\_dom\_seg p2m\_seg
- 3.139.1.29 struct elf\_dom\_parms parms
- 3.139.1.30 unsigned int pg\_l1
- 3.139.1.31 unsigned int pg\_l2
- 3.139.1.32 unsigned int pg\_l3
- 3.139.1.33 unsigned int pg\_l4
- 3.139.1.34 unsigned int pgtables

- 3.139.1.35 struct xc\_dom\_seg pgtables\_seg
- 3.139.1.36 struct xc\_dom\_phys\* phys\_pages
- 3.139.1.37 void\* private\_loader
- 3.139.1.38 void\* ramdisk\_blob
- 3.139.1.39 struct xc\_dom\_seg ramdisk\_seg
- 3.139.1.40 size\_t ramdisk\_size
- 3.139.1.41 int realmodearea\_log
- 3.139.1.42 int shadow\_enabled
- 3.139.1.43 xen\_pfn\_t shared\_info\_mfn
- 3.139.1.44 xen\_pfn\_t shared\_info\_pfn
- 3.139.1.45 xen\_pfn\_t start\_info\_pfn
- 3.139.1.46 int8\_t superpages
- 3.139.1.47 xen\_pfn\_t total\_pages
- 3.139.1.48 int8\_t vhpt\_size\_log2
- 3.139.1.49 xen\_vaddr\_t virt\_alloc\_end
- 3.139.1.50 xen\_vaddr\_t virt\_pgtab\_end
- 3.139.1.51 xc\_interface\* xch
- 3.139.1.52 xen\_capabilities\_info\_t xen\_caps
- 3.139.1.53 int xen\_version
- 3.139.1.54 unsigned int xenstore\_evtchn
- 3.139.1.55 xen\_pfn\_t xenstore\_pfn

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xc\\_dom.h](#)

## 3.140 xc\_dom\_loader Struct Reference

```
#include <xc_dom.h>
```

### Data Fields

- char \* [name](#)
- int(\* [probe](#) )(struct [xc\\_dom\\_image](#) \*dom)

- `int(* parser)(struct xc_dom_image *dom)`
- `int(* loader)(struct xc_dom_image *dom)`
- `struct xc_dom_loader * next`

### 3.140.1 Field Documentation

3.140.1.1 `int(* loader)(struct xc_dom_image *dom)`

3.140.1.2 `char* name`

3.140.1.3 `struct xc_dom_loader* next`

3.140.1.4 `int(* parser)(struct xc_dom_image *dom)`

3.140.1.5 `int(* probe)(struct xc_dom_image *dom)`

The documentation for this struct was generated from the following file:

- `xen/tools/libxc/xc_dom.h`

## 3.141 xc\_dom\_mem Struct Reference

```
#include <xc_dom.h>
```

### Data Fields

- `struct xc_dom_mem * next`
- `void * mmap_ptr`
- `size_t mmap_len`
- `unsigned char memory [0]`

### 3.141.1 Field Documentation

3.141.1.1 `unsigned char memory[0]`

3.141.1.2 `size_t mmap_len`

3.141.1.3 `void* mmap_ptr`

3.141.1.4 `struct xc_dom_mem* next`

The documentation for this struct was generated from the following file:

- `xen/tools/libxc/xc_dom.h`

## 3.142 xc\_dom\_phys Struct Reference

```
#include <xc_dom.h>
```



## Data Fields

- struct [xc\\_dom\\_phys](#) \* [next](#)
- void \* [ptr](#)
- xen\_pfn\_t [first](#)
- xen\_pfn\_t [count](#)

### 3.142.1 Field Documentation

3.142.1.1 [xen\\_pfn\\_t](#) [count](#)

3.142.1.2 [xen\\_pfn\\_t](#) [first](#)

3.142.1.3 [struct xc\\_dom\\_phys](#)\* [next](#)

3.142.1.4 [void](#)\* [ptr](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xc\\_dom.h](#)

## 3.143 xc\_dom\_seg Struct Reference

```
#include <xc_dom.h>
```

## Data Fields

- [xen\\_vaddr\\_t](#) [vstart](#)
- [xen\\_vaddr\\_t](#) [vend](#)
- [xen\\_pfn\\_t](#) [pfn](#)

### 3.143.1 Field Documentation

3.143.1.1 [xen\\_pfn\\_t](#) [pfn](#)

3.143.1.2 [xen\\_vaddr\\_t](#) [vend](#)

3.143.1.3 [xen\\_vaddr\\_t](#) [vstart](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xc\\_dom.h](#)

## 3.144 xc\_dominfo Struct Reference

```
#include <xenctrl.h>
```

## Data Fields

- uint32\_t [domid](#)
- uint32\_t [ssidref](#)
- unsigned int [dying](#):1
- unsigned int [crashed](#):1
- unsigned int [shutdown](#):1
- unsigned int [paused](#):1
- unsigned int [blocked](#):1
- unsigned int [running](#):1
- unsigned int [hvm](#):1
- unsigned int [debugged](#):1
- unsigned int [shutdown\\_reason](#)
- unsigned long [nr\\_pages](#)
- unsigned long [nr\\_shared\\_pages](#)
- unsigned long [shared\\_info\\_frame](#)
- uint64\_t [cpu\\_time](#)
- unsigned long [max\\_memkb](#)
- unsigned int [nr\\_online\\_vcpus](#)
- unsigned int [max\\_vcpu\\_id](#)
- [xen\\_domain\\_handle\\_t](#) [handle](#)
- unsigned int [cpupool](#)

### 3.144.1 Field Documentation

3.144.1.1 unsigned int [blocked](#)

3.144.1.2 uint64\_t [cpu\\_time](#)

3.144.1.3 unsigned int [cpupool](#)

3.144.1.4 unsigned int [crashed](#)

3.144.1.5 unsigned int [debugged](#)

3.144.1.6 uint32\_t [domid](#)

3.144.1.7 unsigned int [dying](#)

3.144.1.8 [xen\\_domain\\_handle\\_t](#) [handle](#)

3.144.1.9 unsigned int [hvm](#)

3.144.1.10 unsigned long [max\\_memkb](#)

3.144.1.11 unsigned int [max\\_vcpu\\_id](#)

3.144.1.12 unsigned int [nr\\_online\\_vcpus](#)

3.144.1.13 unsigned long [nr\\_pages](#)

3.144.1.14 unsigned long [nr\\_shared\\_pages](#)

3.144.1.15 unsigned int [paused](#)

3.144.1.16 unsigned int running

3.144.1.17 unsigned long shared\_info\_frame

3.144.1.18 unsigned int shutdown

3.144.1.19 unsigned int shutdown\_reason

3.144.1.20 uint32\_t ssidref

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.145 xc\_error Struct Reference

```
#include <xenctrl.h>
```

### Data Fields

- enum [xc\\_error\\_code](#) code
- char [message](#)[XC\_MAX\_ERROR\_MSG\_LEN]

#### 3.145.1 Field Documentation

3.145.1.1 enum [xc\\_error\\_code](#) code

3.145.1.2 char [message](#)[XC\_MAX\_ERROR\_MSG\_LEN]

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.146 xc\_get\_cpufreq\_para Struct Reference

```
#include <xenctrl.h>
```

### Data Fields

- uint32\_t [cpu\\_num](#)
- uint32\_t [freq\\_num](#)
- uint32\_t [gov\\_num](#)
- uint32\_t \* [affected\\_cpus](#)
- uint32\_t \* [scaling\\_available\\_frequencies](#)
- char \* [scaling\\_available\\_governors](#)
- char [scaling\\_driver](#)[CPUFREQ\_NAME\_LEN]
- uint32\_t [cpuinfo\\_cur\\_freq](#)
- uint32\_t [cpuinfo\\_max\\_freq](#)
- uint32\_t [cpuinfo\\_min\\_freq](#)
- uint32\_t [scaling\\_cur\\_freq](#)
- char [scaling\\_governor](#)[CPUFREQ\_NAME\_LEN]

- uint32\_t [scaling\\_max\\_freq](#)
- uint32\_t [scaling\\_min\\_freq](#)
- union {
  - [xc\\_userspace\\_t](#) userspace
  - [xc\\_ondemand\\_t](#) ondemand
 } u
- int32\_t [turbo\\_enabled](#)

### 3.146.1 Field Documentation

- 3.146.1.1 uint32\_t\* [affected\\_cpus](#)
- 3.146.1.2 uint32\_t [cpu\\_num](#)
- 3.146.1.3 uint32\_t [cpuinfo\\_cur\\_freq](#)
- 3.146.1.4 uint32\_t [cpuinfo\\_max\\_freq](#)
- 3.146.1.5 uint32\_t [cpuinfo\\_min\\_freq](#)
- 3.146.1.6 uint32\_t [freq\\_num](#)
- 3.146.1.7 uint32\_t [gov\\_num](#)
- 3.146.1.8 [xc\\_ondemand\\_t](#) ondemand
- 3.146.1.9 uint32\_t\* [scaling\\_available\\_frequencies](#)
- 3.146.1.10 char\* [scaling\\_available\\_governors](#)
- 3.146.1.11 uint32\_t [scaling\\_cur\\_freq](#)
- 3.146.1.12 char [scaling\\_driver](#)[CPUFREQ\_NAME\_LEN]
- 3.146.1.13 char [scaling\\_governor](#)[CPUFREQ\_NAME\_LEN]
- 3.146.1.14 uint32\_t [scaling\\_max\\_freq](#)
- 3.146.1.15 uint32\_t [scaling\\_min\\_freq](#)
- 3.146.1.16 int32\_t [turbo\\_enabled](#)
- 3.146.1.17 union { ... } u
- 3.146.1.18 [xc\\_userspace\\_t](#) userspace

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.147 xc\_hypercall\_buffer Struct Reference

```
#include <xenctrl.h>
```

## Data Fields

- void \* [hbuf](#)
- struct [xc\\_hypcall\\_buffer](#) \* [param\\_shadow](#)
- int [dir](#)
- void \* [ubuf](#)
- size\_t [sz](#)

### 3.147.1 Field Documentation

3.147.1.1 int [dir](#)

3.147.1.2 void\* [hbuf](#)

3.147.1.3 struct [xc\\_hypcall\\_buffer](#)\* [param\\_shadow](#)

3.147.1.4 size\_t [sz](#)

3.147.1.5 void\* [ubuf](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

## 3.148 xc\_px\_stat Struct Reference

```
#include <xenctrl.h>
```

## Data Fields

- uint8\_t [total](#)
- uint8\_t [usable](#)
- uint8\_t [last](#)
- uint8\_t [cur](#)
- uint64\_t \* [trans\\_pt](#)
- struct [xc\\_px\\_val](#) \* [pt](#)

### 3.148.1 Field Documentation

3.148.1.1 uint8\_t [cur](#)

3.148.1.2 uint8\_t [last](#)

3.148.1.3 struct [xc\\_px\\_val](#)\* [pt](#)

3.148.1.4 uint8\_t [total](#)

3.148.1.5 uint64\_t\* [trans\\_pt](#)

3.148.1.6 uint8\_t [usable](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

### 3.149 xc\_px\_val Struct Reference

```
#include <xenctrl.h>
```

#### Data Fields

- uint64\_t [freq](#)
- uint64\_t [residency](#)
- uint64\_t [count](#)

#### 3.149.1 Field Documentation

3.149.1.1 [uint64\\_t count](#)

3.149.1.2 [uint64\\_t freq](#)

3.149.1.3 [uint64\\_t residency](#)

The documentation for this struct was generated from the following file:

- [xen/tools/libxc/xenctrl.h](#)

### 3.150 xen\_domctl Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- uint32\_t [cmd](#)
- uint32\_t [interface\\_version](#)
- [domid\\_t](#) [domain](#)
- union {
  - struct [xen\\_domctl\\_createdomain](#) [createdomain](#)
  - struct [xen\\_domctl\\_getdomaininfo](#) [getdomaininfo](#)
  - struct [xen\\_domctl\\_getmemlist](#) [getmemlist](#)
  - struct [xen\\_domctl\\_getpageframeinfo](#) [getpageframeinfo](#)
  - struct [xen\\_domctl\\_getpageframeinfo2](#) [getpageframeinfo2](#)
  - struct [xen\\_domctl\\_getpageframeinfo3](#) [getpageframeinfo3](#)
  - struct [xen\\_domctl\\_vcpuaffinity](#) [vcpuaffinity](#)
  - struct [xen\\_domctl\\_shadow\\_op](#) [shadow\\_op](#)
  - struct [xen\\_domctl\\_max\\_mem](#) [max\\_mem](#)
  - struct [xen\\_domctl\\_vcpucontext](#) [vcpucontext](#)
  - struct [xen\\_domctl\\_getvcpuinfo](#) [getvcpuinfo](#)
  - struct [xen\\_domctl\\_max\\_vcpus](#) [max\\_vcpus](#)
  - struct [xen\\_domctl\\_scheduler\\_op](#) [scheduler\\_op](#)
  - struct [xen\\_domctl\\_setdomainhandle](#) [setdomainhandle](#)
  - struct [xen\\_domctl\\_setdebugging](#) [setdebugging](#)
  - struct [xen\\_domctl\\_irq\\_permission](#) [irq\\_permission](#)
  - struct [xen\\_domctl\\_iomem\\_permission](#) [iomem\\_permission](#)
  - struct [xen\\_domctl\\_ioport\\_permission](#) [ioport\\_permission](#)
  - struct [xen\\_domctl\\_hypercall\\_init](#) [hypercall\\_init](#)
  - struct [xen\\_domctl\\_arch\\_setup](#) [arch\\_setup](#)
  - struct [xen\\_domctl\\_settimeoffset](#) [settimeoffset](#)

```

struct xen_domctl_disable_migrate disable_migrate
struct xen_domctl_tsc_info tsc_info
struct xen_domctl_real_mode_area real_mode_area
struct xen_domctl_hvmcontext hvmcontext
struct
xen_domctl_hvmcontext_partial hvmcontext_partial
struct xen_domctl_address_size address_size
struct xen_domctl_sendtrigger sendtrigger
struct xen_domctl_get_device_group get_device_group
struct xen_domctl_assign_device assign_device
struct xen_domctl_bind_pt_irq bind_pt_irq
struct xen_domctl_memory_mapping memory_mapping
struct xen_domctl_ioport_mapping ioport_mapping
struct xen_domctl_pin_mem_cacheattr pin_mem_cacheattr
struct xen_domctl_ext_vcpucontext ext_vcpucontext
struct xen_domctl_set_opt_feature set_opt_feature
struct xen_domctl_set_target set_target
struct xen_domctl_subscribe subscribe
struct xen_domctl_debug_op debug_op
struct xen_domctl_mem_event_op mem_event_op
struct xen_domctl_mem_sharing_op mem_sharing_op
struct
xen_domctl_set_access_required access_required
struct xen_domctl_gdbsx_memio gdbsx_guest_memio
struct
xen_domctl_gdbsx_pauseunp_vcpu gdbsx_pauseunp_vcpu
struct xen_domctl_gdbsx_domstatus gdbsx_domstatus
uint8_t pad [128]
} u

```

### 3.150.1 Field Documentation

- 3.150.1.1 struct xen\_domctl\_set\_access\_required access\_required
- 3.150.1.2 struct xen\_domctl\_address\_size address\_size
- 3.150.1.3 struct xen\_domctl\_arch\_setup arch\_setup
- 3.150.1.4 struct xen\_domctl\_assign\_device assign\_device
- 3.150.1.5 struct xen\_domctl\_bind\_pt\_irq bind\_pt\_irq
- 3.150.1.6 uint32\_t cmd
- 3.150.1.7 struct xen\_domctl\_createdomain createdomain
- 3.150.1.8 struct xen\_domctl\_debug\_op debug\_op
- 3.150.1.9 struct xen\_domctl\_disable\_migrate disable\_migrate
- 3.150.1.10 domid\_t domain
- 3.150.1.11 struct xen\_domctl\_ext\_vcpucontext ext\_vcpucontext
- 3.150.1.12 struct xen\_domctl\_gdbsx\_domstatus gdbsx\_domstatus

- 3.150.1.13 struct xen\_domctl\_gdbstx\_memio gdbstx\_guest\_memio
- 3.150.1.14 struct xen\_domctl\_gdbstx\_pauseunp\_vcpu gdbstx\_pauseunp\_vcpu
- 3.150.1.15 struct xen\_domctl\_get\_device\_group get\_device\_group
- 3.150.1.16 struct xen\_domctl\_getdomaininfo getdomaininfo
- 3.150.1.17 struct xen\_domctl\_getmemlist getmemlist
- 3.150.1.18 struct xen\_domctl\_getpageframeinfo getpageframeinfo
- 3.150.1.19 struct xen\_domctl\_getpageframeinfo2 getpageframeinfo2
- 3.150.1.20 struct xen\_domctl\_getpageframeinfo3 getpageframeinfo3
- 3.150.1.21 struct xen\_domctl\_getvcpuinfo getvcpuinfo
- 3.150.1.22 struct xen\_domctl\_hvmcontext hvmcontext
- 3.150.1.23 struct xen\_domctl\_hvmcontext\_partial hvmcontext\_partial
- 3.150.1.24 struct xen\_domctl\_hypercall\_init hypercall\_init
- 3.150.1.25 uint32\_t interface\_version
- 3.150.1.26 struct xen\_domctl\_iomem\_permission iomem\_permission
- 3.150.1.27 struct xen\_domctl\_ioport\_mapping ioport\_mapping
- 3.150.1.28 struct xen\_domctl\_ioport\_permission ioport\_permission
- 3.150.1.29 struct xen\_domctl\_irq\_permission irq\_permission
- 3.150.1.30 struct xen\_domctl\_max\_mem max\_mem
- 3.150.1.31 struct xen\_domctl\_max\_vcpus max\_vcpus
- 3.150.1.32 struct xen\_domctl\_mem\_event\_op mem\_event\_op
- 3.150.1.33 struct xen\_domctl\_mem\_sharing\_op mem\_sharing\_op
- 3.150.1.34 struct xen\_domctl\_memory\_mapping memory\_mapping
- 3.150.1.35 uint8\_t pad[128]
- 3.150.1.36 struct xen\_domctl\_pin\_mem\_cacheattr pin\_mem\_cacheattr
- 3.150.1.37 struct xen\_domctl\_real\_mode\_area real\_mode\_area
- 3.150.1.38 struct xen\_domctl\_scheduler\_op scheduler\_op
- 3.150.1.39 struct xen\_domctl\_sendtrigger sendtrigger
- 3.150.1.40 struct xen\_domctl\_set\_opt\_feature set\_opt\_feature



- 3.150.1.41 struct xen\_domctl\_set\_target set\_target
- 3.150.1.42 struct xen\_domctl\_setdebugging setdebugging
- 3.150.1.43 struct xen\_domctl\_setdomainhandle setdomainhandle
- 3.150.1.44 struct xen\_domctl\_settimeoffset settimeoffset
- 3.150.1.45 struct xen\_domctl\_shadow\_op shadow\_op
- 3.150.1.46 struct xen\_domctl\_subscribe subscribe
- 3.150.1.47 struct xen\_domctl\_tsc\_info tsc\_info
- 3.150.1.48 union { ... } u
- 3.150.1.49 struct xen\_domctl\_vcpuaffinity vcpuaffinity
- 3.150.1.50 struct xen\_domctl\_vcpucontext vcpucontext

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.151 xen\_domctl\_address\_size Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [size](#)

### 3.151.1 Field Documentation

#### 3.151.1.1 uint32\_t size

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.152 xen\_domctl\_arch\_setup Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint64\_aligned\_t [flags](#)

### 3.152.1 Field Documentation

#### 3.152.1.1 uint64\_aligned\_t flags

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.153 xen\_domctl\_assign\_device Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [machine\\_bdf](#)

### 3.153.1 Field Documentation

#### 3.153.1.1 uint32\_t machine\_bdf

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.154 xen\_domctl\_bind\_pt\_irq Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [machine\\_irq](#)
- [pt\\_irq\\_type\\_t](#) irq\_type
- uint32\_t [hvm\\_domid](#)
- union {
  - struct {
    - uint8\_t [isa\\_irq](#)
  - [isa](#)
  - struct {
    - uint8\_t [bus](#)
    - uint8\_t [device](#)
    - uint8\_t [intx](#)
  - [pci](#)
  - struct {
    - uint8\_t [gvec](#)
    - uint32\_t [gflags](#)
    - uint64\_aligned\_t [gtable](#)
  - [msi](#)
- [u](#)

### 3.154.1 Field Documentation

3.154.1.1 `uint8_t` bus

3.154.1.2 `uint8_t` device

3.154.1.3 `uint32_t` gflags

3.154.1.4 `uint64_aligned_t` gtable

3.154.1.5 `uint8_t` gvec

3.154.1.6 `uint32_t` hvm\_domid

3.154.1.7 `uint8_t` intx

3.154.1.8 `pt_irq_type_t` irq\_type

3.154.1.9 `struct { ... }` isa

3.154.1.10 `uint8_t` isa\_irq

3.154.1.11 `uint32_t` machine\_irq

3.154.1.12 `struct { ... }` msi

3.154.1.13 `struct { ... }` pci

3.154.1.14 `union { ... }` u

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.155 xen\_domctl\_createdomain Struct Reference

```
#include <domctl.h>
```

### Data Fields

- `uint32_t` ssidref
- `xen_domain_handle_t` handle
- `uint32_t` flags

### 3.155.1 Field Documentation

3.155.1.1 `uint32_t` flags

3.155.1.2 `xen_domain_handle_t` handle

3.155.1.3 `uint32_t` ssidref

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.156 xen\_domctl\_debug\_op Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- [uint32\\_t op](#)
- [uint32\\_t vcpu](#)

#### 3.156.1 Field Documentation

3.156.1.1 [uint32\\_t op](#)

3.156.1.2 [uint32\\_t vcpu](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.157 xen\_domctl\_disable\_migrate Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- [uint32\\_t disable](#)

#### 3.157.1 Field Documentation

3.157.1.1 [uint32\\_t disable](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.158 xen\_domctl\_ext\_vcpucontext Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- [uint32\\_t vcpu](#)
- [uint32\\_t size](#)

### 3.158.1 Field Documentation

3.158.1.1 uint32\_t size

3.158.1.2 uint32\_t vcpu

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.159 xen\_domctl\_gdbsx\_domstatus Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint8\_t [paused](#)
- uint32\_t [vcpu\\_id](#)
- uint32\_t [vcpu\\_ev](#)

### 3.159.1 Field Documentation

3.159.1.1 uint8\_t paused

3.159.1.2 uint32\_t vcpu\_ev

3.159.1.3 uint32\_t vcpu\_id

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.160 xen\_domctl\_gdbsx\_memio Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint64\_aligned\_t [pgd3val](#)
- uint64\_aligned\_t [gva](#)
- uint64\_aligned\_t [uva](#)
- uint32\_t [len](#)
- uint8\_t [gwr](#)
- uint32\_t [remain](#)

### 3.160.1 Field Documentation

3.160.1.1 uint64\_aligned\_t gva

3.160.1.2 uint8\_t gwr

3.160.1.3 `uint32_t len`

3.160.1.4 `uint64_aligned_t pgd3val`

3.160.1.5 `uint32_t remain`

3.160.1.6 `uint64_aligned_t uva`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.161 `xen_domctl_gdbsx_pauseunp_vcpu` Struct Reference

```
#include <domctl.h>
```

### Data Fields

- `uint32_t vcpu`

### 3.161.1 Field Documentation

3.161.1.1 `uint32_t vcpu`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.162 `xen_domctl_get_device_group` Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- `XEN_GUEST_HANDLE_64` (`uint32_t`) `sdev_array`

### Data Fields

- `uint32_t machine_bdf`
- `uint32_t max_sdevs`
- `uint32_t num_sdevs`

### 3.162.1 Member Function Documentation

3.162.1.1 `XEN_GUEST_HANDLE_64` ( `uint32_t` )

### 3.162.2 Field Documentation

3.162.2.1 `uint32_t machine_bdf`

3.162.2.2 uint32\_t max\_sdevs

3.162.2.3 uint32\_t num\_sdevs

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.163 xen\_domctl\_getdomaininfo Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [domid\\_t](#) domain
- [uint32\\_t](#) flags
- [uint64\\_aligned\\_t](#) tot\_pages
- [uint64\\_aligned\\_t](#) max\_pages
- [uint64\\_aligned\\_t](#) shr\_pages
- [uint64\\_aligned\\_t](#) shared\_info\_frame
- [uint64\\_aligned\\_t](#) cpu\_time
- [uint32\\_t](#) nr\_online\_vcpus
- [uint32\\_t](#) max\_vcpu\_id
- [uint32\\_t](#) ssidref
- [xen\\_domain\\_handle\\_t](#) handle
- [uint32\\_t](#) cpupool

### 3.163.1 Field Documentation

3.163.1.1 [uint64\\_aligned\\_t](#) cpu\_time

3.163.1.2 [uint32\\_t](#) cpupool

3.163.1.3 [domid\\_t](#) domain

3.163.1.4 [uint32\\_t](#) flags

3.163.1.5 [xen\\_domain\\_handle\\_t](#) handle

3.163.1.6 [uint64\\_aligned\\_t](#) max\_pages

3.163.1.7 [uint32\\_t](#) max\_vcpu\_id

3.163.1.8 [uint32\\_t](#) nr\_online\_vcpus

3.163.1.9 [uint64\\_aligned\\_t](#) shared\_info\_frame

3.163.1.10 [uint64\\_aligned\\_t](#) shr\_pages

3.163.1.11 [uint32\\_t](#) ssidref

3.163.1.12 [uint64\\_aligned\\_t](#) tot\_pages

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.164 xen\_domctl\_getmemlist Struct Reference

```
#include <domctl.h>
```

#### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (uint64) [buffer](#)

#### Data Fields

- uint64\_aligned\_t [max\\_pfns](#)
- uint64\_aligned\_t [start\\_pfn](#)
- uint64\_aligned\_t [num\\_pfns](#)

#### 3.164.1 Member Function Documentation

3.164.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( uint64 )

#### 3.164.2 Field Documentation

3.164.2.1 [uint64\\_aligned\\_t](#) [max\\_pfns](#)

3.164.2.2 [uint64\\_aligned\\_t](#) [num\\_pfns](#)

3.164.2.3 [uint64\\_aligned\\_t](#) [start\\_pfn](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.165 xen\_domctl\_getpageframeinfo Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- uint64\_aligned\_t [gmfn](#)
- uint32\_t [type](#)

#### 3.165.1 Field Documentation

3.165.1.1 [uint64\\_aligned\\_t](#) [gmfn](#)

3.165.1.2 [uint32\\_t](#) [type](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)



## 3.166 xen\_domctl\_getpageframeinfo2 Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (uint32) array

### Data Fields

- uint64\_aligned\_t [num](#)

#### 3.166.1 Member Function Documentation

3.166.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( uint32 )

#### 3.166.2 Field Documentation

3.166.2.1 uint64\_aligned\_t [num](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.167 xen\_domctl\_getpageframeinfo3 Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (xen\_pfn\_t) array

### Data Fields

- uint64\_aligned\_t [num](#)

#### 3.167.1 Member Function Documentation

3.167.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( xen\_pfn\_t )

#### 3.167.2 Field Documentation

3.167.2.1 uint64\_aligned\_t [num](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.168 xen\_domctl\_getvcpuinfo Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [vcpu](#)
- uint8\_t [online](#)
- uint8\_t [blocked](#)
- uint8\_t [running](#)
- uint64\_aligned\_t [cpu\\_time](#)
- uint32\_t [cpu](#)

### 3.168.1 Field Documentation

3.168.1.1 uint8\_t [blocked](#)

3.168.1.2 uint32\_t [cpu](#)

3.168.1.3 uint64\_aligned\_t [cpu\\_time](#)

3.168.1.4 uint8\_t [online](#)

3.168.1.5 uint8\_t [running](#)

3.168.1.6 uint32\_t [vcpu](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.169 xen\_domctl\_hvmcontext Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (uint8) [buffer](#)

### Data Fields

- uint32\_t [size](#)

### 3.169.1 Member Function Documentation

3.169.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( uint8 )

### 3.169.2 Field Documentation

3.169.2.1 uint32\_t [size](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.170 xen\_domctl\_hvmcontext\_partial Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (uint8) [buffer](#)

### Data Fields

- uint32\_t [type](#)
- uint32\_t [instance](#)

### 3.170.1 Member Function Documentation

#### 3.170.1.1 XEN\_GUEST\_HANDLE\_64 ( uint8 )

### 3.170.2 Field Documentation

#### 3.170.2.1 uint32\_t instance

#### 3.170.2.2 uint32\_t type

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.171 xen\_domctl\_hypercall\_init Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint64\_aligned\_t [gmfn](#)

### 3.171.1 Field Documentation

#### 3.171.1.1 uint64\_aligned\_t gmfn

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.172 xen\_domctl\_iomem\_permission Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint64\_aligned\_t [first\\_mfn](#)
- uint64\_aligned\_t [nr\\_mfns](#)
- uint8\_t [allow\\_access](#)

### 3.172.1 Field Documentation

3.172.1.1 uint8\_t [allow\\_access](#)

3.172.1.2 uint64\_aligned\_t [first\\_mfn](#)

3.172.1.3 uint64\_aligned\_t [nr\\_mfns](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.173 xen\_domctl\_ioport\_mapping Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint32\_t [first\\_gport](#)
- uint32\_t [first\\_mport](#)
- uint32\_t [nr\\_ports](#)
- uint32\_t [add\\_mapping](#)

### 3.173.1 Field Documentation

3.173.1.1 uint32\_t [add\\_mapping](#)

3.173.1.2 uint32\_t [first\\_gport](#)

3.173.1.3 uint32\_t [first\\_mport](#)

3.173.1.4 uint32\_t [nr\\_ports](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.174 xen\_domctl\_ioport\_permission Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint32\_t [first\\_port](#)
- uint32\_t [nr\\_ports](#)
- uint8\_t [allow\\_access](#)

### 3.174.1 Field Documentation

3.174.1.1 `uint8_t allow_access`

3.174.1.2 `uint32_t first_port`

3.174.1.3 `uint32_t nr_ports`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.175 xen\_domctl\_irq\_permission Struct Reference

```
#include <domctl.h>
```

### Data Fields

- `uint8_t pirq`
- `uint8_t allow_access`

### 3.175.1 Field Documentation

3.175.1.1 `uint8_t allow_access`

3.175.1.2 `uint8_t pirq`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.176 xen\_domctl\_max\_mem Struct Reference

```
#include <domctl.h>
```

### Data Fields

- `uint64_aligned_t max_memkb`

### 3.176.1 Field Documentation

3.176.1.1 `uint64_aligned_t max_memkb`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.177 xen\_domctl\_max\_vcpus Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint32\_t [max](#)

### 3.177.1 Field Documentation

#### 3.177.1.1 uint32\_t max

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.178 xen\_domctl\_mem\_event\_op Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint32\_t [op](#)
- uint32\_t [mode](#)
- uint64\_aligned\_t [shared\\_addr](#)
- uint64\_aligned\_t [ring\\_addr](#)
- uint64\_aligned\_t [gfn](#)

### 3.178.1 Field Documentation

#### 3.178.1.1 uint64\_aligned\_t gfn

#### 3.178.1.2 uint32\_t mode

#### 3.178.1.3 uint32\_t op

#### 3.178.1.4 uint64\_aligned\_t ring\_addr

#### 3.178.1.5 uint64\_aligned\_t shared\_addr

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.179 xen\_domctl\_mem\_sharing\_op Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint8\_t [op](#)

```

• union {
    uint8_t enable
    struct mem_sharing_op_nominate {
        union {
            uint64_aligned_t gfn
            uint32_t grant_ref
        } u
        uint64_aligned_t handle
    } nominate
    struct mem_sharing_op_share {
        uint64_aligned_t source_handle
        uint64_aligned_t client_handle
    } share
    struct mem_sharing_op_debug {
        union {
            uint64_aligned_t gfn
            uint64_aligned_t mfn
            grant_ref_t gref
        } u
    } debug
} u

```

### 3.179.1 Field Documentation

- 3.179.1.1 uint64\_aligned\_t client\_handle
- 3.179.1.2 struct { ... } ::mem\_sharing\_op\_debug debug
- 3.179.1.3 uint8\_t enable
- 3.179.1.4 uint64\_aligned\_t gfn
- 3.179.1.5 uint32\_t grant\_ref
- 3.179.1.6 grant\_ref\_t gref
- 3.179.1.7 uint64\_aligned\_t handle
- 3.179.1.8 uint64\_aligned\_t mfn
- 3.179.1.9 struct { ... } ::mem\_sharing\_op\_nominate nominate
- 3.179.1.10 uint8\_t op
- 3.179.1.11 struct { ... } ::mem\_sharing\_op\_share share
- 3.179.1.12 uint64\_aligned\_t source\_handle
- 3.179.1.13 union { ... } u
- 3.179.1.14 union { ... } u
- 3.179.1.15 union { ... } u

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.180 xen\_domctl\_memory\_mapping Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- uint64\_aligned\_t [first\\_gfn](#)
- uint64\_aligned\_t [first\\_mfn](#)
- uint64\_aligned\_t [nr\\_mfns](#)
- uint32\_t [add\\_mapping](#)
- uint32\_t [padding](#)

#### 3.180.1 Field Documentation

3.180.1.1 uint32\_t [add\\_mapping](#)

3.180.1.2 uint64\_aligned\_t [first\\_gfn](#)

3.180.1.3 uint64\_aligned\_t [first\\_mfn](#)

3.180.1.4 uint64\_aligned\_t [nr\\_mfns](#)

3.180.1.5 uint32\_t [padding](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

### 3.181 xen\_domctl\_pin\_mem\_cacheattr Struct Reference

```
#include <domctl.h>
```

#### Data Fields

- uint64\_aligned\_t [start](#)
- uint64\_aligned\_t [end](#)
- uint32\_t [type](#)

#### 3.181.1 Field Documentation

3.181.1.1 uint64\_aligned\_t [end](#)

3.181.1.2 uint64\_aligned\_t [start](#)

3.181.1.3 uint32\_t [type](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)



## 3.182 xen\_domctl\_real\_mode\_area Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [log](#)

### 3.182.1 Field Documentation

#### 3.182.1.1 uint32\_t log

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.183 xen\_domctl\_scheduler\_op Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [sched\\_id](#)
- uint32\_t [cmd](#)
- union {
  - struct [xen\\_domctl\\_sched\\_sedf](#) {
    - uint64\_aligned\_t [period](#)
    - uint64\_aligned\_t [slice](#)
    - uint64\_aligned\_t [latency](#)
    - uint32\_t [extratime](#)
    - uint32\_t [weight](#)
  - [sedf](#)
  - struct [xen\\_domctl\\_sched\\_credit](#) {
    - uint16\_t [weight](#)
    - uint16\_t [cap](#)
  - [credit](#)
  - struct [xen\\_domctl\\_sched\\_credit2](#) {
    - uint16\_t [weight](#)
  - [credit2](#)
- [u](#)

### 3.183.1 Field Documentation

#### 3.183.1.1 uint16\_t cap

#### 3.183.1.2 uint32\_t cmd

#### 3.183.1.3 struct { ... } ::xen\_domctl\_sched\_credit credit

#### 3.183.1.4 struct { ... } ::xen\_domctl\_sched\_credit2 credit2

3.183.1.5 uint32\_t extratime

3.183.1.6 uint64\_aligned\_t latency

3.183.1.7 uint64\_aligned\_t period

3.183.1.8 uint32\_t sched\_id

3.183.1.9 struct { ... } ::xen\_domctl\_sched\_sedf sedf

3.183.1.10 uint64\_aligned\_t slice

3.183.1.11 union { ... } u

3.183.1.12 uint32\_t weight

3.183.1.13 uint16\_t weight

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.184 xen\_domctl\_sendtrigger Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [trigger](#)
- uint32\_t [vcpu](#)

### 3.184.1 Field Documentation

3.184.1.1 uint32\_t trigger

3.184.1.2 uint32\_t vcpu

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.185 xen\_domctl\_set\_access\_required Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint8\_t [access\\_required](#)

### 3.185.1 Field Documentation

#### 3.185.1.1 uint8\_t access\_required

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.186 xen\_domctl\_set\_opt\_feature Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [uint64\\_t dummy](#)

### 3.186.1 Field Documentation

#### 3.186.1.1 uint64\_t dummy

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.187 xen\_domctl\_set\_target Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [domid\\_t target](#)

### 3.187.1 Field Documentation

#### 3.187.1.1 domid\_t target

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.188 xen\_domctl\_setdebugging Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [uint8\\_t enable](#)

### 3.188.1 Field Documentation

#### 3.188.1.1 uint8\_t enable

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.189 xen\_domctl\_setdomainhandle Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [xen\\_domain\\_handle\\_t handle](#)

### 3.189.1 Field Documentation

#### 3.189.1.1 xen\_domain\_handle\_t handle

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.190 xen\_domctl\_settimeoffset Struct Reference

```
#include <domctl.h>
```

### Data Fields

- [int32\\_t time\\_offset\\_seconds](#)

### 3.190.1 Field Documentation

#### 3.190.1.1 int32\_t time\_offset\_seconds

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.191 xen\_domctl\_shadow\_op Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (uint8) dirty\_bitmap

## Data Fields

- uint32\_t [op](#)
- uint32\_t [mode](#)
- uint32\_t [mb](#)
- uint64\_aligned\_t [pages](#)
- struct [xen\\_domctl\\_shadow\\_op\\_stats](#) [stats](#)

### 3.191.1 Member Function Documentation

3.191.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( [uint8](#) )

### 3.191.2 Field Documentation

3.191.2.1 [uint32\\_t](#) [mb](#)

3.191.2.2 [uint32\\_t](#) [mode](#)

3.191.2.3 [uint32\\_t](#) [op](#)

3.191.2.4 [uint64\\_aligned\\_t](#) [pages](#)

3.191.2.5 [struct xen\\_domctl\\_shadow\\_op\\_stats](#) [stats](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.192 xen\_domctl\_shadow\_op\_stats Struct Reference

```
#include <domctl.h>
```

## Data Fields

- uint32\_t [fault\\_count](#)
- uint32\_t [dirty\\_count](#)

### 3.192.1 Field Documentation

3.192.1.1 [uint32\\_t](#) [dirty\\_count](#)

3.192.1.2 [uint32\\_t](#) [fault\\_count](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.193 xen\_domctl\_subscribe Struct Reference

```
#include <domctl.h>
```

## Data Fields

- [uint32\\_t port](#)

### 3.193.1 Field Documentation

#### 3.193.1.1 [uint32\\_t port](#)

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.194 [xen\\_domctl\\_tsc\\_info](#) Struct Reference

```
#include <domctl.h>
```

## Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) ([xen\\_guest\\_tsc\\_info\\_t](#)) out\_info

## Data Fields

- [xen\\_guest\\_tsc\\_info\\_t](#) info

### 3.194.1 Member Function Documentation

#### 3.194.1.1 [XEN\\_GUEST\\_HANDLE\\_64](#) ( [xen\\_guest\\_tsc\\_info\\_t](#) )

### 3.194.2 Field Documentation

#### 3.194.2.1 [xen\\_guest\\_tsc\\_info\\_t](#) info

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.195 [xen\\_domctl\\_vcpuaffinity](#) Struct Reference

```
#include <domctl.h>
```

## Data Fields

- [uint32\\_t vcpu](#)
- [struct xenctl\\_cpumap](#) [cpumap](#)

### 3.195.1 Field Documentation

#### 3.195.1.1 struct xenctl\_cpumap cpumap

#### 3.195.1.2 uint32\_t vcpu

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.196 xen\_domctl\_vcpucontext Struct Reference

```
#include <domctl.h>
```

### Public Member Functions

- [XEN\\_GUEST\\_HANDLE\\_64](#) (vcpu\_guest\_context\_t) ctxt

### Data Fields

- uint32\_t [vcpu](#)

### 3.196.1 Member Function Documentation

#### 3.196.1.1 XEN\_GUEST\_HANDLE\_64 ( vcpu\_guest\_context\_t )

### 3.196.2 Field Documentation

#### 3.196.2.1 uint32\_t vcpu

The documentation for this struct was generated from the following file:

- [xen/xen/include/public/domctl.h](#)

## 3.197 xen\_guest\_tsc\_info Struct Reference

```
#include <domctl.h>
```

### Data Fields

- uint32\_t [tsc\\_mode](#)
- uint32\_t [gtsc\\_khz](#)
- uint32\_t [incarnation](#)
- uint32\_t [pad](#)
- uint64\_aligned\_t [elapsed\\_nsec](#)

### 3.197.1 Field Documentation

3.197.1.1 `uint64_aligned_t` `elapsed_nsec`

3.197.1.2 `uint32_t` `gtsc_khz`

3.197.1.3 `uint32_t` `incarnation`

3.197.1.4 `uint32_t` `pad`

3.197.1.5 `uint32_t` `tsc_mode`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/domctl.h`

## 3.198 `xen_multiboot_mod_list` Struct Reference

```
#include <xen.h>
```

### Data Fields

- `uint32_t` `mod_start`
- `uint32_t` `mod_end`
- `uint32_t` `cmdline`
- `uint32_t` `pad`

### 3.198.1 Field Documentation

3.198.1.1 `uint32_t` `cmdline`

3.198.1.2 `uint32_t` `mod_end`

3.198.1.3 `uint32_t` `mod_start`

3.198.1.4 `uint32_t` `pad`

The documentation for this struct was generated from the following file:

- `xen/xen/include/public/xen.h`

## 3.199 `xl_cmd` Struct Reference

```
#include <xl_cmd_intf.h>
```

### Data Fields

- enum `xl_cmdtype` `type`
- union {
  - struct `xl_cmd_ping` `ping`
  - struct `xl_cmd_getpage` `getpage`



```
struct xlc_cmd_retpage retpage
struct xlc_cmd_dommake dommake
struct xlc_cmd_dombuild dombuild
struct xlc_cmd_domcreate domcreate
} u
```

### 3.199.1 Field Documentation

- 3.199.1.1 struct xlc\_cmd\_dombuild dombuild
- 3.199.1.2 struct xlc\_cmd\_domcreate domcreate
- 3.199.1.3 struct xlc\_cmd\_dommake dommake
- 3.199.1.4 struct xlc\_cmd\_getpage getpage
- 3.199.1.5 struct xlc\_cmd\_ping ping
- 3.199.1.6 struct xlc\_cmd\_retpage retpage
- 3.199.1.7 enum xlc\_cmdtype type
- 3.199.1.8 union { ... } u

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

## 3.200 xlc\_cmd\_dombuild Struct Reference

```
#include <xlc_intf.h>
```

### Data Fields

- uint32\_t [in\\_gref](#)
- uint32\_t [out\\_gref](#)
- uint32\_t [domid](#)
- int [ret](#)

### 3.200.1 Field Documentation

- 3.200.1.1 uint32\_t domid
- 3.200.1.2 uint32\_t in\_gref
- 3.200.1.3 uint32\_t out\_gref
- 3.200.1.4 int ret

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

## 3.201 xlc\_cmd\_domcreate Struct Reference

```
#include <xlc_intf.h>
```

### Data Fields

- uint32\_t [in\\_gref](#)
- uint32\_t [out\\_gref](#)
- int [restore\\_fd](#)
- int [ret](#)
- uint32\_t [domid](#)
- uint32\_t [domid\\_dc](#)

### 3.201.1 Field Documentation

3.201.1.1 uint32\_t domid

3.201.1.2 uint32\_t domid\_dc

3.201.1.3 uint32\_t in\_gref

3.201.1.4 uint32\_t out\_gref

3.201.1.5 int restore\_fd

3.201.1.6 int ret

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

## 3.202 xlc\_cmd\_dommake Struct Reference

```
#include <xlc_intf.h>
```

### Data Fields

- uint32\_t [in\\_gref](#)
- uint32\_t [out\\_gref](#)
- int [ret](#)
- uint32\_t [domid](#)

### 3.202.1 Field Documentation

3.202.1.1 uint32\_t domid

3.202.1.2 uint32\_t in\_gref

3.202.1.3 uint32\_t out\_gref

#### 3.202.1.4 int ret

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

### 3.203 xlc\_cmd\_getpage Struct Reference

```
#include <xlc_intf.h>
```

#### Data Fields

- uint32\_t [in\\_gref](#)
- uint32\_t [out\\_gref](#)

#### 3.203.1 Detailed Description

Open a page used for sending commands

#### 3.203.2 Field Documentation

##### 3.203.2.1 uint32\_t in\_gref

##### 3.203.2.2 uint32\_t out\_gref

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

### 3.204 xlc\_cmd\_ping Struct Reference

```
#include <xlc_intf.h>
```

#### Data Fields

- int [ret](#)

#### 3.204.1 Detailed Description

Test if a DomT is running

#### 3.204.2 Field Documentation

##### 3.204.2.1 int ret

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

## 3.205 xlc\_cmd\_retpage Struct Reference

```
#include <xlc_intf.h>
```

### Data Fields

- uint32\_t [out\\_gref](#)

### 3.205.1 Field Documentation

#### 3.205.1.1 uint32\_t out\_gref

The documentation for this struct was generated from the following file:

- [xen/stubdom/domt/xlc\\_intf.h](#)

## 3.206 XLObject Struct Reference

### Data Fields

- [PyObject\\_HEAD](#)
- [libxl\\_ctx](#) ctx
- xentoollog\_logger\_stdiostream \* [logger](#)
- xentoollog\_level [minmsglevel](#)

### 3.206.1 Field Documentation

#### 3.206.1.1 libxl\_ctx ctx

#### 3.206.1.2 xentoollog\_logger\_stdiostream\* logger

#### 3.206.1.3 xentoollog\_level minmsglevel

#### 3.206.1.4 PyObject\_HEAD

The documentation for this struct was generated from the following file:

- [xen/tools/python/xen/lowlevel/xl/xl.c](#)

## 3.207 yyalloc Union Reference

### Data Fields

- [yytype\\_int16](#) yyss\_alloc
- [YYSTYPE](#) yyvs\_alloc
- [YYLTYPE](#) yyys\_alloc

### 3.207.1 Field Documentation

#### 3.207.1.1 YYLTYPE yyls\_alloc

#### 3.207.1.2 yytype\_int16 yyss\_alloc

#### 3.207.1.3 YYSTYPE yyvs\_alloc

The documentation for this union was generated from the following file:

- [xen/tools/libxl/libxlu\\_cfg\\_y.c](#)

## 3.208 YYLTYPE Struct Reference

```
#include <libxlu_cfg_y.h>
```

### Data Fields

- int [first\\_line](#)
- int [first\\_column](#)
- int [last\\_line](#)
- int [last\\_column](#)

### 3.208.1 Field Documentation

#### 3.208.1.1 int first\_column

#### 3.208.1.2 int first\_line

#### 3.208.1.3 int last\_column

#### 3.208.1.4 int last\_line

The documentation for this struct was generated from the following files:

- [xen/tools/libxl/libxlu\\_cfg\\_y.c](#)
- [xen/tools/libxl/libxlu\\_cfg\\_y.h](#)

## 3.209 YYSTYPE Union Reference

```
#include <libxlu_cfg_y.h>
```

### Data Fields

- char \* [string](#)
- XLU\_ConfigSetting \* [setting](#)

### 3.209.1 Field Documentation

3.209.1.1 XLU\_ConfigSetting \* setting

3.209.1.2 char \* string

The documentation for this union was generated from the following files:

- [xen/tools/libxl/libxlu\\_cfg\\_y.c](#)
- [xen/tools/libxl/libxlu\\_cfg\\_y.h](#)

## Chapter 4

# File Documentation

### 4.1 xen/extras/mini-os/arch/ia64/iorw.c File Reference

```
#include <mini-os/iorw.h>
#include <mini-os/console.h>
```

#### Functions

- void [iowrite8](#) (volatile void \*addr, uint8\_t val)
- void [iowrite32](#) (volatile void \*addr, uint32\_t val)
- uint8\_t [ioread8](#) (volatile void \*addr)
- uint32\_t [ioread32](#) (volatile void \*addr)

#### 4.1.1 Function Documentation

4.1.1.1 uint32\_t ioread32 ( volatile void \* *addr* )

4.1.1.2 uint8\_t ioread8 ( volatile void \* *addr* )

4.1.1.3 void iowrite32 ( volatile void \* *addr*, uint32\_t *val* )

4.1.1.4 void iowrite8 ( volatile void \* *addr*, uint8\_t *val* )

### 4.2 xen/extras/mini-os/arch/x86/iorw.c File Reference

```
#include <mini-os/iorw.h>
```

#### Functions

- void [iowrite8](#) (volatile void \*addr, uint8\_t val)
- void [iowrite32](#) (volatile void \*addr, uint32\_t val)
- uint8\_t [ioread8](#) (volatile void \*addr)
- uint32\_t [ioread32](#) (volatile void \*addr)

## 4.2.1 Function Documentation

4.2.1.1 `uint32_t ioread32 ( volatile void * addr )`

4.2.1.2 `uint8_t ioread8 ( volatile void * addr )`

4.2.1.3 `void iowrite32 ( volatile void * addr, uint32_t val )`

4.2.1.4 `void iowrite8 ( volatile void * addr, uint8_t val )`

## 4.3 xen/extras/mini-os/arch/x86/mm.c File Reference

```
#include <mini-os/os.h>
#include <mini-os/hypervisor.h>
#include <mini-os/mm.h>
#include <mini-os/types.h>
#include <mini-os/lib.h>
#include <mini-os/xmalloc.h>
#include <xen/memory.h>
```

### Macros

- `#define DEBUG(_f, _a...) ((void)0)`
- `#define MEM_TEST_MAX_ERRORS 10`
- `#define DEMAND_MAP_PAGES ((2ULL << 30) / PAGE_SIZE)`
- `#define HEAP_PAGES 0`
- `#define MAP_BATCH ((STACK_SIZE / 2) / sizeof(mmu_update_t))`
- `#define UNMAP_BATCH ((STACK_SIZE / 2) / sizeof(multicall_entry_t))`
- `#define MAX_CONTIG_ORDER 9 /* 2MB */`
- `#define L1_P2M_SHIFT 10`
- `#define L2_P2M_SHIFT 20`
- `#define L3_P2M_SHIFT 30`
- `#define L1_P2M_ENTRIES (1 << L1_P2M_SHIFT)`
- `#define L2_P2M_ENTRIES (1 << (L2_P2M_SHIFT - L1_P2M_SHIFT))`
- `#define L3_P2M_ENTRIES (1 << (L3_P2M_SHIFT - L2_P2M_SHIFT))`
- `#define L1_P2M_MASK (L1_P2M_ENTRIES - 1)`
- `#define L2_P2M_MASK (L2_P2M_ENTRIES - 1)`
- `#define L3_P2M_MASK (L3_P2M_ENTRIES - 1)`

### Functions

- void `page_walk` (unsigned long *va*)
- int `mem_test` (unsigned long *\*start\_va*, unsigned long *\*end\_va*, int *verbose*)
- `pentry_t *` `need_pgt` (unsigned long *va*)
- void `arch_init_demand_mapping_area` (unsigned long *cur\_pfn*)
- unsigned long `allocate_ondemand` (unsigned long *n*, unsigned long *alignment*)
- void `do_map_frames` (unsigned long *va*, const unsigned long *\*mfns*, unsigned long *n*, unsigned long *stride*, unsigned long *incr*, `domid_t` *id*, int *\*err*, unsigned long *prot*)
- void `* map_frames_ex` (const unsigned long *\*mfns*, unsigned long *n*, unsigned long *stride*, unsigned long *incr*, unsigned long *alignment*, `domid_t` *id*, int *\*err*, unsigned long *prot*)
- int `unmap_frames` (unsigned long *va*, unsigned long *num\_frames*)
- unsigned long `alloc_contig_pages` (int *order*, unsigned int *addr\_bits*)
- int `mfn_is_ram` (unsigned long *mfn*)
- void `arch_init_p2m` (unsigned long *max\_pfn*)
- void `arch_init_mm` (unsigned long *\*start\_pfn\_p*, unsigned long *\*max\_pfn\_p*)



## Variables

- unsigned long \* [phys\\_to\\_machine\\_mapping](#)
- unsigned long [mfn\\_zero](#)
- char [stack](#) []
- struct [shared\\_info](#) [shared\\_info](#)

### 4.3.1 Macro Definition Documentation

4.3.1.1 `#define DEBUG( _f, _a... ) ((void)0)`

4.3.1.2 `#define DEMAND_MAP_PAGES ((2ULL << 30) / PAGE_SIZE)`

4.3.1.3 `#define HEAP_PAGES 0`

4.3.1.4 `#define L1_P2M_ENTRIES (1 << L1_P2M_SHIFT)`

4.3.1.5 `#define L1_P2M_MASK (L1_P2M_ENTRIES - 1)`

4.3.1.6 `#define L1_P2M_SHIFT 10`

4.3.1.7 `#define L2_P2M_ENTRIES (1 << (L2_P2M_SHIFT - L1_P2M_SHIFT))`

4.3.1.8 `#define L2_P2M_MASK (L2_P2M_ENTRIES - 1)`

4.3.1.9 `#define L2_P2M_SHIFT 20`

4.3.1.10 `#define L3_P2M_ENTRIES (1 << (L3_P2M_SHIFT - L2_P2M_SHIFT))`

4.3.1.11 `#define L3_P2M_MASK (L3_P2M_ENTRIES - 1)`

4.3.1.12 `#define L3_P2M_SHIFT 30`

4.3.1.13 `#define MAP_BATCH ((STACK_SIZE / 2) / sizeof(mmu_update_t))`

Map an array of MFNs contiguously into virtual address space starting at va. map f[i\*stride]+i\*increment for i in 0..n-1.

4.3.1.14 `#define MAX_CONTIG_ORDER 9 /* 2MB */`

Allocate pages which are contiguous in machine memory. Returns a VA to where they are mapped or 0 on failure.

addr\_bits indicates if the region has restrictions on where it is located. Typical values are 32 (if for example PCI devices can't access 64bit memory) or 0 for no restrictions.

Allocated pages can be freed using the page allocators free\_pages() function.

based on Linux function xen\_create\_contiguous\_region()

4.3.1.15 `#define MEM_TEST_MAX_ERRORS 10`

A useful mem testing function. Write the address to every address in the range provided and read back the value. If verbose, print page walk to some VA

If we get MEM\_TEST\_MAX\_ERRORS we might as well stop

4.3.1.16 `#define UNMAP_BATCH ((STACK_SIZE / 2) / sizeof(multicall_entry_t))`

Unmap `nun_frames` frames mapped at virtual address `va`.

## 4.3.2 Function Documentation

4.3.2.1 `unsigned long alloc_contig_pages ( int order, unsigned int addr_bits )`

4.3.2.2 `unsigned long allocate_ondemand ( unsigned long n, unsigned long alignment )`

4.3.2.3 `void arch_init_demand_mapping_area ( unsigned long cur_pfn )`

4.3.2.4 `void arch_init_mm ( unsigned long * start_pfn_p, unsigned long * max_pfn_p )`

4.3.2.5 `void arch_init_p2m ( unsigned long max_pfn )`

4.3.2.6 `void do_map_frames ( unsigned long va, const unsigned long * mfns, unsigned long n, unsigned long stride, unsigned long incr, domid_t id, int * err, unsigned long prot )`

4.3.2.7 `void* map_frames_ex ( const unsigned long * mfns, unsigned long n, unsigned long stride, unsigned long incr, unsigned long alignment, domid_t id, int * err, unsigned long prot )`

Map an array of MFNs contiguous into virtual address space. Virtual addresses are allocated from the on demand area.

4.3.2.8 `int mem_test ( unsigned long * start_va, unsigned long * end_va, int verbose )`

4.3.2.9 `int mfn_is_ram ( unsigned long mfn )`

4.3.2.10 `pentry_t* need_pgt ( unsigned long va )`

return a valid PTE for a given virtual address. If PTE does not exist, allocate page-table pages.

4.3.2.11 `void page_walk ( unsigned long va )`

4.3.2.12 `int unmap_frames ( unsigned long va, unsigned long num_frames )`

## 4.3.3 Variable Documentation

4.3.3.1 `unsigned long mfn_zero`

4.3.3.2 `unsigned long* phys_to_machine_mapping`

4.3.3.3 `struct shared_info shared_info`

Mark portion of the address space read only.

4.3.3.4 `char stack[]`

## 4.4 xen/xen/arch/x86/mm.c File Reference

```
#include <xen/config.h>
#include <xen/init.h>
#include <xen/kernel.h>
#include <xen/lib.h>
#include <xen/mm.h>
#include <xen/domain.h>
#include <xen/sched.h>
#include <xen/errno.h>
#include <xen/perfc.h>
#include <xen/irq.h>
#include <xen/softirq.h>
#include <xen/domain_page.h>
#include <xen/event.h>
#include <xen/iocap.h>
#include <xen/guest_access.h>
#include <asm/paging.h>
#include <asm/shadow.h>
#include <asm/page.h>
#include <asm/flush_tlb.h>
#include <asm/io.h>
#include <asm/ldt.h>
#include <asm/x86_emulate.h>
#include <asm/e820.h>
#include <asm/hypercall.h>
#include <asm/shared.h>
#include <public/memory.h>
#include <public/sched.h>
#include <xsm/xsm.h>
#include <xen/trace.h>
#include <asm/setup.h>
#include <asm/fixmap.h>
#include <asm/mem_sharing.h>
```

### Data Structures

- struct [memory\\_map\\_context](#)
- struct [ptwr\\_emulate\\_ctxt](#)

### Macros

- #define [MEM\\_LOG](#)(\_f, \_a...) gdprintk(XENLOG\_WARNING, \_f "\n", ## \_a)
- #define [PTE\\_UPDATE\\_WITH\\_CMPXCHG](#)
- #define [PAGE\\_CACHE\\_ATTRS](#) (\_PAGE\_PAT|\_PAGE\_PCD|\_PAGE\_PWT)
- #define [l1\\_disallow\\_mask](#)(d)
- #define [l3\\_disallow\\_mask](#)(d) L3\_DISALLOW\_MASK
- #define [define\\_get\\_linear\\_pagetable](#)(level)
- #define [adjust\\_guest\\_l1e](#)(\_p, \_d) ((void)(\_d))
- #define [adjust\\_guest\\_l2e](#)(\_p, \_d) ((void)(\_d))
- #define [adjust\\_guest\\_l3e](#)(\_p, \_d) ((void)(\_d))
- #define [unadjust\\_guest\\_l3e](#)(\_p, \_d) ((void)(\_d))
- #define [pae\\_flush\\_pg](#)(mfn, idx, n13e) ((void)0)
- #define [alloc\\_l4\\_table](#)(page, preemptible) (-EINVAL)

- `#define free_l4_table(page, preemptible) (-EINVAL)`
- `#define UPDATE_ENTRY(_t, _p, _o, _n, _m, _v, _ad)`
- `#define fixmap_domain_page(mfn) mfn_to_virt(mfn)`
- `#define fixunmap_domain_page(ptr) ((void)(ptr))`
- `#define l1f_to_lnf(f) (((f) & _PAGE_PRESENT) ? ((f) | _PAGE_PSE) : (f))`
- `#define lnf_to_l1f(f) (((f) & _PAGE_PRESENT) ? ((f) & ~_PAGE_PSE) : (f))`
- `#define flush_area(v, f)`

## Typedefs

- `typedef struct e820entry e820entry_t`

## Functions

- `l1_pgentry_t __attribute__((__section__(".bss.page_aligned")))`
- `boolean_param("allowsuperpage", opt_allow_superpage)`
- `void __init init_frametable(void)`
- `void __init arch_init_memory(void)`
- `int page_is_ram_type(unsigned long mfn, unsigned long mem_type)`
- `unsigned long domain_get_maximum_gpfn(struct domain *d)`
- `void share_xen_page_with_guest(struct page_info *page, struct domain *d, int readonly)`
- `void share_xen_page_with_privileged_guests(struct page_info *page, int readonly)`
- `void make_cr3(struct vcpu *v, unsigned long mfn)`
- `void write_ptbase(struct vcpu *v)`
- `void update_cr3(struct vcpu *v)`
- `int map_ldt_shadow_page(unsigned int off)`
- `int is_iomem_page(unsigned long mfn)`
- `int get_page_from_l1e(l1_pgentry_t l1e, struct domain *l1e_owner, struct domain *pg_owner)`
- `define_get_linear_pagetable(l2)`
- `define_get_linear_pagetable(l3)`
- `void put_page_from_l1e(l1_pgentry_t l1e, struct domain *l1e_owner)`
- `void put_page(struct page_info *page)`
- `struct domain * page_get_owner_and_reference(struct page_info *page)`
- `int get_page(struct page_info *page, struct domain *domain)`
- `int free_page_type(struct page_info *page, unsigned long type, int preemptible)`
- `void put_page_type(struct page_info *page)`
- `int get_page_type(struct page_info *page, unsigned long type)`
- `int put_page_type_preemptible(struct page_info *page)`
- `int get_page_type_preemptible(struct page_info *page, unsigned long type)`
- `void clear_superpage_mark(struct page_info *page)`
- `int new_guest_cr3(unsigned long mfn)`
- `int do_mmuxxt_op(XEN_GUEST_HANDLE(mmuext_op_t) uops, unsigned int count, XEN_GUEST_HANDLE(uint) pdone, unsigned int foreigndom)`
- `int do_mmu_update(XEN_GUEST_HANDLE(mmu_update_t) ureqs, unsigned int count, XEN_GUEST_HANDLE(uint) pdone, unsigned int foreigndom)`
- `int create_grant_host_mapping(uint64_t addr, unsigned long frame, unsigned int flags, unsigned int cache_flags)`
- `int replace_grant_host_mapping(uint64_t addr, unsigned long frame, uint64_t new_addr, unsigned int flags)`
- `int donate_page(struct domain *d, struct page_info *page, unsigned int memflags)`
- `int steal_page(struct domain *d, struct page_info *page, unsigned int memflags)`
- `int page_make_sharable(struct domain *d, struct page_info *page, int expected_refcnt)`
- `int page_make_private(struct domain *d, struct page_info *page)`
- `int do_update_va_mapping(unsigned long va, u64 val64, unsigned long flags)`
- `int do_update_va_mapping_otherdomain(unsigned long va, u64 val64, unsigned long flags, domid_t domid)`

- void `destroy_gdt` (struct `vcpu` \*v)
- long `set_gdt` (struct `vcpu` \*v, unsigned long \*frames, unsigned int entries)
- long `do_set_gdt` (XEN\_GUEST\_HANDLE(ulong) frame\_list, unsigned int entries)
- long `do_update_descriptor` (u64 pa, u64 desc)
- `DEFINE_XEN_GUEST_HANDLE` (e820entry\_t)
- long `arch_memory_op` (int op, XEN\_GUEST\_HANDLE(void) arg)
- int `ptwr_do_page_fault` (struct `vcpu` \*v, unsigned long addr, struct `cpu_user_regs` \*regs)
- void `free_xen_pagetable` (void \*v)
- int `map_pages_to_xen` (unsigned long virt, unsigned long mfn, unsigned long nr\_mfns, unsigned int flags)
- void `destroy_xen_mappings` (unsigned long s, unsigned long e)
- void `__set_fixmap` (enum `fixed_addresses` idx, unsigned long mfn, unsigned long flags)
- void `memguard_guard_stack` (void \*p)
- void `memguard_unguard_stack` (void \*p)

## Variables

- bool\_t `__read_mostly opt_allow_superpage`

### 4.4.1 Macro Definition Documentation

4.4.1.1 `#define adjust_guest_l1e( _p, _d ) ((void)(_d))`

4.4.1.2 `#define adjust_guest_l2e( _p, _d ) ((void)(_d))`

4.4.1.3 `#define adjust_guest_l3e( _p, _d ) ((void)(_d))`

4.4.1.4 `#define alloc_l4_table( page, preemptible ) (-EINVAL)`

4.4.1.5 `#define define_get_linear_pagetable( level )`

4.4.1.6 `#define fixmap_domain_page( mfn ) mfn_to_virt(mfn)`

4.4.1.7 `#define fixunmap_domain_page( ptr ) ((void)(ptr))`

4.4.1.8 `#define flush_area( v, f )`

**Value:**

```
(!local_irq_is_enabled() ?
    flush_area_local((const void *)v, f) : \
    flush_area_all((const void *)v, f))
```

4.4.1.9 `#define free_l4_table( page, preemptible ) (-EINVAL)`

4.4.1.10 `#define l1_disallow_mask( d )`

**Value:**

```
((d != dom_io) &&
    (rangeset_is_empty((d)->iomem_caps) &&
    rangeset_is_empty((d)->arch.ioport_caps) &&
    !has_arch_pdevs(d) &&
    !is_hvm_domain(d)) ?
    L1_DISALLOW_MASK : (L1_DISALLOW_MASK & ~PAGE_CACHE_ATTRS))
```

```

4.4.1.11 #define l1f_to_l1f( f ) (((f) & _PAGE_PRESENT) ? ((f) | _PAGE_PSE) : (f))

4.4.1.12 #define l3_disallow_mask( d ) L3_DISALLOW_MASK

4.4.1.13 #define l1f_to_l1f( f ) (((f) & _PAGE_PRESENT) ? ((f) & ~_PAGE_PSE) : (f))

4.4.1.14 #define MEM_LOG( _f, _a... ) gdprintk(XENLOG_WARNING, _f "\n", ## _a)

4.4.1.15 #define pae_flush_pgd( mfn, idx, nl3e ) ((void)0)

4.4.1.16 #define PAGE_CACHE_ATTRS ( _PAGE_PAT | _PAGE_PCD | _PAGE_PWT)

4.4.1.17 #define PTE_UPDATE_WITH_CMPXCHG

4.4.1.18 #define unadjust_guest_l3e( _p, _d ) ((void)(_d))

4.4.1.19 #define UPDATE_ENTRY( _t, _p, _o, _n, _m, _v, _ad )

```

**Value:**

```

update_intpte(&_t ## e_get_intpte(*(_p)), \
              _t ## e_get_intpte(_o), _t ## e_get_intpte(_n), \
              (_m), (_v), (_ad))

```

**4.4.2 Typedef Documentation**

4.4.2.1 typedef struct e820entry e820entry\_t

**4.4.3 Function Documentation**

```

4.4.3.1 l1_pgentry_t __attribute__((__section__(".bss.page_aligned")))

4.4.3.2 void __set_fixmap ( enum fixed_addresses idx, unsigned long mfn, unsigned long flags )

4.4.3.3 void __init arch_init_memory ( void )

4.4.3.4 long arch_memory_op ( int op, XEN_GUEST_HANDLE(void) arg )

4.4.3.5 boolean_param ( "allowsuperpage", opt_allow_superpage )

4.4.3.6 void clear_superpage_mark ( struct page_info * page )

4.4.3.7 int create_grant_host_mapping ( uint64_t addr, unsigned long frame, unsigned int flags, unsigned int cache_flags )

4.4.3.8 define_get_linear_pagetable ( l2 )

```

NB. Virtual address 'l2e' maps to a machine address within frame 'pfn'.

```

4.4.3.9 define_get_linear_pagetable ( l3 )

4.4.3.10 DEFINE_XEN_GUEST_HANDLE ( e820entry_t )

4.4.3.11 void destroy_gdt ( struct vcpu * v )

4.4.3.12 void destroy_xen_mappings ( unsigned long s, unsigned long e )

```

- 4.4.3.13 int do\_mmu\_update ( XEN\_GUEST\_HANDLE(mmu\_update\_t) *ureqs*, unsigned int *count*, XEN\_GUEST\_HANDLE(uint) *pdone*, unsigned int *foreigndom* )
- 4.4.3.14 int do\_mmuext\_op ( XEN\_GUEST\_HANDLE(mmuext\_op\_t) *uops*, unsigned int *count*, XEN\_GUEST\_HANDLE(uint) *pdone*, unsigned int *foreigndom* )
- 4.4.3.15 long do\_set\_gdt ( XEN\_GUEST\_HANDLE(ulong) *frame\_list*, unsigned int *entries* )
- 4.4.3.16 long do\_update\_descriptor ( u64 *pa*, u64 *desc* )
- 4.4.3.17 int do\_update\_va\_mapping ( unsigned long *va*, u64 *val64*, unsigned long *flags* )
- 4.4.3.18 int do\_update\_va\_mapping\_otherdomain ( unsigned long *va*, u64 *val64*, unsigned long *flags*, domid\_t *domid* )
- 4.4.3.19 unsigned long domain\_get\_maximum\_gpfn ( struct domain \* *d* )
- 4.4.3.20 int donate\_page ( struct domain \* *d*, struct page\_info \* *page*, unsigned int *memflags* )
- 4.4.3.21 int free\_page\_type ( struct page\_info \* *page*, unsigned long *type*, int *preemptible* )
- 4.4.3.22 void free\_xen\_pagetable ( void \* *v* )
- 4.4.3.23 int get\_page ( struct page\_info \* *page*, struct domain \* *domain* )
- 4.4.3.24 int get\_page\_from\_l1e ( l1\_pgentry\_t *l1e*, struct domain \* *l1e\_owner*, struct domain \* *pg\_owner* )
- 4.4.3.25 int get\_page\_type ( struct page\_info \* *page*, unsigned long *type* )
- 4.4.3.26 int get\_page\_type\_preemptible ( struct page\_info \* *page*, unsigned long *type* )
- 4.4.3.27 void \_\_init init\_frametable ( void )
- 4.4.3.28 int is\_iomem\_page ( unsigned long *mfn* )
- 4.4.3.29 void make\_cr3 ( struct vcpu \* *v*, unsigned long *mfn* )
- 4.4.3.30 int map\_ldt\_shadow\_page ( unsigned int *off* )

Map shadow page at offset .

- 4.4.3.31 int map\_pages\_to\_xen ( unsigned long *virt*, unsigned long *mfn*, unsigned long *nr\_mfns*, unsigned int *flags* )
- 4.4.3.32 void memguard\_guard\_stack ( void \* *p* )
- 4.4.3.33 void memguard\_unguard\_stack ( void \* *p* )
- 4.4.3.34 int new\_guest\_cr3 ( unsigned long *mfn* )
- 4.4.3.35 struct domain\* page\_get\_owner\_and\_reference ( struct page\_info \* *page* )
- 4.4.3.36 int page\_is\_ram\_type ( unsigned long *mfn*, unsigned long *mem\_type* )
- 4.4.3.37 int page\_make\_private ( struct domain \* *d*, struct page\_info \* *page* )
- 4.4.3.38 int page\_make\_sharable ( struct domain \* *d*, struct page\_info \* *page*, int *expected\_refcnt* )

4.4.3.39 `int ptwr_do_page_fault ( struct vcpu * v, unsigned long addr, struct cpu_user_regs * regs )`

Write page fault handler: check if guest is trying to modify a PTE.

4.4.3.40 `void put_page ( struct page_info * page )`

4.4.3.41 `void put_page_from_l1e ( l1_pgentry_t l1e, struct domain * l1e_owner )`

4.4.3.42 `void put_page_type ( struct page_info * page )`

4.4.3.43 `int put_page_type_preemptible ( struct page_info * page )`

4.4.3.44 `int replace_grant_host_mapping ( uint64_t addr, unsigned long frame, uint64_t new_addr, unsigned int flags )`

4.4.3.45 `long set_gdt ( struct vcpu * v, unsigned long * frames, unsigned int entries )`

4.4.3.46 `void share_xen_page_with_guest ( struct page_info * page, struct domain * d, int readonly )`

4.4.3.47 `void share_xen_page_with_privileged_guests ( struct page_info * page, int readonly )`

4.4.3.48 `int steal_page ( struct domain * d, struct page_info * page, unsigned int memflags )`

4.4.3.49 `void update_cr3 ( struct vcpu * v )`

Should be called after CR3 is updated.

Uses values found in `vcpu->arch.(guest_table` and `guest_table_user)`, and for HVM guests, `arch.monitor_table` and `hvm's guest CR3`.

Update ref counts to shadow tables appropriately.

4.4.3.50 `void write_ptbase ( struct vcpu * v )`

#### 4.4.4 Variable Documentation

4.4.4.1 `bool_t __read_mostly opt_allow_superpage`

### 4.5 xen/extras/mini-os/blkfront.c File Reference

```
#include <stdint.h>
#include <mini-os/os.h>
#include <mini-os/xenbus.h>
#include <mini-os/events.h>
#include <errno.h>
#include <xen/io/blkif.h>
#include <xen/io/protocols.h>
#include <mini-os/gnttab.h>
#include <mini-os/xmalloc.h>
#include <time.h>
#include <mini-os/blkfront.h>
#include <mini-os/lib.h>
#include <fcntl.h>
```



## Data Structures

- struct [blk\\_buffer](#)
- struct [blkfront\\_dev](#)

## Macros

- #define [strtoul](#) [simple\\_strtoul](#)
- #define [BLK\\_RING\\_SIZE](#) [\\_\\_RING\\_SIZE](#)((struct [blkif\\_sring](#) \*)0, [PAGE\\_SIZE](#))
- #define [GRANT\\_INVALID\\_REF](#) 0

## Functions

- [DECLARE\\_WAIT\\_QUEUE\\_HEAD](#) ([blkfront\\_queue](#))
- void [blkfront\\_handler](#) ([evtchn\\_port\\_t](#) port, struct [pt\\_regs](#) \*regs, void \*data)
- struct [blkfront\\_dev](#) \* [init\\_blkfront](#) (char \*\_nodename, struct [blkfront\\_info](#) \*info)
- void [shutdown\\_blkfront](#) (struct [blkfront\\_dev](#) \*dev)
- void [blkfront\\_aio](#) (struct [blkfront\\_aiocb](#) \*aiocbp, int write)
- void [blkfront\\_io](#) (struct [blkfront\\_aiocb](#) \*aiocbp, int write)
- void [blkfront\\_aio\\_push\\_operation](#) (struct [blkfront\\_aiocb](#) \*aiocbp, uint8\_t op)
- void [blkfront\\_sync](#) (struct [blkfront\\_dev](#) \*dev)
- int [blkfront\\_aio\\_poll](#) (struct [blkfront\\_dev](#) \*dev)

### 4.5.1 Macro Definition Documentation

4.5.1.1 #define [BLK\\_RING\\_SIZE](#) [\\_\\_RING\\_SIZE](#)((struct [blkif\\_sring](#) \*)0, [PAGE\\_SIZE](#))

4.5.1.2 #define [GRANT\\_INVALID\\_REF](#) 0

4.5.1.3 #define [strtoul](#) [simple\\_strtoul](#)

### 4.5.2 Function Documentation

4.5.2.1 void [blkfront\\_aio](#) ( struct [blkfront\\_aiocb](#) \* *aiocbp*, int *write* )

Issue an aio

4.5.2.2 int [blkfront\\_aio\\_poll](#) ( struct [blkfront\\_dev](#) \* *dev* )

4.5.2.3 void [blkfront\\_aio\\_push\\_operation](#) ( struct [blkfront\\_aiocb](#) \* *aiocbp*, uint8\_t *op* )

4.5.2.4 void [blkfront\\_handler](#) ( [evtchn\\_port\\_t](#) *port*, struct [pt\\_regs](#) \* *regs*, void \* *data* )

4.5.2.5 void [blkfront\\_io](#) ( struct [blkfront\\_aiocb](#) \* *aiocbp*, int *write* )

4.5.2.6 void [blkfront\\_sync](#) ( struct [blkfront\\_dev](#) \* *dev* )

4.5.2.7 [DECLARE\\_WAIT\\_QUEUE\\_HEAD](#) ( [blkfront\\_queue](#) )

4.5.2.8 struct [blkfront\\_dev](#)\* [init\\_blkfront](#) ( char \* *\_nodename*, struct [blkfront\\_info](#) \* *info* )

4.5.2.9 void [shutdown\\_blkfront](#) ( struct [blkfront\\_dev](#) \* *dev* )

## 4.6 xen/extras/mini-os/console/console.c File Reference

```
#include <mini-os/types.h>
#include <mini-os/wait.h>
#include <mini-os/mm.h>
#include <mini-os/hypervisor.h>
#include <mini-os/events.h>
#include <mini-os/os.h>
#include <mini-os/lib.h>
#include <mini-os/xenbus.h>
#include <xen/io/console.h>
```

### Macros

- #define [USE\\_XEN\\_CONSOLE](#)

### Functions

- void [xencons\\_rx](#) (char \*buf, unsigned len, struct pt\_regs \*regs)
- void [xencons\\_tx](#) (void)
- void [console\\_print](#) (struct consfront\_dev \*dev, char \*data, int length)
- void [print](#) (int direct, const char \*fmt, va\_list args)
- void [printk](#) (const char \*fmt,...)
- void [xprintk](#) (const char \*fmt,...)
- void [init\\_console](#) (void)
- void [fini\\_console](#) (struct consfront\_dev \*dev)

### 4.6.1 Macro Definition Documentation

#### 4.6.1.1 #define USE\_XEN\_CONSOLE

### 4.6.2 Function Documentation

#### 4.6.2.1 void console\_print ( struct consfront\_dev \* dev, char \* data, int length )

#### 4.6.2.2 void fini\_console ( struct consfront\_dev \* dev )

#### 4.6.2.3 void init\_console ( void )

#### 4.6.2.4 void print ( int direct, const char \* fmt, va\_list args )

#### 4.6.2.5 void printk ( const char \* fmt, ... )

#### 4.6.2.6 void xencons\_rx ( char \* buf, unsigned len, struct pt\_regs \* regs )

#### 4.6.2.7 void xencons\_tx ( void )

#### 4.6.2.8 void xprintk ( const char \* fmt, ... )

## 4.7 xen/extras/mini-os/discovery.c File Reference

## 4.8 xen/extras/mini-os/include/blkfront.h File Reference

```
#include <mini-os/wait.h>
#include <xen/io/blkif.h>
#include <mini-os/types.h>
```

### Data Structures

- struct [blkfront\\_aiocb](#)
- struct [blkfront\\_info](#)

### Macros

- #define [blkfront\\_aio\\_read](#)(aiocbp) [blkfront\\_aio](#)(aiocbp, 0)
- #define [blkfront\\_aio\\_write](#)(aiocbp) [blkfront\\_aio](#)(aiocbp, 1)
- #define [blkfront\\_read](#)(aiocbp) [blkfront\\_io](#)(aiocbp, 0)
- #define [blkfront\\_write](#)(aiocbp) [blkfront\\_io](#)(aiocbp, 1)

### Functions

- struct [blkfront\\_dev](#) \* [init\\_blkfront](#) (char \*nodename, struct [blkfront\\_info](#) \*info)
- void [blkfront\\_aio](#) (struct [blkfront\\_aiocb](#) \*aiocbp, int write)
- void [blkfront\\_io](#) (struct [blkfront\\_aiocb](#) \*aiocbp, int write)
- void [blkfront\\_aio\\_push\\_operation](#) (struct [blkfront\\_aiocb](#) \*aiocbp, uint8\_t op)
- int [blkfront\\_aio\\_poll](#) (struct [blkfront\\_dev](#) \*dev)
- void [blkfront\\_sync](#) (struct [blkfront\\_dev](#) \*dev)
- void [shutdown\\_blkfront](#) (struct [blkfront\\_dev](#) \*dev)

### Variables

- struct wait\_queue\_head [blkfront\\_queue](#)

#### 4.8.1 Macro Definition Documentation

4.8.1.1 #define [blkfront\\_aio\\_read](#)( *aiocbp* ) [blkfront\\_aio](#)(aiocbp, 0)

4.8.1.2 #define [blkfront\\_aio\\_write](#)( *aiocbp* ) [blkfront\\_aio](#)(aiocbp, 1)

4.8.1.3 #define [blkfront\\_read](#)( *aiocbp* ) [blkfront\\_io](#)(aiocbp, 0)

4.8.1.4 #define [blkfront\\_write](#)( *aiocbp* ) [blkfront\\_io](#)(aiocbp, 1)

#### 4.8.2 Function Documentation

4.8.2.1 void [blkfront\\_aio](#) ( struct [blkfront\\_aiocb](#) \* *aiocbp*, int *write* )

Issue an aio

- 4.8.2.2 `int blkfront_aio_poll ( struct blkfront_dev * dev )`
- 4.8.2.3 `void blkfront_aio_push_operation ( struct blkfront_aiocb * aiocbp, uint8_t op )`
- 4.8.2.4 `void blkfront_io ( struct blkfront_aiocb * aiocbp, int write )`
- 4.8.2.5 `void blkfront_sync ( struct blkfront_dev * dev )`
- 4.8.2.6 `struct blkfront_dev* init_blkfront ( char * nodename, struct blkfront_info * info )`
- 4.8.2.7 `void shutdown_blkfront ( struct blkfront_dev * dev )`

### 4.8.3 Variable Documentation

- 4.8.3.1 `struct wait_queue_head blkfront_queue`

## 4.9 xen/extras/mini-os/include/byteorder.h File Reference

```
#include <mini-os/byteswap.h>
#include <mini-os/endian.h>
```

### Macros

- `#define be16_to_cpu(v) bswap_16(v)`
- `#define be32_to_cpu(v) bswap_32(v)`
- `#define be64_to_cpu(v) bswap_64(v)`
- `#define le16_to_cpu(v) (v)`
- `#define le32_to_cpu(v) (v)`
- `#define le64_to_cpu(v) (v)`
- `#define cpu_to_be16(v) be16_to_cpu(v)`
- `#define cpu_to_be32(v) be32_to_cpu(v)`
- `#define cpu_to_be64(v) be64_to_cpu(v)`
- `#define cpu_to_le16(v) le16_to_cpu(v)`
- `#define cpu_to_le32(v) le32_to_cpu(v)`
- `#define cpu_to_le64(v) le64_to_cpu(v)`

### 4.9.1 Macro Definition Documentation

- 4.9.1.1 `#define be16_to_cpu( v ) bswap_16(v)`
- 4.9.1.2 `#define be32_to_cpu( v ) bswap_32(v)`
- 4.9.1.3 `#define be64_to_cpu( v ) bswap_64(v)`
- 4.9.1.4 `#define cpu_to_be16( v ) be16_to_cpu(v)`
- 4.9.1.5 `#define cpu_to_be32( v ) be32_to_cpu(v)`
- 4.9.1.6 `#define cpu_to_be64( v ) be64_to_cpu(v)`
- 4.9.1.7 `#define cpu_to_le16( v ) le16_to_cpu(v)`

4.9.1.8 `#define cpu_to_le32( v ) le32_to_cpu(v)`

4.9.1.9 `#define cpu_to_le64( v ) le64_to_cpu(v)`

4.9.1.10 `#define le16_to_cpu( v ) (v)`

4.9.1.11 `#define le32_to_cpu( v ) (v)`

4.9.1.12 `#define le64_to_cpu( v ) (v)`

## 4.10 xen/extras/mini-os/include/byteswap.h File Reference

`#include <mini-os/types.h>`

### Macros

- `#define bswap_16(x)`
- `#define bswap_32(x)`
- `#define bswap_64(x)`

### 4.10.1 Macro Definition Documentation

4.10.1.1 `#define bswap_16( x )`

**Value:**

```
((uint16_t) (
    (((uint16_t)(x) & (uint16_t)0x00ffU) << 8) |
    (((uint16_t)(x) & (uint16_t)0xff00U) >> 8)))
```

4.10.1.2 `#define bswap_32( x )`

**Value:**

```
((uint32_t) (
    (((uint32_t)(x) & (uint32_t)0x000000ffU) << 24) |
    (((uint32_t)(x) & (uint32_t)0x0000ff00U) << 8) |
    (((uint32_t)(x) & (uint32_t)0x00ff0000U) >> 8) |
    (((uint32_t)(x) & (uint32_t)0xff000000U) >> 24)))
```

4.10.1.3 `#define bswap_64( x )`

**Value:**

```
((uint64_t) (
    (((uint64_t)(x) & (uint64_t)0x00000000000000ffU) << 56) |
    (((uint64_t)(x) & (uint64_t)0x000000000000ff00U) << 40) |
    (((uint64_t)(x) & (uint64_t)0x0000000000ff0000U) << 24) |
    (((uint64_t)(x) & (uint64_t)0x00000000ff000000U) << 8) |
    (((uint64_t)(x) & (uint64_t)0x000000ff00000000U) >> 8) |
    (((uint64_t)(x) & (uint64_t)0x0000ff0000000000U) >> 24) |
    (((uint64_t)(x) & (uint64_t)0x00ff000000000000U) >> 40) |
    (((uint64_t)(x) & (uint64_t)0xff00000000000000U) >> 56)))
```

## 4.11 xen/extras/mini-os/include/discovery.h File Reference

### Functions

- void [discovery](#) (void)
- void [discovery\\_start\\_kernel](#) (void \*p)

#### 4.11.1 Function Documentation

4.11.1.1 void [discovery](#) ( void )

4.11.1.2 void [discovery\\_start\\_kernel](#) ( void \* *p* )

## 4.12 xen/extras/mini-os/include/endian.h File Reference

```
#include <arch_endian.h>
#include <arch_wordsize.h>
```

### Macros

- #define [\\_\\_LITTLE\\_ENDIAN](#) 1234
- #define [\\_\\_BIG\\_ENDIAN](#) 4321
- #define [\\_\\_PDP\\_ENDIAN](#) 3412
- #define [ARCH\\_ENDIAN\\_H](#)
- #define [\\_\\_FLOAT\\_WORD\\_ORDER](#) \_\_BYTE\_ORDER

#### 4.12.1 Macro Definition Documentation

4.12.1.1 #define [\\_\\_BIG\\_ENDIAN](#) 4321

4.12.1.2 #define [\\_\\_FLOAT\\_WORD\\_ORDER](#) \_\_BYTE\_ORDER

4.12.1.3 #define [\\_\\_LITTLE\\_ENDIAN](#) 1234

4.12.1.4 #define [\\_\\_PDP\\_ENDIAN](#) 3412

4.12.1.5 #define [ARCH\\_ENDIAN\\_H](#)

## 4.13 xen/extras/mini-os/include/ia64/arch\_endian.h File Reference

## 4.14 xen/extras/mini-os/include/x86/arch\_endian.h File Reference

## 4.15 xen/extras/mini-os/include/ia64/arch\_wordsize.h File Reference

### Macros

- #define [\\_\\_WORDSIZE](#) 64

### 4.15.1 Macro Definition Documentation

4.15.1.1 `#define __WORDSIZE 64`

## 4.16 xen/extras/mini-os/include/x86/x86\_32/arch\_wordsize.h File Reference

### Macros

- `#define __WORDSIZE 32`

### 4.16.1 Macro Definition Documentation

4.16.1.1 `#define __WORDSIZE 32`

## 4.17 xen/extras/mini-os/include/x86/x86\_64/arch\_wordsize.h File Reference

### Macros

- `#define __WORDSIZE 64`
- `#define __WORDSIZE_COMPAT32 1`

### 4.17.1 Macro Definition Documentation

4.17.1.1 `#define __WORDSIZE 64`

4.17.1.2 `#define __WORDSIZE_COMPAT32 1`

## 4.18 xen/extras/mini-os/include/iorw.h File Reference

```
#include <mini-os/types.h>
```

### Functions

- void `iowrite8` (volatile void \**addr*, uint8\_t *val*)
- void `iowrite32` (volatile void \**addr*, uint32\_t *val*)
- uint8\_t `ioread8` (volatile void \**addr*)
- uint32\_t `ioread32` (volatile void \**addr*)

### 4.18.1 Function Documentation

4.18.1.1 `uint32_t ioread32 ( volatile void * addr )`

4.18.1.2 `uint8_t ioread8 ( volatile void * addr )`

4.18.1.3 `void iowrite32 ( volatile void * addr, uint32_t val )`

4.18.1.4 `void iowrite8 ( volatile void * addr, uint8_t val )`

## 4.19 xen/extras/mini-os/include/lib.h File Reference

```
#include <stdarg.h>
#include <stddef.h>
#include <xen/xen.h>
#include <xen/event_channel.h>
#include "gntmap.h"
#include <mini-os/console.h>
#include <mini-os/xenbus.h>
```

### Data Structures

- struct [kvec](#)

### Macros

- #define [\\_p\(\\_x\)](#) ((void\*)(unsigned long)(\_x))
- #define [RAND\\_MIX](#) 2654435769U
- #define [ARRAY\\_SIZE\(x\)](#) (sizeof(x) / sizeof((x)[0]))
- #define [ASSERT\(x\)](#)
- #define [BUG\\_ON\(x\)](#) [ASSERT\(!\(x\)\)](#)

### Functions

- int [vsprintf](#) (char \*buf, size\_t size, const char \*fmt, va\_list args)
- int [vsnprintf](#) (char \*buf, size\_t size, const char \*fmt, va\_list args)
- int [snprintf](#) (char \*buf, size\_t size, const char \*fmt,...)
- int [scnprintf](#) (char \*buf, size\_t size, const char \*fmt,...)
- int [vsprintf](#) (char \*buf, const char \*fmt, va\_list args)
- int [sprintf](#) (char \*buf, const char \*fmt,...)
- int [vsscanf](#) (const char \*buf, const char \*fmt, va\_list args)
- int [sscanf](#) (const char \*buf, const char \*fmt,...)
- long [simple\\_strtol](#) (const char \*cp, char \*\*endp, unsigned int base)
- unsigned long [simple\\_strtoul](#) (const char \*cp, char \*\*endp, unsigned int base)
- long long [simple\\_strtoll](#) (const char \*cp, char \*\*endp, unsigned int base)
- unsigned long long [simple\\_strtoull](#) (const char \*cp, char \*\*endp, unsigned int base)
- int [memcmp](#) (const void \*cs, const void \*ct, size\_t count)
- void \* [memcpy](#) (void \*dest, const void \*src, size\_t count)
- int [strncmp](#) (const char \*cs, const char \*ct, size\_t count)
- int [strcmp](#) (const char \*cs, const char \*ct)
- char \* [strcpy](#) (char \*dest, const char \*src)
- char \* [strncpy](#) (char \*dest, const char \*src, size\_t count)
- void \* [memset](#) (void \*s, int c, size\_t count)
- size\_t [strlen](#) (const char \*s, size\_t count)
- size\_t [strlen](#) (const char \*s)
- char \* [strchr](#) (const char \*s, int c)
- char \* [strrchr](#) (const char \*s, int c)
- char \* [strstr](#) (const char \*s1, const char \*s2)
- char \* [strcat](#) (char \*dest, const char \*src)
- char \* [strdup](#) (const char \*s)
- int [rand](#) (void)
- void [sanity\\_check](#) (void)



## 4.19.1 Macro Definition Documentation

4.19.1.1 `#define _p( _x ) ((void *) (unsigned long)( _x ))`

4.19.1.2 `#define ARRAY_SIZE( x ) (sizeof(x) / sizeof((x)[0]))`

4.19.1.3 `#define ASSERT( x )`

**Value:**

```
do {
    if (!(x)) {
        printk("ASSERTION FAILED: %s at %s:%d.\n",
               # x ,
               __FILE__,
               __LINE__ );
        BUG();
    }
} while (0)
```

4.19.1.4 `#define BUG_ON( x ) ASSERT(!(x))`

4.19.1.5 `#define RAND_MIX 2654435769U`

## 4.19.2 Function Documentation

4.19.2.1 `int memcmp ( const void * cs, const void * ct, size_t count )`

4.19.2.2 `void* memcpy ( void * dest, const void * src, size_t count )`

4.19.2.3 `void* memset ( void * s, int c, size_t count )`

4.19.2.4 `int rand ( void )`

4.19.2.5 `void sanity_check ( void )`

4.19.2.6 `int scnprintf ( char * buf, size_t size, const char * fmt, ... )`

4.19.2.7 `long simple_strtol ( const char * cp, char ** endp, unsigned int base )`

4.19.2.8 `long long simple_strtoll ( const char * cp, char ** endp, unsigned int base )`

4.19.2.9 `unsigned long simple_strtoul ( const char * cp, char ** endp, unsigned int base )`

4.19.2.10 `unsigned long long simple_strtoull ( const char * cp, char ** endp, unsigned int base )`

4.19.2.11 `int snprintf ( char * buf, size_t size, const char * fmt, ... )`

4.19.2.12 `int sprintf ( char * buf, const char * fmt, ... )`

4.19.2.13 `int sscanf ( const char * buf, const char * fmt, ... )`

4.19.2.14 `char* strcat ( char * dest, const char * src )`

4.19.2.15 `char* strchr ( const char * s, int c )`

4.19.2.16 `int strcmp ( const char * cs, const char * ct )`

- 4.19.2.17 `char* strcpy ( char * dest, const char * src )`
- 4.19.2.18 `char* strdup ( const char * s )`
- 4.19.2.19 `size_t strlen ( const char * s )`
- 4.19.2.20 `int strncmp ( const char * cs, const char * ct, size_t count )`
- 4.19.2.21 `char* strncpy ( char * dest, const char * src, size_t count )`
- 4.19.2.22 `size_t strnlen ( const char * s, size_t count )`
- 4.19.2.23 `char* strrchr ( const char * s, int c )`
- 4.19.2.24 `char* strstr ( const char * s1, const char * s2 )`
- 4.19.2.25 `int vsnprintf ( char * buf, size_t size, const char * fmt, va_list args )`
- 4.19.2.26 `int vsnprintf ( char * buf, size_t size, const char * fmt, va_list args )`
- 4.19.2.27 `int vsprintf ( char * buf, const char * fmt, va_list args )`
- 4.19.2.28 `int vsscanf ( const char * buf, const char * fmt, va_list args )`

## 4.20 xen/extras/mini-os/include/minios\_macros.h File Reference

## 4.21 xen/stubdom/domt/integrated/minios\_macros.h File Reference

### Macros

- `#define DEBUG_ON`
- `#define DEBUG_TEST 1`
- `#define PANIC(...) { ERROR("PANIC() " __VA_ARGS__); while (1); }`
- `#define DEBUG(fmt,...)`
- `#define ERROR(fmt,...)`
- `#define XENBUS_XWRITE(path, value)`
- `#define XENBUS_XWRITE_INT(path, value)`
- `#define XENBUS_XPRINTF(path, key, value,...)`
- `#define XENBUS_XSET_PERMS(path, domid, perm)`
- `#define XENBUS_XREAD(path, value)`
- `#define XENBUS_XREAD2(path1, path2, value)`
- `#define XENBUS_READ2(path1, path2, value)`
- `#define XENBUS_XREAD_INTEGER2(path1, path2, value)`

### 4.21.1 Macro Definition Documentation

#### 4.21.1.1 `#define DEBUG( fmt, ... )`

#### Value:

```
do { if (DEBUG_TEST) printk("%s:%d:%s(): " fmt, __FILE__,
    __LINE__, __func__, ##__VA_ARGS__); } while (0)
```

4.21.1.2 `#define DEBUG_ON`

Supplemental print and debug macros.

These are somewhat more verbose than the standard `printk()` offered by Mini-OS.

4.21.1.3 `#define DEBUG_TEST 1`4.21.1.4 `#define ERROR( fmt, ... )`

**Value:**

```
printk("***ERROR %s:%d:%s(): " fmt, __FILE__, __LINE__,
        __func__, ##__VA_ARGS__)
```

4.21.1.5 `#define PANIC( ... ) { ERROR("PANIC() " __VA_ARGS__); while (1); }`4.21.1.6 `#define XENBUS_READ2( path1, path2, value )`

**Value:**

```
do { char *path;
    asprintf(&path, "%s/%s", path1, path2);
    xenbus_read(XBT_NIL, path, value);
    free(path);
} while (0)
```

4.21.1.7 `#define XENBUS_XPRINTF( path, key, value, ... )`

**Value:**

```
do { char *msg = xenbus_printf(XBT_NIL, path, key, value, ##__VA_ARGS__);
    if (msg) {
        PANIC("Error %s while writing xenstore at %s/%s\n",
              msg, path, key);
    } } while (0)
```

4.21.1.8 `#define XENBUS_XREAD( path, value )`

**Value:**

```
do { char *msg = xenbus_read(XBT_NIL, path, value);
    if (msg) {
        PANIC("Error %s while reading xenstore at %s\n",
              msg, path);
    } } while (0)
```

4.21.1.9 `#define XENBUS_XREAD2( path1, path2, value )`

**Value:**

```
do { char *msg, *path;
    asprintf(&path, "%s/%s", path1, path2);
    msg = xenbus_read(XBT_NIL, path, value);
    if (msg) {
        PANIC("Error %s while reading xenstore at %s\n", msg, path);
    }
    free(path);
} while (0)
```

#### 4.21.1.10 #define XENBUS\_XREAD\_INTEGER2( path1, path2, value )

Value:

```
do { char *path;
    asprintf(&path, "%s/%s", path1, path2);
    *value = xenbus_read_integer(path);
    if (*value == -1) {
        PANIC("Error while reading xenstore at %s\n", path);
    }
    free(path);
} while (0)
```

#### 4.21.1.11 #define XENBUS\_XSET\_PERMS( path, domid, perm )

Value:

```
do { char *msg = xenbus_set_perms(XBT_NIL, path, domid, perm);
    if (msg) {
        PANIC("Error %s while setting permissions in xenstore at %s\n",
              msg, path);
    } } while (0)
```

#### 4.21.1.12 #define XENBUS\_XWRITE( path, value )

Value:

```
do { char *msg = xenbus_write(XBT_NIL, path, value);
    if (msg) { \
        PANIC("Error %s while writing xenstore at %s\n", msg, path); \
    } } while (0)
```

#### 4.21.1.13 #define XENBUS\_XWRITE\_INT( path, value )

Value:

```
do { char *msg, *tmp;
    asprintf(&tmp, "%d", value);
    msg = xenbus_write(XBT_NIL, path, tmp);
    if (msg) {
        PANIC("Error %s while writing xenstore at %s\n", msg, path);
    }
    free(tmp);
} while (0)
```

## 4.22 xen/extras/mini-os/include/tpm\_tis.h File Reference

```
#include <mini-os/os.h>
#include <mini-os/types.h>
#include <mini-os/byteorder.h>
```

### Macros

- #define [TPM\\_TIS\\_EN\\_LOCL0](#) 1
- #define [TPM\\_TIS\\_EN\\_LOCL1](#) (1 << 1)
- #define [TPM\\_TIS\\_EN\\_LOCL2](#) (1 << 2)
- #define [TPM\\_TIS\\_EN\\_LOCL3](#) (1 << 3)

- #define `TPM_TIS_EN_LOCL4` (1 << 4)
- #define `TPM_TIS_EN_LOCLALL` (`TPM_TIS_EN_LOCL0` | `TPM_TIS_EN_LOCL1` | `TPM_TIS_EN_LOCL2` | `TPM_TIS_EN_LOCL3` | `TPM_TIS_EN_LOCL4`)
- #define `TPM_TIS_LOCL_INT_TO_FLAG(x)` (1 << x)
- #define `TPM_BASEADDR` 0xFED40000
- #define `TPM_PROBE_IRQ` 0xFFFF

## Functions

- struct `tpm_chip` \* `init_tpm_tis` (unsigned long *baseaddr*, int *localities*, unsigned int *irq*)
- void `shutdown_tpm_tis` (struct `tpm_chip` \**tpm*)
- int `tpm_tis_request_locality` (struct `tpm_chip` \**tpm*, int *locality*)
- int `tpm_tis_cmd` (struct `tpm_chip` \**tpm*, uint8\_t \**req*, size\_t *reqlen*, uint8\_t \*\**resp*, size\_t \**resplen*)
- int `tpm_tis_transmit` (struct `tpm_chip` \**chip*, struct `tpm_cmd` \**cmd*, int *len*, const char \**desc*)

### 4.22.1 Macro Definition Documentation

4.22.1.1 #define `TPM_BASEADDR` 0xFED40000

4.22.1.2 #define `TPM_PROBE_IRQ` 0xFFFF

4.22.1.3 #define `TPM_TIS_EN_LOCL0` 1

4.22.1.4 #define `TPM_TIS_EN_LOCL1` (1 << 1)

4.22.1.5 #define `TPM_TIS_EN_LOCL2` (1 << 2)

4.22.1.6 #define `TPM_TIS_EN_LOCL3` (1 << 3)

4.22.1.7 #define `TPM_TIS_EN_LOCL4` (1 << 4)

4.22.1.8 #define `TPM_TIS_EN_LOCLALL` (`TPM_TIS_EN_LOCL0` | `TPM_TIS_EN_LOCL1` | `TPM_TIS_EN_LOCL2` | `TPM_TIS_EN_LOCL3` | `TPM_TIS_EN_LOCL4`)

4.22.1.9 #define `TPM_TIS_LOCL_INT_TO_FLAG( x )` (1 << x)

### 4.22.2 Function Documentation

4.22.2.1 struct `tpm_chip`\* `init_tpm_tis` ( unsigned long *baseaddr*, int *localities*, unsigned int *irq* )

4.22.2.2 void `shutdown_tpm_tis` ( struct `tpm_chip` \* *tpm* )

4.22.2.3 int `tpm_tis_cmd` ( struct `tpm_chip` \* *tpm*, uint8\_t \* *req*, size\_t *reqlen*, uint8\_t \*\* *resp*, size\_t \* *resplen* )

4.22.2.4 int `tpm_tis_request_locality` ( struct `tpm_chip` \* *tpm*, int *locality* )

4.22.2.5 int `tpm_tis_transmit` ( struct `tpm_chip` \* *chip*, struct `tpm_cmd` \* *cmd*, int *len*, const char \* *desc* )

## 4.23 xen/extras/mini-os/include/tpmback.h File Reference

```
#include <xen/io/tpmif.h>
#include <xen/io/xenbus.h>
#include <mini-os/types.h>
#include <xen/xen.h>
```

## Data Structures

- struct [tpmcmd](#)

## Typedefs

- typedef struct [tpmcmd](#) [tpmcmd\\_t](#)

## Functions

- void [init\\_tpmback](#) (char \*\*exclusive\_uuids)
- void [shutdown\\_tpmback](#) (void)
- [tpmcmd\\_t](#) \* [tpmback\\_req\\_any](#) (void)
- [tpmcmd\\_t](#) \* [tpmback\\_req](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- void [tpmback\\_resp](#) ([tpmcmd\\_t](#) \*tpmcmd)
- int [tpmback\\_wait\\_for\\_frontend\\_connect](#) ([domid\\_t](#) \*domid, unsigned int \*[handle](#))
- int [tpmback\\_num\\_frontends](#) (void)
- char \* [tpmback\\_get\\_uuid](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- void [tpmback\\_set\\_open\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_close\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_suspend\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_resume\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))

### 4.23.1 Typedef Documentation

#### 4.23.1.1 typedef struct tpmcmd tpmcmd\_t

### 4.23.2 Function Documentation

#### 4.23.2.1 void init\_tpmback ( char \*\* exclusive\_uuids )

Initialize the tpm backend driver - This is NULL terminated list of vtpm uuid strings. If this list is non-empty, then only frontend domains with vtpm uuid's matching entries in this list will be allowed to connect. Other connections will be immediatly closed. Set this argument to NULL to allow any vtpm to connect.

#### 4.23.2.2 void shutdown\_tpmback ( void )

Shutdown tpm backend driver

#### 4.23.2.3 char\* tpmback\_get\_uuid ( domid\_t domid, unsigned int handle )

Returns the uuid of the specified frontend, NULL on error

#### 4.23.2.4 int tpmback\_num\_frontends ( void )

returns the number of frontends connected

#### 4.23.2.5 tpmcmd\_t\* tpmback\_req ( domid\_t domid, unsigned int handle )

Blocks until a tpm command from the frontend at domid/handle is sent. Returns NULL if domid/handle is not connected, tpmback is shutdown or shutting down, or if there is an error

4.23.2.6 `tpmcmd_t* tpmback_req_any ( void )`

Blocks until a tpm command is sent from any front end. Returns a pointer to the tpm command to handle. Do not try to free this pointer or the req buffer This function will return NULL if the tpm backend driver is shutdown or any other error occurs

4.23.2.7 `void tpmback_resp ( tpmcmd_t * tpmcmd )`

Send the response to the tpm command back to the frontend This function will free the tpmcmd object, but you must free the resp buffer yourself

4.23.2.8 `void tpmback_set_close_callback ( void(*)(domid_t, unsigned int) cb )`

Specify a function to call when a tpm device disconnects

4.23.2.9 `void tpmback_set_open_callback ( void(*)(domid_t, unsigned int) cb )`

Specify a function to call when a new tpm device connects

4.23.2.10 `void tpmback_set_resume_callback ( void(*)(domid_t, unsigned int) cb )`4.23.2.11 `void tpmback_set_suspend_callback ( void(*)(domid_t, unsigned int) cb )`4.23.2.12 `int tpmback_wait_for_frontend_connect ( domid_t * domid, unsigned int * handle )`

Waits for the first frontend to connect and then sets domid and handle appropriately. If one or more frontends are already connected, this will set domid and handle to one of them arbitrarily. The main use for this function is to wait until a single frontend connection has occurred. returns 0 on success, non-zero on failure

## 4.24 xen/extras/mini-os/include/tpmfront.h File Reference

```
#include <mini-os/types.h>
#include <mini-os/os.h>
#include <mini-os/events.h>
#include <mini-os/wait.h>
#include <xen/xen.h>
#include <xen/io/xenbus.h>
#include <xen/io/tpmif.h>
```

## Data Structures

- struct [tpmfront\\_dev](#)

## Functions

- struct [tpmfront\\_dev](#) \* [init\\_tpmfront](#) (const char \*nodename)
- void [shutdown\\_tpmfront](#) (struct [tpmfront\\_dev](#) \*dev)
- int [tpmfront\\_cmd](#) (struct [tpmfront\\_dev](#) \*dev, uint8\_t \*req, size\_t reqlen, uint8\_t \*\*resp, size\_t \*resplen)

#### 4.24.1 Function Documentation

4.24.1.1 `struct tpmfront_dev* init_tpmfront ( const char * nodename )`

4.24.1.2 `void shutdown_tpmfront ( struct tpmfront_dev * dev )`

4.24.1.3 `int tpmfront_cmd ( struct tpmfront_dev * dev, uint8_t * req, size_t reqlen, uint8_t ** resp, size_t * resplen )`

### 4.25 xen/extras/mini-os/include/uuid.h File Reference

#### Typedefs

- typedef unsigned char `uuid_t`

#### Functions

- void `uuid_create (uuid_t *uuid, uint32_t *status)`
- void `uuid_to_string (uuid_t *uuid, char **str, uint32_t *status)`
- int `uuid_is_nil (uuid_t *uuid, unsigned int *status)`

#### 4.25.1 Typedef Documentation

4.25.1.1 typedef unsigned char `uuid_t`

#### 4.25.2 Function Documentation

4.25.2.1 `void uuid_create ( uuid_t * uuid, uint32_t * status )`

just fill the struct with random values for now

4.25.2.2 `int uuid_is_nil ( uuid_t * uuid, unsigned int * status )`

4.25.2.3 `void uuid_to_string ( uuid_t * uuid, char ** str, uint32_t * status )`

convert the struct to a printable string

### 4.26 xen/extras/mini-os/include/vfs.h File Reference

### 4.27 xen/stubdom/domt/integrated/vfs.h File Reference

```
#include <minios_macros.h>
#include <vfs_intf.h>
```

#### Functions

- int `vfs_init (void)`
- int `vfs_command (struct vfs_cmd *cmd)`



### 4.27.1 Function Documentation

4.27.1.1 `int vfs_command ( struct vfs_cmd * cmd )`

4.27.1.2 `int vfs_init ( void )`

See `vfs_hooks.h` for the bulk of public functions.

The `init` command has become deprecated. Maybe it can be useful to test if VFS is alive at boottime. It might also play a role in detection of `xenvfsd` restart/crashing; since then we need to invalidate all `vfs` file descriptors. Not a priority task, however.

## 4.28 xen/extras/mini-os/include/vfs\_hooks.h File Reference

### 4.29 xen/stubdom/domt/integrated/vfs\_hooks.h File Reference

```
#include <vfs_intf.h>
```

#### Functions

- `int vfs_open` (const char \*pathname, int flags, mode\_t mode)
- `int vfs_close` (int fd)
- `int vfs_stat` (const char \*path, struct stat \*buf)
- `off_t vfs_lseek` (int fd, off\_t offset, int whence)
- `ssize_t vfs_read` (int fd, void \*buf, size\_t nbyte)
- `ssize_t vfs_write` (int fd, const void \*buf, size\_t nbyte)
- `void vfs_major_minor` (const char \*physpath, int \*major, int \*minor)
- `void * vfs_mmap` (void \*addr, size\_t length, int prot, int flags, int fd, off\_t offset)

### 4.29.1 Function Documentation

4.29.1.1 `int vfs_close ( int fd )`

4.29.1.2 `off_t vfs_lseek ( int fd, off_t offset, int whence )`

4.29.1.3 `void vfs_major_minor ( const char * physpath, int * major, int * minor )`

combines the major and minor calls into one

4.29.1.4 `void* vfs_mmap ( void * addr, size_t length, int prot, int flags, int fd, off_t offset )`

For now, `mmap` is synthetic. it is a TODO to build a native solution via grant tables. We probably want to share a writable `gref_map` with Dom0 then. Until then, this function does not sync back changes. Since this call is only used in the domainbuilder for reading in the images, priority for this fix is low currently.

4.29.1.5 `int vfs_open ( const char * pathname, int flags, mode_t mode )`

NOTE: `munmap`, `close`, are not hooked since minios default implementation suffices. NOTE: `newlib` implements for us: `fread()` `fwrite()` `fopen()` `fclose()`.

4.29.1.6 `ssize_t vfs_read ( int fd, void * buf, size_t nbyte )`

4.29.1.7 `int vfs_stat ( const char * path, struct stat * buf )`

4.29.1.8 `ssize_t vfs_write ( int fd, const void * buf, size_t nbyte )`

## 4.30 xen/extras/mini-os/include/vfs\_intf.h File Reference

## 4.31 xen/stubdom/domt/integrated/vfs\_intf.h File Reference

```
#include <sys/stat.h>
#include <fcntl.h>
```

### Data Structures

- struct [vfs\\_cmd\\_init](#)
- struct [vfs\\_cmd\\_open](#)
- struct [vfs\\_cmd\\_close](#)
- struct [vfs\\_cmd\\_stat](#)
- struct [vfs\\_cmd\\_lseek](#)
- struct [vfs\\_cmd\\_read](#)
- struct [vfs\\_cmd\\_write](#)
- struct [vfs\\_cmd\\_majorminor](#)
- struct [vfs\\_cmd](#)
- struct [vfs\\_gref\\_map](#)

### Macros

- `#define VFS_PATH "/local/domain/0/domt/vfs"`
- `#define PAGE_SIZE 4096`
- `#define GREFS_PER_GREFMAP ((PAGE_SIZE - sizeof(unsigned int)) / sizeof(unsigned int))`
- `#define VFS_O_RDONLY 1UL << 1`
- `#define VFS_O_WRONLY 1UL << 2`
- `#define VFS_O_RDWR 1UL << 2`
- `#define VFS_O_CREAT 1UL << 4`
- `#define VFS_O_TRUNC 1UL << 5`

### Enumerations

- enum [vfs\\_cmdtype](#) {  
[VFS\\_INIT](#) = 1, [VFS\\_OPEN](#) = 2, [VFS\\_CLOSE](#) = 3, [VFS\\_STAT](#) = 4,  
[VFS\\_LSEEK](#) = 5, [VFS\\_READ](#) = 6, [VFS\\_WRITE](#) = 7, [VFS\\_MAJORMINOR](#) = 8 }

#### 4.31.1 Macro Definition Documentation

4.31.1.1 `#define GREFS_PER_GREFMAP ((PAGE_SIZE - sizeof(unsigned int)) / sizeof(unsigned int))`

4.31.1.2 `#define PAGE_SIZE 4096`

4.31.1.3 `#define VFS_O_CREAT 1UL << 4`

4.31.1.4 `#define VFS_O_RDONLY 1UL << 1`

Mini-OS (newlib) and Linux don't agree on the values of certain macros, therefore we introduce platform independent conversions.

4.31.1.5 `#define VFS_O_RDWR 1UL << 2`

4.31.1.6 `#define VFS_O_TRUNC 1UL << 5`

4.31.1.7 `#define VFS_O_WRONLY 1UL << 2`

4.31.1.8 `#define VFS_PATH "/local/domain/0/domt/vfs"`

VFS structures shared between Mini-OS and tools.

## 4.31.2 Enumeration Type Documentation

4.31.2.1 `enum vfs_cmdtype`

Enumerator

***VFS\_INIT***  
***VFS\_OPEN***  
***VFS\_CLOSE***  
***VFS\_STAT***  
***VFS\_LSEEK***  
***VFS\_READ***  
***VFS\_WRITE***  
***VFS\_MAJORMINOR***

## 4.32 xen/tools/xenvfsd/include/vfs\_intf.h File Reference

## 4.33 xen/extras/mini-os/include/x86/arch\_limits.h File Reference

Macros

- `#define __PAGE_SHIFT 12`
- `#define __PAGE_SIZE (1ULL << __PAGE_SHIFT)`
- `#define __STACK_SIZE_PAGE_ORDER 6`
- `#define __STACK_SIZE (__PAGE_SIZE * (1 << __STACK_SIZE_PAGE_ORDER))`

### 4.33.1 Macro Definition Documentation

4.33.1.1 `#define __PAGE_SHIFT 12`

4.33.1.2 `#define __PAGE_SIZE (1ULL << __PAGE_SHIFT)`

4.33.1.3 `#define __STACK_SIZE (__PAGE_SIZE * (1 << __STACK_SIZE_PAGE_ORDER))`

4.33.1.4 `#define __STACK_SIZE_PAGE_ORDER 6`

## 4.34 xen/extras/mini-os/kernel.c File Reference

```
#include <mini-os/os.h>
#include <mini-os/hypervisor.h>
#include <mini-os/mm.h>
#include <mini-os/events.h>
#include <mini-os/time.h>
#include <mini-os/types.h>
#include <mini-os/lib.h>
#include <mini-os/sched.h>
#include <mini-os/xenbus.h>
#include <mini-os/gnttab.h>
#include <mini-os/netfront.h>
#include <mini-os/blkfront.h>
#include <mini-os/fbfront.h>
#include <mini-os/pcifront.h>
#include <mini-os/xmalloc.h>
#include <fcntl.h>
#include <xen/features.h>
#include <xen/version.h>
```

### Data Structures

- struct [blk\\_req](#)

### Macros

- #define [WIDTH](#) 800
- #define [HEIGHT](#) 600
- #define [DEPTH](#) 32

### Functions

- void [setup\\_xen\\_features](#) (void)
- void [test\\_xenbus](#) (void)
- [\\_\\_attribute\\_\\_](#) ((weak))
- void [start\\_kernel](#) ([start\\_info\\_t](#) \*si)
- void [stop\\_kernel](#) (void)
- void [do\\_exit](#) (void)

### Variables

- uint8\_t [xen\\_features](#) [XENFEAT\_NR\_SUBMAPS \*32]

#### 4.34.1 Macro Definition Documentation

##### 4.34.1.1 #define DEPTH 32

##### 4.34.1.2 #define HEIGHT 600

##### 4.34.1.3 #define WIDTH 800

### 4.34.2 Function Documentation

4.34.2.1 `__attribute__ ( weak )`

4.34.2.2 `void do_exit ( void )`

This is called whenever an IRET fails in entry.S. This will generally be because an application has got itself into a really bad state (probably a bad CS or SS). It must be killed. Of course, minimal OS doesn't have applications :-)

4.34.2.3 `void setup_xen_features ( void )`

4.34.2.4 `void start_kernel ( start_info_t * si )`

4.34.2.5 `void stop_kernel ( void )`

4.34.2.6 `void test_xenbus ( void )`

### 4.34.3 Variable Documentation

4.34.3.1 `uint8_t xen_features[XENFEAT_NR_SUBMAPS *32]`

## 4.35 xen/extras/mini-os/lib/sys.c File Reference

### Macros

- `#define DEBUG(fmt,...)`

### 4.35.1 Macro Definition Documentation

4.35.1.1 `#define DEBUG( fmt, ... )`

## 4.36 xen/extras/mini-os/lib/xs.c File Reference

## 4.37 xen/extras/mini-os/main.c File Reference

## 4.38 xen/stubdom/domc/main.c File Reference

```
#include <stdlib.h>
#include <unistd.h>
#include <xenctrl.h>
#include <os.h>
#include <time.h>
#include <mini-os/gnttab.h>
#include <mini-os/events.h>
#include <sys/select.h>
#include <inttypes.h>
#include <stdio.h>
#include <string.h>
#include <byteorder.h>
#include <mini-os/blkfront.h>
#include "blkback.h"
#include "version.h"
#include "main.h"
#include "crypt.h"
#include "cspipe_intf.h"
```

### Macros

- #define [DIM\(x\)](#) (sizeof(x)/sizeof(\*(x)))

### Functions

- int [wrap\\_notify\\_remote\\_via\\_evtchn](#) (evtchn\_port\_t port)
- int [main](#) (int argc, char \*argv[])

### Variables

- grant\_entry\_v1\_t \* [grant\\_table](#)
- grant\_entry\_v2\_t \* [grant\\_table2](#)
- gnttab\_setup\_table\_t [setup\\_op](#)

#### 4.38.1 Macro Definition Documentation

4.38.1.1 #define [DIM\( x \)](#) (sizeof(x)/sizeof(\*(x)))

#### 4.38.2 Function Documentation

4.38.2.1 int [main](#) ( int *argc*, char \* *argv*[] )

4.38.2.2 int [wrap\\_notify\\_remote\\_via\\_evtchn](#) ( evtchn\_port\_t *port* )

#### 4.38.3 Variable Documentation

4.38.3.1 grant\_entry\_v1\_t\* [grant\\_table](#)

4.38.3.2 grant\_entry\_v2\_t\* [grant\\_table2](#)

## 4.38.3.3 gnttab\_setup\_table\_t setup\_op

## 4.39 xen/stubdom/domt/main.c File Reference

```
#include <stdlib.h>
#include <unistd.h>
#include <xenctrl.h>
#include <os.h>
#include <time.h>
#include <mini-os/gnttab.h>
#include <mini-os/events.h>
#include <sys/select.h>
#include <inttypes.h>
#include <stdio.h>
#include <string.h>
#include <libxl.h>
#include "version.h"
#include "main.h"
#include "caas.h"
#include "xlc.h"
#include "tc.h"
```

## Functions

- int [main](#) (int argc, char \*argv[])

## 4.39.1 Function Documentation

## 4.39.1.1 int main ( int argc, char \* argv[] )

## 4.40 xen/extras/mini-os/tpm\_tis.c File Reference

```
#include <mini-os/ioremap.h>
#include <mini-os/iorw.h>
#include <mini-os/tpm_tis.h>
#include <mini-os/os.h>
#include <mini-os/sched.h>
#include <mini-os/list.h>
#include <mini-os/byteorder.h>
#include <mini-os/events.h>
#include <mini-os/wait.h>
#include <mini-os/xmalloc.h>
#include <errno.h>
#include <stdbool.h>
```

## Data Structures

- struct [tpm\\_input\\_header](#)
- struct [tpm\\_output\\_header](#)
- struct [stclear\\_flags\\_t](#)
- struct [tpm\\_version\\_t](#)
- struct [tpm\\_version\\_1\\_2\\_t](#)

- struct [timeout\\_t](#)
- struct [duration\\_t](#)
- struct [permanent\\_flags\\_t](#)
- union [cap\\_t](#)
- struct [tpm\\_getcap\\_params\\_in](#)
- struct [tpm\\_getcap\\_params\\_out](#)
- struct [tpm\\_readpubek\\_params\\_out](#)
- union [tpm\\_cmd\\_header](#)
- struct [tpm\\_pcrread\\_out](#)
- struct [tpm\\_pcrread\\_in](#)
- struct [tpm\\_pcrextend\\_in](#)
- union [tpm\\_cmd\\_params](#)
- struct [tpm\\_cmd](#)
- struct [tpm\\_chip](#)

## Macros

- #define [min\(a, b\)](#) ( ((a) < (b)) ? (a) : (b) )
- #define [TPM\\_HEADER\\_SIZE](#) 10
- #define [TPM\\_BUFSIZE](#) 2048
- #define [TPM\\_DIGEST\\_SIZE](#) 20
- #define [TPM\\_MAX\\_ORDINAL](#) 243
- #define [TPM\\_MAX\\_PROTECTED\\_ORDINAL](#) 12
- #define [TPM\\_PROTECTED\\_ORDINAL\\_MASK](#) 0xFF
- #define [TPM\\_DIGEST\\_SIZE](#) 20
- #define [TPM\\_ERROR\\_SIZE](#) 10
- #define [TPM\\_RET\\_CODE\\_IDX](#) 6
- #define [TPM\\_CAP\\_FLAG\\_cpu\\_to\\_be32](#)(4)
- #define [TPM\\_CAP\\_PROP\\_cpu\\_to\\_be32](#)(5)
- #define [CAP\\_VERSION\\_1\\_1\\_cpu\\_to\\_be32](#)(0x06)
- #define [CAP\\_VERSION\\_1\\_2\\_cpu\\_to\\_be32](#)(0x1A)
- #define [TPM\\_CAP\\_PROP\\_PCR\\_cpu\\_to\\_be32](#)(0x101)
- #define [TPM\\_CAP\\_PROP\\_MANUFACTURER\\_cpu\\_to\\_be32](#)(0x103)
- #define [TPM\\_CAP\\_FLAG\\_PERM\\_cpu\\_to\\_be32](#)(0x108)
- #define [TPM\\_CAP\\_FLAG\\_VOL\\_cpu\\_to\\_be32](#)(0x109)
- #define [TPM\\_CAP\\_PROP\\_OWNER\\_cpu\\_to\\_be32](#)(0x111)
- #define [TPM\\_CAP\\_PROP\\_TIS\\_TIMEOUT\\_cpu\\_to\\_be32](#)(0x115)
- #define [TPM\\_CAP\\_PROP\\_TIS\\_DURATION\\_cpu\\_to\\_be32](#)(0x120)
- #define [TPM\\_INTERNAL\\_RESULT\\_SIZE](#) 200
- #define [TPM\\_TAG\\_RQU\\_COMMAND\\_cpu\\_to\\_be16](#)(193)
- #define [TPM\\_ORD\\_GET\\_CAP\\_cpu\\_to\\_be32](#)(101)
- #define [TPM\\_TIMEOUT](#) 5
- #define [TPM\\_ACCESS](#)(t, l) (((uint8\_t\*)t->pages[l]) + 0x0000)
- #define [TPM\\_INT\\_ENABLE](#)(t, l) ((uint32\_t\*)((uint8\_t\*)t->pages[l]) + 0x0008)
- #define [TPM\\_INT\\_VECTOR](#)(t, l) (((uint8\_t\*)t->pages[l]) + 0x000C)
- #define [TPM\\_INT\\_STATUS](#)(t, l) (((uint8\_t\*)t->pages[l]) + 0x0010)
- #define [TPM\\_INTF\\_CAPS](#)(t, l) ((uint32\_t\*)((uint8\_t\*)t->pages[l]) + 0x0014)
- #define [TPM\\_STS](#)(t, l) ((uint8\_t\*)((uint8\_t\*)t->pages[l]) + 0x0018)
- #define [TPM\\_DATA\\_FIFO](#)(t, l) (((uint8\_t\*)t->pages[l]) + 0x0024)
- #define [TPM\\_DID\\_VID](#)(t, l) ((uint32\_t\*)((uint8\_t\*)t->pages[l]) + 0x0F00)
- #define [TPM\\_RID](#)(t, l) (((uint8\_t\*)t->pages[l]) + 0x0F04)
- #define [tpm\\_tis\\_cancel\\_cmd](#)(v) tpm\_tis\_ready(v)



## Enumerations

- enum `tpm_duration` {  
`TPM_SHORT` = 0, `TPM_MEDIUM` = 1, `TPM_LONG` = 2, `TPM_UNDEFINED`,  
`TPM_SHORT` = 0, `TPM_MEDIUM` = 1, `TPM_LONG` = 2, `TPM_UNDEFINED` }
- enum `tis_access` {  
`TPM_ACCESS_VALID` = 0x80, `TPM_ACCESS_ACTIVE_LOCALITY` = 0x20, `TPM_ACCESS_RELINQUISH_LOCALITY` = 0x20, `TPM_ACCESS_REQUEST_PENDING` = 0x04,  
`TPM_ACCESS_REQUEST_USE` = 0x02 }
- enum `tis_status` {  
`TPM_STS_VALID` = 0x80, `TPM_STS_COMMAND_READY` = 0x40, `TPM_STS_DATA_AVAIL` = 0x10, `TPM_STS_DATA_EXPECT` = 0x08,  
`TPM_STS_GO` = 0x20 }
- enum `tis_int_flags` {  
`TPM_GLOBAL_INT_ENABLE` = 0x80000000, `TPM_INTF_BURST_COUNT_STATIC` = 0x100, `TPM_INTF_CMD_READY_INT` = 0x080, `TPM_INTF_INT_EDGE_FALLING` = 0x040,  
`TPM_INTF_INT_EDGE_RISING` = 0x020, `TPM_INTF_INT_LEVEL_LOW` = 0x010, `TPM_INTF_INT_LEVEL_HIGH` = 0x008, `TPM_INTF_LOCALITY_CHANGE_INT` = 0x004,  
`TPM_INTF_STS_VALID_INT` = 0x002, `TPM_INTF_DATA_AVAIL_INT` = 0x001 }
- enum `tis_defaults` { `TIS_MEM_BASE` = 0xFED40000, `TIS_MEM_LEN` = 0x5000, `TIS_SHORT_TIMEOUT` = 750, `TIS_LONG_TIMEOUT` = 2000 }

## Functions

- struct `tpm_input_header` `__attribute__((packed))`
- `MINIOS_LIST_HEAD` (`tpm_list`)
- `s_time_t` `tpm_calc_ordinal_duration` (struct `tpm_chip` \*`chip`, `uint32_t` `ordinal`)
- void `release_locality` (struct `tpm_chip` \*`tpm`, int `l`, int `force`)
- int `tpm_tis_request_locality` (struct `tpm_chip` \*`tpm`, int `l`)
- int `tpm_tis_recv` (struct `tpm_chip` \*`tpm`, `uint8_t` \*`buf`, `size_t` `count`)
- int `tpm_tis_send` (struct `tpm_chip` \*`tpm`, `uint8_t` \*`buf`, `size_t` `len`)
- int `tpm_tis_transmit` (struct `tpm_chip` \*`chip`, struct `tpm_cmd` \*`cmd`, int `len`, const char \*`desc`)
- void `tpm_get_timeouts` (struct `tpm_chip` \*`chip`)
- void `tpm_continue_selftest` (struct `tpm_chip` \*`chip`)
- `ssize_t` `tpm_getcap` (struct `tpm_chip` \*`chip`, `uint32_t` `subcap_id`, `cap_t` \*`cap`, const char \*`desc`)
- struct `tpm_chip` \* `init_tpm_tis` (unsigned long `baseaddr`, int `localities`, unsigned int `irq`)
- void `shutdown_tpm_tis` (struct `tpm_chip` \*`tpm`)
- int `tpm_tis_cmd` (struct `tpm_chip` \*`tpm`, `uint8_t` \*`req`, `size_t` `reqlen`, `uint8_t` \*\*`resp`, `size_t` \*`resplen`)

## Variables

- typedef `__attribute__`
- `uint16_t` `tag`
- `uint32_t` `length`
- `uint32_t` `ordinal`
- `uint32_t` `return_code`
- `uint8_t` `deactivated`
- `uint8_t` `disableForceClear`
- `uint8_t` `physicalPresence`
- `uint8_t` `physicalPresenceLock`
- `uint8_t` `bGlobalLock`
- `uint8_t` `Major`
- `uint8_t` `Minor`
- `uint8_t` `revMajor`
- `uint8_t` `revMinor`

- uint32\_t [a](#)
- uint32\_t [b](#)
- uint32\_t [c](#)
- uint32\_t [d](#)
- uint32\_t [tpm\\_short](#)
- uint32\_t [tpm\\_medium](#)
- uint32\_t [tpm\\_long](#)
- uint8\_t [disable](#)
- uint8\_t [ownership](#)
- uint8\_t [readPubek](#)
- uint8\_t [disableOwnerClear](#)
- uint8\_t [allowMaintenance](#)
- uint8\_t [physicalPresenceLifetimeLock](#)
- uint8\_t [physicalPresenceHwEnable](#)
- uint8\_t [physicalPresenceCMDEnable](#)
- uint8\_t [CEKPUSED](#)
- uint8\_t [TPMpost](#)
- uint8\_t [TPMpostLock](#)
- uint8\_t [FIPS](#)
- uint8\_t [operator](#)
- uint8\_t [enableRevokeEK](#)
- uint8\_t [nvLocked](#)
- uint8\_t [readSRKPub](#)
- uint8\_t [tpmEstablished](#)
- uint8\_t [maintenanceDone](#)
- uint8\_t [disableFullIDALogicInfo](#)
- uint32\_t [cap](#)
- uint32\_t [subcap\\_size](#)
- uint32\_t [subcap](#)
- uint32\_t [cap\\_size](#)
- uint8\_t [algorithm](#) [4]
- uint8\_t [encscheme](#) [2]
- uint8\_t [sigscheme](#) [2]
- uint32\_t [paramsize](#)
- uint8\_t [parameters](#) [12]
- uint32\_t [keysize](#)
- uint8\_t [modulus](#) [256]
- uint8\_t [checksum](#) [20]
- uint8\_t [pcr\\_result](#) [TPM\_DIGEST\_SIZE]
- uint32\_t [pcr\\_idx](#)
- uint8\_t [hash](#) [TPM\_DIGEST\_SIZE]
- [tpm\\_cmd\\_header](#) header
- [tpm\\_cmd\\_params](#) params
- const uint8\_t [tpm\\_protected\\_ordinal\\_duration](#) [TPM\_MAX\_PROTECTED\_ORDINAL]
- const uint8\_t [tpm\\_ordinal\\_duration](#) [TPM\_MAX\_ORDINAL]
- struct [tpm\\_input\\_header](#) [tpm\\_getcap\\_header](#)

## 4.40.1 Macro Definition Documentation

- 4.40.1.1 `#define CAP_VERSION_1_1 cpu_to_be32(0x06)`
- 4.40.1.2 `#define CAP_VERSION_1_2 cpu_to_be32(0x1A)`
- 4.40.1.3 `#define min( a, b ) ((a) < (b)) ? (a) : (b)`
- 4.40.1.4 `#define TPM_ACCESS( t, l ) (((uint8_t*)t->pages[l]) + 0x0000)`
- 4.40.1.5 `#define TPM_BUFSIZE 2048`
- 4.40.1.6 `#define TPM_CAP_FLAG cpu_to_be32(4)`
- 4.40.1.7 `#define TPM_CAP_FLAG_PERM cpu_to_be32(0x108)`
- 4.40.1.8 `#define TPM_CAP_FLAG_VOL cpu_to_be32(0x109)`
- 4.40.1.9 `#define TPM_CAP_PROP cpu_to_be32(5)`
- 4.40.1.10 `#define TPM_CAP_PROP_MANUFACTURER cpu_to_be32(0x103)`
- 4.40.1.11 `#define TPM_CAP_PROP_OWNER cpu_to_be32(0x111)`
- 4.40.1.12 `#define TPM_CAP_PROP_PCR cpu_to_be32(0x101)`
- 4.40.1.13 `#define TPM_CAP_PROP_TIS_DURATION cpu_to_be32(0x120)`
- 4.40.1.14 `#define TPM_CAP_PROP_TIS_TIMEOUT cpu_to_be32(0x115)`
- 4.40.1.15 `#define TPM_DATA_FIFO( t, l ) (((uint8_t*)t->pages[l]) + 0x0024)`
- 4.40.1.16 `#define TPM_DID_VID( t, l ) ((uint32_t*)((uint8_t*)t->pages[l]) + 0x0F00)`
- 4.40.1.17 `#define TPM_DIGEST_SIZE 20`
- 4.40.1.18 `#define TPM_DIGEST_SIZE 20`
- 4.40.1.19 `#define TPM_ERROR_SIZE 10`
- 4.40.1.20 `#define TPM_HEADER_SIZE 10`
- 4.40.1.21 `#define TPM_INT_ENABLE( t, l ) ((uint32_t*)((uint8_t*)t->pages[l]) + 0x0008)`
- 4.40.1.22 `#define TPM_INT_STATUS( t, l ) (((uint8_t*)t->pages[l]) + 0x0010)`
- 4.40.1.23 `#define TPM_INT_VECTOR( t, l ) (((uint8_t*)t->pages[l]) + 0x000C)`
- 4.40.1.24 `#define TPM_INTERNAL_RESULT_SIZE 200`
- 4.40.1.25 `#define TPM_INTF_CAPS( t, l ) ((uint32_t*)((uint8_t*)t->pages[l]) + 0x0014)`
- 4.40.1.26 `#define TPM_MAX_ORDINAL 243`
- 4.40.1.27 `#define TPM_MAX_PROTECTED_ORDINAL 12`

```

4.40.1.28 #define TPM_ORD_GET_CAP cpu_to_be32(101)

4.40.1.29 #define TPM_PROTECTED_ORDINAL_MASK 0xFF

4.40.1.30 #define TPM_RET_CODE_IDX 6

4.40.1.31 #define TPM_RID( t, l ) (((uint8_t*)t->pages[l]) + 0x0F04)

4.40.1.32 #define TPM_STS( t, l ) (((uint8_t*)(((uint8_t*)t->pages[l]) + 0x0018))

4.40.1.33 #define TPM_TAG_RQU_COMMAND cpu_to_be16(193)

4.40.1.34 #define TPM_TIMEOUT 5

4.40.1.35 #define tpm_tis_cancel_cmd( v ) tpm_tis_ready(v)

```

## 4.40.2 Enumeration Type Documentation

### 4.40.2.1 enum tis\_access

Enumerator

```

TPM_ACCESS_VALID
TPM_ACCESS_ACTIVE_LOCALITY
TPM_ACCESS_RELINQUISH_LOCALITY
TPM_ACCESS_REQUEST_PENDING
TPM_ACCESS_REQUEST_USE

```

### 4.40.2.2 enum tis\_defaults

Enumerator

```

TIS_MEM_BASE
TIS_MEM_LEN
TIS_SHORT_TIMEOUT
TIS_LONG_TIMEOUT

```

### 4.40.2.3 enum tis\_int\_flags

Enumerator

```

TPM_GLOBAL_INT_ENABLE
TPM_INTF_BURST_COUNT_STATIC
TPM_INTF_CMD_READY_INT
TPM_INTF_INT_EDGE_FALLING
TPM_INTF_INT_EDGE_RISING
TPM_INTF_INT_LEVEL_LOW
TPM_INTF_INT_LEVEL_HIGH
TPM_INTF_LOCALITY_CHANGE_INT
TPM_INTF_STS_VALID_INT
TPM_INTF_DATA_AVAIL_INT

```

## 4.40.2.4 enum tis\_status

Enumerator

***TPM\_STS\_VALID***  
***TPM\_STS\_COMMAND\_READY***  
***TPM\_STS\_DATA\_AVAIL***  
***TPM\_STS\_DATA\_EXPECT***  
***TPM\_STS\_GO***

## 4.40.2.5 enum tpm\_duration

Enumerator

***TPM\_SHORT***  
***TPM\_MEDIUM***  
***TPM\_LONG***  
***TPM\_UNDEFINED***  
***TPM\_SHORT***  
***TPM\_MEDIUM***  
***TPM\_LONG***  
***TPM\_UNDEFINED***

## 4.40.3 Function Documentation

4.40.3.1 struct tpm\_input\_header \_\_attribute\_\_((packed))

4.40.3.2 struct tpm\_chip\* init\_tpm\_tis( unsigned long *baseaddr*, int *localities*, unsigned int *irq* )

4.40.3.3 MINIOS\_LIST\_HEAD( tpm\_list )

4.40.3.4 void release\_locality( struct tpm\_chip \* *tpm*, int *l*, int *force* )4.40.3.5 void shutdown\_tpm\_tis( struct tpm\_chip \* *tpm* )4.40.3.6 s\_time\_t tpm\_calc\_ordinal\_duration( struct tpm\_chip \* *chip*, uint32\_t *ordinal* )

Returns max number of nsecs to wait

4.40.3.7 void tpm\_continue\_selftest( struct tpm\_chip \* *chip* )4.40.3.8 void tpm\_get\_timeouts( struct tpm\_chip \* *chip* )4.40.3.9 ssize\_t tpm\_getcap( struct tpm\_chip \* *chip*, uint32\_t *subcap\_id*, cap\_t \* *cap*, const char \* *desc* )4.40.3.10 int tpm\_tis\_cmd( struct tpm\_chip \* *tpm*, uint8\_t \* *req*, size\_t *reqlen*, uint8\_t \*\* *resp*, size\_t \* *resplen* )4.40.3.11 int tpm\_tis\_recv( struct tpm\_chip \* *tpm*, uint8\_t \* *buf*, size\_t *count* )4.40.3.12 int tpm\_tis\_request\_locality( struct tpm\_chip \* *tpm*, int *l* )4.40.3.13 int tpm\_tis\_send( struct tpm\_chip \* *tpm*, uint8\_t \* *buf*, size\_t *len* )

4.40.3.14 `int tpm_tis_transmit ( struct tpm_chip * chip, struct tpm_cmd * cmd, int len, const char * desc )`

#### 4.40.4 Variable Documentation

4.40.4.1 `enum tpm_duration __attribute__`

4.40.4.2 `uint32_t a`

4.40.4.3 `uint8_t algorithm[4]`

4.40.4.4 `uint8_t allowMaintenance`

4.40.4.5 `uint32_t b`

4.40.4.6 `uint8_t bGlobalLock`

4.40.4.7 `uint32_t c`

4.40.4.8 `cap_t cap`

4.40.4.9 `uint32_t cap_size`

4.40.4.10 `uint8_t CEKPUsed`

4.40.4.11 `uint8_t checksum[20]`

4.40.4.12 `uint32_t d`

4.40.4.13 `uint8_t deactivated`

4.40.4.14 `uint8_t disable`

4.40.4.15 `uint8_t disableForceClear`

4.40.4.16 `uint8_t disableFullIDALogicInfo`

4.40.4.17 `uint8_t disableOwnerClear`

4.40.4.18 `uint8_t enableRevokeEK`

4.40.4.19 `uint8_t encscheme[2]`

4.40.4.20 `uint8_t FIPS`

4.40.4.21 `uint8_t hash[TPM_DIGEST_SIZE]`

4.40.4.22 `tpm_cmd_header header`

4.40.4.23 `uint32_t keysize`

4.40.4.24 `uint32_t length`

4.40.4.25 `uint8_t maintenanceDone`

4.40.4.26 `uint8_t Major`

- 4.40.4.27 `uint8_t` Minor
- 4.40.4.28 `uint8_t` modulus[256]
- 4.40.4.29 `uint8_t` nvLocked
- 4.40.4.30 `uint8_t` operator
- 4.40.4.31 `uint32_t` ordinal
- 4.40.4.32 `uint8_t` ownership
- 4.40.4.33 `uint8_t` parameters[12]
- 4.40.4.34 `tpm_cmd_params` params
- 4.40.4.35 `uint32_t` paramsize
- 4.40.4.36 `uint32_t` pcr\_idx
- 4.40.4.37 `uint8_t` pcr\_result[TPM\_DIGEST\_SIZE]
- 4.40.4.38 `uint8_t` physicalPresence
- 4.40.4.39 `uint8_t` physicalPresenceCMDEnable
- 4.40.4.40 `uint8_t` physicalPresenceHWEnable
- 4.40.4.41 `uint8_t` physicalPresenceLifetimeLock
- 4.40.4.42 `uint8_t` physicalPresenceLock
- 4.40.4.43 `uint8_t` readPubek
- 4.40.4.44 `uint8_t` readSRKPub
- 4.40.4.45 `uint32_t` return\_code
- 4.40.4.46 `uint8_t` revMajor
- 4.40.4.47 `uint8_t` revMinor
- 4.40.4.48 `uint8_t` sigscheme[2]
- 4.40.4.49 `uint32_t` subcap
- 4.40.4.50 `uint32_t` subcap\_size
- 4.40.4.51 `uint16_t` tag
- 4.40.4.52 `struct tpm_input_header` tpm\_getcap\_header

**Initial value:**

```
= {
    .tag = TPM_TAG_RQU_COMMAND,
    .length = cpu_to_be32(22),
    .ordinal = TPM_ORD_GET_CAP
```

```
}
```

4.40.4.53 `uint32_t tpm_long`

4.40.4.54 `uint32_t tpm_medium`

4.40.4.55 `const uint8_t tpm_ordinal_duration`

4.40.4.56 `const uint8_t tpm_protected_ordinal_duration`

**Initial value:**

```
= {
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_UNDEFINED,
    TPM_SHORT,
    TPM_SHORT,
}
```

4.40.4.57 `uint32_t tpm_short`

4.40.4.58 `uint8_t tpmEstablished`

4.40.4.59 `uint8_t TPMpost`

4.40.4.60 `uint8_t TPMpostLock`

## 4.41 xen/extras/mini-os/tpmback.c File Reference

```
#include <mini-os/os.h>
#include <mini-os/xenbus.h>
#include <mini-os/events.h>
#include <errno.h>
#include <mini-os/gnttab.h>
#include <xen/io/xenbus.h>
#include <xen/io/tpmif.h>
#include <xen/io/protocols.h>
#include <mini-os/xmalloc.h>
#include <time.h>
#include <mini-os/tpmback.h>
#include <mini-os/lib.h>
#include <fcntl.h>
#include <mini-os/mm.h>
#include <mini-os/posix/sys/mman.h>
#include <mini-os/semaphore.h>
#include <mini-os/wait.h>
```

## Data Structures

- struct `tpmif`



- struct [tpmback\\_dev](#)

## Macros

- #define [strtoul simple\\_strtoul](#)
- #define [TPMBACK\\_DEBUG](#)(fmt,...)
- #define [TPMBACK\\_ERR](#)(fmt,...) [printk](#)("Tpmback:Error " fmt, ##\_\_VA\_ARGS\_\_)
- #define [TPMBACK\\_LOG](#)(fmt,...) [printk](#)("Tpmback:Info " fmt, ##\_\_VA\_ARGS\_\_)
- #define [min](#)(a, b) (((a) < (b)) ? (a) : (b))
- #define [DEF\\_ARRAY\\_SIZE](#) 1
- #define [TPMIF\\_CLOSED](#) 1
- #define [TPMIF\\_REQ\\_READY](#) 2

## Typedefs

- typedef struct [tpmif](#) [tpmif\\_t](#)
- typedef struct [tpmback\\_dev](#) [tpmback\\_dev\\_t](#)

## Enumerations

- enum { [EV\\_NONE](#), [EV\\_NEWFE](#), [EV\\_STCHNG](#) }

## Functions

- void [tpmif\\_req\\_ready](#) ([tpmif\\_t](#) \*[tpmif](#))
- void [tpmdev\\_check\\_req](#) (void)
- void [tpmif\\_req\\_finished](#) ([tpmif\\_t](#) \*[tpmif](#))
- int [\\_\\_get\\_tpmif\\_index](#) (int st, int n, [domid\\_t](#) domid, unsigned int [handle](#))
- int [get\\_tpmif\\_index](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- [tpmif\\_t](#) \* [get\\_tpmif](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- int [remove\\_tpmif](#) ([tpmif\\_t](#) \*[tpmif](#))
- int [insert\\_tpmif](#) ([tpmif\\_t](#) \*[tpmif](#))
- int [tpmif\\_change\\_state](#) ([tpmif\\_t](#) \*[tpmif](#), enum xenbus\_state state)
- [tpmif\\_t](#) \* [\\_\\_init\\_tpmif](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- void [\\_\\_free\\_tpmif](#) ([tpmif\\_t](#) \*[tpmif](#))
- [tpmif\\_t](#) \* [new\\_tpmif](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- void [free\\_tpmif](#) ([tpmif\\_t](#) \*[tpmif](#))
- void [tpmback\\_handler](#) (evchn\_port\_t port, struct pt\_regs \*regs, void \*data)
- int [connect\\_fe](#) ([tpmif\\_t](#) \*[tpmif](#))
- void [handle\\_backend\\_event](#) (char \*evstr)
- char \* [tpmback\\_get\\_uuid](#) ([domid\\_t](#) domid, unsigned int [handle](#))
- void [tpmback\\_set\\_open\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_close\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_suspend\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [tpmback\\_set\\_resume\\_callback](#) (void(\*cb)([domid\\_t](#), unsigned int))
- void [event\\_thread](#) (void \*p)
- void [init\\_tpmback](#) (char \*\*exclusive\_uuids)
- void [shutdown\\_tpmback](#) (void)
- void [init\\_tpmcmd](#) ([tpmcmd\\_t](#) \*[tpmcmd](#), [domid\\_t](#) domid, unsigned int [handle](#), char \*uuid)
- [tpmcmd\\_t](#) \* [get\\_request](#) ([tpmif\\_t](#) \*[tpmif](#))
- void [send\\_response](#) ([tpmcmd\\_t](#) \*cmd, [tpmif\\_t](#) \*[tpmif](#))
- [tpmcmd\\_t](#) \* [tpmback\\_req\\_any](#) (void)
- [tpmcmd\\_t](#) \* [tpmback\\_req](#) ([domid\\_t](#) domid, unsigned int [handle](#))

- void `tpmback_resp` (`tpmcmd_t *tpmcmd`)
- int `tpmback_wait_for_frontend_connect` (`domid_t *domid`, unsigned int `*handle`)
- int `tpmback_num_frontends` (void)

## Variables

- enum { ... } `tpm_ev_enum`
- struct wait\_queue\_head `waitq`
- int `globalinit` = 0

## 4.41.1 Macro Definition Documentation

4.41.1.1 `#define DEF_ARRAY_SIZE 1`

4.41.1.2 `#define min( a, b ) (((a) < (b)) ? (a) : (b))`

4.41.1.3 `#define strtoul simple_strtoul`

4.41.1.4 `#define TPMBACK_DEBUG( fmt, ... )`

4.41.1.5 `#define TPMBACK_ERR( fmt, ... ) printk("Tpmback:Error " fmt, ##__VA_ARGS__)`

4.41.1.6 `#define TPMBACK_LOG( fmt, ... ) printk("Tpmback:Info " fmt, ##__VA_ARGS__)`

4.41.1.7 `#define TPMIF_CLOSED 1`

4.41.1.8 `#define TPMIF_REQ_READY 2`

## 4.41.2 Typedef Documentation

4.41.2.1 `typedef struct tpmback_dev tpmback_dev_t`

4.41.2.2 `typedef struct tpmif tpmif_t`

## 4.41.3 Enumeration Type Documentation

4.41.3.1 anonymous enum

### Enumerator

***EV\_NONE***

***EV\_NEWFE***

***EV\_STCHNG***

## 4.41.4 Function Documentation

4.41.4.1 `void __free_tpmif ( tpmif_t * tpmif )`

4.41.4.2 `int __get_tpmif_index ( int st, int n, domid_t domid, unsigned int handle )`

4.41.4.3 `tpmif_t* __init_tpmif ( domid_t domid, unsigned int handle )` `[inline]`

4.41.4.4 `int connect_fe ( tpmif_t * tpmif )`

Connect to frontend

4.41.4.5 void event\_thread ( void \* p )

4.41.4.6 void free\_tpmif ( tpmif\_t \* tpmif )

Removes tpmif from dev->tpmlist and frees it's memory usage

4.41.4.7 tpmcmd\_t\* get\_request ( tpmif\_t \* tpmif )

4.41.4.8 tpmif\_t\* get\_tpmif ( domid\_t domid, unsigned int handle )

Returns the tpmif domid/handle or NULL if none exists

4.41.4.9 int get\_tpmif\_index ( domid\_t domid, unsigned int handle )

Returns the array index of the tpmif domid/handle. Returns -1 if no such tpmif exists

4.41.4.10 void handle\_backend\_event ( char \* evstr )

4.41.4.11 void init\_tpmback ( char \*\* exclusive\_uuids )

Initialize the tpm backend driver - This is NULL terminated list of vtpm uuid strings. If this list is non-empty, then only frontend domains with vtpm uuid's matching entries in this list will be allowed to connect. Other connections will be immediatly closed. Set this argument to NULL to allow any vtpm to connect.

4.41.4.12 void init\_tpmcmd ( tpmcmd\_t \* tpmcmd, domid\_t domid, unsigned int handle, char \* uuid ) [inline]

4.41.4.13 int insert\_tpmif ( tpmif\_t \* tpmif )

Insert tpmif into dev->tpmlist. Returns 0 on success and non zero on error. It is an error to insert a tpmif with the same domid and handle number as something already in the list

4.41.4.14 tpmif\_t\* new\_tpmif ( domid\_t domid, unsigned int handle )

Creates a new tpm interface, adds it to the sorted array and returns it. returns NULL on error If the tpm interface already exists, it is returned

4.41.4.15 int remove\_tpmif ( tpmif\_t \* tpmif )

Remove the given tpmif. Returns 0 if it was removed, -1 if it was not removed

4.41.4.16 void send\_response ( tpmcmd\_t \* cmd, tpmif\_t \* tpmif )

4.41.4.17 void shutdown\_tpmback ( void )

Shutdown tpm backend driver

4.41.4.18 char\* tpmback\_get\_uuid ( domid\_t domid, unsigned int handle )

Returns the uuid of the specified frontend, NULL on error

4.41.4.19 `void tpmback_handler ( evchn_port_t port, struct pt_regs * regs, void * data )`

Event channel handler

4.41.4.20 `int tpmback_num_frontends ( void )`

returns the number of frontends connected

4.41.4.21 `tpmcmd_t* tpmback_req ( domid_t domid, unsigned int handle )`

Blocks until a tpm command from the frontend at domid/handle is sent. Returns NULL if domid/handle is not connected, tpmback is shutdown or shutting down, or if there is an error

4.41.4.22 `tpmcmd_t* tpmback_req_any ( void )`

Blocks until a tpm command is sent from any front end. Returns a pointer to the tpm command to handle. Do not try to free this pointer or the req buffer This function will return NULL if the tpm backend driver is shutdown or any other error occurs

4.41.4.23 `void tpmback_resp ( tpmcmd_t * tpmcmd )`

Send the response to the tpm command back to the frontend This function will free the tpmcmd object, but you must free the resp buffer yourself

4.41.4.24 `void tpmback_set_close_callback ( void(*)(domid_t, unsigned int) cb )`

Specify a function to call when a tpm device disconnects

4.41.4.25 `void tpmback_set_open_callback ( void(*)(domid_t, unsigned int) cb )`

Specify a function to call when a new tpm device connects

4.41.4.26 `void tpmback_set_resume_callback ( void(*)(domid_t, unsigned int) cb )`

4.41.4.27 `void tpmback_set_suspend_callback ( void(*)(domid_t, unsigned int) cb )`

4.41.4.28 `int tpmback_wait_for_frontend_connect ( domid_t * domid, unsigned int * handle )`

Waits for the first frontend to connect and then sets domid and handle appropriately. If one or more frontends are already connected, this will set domid and handle to one of them arbitrarily. The main use for this function is to wait until a single frontend connection has occurred. returns 0 on success, non-zero on failure

4.41.4.29 `void tpmdev_check_req ( void )` `[inline]`

4.41.4.30 `int tpmif_change_state ( tpmif_t * tpmif, enum xenbus_state state )`

Attempts to change the backend state in xenstore returns 0 on success and non-zero on error

4.41.4.31 void tpmif\_req\_finished ( tpmif\_t \* *tpmif* ) [inline]

4.41.4.32 void tpmif\_req\_ready ( tpmif\_t \* *tpmif* ) [inline]

## 4.41.5 Variable Documentation

4.41.5.1 int globalinit = 0

4.41.5.2 enum { ... } tpm\_ev\_enum

4.41.5.3 struct wait\_queue\_head waitq

## 4.42 xen/extras/mini-os/tpmfront.c File Reference

```
#include <mini-os/os.h>
#include <mini-os/xenbus.h>
#include <mini-os/xmalloc.h>
#include <mini-os/events.h>
#include <mini-os/wait.h>
#include <mini-os/gnttab.h>
#include <xen/io/xenbus.h>
#include <xen/io/tpmif.h>
#include <mini-os/tpmfront.h>
#include <fcntl.h>
```

## Macros

- #define TPMFRONT\_DEBUG(fmt,...)
- #define TPMFRONT\_ERR(fmt,...) printk("Tpmfront:Error " fmt, ##\_\_VA\_ARGS\_\_)
- #define TPMFRONT\_LOG(fmt,...) printk("Tpmfront:Info " fmt, ##\_\_VA\_ARGS\_\_)
- #define min(a, b) (((a) < (b)) ? (a) : (b))

## Functions

- void tpmfront\_handler (evchn\_port\_t port, struct pt\_regs \*regs, void \*data)
- struct tpmfront\_dev \* init\_tpmfront (const char \* \_nodename)
- void shutdown\_tpmfront (struct tpmfront\_dev \*dev)
- int tpmfront\_send (struct tpmfront\_dev \*dev, const uint8\_t \*msg, size\_t length)
- int tpmfront\_recv (struct tpmfront\_dev \*dev, uint8\_t \*\*msg, size\_t \*length)
- int tpmfront\_cmd (struct tpmfront\_dev \*dev, uint8\_t \*req, size\_t reqlen, uint8\_t \*\*resp, size\_t \*resplen)

### 4.42.1 Macro Definition Documentation

4.42.1.1 #define min( a, b )(((a) < (b)) ? (a) : (b))

4.42.1.2 #define TPMFRONT\_DEBUG( fmt, ... )

4.42.1.3 #define TPMFRONT\_ERR( fmt, ... ) printk("Tpmfront:Error " fmt, ##\_\_VA\_ARGS\_\_)

4.42.1.4 #define TPMFRONT\_LOG( fmt, ... ) printk("Tpmfront:Info " fmt, ##\_\_VA\_ARGS\_\_)

### 4.42.2 Function Documentation

- 4.42.2.1 `struct tpmfront_dev* init_tpmfront ( const char * _nodename )`
- 4.42.2.2 `void shutdown_tpmfront ( struct tpmfront_dev * dev )`
- 4.42.2.3 `int tpmfront_cmd ( struct tpmfront_dev * dev, uint8_t * req, size_t reqlen, uint8_t ** resp, size_t * resplen )`
- 4.42.2.4 `void tpmfront_handler ( evchn_port_t port, struct pt_regs * regs, void * data )`
- 4.42.2.5 `int tpmfront_recv ( struct tpmfront_dev * dev, uint8_t ** msg, size_t * length )`
- 4.42.2.6 `int tpmfront_send ( struct tpmfront_dev * dev, const uint8_t * msg, size_t length )`

## 4.43 xen/extras/mini-os/uuid.c File Reference

```
#include <inttypes.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <uuid.h>
```

### Functions

- void `uuid_create` (`uuid_t` \*uuid, uint32\_t \*status)
- void `uuid_to_string` (`uuid_t` \*uuid, char \*\*str, uint32\_t \*status)
- int `uuid_is_nil` (`uuid_t` \*uuid, unsigned int \*status)

### 4.43.1 Function Documentation

4.43.1.1 `void uuid_create ( uuid_t * uuid, uint32_t * status )`

just fill the struct with random values for now

4.43.1.2 `int uuid_is_nil ( uuid_t * uuid, unsigned int * status )`

4.43.1.3 `void uuid_to_string ( uuid_t * uuid, char ** str, uint32_t * status )`

convert the struct to a printable string

## 4.44 xen/extras/mini-os/vfs.c File Reference

## 4.45 xen/stubdom/domt/integrated/vfs.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <stdint.h>
#include <unistd.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <errno.h>
#include <mini-os/os.h>
#include <mini-os/xenbus.h>
#include <mini-os/events.h>
#include <mini-os/gnttab.h>
#include <vfs.h>
#include <vfs_hooks.h>
```

### Data Structures

- struct [fe\\_gref\\_map](#)

### Macros

- #define [INVALID\\_GREF](#)(ref) (ref >= [NR\\_GRANT\\_ENTRIES](#) || ref < [NR\\_RESERVED\\_ENTRIES](#))
- #define [NR\\_RESERVED\\_ENTRIES](#) 8
- #define [NR\\_GRANT\\_FRAMES](#) 4
- #define [NR\\_GRANT\\_ENTRIES](#) ([NR\\_GRANT\\_FRAMES](#) \* [PAGE\\_SIZE](#) / sizeof(grant\_entry\_t))
- #define [MIN](#)(a, b) (((a) < (b)) ? (a) : (b))

### Functions

- int [vfs\\_init](#) (void)
- int [vfs\\_command](#) (struct [vfs\\_cmd](#) \*cmd)
- int [vfs\\_open](#) (const char \*pathname, int flags, mode\_t mode)
- int [vfs\\_close](#) (int fd)
- int [vfs\\_stat](#) (const char \*path, struct stat \*buf)
- off\_t [vfs\\_lseek](#) (int fd, off\_t offset, int whence)
- ssize\_t [vfs\\_read](#) (int fd, void \*buf, size\_t nbyte)
- ssize\_t [vfs\\_write](#) (int fd, const void \*buf, size\_t nbyte)
- void [vfs\\_major\\_minor](#) (const char \*physpath, int \*major, int \*minor)
- void \* [vfs\\_mmap](#) (void \*addr, size\_t [length](#), int prot, int flags, int fd, off\_t offset)

### Variables

- struct [fe\\_gref\\_map](#) [\\_\\_attribute\\_\\_](#)

#### 4.45.1 Macro Definition Documentation

4.45.1.1 #define [INVALID\\_GREF](#)( [ref](#) ) (ref >= [NR\\_GRANT\\_ENTRIES](#) || ref < [NR\\_RESERVED\\_ENTRIES](#))

4.45.1.2 #define [MIN](#)( [a](#), [b](#) ) (((a) < (b)) ? (a) : (b))

4.45.1.3 `#define NR_GRANT_ENTRIES (NR_GRANT_FRAMES * PAGE_SIZE / sizeof(grant_entry_t))`

4.45.1.4 `#define NR_GRANT_FRAMES 4`

4.45.1.5 `#define NR_RESERVED_ENTRIES 8`

## 4.45.2 Function Documentation

4.45.2.1 `int vfs_close ( int fd )`

4.45.2.2 `int vfs_command ( struct vfs_cmd * cmd )`

4.45.2.3 `int vfs_init ( void )`

The init command has become deprecated. Maybe it can be useful to test if VFS is alive at boottime. It might also play a role in detection of xenvfsd restart/crashing; since then we need to invalidate all vfs file descriptors. Not a priority task, however.

4.45.2.4 `off_t vfs_lseek ( int fd, off_t offset, int whence )`

4.45.2.5 `void vfs_major_minor ( const char * physpath, int * major, int * minor )`

combines the major and minor calls into one

4.45.2.6 `void* vfs_mmap ( void * addr, size_t length, int prot, int flags, int fd, off_t offset )`

For now, mmap is synthetic. it is a TODO to build a native solution via grant tables. We probably want to share a writable `gref_map` with Dom0 then. Until then, this function does not sync back changes. Since this call is only used in the domainbuilder for reading in the images, priority for this fix is low currently.

4.45.2.7 `int vfs_open ( const char * pathname, int flags, mode_t mode )`

NOTE: `munmap`, `close`, are not hooked since minios default implementation suffices. NOTE: `newlib` implements for us: `fread()` `fwrite()` `fopen()` `fclose()`.

4.45.2.8 `ssize_t vfs_read ( int fd, void * buf, size_t nbyte )`

4.45.2.9 `int vfs_stat ( const char * path, struct stat * buf )`

4.45.2.10 `ssize_t vfs_write ( int fd, const void * buf, size_t nbyte )`

## 4.45.3 Variable Documentation

4.45.3.1 `struct fe_gref_map __attribute__`



## 4.46 xen/stubdom/domc/blkback.c File Reference

```
#include <stdint.h>
#include <mini-os/os.h>
#include <mini-os/xenbus.h>
#include <mini-os/events.h>
#include <errno.h>
#include <xen/io/blkif.h>
#include <xen/io/protocols.h>
#include <mini-os/gnttab.h>
#include <mini-os/xmalloc.h>
#include <time.h>
#include <mini-os/blkfront.h>
#include <mini-os/lib.h>
#include <fcntl.h>
#include <unistd.h>
#include "blkback.h"
#include "main.h"
#include "crypt.h"
```

### Macros

- `#define SECTOR_SIZE 512`
- `#define SECTS_PER_PAGE PAGE_SIZE / SECTOR_SIZE`
- `#define SECTS_IN_SEG(req, i)`
- `#define BLK_RING_SIZE __RING_SIZE((struct blkif_sring *)0, PAGE_SIZE)`
- `#define MAX_GNTMAP BLKIF_MAX_SEGMENTS_PER_REQUEST * BLK_RING_SIZE + 1`

### Enumerations

- enum `bridging` { `OLDNEW`, `NEWOLD` }

### Functions

- void `blkback_cb_back` (evtchn\_port\_t port, struct pt\_regs \*regs, void \*data)
- struct `blkback_dev` \* `derive_blkback` (char \*\_nodename, domc\_info\_t \*domc\_info, struct `blkfront_dev` \*f\_dev, struct `blkfront_info` \*f\_dev\_info)
- int `init_blkback` (struct `blkback_dev` \*b\_dev)
- void `shutdown_blkback` (struct `blkback_dev` \*b\_dev)
- void `blkback_monitor` (struct `blkback_dev` \*b\_dev)
- struct `blkback_aiocb` \* `blkback_build_fwd` (struct `blkback_dev` \*b\_dev, blkif\_request\_t \*req)
- int `blkback_aio_poll` (struct `blkback_dev` \*b\_dev)

#### 4.46.1 Macro Definition Documentation

4.46.1.1 `#define BLK_RING_SIZE __RING_SIZE((struct blkif_sring *)0, PAGE_SIZE)`

4.46.1.2 `#define MAX_GNTMAP BLKIF_MAX_SEGMENTS_PER_REQUEST * BLK_RING_SIZE + 1`

the one slot extra is for the sring page

#### 4.46.1.3 #define SECTOR\_SIZE 512

The assumptions about IRQs and pre-emptiveness for the blkfront hold here as well. No wait queues are used, since we cannot trap multiple threads in our functions here. Our design assumes a single thread which never leaves `_monitor()`.

#### 4.46.1.4 #define SECTS\_IN\_SEG( req, i )

**Value:**

```
(1+(req->seg[i].last_sect - \
    req->seg[i].first_sect))
```

#### 4.46.1.5 #define SECTS\_PER\_PAGE PAGE\_SIZE / SECTOR\_SIZE

### 4.46.2 Enumeration Type Documentation

#### 4.46.2.1 enum bridging

Enumerator

***OLDNEW***

***NEWOLD***

### 4.46.3 Function Documentation

#### 4.46.3.1 int blkback\_aio\_poll ( struct blkback\_dev \* b\_dev )

#### 4.46.3.2 struct blkback\_aiocb\* blkback\_build\_fwd ( struct blkback\_dev \* b\_dev, blkif\_request\_t \* req )

Builds an aio to forward from blkback to blkfront

#### 4.46.3.3 void blkback\_cb\_back ( evtchn\_port\_t port, struct pt\_regs \* regs, void \* data )

#### 4.46.3.4 void blkback\_monitor ( struct blkback\_dev \* b\_dev )

We placed the `blkfront_sync()` calls in a separate thread and not after the `blkback_poll()` here. If we'd wait with writing new requests until `blkfront_sync()` returns, we would not be making optimal use of the ring buffer with Dom0.

#### 4.46.3.5 struct blkback\_dev\* derive\_blkback ( char \* \_nodename, domc\_info\_t \* domc\_info, struct blkfront\_dev \* f\_dev, struct blkfront\_info \* f\_dev\_info )

#### 4.46.3.6 int init\_blkback ( struct blkback\_dev \* b\_dev )

#### 4.46.3.7 void shutdown\_blkback ( struct blkback\_dev \* b\_dev )

## 4.47 xen/stubdom/domc/blkback.h File Reference

```
#include <mini-os/blkfront.h>
#include "main.h"
```

## Data Structures

- struct [blkback\\_info](#)
- struct [blkback\\_dev](#)
- struct [blkback\\_aiocb](#)

## Functions

- struct [blkback\\_dev](#) \* [derive\\_blkback](#) (char \* \_nodename, [domc\\_info\\_t](#) \* [domc\\_info](#), struct [blkfront\\_dev](#) \* [f\\_dev](#), struct [blkfront\\_info](#) \* [f\\_dev\\_info](#))
- int [init\\_blkback](#) (struct [blkback\\_dev](#) \* [b\\_dev](#))
- int [blkback\\_aio\\_poll](#) (struct [blkback\\_dev](#) \* [b\\_dev](#))
- void [blkback\\_monitor](#) (struct [blkback\\_dev](#) \* [b\\_dev](#))
- void [shutdown\\_blkback](#) (struct [blkback\\_dev](#) \* [b\\_dev](#))

### 4.47.1 Function Documentation

4.47.1.1 int [blkback\\_aio\\_poll](#) ( struct [blkback\\_dev](#) \* [b\\_dev](#) )

4.47.1.2 void [blkback\\_monitor](#) ( struct [blkback\\_dev](#) \* [b\\_dev](#) )

We placed the [blkfront\\_sync\(\)](#) calls in a separate thread and not after the [blkback\\_poll\(\)](#) here. If we'd wait with writing new requests until [blkfront\\_sync\(\)](#) returns, we would not be making optimal use of the ring buffer with Dom0.

4.47.1.3 struct [blkback\\_dev](#)\* [derive\\_blkback](#) ( char \* [\\_nodename](#), [domc\\_info\\_t](#) \* [domc\\_info](#), struct [blkfront\\_dev](#) \* [f\\_dev](#), struct [blkfront\\_info](#) \* [f\\_dev\\_info](#) )

4.47.1.4 int [init\\_blkback](#) ( struct [blkback\\_dev](#) \* [b\\_dev](#) )

4.47.1.5 void [shutdown\\_blkback](#) ( struct [blkback\\_dev](#) \* [b\\_dev](#) )

## 4.48 xen/stubdom/domc/crypt.c File Reference

```
#include <string.h>
#include "rijndael-api-fst.h"
#include "sha2.h"
#include "main.h"
#include "crypt.h"
```

## Macros

- #define [SECTOR\\_SIZE](#) 512

## Enumerations

- enum [crypt\\_dir](#) { [CRYPT\\_EN](#), [CRYPT\\_DEC](#) }

## Functions

- int [startup\\_crypt](#) (char \*key)
- int [encrypt\\_sectors](#) (unsigned int sector, [BYTE](#) \*in, [BYTE](#) \*out, size\_t len)
- int [decrypt\\_sectors](#) (unsigned int sector, [BYTE](#) \*in, [BYTE](#) \*out, size\_t len)
- void [shutdown\\_crypt](#) (void)
- void [test\\_crypt](#) (void)

## Variables

- [keyInstance](#) [keyInst\\_enc](#)
- [keyInstance](#) [keyInst\\_dec](#)
- [keyInstance](#) [keyInst\\_iv](#)
- [cipherInstance](#) [cipherInst\\_iv](#)

### 4.48.1 Macro Definition Documentation

4.48.1.1 `#define SECTOR_SIZE 512`

### 4.48.2 Enumeration Type Documentation

4.48.2.1 `enum crypt_dir`

#### Enumerator

***CRYPT\_EN***  
***CRYPT\_DEC***

### 4.48.3 Function Documentation

4.48.3.1 `int decrypt_sectors ( unsigned int sector, BYTE * in, BYTE * out, size_t len )`

4.48.3.2 `int encrypt_sectors ( unsigned int sector, BYTE * in, BYTE * out, size_t len )`

4.48.3.3 `void shutdown_crypt ( void )`

4.48.3.4 `int startup_crypt ( char * key )`

4.48.3.5 `void test_crypt ( void )`

### 4.48.4 Variable Documentation

4.48.4.1 `cipherInstance cipherInst_iv`

4.48.4.2 `keyInstance keyInst_dec`

4.48.4.3 `keyInstance keyInst_enc`

4.48.4.4 `keyInstance keyInst_iv`

## 4.49 xen/stubdom/domc/crypt.h File Reference

```
#include "rijndael-api-fst.h"
```

## Functions

- int `startup_crypt` (char \*key)
- int `encrypt_sectors` (unsigned int sector, BYTE \*in, BYTE \*out, size\_t len)
- int `decrypt_sectors` (unsigned int sector, BYTE \*in, BYTE \*out, size\_t len)
- void `shutdown_crypt` (void)
- void `test_crypt` (void)

### 4.49.1 Function Documentation

4.49.1.1 int `decrypt_sectors` ( unsigned int *sector*, BYTE \* *in*, BYTE \* *out*, size\_t *len* )

4.49.1.2 int `encrypt_sectors` ( unsigned int *sector*, BYTE \* *in*, BYTE \* *out*, size\_t *len* )

4.49.1.3 void `shutdown_crypt` ( void )

4.49.1.4 int `startup_crypt` ( char \* *key* )

4.49.1.5 void `test_crypt` ( void )

## 4.50 xen/stubdom/domc/crypto/rijndael-alg-fst.c File Reference

```
#include <assert.h>
#include <stdlib.h>
#include "rijndael-alg-fst.h"
```

## Macros

- #define `FULL_UNROLL`
- #define `SWAP(x)` (`_lrotl(x, 8) & 0x00ff00ff | _lrotr(x, 8) & 0xff00ff00`)
- #define `GETU32(pt)` (`((u32)(pt)[0] << 24) ^ ((u32)(pt)[1] << 16) ^ ((u32)(pt)[2] << 8) ^ ((u32)(pt)[3])`)
- #define `PUTU32(ct, st)` { `(ct)[0] = (u8)((st) >> 24); (ct)[1] = (u8)((st) >> 16); (ct)[2] = (u8)((st) >> 8); (ct)[3] = (u8)(st); }`

## Functions

- int `rijndaelKeySetupEnc` (u32 rk[], const u8 cipherKey[], int keyBits)
- int `rijndaelKeySetupDec` (u32 rk[], const u8 cipherKey[], int keyBits)
- void `rijndaelEncrypt` (const u32 rk[], int Nr, const u8 pt[16], u8 ct[16])
- void `rijndaelDecrypt` (const u32 rk[], int Nr, const u8 ct[16], u8 pt[16])

### 4.50.1 Macro Definition Documentation

4.50.1.1 #define `FULL_UNROLL`

rijndael-alg-fst.c

#### Version

3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

**Author**

Vincent Rijmen [vincent.rijmen@esat.kuleuven.ac.be](mailto:vincent.rijmen@esat.kuleuven.ac.be)  
 Antoon Bosselaers [antoon.bosselaers@esat.kuleuven.ac.be](mailto:antoon.bosselaers@esat.kuleuven.ac.be)  
 Paulo Barreto [paulo.barreto@terra.com.br](mailto:paulo.barreto@terra.com.br)

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
4.50.1.2 #define GETU32( pt ) (((u32)(pt)[0] << 24) ^ ((u32)(pt)[1] << 16) ^ ((u32)(pt)[2] << 8) ^ ((u32)(pt)[3]))
```

```
4.50.1.3 #define PUTU32( ct, st ) { (ct)[0] = (u8)((st) >> 24); (ct)[1] = (u8)((st) >> 16); (ct)[2] = (u8)((st) >> 8); (ct)[3] = (u8)(st); }
```

```
4.50.1.4 #define SWAP( x ) (_lrotl(x, 8) & 0x00ff00ff | _lrotr(x, 8) & 0xff00ff00)
```

**4.50.2 Function Documentation**

```
4.50.2.1 void rijndaelDecrypt ( const u32 rk[], int Nr, const u8 ct[16], u8 pt[16] )
```

```
4.50.2.2 void rijndaelEncrypt ( const u32 rk[], int Nr, const u8 pt[16], u8 ct[16] )
```

```
4.50.2.3 int rijndaelKeySetupDec ( u32 rk[], const u8 cipherKey[], int keyBits )
```

Expand the cipher key into the decryption key schedule.

**Returns**

the number of rounds for the given cipher key size.

```
4.50.2.4 int rijndaelKeySetupEnc ( u32 rk[], const u8 cipherKey[], int keyBits )
```

Expand the cipher key into the encryption key schedule.

**Returns**

the number of rounds for the given cipher key size.

**4.51 xen/stubdom/domc/crypto/rijndael-alg-fst.h File Reference****Macros**

- #define **MAXKC** (256/32)
- #define **MAXKB** (256/8)
- #define **MAXNR** 14

## Typedefs

- typedef unsigned char [u8](#)
- typedef unsigned short [u16](#)
- typedef unsigned int [u32](#)

## Functions

- int [rijndaelKeySetupEnc](#) ([u32](#) rk[], const [u8](#) cipherKey[], int keyBits)
- int [rijndaelKeySetupDec](#) ([u32](#) rk[], const [u8](#) cipherKey[], int keyBits)
- void [rijndaelEncrypt](#) (const [u32](#) rk[], int Nr, const [u8](#) pt[16], [u8](#) ct[16])
- void [rijndaelDecrypt](#) (const [u32](#) rk[], int Nr, const [u8](#) ct[16], [u8](#) pt[16])

### 4.51.1 Macro Definition Documentation

4.51.1.1 `#define MAXKB (256/8)`

4.51.1.2 `#define MAXKC (256/32)`

[rijndael-alg-fst.h](#)

#### Version

3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

#### Author

Vincent Rijmen [vincent.rijmen@esat.kuleuven.ac.be](mailto:vincent.rijmen@esat.kuleuven.ac.be)  
 Antoon Bosselaers [antoon.bosselaers@esat.kuleuven.ac.be](mailto:antoon.bosselaers@esat.kuleuven.ac.be)  
 Paulo Barreto [paulo.barreto@terra.com.br](mailto:paulo.barreto@terra.com.br)

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.51.1.3 `#define MAXNR 14`

### 4.51.2 Typedef Documentation

4.51.2.1 typedef unsigned short [u16](#)

4.51.2.2 typedef unsigned int [u32](#)

4.51.2.3 typedef unsigned char [u8](#)

### 4.51.3 Function Documentation

4.51.3.1 void rijndaelDecrypt ( const u32 *rk*[], int *Nr*, const u8 *ct*[16], u8 *pt*[16] )

4.51.3.2 void rijndaelEncrypt ( const u32 *rk*[], int *Nr*, const u8 *pt*[16], u8 *ct*[16] )

4.51.3.3 int rijndaelKeySetupDec ( u32 *rk*[], const u8 *cipherKey*[], int *keyBits* )

Expand the cipher key into the decryption key schedule.

#### Returns

the number of rounds for the given cipher key size.

4.51.3.4 int rijndaelKeySetupEnc ( u32 *rk*[], const u8 *cipherKey*[], int *keyBits* )

Expand the cipher key into the encryption key schedule.

#### Returns

the number of rounds for the given cipher key size.

## 4.52 xen/stubdom/domc/crypto/rijndael-api-fst.c File Reference

```
#include <assert.h>
#include <stdlib.h>
#include <string.h>
#include "rijndael-alg-fst.h"
#include "rijndael-api-fst.h"
```

### Functions

- int [makeKey](#) ([keyInstance](#) \*key, [BYTE](#) direction, int keyLen, char \*keyMaterial)
- int [cipherInit](#) ([cipherInstance](#) \*cipher, [BYTE](#) mode, [BYTE](#) \*IV)
- int [blockEncrypt](#) ([cipherInstance](#) \*cipher, [keyInstance](#) \*key, [BYTE](#) \*input, int inputLen, [BYTE](#) \*outBuffer)
- int [padEncrypt](#) ([cipherInstance](#) \*cipher, [keyInstance](#) \*key, [BYTE](#) \*input, int inputOctets, [BYTE](#) \*outBuffer)
- int [blockDecrypt](#) ([cipherInstance](#) \*cipher, [keyInstance](#) \*key, [BYTE](#) \*input, int inputLen, [BYTE](#) \*outBuffer)
- int [padDecrypt](#) ([cipherInstance](#) \*cipher, [keyInstance](#) \*key, [BYTE](#) \*input, int inputOctets, [BYTE](#) \*outBuffer)

### 4.52.1 Function Documentation

4.52.1.1 int blockDecrypt ( [cipherInstance](#) \* *cipher*, [keyInstance](#) \* *key*, [BYTE](#) \* *input*, int *inputLen*, [BYTE](#) \* *outBuffer* )

4.52.1.2 int blockEncrypt ( [cipherInstance](#) \* *cipher*, [keyInstance](#) \* *key*, [BYTE](#) \* *input*, int *inputLen*, [BYTE](#) \* *outBuffer* )

4.52.1.3 int cipherInit ( [cipherInstance](#) \* *cipher*, [BYTE](#) *mode*, [BYTE](#) \* *IV* )

4.52.1.4 int makeKey ( [keyInstance](#) \* *key*, [BYTE](#) *direction*, int *keyLen*, char \* *keyMaterial* )

rijndael-api-fst.c



**Version**

2.9 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

**Author**Vincent Rijmen [vincent.rijmen@esat.kuleuven.ac.be](mailto:vincent.rijmen@esat.kuleuven.ac.be)Antoon Bosselaers [antoon.bosselaers@esat.kuleuven.ac.be](mailto:antoon.bosselaers@esat.kuleuven.ac.be)Paulo Barreto [paulo.barreto@terra.com.br](mailto:paulo.barreto@terra.com.br)

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**Acknowledgements:**

We are deeply indebted to the following people for their bug reports, fixes, and improvement suggestions to this implementation. Though we tried to list all contributions, we apologise in advance for any missing reference.

Andrew Bales [Andrew.Bales@Honeywell.com](mailto:Andrew.Bales@Honeywell.com) Markus Friedl [markus.friedl@informatik.uni-erlangen.de](mailto:markus.friedl@informatik.uni-erlangen.de) John Skodon [skodonj@webquill.com](mailto:skodonj@webquill.com)

4.52.1.5 `int padDecrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputOctets, BYTE * outBuffer )`

4.52.1.6 `int padEncrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputOctets, BYTE * outBuffer )`

Encrypt data partitioned in octets, using RFC 2040-like padding.

**Parameters**

<i>input</i>	data to be encrypted (octet sequence)
<i>inputOctets</i>	input length in octets (not bits)
<i>outBuffer</i>	encrypted output data

**Returns**

length in octets (not bits) of the encrypted output buffer.

**4.53 xen/stubdom/domc/crypto/rijndael-api-fst.h File Reference**

```
#include <stdio.h>
#include "rijndael-alg-fst.h"
```

**Data Structures**

- struct [keyInstance](#)
- struct [cipherInstance](#)

## Macros

- `#define DIR_ENCRYPT 0 /* Are we encrypting? */`
- `#define DIR_DECRYPT 1 /* Are we decrypting? */`
- `#define MODE_ECB 1 /* Are we ciphering in ECB mode? */`
- `#define MODE_CBC 2 /* Are we ciphering in CBC mode? */`
- `#define MODE_CFB1 3 /* Are we ciphering in 1-bit CFB mode? */`
- `#define TRUE 1`
- `#define FALSE 0`
- `#define BITS_PER_BLOCK 128 /* Default number of bits in a cipher block */`
- `#define BAD_KEY_DIR -1 /* Key direction is invalid, e.g., unknown value */`
- `#define BAD_KEY_MAT -2 /* Key material not of correct length */`
- `#define BAD_KEY_INSTANCE -3 /* Key passed is not valid */`
- `#define BAD_CIPHER_MODE -4 /* Params struct passed to cipherInit invalid */`
- `#define BAD_CIPHER_STATE -5 /* Cipher in wrong state (e.g., not initialized) */`
- `#define BAD_BLOCK_LENGTH -6`
- `#define BAD_CIPHER_INSTANCE -7`
- `#define BAD_DATA -8 /* Data contents are invalid, e.g., invalid padding */`
- `#define BAD_OTHER -9 /* Unknown error */`
- `#define MAX_KEY_SIZE 64 /* # of ASCII char's needed to represent a key */`
- `#define MAX_IV_SIZE 16 /* # bytes needed to represent an IV */`

## Typedefs

- `typedef unsigned char BYTE`

## Functions

- `int makeKey (keyInstance *key, BYTE direction, int keyLen, char *keyMaterial)`
- `int cipherInit (cipherInstance *cipher, BYTE mode, BYTE *IV)`
- `int blockEncrypt (cipherInstance *cipher, keyInstance *key, BYTE *input, int inputLen, BYTE *outBuffer)`
- `int padEncrypt (cipherInstance *cipher, keyInstance *key, BYTE *input, int inputOctets, BYTE *outBuffer)`
- `int blockDecrypt (cipherInstance *cipher, keyInstance *key, BYTE *input, int inputLen, BYTE *outBuffer)`
- `int padDecrypt (cipherInstance *cipher, keyInstance *key, BYTE *input, int inputOctets, BYTE *outBuffer)`

### 4.53.1 Macro Definition Documentation

4.53.1.1 `#define BAD_BLOCK_LENGTH -6`

4.53.1.2 `#define BAD_CIPHER_INSTANCE -7`

4.53.1.3 `#define BAD_CIPHER_MODE -4 /* Params struct passed to cipherInit invalid */`

4.53.1.4 `#define BAD_CIPHER_STATE -5 /* Cipher in wrong state (e.g., not initialized) */`

4.53.1.5 `#define BAD_DATA -8 /* Data contents are invalid, e.g., invalid padding */`

4.53.1.6 `#define BAD_KEY_DIR -1 /* Key direction is invalid, e.g., unknown value */`

4.53.1.7 `#define BAD_KEY_INSTANCE -3 /* Key passed is not valid */`

4.53.1.8 `#define BAD_KEY_MAT -2 /* Key material not of correct length */`

4.53.1.9 `#define BAD_OTHER -9 /* Unknown error */`

4.53.1.10 `#define BITSPERBLOCK 128 /* Default number of bits in a cipher block */`

4.53.1.11 `#define DIR_DECRYPT 1 /* Are we decrypting? */`

4.53.1.12 `#define DIR_ENCRYPT 0 /* Are we encrypting? */`

[rijndael-api-fst.h](#)

Version

2.9 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

Author

Vincent Rijmen [vincent.rijmen@esat.kuleuven.ac.be](mailto:vincent.rijmen@esat.kuleuven.ac.be)

Antoon Bosselaers [antoon.bosselaers@esat.kuleuven.ac.be](mailto:antoon.bosselaers@esat.kuleuven.ac.be)

Paulo Barreto [paulo.barreto@terra.com.br](mailto:paulo.barreto@terra.com.br)

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Acknowledgements:

We are deeply indebted to the following people for their bug reports, fixes, and improvement suggestions to this implementation. Though we tried to list all contributions, we apologise in advance for any missing reference.

Andrew Bales [Andrew.Bales@Honeywell.com](mailto:Andrew.Bales@Honeywell.com) Markus Friedl [markus.friedl@informatik.uni-erlangen.de](mailto:markus.friedl@informatik.uni-erlangen.de) John Skodon [skodonj@webquill.com](mailto:skodonj@webquill.com)

4.53.1.13 `#define FALSE 0`

4.53.1.14 `#define MAX_IV_SIZE 16 /* # bytes needed to represent an IV */`

4.53.1.15 `#define MAX_KEY_SIZE 64 /* # of ASCII char's needed to represent a key */`

4.53.1.16 `#define MODE_CBC 2 /* Are we ciphering in CBC mode? */`

4.53.1.17 `#define MODE_CFB1 3 /* Are we ciphering in 1-bit CFB mode? */`

4.53.1.18 `#define MODE_ECB 1 /* Are we ciphering in ECB mode? */`

4.53.1.19 `#define TRUE 1`

## 4.53.2 Typedef Documentation

4.53.2.1 `typedef unsigned char BYTE`

### 4.53.3 Function Documentation

4.53.3.1 `int blockDecrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputLen, BYTE * outBuffer )`

4.53.3.2 `int blockEncrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputLen, BYTE * outBuffer )`

4.53.3.3 `int cipherInit ( cipherInstance * cipher, BYTE mode, BYTE * IV )`

4.53.3.4 `int makeKey ( keyInstance * key, BYTE direction, int keyLen, char * keyMaterial )`

rijndael-api-fst.c

#### Version

2.9 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

#### Author

Vincent Rijmen [vincent.rijmen@esat.kuleuven.ac.be](mailto:vincent.rijmen@esat.kuleuven.ac.be)

Antoon Bosselaers [antoon.bosselaers@esat.kuleuven.ac.be](mailto:antoon.bosselaers@esat.kuleuven.ac.be)

Paulo Barreto [paulo.barreto@terra.com.br](mailto:paulo.barreto@terra.com.br)

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

#### Acknowledgements:

We are deeply indebted to the following people for their bug reports, fixes, and improvement suggestions to this implementation. Though we tried to list all contributions, we apologise in advance for any missing reference.

Andrew Bales [Andrew.Bales@Honeywell.com](mailto:Andrew.Bales@Honeywell.com) Markus Friedl [markus.friedl@informatik.uni-erlangen.de](mailto:markus.friedl@informatik.uni-erlangen.de) John Skodon [skodonj@webquill.com](mailto:skodonj@webquill.com)

4.53.3.5 `int padDecrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputOctets, BYTE * outBuffer )`

4.53.3.6 `int padEncrypt ( cipherInstance * cipher, keyInstance * key, BYTE * input, int inputOctets, BYTE * outBuffer )`

Encrypt data partitioned in octets, using RFC 2040-like padding.

#### Parameters

<i>input</i>	data to be encrypted (octet sequence)
<i>inputOctets</i>	input length in octets (not bits)

<i>outBuffer</i>	encrypted output data
------------------	-----------------------

### Returns

length in octets (not bits) of the encrypted output buffer.

## 4.54 xen/stubdom/domc/crypto/sha2.c File Reference

```
#include <string.h>
#include <assert.h>
#include "sha2.h"
```

### Macros

- `#define SHA256_SHORT_BLOCK_LENGTH (SHA256_BLOCK_LENGTH - 8)`
- `#define SHA384_SHORT_BLOCK_LENGTH (SHA384_BLOCK_LENGTH - 16)`
- `#define SHA512_SHORT_BLOCK_LENGTH (SHA512_BLOCK_LENGTH - 16)`
- `#define REVERSE32(w, x)`
- `#define REVERSE64(w, x)`
- `#define ADDINC128(w, n)`
- `#define SHA2_USE_MEMSET_MEMCPY 1`
- `#define MEMSET_BZERO(p, l) memset((p), 0, (l))`
- `#define MEMCPY_BCOPY(d, s, l) memcpy((d), (s), (l))`
- `#define R(b, x) ((x) >> (b))`
- `#define S32(b, x) (((x) >> (b)) | ((x) << (32 - (b))))`
- `#define S64(b, x) (((x) >> (b)) | ((x) << (64 - (b))))`
- `#define Ch(x, y, z) (((x) & (y)) ^ ((~(x)) & (z)))`
- `#define Maj(x, y, z) (((x) & (y)) ^ ((x) & (z)) ^ ((y) & (z)))`
- `#define Sigma0_256(x) (S32(2, (x)) ^ S32(13, (x)) ^ S32(22, (x)))`
- `#define Sigma1_256(x) (S32(6, (x)) ^ S32(11, (x)) ^ S32(25, (x)))`
- `#define sigma0_256(x) (S32(7, (x)) ^ S32(18, (x)) ^ R(3, (x)))`
- `#define sigma1_256(x) (S32(17, (x)) ^ S32(19, (x)) ^ R(10, (x)))`
- `#define Sigma0_512(x) (S64(28, (x)) ^ S64(34, (x)) ^ S64(39, (x)))`
- `#define Sigma1_512(x) (S64(14, (x)) ^ S64(18, (x)) ^ S64(41, (x)))`
- `#define sigma0_512(x) (S64( 1, (x)) ^ S64( 8, (x)) ^ R( 7, (x)))`
- `#define sigma1_512(x) (S64(19, (x)) ^ S64(61, (x)) ^ R( 6, (x)))`

### Typedefs

- `typedef uint8_t sha2_byte`
- `typedef uint32_t sha2_word32`
- `typedef uint64_t sha2_word64`

### Functions

- `void SHA512_Last (SHA512_CTX *)`
- `void SHA256_Transform (SHA256_CTX *, const sha2_word32 *)`
- `void SHA512_Transform (SHA512_CTX *, const sha2_word64 *)`
- `void SHA256_Init (SHA256_CTX *context)`
- `void SHA256_Update (SHA256_CTX *context, const sha2_byte *data, size_t len)`
- `void SHA256_Final (sha2_byte digest[], SHA256_CTX *context)`

- char \* [SHA256\\_End](#) (SHA256\_CTX \*context, char [buffer](#)[])
- char \* [SHA256\\_Data](#) (const [sha2\\_byte](#) \*data, size\_t len, char digest[SHA256\_DIGEST\_STRING\_LENGTH])
- void [SHA512\\_Init](#) (SHA512\_CTX \*context)
- void [SHA512\\_Update](#) (SHA512\_CTX \*context, const [sha2\\_byte](#) \*data, size\_t len)
- void [SHA512\\_Final](#) ([sha2\\_byte](#) digest[], SHA512\_CTX \*context)
- char \* [SHA512\\_End](#) (SHA512\_CTX \*context, char [buffer](#)[])
- char \* [SHA512\\_Data](#) (const [sha2\\_byte](#) \*data, size\_t len, char digest[SHA512\_DIGEST\_STRING\_LENGTH])
- void [SHA384\\_Init](#) (SHA384\_CTX \*context)
- void [SHA384\\_Update](#) (SHA384\_CTX \*context, const [sha2\\_byte](#) \*data, size\_t len)
- void [SHA384\\_Final](#) ([sha2\\_byte](#) digest[], SHA384\_CTX \*context)
- char \* [SHA384\\_End](#) (SHA384\_CTX \*context, char [buffer](#)[])
- char \* [SHA384\\_Data](#) (const [sha2\\_byte](#) \*data, size\_t len, char digest[SHA384\_DIGEST\_STRING\_LENGTH])

## 4.54.1 Macro Definition Documentation

### 4.54.1.1 #define ADDINC128( w, n )

**Value:**

```
{ \
    (w)[0] += (sha2_word64)(n); \
    if ((w)[0] < (n)) { \
        (w)[1]++; \
    } \
}
```

### 4.54.1.2 #define Ch( x, y, z )(((x) & (y)) ^ ((~(x)) & (z)))

### 4.54.1.3 #define Maj( x, y, z )(((x) & (y)) ^ ((x) & (z)) ^ ((y) & (z)))

### 4.54.1.4 #define MEMCPY\_BCOPY( d, s, l ) memcpy((d),(s),(l))

### 4.54.1.5 #define MEMSET\_BZERO( p, l ) memset((p),0,(l))

### 4.54.1.6 #define R( b, x )((x) >> (b))

### 4.54.1.7 #define REVERSE32( w, x )

**Value:**

```
{ \
    sha2_word32 tmp = (w); \
    tmp = (tmp >> 16) | (tmp << 16); \
    (x) = ((tmp & 0xff00ff00UL) >> 8) | ((tmp & 0x00ff00ffUL) << 8); \
}
```

### 4.54.1.8 #define REVERSE64( w, x )

**Value:**

```
{ \
    sha2_word64 tmp = (w); \
    tmp = (tmp >> 32) | (tmp << 32); \
    tmp = ((tmp & 0xff00ff00ff00ff00ULL) >> 8) | \
        ((tmp & 0x00ff00ff00ff00ffULL) << 8); \
    (x) = ((tmp & 0xffff0000ffff0000ULL) >> 16) | \
        ((tmp & 0x0000ffff0000ffffULL) << 16); \
}
```

```

4.54.1.9  #define S32( b, x ) (((x) >> (b)) | ((x) << (32 - (b))))

4.54.1.10 #define S64( b, x ) (((x) >> (b)) | ((x) << (64 - (b))))

4.54.1.11 #define SHA256_SHORT_BLOCK_LENGTH (SHA256_BLOCK_LENGTH - 8)

4.54.1.12 #define SHA2_USE_MEMSET_MEMCPY 1

4.54.1.13 #define SHA384_SHORT_BLOCK_LENGTH (SHA384_BLOCK_LENGTH - 16)

4.54.1.14 #define SHA512_SHORT_BLOCK_LENGTH (SHA512_BLOCK_LENGTH - 16)

4.54.1.15 #define Sigma0_256( x ) (S32(2, (x)) ^ S32(13, (x)) ^ S32(22, (x)))

4.54.1.16 #define sigma0_256( x ) (S32(7, (x)) ^ S32(18, (x)) ^ R(3, (x)))

4.54.1.17 #define Sigma0_512( x ) (S64(28, (x)) ^ S64(34, (x)) ^ S64(39, (x)))

4.54.1.18 #define sigma0_512( x ) (S64( 1, (x)) ^ S64( 8, (x)) ^ R( 7, (x)))

4.54.1.19 #define Sigma1_256( x ) (S32(6, (x)) ^ S32(11, (x)) ^ S32(25, (x)))

4.54.1.20 #define sigma1_256( x ) (S32(17, (x)) ^ S32(19, (x)) ^ R(10, (x)))

4.54.1.21 #define Sigma1_512( x ) (S64(14, (x)) ^ S64(18, (x)) ^ S64(41, (x)))

4.54.1.22 #define sigma1_512( x ) (S64(19, (x)) ^ S64(61, (x)) ^ R( 6, (x)))

```

#### 4.54.2 Typedef Documentation

```

4.54.2.1 typedef uint8_t sha2_byte

4.54.2.2 typedef uint32_t sha2_word32

4.54.2.3 typedef uint64_t sha2_word64

```

#### 4.54.3 Function Documentation

```

4.54.3.1 char* SHA256_Data ( const sha2_byte * data, size_t len, char digest[SHA256_DIGEST_STRING_LENGTH] )

4.54.3.2 char* SHA256_End ( SHA256_CTX * context, char buffer[] )

4.54.3.3 void SHA256_Final ( sha2_byte digest[], SHA256_CTX * context )

4.54.3.4 void SHA256_Init ( SHA256_CTX * context )

4.54.3.5 void SHA256_Transform ( SHA256_CTX * context, const sha2_word32 * data )

4.54.3.6 void SHA256_Update ( SHA256_CTX * context, const sha2_byte * data, size_t len )

4.54.3.7 char* SHA384_Data ( const sha2_byte * data, size_t len, char digest[SHA384_DIGEST_STRING_LENGTH] )

4.54.3.8 char* SHA384_End ( SHA384_CTX * context, char buffer[] )

4.54.3.9 void SHA384_Final ( sha2_byte digest[], SHA384_CTX * context )

```

- 4.54.3.10 void SHA384\_Init ( SHA384\_CTX \* context )
- 4.54.3.11 void SHA384\_Update ( SHA384\_CTX \* context, const sha2\_byte \* data, size\_t len )
- 4.54.3.12 char\* SHA512\_Data ( const sha2\_byte \* data, size\_t len, char digest[SHA512\_DIGEST\_STRING\_LENGTH] )
- 4.54.3.13 char\* SHA512\_End ( SHA512\_CTX \* context, char buffer[] )
- 4.54.3.14 void SHA512\_Final ( sha2\_byte digest[], SHA512\_CTX \* context )
- 4.54.3.15 void SHA512\_Init ( SHA512\_CTX \* context )
- 4.54.3.16 void SHA512\_Last ( SHA512\_CTX \* context )
- 4.54.3.17 void SHA512\_Transform ( SHA512\_CTX \* context, const sha2\_word64 \* data )
- 4.54.3.18 void SHA512\_Update ( SHA512\_CTX \* context, const sha2\_byte \* data, size\_t len )

## 4.55 xen/stubdom/domc/crypto/sha2.h File Reference

```
#include <sys/types.h>
#include <inttypes.h>
```

### Data Structures

- struct [\\_SHA256\\_CTX](#)
- struct [\\_SHA512\\_CTX](#)

### Macros

- #define [SHA2\\_USE\\_INTTYPES\\_H](#)
- #define [LITTLE\\_ENDIAN](#) 1234
- #define [BYTE\\_ORDER](#) LITTLE\_ENDIAN
- #define [SHA256\\_BLOCK\\_LENGTH](#) 64
- #define [SHA256\\_DIGEST\\_LENGTH](#) 32
- #define [SHA256\\_DIGEST\\_STRING\\_LENGTH](#) (SHA256\_DIGEST\_LENGTH \* 2 + 1)
- #define [SHA384\\_BLOCK\\_LENGTH](#) 128
- #define [SHA384\\_DIGEST\\_LENGTH](#) 48
- #define [SHA384\\_DIGEST\\_STRING\\_LENGTH](#) (SHA384\_DIGEST\_LENGTH \* 2 + 1)
- #define [SHA512\\_BLOCK\\_LENGTH](#) 128
- #define [SHA512\\_DIGEST\\_LENGTH](#) 64
- #define [SHA512\\_DIGEST\\_STRING\\_LENGTH](#) (SHA512\_DIGEST\_LENGTH \* 2 + 1)

### Typedefs

- typedef struct [\\_SHA256\\_CTX](#) [SHA256\\_CTX](#)
- typedef struct [\\_SHA512\\_CTX](#) [SHA512\\_CTX](#)
- typedef [SHA512\\_CTX](#) [SHA384\\_CTX](#)



## Functions

- void [SHA256\\_Init](#) (SHA256\_CTX \*)
- void [SHA256\\_Update](#) (SHA256\_CTX \*, const uint8\_t \*, size\_t)
- void [SHA256\\_Final](#) (uint8\_t[SHA256\_DIGEST\_LENGTH], SHA256\_CTX \*)
- char \* [SHA256\\_End](#) (SHA256\_CTX \*, char[SHA256\_DIGEST\_STRING\_LENGTH])
- char \* [SHA256\\_Data](#) (const uint8\_t \*, size\_t, char[SHA256\_DIGEST\_STRING\_LENGTH])
- void [SHA384\\_Init](#) (SHA384\_CTX \*)
- void [SHA384\\_Update](#) (SHA384\_CTX \*, const uint8\_t \*, size\_t)
- void [SHA384\\_Final](#) (uint8\_t[SHA384\_DIGEST\_LENGTH], SHA384\_CTX \*)
- char \* [SHA384\\_End](#) (SHA384\_CTX \*, char[SHA384\_DIGEST\_STRING\_LENGTH])
- char \* [SHA384\\_Data](#) (const uint8\_t \*, size\_t, char[SHA384\_DIGEST\_STRING\_LENGTH])
- void [SHA512\\_Init](#) (SHA512\_CTX \*)
- void [SHA512\\_Update](#) (SHA512\_CTX \*, const uint8\_t \*, size\_t)
- void [SHA512\\_Final](#) (uint8\_t[SHA512\_DIGEST\_LENGTH], SHA512\_CTX \*)
- char \* [SHA512\\_End](#) (SHA512\_CTX \*, char[SHA512\_DIGEST\_STRING\_LENGTH])
- char \* [SHA512\\_Data](#) (const uint8\_t \*, size\_t, char[SHA512\_DIGEST\_STRING\_LENGTH])

### 4.55.1 Macro Definition Documentation

4.55.1.1 `#define BYTE_ORDER LITTLE_ENDIAN`

4.55.1.2 `#define LITTLE_ENDIAN 1234`

4.55.1.3 `#define SHA256_BLOCK_LENGTH 64`

Import u\_intXX\_t size\_t type definitions from system headers. You may need to change this, or define these things yourself in this file.

4.55.1.4 `#define SHA256_DIGEST_LENGTH 32`

4.55.1.5 `#define SHA256_DIGEST_STRING_LENGTH (SHA256_DIGEST_LENGTH * 2 + 1)`

4.55.1.6 `#define SHA2_USE_INTTYPES_H`

4.55.1.7 `#define SHA384_BLOCK_LENGTH 128`

4.55.1.8 `#define SHA384_DIGEST_LENGTH 48`

4.55.1.9 `#define SHA384_DIGEST_STRING_LENGTH (SHA384_DIGEST_LENGTH * 2 + 1)`

4.55.1.10 `#define SHA512_BLOCK_LENGTH 128`

4.55.1.11 `#define SHA512_DIGEST_LENGTH 64`

4.55.1.12 `#define SHA512_DIGEST_STRING_LENGTH (SHA512_DIGEST_LENGTH * 2 + 1)`

### 4.55.2 Typedef Documentation

4.55.2.1 `typedef struct _SHA256_CTX SHA256_CTX`

NOTE: If your architecture does not define either u\_intXX\_t types or uintXX\_t (from inttypes.h), you may need to define things by hand for your system: Most BSD systems already define u\_intXX\_t types, as does Linux. Some systems, however, like Compaq's Tru64 Unix instead can use uintXX\_t types defined by very recent ANSI C standards and included in the file:

```
#include <inttypes.h>
```

If you choose to use <inttypes.h> then please define:

```
#define SHA2_USE_INTTYPES_H
```

Or on the command line during compile:

```
cc -DSHA2_USE_INTTYPES_H ...
```

4.55.2.2 `typedef SHA512_CTX SHA384_CTX`

4.55.2.3 `typedef struct _SHA512_CTX SHA512_CTX`

### 4.55.3 Function Documentation

4.55.3.1 `char* SHA256_Data ( const uint8_t *, size_t, char [SHA256_DIGEST_STRING_LENGTH] )`

4.55.3.2 `char* SHA256_End ( SHA256_CTX *, char [SHA256_DIGEST_STRING_LENGTH] )`

4.55.3.3 `void SHA256_Final ( uint8_t [SHA256_DIGEST_LENGTH], SHA256_CTX * )`

4.55.3.4 `void SHA256_Init ( SHA256_CTX * )`

4.55.3.5 `void SHA256_Update ( SHA256_CTX *, const uint8_t *, size_t )`

4.55.3.6 `char* SHA384_Data ( const uint8_t *, size_t, char [SHA384_DIGEST_STRING_LENGTH] )`

4.55.3.7 `char* SHA384_End ( SHA384_CTX *, char [SHA384_DIGEST_STRING_LENGTH] )`

4.55.3.8 `void SHA384_Final ( uint8_t [SHA384_DIGEST_LENGTH], SHA384_CTX * )`

4.55.3.9 `void SHA384_Init ( SHA384_CTX * )`

4.55.3.10 `void SHA384_Update ( SHA384_CTX *, const uint8_t *, size_t )`

4.55.3.11 `char* SHA512_Data ( const uint8_t *, size_t, char [SHA512_DIGEST_STRING_LENGTH] )`

4.55.3.12 `char* SHA512_End ( SHA512_CTX *, char [SHA512_DIGEST_STRING_LENGTH] )`

4.55.3.13 `void SHA512_Final ( uint8_t [SHA512_DIGEST_LENGTH], SHA512_CTX * )`

4.55.3.14 `void SHA512_Init ( SHA512_CTX * )`

4.55.3.15 `void SHA512_Update ( SHA512_CTX *, const uint8_t *, size_t )`

## 4.56 xen/stubdom/domc/cspipe\_intf.h File Reference

## 4.57 xen/stubdom/domt/cspipe\_intf.h File Reference

```
#include <inttypes.h>
```

### Data Structures

- struct [cspipe](#)

## Functions

- struct [cspipe \\_\\_attribute\\_\\_ \(\(\\_\\_packed\\_\\_\)\)](#)

## Variables

- uint32\_t [magic\\_nr](#)
- uint32\_t [domid\\_domu](#)
- uint32\_t [domid\\_domc](#)
- char [s\\_key](#) [128/8]

### 4.57.1 Function Documentation

4.57.1.1 struct [cspipe \\_\\_attribute\\_\\_ \( \(\\_\\_packed\\_\\_ \)](#)

### 4.57.2 Variable Documentation

4.57.2.1 uint32\_t [domid\\_domc](#)

4.57.2.2 uint32\_t [domid\\_domu](#)

4.57.2.3 uint32\_t [magic\\_nr](#)

4.57.2.4 char [s\\_key](#)[128/8]

## 4.58 xen/stubdom/domc/main.h File Reference

```
#include <mini-os/blkfront.h>
#include <minios_macros.h>
```

## Data Structures

- struct [domc\\_info](#)

## Typedefs

- typedef struct [domc\\_info](#) [domc\\_info\\_t](#)

## Functions

- int [wrap\\_notify\\_remote\\_via\\_evtchn](#) (evtchn\_port\_t port)

### 4.58.1 Typedef Documentation

4.58.1.1 typedef struct [domc\\_info](#) [domc\\_info\\_t](#)

### 4.58.2 Function Documentation

4.58.2.1 int [wrap\\_notify\\_remote\\_via\\_evtchn](#) ( evtchn\_port\_t *port* )

## 4.59 xen/stubdom/domt/main.h File Reference

```
#include <minios_macros.h>
```

## 4.60 xen/stubdom/domt/caas.c File Reference

```
#include <xenctrl.h>
#include <xc_dom.h>
#include <os.h>
#include <sys/mman.h>
#include "main.h"
#include "xlc.h"
#include "tc.h"
#include "caas.h"
```

### Functions

- int [caas\\_init](#) (struct [tc\\_state](#) \*state)
- int [caas\\_domain\\_create](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_domain\\_config](#) \*d\_config, uint32\_t \*domid\_out, uint32\_t \*domid\_out\_dc, int restore\_fd)
- int [caas\\_setup\\_cspipe](#) (void \*info\_p, void \*dom\_p, unsigned int mfn)

### Variables

- struct [tc\\_state](#) \* [tc](#)

### 4.60.1 Function Documentation

**4.60.1.1** int [caas\\_domain\\_create](#) ( [libxl\\_ctx](#) \* *ctx*, [libxl\\_domain\\_config](#) \* *d\_config*, uint32\_t \* *domid\_out*, uint32\_t \* *domid\_out\_dc*, int *restore\_fd* )

The domain creation function specifically finetuned for crypto domain booting.

We create a new domain config which takes the domc\_disks / domc\_vifs from the original domain config as its disks and vifs.

The supplied kernel/ramdisk information is not used for either domain; the DomU will boot a PVGRUB, while the DomC clone will boot a DomC template.

TODO: it would probably be best if the customer in the encrypted payload can embed a policy that a) indicates the uuid of the primary boot device and b) expresses whether any unencrypted disks should even be attached at all. (Right now the sysadmin can attach those as well.)

**4.60.1.2** int [caas\\_init](#) ( struct [tc\\_state](#) \* *state* )

**4.60.1.3** int [caas\\_setup\\_cspipe](#) ( void \* *info\_p*, void \* *dom\_p*, unsigned int *mfn* )

Set-up the CaaS pipe between DomT <-> DomC. Initially, this will contain the AES key; later on it should contain also the vTPM NV, perhaps in the form as a pipe (therefore this naming) in similar form as the vTPM pipes normally in Dom0. Right now, this is a callback from inside libxl; in the future it would be cleaner to use a ring buffer and event channels so that we can set it up after the domain has been built, making the callback unnecessary.

## 4.60.2 Variable Documentation

### 4.60.2.1 struct tc\_state\* tc

Since the module is a singleton, this is OK.

## 4.61 xen/stubdom/domt/caas.h File Reference

```
#include <libxl.h>
#include "tc.h"
#include "cspipe_intf.h"
```

### Data Structures

- struct [vmcb](#)

### Functions

- struct [vmcb](#) [\\_\\_attribute\\_\\_](#) (([\\_\\_packed\\_\\_](#)))
- int [caas\\_init](#) (struct [tc\\_state](#) \*state)
- int [caas\\_domain\\_create](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_domain\\_config](#) \*d\_config, uint32\_t \*domid\_out, uint32\_t \*domid\_out\_dc, int restore\_fd)
- int [caas\\_setup\\_cspipe](#) (void \*info\_p, void \*dom\_p, unsigned int mfn)

### Variables

- char [s\\_key](#) [128/8]

### 4.61.1 Function Documentation

#### 4.61.1.1 struct vmcb \_\_attribute\_\_ ( (\_\_packed\_\_ ) )

#### 4.61.1.2 int caas\_domain\_create ( libxl\_ctx \* ctx, libxl\_domain\_config \* d\_config, uint32\_t \* domid\_out, uint32\_t \* domid\_out\_dc, int restore\_fd )

The domain creation function specifically finetuned for crypto domain booting.

We create a new domain config which takes the domc\_disks / domc\_vifs from the original domain config as its disks and vifs.

The supplied kernel/ramdisk information is not used for either domain; the DomU will boot a PVGRUB, while the DomC clone will boot a DomC template.

TODO: it would probably be best if the customer in the encrypted payload can embed a policy that a) indicates the uuid of the primary boot device and b) expresses whether any unencrypted disks should even be attached at all. (Right now the sysadmin can attach those as well.)

#### 4.61.1.3 int caas\_init ( struct tc\_state \* state )

#### 4.61.1.4 int caas\_setup\_cspipe ( void \* info\_p, void \* dom\_p, unsigned int mfn )

Set-up the CaaS pipe between DomT <-> DomC. Initially, this will contain the AES key; later on it should contain also the vTPM NV, perhaps in the form as a pipe (therefore this naming) in similar form as the vTPM pipes normally

in Dom0. Right now, this is a callback from inside libxl; in the future it would be cleaner to use a ring buffer and event channels so that we can set it up after the domain has been built, making the callback unnecessary.

#### 4.61.2 Variable Documentation

4.61.2.1 `char s_key[128/8]`

### 4.62 xen/stubdom/domt/crypto/hmac.c File Reference

```
#include "hmac.h"
```

#### Functions

- void `tpm_hmac_init` (`tpm_hmac_ctx_t *ctx`, const `uint8_t *key`, `size_t key_len`)
- void `tpm_hmac_update` (`tpm_hmac_ctx_t *ctx`, const `uint8_t *data`, `size_t length`)
- void `tpm_hmac_final` (`tpm_hmac_ctx_t *ctx`, `uint8_t *digest`)

#### 4.62.1 Function Documentation

4.62.1.1 void `tpm_hmac_final` ( `tpm_hmac_ctx_t * ctx`, `uint8_t * digest` )

4.62.1.2 void `tpm_hmac_init` ( `tpm_hmac_ctx_t * ctx`, const `uint8_t * key`, `size_t key_len` )

4.62.1.3 void `tpm_hmac_update` ( `tpm_hmac_ctx_t * ctx`, const `uint8_t * data`, `size_t length` )

### 4.63 xen/stubdom/domt/crypto/hmac.h File Reference

```
#include "sha1.h"
```

#### Data Structures

- struct `tpm_hmac_ctx_t`

#### Macros

- `#define HMAC_PAD_LENGTH 64`

#### Functions

- void `tpm_hmac_init` (`tpm_hmac_ctx_t *ctx`, const `uint8_t *key`, `size_t key_len`)
- void `tpm_hmac_update` (`tpm_hmac_ctx_t *ctx`, const `uint8_t *data`, `size_t length`)
- void `tpm_hmac_final` (`tpm_hmac_ctx_t *ctx`, `uint8_t *digest`)

#### Variables

- typedef `__attribute__`

### 4.63.1 Macro Definition Documentation

4.63.1.1 `#define HMAC_PAD_LENGTH 64`

### 4.63.2 Function Documentation

4.63.2.1 `void tpm_hmac_final ( tpm_hmac_ctx_t * ctx, uint8_t * digest )`

4.63.2.2 `void tpm_hmac_init ( tpm_hmac_ctx_t * ctx, const uint8_t * key, size_t key_len )`

4.63.2.3 `void tpm_hmac_update ( tpm_hmac_ctx_t * ctx, const uint8_t * data, size_t length )`

### 4.63.3 Variable Documentation

4.63.3.1 `typedef __attribute__`

## 4.64 xen/stubdom/domt/crypto/sha1.c File Reference

```
#include "sha1.h"
```

### Macros

- `#define rol(v, b) (((v) << (b)) | ((v) >> (32 - (b))))`
- `#define B0(i) (buf[i] = CPU_TO_BE32(buf[i]))`
- `#define B1(i)`
- `#define F0(x, y, z) ((x & (y ^ z)) ^ z)`
- `#define F1(x, y, z) (x ^ y ^ z)`
- `#define F2(x, y, z) (((x | y) & z) | (x & y))`
- `#define R0(a, b, c, d, e, i) e += F0(b,c,d) + B0(i) + 0x5A827999 + rol(a,5); b = rol(b,30);`
- `#define R1(a, b, c, d, e, i) e += F0(b,c,d) + B1(i) + 0x5A827999 + rol(a,5); b = rol(b,30);`
- `#define R2(a, b, c, d, e, i) e += F1(b,c,d) + B1(i) + 0x6ED9EBA1 + rol(a,5); b = rol(b,30);`
- `#define R3(a, b, c, d, e, i) e += F2(b,c,d) + B1(i) + 0x8F1BBCDC + rol(a,5); b = rol(b,30);`
- `#define R4(a, b, c, d, e, i) e += F1(b,c,d) + B1(i) + 0xCA62C1D6 + rol(a,5); b = rol(b,30);`

### Functions

- `void tpm_sha1_init (tpm_sha1_ctx_t *ctx)`
- `void tpm_sha1_update (tpm_sha1_ctx_t *ctx, const uint8_t *data, size_t length)`
- `void tpm_sha1_final (tpm_sha1_ctx_t *ctx, uint8_t digest[SHA1_DIGEST_LENGTH])`

### 4.64.1 Macro Definition Documentation

4.64.1.1 `#define B0( i ) (buf[i] = CPU_TO_BE32(buf[i]))`

4.64.1.2 `#define B1( i )`

#### Value:

```
(buf[i & 15] = rol(buf[i & 15] ^ buf[(i-14) & 15] \
                    ^ buf[(i-8) & 15] ^ buf[(i-3) & 15], 1))
```

4.64.1.3 `#define F0( x, y, z ) ((x & (y ^ z)) ^ z)`

4.64.1.4 `#define F1( x, y, z ) (x ^ y ^ z)`

4.64.1.5 `#define F2( x, y, z ) (((x | y) & z) | (x & y))`

4.64.1.6 `#define R0( a, b, c, d, e, i ) e += F0(b,c,d) + B0(i) + 0x5A827999 + rol(a,5); b = rol(b,30);`

4.64.1.7 `#define R1( a, b, c, d, e, i ) e += F0(b,c,d) + B1(i) + 0x5A827999 + rol(a,5); b = rol(b,30);`

4.64.1.8 `#define R2( a, b, c, d, e, i ) e += F1(b,c,d) + B1(i) + 0x6ED9EBA1 + rol(a,5); b = rol(b,30);`

4.64.1.9 `#define R3( a, b, c, d, e, i ) e += F2(b,c,d) + B1(i) + 0x8F1BBCDC + rol(a,5); b = rol(b,30);`

4.64.1.10 `#define R4( a, b, c, d, e, i ) e += F1(b,c,d) + B1(i) + 0xCA62C1D6 + rol(a,5); b = rol(b,30);`

4.64.1.11 `#define rol( v, b ) (((v) << (b)) | ((v) >> (32 - (b))))`

## 4.64.2 Function Documentation

4.64.2.1 `void tpm_sha1_final ( tpm_sha1_ctx_t * ctx, uint8_t digest[SHA1_DIGEST_LENGTH] )`

4.64.2.2 `void tpm_sha1_init ( tpm_sha1_ctx_t * ctx )`

4.64.2.3 `void tpm_sha1_update ( tpm_sha1_ctx_t * ctx, const uint8_t * data, size_t length )`

## 4.65 xen/stubdom/domt/crypto/sha1.h File Reference

```
#include <inttypes.h>
#include <os.h>
#include <byteorder.h>
#include <arpa/inet.h>
#include <mini-os/byteorder.h>
#include <unistd.h>
#include <stdbool.h>
```

### Data Structures

- struct [tpm\\_sha1\\_ctx\\_t](#)

### Macros

- `#define CPU_TO_BE32` [cpu\\_to\\_be32](#)
- `#define SHA1_DIGEST_LENGTH` 20

### Functions

- void [tpm\\_sha1\\_init](#) ([tpm\\_sha1\\_ctx\\_t](#) \*ctx)
- void [tpm\\_sha1\\_update](#) ([tpm\\_sha1\\_ctx\\_t](#) \*ctx, const uint8\_t \*data, size\_t length)
- void [tpm\\_sha1\\_final](#) ([tpm\\_sha1\\_ctx\\_t](#) \*ctx, uint8\_t digest[SHA1\_DIGEST\_LENGTH])



## 4.65.1 Macro Definition Documentation

### 4.65.1.1 #define CPU\_TO\_BE32 *cpu\_to\_be32*

Necessary includes for the port of the library to Mini-OS.

### 4.65.1.2 #define SHA1\_DIGEST\_LENGTH 20

## 4.65.2 Function Documentation

### 4.65.2.1 void *tpm\_sha1\_final* ( *tpm\_sha1\_ctx\_t* \* *ctx*, *uint8\_t* *digest*[*SHA1\_DIGEST\_LENGTH*] )

### 4.65.2.2 void *tpm\_sha1\_init* ( *tpm\_sha1\_ctx\_t* \* *ctx* )

### 4.65.2.3 void *tpm\_sha1\_update* ( *tpm\_sha1\_ctx\_t* \* *ctx*, *const uint8\_t* \* *data*, *size\_t* *length* )

## 4.66 xen/stubdom/domt/tc.c File Reference

```
#include <unistd.h>
#include <os.h>
#include <malloc.h>
#include <string.h>
#include <mini-os/ioremap.h>
#include <mini-os/tpm_tis.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdlib.h>
#include "main.h"
#include "tc.h"
#include "tpm_defs.h"
#include "hmac.h"
```

## Macros

- #define [TPM\\_DEFAULT\\_ENCSIZE](#) 256
- #define [TPM\\_DEFAULT\\_PKEYLEN](#) 256
- #define [TC\\_CHECK\\_BUF\\_OVERFLOW](#)(buf, len)
- #define [TC\\_CHECK\\_RC\\_ERROR](#)(ret, rc, rsplen)
- #define [TC\\_CHECK\\_IF\\_ALL\\_READ](#)(p, rsp, rsplen)
- #define [TC\\_CHECK\\_DIGESTS](#)(resAuth, verifyAuth)
- #define [TC\\_OPT\\_INVALIDATE\\_OIAP](#)(s, contAuthSes)
- #define [TC\\_OPT\\_INVALIDATE\\_OSAP](#)(s, contAuthSes)
- #define [TC\\_CHECK\\_TPM\\_TIS\\_CMD](#)(chip, buf, len, rsp, rsplen)
- #define [FILL\\_CONVERT](#) 1
- #define [TC\\_PLACE\\_SZ](#)(SZ)
- #define [tc\\_place](#)\_(a, b, c, d) *tc\_place*(a, b, c, d, NULL, NULL)
- #define [tc\\_place\\_s](#)\_(a, b, c, d, e) *tc\_place*(a, b, c, d, e, NULL)
- #define [tc\\_place\\_h](#)\_(a, b, c, d, e) *tc\_place*(a, b, c, d, NULL, e)
- #define [tc\\_place16](#)\_(a, b, c) *tc\_place16*(a, b, c, NULL, NULL)
- #define [tc\\_place16\\_s](#)\_(a, b, c, d) *tc\_place16*(a, b, c, d, NULL)
- #define [tc\\_place16\\_h](#)\_(a, b, c, d) *tc\_place16*(a, b, c, NULL, d)
- #define [tc\\_place32](#)\_(a, b, c) *tc\_place32*(a, b, c, NULL, NULL)
- #define [tc\\_place32\\_s](#)\_(a, b, c, d) *tc\_place32*(a, b, c, d, NULL)
- #define [tc\\_place32\\_h](#)\_(a, b, c, d) *tc\_place32*(a, b, c, NULL, d)

## Functions

- `int tc_transmit_cmd` (struct `tc_info` \*info, struct `tpm_cmd` \*cmd)
- `int tc_createwrapkey` (struct `tc_info` \*info, struct `osap_sess` \*s, uint32\_t `handle`, uint8\_t \*dataUsageAuth, size\_t dataUsageAuthLen, struct `TPM_KEY12` \*key, struct `TPM_KEY12` \*key2)
- `int tc_loadkey2` (struct `tc_info` \*info, struct `oiap_sess` \*s, uint32\_t `parentHandle`, uint8\_t \*parentAuth, size\_t parentAuthLen, struct `TPM_KEY12` \*key, uint32\_t \*inkeyHandle, uint8\_t contAuthSes)
- `int tc_chk_loadkey2` (struct `tc_info` \*info, struct `oiap_sess` \*s, uint32\_t `parentHandle`, uint8\_t \*parentAuth, size\_t parentAuthLen, struct `TPM_KEY12` \*key, uint32\_t \*inkeyHandle, uint8\_t contAuthSes)
- `int tc_unbind` (struct `tc_info` \*info, struct `oiap_sess` \*s, uint32\_t `keyHandle`, uint8\_t \*keyAuth, size\_t keyAuthLen, uint8\_t \*inData, uint32\_t inDataSize, uint8\_t \*\*outData, uint32\_t \*outDataSize, uint8\_t contAuthSes)
- `int tc_listkeys` (struct `tc_info` \*info, uint32\_t \*\*keylist, uint16\_t \*keylistSize)
- `void tc_flush_all_keys` (struct `tc_info` \*info)
- `void tc_print_pcrs` (struct `tc_info` \*info)
- `int tpm_pcr_extend` (struct `tc_info` \*info, int `pcr_idx`, const uint8\_t \*hash)
- `int tpm_get_random` (struct `tc_info` \*info, unsigned char \*buf, uint32\_t len)
- `int init_oiap` (struct `tc_info` \*info, struct `oiap_sess` \*s)
- `int init_osap` (struct `tc_info` \*info, struct `osap_sess` \*s, const unsigned char \*key, uint16\_t type, uint32\_t `handle`)
- `int tc_create_key` (struct `tc_info` \*info, struct `osap_sess` \*s, struct `TPM_KEY12` \*\*in\_key, uint8\_t \*dataUsageAuth, size\_t len)
- `void tc_free_key` (struct `TPM_KEY12` \*key)
- `int tc_reset` (struct `tc_info` \*info)
- `int tc_flushspecific` (struct `tc_info` \*info, uint32\_t `handle`, uint32\_t `resourceType`)
- `struct tc_state * init_tc` (void)
- `void shutdown_tc` (struct `tc_state` \*state)
- `void tc_test` (struct `tc_state` \*state)
- `void tc_hexdump` (void \*p, size\_t len)

### 4.66.1 Macro Definition Documentation

#### 4.66.1.1 #define FILL\_CONVERT 1

NOTE: we *must* hash everything in big endian to match TPM. However, if dumping to a place in cpu ordering, one can set as a non-zero value != FILL\_CONVERT (e.g., 2). Secondly, setting == NULL is an 'update hash only' modus.

#### 4.66.1.2 #define TC\_CHECK\_BUF\_OVERFLOW( buf, len )

**Value:**

```
DEBUG("[cmd length: %d]\n", len);
if (len > sizeof(buf)) {
    PANIC("FATAL: TPM buffer overflowed!\n");
}
```

Helper macros to make the code with TC commands less cluttered by removing boilerplate code. These should not be used standalone, due to hidden assumptions on context.

#### 4.66.1.3 #define TC\_CHECK\_DIGESTS( resAuth, verifyAuth )

**Value:**

```
if (memcmp(resAuth, verifyAuth, TPM_DIGEST_SIZE)) {
    PANIC("TPM return digest did not match!\n");
}
```

4.66.1.4 #define TC\_CHECK\_IF\_ALL\_READ( *p*, *rsp*, *rsplen* )**Value:**

```

if ((p - rsp) != rsplen) {
    DEBUG("Error: walking pointer has read != rsplen\n");
    goto out;
}

```

4.66.1.5 #define TC\_CHECK\_RC\_ERROR( *ret*, *rc*, *rsplen* )**Value:**

```

if (rsplen == TPM_ERROR_SIZE) {
    DEBUG("warning: rsplen (%d) indicates an error\n", rsplen);
    if ((ret = rc) != 0) {
        DEBUG("TPM error: %d (0x%x)\n", rc, rc);
        goto out;
    }
}

```

4.66.1.6 #define TC\_CHECK\_TPM\_TIS\_CMD( *chip*, *buf*, *len*, *rsp*, *rsplen* )**Value:**

```

if ((ret = tpm_tis_cmd(chip, buf, len, &rsp, &rsplen)) != 0) {
    DEBUG("warning: non-zero return from tpm_tis driver: %d\n", ret);
    /*goto out;*/
}
if (rsplen < sizeof(struct tpm_input_header)) {
    DEBUG("Error: TPM cmd returned an inconsistent rsplen."
        " (Usually caused by bad formatted input.)\n");
    ret = rsplen;
    goto out;
}

```

4.66.1.7 #define TC\_OPT\_INVALIDATE\_OIAP( *s*, *contAuthSes* )**Value:**

```

if (contAuthSes == 0) {
    s->handle = 0;
    memset(s->enonce, 0, sizeof(s->enonce));
}

```

4.66.1.8 #define TC\_OPT\_INVALIDATE\_OSAP( *s*, *contAuthSes* )**Value:**

```

if (contAuthSes == 0) {
    s->handle = 0;
    memset(s->secret, 0, sizeof(s->secret));
    memset(s->enonce, 0, sizeof(s->enonce));
}

```

```

4.66.1.9  #define tc_place16_( a, b, c ) tc_place16(a, b, c, NULL, NULL)

4.66.1.10 #define tc_place16_h( a, b, c, d ) tc_place16(a, b, c, NULL, d)

4.66.1.11 #define tc_place16_s( a, b, c, d ) tc_place16(a, b, c, d, NULL)

4.66.1.12 #define tc_place32_( a, b, c ) tc_place32(a, b, c, NULL, NULL)

4.66.1.13 #define tc_place32_h( a, b, c, d ) tc_place32(a, b, c, NULL, d)

4.66.1.14 #define tc_place32_s( a, b, c, d ) tc_place32(a, b, c, d, NULL)

4.66.1.15 #define tc_place_( a, b, c, d ) tc_place(a, b, c, d, NULL, NULL)

4.66.1.16 #define tc_place_h( a, b, c, d, e ) tc_place(a, b, c, d, NULL, e)

4.66.1.17 #define tc_place_s( a, b, c, d, e ) tc_place(a, b, c, d, e, NULL)

4.66.1.18 #define TC_PLACE_SZ( SZ )

```

**Value:**

```

static void tc_place ## SZ (int fill, uint8_t **p, uint ## SZ ## _t *in,
                             tpm_shal_ctx_t *shh, tpm_hmac_ctx_t *hnh)
{
    uint ## SZ ## _t tmp;
    int len = sizeof(*in);
    BUG_ON(!in);
    if (fill == FILL_CONVERT) {
        tmp = cpu_to_be ## SZ (*in);
        in = &tmp;
    }
    if (!fill)
        memcpy(in, *p, len);
    if (shh != NULL)
        tpm_shal_update(shh, (uint8_t*)in, len);
    if (hnh != NULL)
        tpm_hmac_update(hnh, (uint8_t*)in, len);
    if (!p || !*p)
        return;
    if (fill)
        memcpy(*p, in, len);
    if (!fill)
        *in = cpu_to_be ## SZ (*in);
    *p += len;
}

```

```

4.66.1.19 #define TPM_DEFAULT_ENCSIZE 256

```

Code comments are sparse, refer to TPM spec for details.

```

4.66.1.20 #define TPM_DEFAULT_PKEYLEN 256

```

**4.66.2 Function Documentation**

```

4.66.2.1 int init_oiap ( struct tc_info * info, struct oiap_sess * s )

```

```

4.66.2.2 int init_osap ( struct tc_info * info, struct osap_sess * s, const unsigned char * key, uint16_t type, uint32_t
    handle )

```

Create an object specific authorisation protocol (OSAP) session.

4.66.2.3 `struct tc_state* init_tc ( void )`

4.66.2.4 `void shutdown_tc ( struct tc_state * state )`

4.66.2.5 `int tc_chk_loadkey2 ( struct tc_info * info, struct oiap_sess * s, uint32_t parentHandle, uint8_t * parentAuth, size_t parentAuthLen, struct TPM_KEY12 * key, uint32_t * inkeyHandle, uint8_t contAuthSes )`

4.66.2.6 `int tc_create_key ( struct tc_info * info, struct osap_sess * s, struct TPM_KEY12 ** in_key, uint8_t * dataUsageAuth, size_t len )`

4.66.2.7 `int tc_createwrapkey ( struct tc_info * info, struct osap_sess * s, uint32_t handle, uint8_t * dataUsageAuth, size_t dataUsageAuthLen, struct TPM_KEY12 * key, struct TPM_KEY12 * key2 )`

Because of the variable length arguments, we do not use the `tpm_input_*` structs approach. Passing `==` is allowed.

4.66.2.8 `void tc_flush_all_keys ( struct tc_info * info )`

4.66.2.9 `int tc_flushspecific ( struct tc_info * info, uint32_t handle, uint32_t resourceType )`

4.66.2.10 `void tc_free_key ( struct TPM_KEY12 * key )`

4.66.2.11 `void tc_hexdump ( void * p, size_t len )`

Not a particular TC related function, but heavily used in all modules for debugging.

4.66.2.12 `int tc_listkeys ( struct tc_info * info, uint32_t ** keylist, uint16_t * keylistSize )`

If necessary, this could be generalized to a `get_capability` function.

4.66.2.13 `int tc_loadkey2 ( struct tc_info * info, struct oiap_sess * s, uint32_t parentHandle, uint8_t * parentAuth, size_t parentAuthLen, struct TPM_KEY12 * key, uint32_t * inkeyHandle, uint8_t contAuthSes )`

Caller is responsible for being properly initialized.

4.66.2.14 `void tc_print_pcrs ( struct tc_info * info )`

4.66.2.15 `int tc_reset ( struct tc_info * info )`

RESET is deprecated in 1.2 of spec.

4.66.2.16 `void tc_test ( struct tc_state * state )`

4.66.2.17 `int tc_transmit_cmd ( struct tc_info * info, struct tpm_cmd * cmd )`

Note: not to be used with variable-length commands such as `createwrapkey`, `loadkey2`, etc.

4.66.2.18 `int tc_unbind ( struct tc_info * info, struct oiap_sess * s, uint32_t keyHandle, uint8_t * keyAuth, size_t keyAuthLen, uint8_t * inData, uint32_t inDataSize, uint8_t ** outData, uint32_t * outDataSize, uint8_t contAuthSes )`

4.66.2.19 `int tpm_get_random ( struct tc_info * info, unsigned char * buf, uint32_t len )`

4.66.2.20 `int tpm_pcr_extend ( struct tc_info * info, int pcr_idx, const uint8_t * hash )`

## 4.67 xen/stubdom/domt/tc.h File Reference

```
#include <tpm_tis.h>
#include "tpm_defs.h"
#include "hmac.h"
```

### Data Structures

- struct [tc\\_info](#)
- struct [tc\\_state](#)
- struct [osap\\_sess](#)
- struct [oiap\\_sess](#)

### Macros

- #define [TPM\\_MFN](#) (TPM\_BASEADDR >> PAGE\_SHIFT)
- #define [TPM\\_LEN](#) 5
- #define [TPM\\_LOCS](#) TPM\_TIS\_EN\_LOCL0
- #define [TPM\\_NUM\\_PCR](#) 24
- #define [TC\\_PATH\\_PREFIX](#) "/var/"
- #define [TC\\_PATH\\_KEYDATA](#) TC\_PATH\_PREFIX "domt\_keydata.bin"
- #define [TC\\_PATH\\_PUBKEY](#) TC\_PATH\_PREFIX "domt\_pubkey.bin"

### Functions

- struct [tc\\_state](#) \* [init\\_tc](#) (void)
- void [shutdown\\_tc](#) (struct [tc\\_state](#) \*state)
- int [tc\\_transmit\\_cmd](#) (struct [tc\\_info](#) \*info, struct [tpm\\_cmd](#) \*in)
- int [tc\\_createwrapkey](#) (struct [tc\\_info](#) \*info, struct [osap\\_sess](#) \*s, uint32\_t [handle](#), uint8\_t \*dataUsageAuth, size\_t dataUsageAuthLen, struct [TPM\\_KEY12](#) \*key, struct [TPM\\_KEY12](#) \*key2)
- int [tc\\_loadkey2](#) (struct [tc\\_info](#) \*info, struct [oiap\\_sess](#) \*s, uint32\_t parentHandle, uint8\_t \*parentAuth, size\_t parentAuthLen, struct [TPM\\_KEY12](#) \*key, uint32\_t \*inkeyHandle, uint8\_t contAuthSes)
- int [tc\\_chk\\_loadkey2](#) (struct [tc\\_info](#) \*info, struct [oiap\\_sess](#) \*s, uint32\_t parentHandle, uint8\_t \*parentAuth, size\_t parentAuthLen, struct [TPM\\_KEY12](#) \*key, uint32\_t \*inkeyHandle, uint8\_t contAuthSes)
- int [tc\\_unbind](#) (struct [tc\\_info](#) \*info, struct [oiap\\_sess](#) \*s, uint32\_t keyHandle, uint8\_t \*keyAuth, size\_t keyAuthLen, uint8\_t \*inData, uint32\_t inDataSize, uint8\_t \*\*outData, uint32\_t \*outDataSize, uint8\_t contAuthSes)
- int [tc\\_listkeys](#) (struct [tc\\_info](#) \*info, uint32\_t \*\*keylist, uint16\_t \*keylistSize)
- void [tc\\_flush\\_all\\_keys](#) (struct [tc\\_info](#) \*info)
- void [tc\\_print\\_pcrs](#) (struct [tc\\_info](#) \*info)
- int [tpm\\_pcr\\_extend](#) (struct [tc\\_info](#) \*info, int [pcr\\_idx](#), const uint8\_t \*hash)
- int [tpm\\_get\\_random](#) (struct [tc\\_info](#) \*info, unsigned char \*buf, uint32\_t len)
- int [init\\_oiap](#) (struct [tc\\_info](#) \*info, struct [oiap\\_sess](#) \*s)
- int [init\\_osap](#) (struct [tc\\_info](#) \*info, struct [osap\\_sess](#) \*s, const unsigned char \*key, uint16\_t type, uint32\_t [handle](#))
- int [tc\\_create\\_key](#) (struct [tc\\_info](#) \*info, struct [osap\\_sess](#) \*s, struct [TPM\\_KEY12](#) \*\*in\_key, uint8\_t \*dataUsageAuth, size\_t len)
- void [tc\\_free\\_key](#) (struct [TPM\\_KEY12](#) \*key)
- int [tc\\_reset](#) (struct [tc\\_info](#) \*info)
- int [tc\\_flushspecific](#) (struct [tc\\_info](#) \*info, uint32\_t [handle](#), uint32\_t [resourceType](#))
- void [tc\\_test](#) (struct [tc\\_state](#) \*state)
- void [tc\\_hexdump](#) (void \*p, size\_t len)

## 4.67.1 Macro Definition Documentation

4.67.1.1 `#define TC_PATH_KEYDATA TC_PATH_PREFIX "domt_keydata.bin"`

4.67.1.2 `#define TC_PATH_PREFIX "/var/"`

4.67.1.3 `#define TC_PATH_PUBKEY TC_PATH_PREFIX "domt_pubkey.bin"`

4.67.1.4 `#define TPM_LEN 5`

4.67.1.5 `#define TPM_LOCS TPM_TIS_EN_LOCL0`

4.67.1.6 `#define TPM_MFN (TPM_BASEADDR >> PAGE_SHIFT)`

4.67.1.7 `#define TPM_NUM_PCR 24`

## 4.67.2 Function Documentation

4.67.2.1 `int init_oiap ( struct tc_info * info, struct oiap_sess * s )`

4.67.2.2 `int init_osap ( struct tc_info * info, struct osap_sess * s, const unsigned char * key, uint16_t type, uint32_t handle )`

Create an object specific authorisation protocol (OSAP) session.

4.67.2.3 `struct tc_state* init_tc ( void )`

4.67.2.4 `void shutdown_tc ( struct tc_state * state )`

4.67.2.5 `int tc_chk_loadkey2 ( struct tc_info * info, struct oiap_sess * s, uint32_t parentHandle, uint8_t * parentAuth, size_t parentAuthLen, struct TPM_KEY12 * key, uint32_t * inkeyHandle, uint8_t * contAuthSes )`

4.67.2.6 `int tc_create_key ( struct tc_info * info, struct osap_sess * s, struct TPM_KEY12 ** in_key, uint8_t * dataUsageAuth, size_t len )`

4.67.2.7 `int tc_createwrapkey ( struct tc_info * info, struct osap_sess * s, uint32_t handle, uint8_t * dataUsageAuth, size_t dataUsageAuthLen, struct TPM_KEY12 * key, struct TPM_KEY12 * key2 )`

Because of the variable length arguments, we do not use the `tpm_input_*` structs approach. Passing `==` is allowed.

4.67.2.8 `void tc_flush_all_keys ( struct tc_info * info )`

4.67.2.9 `int tc_flushspecific ( struct tc_info * info, uint32_t handle, uint32_t resourceType )`

4.67.2.10 `void tc_free_key ( struct TPM_KEY12 * key )`

4.67.2.11 `void tc_hexdump ( void * p, size_t len )`

Not a particular TC related function, but heavily used in all modules for debugging.

4.67.2.12 `int tc_listkeys ( struct tc_info * info, uint32_t ** keylist, uint16_t * keylistSize )`

If necessary, this could be generalized to a `get_capability` function.

4.67.2.13 `int tc_loadkey2 ( struct tc_info * info, struct oiap_sess * s, uint32_t parentHandle, uint8_t * parentAuth, size_t parentAuthLen, struct TPM_KEY12 * key, uint32_t * inkeyHandle, uint8_t contAuthSes )`

Caller is responsible for being properly initialized.

4.67.2.14 `void tc_print_pcrs ( struct tc_info * info )`

4.67.2.15 `int tc_reset ( struct tc_info * info )`

RESET is deprecated in 1.2 of spec.

4.67.2.16 `void tc_test ( struct tc_state * state )`

4.67.2.17 `int tc_transmit_cmd ( struct tc_info * info, struct tpm_cmd * cmd )`

Note: not to be used with variable-length commands such as createwrapkey, loadkey2, etc.

4.67.2.18 `int tc_unbind ( struct tc_info * info, struct oiap_sess * s, uint32_t keyHandle, uint8_t * keyAuth, size_t keyAuthLen, uint8_t * inData, uint32_t inDataSize, uint8_t ** outData, uint32_t * outDataSize, uint8_t contAuthSes )`

4.67.2.19 `int tpm_get_random ( struct tc_info * info, unsigned char * buf, uint32_t len )`

4.67.2.20 `int tpm_pcr_extend ( struct tc_info * info, int pcr_idx, const uint8_t * hash )`

## 4.68 xen/stubdom/domt/tpm\_defs.c File Reference

```
#include <mini-os/byteorder.h>
#include "tpm_defs.h"
```

### Variables

- struct `tpm_input_header` `tpm_readpubek_header`
- struct `tpm_input_header` `pcrread_header`
- struct `tpm_input_header` `pcrextend_header`
- struct `tpm_input_header` `oiap_header`
- struct `tpm_input_header` `osap_header`
- struct `tpm_input_header` `getrandom_header`
- struct `tpm_input_header` `reset_header`
- struct `tpm_input_header` `flushspecific_header`

### 4.68.1 Variable Documentation

4.68.1.1 `struct tpm_input_header` `flushspecific_header`

**Initial value:**

```
= {
    .tag = _TPM_TAG_RQU_COMMAND,
    .length = cpu_to_be32(18),
    .ordinal = _TPM_ORD_FLUSHSPECIFIC,
}
```



#### 4.68.1.2 struct tpm\_input\_header getrandom\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(14),  
    .ordinal = _TPM_ORD_GETRANDOM,  
}
```

#### 4.68.1.3 struct tpm\_input\_header oiap\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(10),  
    .ordinal = _TPM_ORD_OIAP,  
}
```

#### 4.68.1.4 struct tpm\_input\_header osap\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(36),  
    .ordinal = _TPM_ORD_OSAP,  
}
```

#### 4.68.1.5 struct tpm\_input\_header pcrextend\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(34),  
    .ordinal = _TPM_ORD_PCR_EXTEND,  
}
```

#### 4.68.1.6 struct tpm\_input\_header pcrread\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(14),  
    .ordinal = _TPM_ORDINAL_PCRREAD,  
}
```

#### 4.68.1.7 struct tpm\_input\_header reset\_header

**Initial value:**

```
= {  
    .tag = _TPM_TAG_RQU_COMMAND,  
    .length = cpu_to_be32(10),  
    .ordinal = _TPM_ORD_RESET,  
}
```

#### 4.68.1.8 struct tpm\_input\_header tpm\_readpubek\_header

##### Initial value:

```
= {
    .tag = _TPM_TAG_RQU_COMMAND,
    .length = cpu_to_be32(30),
    .ordinal = _TPM_ORD_READPUBEK,
}
```

## 4.69 xen/stubdom/domt/tpm\_defs.h File Reference

```
#include <stdbool.h>
#include <inttypes.h>
#include <unistd.h>
#include <os.h>
```

### Data Structures

- struct [tpm\\_input\\_header](#)
- struct [tpm\\_output\\_header](#)
- struct [tpm\\_readpubek\\_params\\_out](#)
- struct [tpm\\_pcrread\\_in](#)
- struct [tpm\\_pcrread\\_out](#)
- struct [tpm\\_pcrextend\\_in](#)
- struct [tpm\\_oiap\\_out](#)
- struct [tpm\\_osap\\_in](#)
- struct [tpm\\_osap\\_out](#)
- struct [tpm\\_getrandom\\_in](#)
- struct [tpm\\_getrandom\\_out](#)
- struct [tpm\\_flushspecific\\_in](#)
- union [tpm\\_cmd\\_header](#)
- union [tpm\\_cmd\\_params](#)
- struct [tpm\\_cmd](#)
- struct [TPM\\_KEY\\_PARMS](#)
- struct [TPM\\_RSA\\_KEY\\_PARMS](#)
- struct [TPM\\_STORE\\_PUBKEY](#)
- struct [TPM\\_KEY12](#)

### Macros

- #define [TPM\\_HEADER\\_SIZE](#) 10
- #define [TPM\\_BUFSIZE](#) 2048
- #define [TPM\\_DIGEST\\_SIZE](#) 20
- #define [TPM\\_ERROR\\_SIZE](#) 10
- #define [TPM\\_RET\\_CODE\\_IDX](#) 6
- #define [TPM\\_INTERNAL\\_RESULT\\_SIZE](#) 200
- #define [MAX\\_RAND\\_SIZE](#) [TPM\\_DIGEST\\_SIZE](#)
- #define [TPM\\_CAP\\_KEY\\_HANDLE](#) 0x00000007
- #define [TPM\\_CAP\\_HANDLE](#) 0x00000014
- #define [TPM\\_TAG\\_RQU\\_COMMAND](#) 0x00C1
- #define [TPM\\_TAG\\_RQU\\_AUTH1\\_COMMAND](#) 0x00C2
- #define [TPM\\_TAG\\_RQU\\_AUTH2\\_COMMAND](#) 0x00C3

- #define `TPM_TAG_RSP_COMMAND` 0x00C4
- #define `TPM_TAG_RSP_AUTH1_COMMAND` 0x00C5
- #define `TPM_TAG_RSP_AUTH2_COMMAND` 0x00C6
- #define `_TPM_TAG_RQU_COMMAND` `cpu_to_be16(0x00C1)`
- #define `_TPM_TAG_RQU_AUTH1_COMMAND` `cpu_to_be16(0x00C2)`
- #define `_TPM_TAG_RQU_AUTH2_COMMAND` `cpu_to_be16(0x00C3)`
- #define `_TPM_ORD_GET_CAP` `cpu_to_be32(101)`
- #define `_TPM_ORD_READPUBKEY` `cpu_to_be32(124)`
- #define `_TPM_ORDINAL_PCRREAD` `cpu_to_be32(21)`
- #define `_TPM_ORD_PCR_EXTEND` `cpu_to_be32(20)`
- #define `_TPM_ORD_OIAP` `cpu_to_be32(10)`
- #define `_TPM_ORD_OSAP` `cpu_to_be32(11)`
- #define `_TPM_ORD_GETRANDOM` `cpu_to_be32(70)`
- #define `_TPM_ORD_RESET` `cpu_to_be32(90)`
- #define `_TPM_ORD_FLUSHSPECIFIC` `cpu_to_be32(186)`
- #define `TPM_ORD_GETCAPABILITY` 101
- #define `TPM_ORD_CREATEWRAPKEY` 31
- #define `TPM_ORD_LOADKEY2` 65
- #define `TPM_ORD_UNBIND` 30
- #define `TPM_RT_KEY` 0x00000001
- #define `TPM_RT_AUTH` 0x00000002
- #define `TPM_RT_HASH` 0x00000003
- #define `TPM_RT_TRANS` 0x00000004
- #define `TPM_RT_CONTEXT` 0x00000005
- #define `TPM_RT_COUNTER` 0x00000006
- #define `TPM_RT_DELEGATE` 0x00000007
- #define `TPM_RT_DAA_TPM` 0x00000008
- #define `TPM_RT_DAA_V0` 0x00000009
- #define `TPM_RT_DAA_V1` 0x0000000A
- #define `TPM_NOSPACE` 17
- #define `TPM_KEY_PARMES_EMPTYSIZE` (4 + 2 + 2 + 4)
- #define `TPM_RSA_KEY_EMPTYSIZE` (4 + 4 + 4)
- #define `TPM_STORE_PUBKEY_EMPTYSIZE` 4
- #define `TPM_KEY12_EMPTYSIZE` (2 + 2 + 2 + 2 + 1 + 4 + 4)
- #define `TPM_KH_SRK` 0x40000000
- #define `_TPM_KH_SRK` `cpu_to_be32(0x40000000)`
- #define `TPM_ET_SRK` 0x04
- #define `_TPM_ET_SRK` `cpu_to_be16(0x04)`
- #define `TPM_KEY_STORAGE` 0x0011
- #define `TPM_KEY_BIND` 0x0014
- #define `_TPM_KEY_BIND` `cpu_to_be16(0x0014)`
- #define `TPM_TAG_KEY12` 0x0028
- #define `_TPM_TAG_KEY12` `cpu_to_be16(0x0028)`
- #define `TPM_AUTH_ALWAYS` 0x01
- #define `TPM_ALG_RSA` 0x01
- #define `_TPM_ALG_RSA` `cpu_to_be32(0x01)`
- #define `TPM_ES_RSAESOAEP_SHA1_MGF1` 0x0003
- #define `_TPM_ES_RSAESOAEP_SHA1_MGF1` `cpu_to_be16(0x0003)`
- #define `TPM_SS_NONE` 0x0001
- #define `_TPM_SS_NONE` `cpu_to_be16(0x0001)`

## Enumerations

- enum `tpm_duration` {  
`TPM_SHORT` = 0, `TPM_MEDIUM` = 1, `TPM_LONG` = 2, `TPM_UNDEFINED`,  
`TPM_SHORT` = 0, `TPM_MEDIUM` = 1, `TPM_LONG` = 2, `TPM_UNDEFINED` }

## Functions

- struct [tpm\\_input\\_header](#) [\\_\\_attribute\\_\\_](#) ((packed))

## Variables

- typedef [\\_\\_attribute\\_\\_](#)
- uint16\_t [tag](#)
- uint32\_t [length](#)
- uint32\_t [ordinal](#)
- uint32\_t [return\\_code](#)
- uint8\_t [algorithm](#) [4]
- uint8\_t [encscheme](#) [2]
- uint8\_t [sigscheme](#) [2]
- uint32\_t [paramsize](#)
- uint8\_t [parameters](#) [12]
- uint32\_t [keysize](#)
- uint8\_t [modulus](#) [256]
- uint8\_t [checksum](#) [20]
- uint32\_t [pcr\\_idx](#)
- uint8\_t [pcr\\_result](#) [TPM\_DIGEST\_SIZE]
- uint8\_t [hash](#) [TPM\_DIGEST\_SIZE]
- uint32\_t [authHandle](#)
- uint8\_t [nonceEven](#) [TPM\_DIGEST\_SIZE]
- uint16\_t [entity\\_type](#)
- uint32\_t [entity\\_value](#)
- uint8\_t [nonce\\_odd\\_osap](#) [TPM\_DIGEST\_SIZE]
- uint32\_t [authhandle](#)
- uint8\_t [nonce\\_even](#) [TPM\_DIGEST\_SIZE]
- uint8\_t [nonce\\_even\\_osap](#) [TPM\_DIGEST\_SIZE]
- uint32\_t [numbytes](#)
- uint32\_t [numrandbytes](#)
- uint8\_t [randbytes](#) [MAX\_RAND\_SIZE]
- uint32\_t [handle](#)
- uint32\_t [resourceType](#)
- [tpm\\_cmd\\_header](#) [header](#)
- [tpm\\_cmd\\_params](#) [params](#)
- struct [tpm\\_input\\_header](#) [tpm\\_readpubek\\_header](#)
- struct [tpm\\_input\\_header](#) [pcrread\\_header](#)
- struct [tpm\\_input\\_header](#) [pcrextend\\_header](#)
- struct [tpm\\_input\\_header](#) [oiap\\_header](#)
- struct [tpm\\_input\\_header](#) [osap\\_header](#)
- struct [tpm\\_input\\_header](#) [getrandom\\_header](#)
- struct [tpm\\_input\\_header](#) [reset\\_header](#)
- struct [tpm\\_input\\_header](#) [flushspecific\\_header](#)

## 4.69.1 Macro Definition Documentation

- 4.69.1.1 `#define _TPM_ALG_RSA cpu_to_be32(0x01)`
- 4.69.1.2 `#define _TPM_ES_RSAESOAEP_SHA1_MGF1 cpu_to_be16(0x0003)`
- 4.69.1.3 `#define _TPM_ET_SRK cpu_to_be16(0x04)`
- 4.69.1.4 `#define _TPM_KEY_BIND cpu_to_be16(0x0014)`
- 4.69.1.5 `#define _TPM_KH_SRK cpu_to_be32(0x40000000)`
- 4.69.1.6 `#define _TPM_ORD_FLUSHSPECIFIC cpu_to_be32(186)`
- 4.69.1.7 `#define _TPM_ORD_GET_CAP cpu_to_be32(101)`
- 4.69.1.8 `#define _TPM_ORD_GETRANDOM cpu_to_be32(70)`
- 4.69.1.9 `#define _TPM_ORD_OIAP cpu_to_be32(10)`
- 4.69.1.10 `#define _TPM_ORD_OSAP cpu_to_be32(11)`
- 4.69.1.11 `#define _TPM_ORD_PCR_EXTEND cpu_to_be32(20)`
- 4.69.1.12 `#define _TPM_ORD_READPUBEK cpu_to_be32(124)`
- 4.69.1.13 `#define _TPM_ORD_RESET cpu_to_be32(90)`
- 4.69.1.14 `#define _TPM_ORDINAL_PCRREAD cpu_to_be32(21)`
- 4.69.1.15 `#define _TPM_SS_NONE cpu_to_be16(0x0001)`
- 4.69.1.16 `#define _TPM_TAG_KEY12 cpu_to_be16(0x0028)`
- 4.69.1.17 `#define _TPM_TAG_RQU_AUTH1_COMMAND cpu_to_be16(0x00C2)`
- 4.69.1.18 `#define _TPM_TAG_RQU_AUTH2_COMMAND cpu_to_be16(0x00C3)`
- 4.69.1.19 `#define _TPM_TAG_RQU_COMMAND cpu_to_be16(0x00C1)`
- 4.69.1.20 `#define MAX_RAND_SIZE TPM_DIGEST_SIZE`
- 4.69.1.21 `#define TPM_ALG_RSA 0x01`
- 4.69.1.22 `#define TPM_AUTH_ALWAYS 0x01`
- 4.69.1.23 `#define TPM_BUFSIZE 2048`
- 4.69.1.24 `#define TPM_CAP_HANDLE 0x00000014`
- 4.69.1.25 `#define TPM_CAP_KEY_HANDLE 0x00000007`
- 4.69.1.26 `#define TPM_DIGEST_SIZE 20`
- 4.69.1.27 `#define TPM_ERROR_SIZE 10`

4.69.1.28 #define TPM\_ES\_RSAESOAEP\_SHA1\_MGF1 0x0003

4.69.1.29 #define TPM\_ET\_SRK 0x04

4.69.1.30 #define TPM\_HEADER\_SIZE 10

4.69.1.31 #define TPM\_INTERNAL\_RESULT\_SIZE 200

4.69.1.32 #define TPM\_KEY12\_EMPTYSIZE (2 + 2 + 2 + 2 + 1 + 4 + 4)

4.69.1.33 #define TPM\_KEY\_BIND 0x0014

4.69.1.34 #define TPM\_KEY\_PARMES\_EMPTYSIZE (4 + 2 + 2 + 4)

4.69.1.35 #define TPM\_KEY\_STORAGE 0x0011

4.69.1.36 #define TPM\_KH\_SRK 0x40000000

4.69.1.37 #define TPM\_NOSPACE 17

4.69.1.38 #define TPM\_ORD\_CREATEWRAPKEY 31

4.69.1.39 #define TPM\_ORD\_GETCAPABILITY 101

4.69.1.40 #define TPM\_ORD\_LOADKEY2 65

4.69.1.41 #define TPM\_ORD\_UNBIND 30

4.69.1.42 #define TPM\_RET\_CODE\_IDX 6

4.69.1.43 #define TPM\_RSA\_KEY\_EMPTYSIZE (4 + 4 + 4)

4.69.1.44 #define TPM\_RT\_AUTH 0x00000002

4.69.1.45 #define TPM\_RT\_CONTEXT 0x00000005

4.69.1.46 #define TPM\_RT\_COUNTER 0x00000006

4.69.1.47 #define TPM\_RT\_DAA\_TPM 0x00000008

4.69.1.48 #define TPM\_RT\_DAA\_V0 0x00000009

4.69.1.49 #define TPM\_RT\_DAA\_V1 0x0000000A

4.69.1.50 #define TPM\_RT\_DELEGATE 0x00000007

4.69.1.51 #define TPM\_RT\_HASH 0x00000003

4.69.1.52 #define TPM\_RT\_KEY 0x00000001

4.69.1.53 #define TPM\_RT\_TRANS 0x00000004

4.69.1.54 #define TPM\_SS\_NONE 0x0001

4.69.1.55 #define TPM\_STORE\_PUBKEY\_EMPTYSIZE 4

4.69.1.56 `#define TPM_TAG_KEY12 0x0028`

4.69.1.57 `#define TPM_TAG_RQU_AUTH1_COMMAND 0x00C2`

4.69.1.58 `#define TPM_TAG_RQU_AUTH2_COMMAND 0x00C3`

4.69.1.59 `#define TPM_TAG_RQU_COMMAND 0x00C1`

4.69.1.60 `#define TPM_TAG_RSP_AUTH1_COMMAND 0x00C5`

4.69.1.61 `#define TPM_TAG_RSP_AUTH2_COMMAND 0x00C6`

4.69.1.62 `#define TPM_TAG_RSP_COMMAND 0x00C4`

## 4.69.2 Enumeration Type Documentation

4.69.2.1 `enum tpm_duration`

Enumerator

***TPM\_SHORT***

***TPM\_MEDIUM***

***TPM\_LONG***

***TPM\_UNDEFINED***

***TPM\_SHORT***

***TPM\_MEDIUM***

***TPM\_LONG***

***TPM\_UNDEFINED***

## 4.69.3 Function Documentation

4.69.3.1 `struct tpm_input_header __attribute__((packed))`

## 4.69.4 Variable Documentation

4.69.4.1 `enum tpm_duration __attribute__((packed))`

4.69.4.2 `uint8_t algorithm[4]`

4.69.4.3 `uint32_t authHandle`

4.69.4.4 `uint32_t authhandle`

4.69.4.5 `uint8_t checksum[20]`

4.69.4.6 `uint8_t encscheme[2]`

4.69.4.7 `uint16_t entity_type`

4.69.4.8 `uint32_t entity_value`

4.69.4.9 `struct tpm_input_header flushspecific_header`

4.69.4.10 `struct tpm_input_header getrandom_header`

- 4.69.4.11 `uint32_t` handle
- 4.69.4.12 `uint8_t` hash[TPM\_DIGEST\_SIZE]
- 4.69.4.13 `tpm_cmd_header` header
- 4.69.4.14 `uint32_t` keysize
- 4.69.4.15 `uint32_t` length
- 4.69.4.16 `uint8_t` modulus[256]
- 4.69.4.17 `uint8_t` nonce\_even[TPM\_DIGEST\_SIZE]
- 4.69.4.18 `uint8_t` nonce\_even\_osap[TPM\_DIGEST\_SIZE]
- 4.69.4.19 `uint8_t` nonce\_odd\_osap[TPM\_DIGEST\_SIZE]
- 4.69.4.20 `uint8_t` nonceEven[TPM\_DIGEST\_SIZE]
- 4.69.4.21 `uint32_t` numbytes
- 4.69.4.22 `uint32_t` numrandbytes
- 4.69.4.23 `struct tpm_input_header` oiap\_header
- 4.69.4.24 `uint32_t` ordinal
- 4.69.4.25 `struct tpm_input_header` osap\_header
- 4.69.4.26 `uint8_t` parameters[12]
- 4.69.4.27 `tpm_cmd_params` params
- 4.69.4.28 `uint32_t` paramsize
- 4.69.4.29 `uint32_t` pcr\_idx
- 4.69.4.30 `uint8_t` pcr\_result[TPM\_DIGEST\_SIZE]
- 4.69.4.31 `struct tpm_input_header` pcrextend\_header
- 4.69.4.32 `struct tpm_input_header` pcrread\_header
- 4.69.4.33 `uint8_t` randbytes[MAX\_RAND\_SIZE]
- 4.69.4.34 `struct tpm_input_header` reset\_header
- 4.69.4.35 `uint32_t` resourceType
- 4.69.4.36 `uint32_t` return\_code
- 4.69.4.37 `uint8_t` sigscheme[2]
- 4.69.4.38 `uint16_t` tag



4.69.4.39 struct tpm\_input\_header tpm\_readpubek\_header

## 4.70 xen/stubdom/domt/xlc.c File Reference

```
#include <xenctrl.h>
#include <os.h>
#include <mini-os/gnttab.h>
#include <sys/mman.h>
#include <unistd.h>
#include <vfs.h>
#include "xlc_intf.h"
#include "main.h"
#include "xlc.h"
#include "caas.h"
```

### Functions

- [libxl\\_ctx \\* xlc\\_init](#) (void)
- void [xlc\\_listen](#) (libxl\_ctx \*ctx)
- int [do\\_create](#) (libxl\_ctx \*ctx, libxl\_domain\_config \*d\_config, uint32\_t \*domid)

### 4.70.1 Function Documentation

4.70.1.1 int [do\\_create](#) ( libxl\_ctx \* *ctx*, libxl\_domain\_config \* *d\_config*, uint32\_t \* *domid* )

Modeled after libxl\_create.c:do\_domain\_create(), with the logic removed which cannot function (yet) with a stubdom domain builder.

4.70.1.2 libxl\_ctx\* [xlc\\_init](#) ( void )

4.70.1.3 void [xlc\\_listen](#) ( libxl\_ctx \* *ctx* )

## 4.71 xen/tools/libxlc/xlc.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <malloc.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/select.h>
#include <sys/mman.h>
#include <assert.h>
#include <xlc_intf.h>
#include "xlc_util.h"
#include "xlc.h"
#include <libxl_internal.h>
```

### Macros

- #define [TIMEOUT\\_SEC\\_INIT](#) 30
- #define [TIMEOUT\\_SEC](#) 100 /\* normal timeout is higher, due to TPM delay involved! \*/

- `#define DOMID_T 1`

## Functions

- `domid_t read_dyn_domid` (void)
- void `print_libxl_domain_config` (int domid, `libxl_domain_config` \*d\_config, `libxl_device_model_info` \*dm\_info)
- int `xc_command` (struct `xs_handle` \*xsh, struct `xc_cmd` \*cmd)
- int `xc_domain_make` (`libxl_ctx` \*ctx, `libxl_domain_create_info` \*info, uint32\_t \*domid)
- int `xc_domain_build` (`libxl_ctx` \*ctx, `libxl_domain_build_info` \*info, uint32\_t domid, `libxl_domain_build_state` \*state)
- int `xc_do_domain_create` (`libxl_ctx` \*ctx, `libxl_domain_config` \*d\_config, `libxl_console_ready` cb, void \*priv, uint32\_t \*domid\_out, int restore\_fd)

### 4.71.1 Macro Definition Documentation

4.71.1.1 `#define DOMID_T 1`

4.71.1.2 `#define TIMEOUT_SEC 100` /\* normal timeout is higher, due to TPM delay involved! \*/

4.71.1.3 `#define TIMEOUT_SEC_INIT 30`

### 4.71.2 Function Documentation

4.71.2.1 void `print_libxl_domain_config` ( int *domid*, `libxl_domain_config` \* *d\_config*, `libxl_device_model_info` \* *dm\_info* )

4.71.2.2 `domid_t read_dyn_domid` ( void )

FIXME: the logging and erroring here should use the libxl-offered logger.

4.71.2.3 int `xc_command` ( struct `xs_handle` \* *xsh*, struct `xc_cmd` \* *cmd* )

4.71.2.4 int `xc_do_domain_create` ( `libxl_ctx` \* *ctx*, `libxl_domain_config` \* *d\_config*, `libxl_console_ready` *cb*, void \* *priv*, uint32\_t \* *domid\_out*, int *restore\_fd* )

4.71.2.5 int `xc_domain_build` ( `libxl_ctx` \* *ctx*, `libxl_domain_build_info` \* *info*, uint32\_t *domid*, `libxl_domain_build_state` \* *state* )

4.71.2.6 int `xc_domain_make` ( `libxl_ctx` \* *ctx*, `libxl_domain_create_info` \* *info*, uint32\_t \* *domid* )

Implementation of stub functions which libxl will call.

## 4.72 xen/stubdom/domt/xc.h File Reference

```
#include <xenctrl.h>
#include <os.h>
#include <libxl.h>
#include <libxl_internal.h>
#include <xc_util.h>
```

## Functions

- `libxl_ctx * xlc_init (void)`
- `void xlc_listen (libxl_ctx *ctx)`
- `int do_create (libxl_ctx *ctx, libxl_domain_config *d_config, uint32_t *domid)`

### 4.72.1 Function Documentation

4.72.1.1 `int do_create ( libxl_ctx * ctx, libxl_domain_config * d_config, uint32_t * domid )`

Modeled after `libxl_create.c:do_domain_create()`, with the logic removed which cannot function (yet) with a stubdom domain builder.

4.72.1.2 `libxl_ctx* xlc_init ( void )`

4.72.1.3 `void xlc_listen ( libxl_ctx * ctx )`

## 4.73 xen/tools/libxlc/xlc.h File Reference

```
#include <errno.h>
#include <libxl.h>
#include <libxl_utils.h>
```

## Macros

- `#define XLC_FAILED -EINVAL`

## Functions

- `int xlc_domain_make (libxl_ctx *ctx, libxl_domain_create_info *info, uint32_t *domid)`
- `int xlc_domain_build (libxl_ctx *ctx, libxl_domain_build_info *info, uint32_t domid, libxl_domain_build_state *state)`
- `int xlc_do_domain_create (libxl_ctx *ctx, libxl_domain_config *d_config, libxl_console_ready cb, void *priv, uint32_t *domid_out, int restore_fd)`

### 4.73.1 Macro Definition Documentation

4.73.1.1 `#define XLC_FAILED -EINVAL`

### 4.73.2 Function Documentation

4.73.2.1 `int xlc_do_domain_create ( libxl_ctx * ctx, libxl_domain_config * d_config, libxl_console_ready cb, void * priv, uint32_t * domid_out, int restore_fd )`

4.73.2.2 `int xlc_domain_build ( libxl_ctx * ctx, libxl_domain_build_info * info, uint32_t domid, libxl_domain_build_state * state )`

4.73.2.3 `int xlc_domain_make ( libxl_ctx * ctx, libxl_domain_create_info * info, uint32_t * domid )`

Implementation of stub functions which libxl will call.

## 4.74 xen/stubdom/domt/xlc\_intf.h File Reference

```
#include <inttypes.h>
```

### Data Structures

- struct [xlc\\_cmd\\_ping](#)
- struct [xlc\\_cmd\\_getpage](#)
- struct [xlc\\_cmd\\_retpage](#)
- struct [xlc\\_cmd\\_dommake](#)
- struct [xlc\\_cmd\\_dombuild](#)
- struct [xlc\\_cmd\\_domcreate](#)
- struct [xlc\\_cmd](#)

### Macros

- #define [XLC\\_PATH](#) "/local/domain/0/domt/xlc"

### Enumerations

- enum [xlc\\_cmdtype](#) {  
[XLC\\_PING](#) = 0, [XLC\\_GETPAGE](#) = 1, [XLC\\_RETPAGE](#) = 2, [XLC\\_DOMMAKE](#) = 3,  
[XLC\\_DOMBUILD](#) = 4, [XLC\\_DOMCREATE](#) = 5 }

#### 4.74.1 Macro Definition Documentation

4.74.1.1 #define [XLC\\_PATH](#) "/local/domain/0/domt/xlc"

#### 4.74.2 Enumeration Type Documentation

4.74.2.1 enum [xlc\\_cmdtype](#)

##### Enumerator

***[XLC\\_PING](#)***  
***[XLC\\_GETPAGE](#)***  
***[XLC\\_RETPAGE](#)***  
***[XLC\\_DOMMAKE](#)***  
***[XLC\\_DOMBUILD](#)***  
***[XLC\\_DOMCREATE](#)***

## 4.75 xen/tools/libxlc/include/xlc\_intf.h File Reference

## 4.76 xen/stubdom/domt/xlc\_util.c File Reference

```
#include <string.h>
#include <errno.h>
#include <libxl_internal.h>
#include "xlc_util.h"
```

## Macros

- `#define CHK_XLC_MEMCPY_OUT(a, b, c, d) {if ((rc=xlc_memcpy_out(a,b,c,d))) goto error;}`
- `#define CHK_XLC_MEMCPY_IN(a, b, c, d) {if ((rc=xlc_memcpy_in(a,b,c,d))) goto error;}`
- `#define CHK_XLC_STRCPY_OUT(a, b, c) {if ((rc=xlc_strcpy_out(a,b,c))) goto error;}`
- `#define CHK_XLC_STRACPY_IN(a, b, c) {if ((rc=xlc_stracpy_in(a,b,c))) goto error;}`
- `#define CHK_XLC_DARRAY_OUT(a, b, c) {if ((rc=xlc_darray_out(a,b,c))) goto error;}`
- `#define CHK_XLC_DARRAY_IN(a, b, c) {if ((rc=xlc_darray_in(a,b,c))) goto error;}`
- `#define CHK_XLC_SERIALIZE_DISKS(a, b, c, d) {if ((rc=xlc_serialize_disks(a,b,c,d))) goto error;}`
- `#define CHK_XLC_SERIALIZE_VIFS(a, b, c, d) {if ((rc=xlc_serialize_vifs(a,b,c,d))) goto error;}`
- `#define CHK_XLC_DESERIALIZE_DISKS(a, b, c, d) {if ((rc=xlc_deserialize_disks(a,b,c,d))) goto error;}`
- `#define CHK_XLC_DESERIALIZE_VIFS(a, b, c, d) {if ((rc=xlc_deserialize_vifs(a,b,c,d))) goto error;}`
- `#define CHK_MEMCMP(a, b) {if (memcmp(&a, &b, sizeof(a))) goto error;}`
- `#define CHK_STRCMP(a, b) {if (a && b && strcmp(a, b)) goto error;}`

## Functions

- `int serialize_domain_create_info (void *buf, size_t len, libxl_domain_create_info *info)`
- `int deserialize_domain_create_info (void *buf, size_t len, libxl_domain_create_info *info)`
- `int serialize_domain_build_info (void *buf, size_t len, libxl_domain_build_info *info, libxl_domain_build_state *state)`
- `int deserialize_domain_build_info (void *buf, size_t len, libxl_domain_build_info *info, libxl_domain_build_state *state)`
- `int serialize_domain_config (void *buf, size_t len, libxl_domain_config *info)`
- `int deserialize_domain_config (void *buf, size_t len, libxl_domain_config *info)`

### 4.76.1 Macro Definition Documentation

4.76.1.1 `#define CHK_MEMCMP( a, b ) {if (memcmp(&a, &b, sizeof(a))) goto error;}`

4.76.1.2 `#define CHK_STRCMP( a, b ) {if (a && b && strcmp(a, b)) goto error;}`

4.76.1.3 `#define CHK_XLC_DARRAY_IN( a, b, c ) {if ((rc=xlc_darray_in(a,b,c))) goto error;}`

4.76.1.4 `#define CHK_XLC_DARRAY_OUT( a, b, c ) {if ((rc=xlc_darray_out(a,b,c))) goto error;}`

4.76.1.5 `#define CHK_XLC_DESERIALIZE_DISKS( a, b, c, d ) {if ((rc=xlc_deserialize_disks(a,b,c,d))) goto error;}`

4.76.1.6 `#define CHK_XLC_DESERIALIZE_VIFS( a, b, c, d ) {if ((rc=xlc_deserialize_vifs(a,b,c,d))) goto error;}`

4.76.1.7 `#define CHK_XLC_MEMCPY_IN( a, b, c, d ) {if ((rc=xlc_memcpy_in(a,b,c,d))) goto error;}`

4.76.1.8 `#define CHK_XLC_MEMCPY_OUT( a, b, c, d ) {if ((rc=xlc_memcpy_out(a,b,c,d))) goto error;}`

4.76.1.9 `#define CHK_XLC_SERIALIZE_DISKS( a, b, c, d ) {if ((rc=xlc_serialize_disks(a,b,c,d))) goto error;}`

4.76.1.10 `#define CHK_XLC_SERIALIZE_VIFS( a, b, c, d ) {if ((rc=xlc_serialize_vifs(a,b,c,d))) goto error;}`

4.76.1.11 `#define CHK_XLC_STRACPY_IN( a, b, c ) {if ((rc=xlc_stracpy_in(a,b,c))) goto error;}`

4.76.1.12 `#define CHK_XLC_STRCPY_OUT( a, b, c ) {if ((rc=xlc_strcpy_out(a,b,c))) goto error;}`

### 4.76.2 Function Documentation

4.76.2.1 `int deserialize_domain_build_info ( void * buf, size_t len, libxl_domain_build_info * info, libxl_domain_build_state * state )`

4.76.2.2 `int deserialize_domain_config ( void * buf, size_t len, libxl_domain_config * info )`

4.76.2.3 `int deserialize_domain_create_info ( void * buf, size_t len, libxl_domain_create_info * info )`

NB: strings are `strdup()`'ed instead of setting the pointer to the string in the shared page (which would waste a lot of memory). caller must be remember to free these strings struct on destruction.

4.76.2.4 `int serialize_domain_build_info ( void * buf, size_t len, libxl_domain_build_info * info, libxl_domain_build_state * state )`

hvm not supported, nor is a pre-mapped image. (i.e., kernel.data and ramdisk.data are set to null!)

4.76.2.5 `int serialize_domain_config ( void * buf, size_t len, libxl_domain_config * info )`

4.76.2.6 `int serialize_domain_create_info ( void * buf, size_t len, libxl_domain_create_info * info )`

## 4.77 xen/tools/libxlc/xlc\_util.c File Reference

## 4.78 xen/stubdom/domt/xlc\_util.h File Reference

```
#include <stdlib.h>
#include <libxl.h>
#include <libxl_utils.h>
```

### Functions

- `int serialize\_domain\_create\_info (void *buf, size_t len, libxl_domain_create_info *info)`
- `int deserialize\_domain\_create\_info (void *buf, size_t len, libxl_domain_create_info *info)`
- `int serialize\_domain\_build\_info (void *buf, size_t len, libxl_domain_build_info *info, libxl_domain_build_state *state)`
- `int deserialize\_domain\_build\_info (void *buf, size_t len, libxl_domain_build_info *info, libxl_domain_build_state *state)`
- `int deserialize\_domain\_config (void *buf, size_t len, libxl_domain_config *info)`
- `int serialize\_domain\_config (void *buf, size_t len, libxl_domain_config *info)`

### 4.78.1 Function Documentation

4.78.1.1 `int deserialize_domain_build_info ( void * buf, size_t len, libxl_domain_build_info * info, libxl_domain_build_state * state )`

4.78.1.2 `int deserialize_domain_config ( void * buf, size_t len, libxl_domain_config * info )`

4.78.1.3 `int deserialize_domain_create_info ( void * buf, size_t len, libxl_domain_create_info * info )`

NB: strings are `strdup()`'ed instead of setting the pointer to the string in the shared page (which would waste a lot of memory). caller must be remember to free these strings struct on destruction.

4.78.1.4 `int serialize_domain_build_info ( void * buf, size_t len, libxl_domain_build_info * info, libxl_domain_build_state * state )`

hvm not supported, nor is a pre-mapped image. (i.e., kernel.data and ramdisk.data are set to null!)

4.78.1.5 `int serialize_domain_config ( void * buf, size_t len, libxl_domain_config * info )`

4.78.1.6 `int serialize_domain_create_info ( void * buf, size_t len, libxl_domain_create_info * info )`

## 4.79 xen/tools/libxlc/xlc\_util.h File Reference

## 4.80 xen/stubdom/vtpm/compat/big\_endian.h File Reference

### Macros

- `#define __cpu_to_le64(x) (__swab64((x)))`
- `#define __le64_to_cpu(x) (__swab64((uint64_t)(x)))`
- `#define __cpu_to_le32(x) (__swab32((x)))`
- `#define __le32_to_cpu(x) (__swab32((uint32_t)(x)))`
- `#define __cpu_to_le16(x) (__swab16((x)))`
- `#define __le16_to_cpu(x) (__swab16((uint16_t)(x)))`
- `#define __cpu_to_be64(x) ((uint64_t)(x))`
- `#define __be64_to_cpu(x) ((uint64_t)(x))`
- `#define __cpu_to_be32(x) ((uint32_t)(x))`
- `#define __be32_to_cpu(x) ((uint32_t)(x))`
- `#define __cpu_to_be16(x) ((uint16_t)(x))`
- `#define __be16_to_cpu(x) ((uint16_t)(x))`

### 4.80.1 Macro Definition Documentation

4.80.1.1 `#define __be16_to_cpu( x ) ((uint16_t)(x))`

4.80.1.2 `#define __be32_to_cpu( x ) ((uint32_t)(x))`

4.80.1.3 `#define __be64_to_cpu( x ) ((uint64_t)(x))`

4.80.1.4 `#define __cpu_to_be16( x ) ((uint16_t)(x))`

4.80.1.5 `#define __cpu_to_be32( x ) ((uint32_t)(x))`

4.80.1.6 `#define __cpu_to_be64( x ) ((uint64_t)(x))`

4.80.1.7 `#define __cpu_to_le16( x ) (__swab16((x)))`

4.80.1.8 `#define __cpu_to_le32( x ) (__swab32((x)))`

4.80.1.9 `#define __cpu_to_le64( x ) (__swab64((x)))`

4.80.1.10 `#define __le16_to_cpu( x ) (__swab16((uint16_t)(x)))`

4.80.1.11 `#define __le32_to_cpu( x ) (__swab32((uint32_t)(x)))`

4.80.1.12 `#define __le64_to_cpu( x ) (__swab64((uint64_t)(x)))`

## 4.81 xen/stubdom/vtpm/compat/ endian\_test.c File Reference

```
#include <stdio.h>
```

### Functions

- int [main](#) (void)

#### 4.81.1 Function Documentation

4.81.1.1 int main ( void )

## 4.82 xen/stubdom/vtpm/compat/little\_endian.h File Reference

### Macros

- #define [\\_\\_cpu\\_to\\_le64](#)(x) ((uint64\_t)(x))
- #define [\\_\\_le64\\_to\\_cpu](#)(x) ((uint64\_t)(x))
- #define [\\_\\_cpu\\_to\\_le32](#)(x) ((uint32\_t)(x))
- #define [\\_\\_le32\\_to\\_cpu](#)(x) ((uint32\_t)(x))
- #define [\\_\\_cpu\\_to\\_le16](#)(x) ((uint16\_t)(x))
- #define [\\_\\_le16\\_to\\_cpu](#)(x) ((uint16\_t)(x))
- #define [\\_\\_cpu\\_to\\_be64](#)(x) ([\\_\\_swab64](#)((x)))
- #define [\\_\\_be64\\_to\\_cpu](#)(x) ([\\_\\_swab64](#)((uint64\_t)(x)))
- #define [\\_\\_cpu\\_to\\_be32](#)(x) ([\\_\\_swab32](#)((x)))
- #define [\\_\\_be32\\_to\\_cpu](#)(x) ([\\_\\_swab32](#)((uint32\_t)(x)))
- #define [\\_\\_cpu\\_to\\_be16](#)(x) ([\\_\\_swab16](#)((x)))
- #define [\\_\\_be16\\_to\\_cpu](#)(x) ([\\_\\_swab16](#)((uint16\_t)(x)))

#### 4.82.1 Macro Definition Documentation

4.82.1.1 #define [\\_\\_be16\\_to\\_cpu](#)( x ) ([\\_\\_swab16](#)((uint16\_t)(x)))

4.82.1.2 #define [\\_\\_be32\\_to\\_cpu](#)( x ) ([\\_\\_swab32](#)((uint32\_t)(x)))

4.82.1.3 #define [\\_\\_be64\\_to\\_cpu](#)( x ) ([\\_\\_swab64](#)((uint64\_t)(x)))

4.82.1.4 #define [\\_\\_cpu\\_to\\_be16](#)( x ) ([\\_\\_swab16](#)((x)))

4.82.1.5 #define [\\_\\_cpu\\_to\\_be32](#)( x ) ([\\_\\_swab32](#)((x)))

4.82.1.6 #define [\\_\\_cpu\\_to\\_be64](#)( x ) ([\\_\\_swab64](#)((x)))

4.82.1.7 #define [\\_\\_cpu\\_to\\_le16](#)( x ) ((uint16\_t)(x))

4.82.1.8 #define [\\_\\_cpu\\_to\\_le32](#)( x ) ((uint32\_t)(x))

4.82.1.9 #define [\\_\\_cpu\\_to\\_le64](#)( x ) ((uint64\_t)(x))

4.82.1.10 #define [\\_\\_le16\\_to\\_cpu](#)( x ) ((uint16\_t)(x))



4.82.1.11 `#define __le32_to_cpu( x ) ((uint32_t)(x))`

4.82.1.12 `#define __le64_to_cpu( x ) ((uint64_t)(x))`

## 4.83 xen/stubdom/vtpm/compat/swab.h File Reference

```
#include <stdint.h>
```

### Macros

- `#define __swab16(x)`
- `#define __swab32(x)`
- `#define __swab64(x)`

#### 4.83.1 Macro Definition Documentation

4.83.1.1 `#define __swab16( x )`

**Value:**

```
((uint16_t) (
    (((uint16_t)(x) & (uint16_t)0x00ffU) << 8) |
    (((uint16_t)(x) & (uint16_t)0xff00U) >> 8)))
```

casts are necessary for constants, because we never know how for sure how U/UL/ULL map to \_\_u16, \_\_u32, \_\_u64. At least not in a portable way.

4.83.1.2 `#define __swab32( x )`

**Value:**

```
((uint32_t) (
    (((uint32_t)(x) & (uint32_t)0x000000ffUL) << 24) |
    (((uint32_t)(x) & (uint32_t)0x0000ff00UL) << 8) |
    (((uint32_t)(x) & (uint32_t)0x00ff0000UL) >> 8) |
    (((uint32_t)(x) & (uint32_t)0xff000000UL) >> 24)))
```

4.83.1.3 `#define __swab64( x )`

**Value:**

```
((uint64_t) (
    (((uint64_t)(x) & (uint64_t)0x00000000000000ffULL) << 56) |
    (((uint64_t)(x) & (uint64_t)0x000000000000ff00ULL) << 40) |
    (((uint64_t)(x) & (uint64_t)0x0000000000ff0000ULL) << 24) |
    (((uint64_t)(x) & (uint64_t)0x00000000ff000000ULL) << 8) |
    (((uint64_t)(x) & (uint64_t)0x000000ff00000000ULL) >> 8) |
    (((uint64_t)(x) & (uint64_t)0x0000ff0000000000ULL) >> 24) |
    (((uint64_t)(x) & (uint64_t)0x00ff000000000000ULL) >> 40) |
    (((uint64_t)(x) & (uint64_t)0xff00000000000000ULL) >> 56)))
```

## 4.84 xen/stubdom/vtpm/vtpm.c File Reference

```
#include <stdio.h>
#include <stdint.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <syslog.h>
#include <errno.h>
#include <sys/time.h>
#include <xen/xen.h>
#include <tpmback.h>
#include <tpmfront.h>
#include "tpm_emulator_config.h"
#include "tpm/tpm_emulator.h"
#include "vtpm.h"
#include "vtpm_cmd.h"
#include "vtpmblk.h"
```

### Functions

- void [tpm\\_log](#) (int priority, const char \*fmt,...)
- void [get\\_rand\\_bytes\\_rand](#) (uint8\_t \*buf, size\_t nbytes)
- void [tpm\\_get\\_extern\\_random\\_bytes](#) (void \*buf, size\_t nbytes)
- uint64\_t [tpm\\_get\\_ticks](#) (void)
- int [tpm\\_read\\_from\\_file](#) (uint8\_t \*\*data, size\_t \*data\_length)
- int [tpm\\_write\\_to\\_file](#) (uint8\_t \*data, size\_t data\_length)
- int [init\\_random](#) (void)
- int [parse\\_cmd\\_line](#) (int argc, char \*\*argv)
- int [main](#) (int argc, char \*\*argv)

### Variables

- struct [Opt\\_args](#) [opt\\_args](#)
- struct [tpmfront\\_dev](#) \* [tpmfront\\_dev](#)

#### 4.84.1 Function Documentation

4.84.1.1 void [get\\_rand\\_bytes\\_rand](#) ( uint8\_t \* *buf*, size\_t *nbytes* )

4.84.1.2 int [init\\_random](#) ( void )

4.84.1.3 int [main](#) ( int *argc*, char \*\* *argv* )

4.84.1.4 int [parse\\_cmd\\_line](#) ( int *argc*, char \*\* *argv* )

4.84.1.5 void [tpm\\_get\\_extern\\_random\\_bytes](#) ( void \* *buf*, size\_t *nbytes* )

4.84.1.6 uint64\_t [tpm\\_get\\_ticks](#) ( void )

4.84.1.7 void [tpm\\_log](#) ( int *priority*, const char \* *fmt*, ... )

4.84.1.8 int [tpm\\_read\\_from\\_file](#) ( uint8\_t \*\* *data*, size\_t \* *data\_length* )

4.84.1.9 `int tpm_write_to_file ( uint8_t * data, size_t data_length )`

## 4.84.2 Variable Documentation

4.84.2.1 `struct Opt_args opt_args`

**Initial value:**

```
= {
    .mode = MODE_PVM,
    .startup = ST_CLEAR,
    .loglevel = LOG_INFO,
    .entropysrc = ENT_TPM,
}
```

Global commandline options - default values

4.84.2.2 `struct tpmfront_dev* tpmfront_dev`

## 4.85 xen/stubdom/vtpm/vtpm.h File Reference

### Data Structures

- [struct Opt\\_args](#)

### Macros

- `#define VERS_CMD "\x00\xC1\x00\x00\x00\x16\x00\x00\x00\x65\x00\x00\x00\x05\x00\x00\x00\x04\x00\x00\x01\x03"`
- `#define VERS_CMD_LEN 22`

### Variables

- [struct Opt\\_args opt\\_args](#)

## 4.85.1 Macro Definition Documentation

4.85.1.1 `#define VERS_CMD "\x00\xC1\x00\x00\x00\x16\x00\x00\x00\x65\x00\x00\x00\x05\x00\x00\x00\x04\x00\x00\x01\x03"`

For testing

4.85.1.2 `#define VERS_CMD_LEN 22`

## 4.85.2 Variable Documentation

4.85.2.1 `struct Opt_args opt_args`

Global commandline options - default values

## 4.86 xen/stubdom/vtpm/vtpm\_cmd.c File Reference

```
#include <types.h>
#include <xen/xen.h>
#include <mm.h>
#include <gnttab.h>
#include "tpm/tpm_marshallling.h"
#include "vtpm_manager.h"
#include "vtpm_cmd.h"
```

### Data Structures

- struct [shpage](#)

### Macros

- #define [TRYFAILGOTO](#)(C)
- #define [TRYFAILGOTOMSG](#)(C, msg)
- #define [CHECKSTATUSGOTO](#)(ret, fname)
- #define [ERR\\_MALFORMED](#) "Malformed response from backend"
- #define [ERR\\_TPMFRONT](#) "Error sending command through frontend device"

### Typedefs

- typedef struct [shpage](#) [shpage\\_t](#)

### Functions

- [TPM\\_RESULT VTPM\\_GetRandom](#) (struct [tpmfront\\_dev](#) \*[tpmfront\\_dev](#), [BYTE](#) \*[bytes](#), [UINT32](#) [numbytes](#))
- [TPM\\_RESULT VTPM\\_LoadKey](#) (struct [tpmfront\\_dev](#) \*[tpmfront\\_dev](#), [uint8\\_t](#) \*\*[data](#), [size\\_t](#) \*[data\\_length](#))
- [TPM\\_RESULT VTPM\\_SaveKey](#) (struct [tpmfront\\_dev](#) \*[tpmfront\\_dev](#), [uint8\\_t](#) \*[data](#), [size\\_t](#) [data\\_length](#))

#### 4.86.1 Macro Definition Documentation

##### 4.86.1.1 #define CHECKSTATUSGOTO( ret, fname )

###### Value:

```
if((ret) != TPM_SUCCESS) { \
    error("%s failed with error code (%lu)", fname, (unsigned long) ret); \
    status = ord; \
    goto abort_egress; \
}
```

##### 4.86.1.2 #define ERR\_MALFORMED "Malformed response from backend"

##### 4.86.1.3 #define ERR\_TPMFRONT "Error sending command through frontend device"

##### 4.86.1.4 #define TRYFAILGOTO( C )

###### Value:

```

if ((C)) { \
    status = TPM_FAIL; \
    goto abort_egress; \
}

```

#### 4.86.1.5 #define TRYFAILGOTOMSG( C, msg )

**Value:**

```

if ((C)) { \
    status = TPM_FAIL; \
    error(msg); \
    goto abort_egress; \
}

```

### 4.86.2 Typedef Documentation

#### 4.86.2.1 typedef struct shpage shpage\_t

### 4.86.3 Function Documentation

#### 4.86.3.1 TPM\_RESULT VTPM\_GetRandom ( struct tpmfront\_dev \* tpmfront\_dev, BYTE \* bytes, UINT32 numbytes )

Request random bytes from hardware tpm, returns 0 on success

#### 4.86.3.2 TPM\_RESULT VTPM\_LoadKey ( struct tpmfront\_dev \* tpmfront\_dev, uint8\_t \*\* data, size\_t \* data\_length )

Retreive 256 bit AES encryption key from manager

#### 4.86.3.3 TPM\_RESULT VTPM\_SaveKey ( struct tpmfront\_dev \* tpmfront\_dev, uint8\_t \* data, size\_t data\_length )

Manager securely saves our 256 bit AES encryption key

## 4.87 xen/stubdom/vtpm/vtpm\_cmd.h File Reference

```

#include <tpmfront.h>
#include "tpm_emulator_config.h"
#include "tpm/tpm_emulator.h"
#include "tpm/tpm_structures.h"

```

### Functions

- [TPM\\_RESULT VTPM\\_GetRandom](#) (struct tpmfront\_dev \*tpmfront\_dev, BYTE \*bytes, UINT32 numbytes)
- [TPM\\_RESULT VTPM\\_SaveNVM](#) (struct tpmfront\_dev \*tpmfront\_dev, BYTE \*data, size\_t data\_length)
- [TPM\\_RESULT VTPM\\_LoadNVM](#) (struct tpmfront\_dev \*tpmfront\_dev, BYTE \*\*data, size\_t \*data\_length)
- [TPM\\_RESULT VTPM\\_GetNVMSize](#) (struct tpmfront\_dev \*tpmfront\_dev, UINT32 \*size)
- [TPM\\_RESULT VTPM\\_LoadKey](#) (struct tpmfront\_dev \*tpmfront\_dev, uint8\_t \*\*data, size\_t \*data\_length)
- [TPM\\_RESULT VTPM\\_SaveKey](#) (struct tpmfront\_dev \*tpmfront\_dev, uint8\_t \*data, size\_t data\_length)

### 4.87.1 Function Documentation

#### 4.87.1.1 TPM\_RESULT VTPM\_GetNVMSize ( struct tpmfront\_dev \* *tpmfront\_dev*, UINT32 \* *size* )

Retreive the size of the NVM data

#### 4.87.1.2 TPM\_RESULT VTPM\_GetRandom ( struct tpmfront\_dev \* *tpmfront\_dev*, BYTE \* *bytes*, UINT32 *numbytes* )

Request random bytes from hardware tpm, returns 0 on success

#### 4.87.1.3 TPM\_RESULT VTPM\_LoadKey ( struct tpmfront\_dev \* *tpmfront\_dev*, uint8\_t \*\* *data*, size\_t \* *data\_length* )

Retreive 256 bit AES encryption key from manager

#### 4.87.1.4 TPM\_RESULT VTPM\_LoadNVM ( struct tpmfront\_dev \* *tpmfront\_dev*, BYTE \*\* *data*, size\_t \* *data\_length* )

Retreive NVM data from manager

#### 4.87.1.5 TPM\_RESULT VTPM\_SaveKey ( struct tpmfront\_dev \* *tpmfront\_dev*, uint8\_t \* *data*, size\_t *data\_length* )

Manager securely saves our 256 bit AES encryption key

#### 4.87.1.6 TPM\_RESULT VTPM\_SaveNVM ( struct tpmfront\_dev \* *tpmfront\_dev*, BYTE \* *data*, size\_t *data\_length* )

Send NVM data to manager for secure storage

## 4.88 xen/stubdom/vtpm/vtpmblk.c File Reference

```
#include <blkfront.h>
#include <fcntl.h>
#include <xmalloc.h>
#include <asm/byteorder.h>
#include "vtpmblk.h"
#include "tpm_emulator_config.h"
#include "tpm/tpm_marshallling.h"
#include "vtpm_cmd.h"
#include "polarssl/aes.h"
```

### Macros

- #define SECTORS\_PER\_BLOCK 8
- #define BLKSZ 16
- #define KEYSZ 32
- #define write\_finish() rwop\_finish(1)
- #define read\_finish() rwop\_finish(0)
- #define write\_data(data, len) rw\_data(data, len, 1)
- #define read\_data(data, len) rw\_data(data, len, 0)

## Functions

- int `init_vtpmbk` (struct `tpmfront_dev` \*`tpmfront_dev`)
- void `shutdown_vtpmbk` (void)
- int `write_vtpmbk_raw` (uint8\_t \*`data`, size\_t `data_length`)
- int `read_vtpmbk_raw` (uint8\_t \*\*`data`, size\_t \*`data_length`)
- int `encrypt_vtpmbk` (uint8\_t \*`clear`, size\_t `clear_len`, uint8\_t \*\*`cipher`, size\_t \*`cipher_len`, uint8\_t \*\*`symkey`)
- int `decrypt_vtpmbk` (uint8\_t \*`cipher`, size\_t `cipher_len`, uint8\_t \*\*`clear`, size\_t \*`clear_len`, uint8\_t \*`symkey`)
- int `write_vtpmbk` (struct `tpmfront_dev` \*`tpmfront_dev`, uint8\_t \*`data`, size\_t `data_length`)
- int `read_vtpmbk` (struct `tpmfront_dev` \*`tpmfront_dev`, uint8\_t \*\*`data`, size\_t \*`data_length`)

## Variables

- struct `blkfront_dev` \* `blkdev` = NULL
- struct `blkfront_info` `blkinfo`
- struct `blkfront_aiocb` `aiocb`
- unsigned long long `disksize` = 0
- unsigned int `blocksize`
- off\_t `blkoff`

### 4.88.1 Macro Definition Documentation

4.88.1.1 `#define` `BLKSZ` 16

Encryption key and block sizes

4.88.1.2 `#define` `KEYSZ` 32

4.88.1.3 `#define` `read_data( data, len )` `rw_data(data, len, 0)`

4.88.1.4 `#define` `read_finish( )` `rwop_finish(0)`

4.88.1.5 `#define` `SECTORS_PER_BLOCK` 8

4.88.1.6 `#define` `write_data( data, len )` `rw_data(data, len, 1)`

4.88.1.7 `#define` `write_finish( )` `rwop_finish(1)`

### 4.88.2 Function Documentation

4.88.2.1 int `decrypt_vtpmbk` ( uint8\_t \* *cipher*, size\_t *cipher\_len*, uint8\_t \*\* *clear*, size\_t \* *clear\_len*, uint8\_t \* *symkey* )

4.88.2.2 int `encrypt_vtpmbk` ( uint8\_t \* *clear*, size\_t *clear\_len*, uint8\_t \*\* *cipher*, size\_t \* *cipher\_len*, uint8\_t \*\* *symkey* )

4.88.2.3 int `init_vtpmbk` ( struct `tpmfront_dev` \* *tpmfront\_dev* )

4.88.2.4 int `read_vtpmbk` ( struct `tpmfront_dev` \* *tpmfront\_dev*, uint8\_t \*\* *data*, size\_t \* *data\_length* )

Reads, Decrypts, and returns data from blk device

4.88.2.5 `int read_vtpmbk_raw ( uint8_t ** data, size_t * data_length )`

4.88.2.6 `void shutdown_vtpmbk ( void )`

4.88.2.7 `int write_vtpmbk ( struct tpmfront_dev * tpmfront_dev, uint8_t * data, size_t data_length )`

Encrypts and writes data to blk device

4.88.2.8 `int write_vtpmbk_raw ( uint8_t * data, size_t data_length )`

### 4.88.3 Variable Documentation

4.88.3.1 `struct blkfront_aiocb aiocb`

4.88.3.2 `struct blkfront_dev* blkdev = NULL`

4.88.3.3 `struct blkfront_info blkinfo`

4.88.3.4 `off_t blkoff`

4.88.3.5 `unsigned int blocksize`

4.88.3.6 `unsigned long long disksize = 0`

## 4.89 xen/stubdom/vtpm/vtpmbk.h File Reference

```
#include <mini-os/types.h>
#include <tpmfront.h>
```

### Functions

- `int init_vtpmbk (struct tpmfront_dev *tpmfront_dev)`
- `void shutdown_vtpmbk (void)`
- `int write_vtpmbk (struct tpmfront_dev *tpmfront_dev, uint8_t *data, size_t data_length)`
- `int read_vtpmbk (struct tpmfront_dev *tpmfront_dev, uint8_t **data, size_t *data_length)`

### 4.89.1 Function Documentation

4.89.1.1 `int init_vtpmbk ( struct tpmfront_dev * tpmfront_dev )`

4.89.1.2 `int read_vtpmbk ( struct tpmfront_dev * tpmfront_dev, uint8_t ** data, size_t * data_length )`

Reads, Decrypts, and returns data from blk device

4.89.1.3 `void shutdown_vtpmbk ( void )`

4.89.1.4 `int write_vtpmbk ( struct tpmfront_dev * tpmfront_dev, uint8_t * data, size_t data_length )`

Encrypts and writes data to blk device



## 4.90 xen/tools/console/daemon/io.c File Reference

```
#include "utils.h"
#include "io.h"
#include <xs.h>
#include <xen/io/console.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sys/select.h>
#include <fcntl.h>
#include <unistd.h>
#include <termios.h>
#include <stdarg.h>
#include <sys/mman.h>
#include <time.h>
#include <assert.h>
```

### Data Structures

- struct [buffer](#)
- struct [domain](#)

### Macros

- #define [\\_GNU\\_SOURCE](#)
- #define [MAX\(a, b\)](#) (((a) > (b)) ? (a) : (b))
- #define [MIN\(a, b\)](#) (((a) < (b)) ? (a) : (b))
- #define [MAX\\_STRLEN\(x\)](#) ((sizeof(x) \* CHAR\_BIT + CHAR\_BIT-1) / 10 \* 3 + 2)
- #define [RATE\\_LIMIT\\_ALLOWANCE](#) 30
- #define [RATE\\_LIMIT\\_PERIOD](#) 200
- #define [DOMID\\_DOMT](#) 1

### Functions

- void [enum\\_domains](#) (void)
- void [handle\\_io](#) (void)

### Variables

- int [log\\_reload](#)
- int [log\\_guest](#)
- int [log\\_hv](#)
- int [log\\_time\\_hv](#)
- int [log\\_time\\_guest](#)
- char \* [log\\_dir](#)
- int [discard\\_overflowed\\_data](#)

## 4.90.1 Macro Definition Documentation

4.90.1.1 `#define _GNU_SOURCE`

4.90.1.2 `#define DOMID_DOMT 1`

4.90.1.3 `#define MAX( a, b ) (((a) > (b)) ? (a) : (b))`

4.90.1.4 `#define MAX_STRLLEN( x ) ((sizeof(x) * CHAR_BIT + CHAR_BIT-1) / 10 * 3 + 2)`

4.90.1.5 `#define MIN( a, b ) (((a) < (b)) ? (a) : (b))`

4.90.1.6 `#define RATE_LIMIT_ALLOWANCE 30`

4.90.1.7 `#define RATE_LIMIT_PERIOD 200`

## 4.90.2 Function Documentation

4.90.2.1 `void enum_domains ( void )`

4.90.2.2 `void handle_io ( void )`

## 4.90.3 Variable Documentation

4.90.3.1 `int discard_overflowed_data`

4.90.3.2 `char* log_dir`

4.90.3.3 `int log_guest`

4.90.3.4 `int log_hv`

4.90.3.5 `int log_reload`

4.90.3.6 `int log_time_guest`

4.90.3.7 `int log_time_hv`

## 4.91 xen/tools/debugger/kdd/.kdd-xen.o.d File Reference

## 4.92 xen/tools/debugger/kdd/.kdd.o.d File Reference

## 4.93 xen/tools/ioemu-dir/alpha-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-alpha"`
- `#define TARGET_ARCH "alpha"`
- `#define TARGET_ALPHA 1`
- `#define CONFIG_USER_ONLY 1`

- #define [CONFIG\\_LINUX\\_USER](#) 1

#### 4.93.1 Macro Definition Documentation

4.93.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.93.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-alpha"

4.93.1.3 #define [CONFIG\\_USER\\_ONLY](#) 1

4.93.1.4 #define [TARGET\\_ALPHA](#) 1

4.93.1.5 #define [TARGET\\_ARCH](#) "alpha"

### 4.94 xen/tools/ioemu-dir/arm-linux-user/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-arm"
- #define [TARGET\\_ARCH](#) "arm"
- #define [TARGET\\_ARM](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1
- #define [TARGET\\_HAS\\_BFLT](#) 1

#### 4.94.1 Macro Definition Documentation

4.94.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.94.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-arm"

4.94.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.94.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.94.1.5 #define [TARGET\\_ARCH](#) "arm"

4.94.1.6 #define [TARGET\\_ARM](#) 1

4.94.1.7 #define [TARGET\\_HAS\\_BFLT](#) 1

### 4.95 xen/tools/ioemu-dir/arm-softhmmu/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-arm"
- #define [TARGET\\_ARCH](#) "arm"
- #define [TARGET\\_ARM](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.95.1 Macro Definition Documentation

4.95.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-arm"

4.95.1.2 #define [CONFIG\\_SOFTFLOAT](#) 1

4.95.1.3 #define [CONFIG\\_SOFTMMU](#) 1

4.95.1.4 #define [TARGET\\_ARCH](#) "arm"

4.95.1.5 #define [TARGET\\_ARM](#) 1

## 4.96 xen/tools/ioemu-dir/armeb-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-armeb"
- #define [TARGET\\_ARCH](#) "arm"
- #define [TARGET\\_ARM](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1
- #define [TARGET\\_HAS\\_BFLT](#) 1

### 4.96.1 Macro Definition Documentation

4.96.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.96.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-armeb"

4.96.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.96.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.96.1.5 #define [TARGET\\_ARCH](#) "arm"

4.96.1.6 #define [TARGET\\_ARM](#) 1

4.96.1.7 #define [TARGET\\_HAS\\_BFLT](#) 1

4.96.1.8 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.97 xen/tools/ioemu-dir/cris-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-cris"
- #define [TARGET\\_ARCH](#) "cris"
- #define [TARGET\\_CRIS](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1

### 4.97.1 Macro Definition Documentation

4.97.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.97.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-cris"

4.97.1.3 #define [CONFIG\\_USER\\_ONLY](#) 1

4.97.1.4 #define [TARGET\\_ARCH](#) "cris"

4.97.1.5 #define [TARGET\\_CRIS](#) 1

## 4.98 xen/tools/ioemu-dir/cris-softhmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-cris"
- #define [TARGET\\_ARCH](#) "cris"
- #define [TARGET\\_CRIS](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1

### 4.98.1 Macro Definition Documentation

4.98.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-cris"

4.98.1.2 #define [CONFIG\\_SOFTMMU](#) 1

4.98.1.3 #define [TARGET\\_ARCH](#) "cris"

4.98.1.4 #define [TARGET\\_CRIS](#) 1

## 4.99 xen/tools/ioemu-dir/i386-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-i386"`
- `#define TARGET_ARCH "i386"`
- `#define TARGET_I386 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`

### 4.99.1 Macro Definition Documentation

4.99.1.1 `#define CONFIG_LINUX_USER 1`

4.99.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-i386"`

4.99.1.3 `#define CONFIG_USER_ONLY 1`

4.99.1.4 `#define TARGET_ARCH "i386"`

4.99.1.5 `#define TARGET_I386 1`

### 4.100 xen/tools/ioemu-dir/i386-softmmu/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-i386"`
- `#define TARGET_ARCH "i386"`
- `#define TARGET_I386 1`
- `#define USE_KQEMU 1`
- `#define CONFIG_KVM 1`
- `#define CONFIG_SOFTMMU 1`

### 4.100.1 Macro Definition Documentation

4.100.1.1 `#define CONFIG_KVM 1`

4.100.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-i386"`

4.100.1.3 `#define CONFIG_SOFTMMU 1`

4.100.1.4 `#define TARGET_ARCH "i386"`

4.100.1.5 `#define TARGET_I386 1`

4.100.1.6 `#define USE_KQEMU 1`

### 4.101 xen/tools/ioemu-dir/m68k-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-m68k"
- #define [TARGET\\_ARCH](#) "m68k"
- #define [TARGET\\_M68K](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1
- #define [TARGET\\_HAS\\_BFLT](#) 1

### 4.101.1 Macro Definition Documentation

4.101.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.101.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-m68k"

4.101.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.101.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.101.1.5 #define [TARGET\\_ARCH](#) "m68k"

4.101.1.6 #define [TARGET\\_HAS\\_BFLT](#) 1

4.101.1.7 #define [TARGET\\_M68K](#) 1

4.101.1.8 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.102 xen/tools/ioemu-dir/m68k-softmmu/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-m68k"
- #define [TARGET\\_ARCH](#) "m68k"
- #define [TARGET\\_M68K](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.102.1 Macro Definition Documentation

4.102.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-m68k"

4.102.1.2 #define [CONFIG\\_SOFTFLOAT](#) 1

4.102.1.3 #define [CONFIG\\_SOFTMMU](#) 1

4.102.1.4 #define [TARGET\\_ARCH](#) "m68k"

4.102.1.5 `#define TARGET_M68K 1`

4.102.1.6 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.103 xen/tools/ioemu-dir/mips-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips"`
- `#define TARGET_ARCH "mips"`
- `#define TARGET_MIPS 1`
- `#define TARGET_ABI_MIPSO32 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`
- `#define CONFIG_SOFTFLOAT 1`

### 4.103.1 Macro Definition Documentation

4.103.1.1 `#define CONFIG_LINUX_USER 1`

4.103.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips"`

4.103.1.3 `#define CONFIG_SOFTFLOAT 1`

4.103.1.4 `#define CONFIG_USER_ONLY 1`

4.103.1.5 `#define TARGET_ABI_MIPSO32 1`

4.103.1.6 `#define TARGET_ARCH "mips"`

4.103.1.7 `#define TARGET_MIPS 1`

4.103.1.8 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.104 xen/tools/ioemu-dir/mips-softmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips"`
- `#define TARGET_ARCH "mips"`
- `#define TARGET_MIPS 1`
- `#define TARGET_ABI_MIPSO32 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_SOFTMMU 1`
- `#define CONFIG_SOFTFLOAT 1`



#### 4.104.1 Macro Definition Documentation

4.104.1.1 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips"`

4.104.1.2 `#define CONFIG_SOFTFLOAT 1`

4.104.1.3 `#define CONFIG_SOFTMMU 1`

4.104.1.4 `#define TARGET_ABI_MIPSO32 1`

4.104.1.5 `#define TARGET_ARCH "mips"`

4.104.1.6 `#define TARGET_MIPS 1`

4.104.1.7 `#define TARGET_WORDS_BIGENDIAN 1`

### 4.105 xen/tools/ioemu-dir/mips64-softmmu/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips64"`
- `#define TARGET_ARCH "mips64"`
- `#define TARGET_MIPS 1`
- `#define TARGET_MIPS64 1`
- `#define TARGET_ABI_MIPSN64 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_SOFTMMU 1`
- `#define CONFIG_SOFTFLOAT 1`

#### 4.105.1 Macro Definition Documentation

4.105.1.1 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mips64"`

4.105.1.2 `#define CONFIG_SOFTFLOAT 1`

4.105.1.3 `#define CONFIG_SOFTMMU 1`

4.105.1.4 `#define TARGET_ABI_MIPSN64 1`

4.105.1.5 `#define TARGET_ARCH "mips64"`

4.105.1.6 `#define TARGET_MIPS 1`

4.105.1.7 `#define TARGET_MIPS64 1`

4.105.1.8 `#define TARGET_WORDS_BIGENDIAN 1`

### 4.106 xen/tools/ioemu-dir/mips64el-softmmu/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-mips64el"
- #define [TARGET\\_ARCH](#) "mips64"
- #define [TARGET\\_MIPS](#) 1
- #define [TARGET\\_MIPS64](#) 1
- #define [TARGET\\_ABI\\_MIPSN64](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.106.1 Macro Definition Documentation

4.106.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-mips64el"

4.106.1.2 #define [CONFIG\\_SOFTFLOAT](#) 1

4.106.1.3 #define [CONFIG\\_SOFTMMU](#) 1

4.106.1.4 #define [TARGET\\_ABI\\_MIPSN64](#) 1

4.106.1.5 #define [TARGET\\_ARCH](#) "mips64"

4.106.1.6 #define [TARGET\\_MIPS](#) 1

4.106.1.7 #define [TARGET\\_MIPS64](#) 1

## 4.107 xen/tools/ioemu-dir/mipsel-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-mipsel"
- #define [TARGET\\_ARCH](#) "mips"
- #define [TARGET\\_MIPS](#) 1
- #define [TARGET\\_ABI\\_MIPSO32](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.107.1 Macro Definition Documentation

4.107.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.107.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-mipsel"

4.107.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.107.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.107.1.5 #define [TARGET\\_ABI\\_MIPSO32](#) 1

4.107.1.6 `#define TARGET_ARCH "mips"`

4.107.1.7 `#define TARGET_MIPS 1`

## 4.108 xen/tools/ioemu-dir/mipsel-softmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mipsel"`
- `#define TARGET_ARCH "mips"`
- `#define TARGET_MIPS 1`
- `#define TARGET_ABI_MIPSO32 1`
- `#define CONFIG_SOFTMMU 1`
- `#define CONFIG_SOFTFLOAT 1`

#### 4.108.1 Macro Definition Documentation

4.108.1.1 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-mipsel"`

4.108.1.2 `#define CONFIG_SOFTFLOAT 1`

4.108.1.3 `#define CONFIG_SOFTMMU 1`

4.108.1.4 `#define TARGET_ABI_MIPSO32 1`

4.108.1.5 `#define TARGET_ARCH "mips"`

4.108.1.6 `#define TARGET_MIPS 1`

## 4.109 xen/tools/ioemu-dir/ppc-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppc"`
- `#define TARGET_ARCH "ppc"`
- `#define TARGET_PPC 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`
- `#define CONFIG_SOFTFLOAT 1`

#### 4.109.1 Macro Definition Documentation

4.109.1.1 `#define CONFIG_LINUX_USER 1`

4.109.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppc"`

4.109.1.3 `#define CONFIG_SOFTFLOAT 1`

4.109.1.4 `#define CONFIG_USER_ONLY 1`

4.109.1.5 `#define TARGET_ARCH "ppc"`

4.109.1.6 `#define TARGET_PPC 1`

4.109.1.7 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.110 xen/tools/ioemu-dir/ppc-softmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppc"`
- `#define TARGET_ARCH "ppc"`
- `#define TARGET_PPC 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_SOFTMMU 1`
- `#define CONFIG_SOFTFLOAT 1`

### 4.110.1 Macro Definition Documentation

4.110.1.1 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppc"`

4.110.1.2 `#define CONFIG_SOFTFLOAT 1`

4.110.1.3 `#define CONFIG_SOFTMMU 1`

4.110.1.4 `#define TARGET_ARCH "ppc"`

4.110.1.5 `#define TARGET_PPC 1`

4.110.1.6 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.111 xen/tools/ioemu-dir/ppc64-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppc64"`
- `#define TARGET_ARCH "ppc64"`
- `#define TARGET_PPC 1`
- `#define TARGET_PPC64 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_USER_ONLY 1`

- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

#### 4.111.1 Macro Definition Documentation

4.111.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.111.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppc64"

4.111.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.111.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.111.1.5 #define [TARGET\\_ARCH](#) "ppc64"

4.111.1.6 #define [TARGET\\_PPC](#) 1

4.111.1.7 #define [TARGET\\_PPC64](#) 1

4.111.1.8 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

### 4.112 xen/tools/ioemu-dir/ppc64-softmmu/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppc64"
- #define [TARGET\\_ARCH](#) "ppc64"
- #define [TARGET\\_PPC](#) 1
- #define [TARGET\\_PPC64](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

#### 4.112.1 Macro Definition Documentation

4.112.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppc64"

4.112.1.2 #define [CONFIG\\_SOFTFLOAT](#) 1

4.112.1.3 #define [CONFIG\\_SOFTMMU](#) 1

4.112.1.4 #define [TARGET\\_ARCH](#) "ppc64"

4.112.1.5 #define [TARGET\\_PPC](#) 1

4.112.1.6 #define [TARGET\\_PPC64](#) 1

4.112.1.7 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.113 xen/tools/ioemu-dir/ppc64abi32-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppc64abi32"
- #define [TARGET\\_ARCH](#) "ppc64"
- #define [TARGET\\_PPC](#) 1
- #define [TARGET\\_PPC64](#) 1
- #define [TARGET\\_ABI32](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.113.1 Macro Definition Documentation

4.113.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.113.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppc64abi32"

4.113.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.113.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.113.1.5 #define [TARGET\\_ABI32](#) 1

4.113.1.6 #define [TARGET\\_ARCH](#) "ppc64"

4.113.1.7 #define [TARGET\\_PPC](#) 1

4.113.1.8 #define [TARGET\\_PPC64](#) 1

4.113.1.9 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.114 xen/tools/ioemu-dir/ppcemb-softmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-ppcemb"
- #define [TARGET\\_ARCH](#) "ppcemb"
- #define [TARGET\\_PPC](#) 1
- #define [TARGET\\_PPCEMB](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

#### 4.114.1 Macro Definition Documentation

4.114.1.1 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-ppcemb"`

4.114.1.2 `#define CONFIG_SOFTFLOAT 1`

4.114.1.3 `#define CONFIG_SOFTMMU 1`

4.114.1.4 `#define TARGET_ARCH "ppcemb"`

4.114.1.5 `#define TARGET_PPC 1`

4.114.1.6 `#define TARGET_PPCEMB 1`

4.114.1.7 `#define TARGET_WORDS_BIGENDIAN 1`

### 4.115 xen/tools/ioemu-dir/sh4-linux-user/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-sh4"`
- `#define TARGET_ARCH "sh4"`
- `#define TARGET_SH4 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`
- `#define TARGET_HAS_BFLT 1`

#### 4.115.1 Macro Definition Documentation

4.115.1.1 `#define CONFIG_LINUX_USER 1`

4.115.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-sh4"`

4.115.1.3 `#define CONFIG_USER_ONLY 1`

4.115.1.4 `#define TARGET_ARCH "sh4"`

4.115.1.5 `#define TARGET_HAS_BFLT 1`

4.115.1.6 `#define TARGET_SH4 1`

### 4.116 xen/tools/ioemu-dir/sh4-softmmu/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-sh4"`
- `#define TARGET_ARCH "sh4"`

- #define `TARGET_SH4` 1
- #define `CONFIG_SOFTMMU` 1

#### 4.116.1 Macro Definition Documentation

4.116.1.1 #define `CONFIG_QEMU_PREFIX` `"/usr/gnemul/qemu-sh4"`

4.116.1.2 #define `CONFIG_SOFTMMU` 1

4.116.1.3 #define `TARGET_ARCH` `"sh4"`

4.116.1.4 #define `TARGET_SH4` 1

### 4.117 xen/tools/ioemu-dir/sh4eb-linux-user/config.h File Reference

```
#include "../config-host.h"
```

#### Macros

- #define `CONFIG_QEMU_PREFIX` `"/usr/gnemul/qemu-sh4eb"`
- #define `TARGET_ARCH` `"sh4"`
- #define `TARGET_SH4` 1
- #define `TARGET_WORDS_BIGENDIAN` 1
- #define `CONFIG_USER_ONLY` 1
- #define `CONFIG_LINUX_USER` 1
- #define `TARGET_HAS_BFLT` 1

#### 4.117.1 Macro Definition Documentation

4.117.1.1 #define `CONFIG_LINUX_USER` 1

4.117.1.2 #define `CONFIG_QEMU_PREFIX` `"/usr/gnemul/qemu-sh4eb"`

4.117.1.3 #define `CONFIG_USER_ONLY` 1

4.117.1.4 #define `TARGET_ARCH` `"sh4"`

4.117.1.5 #define `TARGET_HAS_BFLT` 1

4.117.1.6 #define `TARGET_SH4` 1

4.117.1.7 #define `TARGET_WORDS_BIGENDIAN` 1

### 4.118 xen/tools/ioemu-dir/sh4eb-softmmu/config.h File Reference

```
#include "../config-host.h"
```



## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sh4eb"
- #define [TARGET\\_ARCH](#) "sh4"
- #define [TARGET\\_SH4](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1

### 4.118.1 Macro Definition Documentation

4.118.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sh4eb"

4.118.1.2 #define [CONFIG\\_SOFTMMU](#) 1

4.118.1.3 #define [TARGET\\_ARCH](#) "sh4"

4.118.1.4 #define [TARGET\\_SH4](#) 1

4.118.1.5 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.119 xen/tools/ioemu-dir/sparc-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc"
- #define [TARGET\\_ARCH](#) "sparc"
- #define [TARGET\\_SPARC](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.119.1 Macro Definition Documentation

4.119.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.119.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc"

4.119.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.119.1.4 #define [CONFIG\\_USER\\_ONLY](#) 1

4.119.1.5 #define [TARGET\\_ARCH](#) "sparc"

4.119.1.6 #define [TARGET\\_SPARC](#) 1

4.119.1.7 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.120 xen/tools/ioemu-dir/sparc-softmmu/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc"
- #define [TARGET\\_ARCH](#) "sparc"
- #define [TARGET\\_SPARC](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_SOFTMMU](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.120.1 Macro Definition Documentation

4.120.1.1 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc"

4.120.1.2 #define [CONFIG\\_SOFTFLOAT](#) 1

4.120.1.3 #define [CONFIG\\_SOFTMMU](#) 1

4.120.1.4 #define [TARGET\\_ARCH](#) "sparc"

4.120.1.5 #define [TARGET\\_SPARC](#) 1

4.120.1.6 #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1

## 4.121 xen/tools/ioemu-dir/sparc32plus-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc32plus"
- #define [TARGET\\_ARCH](#) "sparc64"
- #define [TARGET\\_SPARC](#) 1
- #define [TARGET\\_SPARC64](#) 1
- #define [TARGET\\_ABI32](#) 1
- #define [TARGET\\_WORDS\\_BIGENDIAN](#) 1
- #define [CONFIG\\_USER\\_ONLY](#) 1
- #define [CONFIG\\_LINUX\\_USER](#) 1
- #define [CONFIG\\_SOFTFLOAT](#) 1

### 4.121.1 Macro Definition Documentation

4.121.1.1 #define [CONFIG\\_LINUX\\_USER](#) 1

4.121.1.2 #define [CONFIG\\_QEMU\\_PREFIX](#) "/usr/gnemul/qemu-sparc32plus"

4.121.1.3 #define [CONFIG\\_SOFTFLOAT](#) 1

4.121.1.4 `#define CONFIG_USER_ONLY 1`

4.121.1.5 `#define TARGET_ABI32 1`

4.121.1.6 `#define TARGET_ARCH "sparc64"`

4.121.1.7 `#define TARGET_SPARC 1`

4.121.1.8 `#define TARGET_SPARC64 1`

4.121.1.9 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.122 xen/tools/ioemu-dir/sparc64-linux-user/config.h File Reference

```
#include "../config-host.h"
```

### Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-sparc64"`
- `#define TARGET_ARCH "sparc64"`
- `#define TARGET_SPARC 1`
- `#define TARGET_SPARC64 1`
- `#define TARGET_WORDS_BIGENDIAN 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`
- `#define CONFIG_SOFTFLOAT 1`
- `#define TARGET_HAS_ELFLOAD32 1`

### 4.122.1 Macro Definition Documentation

4.122.1.1 `#define CONFIG_LINUX_USER 1`

4.122.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-sparc64"`

4.122.1.3 `#define CONFIG_SOFTFLOAT 1`

4.122.1.4 `#define CONFIG_USER_ONLY 1`

4.122.1.5 `#define TARGET_ARCH "sparc64"`

4.122.1.6 `#define TARGET_HAS_ELFLOAD32 1`

4.122.1.7 `#define TARGET_SPARC 1`

4.122.1.8 `#define TARGET_SPARC64 1`

4.122.1.9 `#define TARGET_WORDS_BIGENDIAN 1`

## 4.123 xen/tools/ioemu-dir/x86\_64-linux-user/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-x86_64"`
- `#define TARGET_ARCH "x86_64"`
- `#define TARGET_I386 1`
- `#define TARGET_X86_64 1`
- `#define CONFIG_USER_ONLY 1`
- `#define CONFIG_LINUX_USER 1`

### 4.123.1 Macro Definition Documentation

4.123.1.1 `#define CONFIG_LINUX_USER 1`

4.123.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-x86_64"`

4.123.1.3 `#define CONFIG_USER_ONLY 1`

4.123.1.4 `#define TARGET_ARCH "x86_64"`

4.123.1.5 `#define TARGET_I386 1`

4.123.1.6 `#define TARGET_X86_64 1`

## 4.124 xen/tools/ioemu-dir/x86\_64-softhmmu/config.h File Reference

```
#include "../config-host.h"
```

## Macros

- `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-x86_64"`
- `#define TARGET_ARCH "x86_64"`
- `#define TARGET_I386 1`
- `#define TARGET_X86_64 1`
- `#define USE_KQEMU 1`
- `#define CONFIG_KVM 1`
- `#define CONFIG_SOFTMMU 1`

### 4.124.1 Macro Definition Documentation

4.124.1.1 `#define CONFIG_KVM 1`

4.124.1.2 `#define CONFIG_QEMU_PREFIX "/usr/gnemul/qemu-x86_64"`

4.124.1.3 `#define CONFIG_SOFTMMU 1`

4.124.1.4 `#define TARGET_ARCH "x86_64"`

4.124.1.5 `#define TARGET_I386 1`

4.124.1.6 `#define TARGET_X86_64 1`

4.124.1.7 `#define USE_KQEMU 1`

## 4.125 xen/tools/ioemu-dir/config-host.h File Reference

```
#include "xen-config-host.h"
```

### Macros

- #define [CONFIG\\_QEMU\\_SHAREDIR](#) "/usr/share/xen/qemu"
- #define [HOST\\_X86\\_64](#) 1
- #define [HOST\\_LONG\\_BITS](#) 64
- #define [HAVE\\_BYTESWAP\\_H](#) 1
- #define [CONFIG\\_AC97](#) 1
- #define [CONFIG\\_ES1370](#) 1
- #define [CONFIG\\_SB16](#) 1
- #define [AUDIO\\_DRIVERS](#) &oss\_audio\_driver, \
- #define [CONFIG\\_VNC\\_TLS](#) 1
- #define [QEMU\\_VERSION](#) "0.10.2"
- #define [CONFIG\\_SDL](#) 1
- #define [CONFIG\\_OPENGL](#) 1
- #define [CONFIG\\_AIO](#) 1
- #define [HAVE\\_IOVEC](#) 1
- #define [CONFIG\\_UNAME\\_RELEASE](#) ""
- #define [DEFAULT\\_NETWORK\\_SCRIPT](#) "/etc/xen/scripts/qemu-ifup"
- #define [DEFAULT\\_NETWORK\\_DOWN\\_SCRIPT](#) "/etc/xen/scripts/qemu-ifdown"

### 4.125.1 Macro Definition Documentation

4.125.1.1 #define [AUDIO\\_DRIVERS](#) &oss\_audio\_driver, \

4.125.1.2 #define [CONFIG\\_AC97](#) 1

4.125.1.3 #define [CONFIG\\_AIO](#) 1

4.125.1.4 #define [CONFIG\\_ES1370](#) 1

4.125.1.5 #define [CONFIG\\_OPENGL](#) 1

4.125.1.6 #define [CONFIG\\_QEMU\\_SHAREDIR](#) "/usr/share/xen/qemu"

4.125.1.7 #define [CONFIG\\_SB16](#) 1

4.125.1.8 #define [CONFIG\\_SDL](#) 1

4.125.1.9 #define [CONFIG\\_UNAME\\_RELEASE](#) ""

4.125.1.10 #define [CONFIG\\_VNC\\_TLS](#) 1

4.125.1.11 #define [DEFAULT\\_NETWORK\\_DOWN\\_SCRIPT](#) "/etc/xen/scripts/qemu-ifdown"

4.125.1.12 #define [DEFAULT\\_NETWORK\\_SCRIPT](#) "/etc/xen/scripts/qemu-ifup"

4.125.1.13 #define [HAVE\\_BYTESWAP\\_H](#) 1

4.125.1.14 #define [HAVE\\_IOVEC](#) 1

4.125.1.15 `#define HOST_LONG_BITS 64`

4.125.1.16 `#define HOST_X86_64 1`

4.125.1.17 `#define QEMU_VERSION "0.10.2"`

## 4.126 xen/tools/ioemu-dir/tests/test-mmap.c File Reference

## 4.127 xen/tools/libxc/xc\_caas.c File Reference

```
#include "xc_private.h"
#include <unistd.h>
#include <stdio.h>
#include <errno.h>
#include <fcntl.h>
#include <string.h>
#include <sys/mman.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <stdint.h>
#include <sys/ioctl.h>
```

### Functions

- int [xc\\_caas\\_op](#) ([xc\\_interface](#) \*xch, [caas\\_op\\_t](#) \*op)
- int [xc\\_caas\\_exempt](#) ([xc\\_interface](#) \*xc\_handle, uint64\_t mfn)

### 4.127.1 Function Documentation

4.127.1.1 int [xc\\_caas\\_exempt](#) ( [xc\\_interface](#) \* [xc\\_handle](#), uint64\_t *mfn* )

4.127.1.2 int [xc\\_caas\\_op](#) ( [xc\\_interface](#) \* *xch*, [caas\\_op\\_t](#) \* *op* )

## 4.128 xen/tools/libxc/xc\_dom.h File Reference

```
#include <xen/libelf/libelf.h>
```

### Data Structures

- struct [xc\\_dom\\_seg](#)
- struct [xc\\_dom\\_mem](#)
- struct [xc\\_dom\\_phys](#)
- struct [xc\\_dom\\_image](#)
- struct [xc\\_dom\\_loader](#)
- struct [xc\\_dom\\_arch](#)

## Macros

- `#define INVALID_P2M_ENTRY ((xen_pfn_t)-1)`
- `#define PRIpfn PRI_xen_pfn`
- `#define __init__attribute__ ((constructor))`
- `#define XC_DOM_PAGE_SHIFT(dom) ((dom)->arch_hooks->page_shift)`
- `#define XC_DOM_PAGE_SIZE(dom) (1 << (dom)->arch_hooks->page_shift)`
- `#define xc_dom_panic(xch, err, fmt, args...) xc_dom_panic_func(xch, __FILE__, __LINE__, err, fmt, ## args)`
- `#define xc_dom_trace(mark) xc_dom_printf("%s:%d: trace %s\n", __FILE__, __LINE__, mark)`

## Typedefs

- `typedef uint64_t xen_vaddr_t`
- `typedef uint64_t xen_paddr_t`

## Functions

- `void xc_dom_register_loader (struct xc_dom_loader *loader)`
- `void xc_dom_register_arch_hooks (struct xc_dom_arch *hooks)`
- `struct xc_dom_image * xc_dom_allocate (xc_interface *xch, const char *cmdline, const char *features)`
- `void xc_dom_release_phys (struct xc_dom_image *dom)`
- `void xc_dom_release (struct xc_dom_image *dom)`
- `int xc_dom_mem_init (struct xc_dom_image *dom, unsigned int mem_mb)`
- `size_t xc_dom_check_gzip (xc_interface *xch, void *blob, size_t ziplen)`
- `int xc_dom_do_gunzip (xc_interface *xch, void *src, size_t srclen, void *dst, size_t dstlen)`
- `int xc_dom_try_gunzip (struct xc_dom_image *dom, void **blob, size_t *size)`
- `int xc_dom_kernel_file (struct xc_dom_image *dom, const char *filename)`
- `int xc_dom_ramdisk_file (struct xc_dom_image *dom, const char *filename)`
- `int xc_dom_kernel_mem (struct xc_dom_image *dom, const void *mem, size_t memsize)`
- `int xc_dom_ramdisk_mem (struct xc_dom_image *dom, const void *mem, size_t memsize)`
- `int xc_dom_parse_image (struct xc_dom_image *dom)`
- `struct xc_dom_arch * xc_dom_find_arch_hooks (xc_interface *xch, char *guest_type)`
- `int xc_dom_build_image (struct xc_dom_image *dom)`
- `int xc_dom_update_guest_p2m (struct xc_dom_image *dom)`
- `int xc_dom_boot_xen_init (struct xc_dom_image *dom, xc_interface *xch, domid_t domid)`
- `int xc_dom_boot_mem_init (struct xc_dom_image *dom)`
- `void * xc_dom_boot_domU_map (struct xc_dom_image *dom, xen_pfn_t pfn, xen_pfn_t count)`
- `int xc_dom_boot_image (struct xc_dom_image *dom)`
- `int xc_dom_compat_check (struct xc_dom_image *dom)`
- `int xc_dom_loginit (xc_interface *xch)`
- `void xc_dom_printf (xc_interface *xch, const char *fmt,...) __attribute__((format(printf`
- `void void xc_dom_panic_func (xc_interface *xch, const char *file, int line, xc_error_code err, const char *fmt,...) __attribute__((format(printf`
- `void xc_dom_log_memory_footprint (struct xc_dom_image *dom)`
- `void * xc_dom_malloc (struct xc_dom_image *dom, size_t size)`
- `void * xc_dom_malloc_page_aligned (struct xc_dom_image *dom, size_t size)`
- `void * xc_dom_malloc_filemap (struct xc_dom_image *dom, const char *filename, size_t *size)`
- `char * xc_dom_strdup (struct xc_dom_image *dom, const char *str)`
- `int xc_dom_alloc_page (struct xc_dom_image *dom, char *name)`
- `int xc_dom_alloc_segment (struct xc_dom_image *dom, struct xc_dom_seg *seg, char *name, xen_vaddr_t start, xen_vaddr_t size)`
- `void * xc_dom_pfn_to_ptr (struct xc_dom_image *dom, xen_pfn_t first, xen_pfn_t count)`
- `void xc_dom_unmap_one (struct xc_dom_image *dom, xen_pfn_t pfn)`
- `void xc_dom_unmap_all (struct xc_dom_image *dom)`
- `int arch_setup_meminit (struct xc_dom_image *dom)`
- `int arch_setup_booteary (struct xc_dom_image *dom)`
- `int arch_setup_bootlate (struct xc_dom_image *dom)`

### 4.128.1 Macro Definition Documentation

4.128.1.1 `#define __init__attribute__ ((constructor))`

4.128.1.2 `#define INVALID_P2M_ENTRY ((xen_pfn_t)-1)`

4.128.1.3 `#define PRIpfn PRI_xen_pfn`

4.128.1.4 `#define XC_DOM_PAGE_SHIFT( dom ) ((dom)->arch_hooks->page_shift)`

4.128.1.5 `#define XC_DOM_PAGE_SIZE( dom ) (1 << (dom)->arch_hooks->page_shift)`

4.128.1.6 `#define xc_dom_panic( xch, err, fmt, args... ) xc_dom_panic_func(xch, __FILE__, __LINE__, err, fmt, ## args)`

4.128.1.7 `#define xc_dom_trace( mark ) xc_dom_printf("%s:%d: trace %s\n", __FILE__, __LINE__, mark)`

### 4.128.2 Typedef Documentation

4.128.2.1 `typedef uint64_t xen_paddr_t`

4.128.2.2 `typedef uint64_t xen_vaddr_t`

### 4.128.3 Function Documentation

4.128.3.1 `int arch_setup_bootearly ( struct xc_dom_image * dom )`

4.128.3.2 `int arch_setup_bootlate ( struct xc_dom_image * dom )`

4.128.3.3 `int arch_setup_meminit ( struct xc_dom_image * dom )`

4.128.3.4 `int xc_dom_alloc_page ( struct xc_dom_image * dom, char * name )`

4.128.3.5 `int xc_dom_alloc_segment ( struct xc_dom_image * dom, struct xc_dom_seg * seg, char * name, xen_vaddr_t start, xen_vaddr_t size )`

4.128.3.6 `struct xc_dom_image* xc_dom_allocate ( xc_interface * xch, const char * cmdline, const char * features )`

4.128.3.7 `void* xc_dom_boot_domU_map ( struct xc_dom_image * dom, xen_pfn_t pfn, xen_pfn_t count )`

4.128.3.8 `int xc_dom_boot_image ( struct xc_dom_image * dom )`

4.128.3.9 `int xc_dom_boot_mem_init ( struct xc_dom_image * dom )`

4.128.3.10 `int xc_dom_boot_xen_init ( struct xc_dom_image * dom, xc_interface * xch, domid_t domid )`

4.128.3.11 `int xc_dom_build_image ( struct xc_dom_image * dom )`

4.128.3.12 `size_t xc_dom_check_gzip ( xc_interface * xch, void * blob, size_t ziplen )`

4.128.3.13 `int xc_dom_compat_check ( struct xc_dom_image * dom )`

4.128.3.14 `int xc_dom_do_gunzip ( xc_interface * xch, void * src, size_t srclen, void * dst, size_t dstlen )`

4.128.3.15 `struct xc_dom_arch* xc_dom_find_arch_hooks ( xc_interface * xch, char * guest_type )`



- 4.128.3.16 `int xc_dom_kernel_file ( struct xc_dom_image * dom, const char * filename )`
- 4.128.3.17 `int xc_dom_kernel_mem ( struct xc_dom_image * dom, const void * mem, size_t memsize )`
- 4.128.3.18 `void xc_dom_log_memory_footprint ( struct xc_dom_image * dom )`
- 4.128.3.19 `int xc_dom_loginit ( xc_interface * xch )`
- 4.128.3.20 `void* xc_dom_malloc ( struct xc_dom_image * dom, size_t size )`
- 4.128.3.21 `void* xc_dom_malloc_filemap ( struct xc_dom_image * dom, const char * filename, size_t * size )`
- 4.128.3.22 `void* xc_dom_malloc_page_aligned ( struct xc_dom_image * dom, size_t size )`
- 4.128.3.23 `int xc_dom_mem_init ( struct xc_dom_image * dom, unsigned int mem_mb )`
- 4.128.3.24 `void void xc_dom_panic_func ( xc_interface * xch, const char * file, int line, xc_error_code err, const char * fmt, ... )`
- 4.128.3.25 `int xc_dom_parse_image ( struct xc_dom_image * dom )`
- 4.128.3.26 `void* xc_dom_pfn_to_ptr ( struct xc_dom_image * dom, xen_pfn_t first, xen_pfn_t count )`
- 4.128.3.27 `void xc_dom_printf ( xc_interface * xch, const char * fmt, ... )`
- 4.128.3.28 `int xc_dom_ramdisk_file ( struct xc_dom_image * dom, const char * filename )`
- 4.128.3.29 `int xc_dom_ramdisk_mem ( struct xc_dom_image * dom, const void * mem, size_t memsize )`
- 4.128.3.30 `void xc_dom_register_arch_hooks ( struct xc_dom_arch * hooks )`
- 4.128.3.31 `void xc_dom_register_loader ( struct xc_dom_loader * loader )`
- 4.128.3.32 `void xc_dom_release ( struct xc_dom_image * dom )`
- 4.128.3.33 `void xc_dom_release_phys ( struct xc_dom_image * dom )`
- 4.128.3.34 `char* xc_dom_strdup ( struct xc_dom_image * dom, const char * str )`
- 4.128.3.35 `int xc_dom_try_gunzip ( struct xc_dom_image * dom, void ** blob, size_t * size )`
- 4.128.3.36 `void xc_dom_unmap_all ( struct xc_dom_image * dom )`
- 4.128.3.37 `void xc_dom_unmap_one ( struct xc_dom_image * dom, xen_pfn_t pfn )`
- 4.128.3.38 `int xc_dom_update_guest_p2m ( struct xc_dom_image * dom )`

## 4.129 xen/tools/libxc/xc\_dom\_core.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <inttypes.h>
#include <zlib.h>
#include <assert.h>
#include "xc_private.h"
#include "xc_dom.h"
```

### Functions

- int [xc\\_dom\\_loginit](#) ([xc\\_interface](#) \*xch)
- void [xc\\_dom\\_printf](#) ([xc\\_interface](#) \*xch, const char \*fmt,...)
- void [xc\\_dom\\_panic\\_func](#) ([xc\\_interface](#) \*xch, const char \*file, int line, [xc\\_error\\_code](#) err, const char \*fmt,...)
- void [xc\\_dom\\_log\\_memory\\_footprint](#) (struct [xc\\_dom\\_image](#) \*dom)
- void \* [xc\\_dom\\_malloc](#) (struct [xc\\_dom\\_image](#) \*dom, size\_t size)
- void \* [xc\\_dom\\_malloc\\_page\\_aligned](#) (struct [xc\\_dom\\_image](#) \*dom, size\_t size)
- void \* [xc\\_dom\\_malloc\\_filemap](#) (struct [xc\\_dom\\_image](#) \*dom, const char \*filename, size\_t \*size)
- char \* [xc\\_dom\\_strdup](#) (struct [xc\\_dom\\_image](#) \*dom, const char \*str)
- size\_t [xc\\_dom\\_check\\_gzip](#) ([xc\\_interface](#) \*xch, void \*blob, size\_t ziplen)
- int [xc\\_dom\\_do\\_gunzip](#) ([xc\\_interface](#) \*xch, void \*src, size\_t srclen, void \*dst, size\_t dstlen)
- int [xc\\_dom\\_try\\_gunzip](#) (struct [xc\\_dom\\_image](#) \*dom, void \*\*blob, size\_t \*size)
- void \* [xc\\_dom\\_pfn\\_to\\_ptr](#) (struct [xc\\_dom\\_image](#) \*dom, xen\_pfn\_t pfn, xen\_pfn\_t count)
- int [xc\\_dom\\_alloc\\_segment](#) (struct [xc\\_dom\\_image](#) \*dom, struct [xc\\_dom\\_seg](#) \*seg, char \*name, [xen\\_vaddr\\_t](#) start, [xen\\_vaddr\\_t](#) size)
- int [xc\\_dom\\_alloc\\_page](#) (struct [xc\\_dom\\_image](#) \*dom, char \*name)
- void [xc\\_dom\\_unmap\\_one](#) (struct [xc\\_dom\\_image](#) \*dom, xen\_pfn\_t pfn)
- void [xc\\_dom\\_unmap\\_all](#) (struct [xc\\_dom\\_image](#) \*dom)
- void [xc\\_dom\\_register\\_loader](#) (struct [xc\\_dom\\_loader](#) \*loader)
- void [xc\\_dom\\_register\\_arch\\_hooks](#) (struct [xc\\_dom\\_arch](#) \*hooks)
- struct [xc\\_dom\\_arch](#) \* [xc\\_dom\\_find\\_arch\\_hooks](#) ([xc\\_interface](#) \*xch, char \*guest\_type)
- void [xc\\_dom\\_release](#) (struct [xc\\_dom\\_image](#) \*dom)
- struct [xc\\_dom\\_image](#) \* [xc\\_dom\\_allocate](#) ([xc\\_interface](#) \*xch, const char \*cmdline, const char \*features)
- int [xc\\_dom\\_kernel\\_file](#) (struct [xc\\_dom\\_image](#) \*dom, const char \*filename)
- int [xc\\_dom\\_ramdisk\\_file](#) (struct [xc\\_dom\\_image](#) \*dom, const char \*filename)
- int [xc\\_dom\\_kernel\\_mem](#) (struct [xc\\_dom\\_image](#) \*dom, const void \*mem, size\_t memsize)
- int [xc\\_dom\\_ramdisk\\_mem](#) (struct [xc\\_dom\\_image](#) \*dom, const void \*mem, size\_t memsize)
- int [xc\\_dom\\_parse\\_image](#) (struct [xc\\_dom\\_image](#) \*dom)
- int [xc\\_dom\\_mem\\_init](#) (struct [xc\\_dom\\_image](#) \*dom, unsigned int mem\_mb)
- int [xc\\_dom\\_update\\_guest\\_p2m](#) (struct [xc\\_dom\\_image](#) \*dom)
- int [xc\\_dom\\_build\\_image](#) (struct [xc\\_dom\\_image](#) \*dom)

### 4.129.1 Function Documentation

4.129.1.1 int [xc\\_dom\\_alloc\\_page](#) ( struct [xc\\_dom\\_image](#) \* *dom*, char \* *name* )

4.129.1.2 int [xc\\_dom\\_alloc\\_segment](#) ( struct [xc\\_dom\\_image](#) \* *dom*, struct [xc\\_dom\\_seg](#) \* *seg*, char \* *name*, [xen\\_vaddr\\_t](#) *start*, [xen\\_vaddr\\_t](#) *size* )

- 4.129.1.3 struct xc\_dom\_image\* xc\_dom\_allocate ( xc\_interface \* xch, const char \* cmdline, const char \* features )
- 4.129.1.4 int xc\_dom\_build\_image ( struct xc\_dom\_image \* dom )
- 4.129.1.5 size\_t xc\_dom\_check\_gzip ( xc\_interface \* xch, void \* blob, size\_t ziplen )
- 4.129.1.6 int xc\_dom\_do\_gunzip ( xc\_interface \* xch, void \* src, size\_t srclen, void \* dst, size\_t dstlen )
- 4.129.1.7 struct xc\_dom\_arch\* xc\_dom\_find\_arch\_hooks ( xc\_interface \* xch, char \* guest\_type )
- 4.129.1.8 int xc\_dom\_kernel\_file ( struct xc\_dom\_image \* dom, const char \* filename )
- 4.129.1.9 int xc\_dom\_kernel\_mem ( struct xc\_dom\_image \* dom, const void \* mem, size\_t memsize )
- 4.129.1.10 void xc\_dom\_log\_memory\_footprint ( struct xc\_dom\_image \* dom )
- 4.129.1.11 int xc\_dom\_loginit ( xc\_interface \* xch )
- 4.129.1.12 void\* xc\_dom\_malloc ( struct xc\_dom\_image \* dom, size\_t size )
- 4.129.1.13 void\* xc\_dom\_malloc\_filemap ( struct xc\_dom\_image \* dom, const char \* filename, size\_t \* size )
- 4.129.1.14 void\* xc\_dom\_malloc\_page\_aligned ( struct xc\_dom\_image \* dom, size\_t size )
- 4.129.1.15 int xc\_dom\_mem\_init ( struct xc\_dom\_image \* dom, unsigned int mem\_mb )
- 4.129.1.16 void xc\_dom\_panic\_func ( xc\_interface \* xch, const char \* file, int line, xc\_error\_code err, const char \* fmt, ... )
- 4.129.1.17 int xc\_dom\_parse\_image ( struct xc\_dom\_image \* dom )
- 4.129.1.18 void\* xc\_dom\_pfn\_to\_ptr ( struct xc\_dom\_image \* dom, xen\_pfn\_t pfn, xen\_pfn\_t count )
- 4.129.1.19 void xc\_dom\_printf ( xc\_interface \* xch, const char \* fmt, ... )
- 4.129.1.20 int xc\_dom\_ramdisk\_file ( struct xc\_dom\_image \* dom, const char \* filename )
- 4.129.1.21 int xc\_dom\_ramdisk\_mem ( struct xc\_dom\_image \* dom, const void \* mem, size\_t memsize )
- 4.129.1.22 void xc\_dom\_register\_arch\_hooks ( struct xc\_dom\_arch \* hooks )
- 4.129.1.23 void xc\_dom\_register\_loader ( struct xc\_dom\_loader \* loader )
- 4.129.1.24 void xc\_dom\_release ( struct xc\_dom\_image \* dom )
- 4.129.1.25 char\* xc\_dom\_strdup ( struct xc\_dom\_image \* dom, const char \* str )
- 4.129.1.26 int xc\_dom\_try\_gunzip ( struct xc\_dom\_image \* dom, void \*\* blob, size\_t \* size )
- 4.129.1.27 void xc\_dom\_unmap\_all ( struct xc\_dom\_image \* dom )
- 4.129.1.28 void xc\_dom\_unmap\_one ( struct xc\_dom\_image \* dom, xen\_pfn\_t pfn )
- 4.129.1.29 int xc\_dom\_update\_guest\_p2m ( struct xc\_dom\_image \* dom )

## 4.130 xen/tools/libxc/xc\_dom\_x86.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <inttypes.h>
#include <xen/xen.h>
#include <xen/foreign/x86_32.h>
#include <xen/foreign/x86_64.h>
#include <xen/hvm/hvm_info_table.h>
#include <xen/io/protocols.h>
#include "xg_private.h"
#include "xc_dom.h"
#include "xenctrl.h"
```

### Macros

- `#define SUPERPAGE_PFN_SHIFT 9`
- `#define SUPERPAGE_NR_PFNS (1UL << SUPERPAGE_PFN_SHIFT)`
- `#define bits_to_mask(bits) (((xen_vaddr_t)1 << (bits))-1)`
- `#define round_down(addr, mask) ((addr) & ~(mask))`
- `#define round_up(addr, mask) ((addr) | (mask))`
- `#define L1_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED)`
- `#define L2_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`
- `#define L3_PROT (_PAGE_PRESENT)`
- `#define pfn_to_paddr(pfn) ((xen_paddr_t)(pfn) << PAGE_SHIFT_X86)`
- `#define L1_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED)`
- `#define L2_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`
- `#define L3_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`
- `#define L4_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`

### Functions

- `int arch_setup_meminit (struct xc_dom_image *dom)`
- `int arch_setup_bootearly (struct xc_dom_image *dom)`
- `int arch_setup_bootlate (struct xc_dom_image *dom)`

#### 4.130.1 Macro Definition Documentation

4.130.1.1 `#define bits_to_mask( bits ) (((xen_vaddr_t)1 << (bits))-1)`

4.130.1.2 `#define L1_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED)`

4.130.1.3 `#define L1_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED)`

4.130.1.4 `#define L2_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`

4.130.1.5 `#define L2_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`

4.130.1.6 `#define L3_PROT (_PAGE_PRESENT)`

4.130.1.7 `#define L3_PROT (_PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`

4.130.1.8 `#define L4_PROT ( _PAGE_PRESENT|_PAGE_RW|_PAGE_ACCESSED|_PAGE_DIRTY|_PAGE_USER)`

4.130.1.9 `#define pfn_to_paddr( pfn ) ((xen_paddr_t)(pfn) << PAGE_SHIFT_X86)`

4.130.1.10 `#define round_down( addr, mask ) ((addr) & ~(mask))`

4.130.1.11 `#define round_up( addr, mask ) ((addr) | (mask))`

4.130.1.12 `#define SUPERPAGE_NR_PFNS (1UL << SUPERPAGE_PFN_SHIFT)`

4.130.1.13 `#define SUPERPAGE_PFN_SHIFT 9`

## 4.130.2 Function Documentation

4.130.2.1 `int arch_setup_bootearly ( struct xc_dom_image * dom )`

4.130.2.2 `int arch_setup_bootlate ( struct xc_dom_image * dom )`

4.130.2.3 `int arch_setup_meminit ( struct xc_dom_image * dom )`

## 4.131 xen/tools/libxc/xenctrl.h File Reference

```
#include <stddef.h>
#include <stdint.h>
#include <stdio.h>
#include <xen/xen.h>
#include <xen/domctl.h>
#include <xen/physdev.h>
#include <xen/sysctl.h>
#include <xen/version.h>
#include <xen/event_channel.h>
#include <xen/sched.h>
#include <xen/memory.h>
#include <xen/grant_table.h>
#include <xen/hvm/params.h>
#include <xen/xsm/acm.h>
#include <xen/xsm/acm_ops.h>
#include <xen/xsm/flask_op.h>
#include <xen/xsm/caas_op.h>
#include <xen/tmem.h>
#include "xentoollog.h"
```

## Data Structures

- struct [xc\\_hypcall\\_buffer](#)
- struct [xc\\_core\\_header](#)
- struct [xc\\_dominfo](#)
- union [vcpu\\_guest\\_context\\_any\\_t](#)
- union [shared\\_info\\_any\\_t](#)
- union [start\\_info\\_any\\_t](#)
- struct [xc\\_cpupoolinfo](#)
- struct [xc\\_error](#)
- struct [xc\\_px\\_val](#)
- struct [xc\\_px\\_stat](#)

- struct [xc\\_cx\\_stat](#)
- struct [xc\\_get\\_cpufreq\\_para](#)
- struct [tmem\\_oid](#)

## Macros

- #define [\\_\\_XEN\\_TOOLS\\_\\_](#) 1
- #define [XC\\_PAGE\\_SHIFT](#) 12
- #define [XC\\_PAGE\\_SIZE](#) (1UL << XC\_PAGE\_SHIFT)
- #define [XC\\_PAGE\\_MASK](#) (~(XC\_PAGE\_SIZE-1))
- #define [INVALID\\_MFN](#) (~0UL)
- #define [XENCTRL\\_HAS\\_XC\\_INTERFACE](#) 1
- #define [XC\\_\\_HYPERCALL\\_BUFFER\\_NAME](#)(\_name) xc\_\_hypercall\_buffer\_##\_name
- #define [HYPERCALL\\_BUFFER](#)(\_name)
- #define [HYPERCALL\\_BUFFER\\_INIT\\_NO\\_BOUNCE](#) .dir = 0, .sz = 0, .ubuf = (void \*)-1
- #define [DECLARE\\_HYPERCALL\\_BUFFER](#)(\_type, \_name)
- #define [DECLARE\\_HYPERCALL\\_BUFFER\\_ARGUMENT](#)(\_name)
- #define [HYPERCALL\\_BUFFER\\_AS\\_ARG](#)(\_name)
- #define [set\\_xen\\_guest\\_handle](#)(\_hnd, \_val)
- #define [xc\\_hypercall\\_buffer\\_alloc](#)(\_xch, \_name, \_size) [xc\\_\\_hypercall\\_buffer\\_alloc](#)(\_xch, [HYPERCALL\\_BUFFER](#)(\_name), \_size)
- #define [xc\\_hypercall\\_buffer\\_free](#)(\_xch, \_name) [xc\\_\\_hypercall\\_buffer\\_free](#)(\_xch, [HYPERCALL\\_BUFFER](#)(\_name))
- #define [xc\\_hypercall\\_buffer\\_alloc\\_pages](#)(\_xch, \_name, \_nr) [xc\\_\\_hypercall\\_buffer\\_alloc\\_pages](#)(\_xch, [HYPERCALL\\_BUFFER](#)(\_name), \_nr)
- #define [xc\\_hypercall\\_buffer\\_free\\_pages](#)(\_xch, \_name, \_nr) [xc\\_\\_hypercall\\_buffer\\_free\\_pages](#)(\_xch, [HYPERCALL\\_BUFFER](#)(\_name), \_nr)
- #define [XC\\_CORE\\_MAGIC](#) 0xF00FEBED
- #define [XC\\_CORE\\_MAGIC\\_HVM](#) 0xF00FEBEE
- #define [XC\\_MAX\\_ERROR\\_MSG\\_LEN](#) 1024

## Typedefs

- typedef struct xc\_interface\_core [xc\\_interface](#)
- typedef struct xc\_interface\_core [xc\\_evtchn](#)
- typedef struct xc\_interface\_core [xc\\_gnttab](#)
- typedef enum [xc\\_error\\_code](#) [xc\\_error\\_code](#)
- typedef struct [xc\\_hypercall\\_buffer](#) [xc\\_hypercall\\_buffer\\_t](#)
- typedef uint8\_t \* [xc\\_cpumap\\_t](#)
- typedef struct [xc\\_core\\_header](#) [xc\\_core\\_header\\_t](#)
- typedef struct [xc\\_dominfo](#) [xc\\_dominfo\\_t](#)
- typedef [xen\\_domctl\\_getdomaininfo\\_t](#) [xc\\_domaininfo\\_t](#)
- typedef int( [dumpcore\\_rtn\\_t](#) )(xc\_interface \*xch, void \*arg, char \*buffer, unsigned int length)
- typedef [xen\\_domctl\\_getvcpuinfo\\_t](#) [xc\\_vcpuinfo\\_t](#)
- typedef [xen\\_domctl\\_shadow\\_op\\_stats\\_t](#) [xc\\_shadow\\_op\\_stats\\_t](#)
- typedef struct [xc\\_cpupoolinfo](#) [xc\\_cpupoolinfo\\_t](#)
- typedef int [evtchn\\_port\\_or\\_error\\_t](#)
- typedef struct evtchn\_status [xc\\_evtchn\\_status\\_t](#)
- typedef [xen\\_sysctl\\_physinfo\\_t](#) [xc\\_physinfo\\_t](#)
- typedef [xen\\_sysctl\\_topologyinfo\\_t](#) [xc\\_topologyinfo\\_t](#)
- typedef [xen\\_sysctl\\_numainfo\\_t](#) [xc\\_numainfo\\_t](#)
- typedef uint32\_t [xc\\_cpu\\_to\\_node\\_t](#)
- typedef uint32\_t [xc\\_cpu\\_to\\_socket\\_t](#)

- typedef uint32\_t [xc\\_cpu\\_to\\_core\\_t](#)
- typedef uint64\_t [xc\\_node\\_to\\_memszie\\_t](#)
- typedef uint64\_t [xc\\_node\\_to\\_memfree\\_t](#)
- typedef uint32\_t [xc\\_node\\_to\\_node\\_dist\\_t](#)
- typedef xen\_sysctl\_cpuinfo\_t [xc\\_cpuinfo\\_t](#)
- typedef xen\_sysctl\_perfc\_desc\_t [xc\\_perfc\\_desc\\_t](#)
- typedef xen\_sysctl\_perfc\_val\_t [xc\\_perfc\\_val\\_t](#)
- typedef xen\_sysctl\_lockprof\_data\_t [xc\\_lockprof\\_data\\_t](#)
- typedef struct [xc\\_error](#) [xc\\_error](#)
- typedef struct [xc\\_cx\\_stat](#) [xc\\_cx\\_stat\\_t](#)
- typedef xen\_userspace\_t [xc\\_userspace\\_t](#)
- typedef xen\_ondemand\_t [xc\\_ondemand\\_t](#)

## Enumerations

- enum [xc\\_open\\_flags](#) { [XC\\_OPENFLAG\\_DUMMY](#) = 1 < 0, [XC\\_OPENFLAG\\_NON\\_REENTRANT](#) = 1 < 1 }
- enum [xc\\_error\\_code](#) {  
[XC\\_ERROR\\_NONE](#) = 0, [XC\\_INTERNAL\\_ERROR](#) = 1, [XC\\_INVALID\\_KERNEL](#) = 2, [XC\\_INVALID\\_PARAM](#) = 3,  
[XC\\_OUT\\_OF\\_MEMORY](#) = 4 }

## Functions

- [xc\\_interface](#) \* [xc\\_interface\\_open](#) (xentoollog\_logger \*logger, xentoollog\_logger \*dombuild\_logger, unsigned open\_flags)
- int [xc\\_interface\\_close](#) ([xc\\_interface](#) \*xch)
- int [xc\\_interface\\_is\\_fake](#) (void)
- [xc\\_hypercall\\_buffer\\_t](#) [XC\\_\\_HYPERCALL\\_BUFFER\\_NAME](#) ([HYPERCALL\\_BUFFER\\_NULL](#))
- void \* [xc\\_\\_hypercall\\_buffer\\_alloc](#) ([xc\\_interface](#) \*xch, [xc\\_hypercall\\_buffer\\_t](#) \*b, size\_t size)
- void [xc\\_\\_hypercall\\_buffer\\_free](#) ([xc\\_interface](#) \*xch, [xc\\_hypercall\\_buffer\\_t](#) \*b)
- void \* [xc\\_\\_hypercall\\_buffer\\_alloc\\_pages](#) ([xc\\_interface](#) \*xch, [xc\\_hypercall\\_buffer\\_t](#) \*b, int nr\_pages)
- void [xc\\_\\_hypercall\\_buffer\\_free\\_pages](#) ([xc\\_interface](#) \*xch, [xc\\_hypercall\\_buffer\\_t](#) \*b, int nr\_pages)
- int [xc\\_get\\_max\\_cpus](#) ([xc\\_interface](#) \*xch)
- int [xc\\_get\\_cpumap\\_size](#) ([xc\\_interface](#) \*xch)
- [xc\\_cpumap\\_t](#) [xc\\_cpumap\\_alloc](#) ([xc\\_interface](#) \*xch)
- int [xc\\_domain\\_create](#) ([xc\\_interface](#) \*xch, uint32\_t ssidref, [xen\\_domain\\_handle\\_t](#) handle, uint32\_t flags, uint32\_t \*pdomid)
- int [xc\\_domain\\_dumpcore](#) ([xc\\_interface](#) \*xch, uint32\_t domid, const char \*corename)
- int [xc\\_domain\\_dumpcore\\_via\\_callback](#) ([xc\\_interface](#) \*xch, uint32\_t domid, void \*arg, [dumpcore\\_rtn\\_t](#) dumpcore\_rtn)
- int [xc\\_domain\\_max\\_vcpus](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned int max)
- int [xc\\_domain\\_pause](#) ([xc\\_interface](#) \*xch, uint32\_t domid)
- int [xc\\_domain\\_unpause](#) ([xc\\_interface](#) \*xch, uint32\_t domid)
- int [xc\\_domain\\_destroy](#) ([xc\\_interface](#) \*xch, uint32\_t domid)
- int [xc\\_domain\\_resume](#) ([xc\\_interface](#) \*xch, uint32\_t domid, int fast)
- int [xc\\_domain\\_shutdown](#) ([xc\\_interface](#) \*xch, uint32\_t domid, int reason)
- int [xc\\_watchdog](#) ([xc\\_interface](#) \*xch, uint32\_t id, uint32\_t timeout)
- int [xc\\_vcpu\\_setaffinity](#) ([xc\\_interface](#) \*xch, uint32\_t domid, int vcpu, [xc\\_cpumap\\_t](#) cpumap)
- int [xc\\_vcpu\\_getaffinity](#) ([xc\\_interface](#) \*xch, uint32\_t domid, int vcpu, [xc\\_cpumap\\_t](#) cpumap)
- int [xc\\_domain\\_getinfo](#) ([xc\\_interface](#) \*xch, uint32\_t first\_domid, unsigned int max\_doms, [xc\\_dominfo\\_t](#) \*info)
- int [xc\\_vcpu\\_setcontext](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint32\_t vcpu, [vcpu\\_guest\\_context\\_any\\_t](#) \*ctxt)
- int [xc\\_domain\\_getinfo](#) ([xc\\_interface](#) \*xch, uint32\_t first\_domain, unsigned int max\_domains, [xc\\_domaininfo\\_t](#) \*info)
- int [xc\\_domain\\_hvm\\_getcontext](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint8\_t \*ctxt\_buf, uint32\_t size)

- `int xc_domain_hvm_getcontext_partial (xc_interface *xch, uint32_t domid, uint16_t typecode, uint16_t instance, void *ctxt_buf, uint32_t size)`
- `int xc_domain_hvm_setcontext (xc_interface *xch, uint32_t domid, uint8_t *hvm_ctxt, uint32_t size)`
- `int xc_vcpu_getcontext (xc_interface *xch, uint32_t domid, uint32_t vcpu, vcpu_guest_context_any_t *ctxt)`
- `int xc_vcpu_getinfo (xc_interface *xch, uint32_t domid, uint32_t vcpu, xc_vcpuinfo_t *info)`
- `long long xc_domain_get_cpu_usage (xc_interface *xch, domid_t domid, int vcpu)`
- `int xc_domain_sethandle (xc_interface *xch, uint32_t domid, xen_domain_handle_t handle)`
- `int xc_shadow_control (xc_interface *xch, uint32_t domid, unsigned int sop, xc_hypervisor_buffer_t *dirty_bitmap, unsigned long pages, unsigned long *mb, uint32_t mode, xc_shadow_op_stats_t *stats)`
- `int xc_sedf_domain_set (xc_interface *xch, uint32_t domid, uint64_t period, uint64_t slice, uint64_t latency, uint16_t extratime, uint16_t weight)`
- `int xc_sedf_domain_get (xc_interface *xch, uint32_t domid, uint64_t *period, uint64_t *slice, uint64_t *latency, uint16_t *extratime, uint16_t *weight)`
- `int xc_sched_credit_domain_set (xc_interface *xch, uint32_t domid, struct xen_domctl_sched_credit *sdom)`
- `int xc_sched_credit_domain_get (xc_interface *xch, uint32_t domid, struct xen_domctl_sched_credit *sdom)`
- `int xc_sched_credit2_domain_set (xc_interface *xch, uint32_t domid, struct xen_domctl_sched_credit2 *sdom)`
- `int xc_sched_credit2_domain_get (xc_interface *xch, uint32_t domid, struct xen_domctl_sched_credit2 *sdom)`
- `int xc_sched_arinc653_schedule_set (xc_interface *xch, struct xen_sysctl_arinc653_schedule *schedule)`
- `int xc_sched_arinc653_schedule_get (xc_interface *xch, struct xen_sysctl_arinc653_schedule *schedule)`
- `int xc_domain_send_trigger (xc_interface *xch, uint32_t domid, uint32_t trigger, uint32_t vcpu)`
- `int xc_domain_setdebugging (xc_interface *xch, uint32_t domid, unsigned int enable)`
- `int xc_domain_set_access_required (xc_interface *xch, uint32_t domid, unsigned int required)`
- `int xc_cpupool_create (xc_interface *xch, uint32_t *ppoolid, uint32_t sched_id)`
- `int xc_cpupool_destroy (xc_interface *xch, uint32_t poolid)`
- `xc_cpupoolinfo_t * xc_cpupool_getinfo (xc_interface *xch, uint32_t poolid)`
- `void xc_cpupool_infotree (xc_interface *xch, xc_cpupoolinfo_t *info)`
- `int xc_cpupool_addcpu (xc_interface *xch, uint32_t poolid, int cpu)`
- `int xc_cpupool_removecpu (xc_interface *xch, uint32_t poolid, int cpu)`
- `int xc_cpupool_movedomain (xc_interface *xch, uint32_t poolid, uint32_t domid)`
- `xc_cpumap_t xc_cpupool_freeinfo (xc_interface *xch)`
- `evtchn_port_or_error_t xc_evtchn_alloc_unbound (xc_interface *xch, uint32_t dom, uint32_t remote_dom)`
- `int xc_evtchn_reset (xc_interface *xch, uint32_t dom)`
- `int xc_evtchn_status (xc_interface *xch, xc_evtchn_status_t *status)`
- `xc_evtchn * xc_evtchn_open (xentoollog_logger *logger, unsigned open_flags)`
- `int xc_evtchn_close (xc_evtchn *xce)`
- `int xc_evtchn_fd (xc_evtchn *xce)`
- `int xc_evtchn_notify (xc_evtchn *xce, evtchn_port_t port)`
- `evtchn_port_or_error_t xc_evtchn_bind_unbound_port (xc_evtchn *xce, int domid)`
- `evtchn_port_or_error_t xc_evtchn_bind_interdomain (xc_evtchn *xce, int domid, evtchn_port_t remote_port)`
- `evtchn_port_or_error_t xc_evtchn_bind_virq (xc_evtchn *xce, unsigned int virq)`
- `int xc_evtchn_unbind (xc_evtchn *xce, evtchn_port_t port)`
- `evtchn_port_or_error_t xc_evtchn_pending (xc_evtchn *xce)`
- `int xc_evtchn_unmask (xc_evtchn *xce, evtchn_port_t port)`
- `int xc_physdev_pci_access_modify (xc_interface *xch, uint32_t domid, int bus, int dev, int func, int enable)`
- `int xc_readconsolering (xc_interface *xch, char *buffer, unsigned int *pnr_chars, int clear, int incremental, uint32_t *pindex)`
- `int xc_send_debug_keys (xc_interface *xch, char *keys)`
- `int xc_physinfo (xc_interface *xch, xc_physinfo_t *info)`
- `int xc_topologyinfo (xc_interface *xch, xc_topologyinfo_t *info)`
- `int xc_numainfo (xc_interface *xch, xc_numainfo_t *info)`
- `int xc_sched_id (xc_interface *xch, int *sched_id)`
- `int xc_machphys_mfn_list (xc_interface *xch, unsigned long max_extents, xen_pfn_t *extent_start)`
- `int xc_getcpuinfo (xc_interface *xch, int max_cpus, xc_cpuinfo_t *info, int *nr_cpus)`



- int [xc\\_domain\\_setmaxmem](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned int max\_memkb)
- int [xc\\_domain\\_set\\_memmap\\_limit](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long map\_limitkb)
- int [xc\\_domain\\_set\\_time\\_offset](#) ([xc\\_interface](#) \*xch, uint32\_t domid, int32\_t time\_offset\_seconds)
- int [xc\\_domain\\_set\\_tsc\\_info](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint32\_t tsc\_mode, uint64\_t elapsed\_nsec, uint32\_t gtsc\_khz, uint32\_t incarnation)
- int [xc\\_domain\\_get\\_tsc\\_info](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint32\_t \*tsc\_mode, uint64\_t \*elapsed\_nsec, uint32\_t \*gtsc\_khz, uint32\_t \*incarnation)
- int [xc\\_domain\\_disable\\_migrate](#) ([xc\\_interface](#) \*xch, uint32\_t domid)
- int [xc\\_domain\\_maximum\\_gpfn](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) domid)
- int [xc\\_domain\\_increase\\_reservation](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, unsigned int mem\_flags, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_increase\\_reservation\\_exact](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, unsigned int mem\_flags, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_decrease\\_reservation](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_decrease\\_reservation\\_exact](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_add\\_to\\_physmap](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned int space, unsigned long idx, xen\_pfn\_t gpfn)
- int [xc\\_domain\\_populate\\_physmap](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, unsigned int mem\_flags, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_populate\\_physmap\\_exact](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long nr\_extents, unsigned int extent\_order, unsigned int mem\_flags, xen\_pfn\_t \*extent\_start)
- int [xc\\_domain\\_memory\\_exchange\\_pages](#) ([xc\\_interface](#) \*xch, int domid, unsigned long nr\_in\_extents, unsigned int in\_order, xen\_pfn\_t \*in\_extents, unsigned long nr\_out\_extents, unsigned int out\_order, xen\_pfn\_t \*out\_extents)
- int [xc\\_domain\\_set\\_pod\\_target](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint64\_t target\_pages, uint64\_t \*tot\_pages, uint64\_t \*pod\_cache\_pages, uint64\_t \*pod\_entries)
- int [xc\\_domain\\_get\\_pod\\_target](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint64\_t \*tot\_pages, uint64\_t \*pod\_cache\_pages, uint64\_t \*pod\_entries)
- int [xc\\_domain\\_ioport\\_permission](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint32\_t first\_port, uint32\_t nr\_ports, uint32\_t allow\_access)
- int [xc\\_domain\\_irq\\_permission](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint8\_t irq, uint8\_t allow\_access)
- int [xc\\_domain\\_iomem\\_permission](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long first\_mfn, unsigned long nr\_mfns, uint8\_t allow\_access)
- int [xc\\_domain\\_pin\\_memory\\_cacheattr](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint64\_t start, uint64\_t end, uint32\_t type)
- unsigned long [xc\\_make\\_page\\_below\\_4G](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long mfn)
- int [xc\\_perfc\\_reset](#) ([xc\\_interface](#) \*xch)
- int [xc\\_perfc\\_query\\_number](#) ([xc\\_interface](#) \*xch, int \*nbr\_desc, int \*nbr\_val)
- int [xc\\_perfc\\_query](#) ([xc\\_interface](#) \*xch, [xc\\_hypcall\\_buffer\\_t](#) \*desc, [xc\\_hypcall\\_buffer\\_t](#) \*val)
- int [xc\\_lockprof\\_reset](#) ([xc\\_interface](#) \*xch)
- int [xc\\_lockprof\\_query\\_number](#) ([xc\\_interface](#) \*xch, uint32\_t \*n\_elems)
- int [xc\\_lockprof\\_query](#) ([xc\\_interface](#) \*xch, uint32\_t \*n\_elems, uint64\_t \*time, [xc\\_hypcall\\_buffer\\_t](#) \*data)
- void \* [xc\\_map\\_foreign\\_range](#) ([xc\\_interface](#) \*xch, uint32\_t dom, int size, int prot, unsigned long mfn)
- void \* [xc\\_map\\_foreign\\_pages](#) ([xc\\_interface](#) \*xch, uint32\_t dom, int prot, const xen\_pfn\_t \*arr, int num)
- void \* [xc\\_map\\_foreign\\_batch](#) ([xc\\_interface](#) \*xch, uint32\_t dom, int prot, xen\_pfn\_t \*arr, int num)
- void \* [xc\\_map\\_foreign\\_bulk](#) ([xc\\_interface](#) \*xch, uint32\_t dom, int prot, const xen\_pfn\_t \*arr, int \*err, unsigned int num)
- unsigned long [xc\\_translate\\_foreign\\_address](#) ([xc\\_interface](#) \*xch, uint32\_t dom, int vcpu, unsigned long long virt)
- int [xc\\_get\\_pfn\\_list](#) ([xc\\_interface](#) \*xch, uint32\_t domid, uint64\_t \*pfn\_buf, unsigned long max\_pfns)
- unsigned long [xc\\_ia64\\_fpsr\\_default](#) (void)
- int [xc\\_copy\\_to\\_domain\\_page](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long dst\_pfn, const char \*src\_page)
- int [xc\\_clear\\_domain\\_page](#) ([xc\\_interface](#) \*xch, uint32\_t domid, unsigned long dst\_pfn)

- int [xc\\_mmuext\\_op](#) ([xc\\_interface](#) \*xch, struct [mmuext\\_op](#) \*op, unsigned int nr\_ops, [domid\\_t](#) dom)
- long [xc\\_maximum\\_ram\\_page](#) ([xc\\_interface](#) \*xch)
- long [xc\\_get\\_tot\\_pages](#) ([xc\\_interface](#) \*xch, [uint32\\_t](#) domid)
- int [xc\\_availheap](#) ([xc\\_interface](#) \*xch, int min\_width, int max\_width, int node, [uint64\\_t](#) \*bytes)
- int [xc\\_tbuf\\_enable](#) ([xc\\_interface](#) \*xch, unsigned long pages, unsigned long \*mfn, unsigned long \*size)
- int [xc\\_tbuf\\_disable](#) ([xc\\_interface](#) \*xch)
- int [xc\\_tbuf\\_set\\_size](#) ([xc\\_interface](#) \*xch, unsigned long size)
- int [xc\\_tbuf\\_get\\_size](#) ([xc\\_interface](#) \*xch, unsigned long \*size)
- int [xc\\_tbuf\\_set\\_cpu\\_mask](#) ([xc\\_interface](#) \*xch, [uint32\\_t](#) mask)
- int [xc\\_tbuf\\_set\\_evt\\_mask](#) ([xc\\_interface](#) \*xch, [uint32\\_t](#) mask)
- int [xc\\_domctl](#) ([xc\\_interface](#) \*xch, struct [xen\\_domctl](#) \*domctl)
- int [xc\\_sysctl](#) ([xc\\_interface](#) \*xch, struct [xen\\_sysctl](#) \*sysctl)
- int [xc\\_version](#) ([xc\\_interface](#) \*xch, int cmd, void \*arg)
- int [xc\\_acm\\_op](#) ([xc\\_interface](#) \*xch, int cmd, void \*arg, unsigned long arg\_size)
- int [xc\\_flask\\_op](#) ([xc\\_interface](#) \*xch, [flask\\_op\\_t](#) \*op)
- int [xc\\_caas\\_op](#) ([xc\\_interface](#) \*xch, [caas\\_op\\_t](#) \*op)
- int [xc\\_domain\\_subscribe\\_for\\_suspend](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) domid, [evtchn\\_port\\_t](#) port)
- [xc\\_gnttab](#) \* [xc\\_gnttab\\_open](#) ([xentoollog\\_logger](#) \*logger, unsigned open\_flags)
- int [xc\\_gnttab\\_close](#) ([xc\\_gnttab](#) \*xcg)
- void \* [xc\\_gnttab\\_map\\_grant\\_ref](#) ([xc\\_gnttab](#) \*xcg, [uint32\\_t](#) domid, [uint32\\_t](#) ref, int prot)
- void \* [xc\\_gnttab\\_map\\_grant\\_refs](#) ([xc\\_gnttab](#) \*xcg, [uint32\\_t](#) count, [uint32\\_t](#) \*domids, [uint32\\_t](#) \*refs, int prot)
- void \* [xc\\_gnttab\\_map\\_domain\\_grant\\_refs](#) ([xc\\_gnttab](#) \*xcg, [uint32\\_t](#) count, [uint32\\_t](#) domid, [uint32\\_t](#) \*refs, int prot)
- int [xc\\_gnttab\\_munmap](#) ([xc\\_gnttab](#) \*xcg, void \*start\_address, [uint32\\_t](#) count)
- int [xc\\_gnttab\\_set\\_max\\_grants](#) ([xc\\_gnttab](#) \*xcg, [uint32\\_t](#) count)
- int [xc\\_gnttab\\_op](#) ([xc\\_interface](#) \*xch, int cmd, void \*op, int op\_size, int count)
- int [xc\\_gnttab\\_get\\_version](#) ([xc\\_interface](#) \*xch, int domid)
- [grant\\_entry\\_v1\\_t](#) \* [xc\\_gnttab\\_map\\_table\\_v1](#) ([xc\\_interface](#) \*xch, int domid, int \*gnt\_num)
- [grant\\_entry\\_v2\\_t](#) \* [xc\\_gnttab\\_map\\_table\\_v2](#) ([xc\\_interface](#) \*xch, int domid, int \*gnt\_num)
- int [xc\\_physdev\\_map\\_pirq](#) ([xc\\_interface](#) \*xch, int domid, int index, int \*pirq)
- int [xc\\_physdev\\_map\\_pirq\\_msi](#) ([xc\\_interface](#) \*xch, int domid, int index, int \*pirq, int devfn, int bus, int entry\_nr, [uint64\\_t](#) table\_base)
- int [xc\\_physdev\\_unmap\\_pirq](#) ([xc\\_interface](#) \*xch, int domid, int pirq)
- int [xc\\_hvm\\_set\\_pci\\_intx\\_level](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint8\\_t](#) domain, [uint8\\_t](#) bus, [uint8\\_t](#) device, [uint8\\_t](#) intx, unsigned int level)
- int [xc\\_hvm\\_set\\_isa\\_irq\\_level](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint8\\_t](#) isa\_irq, unsigned int level)
- int [xc\\_hvm\\_set\\_pci\\_link\\_route](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint8\\_t](#) link, [uint8\\_t](#) isa\_irq)
- int [xc\\_hvm\\_track\\_dirty\\_vram](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint64\\_t](#) first\_pfn, [uint64\\_t](#) nr, unsigned long \*bitmap)
- int [xc\\_hvm\\_modified\\_memory](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint64\\_t](#) first\_pfn, [uint64\\_t](#) nr)
- int [xc\\_hvm\\_set\\_mem\\_type](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [hvmem\\_type\\_t](#) memtype, [uint64\\_t](#) first\_pfn, [uint64\\_t](#) nr)
- int [xc\\_hvm\\_set\\_mem\\_access](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [hvmem\\_access\\_t](#) memaccess, [uint64\\_t](#) first\_pfn, [uint64\\_t](#) nr)
- int [xc\\_hvm\\_get\\_mem\\_access](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, [uint64\\_t](#) pfn, [hvmem\\_access\\_t](#) \*memaccess)
- int [xc\\_hvm\\_inject\\_trap](#) ([xc\\_interface](#) \*xch, [domid\\_t](#) dom, int [vcpu](#), [uint32\\_t](#) trap, [uint32\\_t](#) error\_code, [uint64\\_t](#) cr2)
- const char \* [xc\\_error\\_code\\_to\\_desc](#) (int code)
- const char \* [xc\\_strerror](#) ([xc\\_interface](#) \*xch, int errcode)
- const [xc\\_error](#) \* [xc\\_get\\_last\\_error](#) ([xc\\_interface](#) \*handle)
- void [xc\\_clear\\_last\\_error](#) ([xc\\_interface](#) \*xch)
- int [xc\\_set\\_hvm\\_param](#) ([xc\\_interface](#) \*handle, [domid\\_t](#) dom, int param, unsigned long value)
- int [xc\\_get\\_hvm\\_param](#) ([xc\\_interface](#) \*handle, [domid\\_t](#) dom, int param, unsigned long \*value)
- int [xc\\_ia64\\_save\\_to\\_nvram](#) ([xc\\_interface](#) \*xch, [uint32\\_t](#) dom)

- int [xc\\_ia64\\_nvram\\_init](#) (xc\_interface \*xch, char \*dom\_name, uint32\_t dom)
- int [xc\\_ia64\\_set\\_os\\_type](#) (xc\_interface \*xch, char \*guest\_os\_type, uint32\_t dom)
- int [xc\\_assign\\_device](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t machine\_bdf)
- int [xc\\_get\\_device\\_group](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t machine\_bdf, uint32\_t max\_sdevs, uint32\_t \*num\_sdevs, uint32\_t \*sdev\_array)
- int [xc\\_test\\_assign\\_device](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t machine\_bdf)
- int [xc\\_deassign\\_device](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t machine\_bdf)
- int [xc\\_domain\\_memory\\_mapping](#) (xc\_interface \*xch, uint32\_t domid, unsigned long first\_gfn, unsigned long first\_mfn, unsigned long nr\_mfns, uint32\_t add\_mapping)
- int [xc\\_domain\\_ioport\\_mapping](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t first\_gport, uint32\_t first\_mport, uint32\_t nr\_ports, uint32\_t add\_mapping)
- int [xc\\_domain\\_update\\_msi\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t gvec, uint32\_t pirq, uint32\_t gflags, uint64\_t gtable)
- int [xc\\_domain\\_unbind\\_msi\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t gvec, uint32\_t pirq, uint32\_t gflags)
- int [xc\\_domain\\_bind\\_pt\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint8\_t machine\_irq, uint8\_t irq\_type, uint8\_t bus, uint8\_t device, uint8\_t intx, uint8\_t isa\_irq)
- int [xc\\_domain\\_unbind\\_pt\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint8\_t machine\_irq, uint8\_t irq\_type, uint8\_t bus, uint8\_t device, uint8\_t intx, uint8\_t isa\_irq)
- int [xc\\_domain\\_bind\\_pt\\_pci\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint8\_t machine\_irq, uint8\_t bus, uint8\_t device, uint8\_t intx)
- int [xc\\_domain\\_bind\\_pt\\_isa\\_irq](#) (xc\_interface \*xch, uint32\_t domid, uint8\_t machine\_irq)
- int [xc\\_domain\\_set\\_machine\\_address\\_size](#) (xc\_interface \*xch, uint32\_t domid, unsigned int width)
- int [xc\\_domain\\_get\\_machine\\_address\\_size](#) (xc\_interface \*xch, uint32\_t domid)
- int [xc\\_domain\\_suppress\\_spurious\\_page\\_faults](#) (xc\_interface \*xch, uint32\_t domid)
- int [xc\\_domain\\_set\\_target](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t target)
- int [xc\\_domain\\_debug\\_control](#) (xc\_interface \*xch, uint32\_t domid, uint32\_t sop, uint32\_t vcpu)
- int [xc\\_pm\\_get\\_max\\_px](#) (xc\_interface \*xch, int cpuid, int \*max\_px)
- int [xc\\_pm\\_get\\_pxstat](#) (xc\_interface \*xch, int cpuid, struct [xc\\_px\\_stat](#) \*pxpt)
- int [xc\\_pm\\_reset\\_pxstat](#) (xc\_interface \*xch, int cpuid)
- int [xc\\_pm\\_get\\_max\\_cx](#) (xc\_interface \*xch, int cpuid, int \*max\_cx)
- int [xc\\_pm\\_get\\_cxstat](#) (xc\_interface \*xch, int cpuid, struct [xc\\_cx\\_stat](#) \*cxpt)
- int [xc\\_pm\\_reset\\_cxstat](#) (xc\_interface \*xch, int cpuid)
- int [xc\\_cpu\\_online](#) (xc\_interface \*xch, int cpu)
- int [xc\\_cpu\\_offline](#) (xc\_interface \*xch, int cpu)
- int [xc\\_get\\_cpufreq\\_para](#) (xc\_interface \*xch, int cpuid, struct [xc\\_get\\_cpufreq\\_para](#) \*user\_para)
- int [xc\\_set\\_cpufreq\\_gov](#) (xc\_interface \*xch, int cpuid, char \*govname)
- int [xc\\_set\\_cpufreq\\_para](#) (xc\_interface \*xch, int cpuid, int ctrl\_type, int ctrl\_value)
- int [xc\\_get\\_cpufreq\\_avgfreq](#) (xc\_interface \*xch, int cpuid, int \*avg\_freq)
- int [xc\\_set\\_sched\\_opt\\_smt](#) (xc\_interface \*xch, uint32\_t value)
- int [xc\\_set\\_vcpu\\_migration\\_delay](#) (xc\_interface \*xch, uint32\_t value)
- int [xc\\_get\\_vcpu\\_migration\\_delay](#) (xc\_interface \*xch, uint32\_t \*value)
- int [xc\\_get\\_cpuidle\\_max\\_cstate](#) (xc\_interface \*xch, uint32\_t \*value)
- int [xc\\_set\\_cpuidle\\_max\\_cstate](#) (xc\_interface \*xch, uint32\_t value)
- int [xc\\_enable\\_turbo](#) (xc\_interface \*xch, int cpuid)
- int [xc\\_disable\\_turbo](#) (xc\_interface \*xch, int cpuid)
- int [xc\\_tmem\\_control\\_oid](#) (xc\_interface \*xch, int32\_t pool\_id, uint32\_t subop, uint32\_t cli\_id, uint32\_t arg1, uint32\_t arg2, struct [tmem\\_oid](#) oid, void \*buf)
- int [xc\\_tmem\\_control](#) (xc\_interface \*xch, int32\_t pool\_id, uint32\_t subop, uint32\_t cli\_id, uint32\_t arg1, uint32\_t arg2, uint64\_t arg3, void \*buf)
- int [xc\\_tmem\\_auth](#) (xc\_interface \*xch, int cli\_id, char \*uuid\_str, int arg1)
- int [xc\\_tmem\\_save](#) (xc\_interface \*xch, int dom, int live, int fd, int field\_marker)
- int [xc\\_tmem\\_save\\_extra](#) (xc\_interface \*xch, int dom, int fd, int field\_marker)
- void [xc\\_tmem\\_save\\_done](#) (xc\_interface \*xch, int dom)
- int [xc\\_tmem\\_restore](#) (xc\_interface \*xch, int dom, int fd)

- int [xc\\_tmem\\_restore\\_extra](#) (xc\_interface \*xch, int dom, int fd)
- int [xc\\_mem\\_event\\_control](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned int op, unsigned int mode, void \*shared\_page, void \*ring\_page, unsigned long gfn)
- int [xc\\_mem\\_event\\_enable](#) (xc\_interface \*xch, domid\_t domain\_id, void \*shared\_page, void \*ring\_page)
- int [xc\\_mem\\_event\\_disable](#) (xc\_interface \*xch, domid\_t domain\_id)
- int [xc\\_mem\\_paging\\_nominate](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned long gfn)
- int [xc\\_mem\\_paging\\_evict](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned long gfn)
- int [xc\\_mem\\_paging\\_prep](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned long gfn)
- int [xc\\_mem\\_paging\\_resume](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned long gfn)
- int [xc\\_mem\\_access\\_resume](#) (xc\_interface \*xch, domid\_t domain\_id, unsigned long gfn)
- int [xc\\_memshr\\_control](#) (xc\_interface \*xch, uint32\_t domid, int enable)
- int [xc\\_memshr\\_nominate\\_gfn](#) (xc\_interface \*xch, uint32\_t domid, unsigned long gfn, uint64\_t \*handle)
- int [xc\\_memshr\\_nominate\\_gref](#) (xc\_interface \*xch, uint32\_t domid, grant\_ref\_t gref, uint64\_t \*handle)
- int [xc\\_memshr\\_share](#) (xc\_interface \*xch, uint64\_t source\_handle, uint64\_t client\_handle)
- int [xc\\_memshr\\_domain\\_resume](#) (xc\_interface \*xch, uint32\_t domid)
- int [xc\\_memshr\\_debug\\_gfn](#) (xc\_interface \*xch, uint32\_t domid, unsigned long gfn)
- int [xc\\_memshr\\_debug\\_mfn](#) (xc\_interface \*xch, uint32\_t domid, unsigned long mfn)
- int [xc\\_memshr\\_debug\\_gref](#) (xc\_interface \*xch, uint32\_t domid, grant\_ref\_t gref)
- int [xc\\_flask\\_load](#) (xc\_interface \*xc\_handle, char \*buf, uint32\_t size)
- int [xc\\_flask\\_context\\_to\\_sid](#) (xc\_interface \*xc\_handle, char \*buf, uint32\_t size, uint32\_t \*sid)
- int [xc\\_flask\\_sid\\_to\\_context](#) (xc\_interface \*xc\_handle, int sid, char \*buf, uint32\_t size)
- int [xc\\_flask\\_getenforce](#) (xc\_interface \*xc\_handle)
- int [xc\\_flask\\_setenforce](#) (xc\_interface \*xc\_handle, int mode)
- int [xc\\_flask\\_add\\_pirq](#) (xc\_interface \*xc\_handle, unsigned int pirq, char \*scontext)
- int [xc\\_flask\\_add\\_ioport](#) (xc\_interface \*xc\_handle, unsigned long low, unsigned long high, char \*scontext)
- int [xc\\_flask\\_add\\_iomem](#) (xc\_interface \*xc\_handle, unsigned long low, unsigned long high, char \*scontext)
- int [xc\\_flask\\_add\\_device](#) (xc\_interface \*xc\_handle, unsigned long device, char \*scontext)
- int [xc\\_flask\\_del\\_pirq](#) (xc\_interface \*xc\_handle, unsigned int pirq)
- int [xc\\_flask\\_del\\_ioport](#) (xc\_interface \*xc\_handle, unsigned long low, unsigned long high)
- int [xc\\_flask\\_del\\_iomem](#) (xc\_interface \*xc\_handle, unsigned long low, unsigned long high)
- int [xc\\_flask\\_del\\_device](#) (xc\_interface \*xc\_handle, unsigned long device)
- int [xc\\_flask\\_access](#) (xc\_interface \*xc\_handle, const char \*scon, const char \*tcon, uint16\_t tclass, uint32\_t req, uint32\_t \*allowed, uint32\_t \*decided, uint32\_t \*auditallow, uint32\_t \*auditdeny, uint32\_t \*seqno)
- int [xc\\_flask\\_avc\\_cachestats](#) (xc\_interface \*xc\_handle, char \*buf, int size)
- int [xc\\_flask\\_policyvers](#) (xc\_interface \*xc\_handle, char \*buf, int size)
- int [xc\\_flask\\_avc\\_hashstats](#) (xc\_interface \*xc\_handle, char \*buf, int size)
- int [xc\\_flask\\_getavc\\_threshold](#) (xc\_interface \*xc\_handle)
- int [xc\\_flask\\_setavc\\_threshold](#) (xc\_interface \*xc\_handle, int threshold)
- int [xc\\_caas\\_exempt](#) (xc\_interface \*xc\_handle, uint64\_t mfn)
- void [xc\\_elf\\_set\\_logfile](#) (xc\_interface \*xch, struct elf\_binary \*elf, int verbose)

## 4.131.1 Macro Definition Documentation

### 4.131.1.1 #define \_\_XEN\_TOOLS\_\_ 1

### 4.131.1.2 #define DECLARE\_HYPERCALL\_BUFFER( \_type, \_name )

Value:

```

_type *_name = NULL;
xc_hypcall_buffer_t XC_HYPERCALL_BUFFER_NAME(_name) = { \
    .hbuf = NULL, \
    .param_shadow = NULL, \
    HYPERCALL_BUFFER_INIT_NO_BOUNCE \
}

```

4.131.1.3 #define DECLARE\_HYPERCALL\_BUFFER\_ARGUMENT( *\_name* )**Value:**

```
xc_hypcall_buffer_t XC__HYPERCALL_BUFFER_NAME(_name) = { \
    .hbuf = (void *)-1, \
    .param_shadow = _name, \
    HYPERCALL_BUFFER_INIT_NO_BOUNCE \
}
```

4.131.1.4 #define HYPERCALL\_BUFFER( *\_name* )**Value:**

```
(( xc_hypcall_buffer_t _val1; \
    typeof(XC__HYPERCALL_BUFFER_NAME(_name)) *_val2 = & \
    XC__HYPERCALL_BUFFER_NAME(_name); \
    (void) (&_val1 == _val2); \
    (_val2)->param_shadow ? (_val2)->param_shadow : (_val2); \
))
```

4.131.1.5 #define HYPERCALL\_BUFFER\_AS\_ARG( *\_name* )**Value:**

```
(( xc_hypcall_buffer_t _val1; \
    typeof(XC__HYPERCALL_BUFFER_NAME(_name)) *_val2 = \
    HYPERCALL_BUFFER(_name); \
    (void) (&_val1 == _val2); \
    (unsigned long) (_val2)->hbuf; \
))
```

## 4.131.1.6 #define HYPERCALL\_BUFFER\_INIT\_NO\_BOUNCE .dir = 0, .sz = 0, .ubuf = (void \*)-1

## 4.131.1.7 #define INVALID\_MFN (~0UL)

4.131.1.8 #define set\_xen\_guest\_handle( *\_hnd*, *\_val* )**Value:**

```
do { \
    xc_hypcall_buffer_t _val1; \
    typeof(XC__HYPERCALL_BUFFER_NAME(_val)) *_val2 = \
    HYPERCALL_BUFFER(_val); \
    (void) (&_val1 == _val2); \
    set_xen_guest_handle_raw(_hnd, (_val2)->hbuf; \
} while (0)
```

4.131.1.9 #define XC\_\_HYPERCALL\_BUFFER\_NAME( *\_name* ) xc\_hypcall\_buffer\_##\_name

## 4.131.1.10 #define XC\_CORE\_MAGIC 0xF00FEBED

## 4.131.1.11 #define XC\_CORE\_MAGIC\_HVM 0xF00FEBEE

4.131.1.12 #define xc\_hypcall\_buffer\_alloc( *\_xch*, *\_name*, *\_size* ) xc\_hypcall\_buffer\_alloc(\_xch, HYPERCALL\_BUFFER(\_name), \_size)

```

4.131.1.13 #define xc_hypcall_buffer_alloc_pages( _xch, _name, _nr ) xc__hypcall_buffer_alloc_pages(_xch,
HYPERCALL_BUFFER(_name), _nr)

4.131.1.14 #define xc_hypcall_buffer_free( _xch, _name ) xc__hypcall_buffer_free(_xch,
HYPERCALL_BUFFER(_name))

4.131.1.15 #define xc_hypcall_buffer_free_pages( _xch, _name, _nr ) xc__hypcall_buffer_free_pages(_xch,
HYPERCALL_BUFFER(_name), _nr)

4.131.1.16 #define XC_MAX_ERROR_MSG_LEN 1024

4.131.1.17 #define XC_PAGE_MASK (~(XC_PAGE_SIZE-1))

4.131.1.18 #define XC_PAGE_SHIFT 12

4.131.1.19 #define XC_PAGE_SIZE (1UL << XC_PAGE_SHIFT)

4.131.1.20 #define XENCTRL_HAS_XC_INTERFACE 1

```

#### 4.131.2 Typedef Documentation

```

4.131.2.1 typedef int( dumpcore_rtn_t)(xc_interface *xch, void *arg, char *buffer, unsigned int length)

4.131.2.2 typedef int evtchn_port_or_error_t

4.131.2.3 typedef struct xc_core_header xc_core_header_t

4.131.2.4 typedef uint32_t xc_cpu_to_core_t

4.131.2.5 typedef uint32_t xc_cpu_to_node_t

4.131.2.6 typedef uint32_t xc_cpu_to_socket_t

4.131.2.7 typedef xen_sysctl_cpuinfo_t xc_cpuinfo_t

4.131.2.8 typedef uint8_t* xc_cpumap_t

4.131.2.9 typedef struct xc_cpupoolinfo xc_cpupoolinfo_t

4.131.2.10 typedef struct xc_cx_stat xc_cx_stat_t

4.131.2.11 typedef xen_domctl_getdomaininfo_t xc_domaininfo_t

4.131.2.12 typedef struct xc_dominfo xc_dominfo_t

4.131.2.13 typedef struct xc_error xc_error

4.131.2.14 typedef enum xc_error_code xc_error_code

4.131.2.15 typedef struct xc_interface_core xc_evtchn

4.131.2.16 typedef struct evtchn_status xc_evtchn_status_t

4.131.2.17 typedef struct xc_interface_core xc_gnttab

4.131.2.18 typedef struct xc_hypcall_buffer xc_hypcall_buffer_t

```

- 4.131.2.19 `typedef struct xc_interface_core xc_interface`
- 4.131.2.20 `typedef xen_sysctl_lockprof_data_t xc_lockprof_data_t`
- 4.131.2.21 `typedef uint64_t xc_node_to_memfree_t`
- 4.131.2.22 `typedef uint64_t xc_node_to_memsizes_t`
- 4.131.2.23 `typedef uint32_t xc_node_to_node_dist_t`
- 4.131.2.24 `typedef xen_sysctl_numainfo_t xc_numainfo_t`
- 4.131.2.25 `typedef xen_ondemand_t xc_ondemand_t`
- 4.131.2.26 `typedef xen_sysctl_perfc_desc_t xc_perfc_desc_t`
- 4.131.2.27 `typedef xen_sysctl_perfc_val_t xc_perfc_val_t`
- 4.131.2.28 `typedef xen_sysctl_physinfo_t xc_physinfo_t`
- 4.131.2.29 `typedef xen_domctl_shadow_op_stats_t xc_shadow_op_stats_t`
- 4.131.2.30 `typedef xen_sysctl_topologyinfo_t xc_topologyinfo_t`
- 4.131.2.31 `typedef xen_userspace_t xc_userspace_t`
- 4.131.2.32 `typedef xen_domctl_getvcpuinfo_t xc_vcpuinfo_t`

### 4.131.3 Enumeration Type Documentation

- 4.131.3.1 `enum xc_error_code`

Enumerator

***XC\_ERROR\_NONE***  
***XC\_INTERNAL\_ERROR***  
***XC\_INVALID\_KERNEL***  
***XC\_INVALID\_PARAM***  
***XC\_OUT\_OF\_MEMORY***

- 4.131.3.2 `enum xc_open_flags`

Enumerator

***XC\_OPENFLAG\_DUMMY***  
***XC\_OPENFLAG\_NON\_REENTRANT***

### 4.131.4 Function Documentation

- 4.131.4.1 `void* xc_hypcall_buffer_alloc ( xc_interface * xch, xc_hypcall_buffer_t * b, size_t size )`
- 4.131.4.2 `void* xc_hypcall_buffer_alloc_pages ( xc_interface * xch, xc_hypcall_buffer_t * b, int nr_pages )`
- 4.131.4.3 `void xc_hypcall_buffer_free ( xc_interface * xch, xc_hypcall_buffer_t * b )`

4.131.4.4 void xc\_\_hypercall\_buffer\_free\_pages ( xc\_interface \* xch, xc\_hypervcall\_buffer\_t \* b, int nr\_pages )

4.131.4.5 xc\_hypervcall\_buffer\_t XC\_\_HYPERCALL\_BUFFER\_NAME ( HYPERCALL\_BUFFER\_NULL )

4.131.4.6 int xc\_acm\_op ( xc\_interface \* xch, int cmd, void \* arg, unsigned long arg\_size )

4.131.4.7 int xc\_assign\_device ( xc\_interface \* xch, uint32\_t domid, uint32\_t machine\_bdf )

4.131.4.8 int xc\_availheap ( xc\_interface \* xch, int min\_width, int max\_width, int node, uint64\_t \* bytes )

This function retrieves the the number of bytes available in the heap in a specific range of address-widths and nodes.  
 xch a handle to an open hypervisor interface domid the domain to query min\_width the smallest address width to query (0 if don't care) max\_width the largest address width to query (0 if don't care) node the node to query (-1 for all) \*bytes caller variable to put total bytes counted

#### Returns

0 on success, <0 on failure.

4.131.4.9 int xc\_caas\_exempt ( xc\_interface \* xc\_handle, uint64\_t mfn )

4.131.4.10 int xc\_caas\_op ( xc\_interface \* xch, caas\_op\_t \* op )

4.131.4.11 int xc\_clear\_domain\_page ( xc\_interface \* xch, uint32\_t domid, unsigned long dst\_pfn )

4.131.4.12 void xc\_clear\_last\_error ( xc\_interface \* xch )

4.131.4.13 int xc\_copy\_to\_domain\_page ( xc\_interface \* xch, uint32\_t domid, unsigned long dst\_pfn, const char \* src\_page )

4.131.4.14 int xc\_cpu\_offline ( xc\_interface \* xch, int cpu )

4.131.4.15 int xc\_cpu\_online ( xc\_interface \* xch, int cpu )

4.131.4.16 xc\_cpumap\_t xc\_cpumap\_alloc ( xc\_interface \* xch )

4.131.4.17 int xc\_cpupool\_addcpu ( xc\_interface \* xch, uint32\_t poolid, int cpu )

Add cpu to a cpupool. cpu may be -1 indicating the first unassigned.

xc\_handle a handle to an open hypervisor interface poolid id of the cpupool cpu cpu number to add return 0 on success, -1 on failure

4.131.4.18 int xc\_cpupool\_create ( xc\_interface \* xch, uint32\_t \* ppoolid, uint32\_t sched\_id )

Create a new cpupool.

xc\_handle a handle to an open hypervisor interface ppoolid pointer to the new cpupool id (in/out) sched\_id id of scheduler to use for pool return 0 on success, -1 on failure

4.131.4.19 int xc\_cpupool\_destroy ( xc\_interface \* xch, uint32\_t poolid )

Destroy a cpupool. Pool must be unused and have no cpu assigned.

xc\_handle a handle to an open hypervisor interface poolid id of the cpupool to destroy return 0 on success, -1 on failure



4.131.4.20 `xc_cpumap_t xc_cpupool_freeinfo ( xc_interface * xch )`

Return map of cpus not in any cpupool.

`xc_handle` a handle to an open hypervisor interface return cpumap array on success, NULL else

4.131.4.21 `xc_cpupoolinfo_t* xc_cpupool_getinfo ( xc_interface * xch, uint32_t poolid )`

Get cpupool info. Returns info for up to the specified number of cpupools starting at the given id. `xc_handle` a handle to an open hypervisor interface poolid lowest id for which info is returned return cpupool info ptr (to be freed via `xc_cpupool_info_free`)

4.131.4.22 `void xc_cpupool_info_free ( xc_interface * xch, xc_cpupoolinfo_t * info )`

Free cpupool info. Used to free info obtained via `xc_cpupool_getinfo`. `xc_handle` a handle to an open hypervisor interface info area to free

4.131.4.23 `int xc_cpupool_movedomain ( xc_interface * xch, uint32_t poolid, uint32_t domid )`

Move domain to another cpupool.

`xc_handle` a handle to an open hypervisor interface poolid id of the destination cpupool domid id of the domain to move return 0 on success, -1 on failure

4.131.4.24 `int xc_cpupool_removecpu ( xc_interface * xch, uint32_t poolid, int cpu )`

Remove cpu from cpupool. cpu may be -1 indicating the last cpu of the pool.

`xc_handle` a handle to an open hypervisor interface poolid id of the cpupool cpu cpu number to remove return 0 on success, -1 on failure

4.131.4.25 `int xc_deassign_device ( xc_interface * xch, uint32_t domid, uint32_t machine_bdf )`4.131.4.26 `int xc_disable_turbo ( xc_interface * xch, int cpuid )`4.131.4.27 `int xc_domain_add_to_physmap ( xc_interface * xch, uint32_t domid, unsigned int space, unsigned long idx, xen_pfn_t gpfns )`4.131.4.28 `int xc_domain_bind_pt_irq ( xc_interface * xch, uint32_t domid, uint8_t machine_irq, uint8_t irq_type, uint8_t bus, uint8_t device, uint8_t intx, uint8_t isa_irq )`4.131.4.29 `int xc_domain_bind_pt_isa_irq ( xc_interface * xch, uint32_t domid, uint8_t machine_irq )`4.131.4.30 `int xc_domain_bind_pt_pci_irq ( xc_interface * xch, uint32_t domid, uint8_t machine_irq, uint8_t bus, uint8_t device, uint8_t intx )`4.131.4.31 `int xc_domain_create ( xc_interface * xch, uint32_t ssidref, xen_domain_handle_t handle, uint32_t flags, uint32_t * pdomid )`4.131.4.32 `int xc_domain_debug_control ( xc_interface * xch, uint32_t domid, uint32_t sop, uint32_t vcpu )`4.131.4.33 `int xc_domain_decrease_reservation ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, xen_pfn_t * extent_start )`

4.131.4.34 `int xc_domain_decrease_reservation_exact ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, xen_pfn_t * extent_start )`

4.131.4.35 `int xc_domain_destroy ( xc_interface * xch, uint32_t domid )`

This function will destroy a domain. Destroying a domain removes the domain completely from memory. This function should be called after sending the domain a SHUTDOWN control message to free up the domain resources.

xch a handle to an open hypervisor interface domid the domain id to destroy

#### Returns

0 on success, -1 on failure

4.131.4.36 `int xc_domain_disable_migrate ( xc_interface * xch, uint32_t domid )`

4.131.4.37 `int xc_domain_dumpcore ( xc_interface * xch, uint32_t domid, const char * corename )`

4.131.4.38 `int xc_domain_dumpcore_via_callback ( xc_interface * xch, uint32_t domid, void * arg, dumpcore_rtn_t dump_rtn )`

4.131.4.39 `long long xc_domain_get_cpu_usage ( xc_interface * xch, domid_t domid, int vcpu )`

4.131.4.40 `int xc_domain_get_machine_address_size ( xc_interface * xch, uint32_t domid )`

4.131.4.41 `int xc_domain_get_pod_target ( xc_interface * xch, uint32_t domid, uint64_t * tot_pages, uint64_t * pod_cache_pages, uint64_t * pod_entries )`

4.131.4.42 `int xc_domain_get_tsc_info ( xc_interface * xch, uint32_t domid, uint32_t * tsc_mode, uint64_t * elapsed_nsec, uint32_t * gtsc_khz, uint32_t * incarnation )`

4.131.4.43 `int xc_domain_getinfo ( xc_interface * xch, uint32_t first_domid, unsigned int max_doms, xc_domaininfo_t * info )`

This function will return information about one or more domains. It is designed to iterate over the list of domains. If a single domain is requested, this function will return the next domain in the list - if one exists. It is, therefore, important in this case to make sure the domain requested was the one returned.

xch a handle to an open hypervisor interface first\_domid the first domain to enumerate information from. Domains are currently enumerate in order of creation. max\_doms the number of elements in info info an array of max\_doms size that will contain the information for the enumerated domains.

#### Returns

the number of domains enumerated or -1 on error

4.131.4.44 `int xc_domain_getinfofolist ( xc_interface * xch, uint32_t first_domain, unsigned int max_domains, xc_domaininfo_t * info )`

This function will return information about one or more domains, using a single hypercall. The domain information will be stored into the supplied array of xc\_domaininfo\_t structures.

xch a handle to an open hypervisor interface first\_domain the first domain to enumerate information from. Domains are currently enumerate in order of creation. max\_domains the number of elements in info info an array of max\_doms size that will contain the information for the enumerated domains.

#### Returns

the number of domains enumerated or -1 on error

4.131.4.45 `int xc_domain_hvm_getcontext ( xc_interface * xch, uint32_t domid, uint8_t * ctxt_buf, uint32_t size )`

This function returns information about the context of a hvm domain xch a handle to an open hypervisor interface domid the domain to get information from ctxt\_buf a pointer to a structure to store the execution context of the hvm domain size the size of ctxt\_buf in bytes

#### Returns

0 on success, -1 on failure

4.131.4.46 `int xc_domain_hvm_getcontext_partial ( xc_interface * xch, uint32_t domid, uint16_t typecode, uint16_t instance, void * ctxt_buf, uint32_t size )`

This function returns one element of the context of a hvm domain xch a handle to an open hypervisor interface domid the domain to get information from typecode which type of elemnt required instance which instance of the type ctxt\_buf a pointer to a structure to store the execution context of the hvm domain size the size of ctxt\_buf (must be  $\geq$  HVM\_SAVE\_LENGTH(typecode))

#### Returns

0 on success, -1 on failure

4.131.4.47 `int xc_domain_hvm_setcontext ( xc_interface * xch, uint32_t domid, uint8_t * hvm_ctxt, uint32_t size )`

This function will set the context for hvm domain

xch a handle to an open hypervisor interface domid the domain to set the hvm domain context for hvm\_ctxt pointer to the the hvm context with the values to set size the size of hvm\_ctxt in bytes

#### Returns

0 on success, -1 on failure

4.131.4.48 `int xc_domain_increase_reservation ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, unsigned int mem_flags, xen_pfn_t * extent_start )`

4.131.4.49 `int xc_domain_increase_reservation_exact ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, unsigned int mem_flags, xen_pfn_t * extent_start )`

4.131.4.50 `int xc_domain_iomem_permission ( xc_interface * xch, uint32_t domid, unsigned long first_mfn, unsigned long nr_mfns, uint8_t allow_access )`

4.131.4.51 `int xc_domain_ioport_mapping ( xc_interface * xch, uint32_t domid, uint32_t first_gport, uint32_t first_mport, uint32_t nr_ports, uint32_t add_mapping )`

4.131.4.52 `int xc_domain_ioport_permission ( xc_interface * xch, uint32_t domid, uint32_t first_port, uint32_t nr_ports, uint32_t allow_access )`

4.131.4.53 `int xc_domain_irq_permission ( xc_interface * xch, uint32_t domid, uint8_t pirq, uint8_t allow_access )`

4.131.4.54 `int xc_domain_max_vcpus ( xc_interface * xch, uint32_t domid, unsigned int max )`

4.131.4.55 `int xc_domain_maximum_gpfn ( xc_interface * xch, domid_t domid )`

4.131.4.56 `int xc_domain_memory_exchange_pages ( xc_interface * xch, int domid, unsigned long nr_in_extents, unsigned int in_order, xen_pfn_t * in_extents, unsigned long nr_out_extents, unsigned int out_order, xen_pfn_t * out_extents )`

4.131.4.57 `int xc_domain_memory_mapping ( xc_interface * xch, uint32_t domid, unsigned long first_gfn, unsigned long first_mfn, unsigned long nr_mfns, uint32_t add_mapping )`

4.131.4.58 `int xc_domain_pause ( xc_interface * xch, uint32_t domid )`

This function pauses a domain. A paused domain still exists in memory however it does not receive any timeslices from the hypervisor.

xch a handle to an open hypervisor interface domid the domain id to pause

#### Returns

0 on success, -1 on failure.

4.131.4.59 `int xc_domain_pin_memory_cacheattr ( xc_interface * xch, uint32_t domid, uint64_t start, uint64_t end, uint32_t type )`

4.131.4.60 `int xc_domain_populate_physmap ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, unsigned int mem_flags, xen_pfn_t * extent_start )`

4.131.4.61 `int xc_domain_populate_physmap_exact ( xc_interface * xch, uint32_t domid, unsigned long nr_extents, unsigned int extent_order, unsigned int mem_flags, xen_pfn_t * extent_start )`

4.131.4.62 `int xc_domain_resume ( xc_interface * xch, uint32_t domid, int fast )`

This function resumes a suspended domain. The domain should have been previously suspended.

xch a handle to an open hypervisor interface domid the domain id to resume fast use cooperative resume (guest must support this) return 0 on success, -1 on failure

4.131.4.63 `int xc_domain_send_trigger ( xc_interface * xch, uint32_t domid, uint32_t trigger, uint32_t vcpu )`

This function sends a trigger to a domain.

xch a handle to an open hypervisor interface domid the domain id to send trigger trigger the trigger type vcpu the vcpu number to send trigger return 0 on success, -1 on failure

4.131.4.64 `int xc_domain_set_access_required ( xc_interface * xch, uint32_t domid, unsigned int required )`

This function sets or clears the requirement that an access memory event listener is required on the domain.

xch a handle to an open hypervisor interface domid the domain id to send trigger enable true to require a listener return 0 on success, -1 on failure

4.131.4.65 `int xc_domain_set_machine_address_size ( xc_interface * xch, uint32_t domid, unsigned int width )`

4.131.4.66 `int xc_domain_set_memmap_limit ( xc_interface * xch, uint32_t domid, unsigned long map_limitkb )`

4.131.4.67 `int xc_domain_set_pod_target ( xc_interface * xch, uint32_t domid, uint64_t target_pages, uint64_t * tot_pages, uint64_t * pod_cache_pages, uint64_t * pod_entries )`

4.131.4.68 `int xc_domain_set_target ( xc_interface * xch, uint32_t domid, uint32_t target )`

4.131.4.69 `int xc_domain_set_time_offset ( xc_interface * xch, uint32_t domid, int32_t time_offset_seconds )`

4.131.4.70 `int xc_domain_set_tsc_info ( xc_interface * xch, uint32_t domid, uint32_t tsc_mode, uint64_t elapsed_nsec, uint32_t gtsc_khz, uint32_t incarnation )`

4.131.4.71 `int xc_domain_setdebugging ( xc_interface * xch, uint32_t domid, unsigned int enable )`

This function enables or disable debugging of a domain.

xch a handle to an open hypervisor interface domid the domain id to send trigger enable true to enable debugging return 0 on success, -1 on failure

4.131.4.72 `int xc_domain_sethandle ( xc_interface * xch, uint32_t domid, xen_domain_handle_t handle )`

4.131.4.73 `int xc_domain_setmaxmem ( xc_interface * xch, uint32_t domid, unsigned int max_memkb )`

4.131.4.74 `int xc_domain_shutdown ( xc_interface * xch, uint32_t domid, int reason )`

This function will shutdown a domain. This is intended for use in fully-virtualized domains where this operation is analogous to the sched\_op operations in a paravirtualized domain. The caller is expected to give the reason for the shutdown.

xch a handle to an open hypervisor interface domid the domain id to destroy reason is the reason (SHUTDOWN\_ - xxx) for the shutdown

#### Returns

0 on success, -1 on failure

4.131.4.75 `int xc_domain_subscribe_for_suspend ( xc_interface * xch, domid_t domid, evtchn_port_t port )`

4.131.4.76 `int xc_domain_suppress_spurious_page_faults ( xc_interface * xch, uint32_t domid )`

4.131.4.77 `int xc_domain_unbind_msi_irq ( xc_interface * xch, uint32_t domid, uint32_t gvec, uint32_t pirq, uint32_t gflags )`

4.131.4.78 `int xc_domain_unbind_pt_irq ( xc_interface * xch, uint32_t domid, uint8_t machine_irq, uint8_t irq_type, uint8_t bus, uint8_t device, uint8_t intx, uint8_t isa_irq )`

4.131.4.79 `int xc_domain_unpause ( xc_interface * xch, uint32_t domid )`

This function unpauses a domain. The domain should have been previously paused.

xch a handle to an open hypervisor interface domid the domain id to unpause return 0 on success, -1 on failure

4.131.4.80 `int xc_domain_update_msi_irq ( xc_interface * xch, uint32_t domid, uint32_t gvec, uint32_t pirq, uint32_t gflags, uint64_t gtable )`

4.131.4.81 `int xc_domctl ( xc_interface * xch, struct xen_domctl * domctl )`

4.131.4.82 `void xc_elf_set_logfile ( xc_interface * xch, struct elf_binary * elf, int verbose )`

4.131.4.83 `int xc_enable_turbo ( xc_interface * xch, int cpuid )`

4.131.4.84 `const char* xc_error_code_to_desc ( int code )`

4.131.4.85 `evtchn_port_or_error_t xc_evtchn_alloc_unbound ( xc_interface * xch, uint32_t dom, uint32_t remote_dom )`

This function allocates an unbound port. Ports are named endpoints used for interdomain communication. This function is most useful in opening a well-known port within a domain to receive events on.

NOTE: If you are allocating a *local* unbound port, you probably want to use [xc\\_evtchn\\_bind\\_unbound\\_port\(\)](#). This function is intended for allocating ports *only* during domain creation.

xch a handle to an open hypervisor interface dom the ID of the local domain (the 'allocattee') remote\_dom the ID of the domain who will later bind

#### Returns

allocated port (in ) on success, -1 on failure

4.131.4.86 `evtchn_port_or_error_t xc_evtchn_bind_interdomain ( xc_evtchn * xce, int domid, evtchn_port_t remote_port )`

4.131.4.87 `evtchn_port_or_error_t xc_evtchn_bind_unbound_port ( xc_evtchn * xce, int domid )`

4.131.4.88 `evtchn_port_or_error_t xc_evtchn_bind_virq ( xc_evtchn * xce, unsigned int virq )`

4.131.4.89 `int xc_evtchn_close ( xc_evtchn * xce )`

4.131.4.90 `int xc_evtchn_fd ( xc_evtchn * xce )`

4.131.4.91 `int xc_evtchn_notify ( xc_evtchn * xce, evtchn_port_t port )`

4.131.4.92 `xc_evtchn* xc_evtchn_open ( xentoollog_logger * logger, unsigned open_flags )`

4.131.4.93 `evtchn_port_or_error_t xc_evtchn_pending ( xc_evtchn * xce )`

4.131.4.94 `int xc_evtchn_reset ( xc_interface * xch, uint32_t dom )`

4.131.4.95 `int xc_evtchn_status ( xc_interface * xch, xc_evtchn_status_t * status )`

4.131.4.96 `int xc_evtchn_unbind ( xc_evtchn * xce, evtchn_port_t port )`

4.131.4.97 `int xc_evtchn_unmask ( xc_evtchn * xce, evtchn_port_t port )`

4.131.4.98 `int xc_flask_access ( xc_interface * xc_handle, const char * scon, const char * tcon, uint16_t tclass, uint32_t req, uint32_t * allowed, uint32_t * decided, uint32_t * auditallow, uint32_t * auditdeny, uint32_t * seqno )`

4.131.4.99 `int xc_flask_add_device ( xc_interface * xc_handle, unsigned long device, char * scontext )`

4.131.4.100 `int xc_flask_add_iomem ( xc_interface * xc_handle, unsigned long low, unsigned long high, char * scontext )`

4.131.4.101 `int xc_flask_add_ioport ( xc_interface * xc_handle, unsigned long low, unsigned long high, char * scontext )`

4.131.4.102 `int xc_flask_add_pirq ( xc_interface * xc_handle, unsigned int pirq, char * scontext )`

4.131.4.103 `int xc_flask_avc_cachestats ( xc_interface * xc_handle, char * buf, int size )`

4.131.4.104 `int xc_flask_avc_hashstats ( xc_interface * xc_handle, char * buf, int size )`

4.131.4.105 `int xc_flask_context_to_sid ( xc_interface * xc_handle, char * buf, uint32_t size, uint32_t * sid )`

- 4.131.4.106 `int xc_flask_del_device ( xc_interface * xc_handle, unsigned long device )`
  - 4.131.4.107 `int xc_flask_del_iomem ( xc_interface * xc_handle, unsigned long low, unsigned long high )`
  - 4.131.4.108 `int xc_flask_del_ioport ( xc_interface * xc_handle, unsigned long low, unsigned long high )`
  - 4.131.4.109 `int xc_flask_del_pirq ( xc_interface * xc_handle, unsigned int pirq )`
  - 4.131.4.110 `int xc_flask_getavc_threshold ( xc_interface * xc_handle )`
  - 4.131.4.111 `int xc_flask_getenforce ( xc_interface * xc_handle )`
  - 4.131.4.112 `int xc_flask_load ( xc_interface * xc_handle, char * buf, uint32_t size )`
  - 4.131.4.113 `int xc_flask_op ( xc_interface * xch, flask_op_t * op )`
  - 4.131.4.114 `int xc_flask_policyvers ( xc_interface * xc_handle, char * buf, int size )`
  - 4.131.4.115 `int xc_flask_setavc_threshold ( xc_interface * xc_handle, int threshold )`
  - 4.131.4.116 `int xc_flask_setenforce ( xc_interface * xc_handle, int mode )`
  - 4.131.4.117 `int xc_flask_sid_to_context ( xc_interface * xc_handle, int sid, char * buf, uint32_t size )`
  - 4.131.4.118 `int xc_get_cpufreq_avgfreq ( xc_interface * xch, int cpuid, int * avg_freq )`
  - 4.131.4.119 `int xc_get_cpufreq_para ( xc_interface * xch, int cpuid, struct xc_get_cpufreq_para * user_para )`
  - 4.131.4.120 `int xc_get_cpuidle_max_cstate ( xc_interface * xch, uint32_t * value )`
  - 4.131.4.121 `int xc_get_cpumap_size ( xc_interface * xch )`
  - 4.131.4.122 `int xc_get_device_group ( xc_interface * xch, uint32_t domid, uint32_t machine_bdf, uint32_t max_sdevs, uint32_t * num_sdevs, uint32_t * sdev_array )`
  - 4.131.4.123 `int xc_get_hvm_param ( xc_interface * handle, domid_t dom, int param, unsigned long * value )`
  - 4.131.4.124 `const xc_error* xc_get_last_error ( xc_interface * handle )`
  - 4.131.4.125 `int xc_get_max_cpus ( xc_interface * xch )`
  - 4.131.4.126 `int xc_get_pfn_list ( xc_interface * xch, uint32_t domid, uint64_t * pfn_buf, unsigned long max_pfns )`
- DEPRECATED. Avoid using this, as it does not correctly account for PFNs without a backing MFN.
- 4.131.4.127 `long xc_get_tot_pages ( xc_interface * xch, uint32_t domid )`
  - 4.131.4.128 `int xc_get_vcpu_migration_delay ( xc_interface * xch, uint32_t * value )`
  - 4.131.4.129 `int xc_getcpuinfo ( xc_interface * xch, int max_cpus, xc_cpuinfo_t * info, int * nr_cpus )`
  - 4.131.4.130 `int xc_gnttab_close ( xc_gnttab * xcg )`
  - 4.131.4.131 `int xc_gnttab_get_version ( xc_interface * xch, int domid )`

4.131.4.132 void\* xc\_gnttab\_map\_domain\_grant\_refs ( xc\_gnttab \* xcg, uint32\_t count, uint32\_t domid, uint32\_t \* refs, int prot )

Memory maps one or more grant references from one domain to a contiguous local address range. Mappings should be unmapped with xc\_gnttab\_munmap. Logs errors.

xcg a handle on an open grant table interface count the number of grant references to be mapped domid the domain to map memory from refs an array of grant references to be mapped prot same flag as in mmap()

4.131.4.133 void\* xc\_gnttab\_map\_grant\_ref ( xc\_gnttab \* xcg, uint32\_t domid, uint32\_t ref, int prot )

4.131.4.134 void\* xc\_gnttab\_map\_grant\_refs ( xc\_gnttab \* xcg, uint32\_t count, uint32\_t \* domids, uint32\_t \* refs, int prot )

Memory maps one or more grant references from one or more domains to a contiguous local address range. Mappings should be unmapped with xc\_gnttab\_munmap. Logs errors.

xcg a handle on an open grant table interface count the number of grant references to be mapped domids an array of domain IDs by which the corresponding were granted refs an array of grant references to be mapped prot same flag as in mmap()

4.131.4.135 grant\_entry\_v1\_t\* xc\_gnttab\_map\_table\_v1 ( xc\_interface \* xch, int domid, int \* gnt\_num )

4.131.4.136 grant\_entry\_v2\_t\* xc\_gnttab\_map\_table\_v2 ( xc\_interface \* xch, int domid, int \* gnt\_num )

4.131.4.137 int xc\_gnttab\_munmap ( xc\_gnttab \* xcg, void \* start\_address, uint32\_t count )

4.131.4.138 int xc\_gnttab\_op ( xc\_interface \* xch, int cmd, void \* op, int op\_size, int count )

4.131.4.139 xc\_gnttab\* xc\_gnttab\_open ( xentoollog\_logger \* logger, unsigned open\_flags )

4.131.4.140 int xc\_gnttab\_set\_max\_grants ( xc\_gnttab \* xcg, uint32\_t count )

4.131.4.141 int xc\_hvm\_get\_mem\_access ( xc\_interface \* xch, domid\_t dom, uint64\_t pfn, hvmmem\_access\_t \* memaccess )

4.131.4.142 int xc\_hvm\_inject\_trap ( xc\_interface \* xch, domid\_t dom, int vcpu, uint32\_t trap, uint32\_t error\_code, uint64\_t cr2 )

4.131.4.143 int xc\_hvm\_modified\_memory ( xc\_interface \* xch, domid\_t dom, uint64\_t first\_pfn, uint64\_t nr )

4.131.4.144 int xc\_hvm\_set\_isa\_irq\_level ( xc\_interface \* xch, domid\_t dom, uint8\_t isa\_irq, unsigned int level )

4.131.4.145 int xc\_hvm\_set\_mem\_access ( xc\_interface \* xch, domid\_t dom, hvmmem\_access\_t memaccess, uint64\_t first\_pfn, uint64\_t nr )

4.131.4.146 int xc\_hvm\_set\_mem\_type ( xc\_interface \* xch, domid\_t dom, hvmmem\_type\_t memtype, uint64\_t first\_pfn, uint64\_t nr )

4.131.4.147 int xc\_hvm\_set\_pci\_intx\_level ( xc\_interface \* xch, domid\_t dom, uint8\_t domain, uint8\_t bus, uint8\_t device, uint8\_t intx, unsigned int level )

4.131.4.148 int xc\_hvm\_set\_pci\_link\_route ( xc\_interface \* xch, domid\_t dom, uint8\_t link, uint8\_t isa\_irq )

4.131.4.149 int xc\_hvm\_track\_dirty\_vram ( xc\_interface \* xch, domid\_t dom, uint64\_t first\_pfn, uint64\_t nr, unsigned long \* bitmap )



4.131.4.150 unsigned long xc\_ia64\_fpsr\_default ( void )

4.131.4.151 int xc\_ia64\_nvram\_init ( xc\_interface \* xch, char \* dom\_name, uint32\_t dom )

4.131.4.152 int xc\_ia64\_save\_to\_nvram ( xc\_interface \* xch, uint32\_t dom )

4.131.4.153 int xc\_ia64\_set\_os\_type ( xc\_interface \* xch, char \* guest\_os\_type, uint32\_t dom )

4.131.4.154 int xc\_interface\_close ( xc\_interface \* xch )

This function closes an open hypervisor interface.

This function can fail if the handle does not represent an open interface or if there were problems closing the interface. In the latter case the interface is still closed.

xch a handle to an open hypervisor interface

Returns

0 on success, -1 otherwise.

4.131.4.155 int xc\_interface\_is\_fake ( void )

Query the active OS interface (i.e. that which would be returned by xc\_interface\_open) to find out if it is fake (i.e. backends onto something other than an actual Xen hypervisor).

Returns

0 is "real", >0 if fake, -1 on error.

4.131.4.156 xc\_interface\* xc\_interface\_open ( xentoollog\_logger \* logger, xentoollog\_logger \* dombuild\_logger, unsigned open\_flags )

This function opens a handle to the hypervisor interface. This function can be called multiple times within a single process. Multiple processes can have an open hypervisor interface at the same time.

Each call to this function should have a corresponding call to [xc\\_interface\\_close\(\)](#).

This function can fail if the caller does not have superuser permission or if a Xen-enabled kernel is not currently running.

Returns

a handle to the hypervisor interface

4.131.4.157 int xc\_lockprof\_query ( xc\_interface \* xch, uint32\_t \* n\_elems, uint64\_t \* time, xc\_hypercall\_buffer\_t \* data )

4.131.4.158 int xc\_lockprof\_query\_number ( xc\_interface \* xch, uint32\_t \* n\_elems )

4.131.4.159 int xc\_lockprof\_reset ( xc\_interface \* xch )

4.131.4.160 int xc\_machphys\_mfn\_list ( xc\_interface \* xch, unsigned long max\_extents, xen\_pfn\_t \* extent\_start )

4.131.4.161 unsigned long xc\_make\_page\_below\_4G ( xc\_interface \* xch, uint32\_t domid, unsigned long mfn )

4.131.4.162 `void* xc_map_foreign_batch ( xc_interface * xch, uint32_t dom, int prot, xen_pfn_t * arr, int num )`

DEPRECATED - use `xc_map_foreign_bulk()` instead.

Like `xc_map_foreign_pages()`, except it can succeed partially. When a page cannot be mapped, its PFN in is or'ed with 0xF0000000 to indicate the error.

4.131.4.163 `void* xc_map_foreign_bulk ( xc_interface * xch, uint32_t dom, int prot, const xen_pfn_t * arr, int * err, unsigned int num )`

Like `xc_map_foreign_pages()`, except it can succeed partially. When a page cannot be mapped, its respective field in is set to the corresponding errno value.

4.131.4.164 `void* xc_map_foreign_pages ( xc_interface * xch, uint32_t dom, int prot, const xen_pfn_t * arr, int num )`

4.131.4.165 `void* xc_map_foreign_range ( xc_interface * xch, uint32_t dom, int size, int prot, unsigned long mfn )`

Memory maps a range within one domain to a local address range. Mappings should be unmapped with `munmap` and should follow the same rules as `mmap` regarding page alignment. Returns NULL on failure.

`xch` a handle on an open hypervisor interface `dom` the domain to map memory from `size` the amount of memory to map (in multiples of page size) `prot` same flag as in `mmap()`. `mfn` the frame address to map.

4.131.4.166 `long xc_maximum_ram_page ( xc_interface * xch )`

4.131.4.167 `int xc_mem_access_resume ( xc_interface * xch, domid_t domain_id, unsigned long gfn )`

4.131.4.168 `int xc_mem_event_control ( xc_interface * xch, domid_t domain_id, unsigned int op, unsigned int mode, void * shared_page, void * ring_page, unsigned long gfn )`

mem\_event operations

4.131.4.169 `int xc_mem_event_disable ( xc_interface * xch, domid_t domain_id )`

4.131.4.170 `int xc_mem_event_enable ( xc_interface * xch, domid_t domain_id, void * shared_page, void * ring_page )`

4.131.4.171 `int xc_mem_paging_evict ( xc_interface * xch, domid_t domain_id, unsigned long gfn )`

4.131.4.172 `int xc_mem_paging_nominate ( xc_interface * xch, domid_t domain_id, unsigned long gfn )`

4.131.4.173 `int xc_mem_paging_prep ( xc_interface * xch, domid_t domain_id, unsigned long gfn )`

4.131.4.174 `int xc_mem_paging_resume ( xc_interface * xch, domid_t domain_id, unsigned long gfn )`

4.131.4.175 `int xc_memshr_control ( xc_interface * xch, uint32_t domid, int enable )`

memshr operations

4.131.4.176 `int xc_memshr_debug_gfn ( xc_interface * xch, uint32_t domid, unsigned long gfn )`

4.131.4.177 `int xc_memshr_debug_gref ( xc_interface * xch, uint32_t domid, grant_ref_t gref )`

4.131.4.178 `int xc_memshr_debug_mfn ( xc_interface * xch, uint32_t domid, unsigned long mfn )`

- 4.131.4.179 `int xc_memshr_domain_resume ( xc_interface * xch, uint32_t domid )`
- 4.131.4.180 `int xc_memshr_nominate_gfn ( xc_interface * xch, uint32_t domid, unsigned long gfn, uint64_t * handle )`
- 4.131.4.181 `int xc_memshr_nominate_gref ( xc_interface * xch, uint32_t domid, grant_ref_t gref, uint64_t * handle )`
- 4.131.4.182 `int xc_memshr_share ( xc_interface * xch, uint64_t source_handle, uint64_t client_handle )`
- 4.131.4.183 `int xc_mmuext_op ( xc_interface * xch, struct mmuext_op * op, unsigned int nr_ops, domid_t dom )`
- 4.131.4.184 `int xc_numainfo ( xc_interface * xch, xc_numainfo_t * info )`
- 4.131.4.185 `int xc_perfc_query ( xc_interface * xch, xc_hypercall_buffer_t * desc, xc_hypercall_buffer_t * val )`
- 4.131.4.186 `int xc_perfc_query_number ( xc_interface * xch, int * nbr_desc, int * nbr_val )`
- 4.131.4.187 `int xc_perfc_reset ( xc_interface * xch )`
- 4.131.4.188 `int xc_physdev_map_pirq ( xc_interface * xch, int domid, int index, int * pirq )`
- 4.131.4.189 `int xc_physdev_map_pirq_msi ( xc_interface * xch, int domid, int index, int * pirq, int devfn, int bus, int entry_nr, uint64_t table_base )`
- 4.131.4.190 `int xc_physdev_pci_access_modify ( xc_interface * xch, uint32_t domid, int bus, int dev, int func, int enable )`
- 4.131.4.191 `int xc_physdev_unmap_pirq ( xc_interface * xch, int domid, int pirq )`
- 4.131.4.192 `int xc_physinfo ( xc_interface * xch, xc_physinfo_t * info )`
- 4.131.4.193 `int xc_pm_get_cxstat ( xc_interface * xch, int cpuid, struct xc_cx_stat * cxpt )`
- 4.131.4.194 `int xc_pm_get_max_cx ( xc_interface * xch, int cpuid, int * max_cx )`
- 4.131.4.195 `int xc_pm_get_max_px ( xc_interface * xch, int cpuid, int * max_px )`
- 4.131.4.196 `int xc_pm_get_pxstat ( xc_interface * xch, int cpuid, struct xc_px_stat * pxpt )`
- 4.131.4.197 `int xc_pm_reset_cxstat ( xc_interface * xch, int cpuid )`
- 4.131.4.198 `int xc_pm_reset_pxstat ( xc_interface * xch, int cpuid )`
- 4.131.4.199 `int xc_readconsolering ( xc_interface * xch, char * buffer, unsigned int * pnr_chars, int clear, int incremental, uint32_t * pindex )`
- 4.131.4.200 `int xc_sched_arinc653_schedule_get ( xc_interface * xch, struct xen_sysctl_arinc653_schedule * schedule )`
- 4.131.4.201 `int xc_sched_arinc653_schedule_set ( xc_interface * xch, struct xen_sysctl_arinc653_schedule * schedule )`
- 4.131.4.202 `int xc_sched_credit2_domain_get ( xc_interface * xch, uint32_t domid, struct xen_domctl_sched_credit2 * sdom )`
- 4.131.4.203 `int xc_sched_credit2_domain_set ( xc_interface * xch, uint32_t domid, struct xen_domctl_sched_credit2 * sdom )`
- 4.131.4.204 `int xc_sched_credit_domain_get ( xc_interface * xch, uint32_t domid, struct xen_domctl_sched_credit * sdom )`

- 4.131.4.205 `int xc_sched_credit_domain_set ( xc_interface * xch, uint32_t domid, struct xen_domctl_sched_credit * sdom )`
- 4.131.4.206 `int xc_sched_id ( xc_interface * xch, int * sched_id )`
- 4.131.4.207 `int xc_sedf_domain_get ( xc_interface * xch, uint32_t domid, uint64_t * period, uint64_t * slice, uint64_t * latency, uint16_t * extratime, uint16_t * weight )`
- 4.131.4.208 `int xc_sedf_domain_set ( xc_interface * xch, uint32_t domid, uint64_t period, uint64_t slice, uint64_t latency, uint16_t extratime, uint16_t weight )`
- 4.131.4.209 `int xc_send_debug_keys ( xc_interface * xch, char * keys )`
- 4.131.4.210 `int xc_set_cpufreq_gov ( xc_interface * xch, int cpuid, char * govname )`
- 4.131.4.211 `int xc_set_cpufreq_para ( xc_interface * xch, int cpuid, int ctrl_type, int ctrl_value )`
- 4.131.4.212 `int xc_set_cpuidle_max_cstate ( xc_interface * xch, uint32_t value )`
- 4.131.4.213 `int xc_set_hvm_param ( xc_interface * handle, domid_t dom, int param, unsigned long value )`
- 4.131.4.214 `int xc_set_sched_opt_smt ( xc_interface * xch, uint32_t value )`
- 4.131.4.215 `int xc_set_vcpu_migration_delay ( xc_interface * xch, uint32_t value )`
- 4.131.4.216 `int xc_shadow_control ( xc_interface * xch, uint32_t domid, unsigned int sop, xc_hypcall_buffer_t * dirty_bitmap, unsigned long pages, unsigned long * mb, uint32_t mode, xc_shadow_op_stats_t * stats )`
- 4.131.4.217 `const char* xc_strerror ( xc_interface * xch, int errcode )`
- 4.131.4.218 `int xc_sysctl ( xc_interface * xch, struct xen_sysctl * sysctl )`
- 4.131.4.219 `int xc_tbuf_disable ( xc_interface * xch )`
- 4.131.4.220 `int xc_tbuf_enable ( xc_interface * xch, unsigned long pages, unsigned long * mfn, unsigned long * size )`

`xc_tbuf_enable` - enable tracing buffers

`xch` a handle to an open hypervisor interface `cnt` size of tracing buffers to create (in pages) `mfn` location to store mfn of the trace buffers to size location to store the size (in bytes) of a trace buffer to

Gets the machine address of the trace pointer area and the size of the per CPU buffers.

- 4.131.4.221 `int xc_tbuf_get_size ( xc_interface * xch, unsigned long * size )`

This function retrieves the current size of the trace buffers. Note that the size returned is in terms of bytes, not pages.

`xch` a handle to an open hypervisor interface `size` will contain the size in bytes for the trace buffers

#### Returns

0 on success, -1 on failure.

- 4.131.4.222 `int xc_tbuf_set_cpu_mask ( xc_interface * xch, uint32_t mask )`

- 4.131.4.223 `int xc_tbuf_set_evt_mask ( xc_interface * xch, uint32_t mask )`

4.131.4.224 `int xc_tbuf_set_size ( xc_interface * xch, unsigned long size )`

This function sets the size of the trace buffers. Setting the size is currently a one-shot operation that may be performed either at boot time or via this interface, not both. The buffer size must be set before enabling tracing.

xch a handle to an open hypervisor interface size the size in pages per cpu for the trace buffers

#### Returns

0 on success, -1 on failure.

4.131.4.225 `int xc_test_assign_device ( xc_interface * xch, uint32_t domid, uint32_t machine_bdf )`

4.131.4.226 `int xc_tmem_auth ( xc_interface * xch, int cli_id, char * uuid_str, int arg1 )`

4.131.4.227 `int xc_tmem_control ( xc_interface * xch, int32_t pool_id, uint32_t subop, uint32_t cli_id, uint32_t arg1, uint32_t arg2, uint64_t arg3, void * buf )`

4.131.4.228 `int xc_tmem_control_oid ( xc_interface * xch, int32_t pool_id, uint32_t subop, uint32_t cli_id, uint32_t arg1, uint32_t arg2, struct tmem_oid oid, void * buf )`

4.131.4.229 `int xc_tmem_restore ( xc_interface * xch, int dom, int fd )`

4.131.4.230 `int xc_tmem_restore_extra ( xc_interface * xch, int dom, int fd )`

4.131.4.231 `int xc_tmem_save ( xc_interface * xch, int dom, int live, int fd, int field_marker )`

4.131.4.232 `void xc_tmem_save_done ( xc_interface * xch, int dom )`

4.131.4.233 `int xc_tmem_save_extra ( xc_interface * xch, int dom, int fd, int field_marker )`

4.131.4.234 `int xc_topologyinfo ( xc_interface * xch, xc_topologyinfo_t * info )`

4.131.4.235 `unsigned long xc_translate_foreign_address ( xc_interface * xch, uint32_t dom, int vcpu, unsigned long long virt )`

Translates a virtual address in the context of a given domain and vcpu returning the GFN containing the address (that is, an MFN for PV guests, a PFN for HVM guests). Returns 0 for failure.

xch a handle on an open hypervisor interface dom the domain to perform the translation in vcpu the vcpu to perform the translation on virt the virtual address to translate

4.131.4.236 `int xc_vcpu_getaffinity ( xc_interface * xch, uint32_t domid, int vcpu, xc_cpumap_t cpumap )`

4.131.4.237 `int xc_vcpu_getcontext ( xc_interface * xch, uint32_t domid, uint32_t vcpu, vcpu_guest_context_any_t * ctxt )`

This function returns information about the execution context of a particular vcpu of a domain.

xch a handle to an open hypervisor interface domid the domain to get information from vcpu the vcpu number ctxt a pointer to a structure to store the execution context of the domain

#### Returns

0 on success, -1 on failure

4.131.4.238 `int xc_vcpu_getinfo ( xc_interface * xch, uint32_t domid, uint32_t vcpu, xc_vcpuinfo_t * info )`

4.131.4.239 `int xc_vcpu_setaffinity ( xc_interface * xch, uint32_t domid, int vcpu, xc_cpumap_t cpumap )`

4.131.4.240 `int xc_vcpu_setcontext ( xc_interface * xch, uint32_t domid, uint32_t vcpu, vcpu_guest_context_any_t * ctxt )`

This function will set the execution context for the specified vcpu.

xch a handle to an open hypervisor interface domid the domain to set the vcpu context for vcpu the vcpu number for the context ctxt pointer to the the cpu context with the values to set

#### Returns

the number of domains enumerated or -1 on error

4.131.4.241 `int xc_version ( xc_interface * xch, int cmd, void * arg )`

4.131.4.242 `int xc_watchdog ( xc_interface * xch, uint32_t id, uint32_t timeout )`

## 4.132 xen/tools/libxl/libxl.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <sys/select.h>
#include <sys/wait.h>
#include <sys/time.h>
#include <signal.h>
#include <unistd.h>
#include <stdint.h>
#include <inttypes.h>
#include <assert.h>
#include "libxl.h"
#include "libxl_utils.h"
#include "libxl_internal.h"
#include "flexarray.h"
```

#### Macros

- `#define PAGE_TO_MEMKB(pages) ((pages) * 4)`
- `#define BACKEND_STRING_SIZE 5`
- `#define STRINGIFY(x) #x`
- `#define TOSTRING(x) STRINGIFY(x)`

#### Functions

- `int libxl_ctx_init (libxl_ctx *ctx, int version, xentoollog_logger *lg)`
- `int libxl_ctx_free (libxl_ctx *ctx)`
- `void libxl_string_list_destroy (libxl_string_list *psl)`

- void [libxl\\_key\\_value\\_list\\_destroy](#) ([libxl\\_key\\_value\\_list](#) \*pkvl)
- int [libxl\\_domain\\_rename](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, const char \*old\_name, const char \*new\_name, xs\_transaction\_t trans)
- int [libxl\\_domain\\_resume](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_domain\\_preserve](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_create\\_info](#) \*info, const char \*name\_suffix, libxl\_uuid new\_uuid)
- [libxl\\_dominfo](#) \* [libxl\\_list\\_domain](#) ([libxl\\_ctx](#) \*ctx, int \*nb\_domain)
- int [libxl\\_domain\\_info](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_dominfo](#) \*info\_r, uint32\_t domid)
- [libxl\\_cpupoolinfo](#) \* [libxl\\_list\\_cpupool](#) ([libxl\\_ctx](#) \*ctx, int \*nb\_pool)
- [libxl\\_vminfo](#) \* [libxl\\_list\\_vm](#) ([libxl\\_ctx](#) \*ctx, int \*nb\_vm)
- int [libxl\\_domain\\_suspend](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_domain\\_suspend\\_info](#) \*info, uint32\_t domid, int fd)
- int [libxl\\_domain\\_pause](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_domain\\_core\\_dump](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, const char \*filename)
- int [libxl\\_domain\\_unpause](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_domain\\_shutdown](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int req)
- int [libxl\\_get\\_wait\\_fd](#) ([libxl\\_ctx](#) \*ctx, int \*fd)
- int [libxl\\_wait\\_for\\_domain\\_death](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_waiter](#) \*waiter)
- int [libxl\\_wait\\_for\\_disk\\_ejects](#) ([libxl\\_ctx](#) \*ctx, uint32\_t guest\_domid, [libxl\\_device\\_disk](#) \*disks, int num\_disks, [libxl\\_waiter](#) \*waiter)
- int [libxl\\_get\\_event](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_event](#) \*event)
- int [libxl\\_stop\\_waiting](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_waiter](#) \*waiter)
- int [libxl\\_free\\_event](#) ([libxl\\_event](#) \*event)
- int [libxl\\_free\\_waiter](#) ([libxl\\_waiter](#) \*waiter)
- int [libxl\\_event\\_get\\_domain\\_death\\_info](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_event](#) \*event, [libxl\\_dominfo](#) \*info)
- int [libxl\\_event\\_get\\_disk\\_eject\\_info](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_event](#) \*event, [libxl\\_device\\_disk](#) \*disk)
- int [libxl\\_domain\\_destroy](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int force)
- int [libxl\\_console\\_exec](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int cons\_num, [libxl\\_console\\_constype](#) type)
- int [libxl\\_primary\\_console\\_exec](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid\_vm)
- int [libxl\\_vncviewer\\_exec](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int autopass)
- int [libxl\\_device\\_disk\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_disk](#) \*disk)
- int [libxl\\_device\\_disk\\_del](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_device\\_disk](#) \*disk, int wait)
- char \* [libxl\\_device\\_disk\\_local\\_attach](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_device\\_disk](#) \*disk)
- int [libxl\\_device\\_disk\\_local\\_detach](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_device\\_disk](#) \*disk)
- int [libxl\\_device\\_nic\\_init](#) ([libxl\\_device\\_nic](#) \*nic\_info, int devnum)
- int [libxl\\_device\\_nic\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_nic](#) \*nic)
- int [libxl\\_device\\_nic\\_del](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_device\\_nic](#) \*nic, int wait)
- [libxl\\_nicinfo](#) \* [libxl\\_list\\_nics](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, unsigned int \*nb)
- void [libxl\\_device\\_net2\\_init](#) ([libxl\\_device\\_net2](#) \*net2\_info, int devnum)
- int [libxl\\_device\\_net2\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_net2](#) \*net2)
- [libxl\\_net2info](#) \* [libxl\\_device\\_net2\\_list](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, unsigned int \*nb)
- int [libxl\\_device\\_net2\\_del](#) ([libxl\\_ctx](#) \*ctx, [libxl\\_device\\_net2](#) \*net2, int wait)
- int [libxl\\_device\\_console\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_console](#) \*console)
- void [libxl\\_device\\_vkb\\_init](#) ([libxl\\_device\\_vkb](#) \*vkb, int dev\_num)
- int [libxl\\_device\\_vkb\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_vkb](#) \*vkb)
- int [libxl\\_device\\_vkb\\_clean\\_shutdown](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_device\\_vkb\\_hard\\_shutdown](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- [libxl\\_device\\_disk](#) \* [libxl\\_device\\_disk\\_list](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int \*num)
- int [libxl\\_device\\_disk\\_getinfo](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_disk](#) \*disk, [libxl\\_diskinfo](#) \*diskinfo)
- int [libxl\\_cdrom\\_insert](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_disk](#) \*disk)
- void [libxl\\_device\\_vfb\\_init](#) ([libxl\\_device\\_vfb](#) \*vfb, int dev\_num)
- int [libxl\\_device\\_vfb\\_add](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_device\\_vfb](#) \*vfb)
- int [libxl\\_device\\_vfb\\_clean\\_shutdown](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_device\\_vfb\\_hard\\_shutdown](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_domain\\_setmaxmem](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, uint32\_t max\_memkb)
- int [libxl\\_set\\_memory\\_target](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int32\_t target\_memkb, int relative, int enforce)

- `int libxl_get_memory_target (libxl_ctx *ctx, uint32_t domid, uint32_t *out_target)`
- `int libxl_domain_need_memory (libxl_ctx *ctx, libxl_domain_build_info *b_info, libxl_device_model_info *dm_info, int domid, uint32_t *need_memkb)`
- `int libxl_get_free_memory (libxl_ctx *ctx, uint32_t *memkb)`
- `int libxl_wait_for_free_memory (libxl_ctx *ctx, uint32_t domid, uint32_t memory_kb, int wait_secs)`
- `int libxl_wait_for_memory_target (libxl_ctx *ctx, uint32_t domid, int wait_secs)`
- `int libxl_button_press (libxl_ctx *ctx, uint32_t domid, libxl_button button)`
- `int libxl_get_physinfo (libxl_ctx *ctx, libxl_physinfo *physinfo)`
- `int libxl_get_topologyinfo (libxl_ctx *ctx, libxl_topologyinfo *info)`
- `const libxl_version_info * libxl_get_version_info (libxl_ctx *ctx)`
- `libxl_vcpuinfo * libxl_list_vcpu (libxl_ctx *ctx, uint32_t domid, int *nb_vcpu, int *nr_cpus)`
- `int libxl_set_vcpuaffinity (libxl_ctx *ctx, uint32_t domid, uint32_t vcpuid, libxl_cpumap *cpumap)`
- `int libxl_set_vcpuonline (libxl_ctx *ctx, uint32_t domid, libxl_cpumap *cpumap)`
- `int libxl_get_sched_id (libxl_ctx *ctx)`
- `int libxl_sched_credit_domain_get (libxl_ctx *ctx, uint32_t domid, libxl_sched_credit *scinfo)`
- `int libxl_sched_credit_domain_set (libxl_ctx *ctx, uint32_t domid, libxl_sched_credit *scinfo)`
- `int libxl_send_trigger (libxl_ctx *ctx, uint32_t domid, char *trigger_name, uint32_t vcpuid)`
- `int libxl_send_sysrq (libxl_ctx *ctx, uint32_t domid, char sysrq)`
- `int libxl_send_debug_keys (libxl_ctx *ctx, char *keys)`
- `libxl_xen_console_reader * libxl_xen_console_read_start (libxl_ctx *ctx, int clear)`
- `int libxl_xen_console_read_line (libxl_ctx *ctx, libxl_xen_console_reader *cr, char **line_r)`
- `void libxl_xen_console_read_finish (libxl_ctx *ctx, libxl_xen_console_reader *cr)`
- `uint32_t libxl_vm_get_start_time (libxl_ctx *ctx, uint32_t domid)`
- `char * libxl_tmem_list (libxl_ctx *ctx, uint32_t domid, int use_long)`
- `int libxl_tmem_freeze (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_destroy (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_thaw (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_set (libxl_ctx *ctx, uint32_t domid, char *name, uint32_t set)`
- `int libxl_tmem_shared_auth (libxl_ctx *ctx, uint32_t domid, char *uuid, int auth)`
- `int libxl_tmem_freeable (libxl_ctx *ctx)`
- `void libxl_file_reference_destroy (libxl_file_reference *f)`
- `int libxl_get_freecpus (libxl_ctx *ctx, libxl_cpumap *cpumap)`
- `int libxl_create_cpupool (libxl_ctx *ctx, const char *name, int schedid, libxl_cpumap cpumap, libxl_uuid *uuid, uint32_t *poolid)`
- `int libxl_destroy_cpupool (libxl_ctx *ctx, uint32_t poolid)`
- `int libxl_cpupool_rename (libxl_ctx *ctx, const char *name, uint32_t poolid)`
- `int libxl_cpupool_cpuadd (libxl_ctx *ctx, uint32_t poolid, int cpu)`
- `int libxl_cpupool_cpuadd_node (libxl_ctx *ctx, uint32_t poolid, int node, int *cpus)`
- `int libxl_cpupool_cpuremove (libxl_ctx *ctx, uint32_t poolid, int cpu)`
- `int libxl_cpupool_cpuremove_node (libxl_ctx *ctx, uint32_t poolid, int node, int *cpus)`
- `int libxl_cpupool_movedomain (libxl_ctx *ctx, uint32_t poolid, uint32_t domid)`

## 4.132.1 Macro Definition Documentation

4.132.1.1 `#define BACKEND_STRING_SIZE 5`

4.132.1.2 `#define PAGE_TO_MEMKB( pages ) ((pages) * 4)`

4.132.1.3 `#define STRINGIFY( x ) #x`

4.132.1.4 `#define TOSTRING( x ) STRINGIFY(x)`

## 4.132.2 Function Documentation



- 4.132.2.1 int libxl\_button\_press ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_button *button* )
  - 4.132.2.2 int libxl\_cdrom\_insert ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_disk \* *disk* )
  - 4.132.2.3 int libxl\_console\_exec ( libxl\_ctx \* *ctx*, uint32\_t *domid*, int *cons\_num*, libxl\_console\_constype *type* )
  - 4.132.2.4 int libxl\_cpupool\_cpuadd ( libxl\_ctx \* *ctx*, uint32\_t *poolid*, int *cpu* )
  - 4.132.2.5 int libxl\_cpupool\_cpuadd\_node ( libxl\_ctx \* *ctx*, uint32\_t *poolid*, int *node*, int \* *cpus* )
  - 4.132.2.6 int libxl\_cpupool\_cpuremove ( libxl\_ctx \* *ctx*, uint32\_t *poolid*, int *cpu* )
  - 4.132.2.7 int libxl\_cpupool\_cpuremove\_node ( libxl\_ctx \* *ctx*, uint32\_t *poolid*, int *node*, int \* *cpus* )
  - 4.132.2.8 int libxl\_cpupool\_movedomain ( libxl\_ctx \* *ctx*, uint32\_t *poolid*, uint32\_t *domid* )
  - 4.132.2.9 int libxl\_cpupool\_rename ( libxl\_ctx \* *ctx*, const char \* *name*, uint32\_t *poolid* )
  - 4.132.2.10 int libxl\_create\_cpupool ( libxl\_ctx \* *ctx*, const char \* *name*, int *schedid*, libxl\_cpumap *cpumap*, libxl\_uuid \* *uuid*, uint32\_t \* *poolid* )
  - 4.132.2.11 int libxl\_ctx\_free ( libxl\_ctx \* *ctx* )
  - 4.132.2.12 int libxl\_ctx\_init ( libxl\_ctx \* *ctx*, int *version*, xentoollog\_logger \* *lg* )
  - 4.132.2.13 int libxl\_destroy\_cpupool ( libxl\_ctx \* *ctx*, uint32\_t *poolid* )
  - 4.132.2.14 int libxl\_device\_console\_add ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_console \* *console* )
  - 4.132.2.15 int libxl\_device\_disk\_add ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_disk \* *disk* )
  - 4.132.2.16 int libxl\_device\_disk\_del ( libxl\_ctx \* *ctx*, libxl\_device\_disk \* *disk*, int *wait* )
  - 4.132.2.17 int libxl\_device\_disk\_getinfo ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_disk \* *disk*, libxl\_diskinfo \* *diskinfo* )
  - 4.132.2.18 libxl\_device\_disk\* libxl\_device\_disk\_list ( libxl\_ctx \* *ctx*, uint32\_t *domid*, int \* *num* )
  - 4.132.2.19 char\* libxl\_device\_disk\_local\_attach ( libxl\_ctx \* *ctx*, libxl\_device\_disk \* *disk* )
- Make a disk available in this domain. Returns path to a device.
- 4.132.2.20 int libxl\_device\_disk\_local\_detach ( libxl\_ctx \* *ctx*, libxl\_device\_disk \* *disk* )
  - 4.132.2.21 int libxl\_device\_net2\_add ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_net2 \* *net2* )
  - 4.132.2.22 int libxl\_device\_net2\_del ( libxl\_ctx \* *ctx*, libxl\_device\_net2 \* *net2*, int *wait* )
  - 4.132.2.23 void libxl\_device\_net2\_init ( libxl\_device\_net2 \* *net2\_info*, int *devnum* )
  - 4.132.2.24 libxl\_net2info\* libxl\_device\_net2\_list ( libxl\_ctx \* *ctx*, uint32\_t *domid*, unsigned int \* *nb* )
  - 4.132.2.25 int libxl\_device\_nic\_add ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_device\_nic \* *nic* )
  - 4.132.2.26 int libxl\_device\_nic\_del ( libxl\_ctx \* *ctx*, libxl\_device\_nic \* *nic*, int *wait* )

- 4.132.2.27 `int libxl_device_nic_init ( libxl_device_nic * nic_info, int devnum )`
- 4.132.2.28 `int libxl_device_vfb_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_vfb * vfb )`
- 4.132.2.29 `int libxl_device_vfb_clean_shutdown ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.30 `int libxl_device_vfb_hard_shutdown ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.31 `void libxl_device_vfb_init ( libxl_device_vfb * vfb, int dev_num )`
- 4.132.2.32 `int libxl_device_vkb_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_vkb * vkb )`
- 4.132.2.33 `int libxl_device_vkb_clean_shutdown ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.34 `int libxl_device_vkb_hard_shutdown ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.35 `void libxl_device_vkb_init ( libxl_device_vkb * vkb, int dev_num )`
- 4.132.2.36 `int libxl_domain_core_dump ( libxl_ctx * ctx, uint32_t domid, const char * filename )`
- 4.132.2.37 `int libxl_domain_destroy ( libxl_ctx * ctx, uint32_t domid, int force )`
- 4.132.2.38 `int libxl_domain_info ( libxl_ctx * ctx, libxl_dominfo * info_r, uint32_t domid )`
- 4.132.2.39 `int libxl_domain_need_memory ( libxl_ctx * ctx, libxl_domain_build_info * b_info, libxl_device_model_info * dm_info, int domc, uint32_t * need_memkb )`

how much free memory in the system a domain needs to be built

- 4.132.2.40 `int libxl_domain_pause ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.41 `int libxl_domain_preserve ( libxl_ctx * ctx, uint32_t domid, libxl_domain_create_info * info, const char * name_suffix, libxl_uuid new_uuid )`

Preserves a domain but rewrites xenstore etc to make it unique so that the domain can be restarted.

Does not modify info so that it may be reused.

- 4.132.2.42 `int libxl_domain_rename ( libxl_ctx * ctx, uint32_t domid, const char * old_name, const char * new_name, xs_transaction_t trans )`
- 4.132.2.43 `int libxl_domain_resume ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.44 `int libxl_domain_setmaxmem ( libxl_ctx * ctx, uint32_t domid, uint32_t max_memkb )`
- 4.132.2.45 `int libxl_domain_shutdown ( libxl_ctx * ctx, uint32_t domid, int req )`
- 4.132.2.46 `int libxl_domain_suspend ( libxl_ctx * ctx, libxl_domain_suspend_info * info, uint32_t domid, int fd )`
- 4.132.2.47 `int libxl_domain_unpause ( libxl_ctx * ctx, uint32_t domid )`
- 4.132.2.48 `int libxl_event_get_disk_eject_info ( libxl_ctx * ctx, uint32_t domid, libxl_event * event, libxl_device_disk * disk )`

Returns true and fills \*disk if the caller should eject the disk

4.132.2.49 `int libxl_event_get_domain_death_info ( libxl_ctx * ctx, uint32_t domid, libxl_event * event, libxl_dominfo * info )`

Returns:

- 0 if the domain is dead but there is no cleanup to be done. e.g because someone else has already done it.
- 1 if the domain is dead and there is cleanup to be done.

Can return error if the domain exists and is still running.

\*info will contain valid domain state iff 1 is returned. In particular if 1 is returned then info->shutdown\_reason is guaranteed to be valid since by definition the domain is (shutdown||dying))

4.132.2.50 `void libxl_file_reference_destroy ( libxl_file_reference * f )`

4.132.2.51 `int libxl_free_event ( libxl_event * event )`

4.132.2.52 `int libxl_free_waiter ( libxl_waiter * waiter )`

4.132.2.53 `int libxl_get_event ( libxl_ctx * ctx, libxl_event * event )`

4.132.2.54 `int libxl_get_free_memory ( libxl_ctx * ctx, uint32_t * memkb )`

how much free memory is available in the system

4.132.2.55 `int libxl_get_freecpus ( libxl_ctx * ctx, libxl_cpumap * cpumap )`

4.132.2.56 `int libxl_get_memory_target ( libxl_ctx * ctx, uint32_t domid, uint32_t * out_target )`

4.132.2.57 `int libxl_get_physinfo ( libxl_ctx * ctx, libxl_physinfo * physinfo )`

4.132.2.58 `int libxl_get_sched_id ( libxl_ctx * ctx )`

returns one of the XEN\_SCHEDULER\_\* constants from [public/domctl.h](http://public.domctl.h)

4.132.2.59 `int libxl_get_topologyinfo ( libxl_ctx * ctx, libxl_topologyinfo * info )`

4.132.2.60 `const libxl_version_info* libxl_get_version_info ( libxl_ctx * ctx )`

4.132.2.61 `int libxl_get_wait_fd ( libxl_ctx * ctx, int * fd )`

4.132.2.62 `void libxl_key_value_list_destroy ( libxl_key_value_list * pkvl )`

4.132.2.63 `libxl_cpupoolinfo* libxl_list_cpupool ( libxl_ctx * ctx, int * nb_pool )`

4.132.2.64 `libxl_dominfo* libxl_list_domain ( libxl_ctx * ctx, int * nb_domain )`

4.132.2.65 `libxl_nicinfo* libxl_list_nics ( libxl_ctx * ctx, uint32_t domid, unsigned int * nb )`

4.132.2.66 `libxl_vcpuinfo* libxl_list_vcpu ( libxl_ctx * ctx, uint32_t domid, int * nb_vcpu, int * nrcpus )`

4.132.2.67 `libxl_vminfo* libxl_list_vm ( libxl_ctx * ctx, int * nb_vm )`

this API call only list VM running on this host. a VM can be an aggregate of multiple domains.

4.132.2.68 `int libxl_primary_console_exec ( libxl_ctx * ctx, uint32_t domid_vm )`

`libxl_primary_console_exec` finds the domid and console number corresponding to the primary console of the given vm, then calls `libxl_console_exec` with the right arguments (domid might be different if the guest is using stubdoms). This function can be called after creating the device model, in case of HVM guests, and before `libxl_run_bootloader` in case of PV guests using pygrub.

4.132.2.69 `int libxl_sched_credit_domain_get ( libxl_ctx * ctx, uint32_t domid, libxl_sched_credit * scinfo )`

4.132.2.70 `int libxl_sched_credit_domain_set ( libxl_ctx * ctx, uint32_t domid, libxl_sched_credit * scinfo )`

4.132.2.71 `int libxl_send_debug_keys ( libxl_ctx * ctx, char * keys )`

4.132.2.72 `int libxl_send_sysrq ( libxl_ctx * ctx, uint32_t domid, char sysrq )`

4.132.2.73 `int libxl_send_trigger ( libxl_ctx * ctx, uint32_t domid, char * trigger_name, uint32_t vcpuid )`

4.132.2.74 `int libxl_set_memory_target ( libxl_ctx * ctx, uint32_t domid, int32_t target_memkb, int relative, int enforce )`

4.132.2.75 `int libxl_set_vcpuaffinity ( libxl_ctx * ctx, uint32_t domid, uint32_t vcpuid, libxl_cpumap * cpumap )`

4.132.2.76 `int libxl_set_vcpuonline ( libxl_ctx * ctx, uint32_t domid, libxl_cpumap * cpumap )`

4.132.2.77 `int libxl_stop_waiting ( libxl_ctx * ctx, libxl_waiter * waiter )`

4.132.2.78 `void libxl_string_list_destroy ( libxl_string_list * psf )`

4.132.2.79 `int libxl_tmem_destroy ( libxl_ctx * ctx, uint32_t domid )`

4.132.2.80 `int libxl_tmem_freeable ( libxl_ctx * ctx )`

4.132.2.81 `int libxl_tmem_freeze ( libxl_ctx * ctx, uint32_t domid )`

4.132.2.82 `char* libxl_tmem_list ( libxl_ctx * ctx, uint32_t domid, int use_long )`

4.132.2.83 `int libxl_tmem_set ( libxl_ctx * ctx, uint32_t domid, char * name, uint32_t set )`

4.132.2.84 `int libxl_tmem_shared_auth ( libxl_ctx * ctx, uint32_t domid, char * uuid, int auth )`

4.132.2.85 `int libxl_tmem_thaw ( libxl_ctx * ctx, uint32_t domid )`

4.132.2.86 `uint32_t libxl_vm_get_start_time ( libxl_ctx * ctx, uint32_t domid )`

4.132.2.87 `int libxl_vncviewer_exec ( libxl_ctx * ctx, uint32_t domid, int autopass )`

4.132.2.88 `int libxl_wait_for_disk_ejects ( libxl_ctx * ctx, uint32_t guest_domid, libxl_device_disk * disks, int num_disks, libxl_waiter * waiter )`

4.132.2.89 `int libxl_wait_for_domain_death ( libxl_ctx * ctx, uint32_t domid, libxl_waiter * waiter )`

4.132.2.90 `int libxl_wait_for_free_memory ( libxl_ctx * ctx, uint32_t domid, uint32_t memory_kb, int wait_secs )`

wait for a given amount of memory to be free in the system

4.132.2.91 `int libxl_wait_for_memory_target ( libxl_ctx * ctx, uint32_t domid, int wait_secs )`

wait for the memory target of a domain to be reached

4.132.2.92 `void libxl_xen_console_read_finish ( libxl_ctx * ctx, libxl_xen_console_reader * cr )`

4.132.2.93 `int libxl_xen_console_read_line ( libxl_ctx * ctx, libxl_xen_console_reader * cr, char ** line_r )`

return values: *line\_r* 1 success, whole line obtained from buffer non-0 0 no more lines available right now 0 negative error code *ERROR\_0* On success \*line\_r is updated to point to a nul-terminated string which is valid until the next call on the same console reader. The libxl caller may overwrite parts of the string if it wishes.

4.132.2.94 `libxl_xen_console_reader* libxl_xen_console_read_start ( libxl_ctx * ctx, int clear )`

## 4.133 xen/tools/libxl/libxl.h File Reference

```
#include <stdint.h>
#include <stdarg.h>
#include <netinet/in.h>
#include <xenctrl.h>
#include <xs.h>
#include <sys/wait.h>
#include "libxl_uuid.h"
#include "_libxl_types.h"
```

### Data Structures

- struct [libxl\\_cpumap](#)
- struct [libxl\\_cpuarray](#)
- struct [libxl\\_file\\_reference](#)
- struct [libxl\\_ctx](#)
- struct [libxl\\_domain\\_suspend\\_info](#)
- struct [libxl\\_domain\\_config](#)
- struct [libxl\\_event](#)
- struct [libxl\\_waiter](#)

### Macros

- #define [LIBXL\\_CPUARRAY\\_INVALID\\_ENTRY](#) ~0
- #define [LIBXL\\_PCI\\_FUNC\\_ALL](#) (~0U)
- #define [XL\\_SUSPEND\\_DEBUG](#) 1
- #define [XL\\_SUSPEND\\_LIVE](#) 2
- #define [LIBXL\\_VERSION](#) 0

### Typedefs

- typedef uint8\_t [libxl\\_mac](#) [6]
- typedef char \*\* [libxl\\_string\\_list](#)
- typedef char \*\* [libxl\\_key\\_value\\_list](#)
- typedef uint32\_t [libxl\\_hwcap](#) [8]
- typedef struct [libxl\\_cpuid\\_policy](#) [libxl\\_cpuid\\_policy](#)

- typedef libxl\_cpuid\_policy \* libxl\_cpuid\_policy\_list
- typedef int(\* libxl\_console\_ready)(libxl\_ctx \*ctx, uint32\_t domid, void \*priv)
- typedef struct  
libxl\_\_xen\_console\_reader libxl\_xen\_console\_reader

## Enumerations

- enum libxl\_qemu\_machine\_type { XENFV = 1, XENPV }
- enum libxl\_console\_constype { LIBXL\_CONSTYPE\_SERIAL = 1, LIBXL\_CONSTYPE\_PV }
- enum libxl\_console\_consbk { LIBXL\_CONSBK\_XENCONSOLED, LIBXL\_CONSBK\_IOEMU }
- enum libxl\_disk\_format {  
DISK\_FORMAT\_UNKNOWN = 0, DISK\_FORMAT\_QCOW, DISK\_FORMAT\_QCOW2, DISK\_FORMAT\_VHD,  
DISK\_FORMAT\_RAW, DISK\_FORMAT\_EMPTY }
- enum libxl\_disk\_backend { DISK\_BACKEND\_UNKNOWN = 0, DISK\_BACKEND\_PHY, DISK\_BACKEND\_TAP, DISK\_BACKEND\_QDISK }
- enum libxl\_nic\_type { NICTYPE\_IOEMU = 1, NICTYPE\_VIF }
- enum {  
ERROR\_NONSPECIFIC = -1, ERROR\_VERSION = -2, ERROR\_FAIL = -3, ERROR\_NI = -4,  
ERROR\_NOMEM = -5, ERROR\_INVAL = -6, ERROR\_BADFAIL = -7, ERROR\_GUEST\_TIMEDOUT = -8 }
- enum libxl\_action\_on\_shutdown {  
LIBXL\_ACTION\_DESTROY, LIBXL\_ACTION\_RESTART, LIBXL\_ACTION\_RESTART\_RENAME, LIBXL\_ACTION\_PRESERVE,  
LIBXL\_ACTION\_COREDUMP\_DESTROY, LIBXL\_ACTION\_COREDUMP\_RESTART }
- enum libxl\_event\_type { LIBXL\_EVENT\_DOMAIN\_DEATH, LIBXL\_EVENT\_DISK\_EJECT }
- enum libxl\_button { POWER\_BUTTON, SLEEP\_BUTTON }

## Functions

- void libxl\_string\_list\_destroy (libxl\_string\_list \*sl)
- void libxl\_key\_value\_list\_destroy (libxl\_key\_value\_list \*kvl)
- void libxl\_cpumap\_destroy (libxl\_cpumap \*map)
- void libxl\_cpuarray\_destroy (libxl\_cpuarray \*array)
- void libxl\_file\_reference\_destroy (libxl\_file\_reference \*p)
- void libxl\_cpuid\_destroy (libxl\_cpuid\_policy\_list \*cpuid\_list)
- const libxl\_version\_info \* libxl\_get\_version\_info (libxl\_ctx \*ctx)
- int libxl\_ctx\_init (libxl\_ctx \*ctx, int version, xentoollog\_logger \*)
- int libxl\_ctx\_free (libxl\_ctx \*ctx)
- int libxl\_ctx\_set\_log (libxl\_ctx \*ctx, xentoollog\_logger \*)
- int libxl\_ctx\_postfork (libxl\_ctx \*ctx)
- void libxl\_init\_create\_info (libxl\_domain\_create\_info \*c\_info)
- void libxl\_init\_build\_info (libxl\_domain\_build\_info \*b\_info, libxl\_domain\_create\_info \*c\_info)
- void libxl\_init\_dm\_info (libxl\_device\_model\_info \*dm\_info, libxl\_domain\_create\_info \*c\_info, libxl\_domain\_build\_info \*b\_info)
- int libxl\_domain\_create\_new (libxl\_ctx \*ctx, libxl\_domain\_config \*d\_config, libxl\_console\_ready cb, void \*priv, uint32\_t \*domid)
- int libxl\_domain\_create\_restore (libxl\_ctx \*ctx, libxl\_domain\_config \*d\_config, libxl\_console\_ready cb, void \*priv, uint32\_t \*domid, int restore\_fd)
- void libxl\_domain\_config\_destroy (libxl\_domain\_config \*d\_config)
- int libxl\_domain\_suspend (libxl\_ctx \*ctx, libxl\_domain\_suspend\_info \*info, uint32\_t domid, int fd)
- int libxl\_domain\_resume (libxl\_ctx \*ctx, uint32\_t domid)
- int libxl\_domain\_shutdown (libxl\_ctx \*ctx, uint32\_t domid, int req)
- int libxl\_domain\_destroy (libxl\_ctx \*ctx, uint32\_t domid, int force)
- int libxl\_domain\_preserve (libxl\_ctx \*ctx, uint32\_t domid, libxl\_domain\_create\_info \*info, const char \*name\_suffix, libxl\_uuid new\_uuid)

- `int libxl_get_max_cpus (libxl_ctx *ctx)`
- `int libxl_run_bootloader (libxl_ctx *ctx, libxl_domain_build_info *info, libxl_device_disk *disk, uint32_t domid)`
- `int libxl_get_wait_fd (libxl_ctx *ctx, int *fd)`
- `int libxl_wait_for_domain_death (libxl_ctx *ctx, uint32_t domid, libxl_waiter *waiter)`
- `int libxl_wait_for_disk_ejects (libxl_ctx *ctx, uint32_t domid, libxl_device_disk *disks, int num_disks, libxl_waiter *waiter)`
- `int libxl_get_event (libxl_ctx *ctx, libxl_event *event)`
- `int libxl_stop_waiting (libxl_ctx *ctx, libxl_waiter *waiter)`
- `int libxl_free_event (libxl_event *event)`
- `int libxl_free_waiter (libxl_waiter *waiter)`
- `int libxl_event_get_domain_death_info (libxl_ctx *ctx, uint32_t domid, libxl_event *event, libxl_dominfo *info)`
- `int libxl_event_get_disk_eject_info (libxl_ctx *ctx, uint32_t domid, libxl_event *event, libxl_device_disk *disk)`
- `int libxl_domain_rename (libxl_ctx *ctx, uint32_t domid, const char *old_name, const char *new_name, xs_transaction_t trans)`
- `int libxl_domain_pause (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_domain_unpause (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_domain_core_dump (libxl_ctx *ctx, uint32_t domid, const char *filename)`
- `int libxl_domain_setmaxmem (libxl_ctx *ctx, uint32_t domid, uint32_t target_memkb)`
- `int libxl_set_memory_target (libxl_ctx *ctx, uint32_t domid, int32_t target_memkb, int relative, int enforce)`
- `int libxl_get_memory_target (libxl_ctx *ctx, uint32_t domid, uint32_t *out_target)`
- `int libxl_domain_need_memory (libxl_ctx *ctx, libxl_domain_build_info *b_info, libxl_device_model_info *dm_info, int domc, uint32_t *need_memkb)`
- `int libxl_get_free_memory (libxl_ctx *ctx, uint32_t *memkb)`
- `int libxl_wait_for_free_memory (libxl_ctx *ctx, uint32_t domid, uint32_t memory_kb, int wait_secs)`
- `int libxl_wait_for_memory_target (libxl_ctx *ctx, uint32_t domid, int wait_secs)`
- `int libxl_vncviewer_exec (libxl_ctx *ctx, uint32_t domid, int autopass)`
- `int libxl_console_exec (libxl_ctx *ctx, uint32_t domid, int cons_num, libxl_console_constype type)`
- `int libxl_primary_console_exec (libxl_ctx *ctx, uint32_t domid_vm)`
- `int libxl_domain_info (libxl_ctx *, libxl_dominfo *info_r, uint32_t domid)`
- `libxl_dominfo * libxl_list_domain (libxl_ctx *, int *nb_domain)`
- `libxl_cpupoolinfo * libxl_list_cpupool (libxl_ctx *, int *nb_pool)`
- `libxl_vminfo * libxl_list_vm (libxl_ctx *ctx, int *nb_vm)`
- `int libxl_device_disk_add (libxl_ctx *ctx, uint32_t domid, libxl_device_disk *disk)`
- `int libxl_device_disk_del (libxl_ctx *ctx, libxl_device_disk *disk, int wait)`
- `libxl_device_disk * libxl_device_disk_list (libxl_ctx *ctx, uint32_t domid, int *num)`
- `int libxl_device_disk_getinfo (libxl_ctx *ctx, uint32_t domid, libxl_device_disk *disk, libxl_diskinfo *diskinfo)`
- `int libxl_cdrom_insert (libxl_ctx *ctx, uint32_t domid, libxl_device_disk *disk)`
- `char * libxl_device_disk_local_attach (libxl_ctx *ctx, libxl_device_disk *disk)`
- `int libxl_device_disk_local_detach (libxl_ctx *ctx, libxl_device_disk *disk)`
- `int libxl_device_nic_init (libxl_device_nic *nic, int dev_num)`
- `int libxl_device_nic_add (libxl_ctx *ctx, uint32_t domid, libxl_device_nic *nic)`
- `int libxl_device_nic_del (libxl_ctx *ctx, libxl_device_nic *nic, int wait)`
- `libxl_nicinfo * libxl_list_nics (libxl_ctx *ctx, uint32_t domid, unsigned int *nb)`
- `int libxl_device_console_add (libxl_ctx *ctx, uint32_t domid, libxl_device_console *console)`
- `void libxl_device_vkb_init (libxl_device_vkb *vkb, int dev_num)`
- `int libxl_device_vkb_add (libxl_ctx *ctx, uint32_t domid, libxl_device_vkb *vkb)`
- `int libxl_device_vkb_clean_shutdown (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_device_vkb_hard_shutdown (libxl_ctx *ctx, uint32_t domid)`
- `void libxl_device_vfb_init (libxl_device_vfb *vfb, int dev_num)`
- `int libxl_device_vfb_add (libxl_ctx *ctx, uint32_t domid, libxl_device_vfb *vfb)`
- `int libxl_device_vfb_clean_shutdown (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_device_vfb_hard_shutdown (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_device_pci_add (libxl_ctx *ctx, uint32_t domid, libxl_device_pci *pcidev)`
- `int libxl_device_pci_remove (libxl_ctx *ctx, uint32_t domid, libxl_device_pci *pcidev, int force)`
- `int libxl_device_pci_shutdown (libxl_ctx *ctx, uint32_t domid)`

- `int libxl_device_pci_list_assigned (libxl_ctx *ctx, libxl_device_pci **list, uint32_t domid, int *num)`
- `int libxl_device_pci_list_assignable (libxl_ctx *ctx, libxl_device_pci **list, int *num)`
- `int libxl_device_pci_parse_bdf (libxl_ctx *ctx, libxl_device_pci *pcidev, const char *str)`
- `int libxl_cpuid_parse_config (libxl_cpuid_policy_list *cpuid, const char *str)`
- `int libxl_cpuid_parse_config_xend (libxl_cpuid_policy_list *cpuid, const char *str)`
- `void libxl_cpuid_apply_policy (libxl_ctx *ctx, uint32_t domid)`
- `void libxl_cpuid_set (libxl_ctx *ctx, uint32_t domid, libxl_cpuid_policy_list cpuid)`
- `int libxl_userdata_store (libxl_ctx *ctx, uint32_t domid, const char *userdata_userid, const uint8_t *data, int datalen)`
- `int libxl_userdata_retrieve (libxl_ctx *ctx, uint32_t domid, const char *userdata_userid, uint8_t **data_r, int *datalen_r)`
- `int libxl_button_press (libxl_ctx *ctx, uint32_t domid, libxl_button button)`
- `int libxl_get_physinfo (libxl_ctx *ctx, libxl_physinfo *physinfo)`
- `int libxl_get_topologyinfo (libxl_ctx *ctx, libxl_topologyinfo *info)`
- `libxl_vcpuinfo * libxl_list_vcpu (libxl_ctx *ctx, uint32_t domid, int *nb_vcpu, int *nrcpus)`
- `int libxl_set_vcpuaffinity (libxl_ctx *ctx, uint32_t domid, uint32_t vcpuid, libxl_cpumap *cpumap)`
- `int libxl_set_vcpuonline (libxl_ctx *ctx, uint32_t domid, libxl_cpumap *cpumap)`
- `int libxl_get_sched_id (libxl_ctx *ctx)`
- `int libxl_sched_credit_domain_get (libxl_ctx *ctx, uint32_t domid, libxl_sched_credit *scinfo)`
- `int libxl_sched_credit_domain_set (libxl_ctx *ctx, uint32_t domid, libxl_sched_credit *scinfo)`
- `int libxl_send_trigger (libxl_ctx *ctx, uint32_t domid, char *trigger_name, uint32_t vcpuid)`
- `int libxl_send_sysrq (libxl_ctx *ctx, uint32_t domid, char sysrq)`
- `int libxl_send_debug_keys (libxl_ctx *ctx, char *keys)`
- `libxl_xen_console_reader * libxl_xen_console_read_start (libxl_ctx *ctx, int clear)`
- `int libxl_xen_console_read_line (libxl_ctx *ctx, libxl_xen_console_reader *cr, char **line_r)`
- `void libxl_xen_console_read_finish (libxl_ctx *ctx, libxl_xen_console_reader *cr)`
- `uint32_t libxl_vm_get_start_time (libxl_ctx *ctx, uint32_t domid)`
- `char * libxl_tmem_list (libxl_ctx *ctx, uint32_t domid, int use_long)`
- `int libxl_tmem_freeze (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_destroy (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_thaw (libxl_ctx *ctx, uint32_t domid)`
- `int libxl_tmem_set (libxl_ctx *ctx, uint32_t domid, char *name, uint32_t set)`
- `int libxl_tmem_shared_auth (libxl_ctx *ctx, uint32_t domid, char *uuid, int auth)`
- `int libxl_tmem_freeable (libxl_ctx *ctx)`
- `void libxl_device_net2_init (libxl_device_net2 *net2, int dev_num)`
- `int libxl_device_net2_add (libxl_ctx *ctx, uint32_t domid, libxl_device_net2 *net2)`
- `libxl_net2info * libxl_device_net2_list (libxl_ctx *ctx, uint32_t domid, unsigned int *nb)`
- `int libxl_device_net2_del (libxl_ctx *ctx, libxl_device_net2 *net2, int wait)`
- `int libxl_get_freecpus (libxl_ctx *ctx, libxl_cpumap *cpumap)`
- `int libxl_create_cpupool (libxl_ctx *ctx, const char *name, int schedid, libxl_cpumap cpumap, libxl_uuid *uuid, uint32_t *poolid)`
- `int libxl_destroy_cpupool (libxl_ctx *ctx, uint32_t poolid)`
- `int libxl_cpupool_rename (libxl_ctx *ctx, const char *name, uint32_t poolid)`
- `int libxl_cpupool_cpuadd (libxl_ctx *ctx, uint32_t poolid, int cpu)`
- `int libxl_cpupool_cpuadd_node (libxl_ctx *ctx, uint32_t poolid, int node, int *cpus)`
- `int libxl_cpupool_cpuremove (libxl_ctx *ctx, uint32_t poolid, int cpu)`
- `int libxl_cpupool_cpuremove_node (libxl_ctx *ctx, uint32_t poolid, int node, int *cpus)`
- `int libxl_cpupool_movedomain (libxl_ctx *ctx, uint32_t poolid, uint32_t domid)`
- `int init_console_info (libxl_device_console *console, int dev_num, libxl_domain_build_state *state)`
- `const char * libxl_sbindir_path (void)`
- `const char * libxl_bindir_path (void)`
- `const char * libxl_libexec_path (void)`
- `const char * libxl_libdir_path (void)`
- `const char * libxl_sharedir_path (void)`
- `const char * libxl_private_bindir_path (void)`



- const char \* [libxl\\_xenfirmware\\_dir\\_path](#) (void)
- const char \* [libxl\\_xen\\_config\\_dir\\_path](#) (void)
- const char \* [libxl\\_xen\\_script\\_dir\\_path](#) (void)
- const char \* [libxl\\_lock\\_dir\\_path](#) (void)

#### 4.133.1 Macro Definition Documentation

4.133.1.1 `#define LIBXL_CPUARRAY_INVALID_ENTRY ~0`

4.133.1.2 `#define LIBXL_PCI_FUNC_ALL (~0U)`

4.133.1.3 `#define LIBXL_VERSION 0`

4.133.1.4 `#define XL_SUSPEND_DEBUG 1`

4.133.1.5 `#define XL_SUSPEND_LIVE 2`

#### 4.133.2 Typedef Documentation

4.133.2.1 `typedef int(* libxl_console_ready)(libxl_ctx *ctx, uint32_t domid, void *priv)`

4.133.2.2 `typedef struct libxl__cpuid_policy libxl_cpuid_policy`

4.133.2.3 `typedef libxl_cpuid_policy* libxl_cpuid_policy_list`

4.133.2.4 `typedef uint32_t libxl_hwcap[8]`

4.133.2.5 `typedef char** libxl_key_value_list`

4.133.2.6 `typedef uint8_t libxl_mac[6]`

4.133.2.7 `typedef char** libxl_string_list`

4.133.2.8 `typedef struct libxl__xen_console_reader libxl_xen_console_reader`

#### 4.133.3 Enumeration Type Documentation

4.133.3.1 anonymous enum

Enumerator

***ERROR\_NONSPECIFIC***

***ERROR\_VERSION***

***ERROR\_FAIL***

***ERROR\_NI***

***ERROR\_NOMEM***

***ERROR\_INVAL***

***ERROR\_BADFAIL***

***ERROR\_GUEST\_TIMEOUT***

## 4.133.3.2 enum libxl\_action\_on\_shutdown

Enumerator

***LIBXL\_ACTION\_DESTROY***  
***LIBXL\_ACTION\_RESTART***  
***LIBXL\_ACTION\_RESTART\_RENAME***  
***LIBXL\_ACTION\_PRESERVE***  
***LIBXL\_ACTION\_COREDUMP\_DESTROY***  
***LIBXL\_ACTION\_COREDUMP\_RESTART***

## 4.133.3.3 enum libxl\_button

Enumerator

***POWER\_BUTTON***  
***SLEEP\_BUTTON***

## 4.133.3.4 enum libxl\_console\_consback

Enumerator

***LIBXL\_CONSBACK\_XENCONSOLED***  
***LIBXL\_CONSBACK\_IOEMU***

## 4.133.3.5 enum libxl\_console\_constype

Enumerator

***LIBXL\_CONSTYPE\_SERIAL***  
***LIBXL\_CONSTYPE\_PV***

## 4.133.3.6 enum libxl\_disk\_backend

Enumerator

***DISK\_BACKEND\_UNKNOWN***  
***DISK\_BACKEND\_PHY***  
***DISK\_BACKEND\_TAP***  
***DISK\_BACKEND\_QDISK***

## 4.133.3.7 enum libxl\_disk\_format

Enumerator

***DISK\_FORMAT\_UNKNOWN***  
***DISK\_FORMAT\_QCOW***  
***DISK\_FORMAT\_QCOW2***  
***DISK\_FORMAT\_VHD***  
***DISK\_FORMAT\_RAW***  
***DISK\_FORMAT\_EMPTY***

## 4.133.3.8 enum libxl\_event\_type

Enumerator

***LIBXL\_EVENT\_DOMAIN\_DEATH***  
***LIBXL\_EVENT\_DISK\_EJECT***

## 4.133.3.9 enum libxl\_nic\_type

Enumerator

***NICTYPE\_IOEMU***  
***NICTYPE\_VIF***

## 4.133.3.10 enum libxl\_qemu\_machine\_type

Enumerator

***XENFV***  
***XENPV***

## 4.133.4 Function Documentation

4.133.4.1 int init\_console\_info ( libxl\_device\_console \* console, int dev\_num, libxl\_domain\_build\_state \* state )

4.133.4.2 const char\* libxl\_bindir\_path ( void )

4.133.4.3 int libxl\_button\_press ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_button button )

4.133.4.4 int libxl\_cdrom\_insert ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_device\_disk \* disk )

4.133.4.5 int libxl\_console\_exec ( libxl\_ctx \* ctx, uint32\_t domid, int cons\_num, libxl\_console\_constype type )

4.133.4.6 void libxl\_cpumap\_destroy ( libxl\_cpumap \* array )

4.133.4.7 void libxl\_cpuid\_apply\_policy ( libxl\_ctx \* ctx, uint32\_t domid )

4.133.4.8 void libxl\_cpuid\_destroy ( libxl\_cpuid\_policy\_list \* cpuid\_list )

4.133.4.9 int libxl\_cpuid\_parse\_config ( libxl\_cpuid\_policy\_list \* cpuid, const char \* str )

4.133.4.10 int libxl\_cpuid\_parse\_config\_xend ( libxl\_cpuid\_policy\_list \* cpuid, const char \* str )

4.133.4.11 void libxl\_cpuid\_set ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_cpuid\_policy\_list cpuid )

4.133.4.12 void libxl\_cpumap\_destroy ( libxl\_cpumap \* map )

4.133.4.13 int libxl\_cpupool\_cpuadd ( libxl\_ctx \* ctx, uint32\_t poolid, int cpu )

4.133.4.14 int libxl\_cpupool\_cpuadd\_node ( libxl\_ctx \* ctx, uint32\_t poolid, int node, int \* cpus )

4.133.4.15 int libxl\_cpupool\_cpuremove ( libxl\_ctx \* ctx, uint32\_t poolid, int cpu )

4.133.4.16 int libxl\_cpupool\_cpuremove\_node ( libxl\_ctx \* ctx, uint32\_t poolid, int node, int \* cpus )

- 4.133.4.17 `int libxl_cpupool_movedomain ( libxl_ctx * ctx, uint32_t poolid, uint32_t domid )`
  - 4.133.4.18 `int libxl_cpupool_rename ( libxl_ctx * ctx, const char * name, uint32_t poolid )`
  - 4.133.4.19 `int libxl_create_cpupool ( libxl_ctx * ctx, const char * name, int schedid, libxl_cpumap cpumap, libxl_uuid * uuid, uint32_t * poolid )`
  - 4.133.4.20 `int libxl_ctx_free ( libxl_ctx * ctx )`
  - 4.133.4.21 `int libxl_ctx_init ( libxl_ctx * ctx, int version, xentoollog_logger * )`
  - 4.133.4.22 `int libxl_ctx_postfork ( libxl_ctx * ctx )`
  - 4.133.4.23 `int libxl_ctx_set_log ( libxl_ctx * ctx, xentoollog_logger * )`
  - 4.133.4.24 `int libxl_destroy_cpupool ( libxl_ctx * ctx, uint32_t poolid )`
  - 4.133.4.25 `int libxl_device_console_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_console * console )`
  - 4.133.4.26 `int libxl_device_disk_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_disk * disk )`
  - 4.133.4.27 `int libxl_device_disk_del ( libxl_ctx * ctx, libxl_device_disk * disk, int wait )`
  - 4.133.4.28 `int libxl_device_disk_getinfo ( libxl_ctx * ctx, uint32_t domid, libxl_device_disk * disk, libxl_diskinfo * diskinfo )`
  - 4.133.4.29 `libxl_device_disk* libxl_device_disk_list ( libxl_ctx * ctx, uint32_t domid, int * num )`
  - 4.133.4.30 `char* libxl_device_disk_local_attach ( libxl_ctx * ctx, libxl_device_disk * disk )`
- Make a disk available in this domain. Returns path to a device.
- 4.133.4.31 `int libxl_device_disk_local_detach ( libxl_ctx * ctx, libxl_device_disk * disk )`
  - 4.133.4.32 `int libxl_device_net2_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_net2 * net2 )`
  - 4.133.4.33 `int libxl_device_net2_del ( libxl_ctx * ctx, libxl_device_net2 * net2, int wait )`
  - 4.133.4.34 `void libxl_device_net2_init ( libxl_device_net2 * net2, int dev_num )`
  - 4.133.4.35 `libxl_net2info* libxl_device_net2_list ( libxl_ctx * ctx, uint32_t domid, unsigned int * nb )`
  - 4.133.4.36 `int libxl_device_nic_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_nic * nic )`
  - 4.133.4.37 `int libxl_device_nic_del ( libxl_ctx * ctx, libxl_device_nic * nic, int wait )`
  - 4.133.4.38 `int libxl_device_nic_init ( libxl_device_nic * nic, int dev_num )`
  - 4.133.4.39 `int libxl_device_pci_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_pci * pcidev )`
  - 4.133.4.40 `int libxl_device_pci_list_assignable ( libxl_ctx * ctx, libxl_device_pci ** list, int * num )`
  - 4.133.4.41 `int libxl_device_pci_list_assigned ( libxl_ctx * ctx, libxl_device_pci ** list, uint32_t domid, int * num )`
  - 4.133.4.42 `int libxl_device_pci_parse_bdf ( libxl_ctx * ctx, libxl_device_pci * pcidev, const char * str )`

- 4.133.4.43 int libxl\_device\_pci\_remove ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_device\_pci \* pcidev, int force )
- 4.133.4.44 int libxl\_device\_pci\_shutdown ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.45 int libxl\_device\_vfb\_add ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_device\_vfb \* vfb )
- 4.133.4.46 int libxl\_device\_vfb\_clean\_shutdown ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.47 int libxl\_device\_vfb\_hard\_shutdown ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.48 void libxl\_device\_vfb\_init ( libxl\_device\_vfb \* vfb, int dev\_num )
- 4.133.4.49 int libxl\_device\_vkb\_add ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_device\_vkb \* vkb )
- 4.133.4.50 int libxl\_device\_vkb\_clean\_shutdown ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.51 int libxl\_device\_vkb\_hard\_shutdown ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.52 void libxl\_device\_vkb\_init ( libxl\_device\_vkb \* vkb, int dev\_num )
- 4.133.4.53 void libxl\_domain\_config\_destroy ( libxl\_domain\_config \* d\_config )
- 4.133.4.54 int libxl\_domain\_core\_dump ( libxl\_ctx \* ctx, uint32\_t domid, const char \* filename )
- 4.133.4.55 int libxl\_domain\_create\_new ( libxl\_ctx \* ctx, libxl\_domain\_config \* d\_config, libxl\_console\_ready cb, void \* priv, uint32\_t \* domid )
- 4.133.4.56 int libxl\_domain\_create\_restore ( libxl\_ctx \* ctx, libxl\_domain\_config \* d\_config, libxl\_console\_ready cb, void \* priv, uint32\_t \* domid, int restore\_fd )
- 4.133.4.57 int libxl\_domain\_destroy ( libxl\_ctx \* ctx, uint32\_t domid, int force )
- 4.133.4.58 int libxl\_domain\_info ( libxl\_ctx \* , libxl\_dominfo \* info\_r, uint32\_t domid )
- 4.133.4.59 int libxl\_domain\_need\_memory ( libxl\_ctx \* ctx, libxl\_domain\_build\_info \* b\_info, libxl\_device\_model\_info \* dm\_info, int domc, uint32\_t \* need\_memkb )

how much free memory in the system a domain needs to be built

- 4.133.4.60 int libxl\_domain\_pause ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.61 int libxl\_domain\_preserve ( libxl\_ctx \* ctx, uint32\_t domid, libxl\_domain\_create\_info \* info, const char \* name\_suffix, libxl\_uuid new\_uuid )

Preserves a domain but rewrites xenstore etc to make it unique so that the domain can be restarted.

Does not modify info so that it may be reused.

- 4.133.4.62 int libxl\_domain\_rename ( libxl\_ctx \* ctx, uint32\_t domid, const char \* old\_name, const char \* new\_name, xs\_transaction\_t trans )
- 4.133.4.63 int libxl\_domain\_resume ( libxl\_ctx \* ctx, uint32\_t domid )
- 4.133.4.64 int libxl\_domain\_setmaxmem ( libxl\_ctx \* ctx, uint32\_t domid, uint32\_t target\_memkb )

4.133.4.65 `int libxl_domain_shutdown ( libxl_ctx * ctx, uint32_t domid, int req )`

4.133.4.66 `int libxl_domain_suspend ( libxl_ctx * ctx, libxl_domain_suspend_info * info, uint32_t domid, int fd )`

4.133.4.67 `int libxl_domain_unpause ( libxl_ctx * ctx, uint32_t domid )`

4.133.4.68 `int libxl_event_get_disk_eject_info ( libxl_ctx * ctx, uint32_t domid, libxl_event * event, libxl_device_disk * disk )`

Returns true and fills \*disk if the caller should eject the disk

4.133.4.69 `int libxl_event_get_domain_death_info ( libxl_ctx * ctx, uint32_t domid, libxl_event * event, libxl_dominfo * info )`

Returns:

- 0 if the domain is dead but there is no cleanup to be done. e.g because someone else has already done it.
- 1 if the domain is dead and there is cleanup to be done.

Can return error if the domain exists and is still running.

\*info will contain valid domain state iff 1 is returned. In particular if 1 is returned then info->shutdown\_reason is guaranteed to be valid since by definition the domain is (shutdown||dying))

4.133.4.70 `void libxl_file_reference_destroy ( libxl_file_reference * p )`

4.133.4.71 `int libxl_free_event ( libxl_event * event )`

4.133.4.72 `int libxl_free_waiter ( libxl_waiter * waiter )`

4.133.4.73 `int libxl_get_event ( libxl_ctx * ctx, libxl_event * event )`

4.133.4.74 `int libxl_get_free_memory ( libxl_ctx * ctx, uint32_t * memkb )`

how much free memory is available in the system

4.133.4.75 `int libxl_get_freecpus ( libxl_ctx * ctx, libxl_cpumap * cpumap )`

4.133.4.76 `int libxl_get_max_cpus ( libxl_ctx * ctx )`

4.133.4.77 `int libxl_get_memory_target ( libxl_ctx * ctx, uint32_t domid, uint32_t * out_target )`

4.133.4.78 `int libxl_get_physinfo ( libxl_ctx * ctx, libxl_physinfo * physinfo )`

4.133.4.79 `int libxl_get_sched_id ( libxl_ctx * ctx )`

returns one of the XEN\_SCHEDULER\_\* constants from [public/domctl.h](http://public.domctl.h)

4.133.4.80 `int libxl_get_topologyinfo ( libxl_ctx * ctx, libxl_topologyinfo * info )`

4.133.4.81 `const libxl_version_info* libxl_get_version_info ( libxl_ctx * ctx )`

4.133.4.82 `int libxl_get_wait_fd ( libxl_ctx * ctx, int * fd )`

4.133.4.83 void libxl\_init\_build\_info ( libxl\_domain\_build\_info \* *b\_info*, libxl\_domain\_create\_info \* *c\_info* )

4.133.4.84 void libxl\_init\_create\_info ( libxl\_domain\_create\_info \* *c\_info* )

4.133.4.85 void libxl\_init\_dm\_info ( libxl\_device\_model\_info \* *dm\_info*, libxl\_domain\_create\_info \* *c\_info*, libxl\_domain\_build\_info \* *b\_info* )

4.133.4.86 void libxl\_key\_value\_list\_destroy ( libxl\_key\_value\_list \* *kvl* )

4.133.4.87 const char\* libxl\_libdir\_path ( void )

4.133.4.88 const char\* libxl\_libexec\_path ( void )

4.133.4.89 libxl\_cpupoolinfo\* libxl\_list\_cpupool ( libxl\_ctx \* , int \* *nb\_pool* )

4.133.4.90 libxl\_dominfo\* libxl\_list\_domain ( libxl\_ctx \* , int \* *nb\_domain* )

4.133.4.91 libxl\_nicinfo\* libxl\_list\_nics ( libxl\_ctx \* *ctx*, uint32\_t *domid*, unsigned int \* *nb* )

4.133.4.92 libxl\_vcpuinfo\* libxl\_list\_vcpu ( libxl\_ctx \* *ctx*, uint32\_t *domid*, int \* *nb\_vcpu*, int \* *nrcpus* )

4.133.4.93 libxl\_vminfo\* libxl\_list\_vm ( libxl\_ctx \* *ctx*, int \* *nb\_vm* )

this API call only list VM running on this host. a VM can be an aggregate of multiple domains.

4.133.4.94 const char\* libxl\_lock\_dir\_path ( void )

4.133.4.95 int libxl\_primary\_console\_exec ( libxl\_ctx \* *ctx*, uint32\_t *domid\_vm* )

libxl\_primary\_console\_exec finds the domid and console number corresponding to the primary console of the given vm, then calls libxl\_console\_exec with the right arguments (domid might be different if the guest is using stubdoms). This function can be called after creating the device model, in case of HVM guests, and before libxl\_run\_bootloader in case of PV guests using pygrub.

4.133.4.96 const char\* libxl\_private\_bindir\_path ( void )

4.133.4.97 int libxl\_run\_bootloader ( libxl\_ctx \* *ctx*, libxl\_domain\_build\_info \* *info*, libxl\_device\_disk \* *disk*, uint32\_t *domid* )

Run the configured bootloader for a PV domain and update info->kernel, info->u.pv.ramdisk and info->u.pv.cmdline as appropriate (any initial values present in these fields must have been allocated with malloc).

Is a NOP on non-PV domains or those with no bootloader configured.

Users should call libxl\_file\_reference\_unmap on the kernel and ramdisk to cleanup or rely on libxl\_domain\_{build,restore} to do it.

4.133.4.98 const char\* libxl\_sbindir\_path ( void )

4.133.4.99 int libxl\_sched\_credit\_domain\_get ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_sched\_credit \* *scinfo* )

4.133.4.100 int libxl\_sched\_credit\_domain\_set ( libxl\_ctx \* *ctx*, uint32\_t *domid*, libxl\_sched\_credit \* *scinfo* )

4.133.4.101 int libxl\_send\_debug\_keys ( libxl\_ctx \* *ctx*, char \* *keys* )

- 4.133.4.102 `int libxl_send_sysrq ( libxl_ctx * ctx, uint32_t domid, char sysrq )`
- 4.133.4.103 `int libxl_send_trigger ( libxl_ctx * ctx, uint32_t domid, char * trigger_name, uint32_t vcpuid )`
- 4.133.4.104 `int libxl_set_memory_target ( libxl_ctx * ctx, uint32_t domid, int32_t target_memkb, int relative, int enforce )`
- 4.133.4.105 `int libxl_set_vcpuaffinity ( libxl_ctx * ctx, uint32_t domid, uint32_t vcpuid, libxl_cpumap * cpumap )`
- 4.133.4.106 `int libxl_set_vcpuonline ( libxl_ctx * ctx, uint32_t domid, libxl_cpumap * cpumap )`
- 4.133.4.107 `const char* libxl_sharedir_path ( void )`
- 4.133.4.108 `int libxl_stop_waiting ( libxl_ctx * ctx, libxl_waiter * waiter )`
- 4.133.4.109 `void libxl_string_list_destroy ( libxl_string_list * sl )`
- 4.133.4.110 `int libxl_tmem_destroy ( libxl_ctx * ctx, uint32_t domid )`
- 4.133.4.111 `int libxl_tmem_freeable ( libxl_ctx * ctx )`
- 4.133.4.112 `int libxl_tmem_freeze ( libxl_ctx * ctx, uint32_t domid )`
- 4.133.4.113 `char* libxl_tmem_list ( libxl_ctx * ctx, uint32_t domid, int use_long )`
- 4.133.4.114 `int libxl_tmem_set ( libxl_ctx * ctx, uint32_t domid, char * name, uint32_t set )`
- 4.133.4.115 `int libxl_tmem_shared_auth ( libxl_ctx * ctx, uint32_t domid, char * uuid, int auth )`
- 4.133.4.116 `int libxl_tmem_thaw ( libxl_ctx * ctx, uint32_t domid )`
- 4.133.4.117 `int libxl_userdata_retrieve ( libxl_ctx * ctx, uint32_t domid, const char * userdata_userid, uint8_t ** data_r, int * datalen_r )`
- 4.133.4.118 `int libxl_userdata_store ( libxl_ctx * ctx, uint32_t domid, const char * userdata_userid, const uint8_t * data, int datalen )`
- 4.133.4.119 `uint32_t libxl_vm_get_start_time ( libxl_ctx * ctx, uint32_t domid )`
- 4.133.4.120 `int libxl_vncviewer_exec ( libxl_ctx * ctx, uint32_t domid, int autopass )`
- 4.133.4.121 `int libxl_wait_for_disk_ejects ( libxl_ctx * ctx, uint32_t domid, libxl_device_disk * disks, int num_disks, libxl_waiter * waiter )`
- 4.133.4.122 `int libxl_wait_for_domain_death ( libxl_ctx * ctx, uint32_t domid, libxl_waiter * waiter )`
- 4.133.4.123 `int libxl_wait_for_free_memory ( libxl_ctx * ctx, uint32_t domid, uint32_t memory_kb, int wait_secs )`  
wait for a given amount of memory to be free in the system
- 4.133.4.124 `int libxl_wait_for_memory_target ( libxl_ctx * ctx, uint32_t domid, int wait_secs )`  
wait for the memory target of a domain to be reached
- 4.133.4.125 `const char* libxl_xen_config_dir_path ( void )`



4.133.4.126 void libxl\_xen\_console\_read\_finish ( libxl\_ctx \* ctx, libxl\_xen\_console\_reader \* cr )

4.133.4.127 int libxl\_xen\_console\_read\_line ( libxl\_ctx \* ctx, libxl\_xen\_console\_reader \* cr, char \*\* line\_r )

return values: *line\_r* 1 success, whole line obtained from buffer non-0 0 no more lines available right now 0 negative error code *ERROR* 0 On success \*line\_r is updated to point to a nul-terminated string which is valid until the next call on the same console reader. The libxl caller may overwrite parts of the string if it wishes.

4.133.4.128 libxl\_xen\_console\_reader\* libxl\_xen\_console\_read\_start ( libxl\_ctx \* ctx, int clear )

4.133.4.129 const char\* libxl\_xen\_script\_dir\_path ( void )

4.133.4.130 const char\* libxl\_xenfirmware\_dir\_path ( void )

## 4.134 xen/tools/libxl/libxl.idl File Reference

### 4.135 xen/tools/libxl/libxl\_blkmap2.c File Reference

```
#include "libxl.h"
#include "libxl_osdeps.h"
#include "libxl_internal.h"
#include "tap-ctl.h"
```

#### Functions

- int libxl\_\_blkmap\_enabled (libxl\_\_gc \*gc)
- const char \* libxl\_\_blkmap\_devpath (libxl\_\_gc \*gc, const char \*disk, libxl\_disk\_format format)

#### 4.135.1 Function Documentation

4.135.1.1 const char\* libxl\_\_blkmap\_devpath ( libxl\_\_gc \* gc, const char \* disk, libxl\_disk\_format format )

4.135.1.2 int libxl\_\_blkmap\_enabled ( libxl\_\_gc \* gc )

### 4.136 xen/tools/libxl/libxl\_create.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <xenctrl.h>
#include <xc_dom.h>
#include <xenguest.h>
#include <assert.h>
#include "libxl.h"
#include "libxl_utils.h"
#include "libxl_internal.h"
#include "flexarray.h"
```

## Functions

- void `libxl_domain_config_destroy` (`libxl_domain_config` \*d\_config)
- void `libxl_init_create_info` (`libxl_domain_create_info` \*c\_info)
- void `libxl_init_build_info` (`libxl_domain_build_info` \*b\_info, `libxl_domain_create_info` \*c\_info)
- void `libxl_init_dm_info` (`libxl_device_model_info` \*dm\_info, `libxl_domain_create_info` \*c\_info, `libxl_domain_build_info` \*b\_info)
- int `init_console_info` (`libxl_device_console` \*console, int dev\_num, `libxl_domain_build_state` \*state)
- int `libxl__domain_build` (`libxl_ctx` \*ctx, `libxl_domain_build_info` \*info, uint32\_t domid, `libxl_domain_build_state` \*state)
- int `libxl__domain_make` (`libxl_ctx` \*ctx, `libxl_domain_create_info` \*info, uint32\_t \*domid)
- int `libxl_domain_create_new` (`libxl_ctx` \*ctx, `libxl_domain_config` \*d\_config, `libxl_console_ready` cb, void \*priv, uint32\_t \*domid)
- int `libxl_domain_create_restore` (`libxl_ctx` \*ctx, `libxl_domain_config` \*d\_config, `libxl_console_ready` cb, void \*priv, uint32\_t \*domid, int restore\_fd)

### 4.136.1 Function Documentation

4.136.1.1 int `init_console_info` ( `libxl_device_console` \* console, int dev\_num, `libxl_domain_build_state` \* state )

4.136.1.2 int `libxl__domain_build` ( `libxl_ctx` \* ctx, `libxl_domain_build_info` \* info, uint32\_t domid, `libxl_domain_build_state` \* state )

4.136.1.3 int `libxl__domain_make` ( `libxl_ctx` \* ctx, `libxl_domain_create_info` \* info, uint32\_t \* domid )

4.136.1.4 void `libxl_domain_config_destroy` ( `libxl_domain_config` \* d\_config )

4.136.1.5 int `libxl_domain_create_new` ( `libxl_ctx` \* ctx, `libxl_domain_config` \* d\_config, `libxl_console_ready` cb, void \* priv, uint32\_t \* domid )

4.136.1.6 int `libxl_domain_create_restore` ( `libxl_ctx` \* ctx, `libxl_domain_config` \* d\_config, `libxl_console_ready` cb, void \* priv, uint32\_t \* domid, int restore\_fd )

4.136.1.7 void `libxl_init_build_info` ( `libxl_domain_build_info` \* b\_info, `libxl_domain_create_info` \* c\_info )

4.136.1.8 void `libxl_init_create_info` ( `libxl_domain_create_info` \* c\_info )

4.136.1.9 void `libxl_init_dm_info` ( `libxl_device_model_info` \* dm\_info, `libxl_domain_create_info` \* c\_info, `libxl_domain_build_info` \* b\_info )

### 4.137 xen/tools/libxl/libxl\_device.c File Reference

```
#include "libxl_osdeps.h"
#include <string.h>
#include <stdio.h>
#include <sys/time.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <fcntl.h>
#include "libxl.h"
#include "libxl_internal.h"
```

## Functions

- char \* [libxl\\_\\_device\\_frontend\\_path](#) (libxl\_\_gc \*gc, libxl\_\_device \*device)
- char \* [libxl\\_\\_device\\_backend\\_path](#) (libxl\_\_gc \*gc, libxl\_\_device \*device)
- int [libxl\\_\\_device\\_generic\\_add](#) (libxl\_ctx \*ctx, libxl\_\_device \*device, char \*\*bents, char \*\*fents)
- char \* [libxl\\_\\_device\\_disk\\_string\\_of\\_format](#) (libxl\_disk\_format format)
- char \* [libxl\\_\\_device\\_disk\\_string\\_of\\_backend](#) (libxl\_disk\_backend backend)
- int [libxl\\_\\_device\\_physdisk\\_major\\_minor](#) (const char \*physpath, int \*major, int \*minor)
- int [libxl\\_\\_device\\_disk\\_dev\\_number](#) (const char \*virtpath)
- int [libxl\\_\\_device\\_destroy](#) (libxl\_ctx \*ctx, char \*be\_path, int force)
- int [libxl\\_\\_devices\\_destroy](#) (libxl\_ctx \*ctx, uint32\_t domid, int force)
- int [libxl\\_\\_device\\_del](#) (libxl\_ctx \*ctx, libxl\_\_device \*dev, int wait)
- int [libxl\\_\\_wait\\_for\\_device\\_model](#) (libxl\_ctx \*ctx, uint32\_t domid, char \*state, int(\*check\_callback)(libxl\_ctx \*ctx, uint32\_t domid, const char \*state, void \*userdata), void \*check\_callback\_userdata)
- int [libxl\\_\\_wait\\_for\\_backend](#) (libxl\_ctx \*ctx, char \*be\_path, char \*state)

### 4.137.1 Function Documentation

- 4.137.1.1 char\* [libxl\\_\\_device\\_backend\\_path](#) ( libxl\_\_gc \* gc, libxl\_\_device \* device )
- 4.137.1.2 int [libxl\\_\\_device\\_del](#) ( libxl\_ctx \* ctx, libxl\_\_device \* dev, int wait )
- 4.137.1.3 int [libxl\\_\\_device\\_destroy](#) ( libxl\_ctx \* ctx, char \* be\_path, int force )
- 4.137.1.4 int [libxl\\_\\_device\\_disk\\_dev\\_number](#) ( const char \* virtpath )
- 4.137.1.5 char\* [libxl\\_\\_device\\_disk\\_string\\_of\\_backend](#) ( libxl\_disk\_backend backend )
- 4.137.1.6 char\* [libxl\\_\\_device\\_disk\\_string\\_of\\_format](#) ( libxl\_disk\_format format )
- 4.137.1.7 char\* [libxl\\_\\_device\\_frontend\\_path](#) ( libxl\_\_gc \* gc, libxl\_\_device \* device )
- 4.137.1.8 int [libxl\\_\\_device\\_generic\\_add](#) ( libxl\_ctx \* ctx, libxl\_\_device \* device, char \*\* bents, char \*\* fents )
- 4.137.1.9 int [libxl\\_\\_device\\_physdisk\\_major\\_minor](#) ( const char \* physpath, int \* major, int \* minor )
- 4.137.1.10 int [libxl\\_\\_devices\\_destroy](#) ( libxl\_ctx \* ctx, uint32\_t domid, int force )
- 4.137.1.11 int [libxl\\_\\_wait\\_for\\_backend](#) ( libxl\_ctx \* ctx, char \* be\_path, char \* state )
- 4.137.1.12 int [libxl\\_\\_wait\\_for\\_device\\_model](#) ( libxl\_ctx \* ctx, uint32\_t domid, char \* state, int(\*) (libxl\_ctx \* ctx, uint32\_t domid, const char \* state, void \* userdata) check\_callback, void \* check\_callback\_userdata )

## 4.138 xen/tools/libxl/libxl\_dm.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <signal.h>
#include <unistd.h>
#include <fcntl.h>
#include <assert.h>
#include "libxl_utils.h"
#include "libxl_internal.h"
#include "libxl.h"
#include "flexarray.h"
```

### Functions

- `int libxl__create_device_model (libxl_ctx *ctx, libxl_device_model_info *info, libxl_device_disk *disks, int num_disks, libxl_device_nic *vifs, int num_vifs, libxl__device_model_starting **starting_r)`
- `int libxl__confirm_device_model_startup (libxl_ctx *ctx, libxl__device_model_starting *starting)`
- `int libxl__destroy_device_model (libxl_ctx *ctx, uint32_t domid)`
- `int libxl__need_xenpv_qemu (libxl_ctx *ctx, int nr_consoles, libxl_device_console *consoles, int nr_vfbs, libxl__device_vfb *vfbs, int nr_disks, libxl_device_disk *disks)`
- `int libxl__create_xenpv_qemu (libxl_ctx *ctx, uint32_t domid, libxl_device_vfb *vfb, libxl__device_model_starting **starting_r)`

### 4.138.1 Function Documentation

4.138.1.1 `int libxl__confirm_device_model_startup ( libxl_ctx * ctx, libxl__device_model_starting * starting )`

4.138.1.2 `int libxl__create_device_model ( libxl_ctx * ctx, libxl_device_model_info * info, libxl_device_disk * disks, int num_disks, libxl_device_nic * vifs, int num_vifs, libxl__device_model_starting ** starting_r )`

4.138.1.3 `int libxl__create_xenpv_qemu ( libxl_ctx * ctx, uint32_t domid, libxl_device_vfb * vfb, libxl__device_model_starting ** starting_r )`

4.138.1.4 `int libxl__destroy_device_model ( libxl_ctx * ctx, uint32_t domid )`

4.138.1.5 `int libxl__need_xenpv_qemu ( libxl_ctx * ctx, int nr_consoles, libxl_device_console * consoles, int nr_vfbs, libxl_device_vfb * vfbs, int nr_disks, libxl_device_disk * disks )`

## 4.139 xen/tools/libxl/libxl\_dom.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <assert.h>
#include <glob.h>
#include <inttypes.h>
#include <string.h>
#include <sys/mman.h>
#include <sys/time.h>
#include <sys/stat.h>
#include <unistd.h>
#include <xenctrl.h>
#include <xc_dom.h>
#include <xenguest.h>
#include <fcntl.h>
#include <xen/hvm/hvm_info_table.h>
#include "libxl.h"
#include "libxl_internal.h"
```

### Data Structures

- struct [suspendinfo](#)

### Functions

- int [libxl\\_\\_domain\\_is\\_hvm](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_\\_domain\\_shutdown\\_reason](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_\\_build\\_pre](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_build\\_info](#) \*info, [libxl\\_domain\\_build\\_state](#) \*state)
- int [libxl\\_\\_build\\_post](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_build\\_info](#) \*info, [libxl\\_domain\\_build\\_state](#) \*state, char \*\*vms\_ents, char \*\*local\_ents)
- int [libxl\\_\\_build\\_pv](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_build\\_info](#) \*info, [libxl\\_domain\\_build\\_state](#) \*state)
- int [libxl\\_\\_build\\_hvm](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_build\\_info](#) \*info, [libxl\\_domain\\_build\\_state](#) \*state)
- int [libxl\\_\\_domain\\_restore\\_common](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, [libxl\\_domain\\_build\\_info](#) \*info, [libxl\\_\\_domain\\_build\\_state](#) \*state, int fd)
- int [libxl\\_\\_domain\\_suspend\\_common](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int fd, int hvm, int live, int debug)
- int [libxl\\_\\_domain\\_save\\_device\\_model](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, int fd)
- char \* [libxl\\_\\_uuid2string](#) ([libxl\\_\\_gc](#) \*gc, const [libxl\\_uuid](#) uuid)
- void [libxl\\_\\_userdata\\_destroyall](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid)
- int [libxl\\_userdata\\_store](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, const char \*userdata\_userid, const uint8\_t \*data, int datalen)
- int [libxl\\_userdata\\_retrieve](#) ([libxl\\_ctx](#) \*ctx, uint32\_t domid, const char \*userdata\_userid, uint8\_t \*\*data\_r, int \*datalen\_r)

### 4.139.1 Function Documentation

4.139.1.1 int [libxl\\_\\_build\\_hvm](#) ( [libxl\\_ctx](#) \* ctx, uint32\_t *domid*, [libxl\\_domain\\_build\\_info](#) \* *info*, [libxl\\_domain\\_build\\_state](#) \* *state* )

- 4.139.1.2 `int libxl__build_post ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state, char ** vms_ents, char ** local_ents )`
- 4.139.1.3 `int libxl__build_pre ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state )`
- 4.139.1.4 `int libxl__build_pv ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state )`
- 4.139.1.5 `int libxl__domain_is_hvm ( libxl_ctx * ctx, uint32_t domid )`
- 4.139.1.6 `int libxl__domain_restore_common ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state, int fd )`
- 4.139.1.7 `int libxl__domain_save_device_model ( libxl_ctx * ctx, uint32_t domid, int fd )`
- 4.139.1.8 `int libxl__domain_shutdown_reason ( libxl_ctx * ctx, uint32_t domid )`
- 4.139.1.9 `int libxl__domain_suspend_common ( libxl_ctx * ctx, uint32_t domid, int fd, int hvm, int live, int debug )`
- 4.139.1.10 `void libxl__userdata_destroyall ( libxl_ctx * ctx, uint32_t domid )`
- 4.139.1.11 `char* libxl__uuid2string ( libxl__gc * gc, const libxl_uuid uuid )`
- 4.139.1.12 `int libxl_userdata_retrieve ( libxl_ctx * ctx, uint32_t domid, const char * userdata_userid, uint8_t ** data_r, int * datalen_r )`
- 4.139.1.13 `int libxl_userdata_store ( libxl_ctx * ctx, uint32_t domid, const char * userdata_userid, const uint8_t * data, int datalen )`

## 4.140 xen/tools/libxl/libxl\_internal.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <stdarg.h>
#include <string.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <sys/mman.h>
#include <unistd.h>
#include "libxl.h"
#include "libxl_internal.h"
#include "libxl_utils.h"
```

### Functions

- `int libxl__error_set (libxl_ctx *ctx, int code)`
- `int libxl__ptr_add (libxl__gc *gc, void *ptr)`
- `void libxl__free_all (libxl__gc *gc)`
- `void * libxl__zalloc (libxl__gc *gc, int bytes)`
- `void * libxl__calloc (libxl__gc *gc, size_t nmemb, size_t size)`
- `char * libxl__sprintf (libxl__gc *gc, const char *fmt,...)`
- `char * libxl__strdup (libxl__gc *gc, const char *c)`

- char \* [libxl\\_\\_dirname](#) ([libxl\\_\\_gc](#) \*gc, const char \*s)
- void [libxl\\_\\_logv](#) ([libxl\\_ctx](#) \*ctx, xentoollog\_level msglevel, int errnoval, const char \*file, int line, const char \*func, char \*fmt, va\_list ap)
- void [libxl\\_\\_log](#) ([libxl\\_ctx](#) \*ctx, xentoollog\_level msglevel, int errnoval, const char \*file, int line, const char \*func, char \*fmt,...)
- char \* [libxl\\_\\_abs\\_path](#) ([libxl\\_\\_gc](#) \*gc, char \*s, const char \*path)
- int [libxl\\_\\_file\\_reference\\_map](#) ([libxl\\_file\\_reference](#) \*f)
- int [libxl\\_\\_file\\_reference\\_unmap](#) ([libxl\\_file\\_reference](#) \*f)

#### 4.140.1 Function Documentation

4.140.1.1 char\* [libxl\\_\\_abs\\_path](#) ( [libxl\\_\\_gc](#) \* gc, char \* s, const char \* path )

4.140.1.2 void\* [libxl\\_\\_calloc](#) ( [libxl\\_\\_gc](#) \* gc, size\_t nmemb, size\_t size )

4.140.1.3 char\* [libxl\\_\\_dirname](#) ( [libxl\\_\\_gc](#) \* gc, const char \* s )

4.140.1.4 int [libxl\\_\\_error\\_set](#) ( [libxl\\_ctx](#) \* ctx, int code )

4.140.1.5 int [libxl\\_\\_file\\_reference\\_map](#) ( [libxl\\_file\\_reference](#) \* f )

4.140.1.6 int [libxl\\_\\_file\\_reference\\_unmap](#) ( [libxl\\_file\\_reference](#) \* f )

4.140.1.7 void [libxl\\_\\_free\\_all](#) ( [libxl\\_\\_gc](#) \* gc )

4.140.1.8 void [libxl\\_\\_log](#) ( [libxl\\_ctx](#) \* ctx, xentoollog\_level msglevel, int errnoval, const char \* file, int line, const char \* func, char \* fmt, ... )

4.140.1.9 void [libxl\\_\\_logv](#) ( [libxl\\_ctx](#) \* ctx, xentoollog\_level msglevel, int errnoval, const char \* file, int line, const char \* func, char \* fmt, va\_list ap )

4.140.1.10 int [libxl\\_\\_ptr\\_add](#) ( [libxl\\_\\_gc](#) \* gc, void \* ptr )

4.140.1.11 char\* [libxl\\_\\_sprintf](#) ( [libxl\\_\\_gc](#) \* gc, const char \* fmt, ... )

4.140.1.12 char\* [libxl\\_\\_strdup](#) ( [libxl\\_\\_gc](#) \* gc, const char \* c )

4.140.1.13 void\* [libxl\\_\\_zalloc](#) ( [libxl\\_\\_gc](#) \* gc, int bytes )

### 4.141 xen/tools/libxl/libxl\_internal.h File Reference

```
#include <stdint.h>
#include <stdarg.h>
#include <stdlib.h>
#include <xs.h>
#include <xenctrl.h>
#include "xentoollog.h"
#include "flexarray.h"
#include "libxl_utils.h"
```

#### Data Structures

- struct [libxl\\_\\_device](#)

- struct `libxl__gc`
- struct `libxl__spawn_starting`
- struct `libxl__device_model_starting`
- struct `libxl__cpuid_policy`
- struct `libxl__xen_console_reader`

## Macros

- `#define _hidden`
- `#define _protected`
- `#define LIBXL_DESTROY_TIMEOUT 10`
- `#define LIBXL_DEVICE_MODEL_START_TIMEOUT 10`
- `#define LIBXL_XENCONSOLE_LIMIT 1048576`
- `#define LIBXL_XENCONSOLE_PROTOCOL "vt100"`
- `#define LIBXL_MAXMEM_CONSTANT 1024`
- `#define LIBXL_PV_EXTRA_MEMORY 1024`
- `#define LIBXL_HVM_EXTRA_MEMORY 2048`
- `#define LIBXL_MIN_DOM0_MEM (128*1024)`
- `#define QEMU_SIGNATURE "DeviceModelRecord0002"`
- `#define STUBDOM_CONSOLE_LOGGING 0`
- `#define STUBDOM_CONSOLE_SAVE 1`
- `#define STUBDOM_CONSOLE_RESTORE 2`
- `#define STUBDOM_CONSOLE_SERIAL 3`
- `#define STUBDOM_SPECIAL_CONSOLES 3`
- `#define SAVEFILE "/var/lib/xen/qemu-save"`
- `#define ARRAY_SIZE(a) (sizeof(a) / sizeof(a[0]))`
- `#define LIBXL__LOGGING_ENABLED`
- `#define LIBXL__LOG(ctx, loglevel, _f, _a...) libxl__log(ctx, loglevel, -1, __FILE__, __LINE__, __func__, _f, ##_a)`
- `#define LIBXL__LOG_ERRNO(ctx, loglevel, _f, _a...) libxl__log(ctx, loglevel, errno, __FILE__, __LINE__, __func__, _f, ##_a)`
- `#define LIBXL__LOG_ERRNOVAL(ctx, loglevel, errnoval, _f, _a...) libxl__log(ctx, loglevel, errnoval, __FILE__, __LINE__, __func__, _f, ##_a)`
- `#define is_valid_device_kind(kind) (((kind) >= DEVICE_VIF) && ((kind) <= DEVICE_CONSOLE))`
- `#define XC_PCI_BDF "0x%x, 0x%x, 0x%x, 0x%x"`
- `#define AUTO_PHP_SLOT 0x100`
- `#define SYSFS_PCI_DEV "/sys/bus/pci/devices"`
- `#define SYSFS_PCIBACK_DRIVER "/sys/bus/pci/drivers/pciback"`
- `#define XENSTORE_PID_FILE "/var/run/xenstored.pid"`
- `#define PROC_PCI_NUM_RESOURCES 7`
- `#define PCI_BAR_IO 0x01`
- `#define PRINTF_ATTRIBUTE(x, y) __attribute__((format(printf, x, y)))`
- `#define LIBXL_INIT_GC(ctx) (libxl__gc){ .alloc_maxsize = 0, .alloc_ptrs = 0, .owner = ctx }`
- `#define LIBXL__LOG_DEBUG XTL_DEBUG`
- `#define LIBXL__LOG_INFO XTL_INFO`
- `#define LIBXL__LOG_WARNING XTL_WARN`
- `#define LIBXL__LOG_ERROR XTL_ERROR`

## Enumerations

- enum `libxl__device_kinds` {  
`DEVICE_VIF = 1, DEVICE_VIF2, DEVICE_VBD, DEVICE_TAP,`  
`DEVICE_QDISK, DEVICE_PCI, DEVICE_VFB, DEVICE_VKBD,`  
`DEVICE_CONSOLE }`



## Functions

- [\\_hidden](#) void [libxl\\_\\_logv](#) ([libxl\\_\\_ctx](#) \*ctx, xentoollog\_level msglevel, int errnoval, const char \*file, int line, const char \*func, char \*fmt, va\_list al) [\\_\\_attribute\\_\\_\(\(format\(printf](#)
- [\\_hidden](#) void [\\_hidden](#) void [libxl\\_\\_log](#) ([libxl\\_\\_ctx](#) \*ctx, xentoollog\_level msglevel, int errnoval, const char \*file, int line, const char \*func, char \*fmt,...) [\\_\\_attribute\\_\\_\(\(format\(printf](#)
- [\\_hidden](#) int [libxl\\_\\_ptr\\_add](#) ([libxl\\_\\_gc](#) \*gc, void \*ptr)
- [\\_hidden](#) void [libxl\\_\\_free\\_all](#) ([libxl\\_\\_gc](#) \*gc)
- [\\_hidden](#) void \* [libxl\\_\\_zalloc](#) ([libxl\\_\\_gc](#) \*gc, int bytes)
- [\\_hidden](#) void \* [libxl\\_\\_calloc](#) ([libxl\\_\\_gc](#) \*gc, size\_t nmem, size\_t size)
- [\\_hidden](#) char \* [libxl\\_\\_sprintf](#) ([libxl\\_\\_gc](#) \*gc, const char \*fmt,...) [PRINTF\\_ATTRIBUTE\(2](#)
- [\\_hidden](#) char [\\_hidden](#) char \* [libxl\\_\\_strdup](#) ([libxl\\_\\_gc](#) \*gc, const char \*c)
- [\\_hidden](#) char \* [libxl\\_\\_dirname](#) ([libxl\\_\\_gc](#) \*gc, const char \*s)
- [\\_hidden](#) char \*\* [libxl\\_\\_xs\\_kvs\\_of\\_flexarray](#) ([libxl\\_\\_gc](#) \*gc, flexarray\_t \*array, int length)
- [\\_hidden](#) int [libxl\\_\\_xs\\_writev](#) ([libxl\\_\\_gc](#) \*gc, xs\_transaction\_t t, char \*dir, char \*\*kvs)
- [\\_hidden](#) int [libxl\\_\\_xs\\_write](#) ([libxl\\_\\_gc](#) \*gc, xs\_transaction\_t t, char \*path, char \*fmt,...) [PRINTF\\_ATTRIBUTE\(4](#)
- [\\_hidden](#) int [\\_hidden](#) char \* [libxl\\_\\_xs\\_get\\_dompath](#) ([libxl\\_\\_gc](#) \*gc, uint32\_t domid)
- [\\_hidden](#) char \* [libxl\\_\\_xs\\_read](#) ([libxl\\_\\_gc](#) \*gc, xs\_transaction\_t t, char \*path)
- [\\_hidden](#) char \*\* [libxl\\_\\_xs\\_directory](#) ([libxl\\_\\_gc](#) \*gc, xs\_transaction\_t t, char \*path, unsigned int \*nb)
- [\\_hidden](#) int [libxl\\_\\_domain\\_is\\_hvm](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid)
- [\\_hidden](#) int [libxl\\_\\_domain\\_shutdown\\_reason](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid)
- [\\_hidden](#) int [libxl\\_\\_build\\_pre](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_domain\_build\_info \*info, libxl\_domain\_build\_state \*state)
- [\\_hidden](#) int [libxl\\_\\_build\\_post](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_domain\_build\_info \*info, libxl\_domain\_build\_state \*state, char \*\*vms\_ents, char \*\*local\_ents)
- [\\_hidden](#) int [libxl\\_\\_build\\_pv](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_domain\_build\_info \*info, libxl\_domain\_build\_state \*state)
- [\\_hidden](#) int [libxl\\_\\_build\\_hvm](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_domain\_build\_info \*info, libxl\_domain\_build\_state \*state)
- [\\_hidden](#) int [libxl\\_\\_domain\\_restore\\_common](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_domain\_build\_info \*info, libxl\_domain\_build\_state \*state, int fd)
- [\\_hidden](#) int [libxl\\_\\_domain\\_suspend\\_common](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, int fd, int hvm, int live, int debug)
- [\\_hidden](#) int [libxl\\_\\_domain\\_save\\_device\\_model](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, int fd)
- [\\_hidden](#) void [libxl\\_\\_userdata\\_destroyall](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid)
- [\\_hidden](#) char \* [libxl\\_\\_device\\_disk\\_string\\_of\\_backend](#) ([libxl\\_\\_disk\\_backend](#) backend)
- [\\_hidden](#) char \* [libxl\\_\\_device\\_disk\\_string\\_of\\_format](#) ([libxl\\_\\_disk\\_format](#) format)
- [\\_hidden](#) int [libxl\\_\\_device\\_physdisk\\_major\\_minor](#) (const char \*physpath, int \*major, int \*minor)
- [\\_hidden](#) int [libxl\\_\\_device\\_disk\\_dev\\_number](#) (const char \*virtpath)
- [\\_hidden](#) int [libxl\\_\\_device\\_generic\\_add](#) ([libxl\\_\\_ctx](#) \*ctx, [libxl\\_\\_device](#) \*device, char \*\*bents, char \*\*fents)
- [\\_hidden](#) char \* [libxl\\_\\_device\\_backend\\_path](#) ([libxl\\_\\_gc](#) \*gc, [libxl\\_\\_device](#) \*device)
- [\\_hidden](#) char \* [libxl\\_\\_device\\_frontend\\_path](#) ([libxl\\_\\_gc](#) \*gc, [libxl\\_\\_device](#) \*device)
- [\\_hidden](#) int [libxl\\_\\_device\\_del](#) ([libxl\\_\\_ctx](#) \*ctx, [libxl\\_\\_device](#) \*dev, int wait)
- [\\_hidden](#) int [libxl\\_\\_device\\_destroy](#) ([libxl\\_\\_ctx](#) \*ctx, char \*be\_path, int force)
- [\\_hidden](#) int [libxl\\_\\_devices\\_destroy](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, int force)
- [\\_hidden](#) int [libxl\\_\\_wait\\_for\\_device\\_model](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, char \*state, int(\*check\_callback)([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, const char \*state, void \*userdata), void \*check\_callback\_userdata)
- [\\_hidden](#) int [libxl\\_\\_wait\\_for\\_backend](#) ([libxl\\_\\_ctx](#) \*ctx, char \*be\_path, char \*state)
- [\\_hidden](#) int [libxl\\_\\_device\\_pci\\_add](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_device\_pci \*pcidev, int starting)
- int [libxl\\_\\_domain\\_make](#) ([libxl\\_\\_ctx](#) \*ctx, libxl\_domain\_create\_info \*info, uint32\_t \*domid)
- int [libxl\\_\\_domain\\_build](#) ([libxl\\_\\_ctx](#) \*ctx, libxl\_domain\_build\_info \*info, uint32\_t domid, libxl\_domain\_build\_state \*state)
- [\\_hidden](#) int [libxl\\_\\_create\\_device\\_model](#) ([libxl\\_\\_ctx](#) \*ctx, libxl\_device\_model\_info \*info, libxl\_device\_disk \*disk, int num\_disks, libxl\_device\_nic \*vifs, int num\_vifs, [libxl\\_\\_device\\_model\\_starting](#) \*\*starting\_r)
- int [libxl\\_\\_create\\_xenpv\\_qemu](#) ([libxl\\_\\_ctx](#) \*ctx, uint32\_t domid, libxl\_device\_vfb \*vfb, [libxl\\_\\_device\\_model\\_starting](#) \*\*starting\_r)

- `_hidden` int `libxl__need_xenpv_gemu` (`libxl_ctx` \*ctx, int nr\_consoles, `libxl_device_console` \*consoles, int nr\_vfbs, `libxl_device_vfb` \*vfbs, int nr\_disks, `libxl_device_disk` \*disks)
- int `libxl__confirm_device_model_startup` (`libxl_ctx` \*ctx, `libxl__device_model_starting` \*starting)
- `_hidden` int `libxl__spawn_spawn` (`libxl_ctx` \*ctx, `libxl__device_model_starting` \*starting, const char \*what, void(\*intermediate\_hook)(void \*for\_spawn, pid\_t innerchild))
- `_hidden` int `libxl__destroy_device_model` (`libxl_ctx` \*ctx, uint32\_t domid)
- `_hidden` int `libxl__spawn_detach` (`libxl_ctx` \*ctx, `libxl__spawn_starting` \*for\_spawn)
- `_hidden` int `libxl__spawn_check` (`libxl_ctx` \*ctx, void \*for\_spawn)
- `_hidden` void `libxl__exec` (int stdinfd, int stdoutfd, int stderrfd, const char \*arg0, char \*\*args)
- `_hidden` void `libxl__log_child_exitstatus` (`libxl__gc` \*gc, const char \*what, pid\_t pid, int status)
- `_hidden` char \* `libxl__abs_path` (`libxl__gc` \*gc, char \*s, const char \*path)
- `_hidden` char \* `libxl__domid_to_name` (`libxl__gc` \*gc, uint32\_t domid)
- `_hidden` char \* `libxl__cpupoolid_to_name` (`libxl__gc` \*gc, uint32\_t poolid)
- `_hidden` int `libxl__blkmap_enabled` (`libxl__gc` \*gc)
- `_hidden` const char \* `libxl__blkmap_devpath` (`libxl__gc` \*gc, const char \*disk, `libxl__disk_format` format)
- `_hidden` char \* `libxl__uuid2string` (`libxl__gc` \*gc, const `libxl__uuid` uuid)
- `_hidden` int `libxl__error_set` (`libxl_ctx` \*ctx, int code)
- `_hidden` int `libxl__file_reference_map` (`libxl__file_reference` \*f)
- `_hidden` int `libxl__file_reference_unmap` (`libxl__file_reference` \*f)

#### 4.141.1 Macro Definition Documentation

4.141.1.1 `#define _hidden`

4.141.1.2 `#define _protected`

4.141.1.3 `#define ARRAY_SIZE( a ) (sizeof(a) / sizeof(a[0]))`

4.141.1.4 `#define AUTO_PHP_SLOT 0x100`

4.141.1.5 `#define is_valid_device_kind( kind ) (((kind) >= DEVICE_VIF) && ((kind) <= DEVICE_CONSOLE))`

4.141.1.6 `#define LIBXL__LOG( ctx, loglevel, _f, _a... ) libxl__log(ctx, loglevel, -1, __FILE__, __LINE__, __func__, _f, ##_a)`

4.141.1.7 `#define LIBXL__LOG_DEBUG XTL_DEBUG`

4.141.1.8 `#define LIBXL__LOG_ERRNO( ctx, loglevel, _f, _a... ) libxl__log(ctx, loglevel, errno, __FILE__, __LINE__, __func__, _f, ##_a)`

4.141.1.9 `#define LIBXL__LOG_ERRNOVAL( ctx, loglevel, errnoval, _f, _a... ) libxl__log(ctx, loglevel, errnoval, __FILE__, __LINE__, __func__, _f, ##_a)`

4.141.1.10 `#define LIBXL__LOG_ERROR XTL_ERROR`

4.141.1.11 `#define LIBXL__LOG_INFO XTL_INFO`

4.141.1.12 `#define LIBXL__LOG_WARNING XTL_WARN`

4.141.1.13 `#define LIBXL__LOGGING_ENABLED`

4.141.1.14 `#define LIBXL_DESTROY_TIMEOUT 10`

4.141.1.15 `#define LIBXL_DEVICE_MODEL_START_TIMEOUT 10`

- 4.141.1.16 `#define LIBXL_HVM_EXTRA_MEMORY 2048`
- 4.141.1.17 `#define LIBXL_INIT_GC( ctx ) (libxl__gc){ .alloc_maxsize = 0, .alloc_ptrs = 0, .owner = ctx }`
- 4.141.1.18 `#define LIBXL_MAXMEM_CONSTANT 1024`
- 4.141.1.19 `#define LIBXL_MIN_DOM0_MEM (128*1024)`
- 4.141.1.20 `#define LIBXL_PV_EXTRA_MEMORY 1024`
- 4.141.1.21 `#define LIBXL_XENCONSOLE_LIMIT 1048576`
- 4.141.1.22 `#define LIBXL_XENCONSOLE_PROTOCOL "vt100"`
- 4.141.1.23 `#define PCI_BAR_IO 0x01`
- 4.141.1.24 `#define PRINTF_ATTRIBUTE( x, y ) __attribute__((format(printf, x, y)))`
- 4.141.1.25 `#define PROC_PCI_NUM_RESOURCES 7`
- 4.141.1.26 `#define QEMU_SIGNATURE "DeviceModelRecord0002"`
- 4.141.1.27 `#define SAVEFILE "/var/lib/xen/qemu-save"`
- 4.141.1.28 `#define STUBDOM_CONSOLE_LOGGING 0`
- 4.141.1.29 `#define STUBDOM_CONSOLE_RESTORE 2`
- 4.141.1.30 `#define STUBDOM_CONSOLE_SAVE 1`
- 4.141.1.31 `#define STUBDOM_CONSOLE_SERIAL 3`
- 4.141.1.32 `#define STUBDOM_SPECIAL_CONSOLES 3`
- 4.141.1.33 `#define SYSFS_PCI_DEV "/sys/bus/pci/devices"`
- 4.141.1.34 `#define SYSFS_PCIBACK_DRIVER "/sys/bus/pci/drivers/pciback"`
- 4.141.1.35 `#define XC_PCI_BDF "0x%x, 0x%x, 0x%x, 0x%x"`
- 4.141.1.36 `#define XENSTORE_PID_FILE "/var/run/xenstored.pid"`

## 4.141.2 Enumeration Type Documentation

- 4.141.2.1 `enum libxl__device_kinds`

Enumerator

***DEVICE\_VIF***

***DEVICE\_VIF2***

***DEVICE\_VBD***

***DEVICE\_TAP***

***DEVICE\_QDISK***

***DEVICE\_PCI***

***DEVICE\_VFB***

**DEVICE\_VKBD****DEVICE\_CONSOLE****4.141.3 Function Documentation**

- 4.141.3.1 `_hidden char* libxl__abs_path ( libxl__gc * gc, char * s, const char * path )`
- 4.141.3.2 `_hidden const char* libxl__blktap_devpath ( libxl__gc * gc, const char * disk, libxl_disk_format format )`
- 4.141.3.3 `_hidden int libxl__blktap_enabled ( libxl__gc * gc )`
- 4.141.3.4 `_hidden int libxl__build_hvm ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state )`
- 4.141.3.5 `_hidden int libxl__build_post ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state, char ** vms_ents, char ** local_ents )`
- 4.141.3.6 `_hidden int libxl__build_pre ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state )`
- 4.141.3.7 `_hidden int libxl__build_pv ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state )`
- 4.141.3.8 `_hidden void* libxl__calloc ( libxl__gc * gc, size_t nmemb, size_t size )`
- 4.141.3.9 `int libxl__confirm_device_model_startup ( libxl_ctx * ctx, libxl__device_model_starting * starting )`
- 4.141.3.10 `_hidden char* libxl__cpupoolid_to_name ( libxl__gc * gc, uint32_t poolid )`
- 4.141.3.11 `_hidden int libxl__create_device_model ( libxl_ctx * ctx, libxl_device_model_info * info, libxl_device_disk * disk, int num_disks, libxl_device_nic * vifs, int num_vifs, libxl__device_model_starting ** starting_r )`
- 4.141.3.12 `int libxl__create_xenpv_qemu ( libxl_ctx * ctx, uint32_t domid, libxl_device_vfb * vfb, libxl__device_model_starting ** starting_r )`
- 4.141.3.13 `_hidden int libxl__destroy_device_model ( libxl_ctx * ctx, uint32_t domid )`
- 4.141.3.14 `_hidden char* libxl__device_backend_path ( libxl__gc * gc, libxl__device * device )`
- 4.141.3.15 `_hidden int libxl__device_del ( libxl_ctx * ctx, libxl__device * dev, int wait )`
- 4.141.3.16 `_hidden int libxl__device_destroy ( libxl_ctx * ctx, char * be_path, int force )`
- 4.141.3.17 `_hidden int libxl__device_disk_dev_number ( const char * virtpath )`
- 4.141.3.18 `_hidden char* libxl__device_disk_string_of_backend ( libxl_disk_backend backend )`
- 4.141.3.19 `_hidden char* libxl__device_disk_string_of_format ( libxl_disk_format format )`
- 4.141.3.20 `_hidden char* libxl__device_frontend_path ( libxl__gc * gc, libxl__device * device )`
- 4.141.3.21 `_hidden int libxl__device_generic_add ( libxl_ctx * ctx, libxl__device * device, char ** bents, char ** fents )`
- 4.141.3.22 `_hidden int libxl__device_pci_add ( libxl_ctx * ctx, uint32_t domid, libxl_device_pci * pcidev, int starting )`

- 4.141.3.23 `_hidden int libxl__device_physdisk_major_minor ( const char * physpath, int * major, int * minor )`
- 4.141.3.24 `_hidden int libxl__devices_destroy ( libxl_ctx * ctx, uint32_t domid, int force )`
- 4.141.3.25 `_hidden char* libxl__dirname ( libxl__gc * gc, const char * s )`
- 4.141.3.26 `int libxl__domain_build ( libxl_ctx * ctx, libxl_domain_build_info * info, uint32_t domid, libxl_domain_build_state * state )`
- 4.141.3.27 `_hidden int libxl__domain_is_hvm ( libxl_ctx * ctx, uint32_t domid )`
- 4.141.3.28 `int libxl__domain_make ( libxl_ctx * ctx, libxl_domain_create_info * info, uint32_t * domid )`
- 4.141.3.29 `_hidden int libxl__domain_restore_common ( libxl_ctx * ctx, uint32_t domid, libxl_domain_build_info * info, libxl_domain_build_state * state, int fd )`
- 4.141.3.30 `_hidden int libxl__domain_save_device_model ( libxl_ctx * ctx, uint32_t domid, int fd )`
- 4.141.3.31 `_hidden int libxl__domain_shutdown_reason ( libxl_ctx * ctx, uint32_t domid )`
- 4.141.3.32 `_hidden int libxl__domain_suspend_common ( libxl_ctx * ctx, uint32_t domid, int fd, int hvm, int live, int debug )`
- 4.141.3.33 `_hidden char* libxl__domid_to_name ( libxl__gc * gc, uint32_t domid )`
- 4.141.3.34 `_hidden int libxl__error_set ( libxl_ctx * ctx, int code )`
- 4.141.3.35 `_hidden void libxl__exec ( int stdinfd, int stdoutfd, int stderrfd, const char * arg0, char ** args )`
- 4.141.3.36 `_hidden int libxl__file_reference_map ( libxl_file_reference * f )`
- 4.141.3.37 `_hidden int libxl__file_reference_unmap ( libxl_file_reference * f )`
- 4.141.3.38 `_hidden void libxl__free_all ( libxl__gc * gc )`
- 4.141.3.39 `_hidden void _hidden void libxl__log ( libxl_ctx * ctx, xentoollog_level msglevel, int errnoval, const char * file, int line, const char * func, char * fmt, ... )`
- 4.141.3.40 `_hidden void libxl__log_child_exitstatus ( libxl__gc * gc, const char * what, pid_t pid, int status )`
- 4.141.3.41 `_hidden void libxl__logv ( libxl_ctx * ctx, xentoollog_level msglevel, int errnoval, const char * file, int line, const char * func, char * fmt, va_list al )`
- 4.141.3.42 `_hidden int libxl__need_xenpv_qemu ( libxl_ctx * ctx, int nr_consoles, libxl_device_console * consoles, int nr_vfbs, libxl_device_vfb * vfbs, int nr_disks, libxl_device_disk * disks )`
- 4.141.3.43 `_hidden int libxl__ptr_add ( libxl__gc * gc, void * ptr )`
- 4.141.3.44 `_hidden int libxl__spawn_check ( libxl_ctx * ctx, void * for_spawn )`
- 4.141.3.45 `_hidden int libxl__spawn_detach ( libxl_ctx * ctx, libxl__spawn_starting * for_spawn )`
- 4.141.3.46 `_hidden int libxl__spawn_spawn ( libxl_ctx * ctx, libxl__device_model_starting * starting, const char * what, void(*) (void *for_spawn, pid_t innerchild) intermediate_hook )`
- 4.141.3.47 `_hidden char* libxl__sprintf ( libxl__gc * gc, const char * fmt, ... )`

- 4.141.3.48 `_hidden char *_hidden char* libxl__strdup ( libxl__gc * gc, const char * c )`
- 4.141.3.49 `_hidden void libxl__userdata_destroyall ( libxl__ctx * ctx, uint32_t domid )`
- 4.141.3.50 `_hidden char* libxl__uuid2string ( libxl__gc * gc, const libxl__uuid uuid )`
- 4.141.3.51 `_hidden int libxl__wait_for_backend ( libxl__ctx * ctx, char * be_path, char * state )`
- 4.141.3.52 `_hidden int libxl__wait_for_device_model ( libxl__ctx * ctx, uint32_t domid, char * state, int(*) (libxl__ctx * ctx, uint32_t domid, const char * state, void * userdata) check_callback, void * check_callback_userdata )`
- 4.141.3.53 `_hidden char** libxl__xs_directory ( libxl__gc * gc, xs_transaction_t t, char * path, unsigned int * nb )`
- 4.141.3.54 `_hidden int _hidden char* libxl__xs_get_dompath ( libxl__gc * gc, uint32_t domid )`
- 4.141.3.55 `_hidden char** libxl__xs_kvs_of_flexarray ( libxl__gc * gc, flexarray_t * array, int length )`
- 4.141.3.56 `_hidden char* libxl__xs_read ( libxl__gc * gc, xs_transaction_t t, char * path )`
- 4.141.3.57 `_hidden int libxl__xs_write ( libxl__gc * gc, xs_transaction_t t, char * path, char * fmt, ... )`
- 4.141.3.58 `_hidden int libxl__xs_writew ( libxl__gc * gc, xs_transaction_t t, char * dir, char ** kvs )`
- 4.141.3.59 `_hidden void* libxl__zalloc ( libxl__gc * gc, int bytes )`

## 4.142 xen/tools/libxl/libxl\_utils.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <xs.h>
#include <xenctrl.h>
#include <ctype.h>
#include <errno.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
#include <assert.h>
#include "libxl_utils.h"
#include "libxl_internal.h"
```

### Data Structures

- struct [schedid\\_name](#)

### Macros

- #define [READ\\_WRITE\\_EXACTLY](#)(rw, zero\_is\_eof, constdata)
- #define [QEMU\\_VERSION\\_STR](#) "QEMU emulator version "

## Functions

- const char \* [libxl\\_basename](#) (const char \*name)
- unsigned long [libxl\\_get\\_required\\_shadow\\_memory](#) (unsigned long maxmem\_kb, unsigned int smp\_cpus)
- char \* [libxl\\_domid\\_to\\_name](#) (libxl\_ctx \*ctx, uint32\_t domid)
- char \* [libxl\\_\\_domid\\_to\\_name](#) (libxl\_\_gc \*gc, uint32\_t domid)
- int [libxl\\_name\\_to\\_domid](#) (libxl\_ctx \*ctx, const char \*name, uint32\_t \*domid)
- char \* [libxl\\_cpupoolid\\_to\\_name](#) (libxl\_ctx \*ctx, uint32\_t poolid)
- char \* [libxl\\_\\_cpupoolid\\_to\\_name](#) (libxl\_\_gc \*gc, uint32\_t poolid)
- int [libxl\\_name\\_to\\_cpupoolid](#) (libxl\_ctx \*ctx, const char \*name, uint32\_t \*poolid)
- int [libxl\\_name\\_to\\_schedid](#) (libxl\_ctx \*ctx, const char \*name)
- char \* [libxl\\_schedid\\_to\\_name](#) (libxl\_ctx \*ctx, int schedid)
- int [libxl\\_get\\_stubdom\\_id](#) (libxl\_ctx \*ctx, int guest\_domid)
- int [libxl\\_is\\_stubdom](#) (libxl\_ctx \*ctx, uint32\_t domid, uint32\_t \*target\_domid)
- int [libxl\\_create\\_logfile](#) (libxl\_ctx \*ctx, char \*name, char \*\*full\_name)
- int [libxl\\_string\\_to\\_backend](#) (libxl\_ctx \*ctx, char \*s, [libxl\\_disk\\_backend](#) \*backend)
- int [libxl\\_read\\_file\\_contents](#) (libxl\_ctx \*ctx, const char \*filename, void \*\*data\_r, int \*datalen\_r)
- int [libxl\\_ctx\\_postfork](#) (libxl\_ctx \*ctx)
- pid\_t [libxl\\_fork](#) (libxl\_ctx \*ctx)
- int [libxl\\_pipe](#) (libxl\_ctx \*ctx, int pipes[2])
- int [libxl\\_mac\\_to\\_device\\_nic](#) (libxl\_ctx \*ctx, uint32\_t domid, const char \*mac, [libxl\\_device\\_nic](#) \*nic)
- int [libxl\\_devid\\_to\\_device\\_nic](#) (libxl\_ctx \*ctx, uint32\_t domid, const char \*devid, [libxl\\_device\\_nic](#) \*nic)
- int [libxl\\_devid\\_to\\_device\\_disk](#) (libxl\_ctx \*ctx, uint32\_t domid, const char \*devid, [libxl\\_device\\_disk](#) \*disk)
- int [libxl\\_devid\\_to\\_device\\_net2](#) (libxl\_ctx \*ctx, uint32\_t domid, const char \*devid, [libxl\\_device\\_net2](#) \*net2)
- int [libxl\\_strtomac](#) (const char \*mac\_s, uint8\_t \*mac)
- int [libxl\\_check\\_device\\_model\\_version](#) (libxl\_ctx \*ctx, char \*path)
- int [libxl\\_cpumap\\_alloc](#) (libxl\_ctx \*ctx, [libxl\\_cpumap](#) \*cpumap)
- void [libxl\\_cpumap\\_destroy](#) (libxl\_cpumap \*map)
- int [libxl\\_cpumap\\_test](#) (libxl\_cpumap \*cpumap, int cpu)
- void [libxl\\_cpumap\\_set](#) (libxl\_cpumap \*cpumap, int cpu)
- void [libxl\\_cpumap\\_reset](#) (libxl\_cpumap \*cpumap, int cpu)
- int [libxl\\_cpumap\\_alloc](#) (libxl\_ctx \*ctx, [libxl\\_cpumap](#) \*cpumap)
- void [libxl\\_cpumap\\_destroy](#) (libxl\_cpumap \*array)
- int [libxl\\_get\\_max\\_cpus](#) (libxl\_ctx \*ctx)

### 4.142.1 Macro Definition Documentation

4.142.1.1 `#define QEMU_VERSION_STR "QEMU emulator version "`

4.142.1.2 `#define READ_WRITE_EXACTLY( rw, zero_is_eof, constdata )`

**Value:**

```
\
int libxl_##rw##_exactly(libxl_ctx *ctx, int fd,
                        constdata void *data, ssize_t sz,
                        const char *filename, const char *what) {
    ssize_t got;

    while (sz > 0) {
        got = rw(fd, data, sz);
        if (got == -1) {
            if (errno == EINTR) continue;
            if (!ctx) return errno;
            LIBXL__LOG_ERRNO(ctx, LIBXL__LOG_ERROR, "failed to " #rw "
                %s%s", \
                        what?what:"", what?" from ":"", filename);
            return errno;
        }
        if (got == 0) {
            if (!ctx) return EPROTO;
        }
    }
}
```

```

        LIBXL__LOG(ctx, LIBXL__LOG_ERROR,
            zero_is_eof
            ? "file/stream truncated reading %s%s%s"
            : "file/stream write returned 0! writing %s%s%s",
            what?what:"", what?" from ":"", filename);
        return EPROTO;
    }
    sz -= got;
    data = (char*)data + got;
}
return 0;
}

```

## 4.142.2 Function Documentation

- 4.142.2.1 `char* libxl__cpupoolid_to_name ( libxl__gc * gc, uint32_t poolid )`
- 4.142.2.2 `char* libxl__domid_to_name ( libxl__gc * gc, uint32_t domid )`
- 4.142.2.3 `const char* libxl_basename ( const char * name )`
- 4.142.2.4 `int libxl_check_device_model_version ( libxl_ctx * ctx, char * path )`
- 4.142.2.5 `int libxl_cpumap_alloc ( libxl_ctx * ctx, libxl_cpumap * cpumap )`
- 4.142.2.6 `void libxl_cpumap_destroy ( libxl_cpumap * array )`
- 4.142.2.7 `int libxl_cpumap_alloc ( libxl_ctx * ctx, libxl_cpumap * cpumap )`
- 4.142.2.8 `void libxl_cpumap_destroy ( libxl_cpumap * map )`
- 4.142.2.9 `void libxl_cpumap_reset ( libxl_cpumap * cpumap, int cpu )`
- 4.142.2.10 `void libxl_cpumap_set ( libxl_cpumap * cpumap, int cpu )`
- 4.142.2.11 `int libxl_cpumap_test ( libxl_cpumap * cpumap, int cpu )`
- 4.142.2.12 `char* libxl_cpupoolid_to_name ( libxl_ctx * ctx, uint32_t poolid )`
- 4.142.2.13 `int libxl_create_logfile ( libxl_ctx * ctx, char * name, char ** full_name )`
- 4.142.2.14 `int libxl_ctx_postfork ( libxl_ctx * ctx )`
- 4.142.2.15 `int libxl_devid_to_device_disk ( libxl_ctx * ctx, uint32_t domid, const char * devid, libxl_device_disk * disk )`
- 4.142.2.16 `int libxl_devid_to_device_net2 ( libxl_ctx * ctx, uint32_t domid, const char * devid, libxl_device_net2 * net2 )`
- 4.142.2.17 `int libxl_devid_to_device_nic ( libxl_ctx * ctx, uint32_t domid, const char * devid, libxl_device_nic * nic )`
- 4.142.2.18 `char* libxl_domid_to_name ( libxl_ctx * ctx, uint32_t domid )`
- 4.142.2.19 `pid_t libxl_fork ( libxl_ctx * ctx )`
- 4.142.2.20 `int libxl_get_max_cpus ( libxl_ctx * ctx )`
- 4.142.2.21 `unsigned long libxl_get_required_shadow_memory ( unsigned long maxmem_kb, unsigned int smp_cpus )`
- 4.142.2.22 `int libxl_get_stubdom_id ( libxl_ctx * ctx, int guest_domid )`



```

4.142.2.23  int libxl_is_stubdom ( libxl_ctx * ctx, uint32_t domid, uint32_t * target_domid )

4.142.2.24  int libxl_mac_to_device_nic ( libxl_ctx * ctx, uint32_t domid, const char * mac, libxl_device_nic * nic )

4.142.2.25  int libxl_name_to_cpupoolid ( libxl_ctx * ctx, const char * name, uint32_t * poolid )

4.142.2.26  int libxl_name_to_domid ( libxl_ctx * ctx, const char * name, uint32_t * domid )

4.142.2.27  int libxl_name_to_schedid ( libxl_ctx * ctx, const char * name )

4.142.2.28  int libxl_pipe ( libxl_ctx * ctx, int pipes[2] )

4.142.2.29  int libxl_read_file_contents ( libxl_ctx * ctx, const char * filename, void ** data_r, int * datalen_r )

4.142.2.30  char* libxl_schedid_to_name ( libxl_ctx * ctx, int schedid )

4.142.2.31  int libxl_string_to_backend ( libxl_ctx * ctx, char * s, libxl_disk_backend * backend )

4.142.2.32  int libxl_strtomac ( const char * mac_s, uint8_t * mac )

```

#### 4.143 xen/tools/libxl/libxl\_uuid.h File Reference

## Macros

- ```
• #define LIBXL_UUID_FMT "%02hhx%02hhx%02hhx%02hhx-%02hhx%02hhx-%02hhx%02hhx-%02hhx%02hhx%02hhx%02hhx%02hhx%02hhx%02hhx"
• #define LIBXL_UUID_BYTES(uuid)
```

#### 4.143.1 Macro Definition Documentation

#### 4.143.1.1 #define LIBXL\_UUID\_BYTES( *uuid* )

**Value:**

```
uuid[0], uuid[1], uuid[2], uuid[3], \
    uuid[4], uuid[5], uuid[6], uuid[7], \
    uuid[8], uuid[9], uuid[10], uuid[11], \
    uuid[12], uuid[13], uuid[14], uuid[15]
```

```
4.143.1.2 #define LIBXL_UUID_FMT "%02hhx%02hhx%02hhx%02hhx-%02hhx%02hhx-%02hhx%02hhx%02hhx%02hhx-  
%02hhx%02hhx%02hhx%02hhx%02hhx%02hhx"
```

#### 4.144 xen/tools/libxl/libxlu\_cfg\_y.c File Reference

```
#include "libxlu_cfg_i.h"
#include "libxlu_cfg_l.h"
```

## Data Structures

- union YYSTYPE
- struct YYLTYPE
- union yyallocc

## Macros

- `#define YYBISON 1`
- `#define YYBISON_VERSION "2.5"`
- `#define YYSKELETON_NAME "yacc.c"`
- `#define YYPURE 1`
- `#define YYPUSH 0`
- `#define YYPULL 1`
- `#define YYLSP_NEEDED 1`
- `#define yyparse xlu__cfg_yyparse`
- `#define yylex xlu__cfg_yylex`
- `#define yyerror xlu__cfg_yyerror`
- `#define yylval xlu__cfg_yylval`
- `#define yychar xlu__cfg_yychar`
- `#define yydebug xlu__cfg_yydebug`
- `#define yynerrs xlu__cfg_yynerrs`
- `#define yylloc xlu__cfg_yylloc`
- `#define YYLEX_PARAM ctx->scanner`
- `#define YYDEBUG 0`
- `#define YYERROR_VERBOSE 1`
- `#define YYTOKEN_TABLE 0`
- `#define YYTOKENTYPE`
- `#define YYSTYPE_IS_TRIVIAL 1`
- `#define yystype YYSTYPE /* obsolescent; will be withdrawn */`
- `#define YYSTYPE_IS_DECLARED 1`
- `#define yytype YYLTYPE /* obsolescent; will be withdrawn */`
- `#define YYLTYPE_IS_DECLARED 1`
- `#define YYLTYPE_IS_TRIVIAL 1`
- `#define YYSIZE_T unsigned int`
- `#define YYSIZE_MAXIMUM ((YYSIZE_T) -1)`
- `#define YY_(msgid) msgid`
- `#define YYUSE(e) ((void) (e))`
- `#define YYID(n) (n)`
- `#define YYSTACK_ALLOC YYMALLOC`
- `#define YYSTACK_FREE YYPFREE`
- `#define YYSTACK_ALLOC_MAXIMUM YYSIZE_MAXIMUM`
- `#define YYMALLOC malloc`
- `#define YYPFREE free`
- `#define YYSTACK_GAP_MAXIMUM (sizeof (union yyalloc) - 1)`
- `#define YYSTACK_BYTES(N)`
- `#define YYCOPY_NEEDED 1`
- `#define YYSTACK_RELOCATE(Stack_alloc, Stack)`
- `#define YYCOPY(To, From, Count)`
- `#define YYFINAL 2`
- `#define YYLAST 23`
- `#define YYNTOKENS 12`
- `#define YYNNTS 10`
- `#define YYNRULES 20`
- `#define YYNSTATES 29`
- `#define YYUNDEFTOK 2`
- `#define YYMAXUTOK 261`
- `#define YYTRANSLATE(YYX) ((unsigned int) (YYX) <= YYMAXUTOK ? yytranslate[YYX] : YYUNDEFTOK)`
- `#define YYPACT_NINF -17`
- `#define YYTABLE_NINF -1`
- `#define yypact_value_is_default(yystate) ((yystate) == (-17))`

- #define `yytable_value_is_error`(yytable\_value) `YYID` (0)
- #define `yyerrok` (yyerrstatus = 0)
- #define `yyclearin` (yychar = `YYEMPTY`)
- #define `YYEMPTY` (-2)
- #define `YYEOF` 0
- #define `YYACCEPT` goto yyacceptlab
- #define `YYABORT` goto yyabortlab
- #define `YYERROR` goto yyerrorlab
- #define `YYFAIL` goto yyerrlab
- #define `YYRECOVERING`() (!yyerrstatus)
- #define `YYBACKUP`(Token, Value)
- #define `YYTERROR` 1
- #define `YYERRCODE` 256
- #define `YYRHSLOC`(Rhs, K) ((Rhs)[K])
- #define `YYLLOC_DEFAULT`(Current, Rhs, N)
- #define `YY_LOCATION_PRINT`(File, Loc)
- #define `YYLEX` yylex (&yylval, &yylloc, `YYLEX_PARAM`)
- #define `YYDPRINTF`(Args)
- #define `YY_SYMBOL_PRINT`(Title, Type, Value, Location)
- #define `YY_STACK_PRINT`(Bottom, Top)
- #define `YY_REDUCE_PRINT`(Rule)
- #define `YYINITDEPTH` 200
- #define `YYMAXDEPTH` 10000
- #define `YYCASE_`(N, S)
- #define `YYPOPSTACK`(N) (yyvsp -= (N), yyssp -= (N), yylsp -= (N))
- #define `YYSYNTAX_ERROR`

## Typedefs

- typedef union `YYSTYPE` `YYSTYPE`
- typedef struct `YYLTYPE` `YYLTYPE`
- typedef unsigned char `yytype_uint8`
- typedef short int `yytype_int8`
- typedef unsigned short int `yytype_uint16`
- typedef short int `yytype_int16`

## Enumerations

- enum `yytokentype` {  
`IDENT` = 258, `STRING` = 259, `NUMBER` = 260, `NEWLINE` = 261,  
`IDENT` = 258, `STRING` = 259, `NUMBER` = 260, `NEWLINE` = 261 }

## Functions

- int `yyparse` ()
- int `yyparse` (CfgParseContext \*ctx)

#### 4.144.1 Macro Definition Documentation

4.144.1.1 `#define YY_( msgid ) msgid`

4.144.1.2 `#define YY_LOCATION_PRINT( File, Loc )`

**Value:**

```
fprintf (File, "%d.%d-%d.%d", \
        (Loc).first_line, (Loc).first_column, \
        (Loc).last_line,  (Loc).last_column)
```

4.144.1.3 `#define YY_REDUCE_PRINT( Rule )`

4.144.1.4 `#define YY_STACK_PRINT( Bottom, Top )`

4.144.1.5 `#define YY_SYMBOL_PRINT( Title, Type, Value, Location )`

4.144.1.6 `#define YYABORT goto yyabortlab`

4.144.1.7 `#define YYACCEPT goto yyacceptlab`

4.144.1.8 `#define YYBACKUP( Token, Value )`

**Value:**

```
do
  if (yychar == YYEMPTY && yylen == 1)
  {
    yychar = (Token);
    yylval = (Value);
    YYPOPSTACK (1);
    goto yybackup;
  }
  else
  {
    yyerror (&yylloc, ctx, YY_("syntax error: cannot back up")); \
    YYERROR;
  }
while (YYID (0))
```

4.144.1.9 `#define YYBISON 1`

4.144.1.10 `#define YYBISON_VERSION "2.5"`

4.144.1.11 `#define YYCASE_( N, S )`

**Value:**

```
case N:
  yyformat = S;
  break
```

4.144.1.12 `#define yychar xlu__cfg_yychar`

4.144.1.13 `#define yyclearin (yychar = YYEMPTY)`

4.144.1.14 `#define YYCOPY( To, From, Count )`

**Value:**

```
do
{
    YYSIZE_T yyi;
    for (yyi = 0; yyi < (Count); yyi++)
        (To)[yyi] = (From)[yyi];
}
while (YYID (0))
```

4.144.1.15 **#define YYCOPY\_NEEDED 1**

4.144.1.16 **#define yydebug xlu\_\_cfg\_yydebug**

4.144.1.17 **#define YYDEBUG 0**

4.144.1.18 **#define YYDPRINTF( Args )**

4.144.1.19 **#define YYEMPTY (-2)**

4.144.1.20 **#define YYEOF 0**

4.144.1.21 **#define YYERRCODE 256**

4.144.1.22 **#define yyerrok (yyerrstatus = 0)**

4.144.1.23 **#define yyerror xlu\_\_cfg\_yyerror**

4.144.1.24 **#define YYERROR goto yyerrorlab**

4.144.1.25 **#define YYERROR\_VERBOSE 1**

4.144.1.26 **#define YYFAIL goto yyerrlab**

4.144.1.27 **#define YYFINAL 2**

4.144.1.28 **#define YYFREE free**

4.144.1.29 **#define YYID( n ) (n)**

4.144.1.30 **#define YYINITDEPTH 200**

4.144.1.31 **#define YYLAST 23**

4.144.1.32 **#define yylex xlu\_\_cfg\_yylex**

4.144.1.33 **#define YYLEX yylex (&yylval, &yylloc, YYLEX\_PARAM)**

4.144.1.34 **#define YYLEX\_PARAM ctx->scanner**

4.144.1.35 **#define yylloc xlu\_\_cfg\_yylloc**

4.144.1.36 **#define YYLLOC\_DEFAULT( Current, Rhs, N )**

**Value:**

```
do
{
    if (YYID (N))
    {
        (Current).first_line = YYRHSLOC (Rhs, 1).first_line;
        (Current).first_column = YYRHSLOC (Rhs, 1).first_column;
    }
}
```

```

        (Current).last_line    = YYRHSLOC (Rhs, N).last_line;    \
        (Current).last_column  = YYRHSLOC (Rhs, N).last_column;  \
    } else
    {
        (Current).first_line   = (Current).last_line   =      \
        YYRHSLOC (Rhs, 0).last_line;                          \
        (Current).first_column = (Current).last_column =      \
        YYRHSLOC (Rhs, 0).last_column;                          \
    }
    while (YYID (0))

```

4.144.1.37 **#define YYLSP\_NEEDED 1**

4.144.1.38 **#define yytype YYLTYPE /\* obsolescent; will be withdrawn \*/**

4.144.1.39 **#define YYLTYPE\_IS\_DECLARED 1**

4.144.1.40 **#define YYLTYPE\_IS\_TRIVIAL 1**

4.144.1.41 **#define yyval xlu\_\_cfg\_yylval**

4.144.1.42 **#define YYMALLOC malloc**

4.144.1.43 **#define YYMAXDEPTH 10000**

4.144.1.44 **#define YYMAXUTOK 261**

4.144.1.45 **#define yynerrs xlu\_\_cfg\_yynerrs**

4.144.1.46 **#define YYNNTS 10**

4.144.1.47 **#define YYNRULES 20**

4.144.1.48 **#define YYNSTATES 29**

4.144.1.49 **#define YYNTOKENS 12**

4.144.1.50 **#define YYPACT\_NINF -17**

4.144.1.51 **#define yypact\_value\_is\_default( yystate ) ((yystate) == (-17))**

4.144.1.52 **#define yyparse xlu\_\_cfg\_yyparse**

4.144.1.53 **#define YPOPSTACK( N ) (yyvsp -= (N), yyssp -= (N), yylsp -= (N))**

4.144.1.54 **#define YYPULL 1**

4.144.1.55 **#define YYPURE 1**

4.144.1.56 **#define YYPUSH 0**

4.144.1.57 **#define YYRECOVERING( ) (!yerrstatus)**

4.144.1.58 **#define YYRHSLOC( Rhs, K ) ((Rhs)[K])**

4.144.1.59 **#define YYSIZE\_MAXIMUM ((YYSIZE\_T) -1)**

4.144.1.60 `#define YYSIZE_T unsigned int`

4.144.1.61 `#define YYSKELETON_NAME "yacc.c"`

4.144.1.62 `#define YYSTACK_ALLOC YYMALLOC`

4.144.1.63 `#define YYSTACK_ALLOC_MAXIMUM YYSIZE_MAXIMUM`

4.144.1.64 `#define YYSTACK_BYTES( N )`

**Value:**

```
((N) * (sizeof (yytype_int16) + sizeof (YYSTYPE) + sizeof (
    YYLTYPE)) \
    + 2 * YYSTACK_GAP_MAXIMUM)
```

4.144.1.65 `#define YYSTACK_FREE YYFREE`

4.144.1.66 `#define YYSTACK_GAP_MAXIMUM (sizeof (union yyallo) - 1)`

4.144.1.67 `#define YYSTACK_RELOCATE( Stack_alloc, Stack )`

**Value:**

```
do
{
    YYSIZE_T yynewbytes;
    YYCOPY (&yyptr->Stack_alloc, Stack, yysize);
    Stack = &yyptr->Stack_alloc;
    yynewbytes = yystacksize * sizeof (*Stack) + YYSTACK_GAP_MAXIMUM;
    yypr += yynewbytes / sizeof (*yyptr);
}
while (YYID (0))
```

4.144.1.68 `#define yystype YYSTYPE /* obsolescent; will be withdrawn */`

4.144.1.69 `#define YYSTYPE_IS_DECLARED 1`

4.144.1.70 `#define YYSTYPE_IS_TRIVIAL 1`

4.144.1.71 `#define YYSYNTAX_ERROR`

**Value:**

```
yysyntax_error (&yymsg_alloc, &yymsg, \
    yyssp, yytoken)
```

4.144.1.72 `#define YYTABLE_NINF -1`

4.144.1.73 `#define yytable_value_is_error( yytable_value ) YYID (0)`

4.144.1.74 `#define YYTERROR 1`

4.144.1.75 `#define YYTOKEN_TABLE 0`

4.144.1.76 `#define YYTOKENTYPE`

4.144.1.77 `#define YYTRANSLATE( YYX ) ((unsigned int) (YYX) <= YYMAXUTOK ? yytranslate[YYX] : YYUNDEFTOK)`

4.144.1.78 `#define YYUNDEFTOK 2`

4.144.1.79 `#define YYUSE( e ) ((void) (e))`

#### 4.144.2 Typedef Documentation

4.144.2.1 `typedef struct YYLTYPE YYLTYPE`

4.144.2.2 `typedef union YYSTYPE YYSTYPE`

4.144.2.3 `typedef short int yytype_int16`

4.144.2.4 `typedef short int yytype_int8`

4.144.2.5 `typedef unsigned short int yytype_uint16`

4.144.2.6 `typedef unsigned char yytype_uint8`

#### 4.144.3 Enumeration Type Documentation

4.144.3.1 `enum yytokentype`

Enumerator

***IDENT***

***STRING***

***NUMBER***

***NEWLINE***

***IDENT***

***STRING***

***NUMBER***

***NEWLINE***

#### 4.144.4 Function Documentation

4.144.4.1 `int yyparse ( )`

4.144.4.2 `int yyparse ( CfgParseContext * ctx )`

### 4.145 xen/tools/libxl/libxlu\_cfg\_y.h File Reference

#### Data Structures

- union [YYSTYPE](#)
- struct [YYLTYPE](#)

#### Macros

- `#define YYSTYPE_IS_TRIVIAL 1`
- `#define YYSTYPE YYSSTYPE /* obsolescent; will be withdrawn */`
- `#define YYSTYPE_IS_DECLARED 1`



- `#define yytype YYLTYPE /* obsolescent; will be withdrawn */`
- `#define YYLTYPE_IS_DECLARED 1`
- `#define YYLTYPE_IS_TRIVIAL 1`

## Typedefs

- `typedef union YYSTYPE YYSTYPE`
- `typedef struct YYLTYPE YYLTYPE`

## Enumerations

- `enum yytokentype {`  
    `IDENT = 258, STRING = 259, NUMBER = 260, NEWLINE = 261,`  
    `IDENT = 258, STRING = 259, NUMBER = 260, NEWLINE = 261 }`

### 4.145.1 Macro Definition Documentation

4.145.1.1 `#define yytype YYLTYPE /* obsolescent; will be withdrawn */`

4.145.1.2 `#define YYLTYPE_IS_DECLARED 1`

4.145.1.3 `#define YYLTYPE_IS_TRIVIAL 1`

4.145.1.4 `#define YYSTYPE YYSTYPE /* obsolescent; will be withdrawn */`

4.145.1.5 `#define YYSTYPE_IS_DECLARED 1`

4.145.1.6 `#define YYSTYPE_IS_TRIVIAL 1`

### 4.145.2 Typedef Documentation

4.145.2.1 `typedef struct YYLTYPE YYLTYPE`

4.145.2.2 `typedef union YYSTYPE YYSTYPE`

### 4.145.3 Enumeration Type Documentation

4.145.3.1 `enum yytokentype`

Enumerator

***IDENT***

***STRING***

***NUMBER***

***NEWLINE***

***IDENT***

***STRING***

***NUMBER***

***NEWLINE***

## 4.146 xen/tools/libxl/xl\_cmdimpl.c File Reference

```
#include "libxl_osdeps.h"
#include <stdio.h>
#include <assert.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <time.h>
#include <getopt.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/time.h>
#include <fcntl.h>
#include <signal.h>
#include <sys/socket.h>
#include <sys/select.h>
#include <sys/utsname.h>
#include <xenctrl.h>
#include <ctype.h>
#include <inttypes.h>
#include "libxl.h"
#include "libxl_utils.h"
#include "libxlutil.h"
#include "xl.h"
```

### Data Structures

- struct [save\\_file\\_header](#)
- struct [domain\\_create](#)

### Macros

- #define [CHK\\_ERRNO](#)(call)
- #define [MUST](#)(call)
- #define [SAVEFILE\\_BYTEORDER\\_VALUE](#) ((uint32\_t)0x01020304UL)
- #define [LOG](#)(\_f, \_a...) dolog(\_\_FILE\_\_, \_\_LINE\_\_, \_\_func\_\_, \_f "\n", ##\_a)
- #define [DSTATE\\_INITIAL](#) 0
- #define [DSTATE\\_TAP](#) 1
- #define [DSTATE\\_PHYSPATH](#) 2
- #define [DSTATE\\_VIRTPATH](#) 3
- #define [DSTATE\\_VIRTTYPE](#) 4
- #define [DSTATE\\_RW](#) 5
- #define [DSTATE\\_TERMINAL](#) 6
- #define [OPTDATA\\_LEFT](#) (hdr.optional\_data\_len - (optdata\_here - optdata\_begin))
- #define [WITH\\_OPTDATA](#)(amt, body)
- #define [ADD\\_OPTDATA](#)(ptr, len)

### Functions

- void [help](#) (const char \*command)
- int [main\\_memmax](#) (int argc, char \*\*argv)
- int [main\\_memset](#) (int argc, char \*\*argv)

- int [main\\_cd\\_eject](#) (int argc, char \*\*argv)
- int [main\\_cd\\_insert](#) (int argc, char \*\*argv)
- int [main\\_console](#) (int argc, char \*\*argv)
- int [main\\_vncviewer](#) (int argc, char \*\*argv)
- int [main\\_pclist\\_assignable](#) (int argc, char \*\*argv)
- int [main\\_pclist](#) (int argc, char \*\*argv)
- int [main\\_pcidetach](#) (int argc, char \*\*argv)
- int [main\\_pciattach](#) (int argc, char \*\*argv)
- int [main\\_restore](#) (int argc, char \*\*argv)
- int [main\\_migrate\\_receive](#) (int argc, char \*\*argv)
- int [main\\_save](#) (int argc, char \*\*argv)
- int [main\\_migrate](#) (int argc, char \*\*argv)
- int [main\\_dump\\_core](#) (int argc, char \*\*argv)
- int [main\\_pause](#) (int argc, char \*\*argv)
- int [main\\_unpause](#) (int argc, char \*\*argv)
- int [main\\_destroy](#) (int argc, char \*\*argv)
- int [main\\_shutdown](#) (int argc, char \*\*argv)
- int [main\\_reboot](#) (int argc, char \*\*argv)
- int [main\\_list](#) (int argc, char \*\*argv)
- int [main\\_list\\_vm](#) (int argc, char \*\*argv)
- int [main\\_create](#) (int argc, char \*\*argv)
- int [main\\_button\\_press](#) (int argc, char \*\*argv)
- int [main\\_vcpulist](#) (int argc, char \*\*argv)
- int [main\\_vcpupin](#) (int argc, char \*\*argv)
- int [main\\_vcpuset](#) (int argc, char \*\*argv)
- int [main\\_info](#) (int argc, char \*\*argv)
- int [main\\_sched\\_credit](#) (int argc, char \*\*argv)
- int [main\\_domid](#) (int argc, char \*\*argv)
- int [main\\_domname](#) (int argc, char \*\*argv)
- int [main\\_rename](#) (int argc, char \*\*argv)
- int [main\\_trigger](#) (int argc, char \*\*argv)
- int [main\\_sysrq](#) (int argc, char \*\*argv)
- int [main\\_debug\\_keys](#) (int argc, char \*\*argv)
- int [main\\_dmesg](#) (int argc, char \*\*argv)
- int [main\\_top](#) (int argc, char \*\*argv)
- int [main\\_networkattach](#) (int argc, char \*\*argv)
- int [main\\_networklist](#) (int argc, char \*\*argv)
- int [main\\_networkdetach](#) (int argc, char \*\*argv)
- int [main\\_blockattach](#) (int argc, char \*\*argv)
- int [main\\_blocklist](#) (int argc, char \*\*argv)
- int [main\\_blockdetach](#) (int argc, char \*\*argv)
- int [main\\_network2attach](#) (int argc, char \*\*argv)
- int [main\\_network2list](#) (int argc, char \*\*argv)
- int [main\\_network2detach](#) (int argc, char \*\*argv)
- int [main\\_uptime](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_list](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_freeze](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_destroy](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_thaw](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_set](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_shared\\_auth](#) (int argc, char \*\*argv)
- int [main\\_tmem\\_freeable](#) (int argc, char \*\*argv)
- int [main\\_cpupoolcreate](#) (int argc, char \*\*argv)
- int [main\\_cpupoollist](#) (int argc, char \*\*argv)
- int [main\\_cpupooldestroy](#) (int argc, char \*\*argv)

- int [main\\_cpupoolrename](#) (int argc, char \*\*argv)
- int [main\\_cpupoolcpuadd](#) (int argc, char \*\*argv)
- int [main\\_cpupoolcpuremove](#) (int argc, char \*\*argv)
- int [main\\_cpupoolmigrate](#) (int argc, char \*\*argv)
- int [main\\_cpupoolnumasplit](#) (int argc, char \*\*argv)

## Variables

- int [logfile](#) = 2
- [libxl\\_ctx](#) ctx

## 4.146.1 Macro Definition Documentation

### 4.146.1.1 #define ADD\_OPTDATA( ptr, len )

#### Value:

```
{
    if ((len)) {
        hdr.optional_data_len += (len);
        optdata_begin = xrealloc(optdata_begin, hdr.optional_data_len);
        memcpy(optdata_begin + hdr.optional_data_len - (len),
               (ptr), (len));
    }
}
```

### 4.146.1.2 #define CHK\_ERRNO( call )

#### Value:

```
{
    int chk_errno = (call);
    if (chk_errno < 0) {
        fprintf(stderr, "xl: fatal error: %s:%d: %s: %s\n",
               __FILE__, __LINE__, strerror(chk_errno), #call);
        exit(-ERROR_FAIL);
    }
}
```

### 4.146.1.3 #define DSTATE\_INITIAL 0

### 4.146.1.4 #define DSTATE\_PHYSPATH 2

### 4.146.1.5 #define DSTATE\_RW 5

### 4.146.1.6 #define DSTATE\_TAP 1

### 4.146.1.7 #define DSTATE\_TERMINAL 6

### 4.146.1.8 #define DSTATE\_VIRTPATH 3

### 4.146.1.9 #define DSTATE\_VIRTTYPE 4

### 4.146.1.10 #define LOG( \_f, \_a... ) dolog(\_\_FILE\_\_, \_\_LINE\_\_, \_\_func\_\_, \_f "\n", ##\_a)

### 4.146.1.11 #define MUST( call )

#### Value:

```

({
    int must_rc = (call);
    if (must_rc < 0) {
        fprintf(stderr, "xl: fatal error: %s:%d, rc=%d: %s\n",
            __FILE__, __LINE__, must_rc, #call);
        exit(-must_rc);
    }
})

```

4.146.1.12 `#define OPTDATA_LEFT (hdr.optional_data_len - (optdata_here - optdata_begin))`

4.146.1.13 `#define SAVEFILE_BYTEORDER_VALUE ((uint32_t)0x01020304UL)`

4.146.1.14 `#define WITH_OPTDATA( amt, body )`

**Value:**

```

if (OPTDATA_LEFT < (amt)) {
    fprintf(stderr, "Savefile truncated.\n");
    return ERROR_INVALID;
} else {
    body;
    optdata_here += (amt);
}

```

## 4.146.2 Function Documentation

4.146.2.1 `void help ( const char * command )`

4.146.2.2 `int main_blockattach ( int argc, char ** argv )`

4.146.2.3 `int main_blockdetach ( int argc, char ** argv )`

4.146.2.4 `int main_blocklist ( int argc, char ** argv )`

4.146.2.5 `int main_button_press ( int argc, char ** argv )`

4.146.2.6 `int main_cd_eject ( int argc, char ** argv )`

4.146.2.7 `int main_cd_insert ( int argc, char ** argv )`

4.146.2.8 `int main_console ( int argc, char ** argv )`

4.146.2.9 `int main_cpupoolcpuadd ( int argc, char ** argv )`

4.146.2.10 `int main_cpupoolcpuremove ( int argc, char ** argv )`

4.146.2.11 `int main_cpupoolcreate ( int argc, char ** argv )`

4.146.2.12 `int main_cpupooldestroy ( int argc, char ** argv )`

4.146.2.13 `int main_cpupoollist ( int argc, char ** argv )`

4.146.2.14 `int main_cpupoolmigrate ( int argc, char ** argv )`

4.146.2.15 `int main_cpupoolnumasplit ( int argc, char ** argv )`

4.146.2.16 `int main_cpupoolrename ( int argc, char ** argv )`

4.146.2.17 `int main_create ( int argc, char ** argv )`

4.146.2.18 `int main_debug_keys ( int argc, char ** argv )`

4.146.2.19 `int main_destroy ( int argc, char ** argv )`

4.146.2.20 `int main_dmesg ( int argc, char ** argv )`

4.146.2.21 `int main_domid ( int argc, char ** argv )`

4.146.2.22 `int main_domname ( int argc, char ** argv )`

4.146.2.23 `int main_dump_core ( int argc, char ** argv )`

4.146.2.24 `int main_info ( int argc, char ** argv )`

4.146.2.25 `int main_list ( int argc, char ** argv )`

4.146.2.26 `int main_list_vm ( int argc, char ** argv )`

4.146.2.27 `int main_memmax ( int argc, char ** argv )`

4.146.2.28 `int main_memset ( int argc, char ** argv )`

4.146.2.29 `int main_migrate ( int argc, char ** argv )`

4.146.2.30 `int main_migrate_receive ( int argc, char ** argv )`

4.146.2.31 `int main_network2attach ( int argc, char ** argv )`

4.146.2.32 `int main_network2detach ( int argc, char ** argv )`

4.146.2.33 `int main_network2list ( int argc, char ** argv )`

4.146.2.34 `int main_networkattach ( int argc, char ** argv )`

4.146.2.35 `int main_networkdetach ( int argc, char ** argv )`

4.146.2.36 `int main_networklist ( int argc, char ** argv )`

4.146.2.37 `int main_pause ( int argc, char ** argv )`

4.146.2.38 `int main_pciattach ( int argc, char ** argv )`

4.146.2.39 `int main_pcidetach ( int argc, char ** argv )`

4.146.2.40 `int main_pclist ( int argc, char ** argv )`

4.146.2.41 `int main_pclist_assignable ( int argc, char ** argv )`

4.146.2.42 `int main_reboot ( int argc, char ** argv )`

4.146.2.43 `int main_rename ( int argc, char ** argv )`

4.146.2.44 `int main_restore ( int argc, char ** argv )`

- 4.146.2.45 `int main_save ( int argc, char ** argv )`
- 4.146.2.46 `int main_sched_credit ( int argc, char ** argv )`
- 4.146.2.47 `int main_shutdown ( int argc, char ** argv )`
- 4.146.2.48 `int main_sysrq ( int argc, char ** argv )`
- 4.146.2.49 `int main_tmem_destroy ( int argc, char ** argv )`
- 4.146.2.50 `int main_tmem_freeable ( int argc, char ** argv )`
- 4.146.2.51 `int main_tmem_freeze ( int argc, char ** argv )`
- 4.146.2.52 `int main_tmem_list ( int argc, char ** argv )`
- 4.146.2.53 `int main_tmem_set ( int argc, char ** argv )`
- 4.146.2.54 `int main_tmem_shared_auth ( int argc, char ** argv )`
- 4.146.2.55 `int main_tmem_thaw ( int argc, char ** argv )`
- 4.146.2.56 `int main_top ( int argc, char ** argv )`
- 4.146.2.57 `int main_trigger ( int argc, char ** argv )`
- 4.146.2.58 `int main_unpause ( int argc, char ** argv )`
- 4.146.2.59 `int main_uptime ( int argc, char ** argv )`
- 4.146.2.60 `int main_vcpulist ( int argc, char ** argv )`
- 4.146.2.61 `int main_vcpupin ( int argc, char ** argv )`
- 4.146.2.62 `int main_vcpuset ( int argc, char ** argv )`
- 4.146.2.63 `int main_vncviewer ( int argc, char ** argv )`

### 4.146.3 Variable Documentation

4.146.3.1 `libxl_ctx ctx`

4.146.3.2 `int logfile = 2`

## 4.147 xen/tools/python/xen/lowlevel/xl/xl.c File Reference

```
#include <Python.h>
#include <stdio.h>
#include <assert.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <time.h>
#include <getopt.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/time.h>
#include <fcntl.h>
#include <signal.h>
#include <sys/socket.h>
#include <sys/select.h>
#include <arpa/inet.h>
#include <xenctrl.h>
#include <ctype.h>
#include <inttypes.h>
#include <libxl.h>
#include <libxl_utils.h>
#include "_pyxl_types.h"
```

### Data Structures

- struct [XIOObject](#)

### Macros

- #define [ARRAY\\_SIZE](#)(x) (sizeof(x) / sizeof((x)[0]))
- #define [PyMODINIT\\_FUNC](#) DL\_EXPORT(void)
- #define [CLS](#) "ctx"
- #define [Py\\_ssize\\_t](#) int
- #define [\\_INT\\_CONST](#)(m, c) PyModule\_AddIntConstant(m, #c, c)
- #define [\\_INT\\_CONST\\_LIBXL](#)(m, c) PyModule\_AddIntConstant(m, #c, LIBXL\_ ## c)

### Functions

- int [genwrap\\_\\_obj\\_init](#) (PyObject \*self, PyObject \*args, PyObject \*kwargs)
- int [genwrap\\_\\_string\\_set](#) (PyObject \*v, char \*\*str)
- PyObject \* [genwrap\\_\\_string\\_get](#) (char \*\*str)
- PyObject \* [genwrap\\_\\_ull\\_get](#) (unsigned long long val)
- int [genwrap\\_\\_ull\\_set](#) (PyObject \*v, unsigned long long \*val, unsigned long long mask)
- PyObject \* [genwrap\\_\\_ll\\_get](#) (long long val)
- int [genwrap\\_\\_ll\\_set](#) (PyObject \*v, long long \*val, long long mask)
- int [attrib\\_\\_libxl\\_cpuid\\_policy\\_list\\_set](#) (PyObject \*v, [libxl\\_cpuid\\_policy\\_list](#) \*pptr)
- int [attrib\\_\\_void\\_ptr\\_set](#) (PyObject \*v, void \*\*caas\_info)
- int [attrib\\_\\_libxl\\_cpumap\\_set](#) (PyObject \*v, [libxl\\_cpumap](#) \*pptr)
- int [attrib\\_\\_libxl\\_cpuarray\\_set](#) (PyObject \*v, [libxl\\_cpuarray](#) \*pptr)
- int [attrib\\_\\_libxl\\_domain\\_build\\_state\\_ptr\\_set](#) (PyObject \*v, [libxl\\_domain\\_build\\_state](#) \*\*pptr)
- int [attrib\\_\\_libxl\\_file\\_reference\\_set](#) (PyObject \*v, [libxl\\_file\\_reference](#) \*pptr)



- int [attrib\\_\\_libxl\\_hwcap\\_set](#) (PyObject \*v, [libxl\\_hwcap](#) \*pptr)
- int [attrib\\_\\_libxl\\_key\\_value\\_list\\_set](#) (PyObject \*v, [libxl\\_key\\_value\\_list](#) \*pptr)
- int [attrib\\_\\_libxl\\_mac\\_set](#) (PyObject \*v, [libxl\\_mac](#) \*pptr)
- int [attrib\\_\\_libxl\\_string\\_list\\_set](#) (PyObject \*v, [libxl\\_string\\_list](#) \*pptr)
- int [attrib\\_\\_libxl\\_uuid\\_set](#) (PyObject \*v, [libxl\\_uuid](#) \*pptr)
- int [attrib\\_\\_struct\\_in\\_addr\\_set](#) (PyObject \*v, [struct in\\_addr](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_cpuid\\_policy\\_list\\_get](#) ([libxl\\_cpuid\\_policy\\_list](#) \*pptr)
- PyObject \* [attrib\\_\\_void\\_ptr\\_get](#) (void \*\*caas\_info)
- PyObject \* [attrib\\_\\_libxl\\_cpumap\\_get](#) ([libxl\\_cpumap](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_cpuarray\\_get](#) ([libxl\\_cpuarray](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_domain\\_build\\_state\\_ptr\\_get](#) ([libxl\\_domain\\_build\\_state](#) \*\*pptr)
- PyObject \* [attrib\\_\\_libxl\\_file\\_reference\\_get](#) ([libxl\\_file\\_reference](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_hwcap\\_get](#) ([libxl\\_hwcap](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_key\\_value\\_list\\_get](#) ([libxl\\_key\\_value\\_list](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_mac\\_get](#) ([libxl\\_mac](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_string\\_list\\_get](#) ([libxl\\_string\\_list](#) \*pptr)
- PyObject \* [attrib\\_\\_libxl\\_uuid\\_get](#) ([libxl\\_uuid](#) \*pptr)
- PyObject \* [attrib\\_\\_struct\\_in\\_addr\\_get](#) ([struct in\\_addr](#) \*pptr)
- [PyMODINIT\\_FUNC](#) [initxl](#) (void)

#### 4.147.1 Macro Definition Documentation

- 4.147.1.1 `#define _INT_CONST( m, c ) PyModule_AddIntConstant(m, #c, c)`
- 4.147.1.2 `#define _INT_CONST_LIBXL( m, c ) PyModule_AddIntConstant(m, #c, LIBXL_ ## c)`
- 4.147.1.3 `#define ARRAY_SIZE( x ) (sizeof(x) / sizeof((x)[0]))`
- 4.147.1.4 `#define CLS "ctx"`
- 4.147.1.5 `#define Py_ssize_t int`
- 4.147.1.6 `#define PyMODINIT_FUNC DL_EXPORT(void)`

#### 4.147.2 Function Documentation

- 4.147.2.1 `PyObject* attrib\_\_libxl\_cpuarray\_get ( libxl\_cpuarray * pptr )`
- 4.147.2.2 `int attrib\_\_libxl\_cpuarray\_set ( PyObject * v, libxl\_cpuarray * pptr )`
- 4.147.2.3 `PyObject* attrib\_\_libxl\_cpuid\_policy\_list\_get ( libxl\_cpuid\_policy\_list * pptr )`
- 4.147.2.4 `int attrib\_\_libxl\_cpuid\_policy\_list\_set ( PyObject * v, libxl\_cpuid\_policy\_list * pptr )`
- 4.147.2.5 `PyObject* attrib\_\_libxl\_cpumap\_get ( libxl\_cpumap * pptr )`
- 4.147.2.6 `int attrib\_\_libxl\_cpumap\_set ( PyObject * v, libxl\_cpumap * pptr )`
- 4.147.2.7 `PyObject* attrib\_\_libxl\_domain\_build\_state\_ptr\_get ( libxl\_domain\_build\_state ** pptr )`
- 4.147.2.8 `int attrib\_\_libxl\_domain\_build\_state\_ptr\_set ( PyObject * v, libxl\_domain\_build\_state ** pptr )`
- 4.147.2.9 `PyObject* attrib\_\_libxl\_file\_reference\_get ( libxl\_file\_reference * pptr )`

```

4.147.2.10  int attrib__libxl_file_reference_set ( PyObject * v, libxl_file_reference * pptr )

4.147.2.11  PyObject* attrib__libxl_hwcap_get ( libxl_hwcap * pptr )

4.147.2.12  int attrib__libxl_hwcap_set ( PyObject * v, libxl_hwcap * pptr )

4.147.2.13  PyObject* attrib__libxl_key_value_list_get ( libxl_key_value_list * pptr )

4.147.2.14  int attrib__libxl_key_value_list_set ( PyObject * v, libxl_key_value_list * pptr )

4.147.2.15  PyObject* attrib__libxl_mac_get ( libxl_mac * pptr )

4.147.2.16  int attrib__libxl_mac_set ( PyObject * v, libxl_mac * pptr )

4.147.2.17  PyObject* attrib__libxl_string_list_get ( libxl_string_list * pptr )

4.147.2.18  int attrib__libxl_string_list_set ( PyObject * v, libxl_string_list * pptr )

4.147.2.19  PyObject* attrib__libxl_uuid_get ( libxl_uuid * pptr )

4.147.2.20  int attrib__libxl_uuid_set ( PyObject * v, libxl_uuid * pptr )

4.147.2.21  PyObject* attrib__struct_in_addr_get ( struct in_addr * pptr )

4.147.2.22  int attrib__struct_in_addr_set ( PyObject * v, struct in_addr * pptr )

4.147.2.23  PyObject* attrib__void_ptr_get ( void ** caas_info )

4.147.2.24  int attrib__void_ptr_set ( PyObject * v, void ** caas_info )

4.147.2.25  PyObject* genwrap__ll_get ( long long val )

4.147.2.26  int genwrap__ll_set ( PyObject * v, long long * val, long long mask )

4.147.2.27  int genwrap__obj_init ( PyObject * self, PyObject * args, PyObject * kwds )

4.147.2.28  PyObject* genwrap__string_get ( char ** str )

4.147.2.29  int genwrap__string_set ( PyObject * v, char ** str )

4.147.2.30  PyObject* genwrap__ull_get ( unsigned long long val )

4.147.2.31  int genwrap__ull_set ( PyObject * v, unsigned long long * val, unsigned long long mask )

4.147.2.32  PyMODINIT_FUNC initsl ( void )

```

## 4.148 xen/tools/vtpm\_manager/crypto/crypto.h File Reference

```

#include <stddef.h>
#include <stdint.h>
#include <stdbool.h>
#include "tcg.h"
#include "sym_crypto.h"

```

## Data Structures

- struct [CRYPTO\\_INFO](#)

## Macros

- #define [CRYPTO\\_MAX\\_SIG\\_SIZE](#) (2048 / 8)
- #define [CRYPTO\\_MAX\\_RSA\\_KEY\\_SIZE](#) (4096 / 8)
- #define [OAEP\\_P](#) "TCPA"
- #define [OAEP\\_P\\_SIZE](#) 4
- #define [CRYPTO\\_ALGORITHM\\_RSA](#) 0x01
- #define [CRYPTO\\_ES\\_NONE](#) 0x0001
- #define [CRYPTO\\_ES\\_RSAESPKCSv15](#) 0x0002
- #define [CRYPTO\\_ES\\_RSAESOAEP\\_SHA1\\_MGF1](#) 0x0003
- #define [CRYPTO\\_SS\\_NONE](#) 0x0001
- #define [CRYPTO\\_SS\\_RSASSAPKCS1v15\\_SHA1](#) 0x0002
- #define [CRYPTO\\_SS\\_RSASSAPKCS1v15\\_DER](#) 0x0003

## Typedefs

- typedef struct [CRYPTO\\_INFO](#) [CRYPTO\\_INFO](#)

## Functions

- void [Crypto\\_Init](#) (const [BYTE](#) \*rand, int size)
- void [Crypto\\_Exit](#) ()
- void [Crypto\\_GetRandom](#) (void \*data, int size)
- void [Crypto\\_HMAC](#) (const [BYTE](#) \*text, int text\_len, const [BYTE](#) \*key, int key\_len, [BYTE](#) \*digest)
- [TPM\\_RESULT](#) [Crypto\\_HMAC\\_buf](#) (const [buffer\\_t](#) \*text, const [buffer\\_t](#) \*key, [BYTE](#) \*o\_digest)
- void [Crypto\\_SHA1Full](#) (const [BYTE](#) \*text, [UINT32](#) size, [BYTE](#) \*hash)
- [TPM\\_RESULT](#) [Crypto\\_SHA1Full\\_buf](#) (const [buffer\\_t](#) \*buf, [BYTE](#) \*o\_hash)
- void [Crypto\\_SHA1Start](#) ([UINT32](#) \*maxNumBytes)
- void [Crypto\\_SHA1Update](#) (int numBytes, const [BYTE](#) \*hashData)
- void [Crypto\\_SHA1Complete](#) (int hashDataSize, const [BYTE](#) \*hashData, [BYTE](#) \*hashValue)
- void [Crypto\\_RSACreateKey](#) ([UINT32](#) keySize, [UINT32](#) pubExpSize, [BYTE](#) \*pubExp, [UINT32](#) \*privExpSize, [BYTE](#) \*privExp, [UINT32](#) \*modulusSize, [BYTE](#) \*modulus, [CRYPTO\\_INFO](#) \*keys)
- void [Crypto\\_RSABuildCryptoInfo](#) ([UINT32](#) pubExpSize, [BYTE](#) \*pubExp, [UINT32](#) privExpSize, [BYTE](#) \*privExp, [UINT32](#) modulusSize, [BYTE](#) \*modulus, [CRYPTO\\_INFO](#) \*cryptoInfo)
- void [Crypto\\_RSABuildCryptoInfoPublic](#) ([UINT32](#) pubExpSize, [BYTE](#) \*pubExp, [UINT32](#) modulusSize, [BYTE](#) \*modulus, [CRYPTO\\_INFO](#) \*cryptoInfo)
- void [Crypto\\_RSACryptoInfoFree](#) ([CRYPTO\\_INFO](#) \*cryptoInfo)
- [TPM\\_RESULT](#) [Crypto\\_RSAPackCryptoInfo](#) (const [CRYPTO\\_INFO](#) \*cryptoInfo, [BYTE](#) \*\*io\_buf, [UINT32](#) \*io\_buf\_len)
- [TPM\\_RESULT](#) [Crypto\\_RSAUnpackCryptoInfo](#) ([CRYPTO\\_INFO](#) \*ci, [BYTE](#) \*in, [UINT32](#) len, [UINT32](#) \*o\_lenread)
- int [Crypto\\_RSAAEnc](#) ([CRYPTO\\_INFO](#) \*keys, [UINT32](#) inDataSize, [BYTE](#) \*inData, [UINT32](#) \*outDataSize, [BYTE](#) \*outData)
- int [Crypto\\_RSADec](#) ([CRYPTO\\_INFO](#) \*keys, [UINT32](#) inDataSize, [BYTE](#) \*inData, [UINT32](#) \*outDataSize, [BYTE](#) \*outData)
- int [Crypto\\_RSASign](#) ([CRYPTO\\_INFO](#) \*keys, [UINT32](#) inDataSize, [BYTE](#) \*inData, [UINT32](#) \*sigSize, [BYTE](#) \*sig)
- bool [Crypto\\_RSASVerify](#) ([CRYPTO\\_INFO](#) \*keys, [UINT32](#) inDataSize, [BYTE](#) \*inData, [UINT32](#) sigSize, [BYTE](#) \*sig)

- int [RSA\\_verify\\_DER](#) (int dtype, unsigned char \*m, unsigned int m\_len, unsigned char \*sigbuf, unsigned int siglen, [CRYPTO\\_INFO](#) \*key)
- int [RSA\\_sign\\_DER](#) (int type, unsigned char \*m, unsigned int m\_len, unsigned char \*sigret, unsigned int \*siglen, [CRYPTO\\_INFO](#) \*key)

#### 4.148.1 Macro Definition Documentation

- 4.148.1.1 `#define CRYPTO_ALGORITHM_RSA 0x01`
- 4.148.1.2 `#define CRYPTO_ES_NONE 0x0001`
- 4.148.1.3 `#define CRYPTO_ES_RSAESOAEP_SHA1_MGF1 0x0003`
- 4.148.1.4 `#define CRYPTO_ES_RSAESPKCSv15 0x0002`
- 4.148.1.5 `#define CRYPTO_MAX_RSA_KEY_SIZE (4096 / 8)`
- 4.148.1.6 `#define CRYPTO_MAX_SIG_SIZE (2048 / 8)`
- 4.148.1.7 `#define CRYPTO_SS_NONE 0x0001`
- 4.148.1.8 `#define CRYPTO_SS_RSASSAPKCS1v15_DER 0x0003`
- 4.148.1.9 `#define CRYPTO_SS_RSASSAPKCS1v15_SHA1 0x0002`
- 4.148.1.10 `#define OAEP_P "TCPA"`
- 4.148.1.11 `#define OAEP_P_SIZE 4`

#### 4.148.2 Typedef Documentation

- 4.148.2.1 `typedef struct CRYPTO_INFO CRYPTO_INFO`

#### 4.148.3 Function Documentation

- 4.148.3.1 `void Crypto_Exit ( )`
- 4.148.3.2 `void Crypto_GetRandom ( void * data, int size )`
- 4.148.3.3 `void Crypto_HMAC ( const BYTE * text, int text_len, const BYTE * key, int key_len, BYTE * digest )`
- 4.148.3.4 `TPM_RESULT Crypto_HMAC_buf ( const buffer_t * text, const buffer_t * key, BYTE * o_digest )`
- 4.148.3.5 `void Crypto_Init ( const BYTE * rand, int size )`
- 4.148.3.6 `void Crypto_RSABuildCryptoInfo ( UINT32 pubExpSize, BYTE * pubExp, UINT32 privExpSize, BYTE * privExp, UINT32 modulusSize, BYTE * modulus, CRYPTO_INFO * cryptoInfo )`
- 4.148.3.7 `void Crypto_RSABuildCryptoInfoPublic ( UINT32 pubExpSize, BYTE * pubExp, UINT32 modulusSize, BYTE * modulus, CRYPTO_INFO * cryptoInfo )`
- 4.148.3.8 `void Crypto_RSACreateKey ( UINT32 keySize, UINT32 pubExpSize, BYTE * pubExp, UINT32 * privExpSize, BYTE * privExp, UINT32 * modulusSize, BYTE * modulus, CRYPTO_INFO * keys )`
- 4.148.3.9 `void Crypto_RSACryptoInfoFree ( CRYPTO_INFO * cryptoInfo )`

- 4.148.3.10 `int Crypto_RSADec ( CRYPTO_INFO * keys, UINT32 inDataSize, BYTE * inData, UINT32 * outDataSize, BYTE * outData )`
- 4.148.3.11 `int Crypto_RSAEnc ( CRYPTO_INFO * keys, UINT32 inDataSize, BYTE * inData, UINT32 * outDataSize, BYTE * outData )`
- 4.148.3.12 `TPM_RESULT Crypto_RSAPackCryptoInfo ( const CRYPTO_INFO * cryptInfo, BYTE ** io_buf, UINT32 * io_bufLen )`
- 4.148.3.13 `int Crypto_RSASign ( CRYPTO_INFO * keys, UINT32 inDataSize, BYTE * inData, UINT32 * sigSize, BYTE * sig )`
- 4.148.3.14 `TPM_RESULT Crypto_RSAUnpackCryptoInfo ( CRYPTO_INFO * ci, BYTE * in, UINT32 len, UINT32 * o_lenread )`
- 4.148.3.15 `bool Crypto_RSAVerify ( CRYPTO_INFO * keys, UINT32 inDataSize, BYTE * inData, UINT32 sigSize, BYTE * sig )`
- 4.148.3.16 `void Crypto_SHA1Complete ( int hashDataSize, const BYTE * hashData, BYTE * hashValue )`
- 4.148.3.17 `void Crypto_SHA1Full ( const BYTE * text, UINT32 size, BYTE * hash )`
- 4.148.3.18 `TPM_RESULT Crypto_SHA1Full_buf ( const buffer_t * buf, BYTE * o_hash )`
- 4.148.3.19 `void Crypto_SHA1Start ( UINT32 * maxNumBytes )`
- 4.148.3.20 `void Crypto_SHA1Update ( int numBytes, const BYTE * hashData )`
- 4.148.3.21 `int RSA_sign_DER ( int type, unsigned char * m, unsigned int m_len, unsigned char * sigret, unsigned int * siglen, CRYPTO_INFO * key )`
- 4.148.3.22 `int RSA_verify_DER ( int dtype, unsigned char * m, unsigned int m_len, unsigned char * sigbuf, unsigned int siglen, CRYPTO_INFO * key )`

## 4.149 xen/tools/vtpm\_manager/crypto/rsa.c File Reference

```
#include <string.h>
#include <openssl/crypto.h>
#include <openssl/evp.h>
#include <openssl/bn.h>
#include <openssl/rsa.h>
#include <openssl/rand.h>
#include <openssl/x509.h>
#include <openssl/err.h>
#include <stdio.h>
#include "tcg.h"
#include "buffer.h"
#include "crypto.h"
#include "log.h"
```

### Functions

- void [Crypto\\_RSACreateKey](#) (UINT32 keySize, UINT32 pubExpSize, BYTE \*pubExp, UINT32 \*privExpSize, BYTE \*privExp, UINT32 \*modulusSize, BYTE \*modulus, CRYPTO\_INFO \*keys)

- void `Crypto_RSABuildCryptoInfo` (UINT32 pubExpSize, BYTE \*pubExp, UINT32 privExpSize, BYTE \*privExp, UINT32 modulusSize, BYTE \*modulus, CRYPTO\_INFO \*cryptoInfo)
- void `Crypto_RSABuildCryptoInfoPublic` (UINT32 pubExpSize, BYTE \*pubExp, UINT32 modulusSize, BYTE \*modulus, CRYPTO\_INFO \*cryptoInfo)
- void `Crypto_RSACryptoInfoFree` (CRYPTO\_INFO \*cryptoInfo)
- int `Crypto_RSAAEnc` (CRYPTO\_INFO \*key, UINT32 inDataSize, BYTE \*inData, UINT32 \*outDataSize, BYTE \*outData)
- int `Crypto_RSADec` (CRYPTO\_INFO \*key, UINT32 inDataSize, BYTE \*inData, UINT32 \*outDataSize, BYTE \*outData)
- int `Crypto_RSASign` (CRYPTO\_INFO \*key, UINT32 inDataSize, BYTE \*inData, UINT32 \*sigSize, BYTE \*sig)
- bool `Crypto_RSAVerify` (CRYPTO\_INFO \*key, UINT32 inDataSize, BYTE \*inData, UINT32 sigSize, BYTE \*sig)
- TPM\_RESULT `Crypto_RSAPackCryptoInfo` (const CRYPTO\_INFO \*cryptoInfo, BYTE \*\*io\_buf, UINT32 \*io\_buflen)
- TPM\_RESULT `Crypto_RSAUnpackCryptoInfo` (CRYPTO\_INFO \*ci, BYTE \*in, UINT32 len, UINT32 \*o\_lenread)

#### 4.149.1 Function Documentation

- 4.149.1.1 void `Crypto_RSABuildCryptoInfo` ( *UINT32 pubExpSize*, *BYTE \* pubExp*, *UINT32 privExpSize*, *BYTE \* privExp*, *UINT32 modulusSize*, *BYTE \* modulus*, *CRYPTO\_INFO \* cryptoInfo* )
- 4.149.1.2 void `Crypto_RSABuildCryptoInfoPublic` ( *UINT32 pubExpSize*, *BYTE \* pubExp*, *UINT32 modulusSize*, *BYTE \* modulus*, *CRYPTO\_INFO \* cryptoInfo* )
- 4.149.1.3 void `Crypto_RSACreateKey` ( *UINT32 keySize*, *UINT32 pubExpSize*, *BYTE \* pubExp*, *UINT32 \* privExpSize*, *BYTE \* privExp*, *UINT32 \* modulusSize*, *BYTE \* modulus*, *CRYPTO\_INFO \* keys* )
- 4.149.1.4 void `Crypto_RSACryptoInfoFree` ( *CRYPTO\_INFO \* cryptoInfo* )
- 4.149.1.5 int `Crypto_RSADec` ( *CRYPTO\_INFO \* key*, *UINT32 inDataSize*, *BYTE \* inData*, *UINT32 \* outDataSize*, *BYTE \* outData* )
- 4.149.1.6 int `Crypto_RSAAEnc` ( *CRYPTO\_INFO \* key*, *UINT32 inDataSize*, *BYTE \* inData*, *UINT32 \* outDataSize*, *BYTE \* outData* )
- 4.149.1.7 TPM\_RESULT `Crypto_RSAPackCryptoInfo` ( *const CRYPTO\_INFO \* cryptoInfo*, *BYTE \*\* io\_buf*, *UINT32 \* io\_buflen* )
- 4.149.1.8 int `Crypto_RSASign` ( *CRYPTO\_INFO \* key*, *UINT32 inDataSize*, *BYTE \* inData*, *UINT32 \* sigSize*, *BYTE \* sig* )
- 4.149.1.9 TPM\_RESULT `Crypto_RSAUnpackCryptoInfo` ( *CRYPTO\_INFO \* ci*, *BYTE \* in*, *UINT32 len*, *UINT32 \* o\_lenread* )
- 4.149.1.10 bool `Crypto_RSAVerify` ( *CRYPTO\_INFO \* key*, *UINT32 inDataSize*, *BYTE \* inData*, *UINT32 sigSize*, *BYTE \* sig* )

#### 4.150 xen/tools/vtpm\_manager/crypto/sym\_crypto.c File Reference

```
#include <openssl/evp.h>
#include <openssl/rand.h>
#include "tcg.h"
#include "log.h"
#include "sym_crypto.h"
```

## Typedefs

- typedef enum [crypt\\_op\\_type\\_t](#) [crypt\\_op\\_type\\_t](#)

## Enumerations

- enum [crypt\\_op\\_type\\_t](#) { [CRYPT\\_ENCRYPT](#), [CRYPT\\_DECRYPT](#) }

## Functions

- [TPM\\_RESULT](#) [ossl\\_symcrypto\\_op](#) (symkey\_t \*key, const [buffer\\_t](#) \*in, const [buffer\\_t](#) \*iv, [buffer\\_t](#) \*out, [crypt\\_op\\_type\\_t](#) optype)
- [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_initkey](#) (symkey\_t \*key, const [buffer\\_t](#) \*keybits)
- [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_genkey](#) (symkey\_t \*key)
- [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_encrypt](#) (symkey\_t \*key, const [buffer\\_t](#) \*clear, [buffer\\_t](#) \*o\_cipher)
- [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_decrypt](#) (symkey\_t \*key, const [buffer\\_t](#) \*cipher, [buffer\\_t](#) \*o\_clear)
- [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_freekey](#) (symkey\_t \*key)

## Variables

- const EVP\_CIPHER \* [SYM\\_CIPHER](#) = NULL
- const [BYTE](#) [ZERO\\_IV](#) [EVP\_MAX\_IV\_LENGTH] = {0}

### 4.150.1 Typedef Documentation

#### 4.150.1.1 typedef enum [crypt\\_op\\_type\\_t](#) [crypt\\_op\\_type\\_t](#)

### 4.150.2 Enumeration Type Documentation

#### 4.150.2.1 enum [crypt\\_op\\_type\\_t](#)

#### Enumerator

***CRYPT\_ENCRYPT***

***CRYPT\_DECRYPT***

### 4.150.3 Function Documentation

#### 4.150.3.1 [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_decrypt](#) ( symkey\_t \* key, const [buffer\\_t](#) \* cipher, [buffer\\_t](#) \* o\_clear )

#### 4.150.3.2 [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_encrypt](#) ( symkey\_t \* key, const [buffer\\_t](#) \* clear, [buffer\\_t](#) \* o\_cipher )

#### 4.150.3.3 [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_freekey](#) ( symkey\_t \* key )

#### 4.150.3.4 [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_genkey](#) ( symkey\_t \* key )

#### 4.150.3.5 [TPM\\_RESULT](#) [Crypto\\_symcrypto\\_initkey](#) ( symkey\_t \* key, const [buffer\\_t](#) \* keybits )

#### 4.150.3.6 [TPM\\_RESULT](#) [ossl\\_symcrypto\\_op](#) ( symkey\_t \* key, const [buffer\\_t](#) \* in, const [buffer\\_t](#) \* iv, [buffer\\_t](#) \* out, [crypt\\_op\\_type\\_t](#) optype )

#### 4.150.4 Variable Documentation

4.150.4.1 `const EVP_CIPHER* SYM_CIPHER = NULL`

4.150.4.2 `const BYTE ZERO_IV[EVP_MAX_IV_LENGTH] = {0}`

### 4.151 xen/tools/vtpm\_manager/manager/dmictl.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include "vtpmppriv.h"
#include "bsg.h"
#include "buffer.h"
#include "log.h"
#include "hashtable.h"
#include "hashtable_itr.h"
#include "vtpm_ipc.h"
```

#### Macros

- `#define TPM_EMULATOR_PATH "/usr/bin/vtpmd"`

#### Functions

- `TPM_RESULT init_dmi (UINT32 dmi_id, BYTE dmi_type, VTPM_DMI_RESOURCE **dmi_res)`
- `TPM_RESULT close_dmi (VTPM_DMI_RESOURCE *dmi_res)`
- `void free_dmi (VTPM_DMI_RESOURCE *dmi_res)`
- `TPM_RESULT VTPM_Handle_New_DMI (const buffer_t *param_buf)`
- `TPM_RESULT VTPM_Handle_Close_DMI (const buffer_t *param_buf)`
- `TPM_RESULT VTPM_Handle_Delete_DMI (const buffer_t *param_buf)`

#### 4.151.1 Macro Definition Documentation

4.151.1.1 `#define TPM_EMULATOR_PATH "/usr/bin/vtpmd"`

#### 4.151.2 Function Documentation

4.151.2.1 `TPM_RESULT close_dmi ( VTPM_DMI_RESOURCE * dmi_res )`

4.151.2.2 `void free_dmi ( VTPM_DMI_RESOURCE * dmi_res )`

4.151.2.3 `TPM_RESULT init_dmi ( UINT32 dmi_id, BYTE dmi_type, VTPM_DMI_RESOURCE ** dmi_res )`

4.151.2.4 `TPM_RESULT VTPM_Handle_Close_DMI ( const buffer_t * param_buf )`

4.151.2.5 `TPM_RESULT VTPM_Handle_Delete_DMI ( const buffer_t * param_buf )`

4.151.2.6 `TPM_RESULT VTPM_Handle_New_DMI ( const buffer_t * param_buf )`



## 4.152 xen/tools/vtpm\_manager/manager/securestorage.c File Reference

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>
#include "tcg.h"
#include "vtpm_manager.h"
#include "vtpmpriv.h"
#include "vtsp.h"
#include "bsg.h"
#include "crypto.h"
#include "hashtable.h"
#include "hashtable_itr.h"
#include "buffer.h"
#include "log.h"
```

### Functions

- [TPM\\_RESULT envelope\\_encrypt](#) (const [buffer\\_t](#) \*inbuf, [CRYPTO\\_INFO](#) \*asymkey, [buffer\\_t](#) \*sealed\_data)
- [TPM\\_RESULT symkey\\_encrypt](#) (const [buffer\\_t](#) \*inbuf, [CRYPTO\\_INFO](#) \*asymkey, [buffer\\_t](#) \*sealed\_key)
- [TPM\\_RESULT envelope\\_decrypt](#) (const [buffer\\_t](#) \*cipher, [TCS\\_CONTEXT\\_HANDLE](#) TCSCContext, [TPM\\_HANDLE](#) keyHandle, const [TPM\\_AUTHDATA](#) \*key\_usage\_auth, [buffer\\_t](#) \*unsealed\_data)
- [TPM\\_RESULT symkey\\_decrypt](#) (const [buffer\\_t](#) \*cipher, [TCS\\_CONTEXT\\_HANDLE](#) TCSCContext, [TPM\\_HANDLE](#) keyHandle, const [TPM\\_AUTHDATA](#) \*key\_usage\_auth, [buffer\\_t](#) \*symkey\_clear)
- [TPM\\_RESULT VTPM\\_Handle\\_Save\\_NVM](#) ([VTPM\\_DMI\\_RESOURCE](#) \*myDMI, const [buffer\\_t](#) \*inbuf, [buffer\\_t](#) \*outbuf)
- [TPM\\_RESULT VTPM\\_Handle\\_Load\\_NVM](#) ([VTPM\\_DMI\\_RESOURCE](#) \*myDMI, const [buffer\\_t](#) \*inbuf, [buffer\\_t](#) \*outbuf)
- [TPM\\_RESULT VTPM\\_Handle\\_Get\\_NVM\\_Size](#) ([VTPM\\_DMI\\_RESOURCE](#) \*myDMI, const [buffer\\_t](#) \*inbuf, [buffer\\_t](#) \*outbuf)
- [TPM\\_RESULT VTPM\\_Handle\\_Load\\_Key](#) ([VTPM\\_DMI\\_RESOURCE](#) \*myDMI, const [buffer\\_t](#) \*inbuf, [buffer\\_t](#) \*outbuf)
- [TPM\\_RESULT VTPM\\_Handle\\_Save\\_Key](#) ([VTPM\\_DMI\\_RESOURCE](#) \*myDMI, const [buffer\\_t](#) \*inbuf, [buffer\\_t](#) \*outbuf)
- [TPM\\_RESULT VTPM\\_SaveManagerData](#) (void)
- [TPM\\_RESULT VTPM\\_LoadManagerData](#) (void)

### 4.152.1 Function Documentation

4.152.1.1 [TPM\\_RESULT envelope\\_decrypt](#) ( const [buffer\\_t](#) \* *cipher*, [TCS\\_CONTEXT\\_HANDLE](#) *TCSCContext*, [TPM\\_HANDLE](#) *keyHandle*, const [TPM\\_AUTHDATA](#) \* *key\_usage\_auth*, [buffer\\_t](#) \* *unsealed\_data* )

4.152.1.2 [TPM\\_RESULT envelope\\_encrypt](#) ( const [buffer\\_t](#) \* *inbuf*, [CRYPTO\\_INFO](#) \* *asymkey*, [buffer\\_t](#) \* *sealed\_data* )

4.152.1.3 [TPM\\_RESULT symkey\\_decrypt](#) ( const [buffer\\_t](#) \* *cipher*, [TCS\\_CONTEXT\\_HANDLE](#) *TCSCContext*, [TPM\\_HANDLE](#) *keyHandle*, const [TPM\\_AUTHDATA](#) \* *key\_usage\_auth*, [buffer\\_t](#) \* *symkey\_clear* )

4.152.1.4 [TPM\\_RESULT symkey\\_encrypt](#) ( const [buffer\\_t](#) \* *inbuf*, [CRYPTO\\_INFO](#) \* *asymkey*, [buffer\\_t](#) \* *sealed\_key* )

4.152.1.5 [TPM\\_RESULT VTPM\\_Handle\\_Get\\_NVM\\_Size](#) ( [VTPM\\_DMI\\_RESOURCE](#) \* *myDMI*, const [buffer\\_t](#) \* *inbuf*, [buffer\\_t](#) \* *outbuf* )

- 4.152.1.6 **TPM\_RESULT** `VTPM_Handle_Load_Key ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.152.1.7 **TPM\_RESULT** `VTPM_Handle_Load_NVM ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.152.1.8 **TPM\_RESULT** `VTPM_Handle_Save_Key ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.152.1.9 **TPM\_RESULT** `VTPM_Handle_Save_NVM ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.152.1.10 **TPM\_RESULT** `VTPM_LoadManagerData ( void )`
- 4.152.1.11 **TPM\_RESULT** `VTPM_SaveManagerData ( void )`

## 4.153 xen/tools/vtpm\_manager/manager/vtpm\_ipc.c File Reference

```
#include <sys/stat.h>
#include "vtpm_ipc.h"
#include "vtpmpriv.h"
#include "log.h"
```

### Functions

- int `vtpm_ipc_init` (`vtpm_ipc_handle_t` \**ipc\_h*, char \**name*, int *flags*, **BOOL** *create*, **BOOL** *preopen*)
- int `vtpm_ipc_create` (`vtpm_ipc_handle_t` \**ipc\_h*)
- int `vtpm_ipc_read` (`vtpm_ipc_handle_t` \**ipc\_h*, `vtpm_ipc_handle_t` \**alt\_ipc\_h*, **BYTE** \**bytes*, **UINT32** *size*)
- int `vtpm_ipc_write` (`vtpm_ipc_handle_t` \**ipc\_h*, `vtpm_ipc_handle_t` \**alt\_ipc\_h*, **BYTE** \**bytes*, **UINT32** *size*)
- void `vtpm_ipc_close` (`vtpm_ipc_handle_t` \**ipc\_h*)

### Variables

- volatile sig\_atomic\_t `IPC_QUIT_FLAG` = 0

#### 4.153.1 Function Documentation

- 4.153.1.1 void `vtpm_ipc_close` ( `vtpm_ipc_handle_t` \* *ipc\_h* )
- 4.153.1.2 int `vtpm_ipc_create` ( `vtpm_ipc_handle_t` \* *ipc\_h* )
- 4.153.1.3 int `vtpm_ipc_init` ( `vtpm_ipc_handle_t` \* *ipc\_h*, char \* *name*, int *flags*, **BOOL** *create*, **BOOL** *preopen* )
- 4.153.1.4 int `vtpm_ipc_read` ( `vtpm_ipc_handle_t` \* *ipc\_h*, `vtpm_ipc_handle_t` \* *alt\_ipc\_h*, **BYTE** \* *bytes*, **UINT32** *size* )
- 4.153.1.5 int `vtpm_ipc_write` ( `vtpm_ipc_handle_t` \* *ipc\_h*, `vtpm_ipc_handle_t` \* *alt\_ipc\_h*, **BYTE** \* *bytes*, **UINT32** *size* )

#### 4.153.2 Variable Documentation

- 4.153.2.1 volatile sig\_atomic\_t `IPC_QUIT_FLAG` = 0

## 4.154 xen/tools/vtpm\_manager/manager/vtpm\_ipc.h File Reference

```
#include <signal.h>
#include "tcg.h"
```

### Data Structures

- struct [vtpm\\_ipc\\_handle\\_t](#)

### Macros

- #define [VTPM\\_IPC\\_CLOSED](#) -1

### Typedefs

- typedef struct [vtpm\\_ipc\\_handle\\_t](#) [vtpm\\_ipc\\_handle\\_t](#)

### Functions

- int [vtpm\\_ipc\\_init](#) ([vtpm\\_ipc\\_handle\\_t](#) \*ioh, char \*name, int flags, [BOOL](#) create, [BOOL](#) preopen)
- int [vtpm\\_ipc\\_create](#) ([vtpm\\_ipc\\_handle\\_t](#) \*ioh)
- int [vtpm\\_ipc\\_read](#) ([vtpm\\_ipc\\_handle\\_t](#) \*ioh, [vtpm\\_ipc\\_handle\\_t](#) \*alt\_ioh, [BYTE](#) \*bytes, [UINT32](#) size)
- int [vtpm\\_ipc\\_write](#) ([vtpm\\_ipc\\_handle\\_t](#) \*ioh, [vtpm\\_ipc\\_handle\\_t](#) \*alt\_ioh, [BYTE](#) \*bytes, [UINT32](#) size)
- void [vtpm\\_ipc\\_close](#) ([vtpm\\_ipc\\_handle\\_t](#) \*ioh)

### Variables

- volatile sig\_atomic\_t [IPC\\_QUIT\\_FLAG](#)

#### 4.154.1 Macro Definition Documentation

4.154.1.1 #define [VTPM\\_IPC\\_CLOSED](#) -1

#### 4.154.2 Typedef Documentation

4.154.2.1 typedef struct [vtpm\\_ipc\\_handle\\_t](#) [vtpm\\_ipc\\_handle\\_t](#)

#### 4.154.3 Function Documentation

4.154.3.1 void [vtpm\\_ipc\\_close](#) ( [vtpm\\_ipc\\_handle\\_t](#) \* *ioh* )

4.154.3.2 int [vtpm\\_ipc\\_create](#) ( [vtpm\\_ipc\\_handle\\_t](#) \* *ioh* )

4.154.3.3 int [vtpm\\_ipc\\_init](#) ( [vtpm\\_ipc\\_handle\\_t](#) \* *ioh*, char \* *name*, int *flags*, [BOOL](#) *create*, [BOOL](#) *preopen* )

4.154.3.4 int [vtpm\\_ipc\\_read](#) ( [vtpm\\_ipc\\_handle\\_t](#) \* *ioh*, [vtpm\\_ipc\\_handle\\_t](#) \* *alt\_ioh*, [BYTE](#) \* *bytes*, [UINT32](#) *size* )

4.154.3.5 int [vtpm\\_ipc\\_write](#) ( [vtpm\\_ipc\\_handle\\_t](#) \* *ioh*, [vtpm\\_ipc\\_handle\\_t](#) \* *alt\_ioh*, [BYTE](#) \* *bytes*, [UINT32](#) *size* )

#### 4.154.4 Variable Documentation

## 4.154.4.1 volatile sig\_atomic\_t IPC\_QUIT\_FLAG

## 4.155 xen/tools/vtpm\_manager/manager/vtpm\_manager.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include "vtpm_manager.h"
#include "vtpmpriv.h"
#include "vtsp.h"
#include "tpmddl.h"
#include "bsg.h"
#include "hashtable.h"
#include "hashtable_itr.h"
#include "log.h"
#include "buffer.h"
```

## Macros

- #define [RLISTSZ](#) 6

## Functions

- [TPM\\_RESULT VTPM\\_FlushResources](#) ()
- [TPM\\_RESULT VTPM\\_Create\\_Manager](#) ()
- [TPM\\_RESULT VTPM\\_Init\\_Manager](#) ()
- void [VTPM\\_Stop\\_Manager](#) ()

## Variables

- [VTPM\\_GLOBALS](#) \* [vtpm\\_globals](#) =NULL
- const [TPM\\_AUTHDATA SRK\\_AUTH](#)

## 4.155.1 Macro Definition Documentation

## 4.155.1.1 #define RLISTSZ 6

## 4.155.2 Function Documentation

## 4.155.2.1 TPM\_RESULT VTPM\_Create\_Manager ( )

## 4.155.2.2 TPM\_RESULT VTPM\_FlushResources ( )

## 4.155.2.3 TPM\_RESULT VTPM\_Init\_Manager ( )

## 4.155.2.4 void VTPM\_Stop\_Manager ( )

## 4.155.3 Variable Documentation

## 4.155.3.1 const TPM\_AUTHDATA SRK\_AUTH

## Initial value:

```
= { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
```

#### 4.155.3.2 VTPM\_GLOBALS\* vtpm\_globals=NULL

## 4.156 xen/tools/vtpm\_manager/manager/vtpm\_manager.h File Reference

### Macros

- #define [VTPM\\_TAG\\_REQ](#) 0x01c1
- #define [VTPM\\_TAG\\_RSP](#) 0x01c4
- #define [COMMAND\\_BUFFER\\_SIZE](#) 4096
- #define [VTPM\\_COMMAND\\_HEADER\\_SIZE\\_CLT](#) ( 2 + 4 + 4)
- #define [VTPM\\_COMMAND\\_HEADER\\_SIZE\\_SRV](#) ( 4 + [VTPM\\_COMMAND\\_HEADER\\_SIZE\\_CLT](#) )
- #define [VTPM\\_ORD\\_BASE](#) 0x0000
- #define [VTPM\\_PRIV\\_MASK](#) 0x01000000
- #define [VTPM\\_PRIV\\_BASE](#) ([VTPM\\_ORD\\_BASE](#) | [VTPM\\_PRIV\\_MASK](#))
- #define [VTPM\\_ORD\\_SAVENVM](#) ([VTPM\\_ORD\\_BASE](#) + 1)
- #define [VTPM\\_ORD\\_LOADNVM](#) ([VTPM\\_ORD\\_BASE](#) + 2)
- #define [VTPM\\_ORD\\_TPMCOMMAND](#) ([VTPM\\_ORD\\_BASE](#) + 3)
- #define [VTPM\\_ORD\\_GET\\_MIG\\_KEY](#) ([VTPM\\_ORD\\_BASE](#) + 4)
- #define [VTPM\\_ORD\\_LOAD\\_MIG\\_KEY](#) ([VTPM\\_ORD\\_BASE](#) + 5)
- #define [VTPM\\_ORD\\_GETNVMSIZE](#) ([VTPM\\_ORD\\_BASE](#) + 6)
- #define [VTPM\\_ORD\\_SAVEKEY](#) ([VTPM\\_ORD\\_BASE](#) + 7)
- #define [VTPM\\_ORD\\_LOADKEY](#) ([VTPM\\_ORD\\_BASE](#) + 8)
- #define [VTPM\\_ORD\\_OPEN](#) ([VTPM\\_PRIV\\_BASE](#) + 1)
- #define [VTPM\\_ORD\\_CLOSE](#) ([VTPM\\_PRIV\\_BASE](#) + 2)
- #define [VTPM\\_ORD\\_DELETE](#) ([VTPM\\_PRIV\\_BASE](#) + 3)
- #define [VTPM\\_ORD\\_MIGRATE\\_IN](#) ([VTPM\\_PRIV\\_BASE](#) + 4)
- #define [VTPM\\_ORD\\_MIGRATE\\_OUT](#) ([VTPM\\_PRIV\\_BASE](#) + 5)
- #define [VTPM\\_TYPE\\_PVM](#) 1
- #define [VTPM\\_TYPE\\_HVM](#) 2
- #define [VTPM\\_SUCCESS](#) 0
- #define [VTPM\\_FAIL](#) 1
- #define [VTPM\\_UNSUPPORTED](#) 2
- #define [VTPM\\_FORBIDDEN](#) 3
- #define [VTPM\\_RESTORE\\_CONTEXT\\_FAILED](#) 4
- #define [VTPM\\_INVALID\\_REQUEST](#) 5
- #define [VTPM\\_TYPE\\_NON\\_MIGRATABLE](#) 0x00
- #define [VTPM\\_TYPE\\_MIGRATABLE](#) 0x01
- #define [VTPM\\_TYPE\\_MIGRATED](#) 0xFF
- #define [TPM\\_EMULATOR\\_PATH](#) "/usr/bin/vtpmd"
- #define [VTPM\\_BE\\_FNAME](#) "/dev/vtpm"
- #define [VTPM\\_DUMMY\\_TX\\_BE\\_FNAME](#) "/var/vtpm/fifos/dummy\_out.fifo"
- #define [VTPM\\_DUMMY\\_RX\\_BE\\_FNAME](#) "/var/vtpm/fifos/dummy\_in.fifo"
- #define [VTPM\\_TX\\_TPM\\_FNAME](#) "/var/vtpm/fifos/tpm\_cmd\_to\_%d.fifo"
- #define [VTPM\\_RX\\_TPM\\_FNAME](#) "/var/vtpm/fifos/tpm\_rsp\_from\_all.fifo"
- #define [VTPM\\_TX\\_VTPM\\_FNAME](#) "/var/vtpm/fifos/vtpm\_rsp\_to\_%d.fifo"
- #define [VTPM\\_RX\\_VTPM\\_FNAME](#) "/var/vtpm/fifos/vtpm\_cmd\_from\_all.fifo"
- #define [VTPM\\_TX\\_HP\\_FNAME](#) "/var/vtpm/fifos/to\_console.fifo"
- #define [VTPM\\_RX\\_HP\\_FNAME](#) "/var/vtpm/fifos/from\_console.fifo"
- #define [VTPM\\_TYPE\\_PVM\\_STRING](#) "pvm"
- #define [VTPM\\_TYPE\\_HVM\\_STRING](#) "hvm"

#### 4.156.1 Macro Definition Documentation

4.156.1.1 `#define COMMAND_BUFFER_SIZE 4096`

4.156.1.2 `#define TPM_EMULATOR_PATH "/usr/bin/vtpmd"`

4.156.1.3 `#define VTPM_BE_FNAME "/dev/vtpm"`

4.156.1.4 `#define VTPM_COMMAND_HEADER_SIZE_CLT ( 2 + 4 + 4 )`

4.156.1.5 `#define VTPM_COMMAND_HEADER_SIZE_SRV ( 4 + VTPM_COMMAND_HEADER_SIZE_CLT )`

4.156.1.6 `#define VTPM_DUMMY_RX_BE_FNAME "/var/vtpm/fifos/dummy_in.fifo"`

4.156.1.7 `#define VTPM_DUMMY_TX_BE_FNAME "/var/vtpm/fifos/dummy_out.fifo"`

4.156.1.8 `#define VTPM_FAIL 1`

4.156.1.9 `#define VTPM_FORBIDDEN 3`

4.156.1.10 `#define VTPM_INVALID_REQUEST 5`

4.156.1.11 `#define VTPM_ORD_BASE 0x0000`

4.156.1.12 `#define VTPM_ORD_CLOSE (VTPM_PRIV_BASE + 2)`

4.156.1.13 `#define VTPM_ORD_DELETE (VTPM_PRIV_BASE + 3)`

4.156.1.14 `#define VTPM_ORD_GET_MIG_KEY (VTPM_ORD_BASE + 4)`

4.156.1.15 `#define VTPM_ORD_GETNVMSIZE (VTPM_ORD_BASE + 6)`

4.156.1.16 `#define VTPM_ORD_LOAD_MIG_KEY (VTPM_ORD_BASE + 5)`

4.156.1.17 `#define VTPM_ORD_LOADKEY (VTPM_ORD_BASE + 8)`

4.156.1.18 `#define VTPM_ORD_LOADNVM (VTPM_ORD_BASE + 2)`

4.156.1.19 `#define VTPM_ORD_MIGRATE_IN (VTPM_PRIV_BASE + 4)`

4.156.1.20 `#define VTPM_ORD_MIGRATE_OUT (VTPM_PRIV_BASE + 5)`

4.156.1.21 `#define VTPM_ORD_OPEN (VTPM_PRIV_BASE + 1)`

4.156.1.22 `#define VTPM_ORD_SAVEKEY (VTPM_ORD_BASE + 7)`

4.156.1.23 `#define VTPM_ORD_SAVENVM (VTPM_ORD_BASE + 1)`

4.156.1.24 `#define VTPM_ORD_TPMCOMMAND (VTPM_ORD_BASE + 3)`

4.156.1.25 `#define VTPM_PRIV_BASE (VTPM_ORD_BASE | VTPM_PRIV_MASK)`

4.156.1.26 `#define VTPM_PRIV_MASK 0x01000000`

4.156.1.27 `#define VTPM_RESTORE_CONTEXT_FAILED 4`

```

4.156.1.28 #define VTPM_RX_HP_FNAME "/var/vtpm/fifos/from_console.fifo"

4.156.1.29 #define VTPM_RX_TPM_FNAME "/var/vtpm/fifos/tpm_rsp_from_all.fifo"

4.156.1.30 #define VTPM_RX_VTPM_FNAME "/var/vtpm/fifos/vtpm_cmd_from_all.fifo"

4.156.1.31 #define VTPM_SUCCESS 0

4.156.1.32 #define VTPM_TAG_REQ 0x01c1

4.156.1.33 #define VTPM_TAG_RSP 0x01c4

4.156.1.34 #define VTPM_TX_HP_FNAME "/var/vtpm/fifos/to_console.fifo"

4.156.1.35 #define VTPM_TX_TPM_FNAME "/var/vtpm/fifos/tpm_cmd_to_%.d.fifo"

4.156.1.36 #define VTPM_TX_VTPM_FNAME "/var/vtpm/fifos/vtpm_rsp_to_%.d.fifo"

4.156.1.37 #define VTPM_TYPE_HVM 2

4.156.1.38 #define VTPM_TYPE_HVM_STRING "hvm"

4.156.1.39 #define VTPM_TYPE_MIGRATABLE 0x01

4.156.1.40 #define VTPM_TYPE_MIGRATED 0xFF

4.156.1.41 #define VTPM_TYPE_NON_MIGRATABLE 0x00

4.156.1.42 #define VTPM_TYPE_PVM 1

4.156.1.43 #define VTPM_TYPE_PVM_STRING "pvm"

4.156.1.44 #define VTPM_UNSUPPORTED 2

```

## 4.157 xen/tools/vtpm\_manager/manager/vtpm\_manager\_handler.c File Reference

```

#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <errno.h>
#include <signal.h>
#include "vtpm_manager.h"
#include "vtpmpriv.h"
#include "vtsp.h"
#include "bsg.h"
#include "hashtable.h"
#include "hashtable_itr.h"
#include "log.h"
#include "buffer.h"

```

### Macros

- #define [vtpmhandlerloginfo](#)(module, fmt, args...) vtpmloginfo (module, "[%s]: " fmt, thread\_name, ##args );
- #define [vtpmhandlerloginfomore](#)(module, fmt, args...) vtpmloginfomore (module, fmt, ##args );

- `#define vtpmhandlerlogerror(module, fmt, args...) vtpmlogerror (module, "[%s]: " fmt, thread_name, ##args );`

## Functions

- `TPM_RESULT vtpm_manager_handle_vtpm_cmd (VTPM_DMI_RESOURCE *dmi_res, TPM_COMMAND_CODE ord, buffer_t *command_buf, buffer_t *result_buf, BOOL is_priv, char *thread_name)`
- `TPM_RESULT vtpm_manager_handle_tpm_cmd (vtpm_ipc_handle_t *tx_ipc_h, vtpm_ipc_handle_t *rx_ipc_h, VTPM_DMI_RESOURCE *dmi_res, BYTE *cmd_header, buffer_t *param_buf, buffer_t *result_buf, char *thread_name)`
- `TPM_RESULT VTPM_Manager_Handler (vtpm_ipc_handle_t *tx_ipc_h, vtpm_ipc_handle_t *rx_ipc_h, BOOL fw_tpm, vtpm_ipc_handle_t *fw_tx_ipc_h, vtpm_ipc_handle_t *fw_rx_ipc_h, BOOL is_priv, char *thread_name)`

## Variables

- volatile sig\_atomic\_t `HANDLER_QUIT_FLAG` = 0

### 4.157.1 Macro Definition Documentation

4.157.1.1 `#define vtpmhandlerlogerror( module, fmt, args... ) vtpmlogerror (module, "[%s]: " fmt, thread_name, ##args );`

4.157.1.2 `#define vtpmhandlerloginfo( module, fmt, args... ) vtpmloginfo (module, "[%s]: " fmt, thread_name, ##args );`

4.157.1.3 `#define vtpmhandlerloginfomore( module, fmt, args... ) vtpmloginfomore (module, fmt, ##args );`

### 4.157.2 Function Documentation

4.157.2.1 `TPM_RESULT vtpm_manager_handle_tpm_cmd ( vtpm_ipc_handle_t * tx_ipc_h, vtpm_ipc_handle_t * rx_ipc_h, VTPM_DMI_RESOURCE * dmi_res, BYTE * cmd_header, buffer_t * param_buf, buffer_t * result_buf, char * thread_name )`

4.157.2.2 `TPM_RESULT vtpm_manager_handle_vtpm_cmd ( VTPM_DMI_RESOURCE * dmi_res, TPM_COMMAND_CODE ord, buffer_t * command_buf, buffer_t * result_buf, BOOL is_priv, char * thread_name )`

4.157.2.3 `TPM_RESULT VTPM_Manager_Handler ( vtpm_ipc_handle_t * tx_ipc_h, vtpm_ipc_handle_t * rx_ipc_h, BOOL fw_tpm, vtpm_ipc_handle_t * fw_tx_ipc_h, vtpm_ipc_handle_t * fw_rx_ipc_h, BOOL is_priv, char * thread_name )`

### 4.157.3 Variable Documentation

4.157.3.1 volatile sig\_atomic\_t `HANDLER_QUIT_FLAG` = 0



## 4.158 xen/tools/vtpm\_manager/manager/vtpmd.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <signal.h>
#include <string.h>
#include <pthread.h>
#include "vtpm_manager.h"
#include "vtpmpriv.h"
#include "tcg.h"
#include "log.h"
#include "vtpm_ipc.h"
```

### Data Structures

- struct [vtpm\\_thread\\_params\\_s](#)

### Functions

- void \* [vtpm\\_manager\\_thread](#) (void \*arg\_void)
- void [signal\\_handler](#) (int signal)
- [TPM\\_RESULT](#) [VTPM\\_New\\_DMI\\_Extra](#) ([VTPM\\_DMI\\_RESOURCE](#) \*dmi\_res, [BYTE](#) vm\_type, [BYTE](#) startup\_mode)
- [TPM\\_RESULT](#) [VTPM\\_Close\\_DMI\\_Extra](#) ([VTPM\\_DMI\\_RESOURCE](#) \*dmi\_res)
- int [main](#) (int argc, char \*\*argv)

### Variables

- struct sigaction [ctl\\_c\\_handler](#)

#### 4.158.1 Function Documentation

4.158.1.1 int [main](#) ( int *argc*, char \*\* *argv* )

4.158.1.2 void [signal\\_handler](#) ( int *signal* )

4.158.1.3 [TPM\\_RESULT](#) [VTPM\\_Close\\_DMI\\_Extra](#) ( [VTPM\\_DMI\\_RESOURCE](#) \* *dmi\_res* )

4.158.1.4 void\* [vtpm\\_manager\\_thread](#) ( void \* *arg\_void* )

4.158.1.5 [TPM\\_RESULT](#) [VTPM\\_New\\_DMI\\_Extra](#) ( [VTPM\\_DMI\\_RESOURCE](#) \* *dmi\_res*, [BYTE](#) *vm\_type*, [BYTE](#) *startup\_mode* )

#### 4.158.2 Variable Documentation

4.158.2.1 struct sigaction [ctl\\_c\\_handler](#)

## 4.159 xen/tools/vtpm\_manager/manager/vtpmpriv.h File Reference

```
#include <signal.h>
#include "vtpm_manager.h"
#include "tcg.h"
#include "tcs.h"
#include "buffer.h"
#include "crypto.h"
#include "vtpm_ipc.h"
```

### Data Structures

- struct [VTPM\\_DMI\\_RESOURCE\\_T](#)
- struct [tdVTPM\\_MIGKEY\\_LIST](#)
- struct [tdVTPM\\_GLOBALS](#)

### Macros

- #define [VTPM\\_MANAGER\\_GEN](#) 2
- #define [STATE\\_FILE](#) "/var/vtpm/VTPM"
- #define [DMI\\_NVM\\_FILE](#) "/var/vtpm/vtpm\_dm\_%d.data"
- #define [VTPM\\_CTL\\_DM](#) 0

### Typedefs

- typedef struct [VTPM\\_DMI\\_RESOURCE\\_T](#) [VTPM\\_DMI\\_RESOURCE](#)
- typedef struct [tdVTPM\\_MIGKEY\\_LIST](#) [VTPM\\_MIGKEY\\_LIST](#)
- typedef struct [tdVTPM\\_GLOBALS](#) [VTPM\\_GLOBALS](#)

### Functions

- [TPM\\_RESULT VTPM\\_Init\\_Manager \(\)](#)
- [void VTPM\\_Stop\\_Manager \(\)](#)
- [TPM\\_RESULT VTPM\\_Manager\\_Handler \(vtpm\\_ipc\\_handle\\_t \\*tx\\_ipc\\_h, vtpm\\_ipc\\_handle\\_t \\*rx\\_ipc\\_h, BOOL fw\\_tpm, vtpm\\_ipc\\_handle\\_t \\*fw\\_tx\\_ipc\\_h, vtpm\\_ipc\\_handle\\_t \\*fw\\_rx\\_ipc\\_h, BOOL is\\_priv, char \\*client\\_name\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Load\\_NVM \(VTPM\\_DMI\\_RESOURCE \\*myDMI, const buffer\\_t \\*inbuf, buffer\\_t \\*outbuf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Save\\_NVM \(VTPM\\_DMI\\_RESOURCE \\*myDMI, const buffer\\_t \\*inbuf, buffer\\_t \\*outbuf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Get\\_NVM\\_Size \(VTPM\\_DMI\\_RESOURCE \\*myDMI, const buffer\\_t \\*inbuf, buffer\\_t \\*outbuf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_TPM\\_Command \(VTPM\\_DMI\\_RESOURCE \\*dmi, buffer\\_t \\*inbuf, buffer\\_t \\*outbuf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_New\\_DMI \(const buffer\\_t \\*param\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Close\\_DMI \(const buffer\\_t \\*param\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Delete\\_DMI \(const buffer\\_t \\*param\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Migrate\\_In \(const buffer\\_t \\*param\\_buf, buffer\\_t \\*result\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Migrate\\_Out \(const buffer\\_t \\*param\\_buf, buffer\\_t \\*result\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_Handle\\_Get\\_Migration\\_key \(const buffer\\_t \\*param\\_buf, buffer\\_t \\*result\\_buf\)](#)
- [TPM\\_RESULT VTPM\\_SaveManagerData \(void\)](#)
- [TPM\\_RESULT VTPM\\_LoadManagerData \(void\)](#)

- `TPM_RESULT VTPM_New_DMI_Extra` (`VTPM_DMI_RESOURCE *dmi_res`, `BYTE vm_type`, `BYTE startup_mode`)
- `TPM_RESULT VTPM_Close_DMI_Extra` (`VTPM_DMI_RESOURCE *dmi_res`)
- `TPM_RESULT close_dmi` (`VTPM_DMI_RESOURCE *dmi_res`)
- `TPM_RESULT init_dmi` (`UINT32 dmi_id`, `BYTE type`, `VTPM_DMI_RESOURCE **dmi_res`)
- `void free_dmi` (`VTPM_DMI_RESOURCE *dmi_res`)
- `TPM_RESULT envelope_encrypt` (`const buffer_t *inbuf`, `CRYPTO_INFO *asymkey`, `buffer_t *sealed_data`)
- `TPM_RESULT envelope_decrypt` (`const buffer_t *cipher`, `TCS_CONTEXT_HANDLE TCSCContext`, `TPM_HANDLE keyHandle`, `const TPM_AUTHDATA *key_usage_auth`, `buffer_t *unsealed_data`)

## Variables

- volatile `sig_atomic_t HANDLER_QUIT_FLAG`
- `VTPM_GLOBALS * vtpm_globals`
- `const TPM_AUTHDATA SRK_AUTH`

## 4.159.1 Macro Definition Documentation

- 4.159.1.1 `#define DMI_NVM_FILE "/var/vtpm/vtpm_dm_%d.data"`
- 4.159.1.2 `#define STATE_FILE "/var/vtpm/VTPM"`
- 4.159.1.3 `#define VTPM_CTL_DM 0`
- 4.159.1.4 `#define VTPM_MANAGER_GEN 2`

## 4.159.2 Typedef Documentation

- 4.159.2.1 `typedef struct VTPM_DMI_RESOURCE_T VTPM_DMI_RESOURCE`
- 4.159.2.2 `typedef struct tdVTPM_GLOBALS VTPM_GLOBALS`
- 4.159.2.3 `typedef struct tdVTPM_MIGKEY_LIST VTPM_MIGKEY_LIST`

## 4.159.3 Function Documentation

- 4.159.3.1 `TPM_RESULT close_dmi ( VTPM_DMI_RESOURCE * dmi_res )`
- 4.159.3.2 `TPM_RESULT envelope_decrypt ( const buffer_t * cipher, TCS_CONTEXT_HANDLE TCSCContext, TPM_HANDLE keyHandle, const TPM_AUTHDATA * key_usage_auth, buffer_t * unsealed_data )`
- 4.159.3.3 `TPM_RESULT envelope_encrypt ( const buffer_t * inbuf, CRYPTO_INFO * asymkey, buffer_t * sealed_data )`
- 4.159.3.4 `void free_dmi ( VTPM_DMI_RESOURCE * dmi_res )`
- 4.159.3.5 `TPM_RESULT init_dmi ( UINT32 dmi_id, BYTE type, VTPM_DMI_RESOURCE ** dmi_res )`
- 4.159.3.6 `TPM_RESULT VTPM_Close_DMI_Extra ( VTPM_DMI_RESOURCE * dmi_res )`
- 4.159.3.7 `TPM_RESULT VTPM_Handle_Close_DMI ( const buffer_t * param_buf )`
- 4.159.3.8 `TPM_RESULT VTPM_Handle_Delete_DMI ( const buffer_t * param_buf )`

- 4.159.3.9 **TPM\_RESULT** `VTPM_Handle_Get_Migration_key ( const buffer_t * param_buf, buffer_t * result_buf )`
- 4.159.3.10 **TPM\_RESULT** `VTPM_Handle_Get_NVM_Size ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.159.3.11 **TPM\_RESULT** `VTPM_Handle_Load_NVM ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.159.3.12 **TPM\_RESULT** `VTPM_Handle_Migrate_In ( const buffer_t * param_buf, buffer_t * result_buf )`
- 4.159.3.13 **TPM\_RESULT** `VTPM_Handle_Migrate_Out ( const buffer_t * param_buf, buffer_t * result_buf )`
- 4.159.3.14 **TPM\_RESULT** `VTPM_Handle_New_DMI ( const buffer_t * param_buf )`
- 4.159.3.15 **TPM\_RESULT** `VTPM_Handle_Save_NVM ( VTPM_DMI_RESOURCE * myDMI, const buffer_t * inbuf, buffer_t * outbuf )`
- 4.159.3.16 **TPM\_RESULT** `VTPM_Handle_TPM_Command ( VTPM_DMI_RESOURCE * dmi, buffer_t * inbuf, buffer_t * outbuf )`
- 4.159.3.17 **TPM\_RESULT** `VTPM_Init_Manager ( )`
- 4.159.3.18 **TPM\_RESULT** `VTPM_LoadManagerData ( void )`
- 4.159.3.19 **TPM\_RESULT** `VTPM_Manager_Handler ( vtpm_ipc_handle_t * tx_ipc_h, vtpm_ipc_handle_t * rx_ipc_h, BOOL fw_tpm, vtpm_ipc_handle_t * fw_tx_ipc_h, vtpm_ipc_handle_t * fw_rx_ipc_h, BOOL is_priv, char * client_name )`
- 4.159.3.20 **TPM\_RESULT** `VTPM_New_DMI_Extra ( VTPM_DMI_RESOURCE * dmi_res, BYTE vm_type, BYTE startup_mode )`
- 4.159.3.21 **TPM\_RESULT** `VTPM_SaveManagerData ( void )`
- 4.159.3.22 `void VTPM_Stop_Manager ( )`

#### 4.159.4 Variable Documentation

- 4.159.4.1 `volatile sig_atomic_t HANDLER_QUIT_FLAG`
- 4.159.4.2 `const TPM_AUTHDATA SRK_AUTH`
- 4.159.4.3 `VTPM_GLOBALS* vtpm_globals`

### 4.160 xen/tools/vtpm\_manager/manager/vtsp.c File Reference

```
#include <string.h>
#include "tcg.h"
#include "tcs.h"
#include "bsg.h"
#include "log.h"
#include "crypto.h"
#include "vtsp.h"
#include "buffer.h"
```

## Macros

- `#define RSA_KEY_SIZE 0x0800`

## Functions

- `TPM_RESULT GenerateAuth` (const `BYTE` \*inParamDigestText, `UINT32` inParamDigestTextSize, const `TPM_SECRET` \*HMACkey, `TCS_AUTH` \*auth)
- `TPM_RESULT VerifyAuth` (const `BYTE` \*outParamDigestText, `UINT32` outParamDigestTextSize, const `TPM_SECRET` \*HMACkey, `TCS_AUTH` \*auth, `TCS_CONTEXT_HANDLE` hContext)
- `TPM_RESULT VTSP_OIAP` (const `TCS_CONTEXT_HANDLE` hContext, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_OSAP` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_ENTITY_TYPE` entityType, const `UINT32` entityValue, const `TPM_AUTHDATA` \*usageAuth, `TPM_SECRET` \*sharedSecret, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_TerminateHandle` (const `TCS_CONTEXT_HANDLE` hContext, const `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_ReadPubek` (const `TCS_CONTEXT_HANDLE` hContext, `CRYPTO_INFO` \*crypto\_info)
- `TPM_RESULT VTSP_TakeOwnership` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_AUTHDATA` \*ownerAuth, const `TPM_AUTHDATA` \*srkAuth, `CRYPTO_INFO` \*ek\_cryptoInfo, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_DisablePubekRead` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_AUTHDATA` \*ownerAuth, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_CreateWrapKey` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_KEY_USAGE` usage, const `TPM_AUTHDATA` \*newKeyAuth, const `TCS_KEY_HANDLE` parentHandle, const `TPM_AUTHDATA` \*osapSharedSecret, `buffer_t` \*pubKeyBuf, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_LoadKey` (const `TCS_CONTEXT_HANDLE` hContext, const `TCS_KEY_HANDLE` hUnwrappingKey, const `buffer_t` \*rgbWrappedKeyBlob, const `TPM_AUTHDATA` \*parentAuth, `TPM_HANDLE` \*newKeyHandle, `TCS_AUTH` \*auth, `CRYPTO_INFO` \*cryptoinfo, const `BOOL` skipTPMLoad)
- `TPM_RESULT VTSP_Unbind` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_KEY_HANDLE` key\_handle, const `buffer_t` \*bound\_data, const `TPM_AUTHDATA` \*usage\_auth, `buffer_t` \*clear\_data, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_Bind` (`CRYPTO_INFO` \*cryptoInfo, const `buffer_t` \*inData, `buffer_t` \*outData)
- `TPM_RESULT VTSP_Seal` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_KEY_HANDLE` keyHandle, const `TPM_AUTHDATA` \*sealDataAuth, const `TPM_PCR_COMPOSITE` \*pcrComp, const `buffer_t` \*inData, `TPM_STORED_DATA` \*sealedData, const `TPM_SECRET` \*osapSharedSecret, `TCS_AUTH` \*auth)
- `TPM_RESULT VTSP_Unseal` (const `TCS_CONTEXT_HANDLE` hContext, const `TPM_KEY_HANDLE` keyHandle, const `TPM_STORED_DATA` \*sealedData, const `TPM_AUTHDATA` \*key\_usage\_auth, const `TPM_AUTHDATA` \*data\_usage\_auth, `buffer_t` \*outData, `TCS_AUTH` \*auth, `TCS_AUTH` \*dataAuth)
- `TPM_RESULT VTSP_SaveState` (const `TCS_CONTEXT_HANDLE` hContext)
- `TPM_RESULT VTSP_RawTransmit` (const `TCS_CONTEXT_HANDLE` hContext, const `buffer_t` \*inbuf, `buffer_t` \*outbuf)

### 4.160.1 Macro Definition Documentation

#### 4.160.1.1 `#define RSA_KEY_SIZE 0x0800`

### 4.160.2 Function Documentation

#### 4.160.2.1 `TPM_RESULT GenerateAuth` ( const `BYTE` \* *inParamDigestText*, `UINT32` *inParamDigestTextSize*, const `TPM_SECRET` \* *HMACkey*, `TCS_AUTH` \* *auth* )

#### 4.160.2.2 `TPM_RESULT VerifyAuth` ( const `BYTE` \* *outParamDigestText*, `UINT32` *outParamDigestTextSize*, const `TPM_SECRET` \* *HMACkey*, `TCS_AUTH` \* *auth*, `TCS_CONTEXT_HANDLE` *hContext* )

#### 4.160.2.3 `TPM_RESULT VTSP_Bind` ( `CRYPTO_INFO` \* *cryptoInfo*, const `buffer_t` \* *inData*, `buffer_t` \* *outData* )

- 4.160.2.4 **TPM\_RESULT VTSP\_CreateWrapKey** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_KEY\_USAGE usage*, *const TPM\_AUTHDATA \* newKeyAuth*, *const TCS\_KEY\_HANDLE parentHandle*, *const TPM\_AUTHDATA \* osapSharedSecret*, *buffer\_t \* pubKeyBuf*, *TCS\_AUTH \* auth* )
- 4.160.2.5 **TPM\_RESULT VTSP\_DisablePubekRead** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_AUTHDATA \* ownerAuth*, *TCS\_AUTH \* auth* )
- 4.160.2.6 **TPM\_RESULT VTSP\_LoadKey** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TCS\_KEY\_HANDLE hUnwrappingKey*, *const buffer\_t \* rgbWrappedKeyBlob*, *const TPM\_AUTHDATA \* parentAuth*, *TPM\_HANDLE \* newKeyHandle*, *TCS\_AUTH \* auth*, *CRYPTO\_INFO \* cryptoinfo*, *const BOOL skipTPMLoad* )
- 4.160.2.7 **TPM\_RESULT VTSP\_OIAP** ( *const TCS\_CONTEXT\_HANDLE hContext*, *TCS\_AUTH \* auth* )
- 4.160.2.8 **TPM\_RESULT VTSP\_OSAP** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_ENTITY\_TYPE entityType*, *const UINT32 entityValue*, *const TPM\_AUTHDATA \* usageAuth*, *TPM\_SECRET \* sharedSecret*, *TCS\_AUTH \* auth* )
- 4.160.2.9 **TPM\_RESULT VTSP\_RawTransmit** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const buffer\_t \* inbuf*, *buffer\_t \* outbuf* )
- 4.160.2.10 **TPM\_RESULT VTSP\_ReadPubek** ( *const TCS\_CONTEXT\_HANDLE hContext*, *CRYPTO\_INFO \* crypto\_info* )
- 4.160.2.11 **TPM\_RESULT VTSP\_SaveState** ( *const TCS\_CONTEXT\_HANDLE hContext* )
- 4.160.2.12 **TPM\_RESULT VTSP\_Seal** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_KEY\_HANDLE keyHandle*, *const TPM\_AUTHDATA \* sealDataAuth*, *const TPM\_PCR\_COMPOSITE \* pcrComp*, *const buffer\_t \* inData*, *TPM\_STORED\_DATA \* sealedData*, *const TPM\_SECRET \* osapSharedSecret*, *TCS\_AUTH \* auth* )
- 4.160.2.13 **TPM\_RESULT VTSP\_TakeOwnership** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_AUTHDATA \* ownerAuth*, *const TPM\_AUTHDATA \* srkAuth*, *CRYPTO\_INFO \* ek\_cryptoinfo*, *TCS\_AUTH \* auth* )
- 4.160.2.14 **TPM\_RESULT VTSP\_TerminateHandle** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TCS\_AUTH \* auth* )
- 4.160.2.15 **TPM\_RESULT VTSP\_Unbind** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_KEY\_HANDLE key\_handle*, *const buffer\_t \* bound\_data*, *const TPM\_AUTHDATA \* usage\_auth*, *buffer\_t \* clear\_data*, *TCS\_AUTH \* auth* )
- 4.160.2.16 **TPM\_RESULT VTSP\_Unseal** ( *const TCS\_CONTEXT\_HANDLE hContext*, *const TPM\_KEY\_HANDLE keyHandle*, *const TPM\_STORED\_DATA \* sealedData*, *const TPM\_AUTHDATA \* key\_usage\_auth*, *const TPM\_AUTHDATA \* data\_usage\_auth*, *buffer\_t \* outData*, *TCS\_AUTH \* auth*, *TCS\_AUTH \* dataAuth* )

## 4.161 xen/tools/vtpm\_manager/migration/vtpm\_manager\_if.c File Reference

```
#include <stdio.h>
#include <fcntl.h>
#include <malloc.h>
#include <string.h>
#include "tcg.h"
#include "buffer.h"
#include "log.h"
#include "vtpm_ipc.h"
#include "bsg.h"
#include "vtpm_migrator.h"
#include "vtpm_manager.h"
```

### Functions

- [TPM\\_RESULT vtpm\\_manager\\_open \(\)](#)
- [void vtpm\\_manager\\_close \(\)](#)
- [TPM\\_RESULT vtpm\\_manager\\_command \(TPM\\_COMMAND\\_CODE ord, buffer\\_t \\*command\\_param\\_buf, TPM\\_RESULT \\*cmd\\_status, buffer\\_t \\*result\\_param\\_buf\)](#)

#### 4.161.1 Function Documentation

4.161.1.1 [void vtpm\\_manager\\_close \( \)](#)

4.161.1.2 [TPM\\_RESULT vtpm\\_manager\\_command \( TPM\\_COMMAND\\_CODE ord, buffer\\_t \\* command\\_param\\_buf, TPM\\_RESULT \\* cmd\\_status, buffer\\_t \\* result\\_param\\_buf \)](#)

4.161.1.3 [TPM\\_RESULT vtpm\\_manager\\_open \( \)](#)

## 4.162 xen/tools/vtpm\_manager/tcs/contextmgr.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <malloc.h>
#include "tcs.h"
#include "contextmgr.h"
#include "log.h"
#include "hashtable.h"
```

### Functions

- [BYTE \\* AddMemBlock \(CONTEXT\\_HANDLE \\*pContextHandle, int BlockSize\)](#)
- [BOOL DeleteMemBlock \(CONTEXT\\_HANDLE \\*pContextHandle, BYTE \\*pTCPA\\_BYTES\)](#)
- [BOOL AddHandleToList \(TCS\\_CONTEXT\\_HANDLE hContext, TPM\\_RESOURCE\\_TYPE type, TPM\\_HANDLE handle\)](#)
- [BOOL DeleteHandleFromList \(TCS\\_CONTEXT\\_HANDLE hContext, TPM\\_HANDLE handle\)](#)
- [BOOL FreeHandleList \(CONTEXT\\_HANDLE \\*pContextHandle\)](#)

### 4.162.1 Function Documentation

4.162.1.1 **BOOL** AddHandleToList ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_RESOURCE\_TYPE** *type*, **TPM\_HANDLE** *handle* )

4.162.1.2 **BYTE\*** AddMemBlock ( **CONTEXT\_HANDLE** \* *pContextHandle*, **int** *BlockSize* )

4.162.1.3 **BOOL** DeleteHandleFromList ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_HANDLE** *handle* )

4.162.1.4 **BOOL** DeleteMemBlock ( **CONTEXT\_HANDLE** \* *pContextHandle*, **BYTE** \* *pTCPA\_BYTES* )

4.162.1.5 **BOOL** FreeHandleList ( **CONTEXT\_HANDLE** \* *pContextHandle* )

## 4.163 xen/tools/vtpm\_manager/tcs/tcs.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <malloc.h>
#include "tcg.h"
#include "bsg.h"
#include "tcs.h"
#include "contextmgr.h"
#include "tpmddl.h"
#include "log.h"
#include "hashtable.h"
#include "hashtable_itr.h"
```

### Macros

- `#define TCPA_MAX_BUFFER_LENGTH 0x2000`

### Functions

- **CONTEXT\_HANDLE** \* [LookupContext](#) (**TCS\_CONTEXT\_HANDLE** *hContext*)
- **TPM\_RESULT** [TCS\\_create](#) ()
- **void** [TCS\\_destroy](#) ()
- **TPM\_RESULT** [TCS\\_Malloc](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **UINT32** *MemSize*, **BYTE** \*\**ppMemPtr*)
- **TPM\_RESULT** [TCS\\_FreeMemory](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **BYTE** \**pMemory*)
- **TPM\_RESULT** [TCS\\_OpenContext](#) (**TCS\_CONTEXT\_HANDLE** \**hContext*)
- **TPM\_RESULT** [TCS\\_CloseContext](#) (**TCS\_CONTEXT\_HANDLE** *hContext*)
- **int** [packAuth](#) (**BYTE** \**dst*, **TCS\_AUTH** \**auth*)
- **int** [unpackAuth](#) (**TCS\_AUTH** \**auth*, **BYTE** \**src*)
- **TPM\_RESULT** [TCSP\\_OIAP](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_AUTHHANDLE** \**authHandle*, **TPM\_NONCE** \**nonce0*)
- **TPM\_RESULT** [TCSP\\_OSAP](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_ENTITY\_TYPE** *entityType*, **UINT32** *entityValue*, **TPM\_NONCE** *nonceOddOSAP*, **TCS\_AUTHHANDLE** \**authHandle*, **TPM\_NONCE** \**nonceEven*, **TPM\_NONCE** \**nonceEvenOSAP*)
- **TPM\_RESULT** [TCSP\\_TakeOwnership](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **UINT16** *protocolID*, **UINT32** *encOwnerAuthSize*, **BYTE** \**encOwnerAuth*, **UINT32** *encSrAuthSize*, **BYTE** \**encSrAuth*, **UINT32** \**SrkSize*, **BYTE** \*\**Srk*, **TCS\_AUTH** \**ownerAuth*)
- **TPM\_RESULT** [TCSP\\_DisablePubekRead](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_AUTH** \**ownerAuth*)
- **TPM\_RESULT** [TCSP\\_TerminateHandle](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_AUTHHANDLE** *handle*)
- **TPM\_RESULT** [TCSP\\_Extend](#) (**TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_PCRINDEX** *pcrNum*, **TPM\_DIGEST** *inDigest*, **TPM\_PCRVALUE** \**outDigest*)



- **TPM\_RESULT TCSP\_Seal** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** keyHandle, **TPM\_ENCAUTH** encAuth, **UINT32** pcrInfoSize, **BYTE** \*PcrInfo, **UINT32** inDataSize, **BYTE** \*inData, **TCS\_AUTH** \*pubAuth, **UINT32** \*SealedDataSize, **BYTE** \*\*SealedData)
- **TPM\_RESULT TCSP\_Unseal** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** parentHandle, **UINT32** SealedDataSize, **BYTE** \*SealedData, **TCS\_AUTH** \*parentAuth, **TCS\_AUTH** \*dataAuth, **UINT32** \*DataSize, **BYTE** \*\*Data)
- **TPM\_RESULT TCSP\_UnBind** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** keyHandle, **UINT32** inDataSize, **BYTE** \*inData, **TCS\_AUTH** \*privAuth, **UINT32** \*outDataSize, **BYTE** \*\*outData)
- **TPM\_RESULT TCSP\_CreateWrapKey** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** hWrappingKey, **TPM\_ENCAUTH** KeyUsageAuth, **TPM\_ENCAUTH** KeyMigrationAuth, **UINT32** \*pcKeySize, **BYTE** \*\*prgbKey, **TCS\_AUTH** \*pAuth)
- **TPM\_RESULT TCSP\_LoadKeyByBlob** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** hUnwrappingKey, **UINT32** cWrappedKeyBlobSize, **BYTE** \*rgbWrappedKeyBlob, **TCS\_AUTH** \*pAuth, **TCS\_KEY\_HANDLE** \*phKeyTCSI, **TCS\_KEY\_HANDLE** \*phKeyHMAC)
- **TPM\_RESULT TCSP\_EvictKey** (**TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** hKey)
- **TPM\_RESULT TCSP\_GetRandom** (**TCS\_CONTEXT\_HANDLE** hContext, **UINT32** \*bytesRequested, **BYTE** \*\*randomBytes)
- **TPM\_RESULT TCSP\_ReadPubek** (**TCS\_CONTEXT\_HANDLE** hContext, **TPM\_NONCE** antiReplay, **UINT32** \*pubEndorsementKeySize, **BYTE** \*\*pubEndorsementKey, **TPM\_DIGEST** \*checksum)
- **TPM\_RESULT TCSP\_SaveState** (**TCS\_CONTEXT\_HANDLE** hContext)
- **TPM\_RESULT TCSP\_RawTransmitData** (**UINT32** inDataSize, **BYTE** \*inData, **UINT32** \*outDataSize, **BYTE** \*outData)

## Variables

- struct hashtable \* **context\_ht**

### 4.163.1 Macro Definition Documentation

- 4.163.1.1 **#define** TCS\_MAX\_BUFFER\_LENGTH 0x2000

### 4.163.2 Function Documentation

- 4.163.2.1 **CONTEXT\_HANDLE\*** LookupContext ( **TCS\_CONTEXT\_HANDLE** hContext )
- 4.163.2.2 **int** packAuth ( **BYTE** \* dst, **TCS\_AUTH** \* auth )
- 4.163.2.3 **TPM\_RESULT** TCS\_CloseContext ( **TCS\_CONTEXT\_HANDLE** hContext )
- 4.163.2.4 **TPM\_RESULT** TCS\_create ( )
- 4.163.2.5 **void** TCS\_destroy ( )
- 4.163.2.6 **TPM\_RESULT** TCS\_FreeMemory ( **TCS\_CONTEXT\_HANDLE** hContext, **BYTE** \* pMemory )
- 4.163.2.7 **TPM\_RESULT** TCS\_Malloc ( **TCS\_CONTEXT\_HANDLE** hContext, **UINT32** MemSize, **BYTE** \*\* ppMemPtr )
- 4.163.2.8 **TPM\_RESULT** TCS\_OpenContext ( **TCS\_CONTEXT\_HANDLE** \* hContext )
- 4.163.2.9 **TPM\_RESULT** TCSP\_CreateWrapKey ( **TCS\_CONTEXT\_HANDLE** hContext, **TCS\_KEY\_HANDLE** hWrappingKey, **TPM\_ENCAUTH** KeyUsageAuth, **TPM\_ENCAUTH** KeyMigrationAuth, **UINT32** \* pcKeySize, **BYTE** \*\* prgbKey, **TCS\_AUTH** \* pAuth )
- 4.163.2.10 **TPM\_RESULT** TCSP\_DisablePubekRead ( **TCS\_CONTEXT\_HANDLE** hContext, **TCS\_AUTH** \* ownerAuth )

- 4.163.2.11 **TPM\_RESULT** TCSP\_EvictKey ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_KEY\_HANDLE** *hKey* )
- 4.163.2.12 **TPM\_RESULT** TCSP\_Extend ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_PCRINDEX** *pcrNum*, **TPM\_DIGEST** *inDigest*, **TPM\_PCRVALUE** \* *outDigest* )
- 4.163.2.13 **TPM\_RESULT** TCSP\_GetRandom ( **TCS\_CONTEXT\_HANDLE** *hContext*, **UINT32** \* *bytesRequested*, **BYTE** \*\* *randomBytes* )
- 4.163.2.14 **TPM\_RESULT** TCSP\_LoadKeyByBlob ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_KEY\_HANDLE** *hUnwrappingKey*, **UINT32** *cWrappedKeyBlobSize*, **BYTE** \* *rgbWrappedKeyBlob*, **TCS\_AUTH** \* *pAuth*, **TCS\_KEY\_HANDLE** \* *phKeyTCSI*, **TCS\_KEY\_HANDLE** \* *phKeyHMAC* )
- 4.163.2.15 **TPM\_RESULT** TCSP\_OIAP ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_AUTHHANDLE** \* *authHandle*, **TPM\_NONCE** \* *nonce0* )
- 4.163.2.16 **TPM\_RESULT** TCSP\_OSAP ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_ENTITY\_TYPE** *entityType*, **UINT32** *entityValue*, **TPM\_NONCE** *nonceOddOSAP*, **TCS\_AUTHHANDLE** \* *authHandle*, **TPM\_NONCE** \* *nonceEven*, **TPM\_NONCE** \* *nonceEvenOSAP* )
- 4.163.2.17 **TPM\_RESULT** TCSP\_RawTransmitData ( **UINT32** *inDataSize*, **BYTE** \* *inData*, **UINT32** \* *outDataSize*, **BYTE** \* *outData* )
- 4.163.2.18 **TPM\_RESULT** TCSP\_ReadPubek ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TPM\_NONCE** *antiReplay*, **UINT32** \* *pubEndorsementKeySize*, **BYTE** \*\* *pubEndorsementKey*, **TPM\_DIGEST** \* *checksum* )
- 4.163.2.19 **TPM\_RESULT** TCSP\_SaveState ( **TCS\_CONTEXT\_HANDLE** *hContext* )
- 4.163.2.20 **TPM\_RESULT** TCSP\_Seal ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_KEY\_HANDLE** *keyHandle*, **TPM\_ENCAUTH** *encAuth*, **UINT32** *pcrInfoSize*, **BYTE** \* *PcrInfo*, **UINT32** *inDataSize*, **BYTE** \* *inData*, **TCS\_AUTH** \* *pubAuth*, **UINT32** \* *SealedDataSize*, **BYTE** \*\* *SealedData* )
- 4.163.2.21 **TPM\_RESULT** TCSP\_TakeOwnership ( **TCS\_CONTEXT\_HANDLE** *hContext*, **UINT16** *protocolID*, **UINT32** *encOwnerAuthSize*, **BYTE** \* *encOwnerAuth*, **UINT32** *encSrAuthSize*, **BYTE** \* *encSrAuth*, **UINT32** \* *SrkSize*, **BYTE** \*\* *Srk*, **TCS\_AUTH** \* *ownerAuth* )
- 4.163.2.22 **TPM\_RESULT** TCSP\_TerminateHandle ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_AUTHHANDLE** *handle* )
- 4.163.2.23 **TPM\_RESULT** TCSP\_UnBind ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_KEY\_HANDLE** *keyHandle*, **UINT32** *inDataSize*, **BYTE** \* *inData*, **TCS\_AUTH** \* *privAuth*, **UINT32** \* *outDataSize*, **BYTE** \*\* *outData* )
- 4.163.2.24 **TPM\_RESULT** TCSP\_Unseal ( **TCS\_CONTEXT\_HANDLE** *hContext*, **TCS\_KEY\_HANDLE** *parentHandle*, **UINT32** *SealedDataSize*, **BYTE** \* *SealedData*, **TCS\_AUTH** \* *parentAuth*, **TCS\_AUTH** \* *dataAuth*, **UINT32** \* *DataSize*, **BYTE** \*\* *Data* )
- 4.163.2.25 **int** unpackAuth ( **TCS\_AUTH** \* *auth*, **BYTE** \* *src* )

### 4.163.3 Variable Documentation

- 4.163.3.1 **struct hashtable\*** context\_ht

## 4.164 xen/tools/vtpm\_manager/tcs/tpmddl.c File Reference

```
#include <string.h>
#include "tpmddl.h"
#include "tcs.h"
#include "bsg.h"
#include "log.h"
```

### Macros

- #define [MIN\(X, Y\)](#) ((X) < (Y) ? (X) : (Y))

### Functions

- [TDDL\\_RESULT TDDL\\_GetCapability](#) ([TDDL\\_UINT32 cap](#), [TDDL\\_UINT32 sub](#), [TDDL\\_BYTE \\*buffer](#), [TDDL\\_UINT32 \\*size](#))
- [TDDL\\_RESULT TDDL\\_FlushSpecific](#) ([TDDL\\_UINT32 handle](#), [TDDL\\_UINT32 res](#))

#### 4.164.1 Macro Definition Documentation

4.164.1.1 #define [MIN\( X, Y \)](#) ((X) < (Y) ? (X) : (Y))

#### 4.164.2 Function Documentation

4.164.2.1 [TDDL\\_RESULT TDDL\\_FlushSpecific](#) ( [TDDL\\_UINT32 handle](#), [TDDL\\_UINT32 res](#) )

4.164.2.2 [TDDL\\_RESULT TDDL\\_GetCapability](#) ( [TDDL\\_UINT32 cap](#), [TDDL\\_UINT32 sub](#), [TDDL\\_BYTE \\* buffer](#), [TDDL\\_UINT32 \\* size](#) )

## 4.165 xen/tools/vtpm\_manager/tcs/tpmddl.h File Reference

### Macros

- #define [TDDL\\_CAP\\_PROP\\_MANUFACTURER](#) 0x0001
- #define [TDDL\\_E\\_FAIL](#) 1
- #define [TDDL\\_E\\_SUCCESS](#) 0
- #define [TDDL\\_SUCCESS](#) 0

### Typedefs

- typedef unsigned int [TDDL\\_UINT32](#)
- typedef [TDDL\\_UINT32](#) [TDDL\\_RESULT](#)
- typedef unsigned char [TDDL\\_BYTE](#)

### Functions

- [TDDL\\_RESULT TDDL\\_Open](#) ()
- void [TDDL\\_Close](#) ()
- [TDDL\\_RESULT TDDL\\_TransmitData](#) ([TDDL\\_BYTE \\*in](#), [TDDL\\_UINT32 insize](#), [TDDL\\_BYTE \\*out](#), [TDDL\\_UINT32 \\*outsize](#))
- [TDDL\\_RESULT TDDL\\_GetStatus](#) ()

- `TDDL_RESULT TDDL_GetCapability (TDDL_UINT32 cap, TDDL_UINT32 sub, TDDL_BYTE *buffer, TDDL_UINT32 *size)`
- `TDDL_RESULT TDDL_SetCapability (TDDL_UINT32 cap, TDDL_UINT32 sub, TDDL_BYTE *buffer, TDDL_UINT32 *size)`
- `TDDL_RESULT TDDL_FlushSpecific (TDDL_UINT32 handle, TDDL_UINT32 res)`

#### 4.165.1 Macro Definition Documentation

4.165.1.1 `#define TDDL_CAP_PROP_MANUFACTURER 0x0001`

4.165.1.2 `#define TDDL_E_FAIL 1`

4.165.1.3 `#define TDDL_E_SUCCESS 0`

4.165.1.4 `#define TDDL_SUCCESS 0`

#### 4.165.2 Typedef Documentation

4.165.2.1 `typedef unsigned char TDDL_BYTE`

4.165.2.2 `typedef TDDL_UINT32 TDDL_RESULT`

4.165.2.3 `typedef unsigned int TDDL_UINT32`

#### 4.165.3 Function Documentation

4.165.3.1 `void TDDL_Close ( )`

4.165.3.2 `TDDL_RESULT TDDL_FlushSpecific ( TDDL_UINT32 handle, TDDL_UINT32 res )`

4.165.3.3 `TDDL_RESULT TDDL_GetCapability ( TDDL_UINT32 cap, TDDL_UINT32 sub, TDDL_BYTE * buffer, TDDL_UINT32 * size )`

4.165.3.4 `TDDL_RESULT TDDL_GetStatus ( )`

4.165.3.5 `TDDL_RESULT TDDL_Open ( )`

4.165.3.6 `TDDL_RESULT TDDL_SetCapability ( TDDL_UINT32 cap, TDDL_UINT32 sub, TDDL_BYTE * buffer, TDDL_UINT32 * size )`

4.165.3.7 `TDDL_RESULT TDDL_TransmitData ( TDDL_BYTE * in, TDDL_UINT32 insize, TDDL_BYTE * out, TDDL_UINT32 * outsize )`

### 4.166 xen/tools/vtpm\_manager/tcs/transmit.c File Reference

```
#include <unistd.h>
#include <stdio.h>
#include <errno.h>
#include <fcntl.h>
#include "tcg.h"
#include "buffer.h"
#include "log.h"
#include "tpmddl.h"
```

## Macros

- `#define TPM_TX_FNAME "/dev/tpm0"`

## Functions

- `TPM_RESULT TDDL_TransmitData (TDDL_BYTE *in, TDDL_UINT32 insize, TDDL_BYTE *out, TDDL_UINT32 *outsize)`
- `TPM_RESULT TDDL_Open ()`
- `void TDDL_Close ()`

### 4.166.1 Macro Definition Documentation

4.166.1.1 `#define TPM_TX_FNAME "/dev/tpm0"`

### 4.166.2 Function Documentation

4.166.2.1 `void TDDL_Close ( )`

4.166.2.2 `TPM_RESULT TDDL_Open ( )`

4.166.2.3 `TPM_RESULT TDDL_TransmitData ( TDDL_BYTE * in, TDDL_UINT32 insize, TDDL_BYTE * out, TDDL_UINT32 * outsize )`

## 4.167 xen/tools/vtpm\_manager/util/buffer.c File Reference

```
#include <stdarg.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <sys/param.h>
#include "tcg.h"
#include "log.h"
#include "bsg.h"
#include "buffer.h"
```

## Functions

- `TPM_RESULT buffer_init (buffer_t *buf, tpm_size_t initsize, const BYTE *initval)`
- `TPM_RESULT buffer_init_convert (buffer_t *buf, tpm_size_t initsize, BYTE *initval)`
- `TPM_RESULT buffer_init_copy (buffer_t *buf, const buffer_t *src)`
- `TPM_RESULT buffer_init_const (buffer_t *buf, tpm_size_t size, const BYTE *val)`
- `TPM_RESULT buffer_init_alias (buffer_t *buf, const buffer_t *b, tpm_size_t offset, tpm_size_t len)`
- `TPM_RESULT buffer_init_alias_convert (buffer_t *buf, tpm_size_t size, BYTE *val)`
- `TPM_RESULT buffer_copy (buffer_t *dest, const buffer_t *src)`
- `BOOL buffer_eq (const buffer_t *a, const buffer_t *b)`
- `void buffer_memset (buffer_t *buf, BYTE b)`
- `TPM_RESULT buffer_append_raw (buffer_t *buf, tpm_size_t len, const BYTE *bytes)`
- `TPM_RESULT buffer_prepend_raw (buffer_t *buf, tpm_size_t len, const BYTE *bytes)`
- `tpm_size_t buffer_len (const buffer_t *buf)`
- `TPM_RESULT buffer_free (buffer_t *buf)`

### 4.167.1 Function Documentation

- 4.167.1.1 `TPM_RESULT buffer_append_raw ( buffer_t * buf, tpm_size_t len, const BYTE * bytes )`
- 4.167.1.2 `TPM_RESULT buffer_copy ( buffer_t * dest, const buffer_t * src )`
- 4.167.1.3 `BOOL buffer_eq ( const buffer_t * a, const buffer_t * b )`
- 4.167.1.4 `TPM_RESULT buffer_free ( buffer_t * buf )`
- 4.167.1.5 `TPM_RESULT buffer_init ( buffer_t * buf, tpm_size_t initsize, const BYTE * initval )`
- 4.167.1.6 `TPM_RESULT buffer_init_alias ( buffer_t * buf, const buffer_t * b, tpm_size_t offset, tpm_size_t len )`
- 4.167.1.7 `TPM_RESULT buffer_init_alias_convert ( buffer_t * buf, tpm_size_t size, BYTE * val )`
- 4.167.1.8 `TPM_RESULT buffer_init_const ( buffer_t * buf, tpm_size_t size, const BYTE * val )`
- 4.167.1.9 `TPM_RESULT buffer_init_convert ( buffer_t * buf, tpm_size_t initsize, BYTE * initval )`
- 4.167.1.10 `TPM_RESULT buffer_init_copy ( buffer_t * buf, const buffer_t * src )`
- 4.167.1.11 `tpm_size_t buffer_len ( const buffer_t * buf )`
- 4.167.1.12 `void buffer_memset ( buffer_t * buf, BYTE b )`
- 4.167.1.13 `TPM_RESULT buffer_prepend_raw ( buffer_t * buf, tpm_size_t len, const BYTE * bytes )`

## 4.168 xen/tools/vtpm\_manager/util/buffer.h File Reference

```
#include <stddef.h>
#include "tcg.h"
```

### Data Structures

- struct [buffer\\_t](#)

### Macros

- #define [NULL\\_BUF](#) {0,0,0,0}

### Typedefs

- typedef [UINT32 tpm\\_size\\_t](#)

### Functions

- `TPM_RESULT buffer_init (buffer_t *buf, tpm\_size\_t initsize, const BYTE *initval)`
- `TPM_RESULT buffer_init_convert (buffer_t *buf, tpm\_size\_t initsize, BYTE *initval)`
- `TPM_RESULT buffer_init_const (buffer_t *buf, tpm\_size\_t size, const BYTE *val)`
- `TPM_RESULT buffer_init_alias (buffer_t *buf, const buffer\_t *b, tpm\_size\_t offset, tpm\_size\_t)`
- `TPM_RESULT buffer_init_alias_convert (buffer_t *buf, tpm\_size\_t size, BYTE *val)`

- [TPM\\_RESULT buffer\\_init\\_copy](#) ([buffer\\_t](#) \*buf, const [buffer\\_t](#) \*src)
- [TPM\\_RESULT buffer\\_copy](#) ([buffer\\_t](#) \*dest, const [buffer\\_t](#) \*src)
- [BOOL buffer\\_eq](#) (const [buffer\\_t](#) \*a, const [buffer\\_t](#) \*b)
- void [buffer\\_memset](#) ([buffer\\_t](#) \*buf, [BYTE](#) b)
- [tpm\\_size\\_t](#) [buffer\\_len](#) (const [buffer\\_t](#) \*buf)
- [TPM\\_RESULT](#) [buffer\\_free](#) ([buffer\\_t](#) \*buf)
- [TPM\\_RESULT](#) [buffer\\_append\\_raw](#) ([buffer\\_t](#) \*buf, [tpm\\_size\\_t](#) len, const [BYTE](#) \*bytes)
- [TPM\\_RESULT](#) [buffer\\_prepend\\_raw](#) ([buffer\\_t](#) \*buf, [tpm\\_size\\_t](#) len, const [BYTE](#) \*bytes)

## 4.168.1 Macro Definition Documentation

4.168.1.1 `#define NULL_BUF {0,0,0,0}`

## 4.168.2 Typedef Documentation

4.168.2.1 `typedef UINT32 tpm_size_t`

## 4.168.3 Function Documentation

4.168.3.1 `TPM_RESULT buffer_append_raw ( buffer\_t * buf, tpm\_size\_t len, const BYTE * bytes )`

4.168.3.2 `TPM_RESULT buffer_copy ( buffer\_t * dest, const buffer\_t * src )`

4.168.3.3 `BOOL buffer_eq ( const buffer\_t * a, const buffer\_t * b )`

4.168.3.4 `TPM_RESULT buffer_free ( buffer\_t * buf )`

4.168.3.5 `TPM_RESULT buffer_init ( buffer\_t * buf, tpm\_size\_t initsize, const BYTE * initval )`

4.168.3.6 `TPM_RESULT buffer_init_alias ( buffer\_t * buf, const buffer\_t * b, tpm\_size\_t offset, tpm\_size\_t )`

4.168.3.7 `TPM_RESULT buffer_init_alias_convert ( buffer\_t * buf, tpm\_size\_t size, BYTE * val )`

4.168.3.8 `TPM_RESULT buffer_init_const ( buffer\_t * buf, tpm\_size\_t size, const BYTE * val )`

4.168.3.9 `TPM_RESULT buffer_init_convert ( buffer\_t * buf, tpm\_size\_t initsize, BYTE * initval )`

4.168.3.10 `TPM_RESULT buffer_init_copy ( buffer\_t * buf, const buffer\_t * src )`

4.168.3.11 `tpm\_size\_t buffer_len ( const buffer\_t * buf )`

4.168.3.12 `void buffer_memset ( buffer\_t * buf, BYTE b )`

4.168.3.13 `TPM_RESULT buffer_prepend_raw ( buffer\_t * buf, tpm\_size\_t len, const BYTE * bytes )`

## 4.169 xen/tools/vtpm\_manager/util/tcg.h File Reference

```
#include <stdio.h>
```

### Data Structures

- struct [TPM\\_VERSION](#)
- struct [TPM\\_DIGEST](#)

- struct [TPM\\_NONCE](#)
- struct [TPM\\_KEY\\_PARMS](#)
- struct [TPM\\_RSA\\_KEY\\_PARMS](#)
- struct [TPM\\_STORE\\_PUBKEY](#)
- struct [TPM\\_PUBKEY](#)
- struct [TPM\\_KEY](#)
- struct [TPM\\_PCR\\_SELECTION](#)
- struct [TPM\\_PCR\\_COMPOSITE](#)
- struct [TPM\\_PCR\\_INFO](#)
- struct [TPM\\_BOUND\\_DATA](#)
- struct [TPM\\_STORED\\_DATA](#)
- struct [TCS\\_AUTH](#)
- struct [pack\\_buf\\_t](#)
- struct [pack\\_constbuf\\_t](#)

## Macros

- `#define TPM\_DIGEST\_SIZE 20`
- `#define NULL\_PACK\_BUF {0,0}`
- `#define TRUE 0x01`
- `#define FALSE 0x00`
- `#define TCPA\_MAX\_BUFFER\_LENGTH 0x2000`
- `#define TPM\_PROTECTED\_ORDINAL 0x00000000UL`
- `#define TPM\_UNPROTECTED\_ORDINAL 0x80000000UL`
- `#define TPM\_CONNECTION\_ORDINAL 0x40000000UL`
- `#define TPM\_VENDOR\_ORDINAL 0x20000000UL`
- `#define TPM\_ORD\_OIAP (10UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_OSAP (11UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ChangeAuth (12UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_TakeOwnership (13UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ChangeAuthAsymStart (14UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ChangeAuthAsymFinish (15UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ChangeAuthOwner (16UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_Extend (20UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_PcrRead (21UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_Quote (22UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_Seal (23UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_Unseal (24UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_DirWriteAuth (25UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_DirRead (26UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_UnBind (30UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_CreateWrapKey (31UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_LoadKey (32UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_GetPubKey (33UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_EvictKey (34UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_CreateMigrationBlob (40UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ReWrapKey (41UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ConvertMigrationBlob (42UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_AuthorizeMigrationKey (43UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_CreateMaintenanceArchive (44UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_LoadMaintenanceArchive (45UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_KillMaintenanceFeature (46UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_LoadManuMaintPub (47UL + TPM_PROTECTED_ORDINAL)`
- `#define TPM\_ORD\_ReadManuMaintPub (48UL + TPM_PROTECTED_ORDINAL)`



- #define `TPM_ORD_CertifyKey` (50UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_Sign` (60UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetRandom` (70UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_StirRandom` (71UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SelfTestFull` (80UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SelfTestStartup` (81UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_CertifySelfTest` (82UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_ContinueSelfTest` (83UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetTestResult` (84UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_Reset` (90UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_OwnerClear` (91UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_DisableOwnerClear` (92UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_ForceClear` (93UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_DisableForceClear` (94UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetCapabilitySigned` (100UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetCapability` (101UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetCapabilityOwner` (102UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_OwnerSetDisable` (110UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_PhysicalEnable` (111UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_PhysicalDisable` (112UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SetOwnerInstall` (113UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_PhysicalSetDeactivated` (114UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SetTempDeactivated` (115UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_CreateEndorsementKeyPair` (120UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_MakeIdentity` (121UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_ActivateIdentity` (122UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_ReadPubek` (124UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_OwnerReadPubek` (125UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_DisablePubekRead` (126UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetAuditEvent` (130UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetAuditEventSigned` (131UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_GetOrdinalAuditStatus` (140UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SetOrdinalAuditStatus` (141UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_Terminate_Handle` (150UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_Init` (151UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SaveState` (152UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_Startup` (153UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SetRedirection` (154UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SHA1Start` (160UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SHA1Update` (161UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SHA1Complete` (162UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SHA1CompleteExtend` (163UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_FieldUpgrade` (170UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SaveKeyContext` (180UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_LoadKeyContext` (181UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SaveAuthContext` (182UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_LoadAuthContext` (183UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_SaveContext` (184UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_LoadContext` (185UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_FlushSpecific` (186UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_PCR_Reset` (200UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_NV_DefineSpace` (204UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_NV_WriteValue` (205UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_NV_WriteValueAuth` (206UL + TPM\_PROTECTED\_ORDINAL)
- #define `TPM_ORD_NV_ReadValue` (207UL + TPM\_PROTECTED\_ORDINAL)

- #define [TPM\\_ORD\\_NV\\_ReadValueAuth](#) (208UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_UpdateVerification](#) (209UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_Manage](#) (210UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_CreateKeyDelegation](#) (212UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_CreateOwnerDelegation](#) (213UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_VerifyDelegation](#) (214UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_LoadOwnerDelegation](#) (216UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_ReadAuth](#) (217UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_Delegate\\_ReadTable](#) (219UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_CreateCounter](#) (220UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_IncrementCounter](#) (221UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_ReadCounter](#) (222UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_ReleaseCounter](#) (223UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_ReleaseCounterOwner](#) (224UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_EstablishTransport](#) (230UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_ExecuteTransport](#) (231UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_ReleaseTransportSigned](#) (232UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_GetTicks](#) (241UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_TickStampBlob](#) (242UL + TPM\_PROTECTED\_ORDINAL)
- #define [TPM\\_ORD\\_MAX](#) (256UL + TPM\_PROTECTED\_ORDINAL)
- #define [TSC\\_ORD\\_PhysicalPresence](#) (10UL + TPM\_CONNECTION\_ORDINAL)
- #define [TPM\\_BASE](#) 0x0
- #define [TPM\\_VENDOR\\_ERROR](#) 0x00000400
- #define [TPM\\_NON\\_FATAL](#) 0x00000800
- #define [TPM\\_SUCCESS](#) [TPM\\_BASE](#)
- #define [TPM\\_AUTHFAIL](#) [TPM\\_BASE](#) + 1
- #define [TPM\\_BADINDEX](#) [TPM\\_BASE](#) + 2
- #define [TPM\\_BAD\\_PARAMETER](#) [TPM\\_BASE](#) + 3
- #define [TPM\\_AUDITFAILURE](#) [TPM\\_BASE](#) + 4
- #define [TPM\\_CLEAR\\_DISABLED](#) [TPM\\_BASE](#) + 5
- #define [TPM\\_DEACTIVATED](#) [TPM\\_BASE](#) + 6
- #define [TPM\\_DISABLED](#) [TPM\\_BASE](#) + 7
- #define [TPM\\_DISABLED\\_CMD](#) [TPM\\_BASE](#) + 8
- #define [TPM\\_FAIL](#) [TPM\\_BASE](#) + 9
- #define [TPM\\_BAD\\_ORDINAL](#) [TPM\\_BASE](#) + 10
- #define [TPM\\_INSTALL\\_DISABLED](#) [TPM\\_BASE](#) + 11
- #define [TPM\\_INVALID\\_KEYHANDLE](#) [TPM\\_BASE](#) + 12
- #define [TPM\\_KEYNOTFOUND](#) [TPM\\_BASE](#) + 13
- #define [TPM\\_INAPPROPRIATE\\_ENC](#) [TPM\\_BASE](#) + 14
- #define [TPM\\_MIGRATEFAIL](#) [TPM\\_BASE](#) + 15
- #define [TPM\\_INVALID\\_PCR\\_INFO](#) [TPM\\_BASE](#) + 16
- #define [TPM\\_NOSPACE](#) [TPM\\_BASE](#) + 17
- #define [TPM\\_NOSRK](#) [TPM\\_BASE](#) + 18
- #define [TPM\\_NOTSEALED\\_BLOB](#) [TPM\\_BASE](#) + 19
- #define [TPM\\_OWNER\\_SET](#) [TPM\\_BASE](#) + 20
- #define [TPM\\_RESOURCES](#) [TPM\\_BASE](#) + 21
- #define [TPM\\_SHORTRANDOM](#) [TPM\\_BASE](#) + 22
- #define [TPM\\_SIZE](#) [TPM\\_BASE](#) + 23
- #define [TPM\\_WRONGPCRVAL](#) [TPM\\_BASE](#) + 24
- #define [TPM\\_BAD\\_PARAM\\_SIZE](#) [TPM\\_BASE](#) + 25
- #define [TPM\\_SHA\\_THREAD](#) [TPM\\_BASE](#) + 26
- #define [TPM\\_SHA\\_ERROR](#) [TPM\\_BASE](#) + 27
- #define [TPM\\_FAILEDSELFTEST](#) [TPM\\_BASE](#) + 28
- #define [TPM\\_AUTH2FAIL](#) [TPM\\_BASE](#) + 29
- #define [TPM\\_BADTAG](#) [TPM\\_BASE](#) + 30

- `#define TPM_IOERROR TPM_BASE + 31`
- `#define TPM_ENCRYPT_ERROR TPM_BASE + 32`
- `#define TPM_DECRYPT_ERROR TPM_BASE + 33`
- `#define TPM_INVALID_AUTHHANDLE TPM_BASE + 34`
- `#define TPM_NO_ENDORSEMENT TPM_BASE + 35`
- `#define TPM_INVALID_KEYUSAGE TPM_BASE + 36`
- `#define TPM_WRONG_ENTITYTYPE TPM_BASE + 37`
- `#define TPM_INVALID_POSTINIT TPM_BASE + 38`
- `#define TPM_INAPPROPRIATE_SIG TPM_BASE + 39`
- `#define TPM_BAD_KEY_PROPERTY TPM_BASE + 40`
- `#define TPM_BAD_MIGRATION TPM_BASE + 41`
- `#define TPM_BAD_SCHEME TPM_BASE + 42`
- `#define TPM_BAD_DATASIZE TPM_BASE + 43`
- `#define TPM_BAD_MODE TPM_BASE + 44`
- `#define TPM_BAD_PRESENCE TPM_BASE + 45`
- `#define TPM_BAD_VERSION TPM_BASE + 46`
- `#define TPM_NO_WRAP_TRANSPORT TPM_BASE + 47`
- `#define TPM_AUDITFAIL_UNSUCCESSFUL TPM_BASE + 48`
- `#define TPM_AUDITFAIL_SUCCESSFUL TPM_BASE + 49`
- `#define TPM_NOTRESETABLE TPM_BASE + 50`
- `#define TPM_NOTLOCAL TPM_BASE + 51`
- `#define TPM_BAD_TYPE TPM_BASE + 52`
- `#define TPM_INVALID_RESOURCE TPM_BASE + 53`
- `#define TPM_NOTFIPS TPM_BASE + 54`
- `#define TPM_INVALID_FAMILY TPM_BASE + 55`
- `#define TPM_NO_NV_PERMISSION TPM_BASE + 56`
- `#define TPM_REQUIRES_SIGN TPM_BASE + 57`
- `#define TPM_KEY_NOTSUPPORTED TPM_BASE + 58`
- `#define TPM_AUTH_CONFLICT TPM_BASE + 59`
- `#define TPM_AREA_LOCKED TPM_BASE + 60`
- `#define TPM_BAD_LOCALITY TPM_BASE + 61`
- `#define TPM_READ_ONLY TPM_BASE + 62`
- `#define TPM_PER_NOWRITE TPM_BASE + 63`
- `#define TPM_FAMILYCOUNT TPM_BASE + 64`
- `#define TPM_WRITE_LOCKED TPM_BASE + 65`
- `#define TPM_BAD_ATTRIBUTES TPM_BASE + 66`
- `#define TPM_INVALID_STRUCTURE TPM_BASE + 67`
- `#define TPM_KEY_OWNER_CONTROL TPM_BASE + 68`
- `#define TPM_BAD_COUNTER TPM_BASE + 69`
- `#define TPM_NOT_FULLWRITE TPM_BASE + 70`
- `#define TPM_CONTEXT_GAP TPM_BASE + 71`
- `#define TPM_MAXNVWRITES TPM_BASE + 72`
- `#define TPM_NOOPERATOR TPM_BASE + 73`
- `#define TPM_RESOURCEMISSING TPM_BASE + 74`
- `#define TPM_DELEGATE_LOCK TPM_BASE + 75`
- `#define TPM_DELEGATE_FAMILY TPM_BASE + 76`
- `#define TPM_DELEGATE_ADMIN TPM_BASE + 77`
- `#define TPM_TRANSPORT_EXCLUSIVE TPM_BASE + 78`
- `#define TPM_ST_CLEAR 0x0001`
- `#define TPM_ST_STATE 0x0002`
- `#define TPM_ST_DEACTIVATED 0x0003`
- `#define TPM_TAG_RQU_COMMAND 0x00c1`
- `#define TPM_TAG_RQU_AUTH1_COMMAND 0x00c2`
- `#define TPM_TAG_RQU_AUTH2_COMMAND 0x00c3`
- `#define TPM_TAG_RSP_COMMAND 0x00c4`

- #define `TPM_TAG_RSP_AUTH1_COMMAND` 0x00c5
- #define `TPM_TAG_RSP_AUTH2_COMMAND` 0x00c6
- #define `TPM_PT_ASYM` 0x01
- #define `TPM_PT_BIND` 0x02
- #define `TPM_PT_MIGRATE` 0x03
- #define `TPM_PT_MAINT` 0x04
- #define `TPM_PT_SEAL` 0x05
- #define `TPM_ET_KEYHANDLE` 0x0001
- #define `TPM_ET_OWNER` 0x0002
- #define `TPM_ET_DATA` 0x0003
- #define `TPM_ET_SRK` 0x0004
- #define `TPM_ET_KEY` 0x0005
- #define `TPM_RT_KEY` 0x00000001

*TPM\_ResourceTypes.*

- #define `TPM_RT_AUTH` 0x00000002
- #define `TPM_RT_HASH` 0x00000003
- #define `TPM_RT_TRANS` 0x00000004
- #define `TPM_RT_CONTEXT` 0x00000005
- #define `TPM_RT_COUNTER` 0x00000006
- #define `TPM_RT_DELEGATE` 0x00000007
- #define `TPM_RT_DAA_TPM` 0x00000008
- #define `TPM_RT_DAA_V0` 0x00000009
- #define `TPM_RT_DAA_V1` 0x0000000A
- #define `TPM_PID_OIAP` 0x0001
- #define `TPM_PID_OSAP` 0x0002
- #define `TPM_PID_ADIP` 0x0003
- #define `TPM_PID_ADCP` 0x0004
- #define `TPM_PID_OWNER` 0x0005
- #define `TPM_ALG_RSA` 0x00000001
- #define `TPM_ALG_DES` 0x00000002
- #define `TPM_ALG_3DES` 0x00000003
- #define `TPM_ALG_SHA` 0x00000004
- #define `TPM_ALG_HMAC` 0x00000005
- #define `TCPA_ALG_AES` 0x00000006
- #define `TPM_ES_NONE` 0x0001
- #define `TPM_ES_RSAESPKCSv15` 0x0002
- #define `TPM_ES_RSAESOAEP_SHA1_MGF1` 0x0003
- #define `TPM_SS_NONE` 0x0001
- #define `TPM_SS_RSASSAPKCS1v15_SHA1` 0x0002
- #define `TPM_SS_RSASSAPKCS1v15_DER` 0x0003
- #define `TPM_CAP_ORD` 0x00000001
- #define `TPM_CAP_ALG` 0x00000002
- #define `TPM_CAP_PID` 0x00000003
- #define `TPM_CAP_FLAG` 0x00000004
- #define `TPM_CAP_PROPERTY` 0x00000005
- #define `TPM_CAP_VERSION` 0x00000006
- #define `TPM_CAP_KEY_HANDLE` 0x00000007
- #define `TPM_CAP_CHECK_LOADED` 0x00000008
- #define `TPM_CAP_SYM_MODE` 0x00000009
- #define `TPM_CAP_KEY_STATUS` 0x0000000C
- #define `TPM_CAP_NV_LIST` 0x0000000D
- #define `TPM_CAP_MFR` 0x00000010
- #define `TPM_CAP_NV_INDEX` 0x00000011
- #define `TPM_CAP_TRANS_ALG` 0x00000012

- #define [TPM\\_CAP\\_HANDLE](#) 0x00000014
- #define [TPM\\_CAP\\_TRANS\\_ES](#) 0x00000015
- #define [TPM\\_CAP\\_AUTH\\_ENCRYPT](#) 0x00000017
- #define [TPM\\_CAP\\_SELECT\\_SIZE](#) 0x00000018
- #define [TPM\\_CAP\\_DA\\_LOGIC](#) 0x00000019
- #define [TPM\\_CAP\\_VERSION\\_VAL](#) 0x0000001A
- #define [TPM\\_CAP\\_PROP\\_PCR](#) 0x00000101
- #define [TPM\\_CAP\\_PROP\\_DIR](#) 0x00000102
- #define [TPM\\_CAP\\_PROP\\_MANUFACTURER](#) 0x00000103
- #define [TPM\\_CAP\\_PROP\\_KEYS](#) 0x00000104
- #define [TPM\\_CAP\\_PROP\\_MIN\\_COUNTER](#) 0x00000107
- #define [TPM\\_CAP\\_FLAG\\_PERMANENT](#) 0x00000108
- #define [TPM\\_CAP\\_FLAG\\_VOLATILE](#) 0x00000109
- #define [TPM\\_CAP\\_PROP\\_AUTHSESS](#) 0x0000010A
- #define [TPM\\_CAP\\_PROP\\_TRANSESS](#) 0x0000010B
- #define [TPM\\_CAP\\_PROP\\_COUNTERS](#) 0x0000010C
- #define [TPM\\_CAP\\_PROP\\_MAX\\_AUTHSESS](#) 0x0000010D
- #define [TPM\\_CAP\\_PROP\\_MAX\\_TRANSESS](#) 0x0000010E
- #define [TPM\\_CAP\\_PROP\\_MAX\\_COUNTERS](#) 0x0000010F
- #define [TPM\\_CAP\\_PROP\\_MAX\\_KEYS](#) 0x00000110
- #define [TPM\\_CAP\\_PROP\\_OWNER](#) 0x00000111
- #define [TPM\\_CAP\\_PROP\\_CONTEXT](#) 0x00000112
- #define [TPM\\_CAP\\_PROP\\_MAX\\_CONTEXT](#) 0x00000113
- #define [TPM\\_CAP\\_PROP\\_FAMILYROWS](#) 0x00000114
- #define [TPM\\_CAP\\_PROP\\_TIS\\_TIMEOUT](#) 0x00000115
- #define [TPM\\_CAP\\_PROP\\_STARTUP\\_EFFECT](#) 0x00000116
- #define [TPM\\_CAP\\_PROP\\_DELEGATE\\_ROW](#) 0x00000117
- #define [TPM\\_CAP\\_PROP\\_MAX\\_DAASESS](#) 0x00000119
- #define [TPM\\_CAP\\_PROP\\_DAASESS](#) 0x0000011A
- #define [TPM\\_CAP\\_PROP\\_CONTEXT\\_DIST](#) 0x0000011B
- #define [TPM\\_CAP\\_PROP\\_DAA\\_INTERRUPT](#) 0x0000011C
- #define [TPM\\_CAP\\_PROP\\_SESSIONS](#) 0x0000011D
- #define [TPM\\_CAP\\_PROP\\_MAX\\_SESSIONS](#) 0x0000011E
- #define [TPM\\_CAP\\_PROP\\_CMK\\_RESTRICTION](#) 0x0000011F
- #define [TPM\\_CAP\\_PROP\\_DURATION](#) 0x00000120
- #define [TPM\\_CAP\\_PROP\\_ACTIVE\\_COUNTER](#) 0x00000122
- #define [TPM\\_CAP\\_PROP\\_MAX\\_NV\\_AVAILABLE](#) 0x00000123
- #define [TPM\\_CAP\\_PROP\\_INPUT\\_BUFFER](#) 0x00000124
- #define [TPM\\_KEY\\_EK](#) 0x0000
- #define [TPM\\_KEY\\_SIGNING](#) 0x0010
- #define [TPM\\_KEY\\_STORAGE](#) 0x0011
- #define [TPM\\_KEY\\_IDENTITY](#) 0x0012
- #define [TPM\\_KEY\\_AUTHCHANGE](#) 0x0013
- #define [TPM\\_KEY\\_BIND](#) 0x0014
- #define [TPM\\_KEY\\_LEGACY](#) 0x0015
- #define [TPM\\_AUTH\\_NEVER](#) 0x00
- #define [TPM\\_AUTH\\_ALWAYS](#) 0x01
- #define [TPM\\_OWNER\\_KEYHANDLE](#) 0x40000001
- #define [TPM\\_SRK\\_KEYHANDLE](#) 0x40000000
- #define [ERRORDIE](#)(s)
- #define [TPMTRY](#)(s, c)
- #define [TPMTRYRETURN](#)(c)

## Typedefs

- typedef unsigned char [BYTE](#)
- typedef unsigned char [BOOL](#)
- typedef unsigned short [UINT16](#)
- typedef unsigned int [UINT32](#)
- typedef unsigned long long [UINT64](#)
- typedef [UINT32](#) [TPM\\_RESULT](#)
- typedef [UINT32](#) [TPM\\_PCRINDEX](#)
- typedef [UINT32](#) [TPM\\_DIRINDEX](#)
- typedef [UINT32](#) [TPM\\_HANDLE](#)
- typedef [TPM\\_HANDLE](#) [TPM\\_AUTHHANDLE](#)
- typedef [TPM\\_HANDLE](#) [TCPA\\_HASHHANDLE](#)
- typedef [TPM\\_HANDLE](#) [TCPA\\_HMACHANDLE](#)
- typedef [TPM\\_HANDLE](#) [TCPA\\_ENCHANDLE](#)
- typedef [TPM\\_HANDLE](#) [TPM\\_KEY\\_HANDLE](#)
- typedef [TPM\\_HANDLE](#) [TCPA\\_ENTITYHANDLE](#)
- typedef [UINT32](#) [TPM\\_RESOURCE\\_TYPE](#)
- typedef [UINT32](#) [TPM\\_COMMAND\\_CODE](#)
- typedef [UINT16](#) [TPM\\_PROTOCOL\\_ID](#)
- typedef [BYTE](#) [TPM\\_AUTH\\_DATA\\_USAGE](#)
- typedef [UINT16](#) [TPM\\_ENTITY\\_TYPE](#)
- typedef [UINT32](#) [TPM\\_ALGORITHM\\_ID](#)
- typedef [UINT16](#) [TPM\\_KEY\\_USAGE](#)
- typedef [UINT16](#) [TPM\\_STARTUP\\_TYPE](#)
- typedef [UINT32](#) [TPM\\_CAPABILITY\\_AREA](#)
- typedef [UINT16](#) [TPM\\_ENC\\_SCHEME](#)
- typedef [UINT16](#) [TPM\\_SIG\\_SCHEME](#)
- typedef [UINT16](#) [TPM\\_MIGRATE\\_SCHEME](#)
- typedef [UINT16](#) [TPM\\_PHYSICAL\\_PRESENCE](#)
- typedef [UINT32](#) [TPM\\_KEY\\_FLAGS](#)
- typedef [BYTE](#) [TPM\\_AUTHDATA](#) [[TPM\\_DIGEST\\_SIZE](#)]
- typedef [TPM\\_AUTHDATA](#) [TPM\\_SECRET](#)
- typedef [TPM\\_AUTHDATA](#) [TPM\\_ENCAUTH](#)
- typedef [BYTE](#) [TPM\\_PAYLOAD\\_TYPE](#)
- typedef [UINT16](#) [TPM\\_TAG](#)
- typedef [UINT32](#) [TCS\\_AUTHHANDLE](#)
- typedef [UINT32](#) [TCS\\_CONTEXT\\_HANDLE](#)
- typedef [UINT32](#) [TCS\\_KEY\\_HANDLE](#)
- typedef struct [TPM\\_VERSION](#) [TPM\\_VERSION](#)
- typedef struct [TPM\\_DIGEST](#) [TPM\\_DIGEST](#)
- typedef [TPM\\_DIGEST](#) [TPM\\_PCRVALUE](#)
- typedef [TPM\\_DIGEST](#) [TPM\\_COMPOSITE\\_HASH](#)
- typedef [TPM\\_DIGEST](#) [TPM\\_DIRVALUE](#)
- typedef [TPM\\_DIGEST](#) [TPM\\_HMAC](#)
- typedef [TPM\\_DIGEST](#) [TPM\\_CHOSENID\\_HASH](#)
- typedef struct [TPM\\_NONCE](#) [TPM\\_NONCE](#)
- typedef struct [TPM\\_KEY\\_PARMS](#) [TPM\\_KEY\\_PARMS](#)
- typedef struct [TPM\\_RSA\\_KEY\\_PARMS](#) [TPM\\_RSA\\_KEY\\_PARMS](#)
- typedef struct [TPM\\_STORE\\_PUBKEY](#) [TPM\\_STORE\\_PUBKEY](#)
- typedef struct [TPM\\_PUBKEY](#) [TPM\\_PUBKEY](#)
- typedef struct [TPM\\_KEY](#) [TPM\\_KEY](#)
- typedef struct [TPM\\_PCR\\_SELECTION](#) [TPM\\_PCR\\_SELECTION](#)
- typedef struct [TPM\\_PCR\\_COMPOSITE](#) [TPM\\_PCR\\_COMPOSITE](#)
- typedef struct [TPM\\_PCR\\_INFO](#) [TPM\\_PCR\\_INFO](#)

- typedef struct [TPM\\_BOUND\\_DATA](#) [TPM\\_BOUND\\_DATA](#)
- typedef struct [TPM\\_STORED\\_DATA](#) [TPM\\_STORED\\_DATA](#)
- typedef struct [TCS\\_AUTH](#) [TCS\\_AUTH](#)
- typedef struct [pack\\_buf\\_t](#) [pack\\_buf\\_t](#)
- typedef struct [pack\\_constbuf\\_t](#) [pack\\_constbuf\\_t](#)

## 4.169.1 Macro Definition Documentation

### 4.169.1.1 #define ERRORDIE( s )

**Value:**

```
do { status = s; \
    fprintf (stderr, "*** ERRORDIE in %s at %s: %i\n", __func__, __FILE__, __LINE__);
    \
    goto abort_egress; } \
while (0)
```

### 4.169.1.2 #define FALSE 0x00

### 4.169.1.3 #define NULL\_PACK\_BUF {0,0}

### 4.169.1.4 #define TCPA\_ALG\_AES 0x00000006

### 4.169.1.5 #define TCPA\_MAX\_BUFFER\_LENGTH 0x2000

### 4.169.1.6 #define TPM\_ALG\_3DES 0X00000003

### 4.169.1.7 #define TPM\_ALG\_DES 0x00000002

### 4.169.1.8 #define TPM\_ALG\_HMAC 0x00000005

### 4.169.1.9 #define TPM\_ALG\_RSA 0x00000001

### 4.169.1.10 #define TPM\_ALG\_SHA 0x00000004

### 4.169.1.11 #define TPM\_AREA\_LOCKED TPM\_BASE + 60

### 4.169.1.12 #define TPM\_AUDITFAIL\_SUCCESSFUL TPM\_BASE + 49

### 4.169.1.13 #define TPM\_AUDITFAIL\_UNSUCCESSFUL TPM\_BASE + 48

### 4.169.1.14 #define TPM\_AUDITFAILURE TPM\_BASE + 4

### 4.169.1.15 #define TPM\_AUTH2FAIL TPM\_BASE + 29

### 4.169.1.16 #define TPM\_AUTH\_ALWAYS 0x01

### 4.169.1.17 #define TPM\_AUTH\_CONFLICT TPM\_BASE + 59

### 4.169.1.18 #define TPM\_AUTH\_NEVER 0x00

### 4.169.1.19 #define TPM\_AUTHFAIL TPM\_BASE + 1

### 4.169.1.20 #define TPM\_BAD\_ATTRIBUTES TPM\_BASE + 66

4.169.1.21 `#define TPM_BAD_COUNTER TPM_BASE + 69`

4.169.1.22 `#define TPM_BAD_DATASIZE TPM_BASE + 43`

4.169.1.23 `#define TPM_BAD_KEY_PROPERTY TPM_BASE + 40`

4.169.1.24 `#define TPM_BAD_LOCALITY TPM_BASE + 61`

4.169.1.25 `#define TPM_BAD_MIGRATION TPM_BASE + 41`

4.169.1.26 `#define TPM_BAD_MODE TPM_BASE + 44`

4.169.1.27 `#define TPM_BAD_ORDINAL TPM_BASE + 10`

4.169.1.28 `#define TPM_BAD_PARAM_SIZE TPM_BASE + 25`

4.169.1.29 `#define TPM_BAD_PARAMETER TPM_BASE + 3`

4.169.1.30 `#define TPM_BAD_PRESENCE TPM_BASE + 45`

4.169.1.31 `#define TPM_BAD_SCHEME TPM_BASE + 42`

4.169.1.32 `#define TPM_BAD_TYPE TPM_BASE + 52`

4.169.1.33 `#define TPM_BAD_VERSION TPM_BASE + 46`

4.169.1.34 `#define TPM_BADINDEX TPM_BASE + 2`

4.169.1.35 `#define TPM_BADTAG TPM_BASE + 30`

4.169.1.36 `#define TPM_BASE 0x0`

4.169.1.37 `#define TPM_CAP_ALG 0x00000002`

4.169.1.38 `#define TPM_CAP_AUTH_ENCRYPT 0x00000017`

4.169.1.39 `#define TPM_CAP_CHECK_LOADED 0x00000008`

4.169.1.40 `#define TPM_CAP_DA_LOGIC 0x00000019`

4.169.1.41 `#define TPM_CAP_FLAG 0x00000004`

4.169.1.42 `#define TPM_CAP_FLAG_PERMANENT 0x00000108`

4.169.1.43 `#define TPM_CAP_FLAG_VOLATILE 0x00000109`

4.169.1.44 `#define TPM_CAP_HANDLE 0x00000014`

4.169.1.45 `#define TPM_CAP_KEY_HANDLE 0x00000007`

4.169.1.46 `#define TPM_CAP_KEY_STATUS 0x0000000C`

4.169.1.47 `#define TPM_CAP_MFR 0x00000010`

4.169.1.48 `#define TPM_CAP_NV_INDEX 0x00000011`



4.169.1.49 `#define TPM_CAP_NV_LIST 0x0000000D`

4.169.1.50 `#define TPM_CAP_ORD 0x00000001`

4.169.1.51 `#define TPM_CAP_PID 0x00000003`

4.169.1.52 `#define TPM_CAP_PROP_ACTIVE_COUNTER 0x00000122`

4.169.1.53 `#define TPM_CAP_PROP_AUTHSESS 0x0000010A`

4.169.1.54 `#define TPM_CAP_PROP_CMK_RESTRICTION 0x0000011F`

4.169.1.55 `#define TPM_CAP_PROP_CONTEXT 0x00000112`

4.169.1.56 `#define TPM_CAP_PROP_CONTEXT_DIST 0x0000011B`

4.169.1.57 `#define TPM_CAP_PROP_COUNTERS 0x0000010C`

4.169.1.58 `#define TPM_CAP_PROP_DAA_INTERRUPT 0x0000011C`

4.169.1.59 `#define TPM_CAP_PROP_DAASESS 0x0000011A`

4.169.1.60 `#define TPM_CAP_PROP_DELEGATE_ROW 0x00000117`

4.169.1.61 `#define TPM_CAP_PROP_DIR 0x00000102`

4.169.1.62 `#define TPM_CAP_PROP_DURATION 0x00000120`

4.169.1.63 `#define TPM_CAP_PROP_FAMILYROWS 0x00000114`

4.169.1.64 `#define TPM_CAP_PROP_INPUT_BUFFER 0x00000124`

4.169.1.65 `#define TPM_CAP_PROP_KEYS 0x00000104`

4.169.1.66 `#define TPM_CAP_PROP_MANUFACTURER 0x00000103`

4.169.1.67 `#define TPM_CAP_PROP_MAX_AUTHSESS 0x0000010D`

4.169.1.68 `#define TPM_CAP_PROP_MAX_CONTEXT 0x00000113`

4.169.1.69 `#define TPM_CAP_PROP_MAX_COUNTERS 0x0000010F`

4.169.1.70 `#define TPM_CAP_PROP_MAX_DAASESS 0x00000119`

4.169.1.71 `#define TPM_CAP_PROP_MAX_KEYS 0x00000110`

4.169.1.72 `#define TPM_CAP_PROP_MAX_NV_AVAILABLE 0x00000123`

4.169.1.73 `#define TPM_CAP_PROP_MAX_SESSIONS 0x0000011E`

4.169.1.74 `#define TPM_CAP_PROP_MAX_TRANSESS 0x0000010E`

4.169.1.75 `#define TPM_CAP_PROP_MIN_COUNTER 0x00000107`

4.169.1.76 `#define TPM_CAP_PROP_OWNER 0x00000111`

4.169.1.77 #define TPM\_CAP\_PROP\_PCR 0x00000101

4.169.1.78 #define TPM\_CAP\_PROP\_SESSIONS 0x0000011D

4.169.1.79 #define TPM\_CAP\_PROP\_STARTUP\_EFFECT 0x00000116

4.169.1.80 #define TPM\_CAP\_PROP\_TIS\_TIMEOUT 0x00000115

4.169.1.81 #define TPM\_CAP\_PROP\_TRANSESS 0x0000010B

4.169.1.82 #define TPM\_CAP\_PROPERTY 0x00000005

4.169.1.83 #define TPM\_CAP\_SELECT\_SIZE 0x00000018

4.169.1.84 #define TPM\_CAP\_SYM\_MODE 0x00000009

4.169.1.85 #define TPM\_CAP\_TRANS\_ALG 0x00000012

4.169.1.86 #define TPM\_CAP\_TRANS\_ES 0x00000015

4.169.1.87 #define TPM\_CAP\_VERSION 0x00000006

4.169.1.88 #define TPM\_CAP\_VERSION\_VAL 0x0000001A

4.169.1.89 #define TPM\_CLEAR\_DISABLED TPM\_BASE + 5

4.169.1.90 #define TPM\_CONNECTION\_ORDINAL 0x40000000UL

4.169.1.91 #define TPM\_CONTEXT\_GAP TPM\_BASE + 71

4.169.1.92 #define TPM\_DEACTIVATED TPM\_BASE + 6

4.169.1.93 #define TPM\_DECRYPT\_ERROR TPM\_BASE + 33

4.169.1.94 #define TPM\_DELEGATE\_ADMIN TPM\_BASE + 77

4.169.1.95 #define TPM\_DELEGATE\_FAMILY TPM\_BASE + 76

4.169.1.96 #define TPM\_DELEGATE\_LOCK TPM\_BASE + 75

4.169.1.97 #define TPM\_DIGEST\_SIZE 20

4.169.1.98 #define TPM\_DISABLED TPM\_BASE + 7

4.169.1.99 #define TPM\_DISABLED\_CMD TPM\_BASE + 8

4.169.1.100 #define TPM\_ENCRYPT\_ERROR TPM\_BASE + 32

4.169.1.101 #define TPM\_ES\_NONE 0x0001

4.169.1.102 #define TPM\_ES\_RSAESOAEP\_SHA1\_MGF1 0x0003

4.169.1.103 #define TPM\_ES\_RSAESPKCSv15 0x0002

4.169.1.104 #define TPM\_ET\_DATA 0x0003

4.169.1.105 `#define TPM_ET_KEY 0x0005`

4.169.1.106 `#define TPM_ET_KEYHANDLE 0x0001`

4.169.1.107 `#define TPM_ET_OWNER 0x0002`

4.169.1.108 `#define TPM_ET_SRK 0x0004`

4.169.1.109 `#define TPM_FAIL TPM_BASE + 9`

4.169.1.110 `#define TPM_FAILEDSELFTEST TPM_BASE + 28`

4.169.1.111 `#define TPM_FAMILYCOUNT TPM_BASE + 64`

4.169.1.112 `#define TPM_INAPPROPRIATE_ENC TPM_BASE + 14`

4.169.1.113 `#define TPM_INAPPROPRIATE_SIG TPM_BASE + 39`

4.169.1.114 `#define TPM_INSTALL_DISABLED TPM_BASE + 11`

4.169.1.115 `#define TPM_INVALID_AUTHHANDLE TPM_BASE + 34`

4.169.1.116 `#define TPM_INVALID_FAMILY TPM_BASE + 55`

4.169.1.117 `#define TPM_INVALID_KEYHANDLE TPM_BASE + 12`

4.169.1.118 `#define TPM_INVALID_KEYUSAGE TPM_BASE + 36`

4.169.1.119 `#define TPM_INVALID_PCR_INFO TPM_BASE + 16`

4.169.1.120 `#define TPM_INVALID_POSTINIT TPM_BASE + 38`

4.169.1.121 `#define TPM_INVALID_RESOURCE TPM_BASE + 53`

4.169.1.122 `#define TPM_INVALID_STRUCTURE TPM_BASE + 67`

4.169.1.123 `#define TPM_IOERROR TPM_BASE + 31`

4.169.1.124 `#define TPM_KEY_AUTHCHANGE 0x0013`

4.169.1.125 `#define TPM_KEY_BIND 0x0014`

4.169.1.126 `#define TPM_KEY_EK 0x0000`

4.169.1.127 `#define TPM_KEY_IDENTITY 0x0012`

4.169.1.128 `#define TPM_KEY_LEGACY 0x0015`

4.169.1.129 `#define TPM_KEY_NOTSUPPORTED TPM_BASE + 58`

4.169.1.130 `#define TPM_KEY_OWNER_CONTROL TPM_BASE + 68`

4.169.1.131 `#define TPM_KEY_SIGNING 0x0010`

4.169.1.132 `#define TPM_KEY_STORAGE 0x0011`

4.169.1.133 #define TPM\_KEYNOTFOUND TPM\_BASE + 13

4.169.1.134 #define TPM\_MAXNVWRITES TPM\_BASE + 72

4.169.1.135 #define TPM\_MIGRATEFAIL TPM\_BASE + 15

4.169.1.136 #define TPM\_NO\_ENDORSEMENT TPM\_BASE + 35

4.169.1.137 #define TPM\_NO\_NV\_PERMISSION TPM\_BASE + 56

4.169.1.138 #define TPM\_NO\_WRAP\_TRANSPORT TPM\_BASE + 47

4.169.1.139 #define TPM\_NON\_FATAL 0x00000800

4.169.1.140 #define TPM\_NOOPERATOR TPM\_BASE + 73

4.169.1.141 #define TPM\_NOSPACE TPM\_BASE + 17

4.169.1.142 #define TPM\_NOSRK TPM\_BASE + 18

4.169.1.143 #define TPM\_NOT\_FULLWRITE TPM\_BASE + 70

4.169.1.144 #define TPM\_NOTFIPS TPM\_BASE + 54

4.169.1.145 #define TPM\_NOTLOCAL TPM\_BASE + 51

4.169.1.146 #define TPM\_NOTRESETABLE TPM\_BASE + 50

4.169.1.147 #define TPM\_NOTSEALED\_BLOB TPM\_BASE + 19

4.169.1.148 #define TPM\_ORD\_ActivateIdentity (122UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.149 #define TPM\_ORD\_AuthorizeMigrationKey (43UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.150 #define TPM\_ORD\_CertifyKey (50UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.151 #define TPM\_ORD\_CertifySelfTest (82UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.152 #define TPM\_ORD\_ChangeAuth (12UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.153 #define TPM\_ORD\_ChangeAuthAsymFinish (15UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.154 #define TPM\_ORD\_ChangeAuthAsymStart (14UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.155 #define TPM\_ORD\_ChangeAuthOwner (16UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.156 #define TPM\_ORD\_ContinueSelfTest (83UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.157 #define TPM\_ORD\_ConvertMigrationBlob (42UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.158 #define TPM\_ORD\_CreateCounter (220UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.159 #define TPM\_ORD\_CreateEndorsementKeyPair (120UL + TPM\_PROTECTED\_ORDINAL)

4.169.1.160 #define TPM\_ORD\_CreateMaintenanceArchive (44UL + TPM\_PROTECTED\_ORDINAL)

- 4.169.1.161 `#define TPM_ORD_CreateMigrationBlob (40UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.162 `#define TPM_ORD_CreateWrapKey (31UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.163 `#define TPM_ORD_Delegate_CreateKeyDelegation (212UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.164 `#define TPM_ORD_Delegate_CreateOwnerDelegation (213UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.165 `#define TPM_ORD_Delegate_LoadOwnerDelegation (216UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.166 `#define TPM_ORD_Delegate_Manage (210UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.167 `#define TPM_ORD_Delegate_ReadAuth (217UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.168 `#define TPM_ORD_Delegate_ReadTable (219UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.169 `#define TPM_ORD_Delegate_UpdateVerification (209UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.170 `#define TPM_ORD_Delegate_VerifyDelegation (214UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.171 `#define TPM_ORD_DirRead (26UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.172 `#define TPM_ORD_DirWriteAuth (25UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.173 `#define TPM_ORD_DisableForceClear (94UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.174 `#define TPM_ORD_DisableOwnerClear (92UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.175 `#define TPM_ORD_DisablePubekRead (126UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.176 `#define TPM_ORD_EstablishTransport (230UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.177 `#define TPM_ORD_EvictKey (34UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.178 `#define TPM_ORD_ExecuteTransport (231UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.179 `#define TPM_ORD_Extend (20UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.180 `#define TPM_ORD_FieldUpgrade (170UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.181 `#define TPM_ORD_FlushSpecific (186UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.182 `#define TPM_ORD_ForceClear (93UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.183 `#define TPM_ORD_GetAuditEvent (130UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.184 `#define TPM_ORD_GetAuditEventSigned (131UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.185 `#define TPM_ORD_GetCapability (101UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.186 `#define TPM_ORD_GetCapabilityOwner (102UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.187 `#define TPM_ORD_GetCapabilitySigned (100UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.188 `#define TPM_ORD_GetOrdinalAuditStatus (140UL + TPM_PROTECTED_ORDINAL)`

4.169.1.189 `#define TPM_ORD_GetPubKey (33UL + TPM_PROTECTED_ORDINAL)`

4.169.1.190 `#define TPM_ORD_GetRandom (70UL + TPM_PROTECTED_ORDINAL)`

4.169.1.191 `#define TPM_ORD_GetTestResult (84UL + TPM_PROTECTED_ORDINAL)`

4.169.1.192 `#define TPM_ORD_GetTicks (241UL + TPM_PROTECTED_ORDINAL)`

4.169.1.193 `#define TPM_ORD_IncrementCounter (221UL + TPM_PROTECTED_ORDINAL)`

4.169.1.194 `#define TPM_ORD_Init (151UL + TPM_PROTECTED_ORDINAL)`

4.169.1.195 `#define TPM_ORD_KillMaintenanceFeature (46UL + TPM_PROTECTED_ORDINAL)`

4.169.1.196 `#define TPM_ORD_LoadAuthContext (183UL + TPM_PROTECTED_ORDINAL)`

4.169.1.197 `#define TPM_ORD_LoadContext (185UL + TPM_PROTECTED_ORDINAL)`

4.169.1.198 `#define TPM_ORD_LoadKey (32UL + TPM_PROTECTED_ORDINAL)`

4.169.1.199 `#define TPM_ORD_LoadKeyContext (181UL + TPM_PROTECTED_ORDINAL)`

4.169.1.200 `#define TPM_ORD_LoadMaintenanceArchive (45UL + TPM_PROTECTED_ORDINAL)`

4.169.1.201 `#define TPM_ORD_LoadManuMaintPub (47UL + TPM_PROTECTED_ORDINAL)`

4.169.1.202 `#define TPM_ORD_MakeIdentity (121UL + TPM_PROTECTED_ORDINAL)`

4.169.1.203 `#define TPM_ORD_MAX (256UL + TPM_PROTECTED_ORDINAL)`

4.169.1.204 `#define TPM_ORD_NV_DefineSpace (204UL + TPM_PROTECTED_ORDINAL)`

4.169.1.205 `#define TPM_ORD_NV_ReadValue (207UL + TPM_PROTECTED_ORDINAL)`

4.169.1.206 `#define TPM_ORD_NV_ReadValueAuth (208UL + TPM_PROTECTED_ORDINAL)`

4.169.1.207 `#define TPM_ORD_NV_WriteValue (205UL + TPM_PROTECTED_ORDINAL)`

4.169.1.208 `#define TPM_ORD_NV_WriteValueAuth (206UL + TPM_PROTECTED_ORDINAL)`

4.169.1.209 `#define TPM_ORD_OIAP (10UL + TPM_PROTECTED_ORDINAL)`

4.169.1.210 `#define TPM_ORD_OSAP (11UL + TPM_PROTECTED_ORDINAL)`

4.169.1.211 `#define TPM_ORD_OwnerClear (91UL + TPM_PROTECTED_ORDINAL)`

4.169.1.212 `#define TPM_ORD_OwnerReadPubek (125UL + TPM_PROTECTED_ORDINAL)`

4.169.1.213 `#define TPM_ORD_OwnerSetDisable (110UL + TPM_PROTECTED_ORDINAL)`

4.169.1.214 `#define TPM_ORD_PCR_Reset (200UL + TPM_PROTECTED_ORDINAL)`

4.169.1.215 `#define TPM_ORD_PcrRead (21UL + TPM_PROTECTED_ORDINAL)`

4.169.1.216 `#define TPM_ORD_PhysicalDisable (112UL + TPM_PROTECTED_ORDINAL)`

- 4.169.1.217 `#define TPM_ORD_PhysicalEnable (111UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.218 `#define TPM_ORD_PhysicalSetDeactivated (114UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.219 `#define TPM_ORD_Quote (22UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.220 `#define TPM_ORD_ReadCounter (222UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.221 `#define TPM_ORD_ReadManuMaintPub (48UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.222 `#define TPM_ORD_ReadPubek (124UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.223 `#define TPM_ORD_ReleaseCounter (223UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.224 `#define TPM_ORD_ReleaseCounterOwner (224UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.225 `#define TPM_ORD_ReleaseTransportSigned (232UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.226 `#define TPM_ORD_Reset (90UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.227 `#define TPM_ORD_ReWrapKey (41UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.228 `#define TPM_ORD_SaveAuthContext (182UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.229 `#define TPM_ORD_SaveContext (184UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.230 `#define TPM_ORD_SaveKeyContext (180UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.231 `#define TPM_ORD_SaveState (152UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.232 `#define TPM_ORD_Seal (23UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.233 `#define TPM_ORD_SelfTestFull (80UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.234 `#define TPM_ORD_SelfTestStartup (81UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.235 `#define TPM_ORD_SetOrdinalAuditStatus (141UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.236 `#define TPM_ORD_SetOwnerInstall (113UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.237 `#define TPM_ORD_SetRedirection (154UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.238 `#define TPM_ORD_SetTempDeactivated (115UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.239 `#define TPM_ORD_SHA1Complete (162UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.240 `#define TPM_ORD_SHA1CompleteExtend (163UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.241 `#define TPM_ORD_SHA1Start (160UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.242 `#define TPM_ORD_SHA1Update (161UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.243 `#define TPM_ORD_Sign (60UL + TPM_PROTECTED_ORDINAL)`
- 4.169.1.244 `#define TPM_ORD_Startup (153UL + TPM_PROTECTED_ORDINAL)`

4.169.1.245 `#define TPM_ORD_StirRandom (71UL + TPM_PROTECTED_ORDINAL)`

4.169.1.246 `#define TPM_ORD_TakeOwnership (13UL + TPM_PROTECTED_ORDINAL)`

4.169.1.247 `#define TPM_ORD_Terminate_Handle (150UL + TPM_PROTECTED_ORDINAL)`

4.169.1.248 `#define TPM_ORD_TickStampBlob (242UL + TPM_PROTECTED_ORDINAL)`

4.169.1.249 `#define TPM_ORD_UnBind (30UL + TPM_PROTECTED_ORDINAL)`

4.169.1.250 `#define TPM_ORD_Unseal (24UL + TPM_PROTECTED_ORDINAL)`

4.169.1.251 `#define TPM_OWNER_KEYHANDLE 0x40000001`

4.169.1.252 `#define TPM_OWNER_SET TPM_BASE + 20`

4.169.1.253 `#define TPM_PER_NOWRITE TPM_BASE + 63`

4.169.1.254 `#define TPM_PID_ADCP 0x0004`

4.169.1.255 `#define TPM_PID_ADIP 0x0003`

4.169.1.256 `#define TPM_PID_OIAP 0x0001`

4.169.1.257 `#define TPM_PID_OSAP 0x0002`

4.169.1.258 `#define TPM_PID_OWNER 0x0005`

4.169.1.259 `#define TPM_PROTECTED_ORDINAL 0x00000000UL`

4.169.1.260 `#define TPM_PT_ASYM 0x01`

4.169.1.261 `#define TPM_PT_BIND 0x02`

4.169.1.262 `#define TPM_PT_MAINT 0x04`

4.169.1.263 `#define TPM_PT_MIGRATE 0x03`

4.169.1.264 `#define TPM_PT_SEAL 0x05`

4.169.1.265 `#define TPM_READ_ONLY TPM_BASE + 62`

4.169.1.266 `#define TPM_REQUIRES_SIGN TPM_BASE + 57`

4.169.1.267 `#define TPM_RESOURCEMISSING TPM_BASE + 74`

4.169.1.268 `#define TPM_RESOURCES TPM_BASE + 21`

4.169.1.269 `#define TPM_RT_AUTH 0x00000002`

4.169.1.270 `#define TPM_RT_CONTEXT 0x00000005`

4.169.1.271 `#define TPM_RT_COUNTER 0x00000006`

4.169.1.272 `#define TPM_RT_DAA_TPM 0x00000008`



4.169.1.273 #define TPM\_RT\_DAA\_V0 0x00000009

4.169.1.274 #define TPM\_RT\_DAA\_V1 0x0000000A

4.169.1.275 #define TPM\_RT\_DELEGATE 0x00000007

4.169.1.276 #define TPM\_RT\_HASH 0x00000003

4.169.1.277 #define TPM\_RT\_KEY 0x00000001

TPM\_ResourceTypes.

4.169.1.278 #define TPM\_RT\_TRANS 0x00000004

4.169.1.279 #define TPM\_SHA\_ERROR TPM\_BASE + 27

4.169.1.280 #define TPM\_SHA\_THREAD TPM\_BASE + 26

4.169.1.281 #define TPM\_SHORTRANDOM TPM\_BASE + 22

4.169.1.282 #define TPM\_SIZE TPM\_BASE + 23

4.169.1.283 #define TPM\_SRK\_KEYHANDLE 0x40000000

4.169.1.284 #define TPM\_SS\_NONE 0x0001

4.169.1.285 #define TPM\_SS\_RSASSAPKCS1v15\_DER 0x0003

4.169.1.286 #define TPM\_SS\_RSASSAPKCS1v15\_SHA1 0x0002

4.169.1.287 #define TPM\_ST\_CLEAR 0x0001

4.169.1.288 #define TPM\_ST\_DEACTIVATED 0x003

4.169.1.289 #define TPM\_ST\_STATE 0x0002

4.169.1.290 #define TPM\_SUCCESS TPM\_BASE

4.169.1.291 #define TPM\_TAG\_RQU\_AUTH1\_COMMAND 0x00c2

4.169.1.292 #define TPM\_TAG\_RQU\_AUTH2\_COMMAND 0x00c3

4.169.1.293 #define TPM\_TAG\_RQU\_COMMAND 0x00c1

4.169.1.294 #define TPM\_TAG\_RSP\_AUTH1\_COMMAND 0x00c5

4.169.1.295 #define TPM\_TAG\_RSP\_AUTH2\_COMMAND 0x00c6

4.169.1.296 #define TPM\_TAG\_RSP\_COMMAND 0x00c4

4.169.1.297 #define TPM\_TRANSPORT\_EXCLUSIVE TPM\_BASE + 78

4.169.1.298 #define TPM\_UNPROTECTED\_ORDINAL 0x80000000UL

4.169.1.299 #define TPM\_VENDOR\_ERROR 0x00000400

4.169.1.300 **#define** TPM\_VENDOR\_ORDINAL 0x20000000UL

4.169.1.301 **#define** TPM\_WRITE\_LOCKED TPM\_BASE + 65

4.169.1.302 **#define** TPM\_WRONG\_ENTITYTYPE TPM\_BASE + 37

4.169.1.303 **#define** TPM\_WRONGPCRVAL TPM\_BASE + 24

4.169.1.304 **#define** TPMTRY( s, c )

**Value:**

```
if (c != TPM_SUCCESS) { \
    status = s; \
    printf("ERROR in %s at %s:%i code: %s.\n", __func__, __FILE__, __LINE__,
    tpm_get_error_name(status)); \
    goto abort_egress; \
} else { \
    status = c; \
}
```

4.169.1.305 **#define** TPMTRYRETURN( c )

**Value:**

```
do { status = c; \
    if (status != TPM_SUCCESS) { \
        fprintf(stderr, "ERROR in %s at %s:%i code: %s.\n", __func__, __FILE__,
        __LINE__, tpm_get_error_name(status)); \
        goto abort_egress; \
    } \
} while(0)
```

4.169.1.306 **#define** TRUE 0x01

4.169.1.307 **#define** TSC\_ORD\_PhysicalPresence (10UL + TPM\_CONNECTION\_ORDINAL)

## 4.169.2 Typedef Documentation

4.169.2.1 **typedef** unsigned char **BOOL**

4.169.2.2 **typedef** unsigned char **BYTE**

4.169.2.3 **typedef** struct pack\_buf\_t pack\_buf\_t

4.169.2.4 **typedef** struct pack\_constbuf\_t pack\_constbuf\_t

4.169.2.5 **typedef** TPM\_HANDLE TCPA\_ENCHANDLE

4.169.2.6 **typedef** TPM\_HANDLE TCPA\_ENTITYHANDLE

4.169.2.7 **typedef** TPM\_HANDLE TCPA\_HASHHANDLE

4.169.2.8 **typedef** TPM\_HANDLE TCPA\_HMACHANDLE

4.169.2.9 **typedef** struct TCS\_AUTH TCS\_AUTH

4.169.2.10 **typedef** UINT32 TCS\_AUTHHANDLE

- 4.169.2.11 typedef UINT32 TCS\_CONTEXT\_HANDLE
- 4.169.2.12 typedef UINT32 TCS\_KEY\_HANDLE
- 4.169.2.13 typedef UINT32 TPM\_ALGORITHM\_ID
- 4.169.2.14 typedef BYTE TPM\_AUTH\_DATA\_USAGE
- 4.169.2.15 typedef BYTE TPM\_AUTHDATA[TPM\_DIGEST\_SIZE]
- 4.169.2.16 typedef TPM\_HANDLE TPM\_AUTHHANDLE
- 4.169.2.17 typedef struct TPM\_BOUND\_DATA TPM\_BOUND\_DATA
- 4.169.2.18 typedef UINT32 TPM\_CAPABILITY\_AREA
- 4.169.2.19 typedef TPM\_DIGEST TPM\_CHOSENID\_HASH
- 4.169.2.20 typedef UINT32 TPM\_COMMAND\_CODE
- 4.169.2.21 typedef TPM\_DIGEST TPM\_COMPOSITE\_HASH
- 4.169.2.22 typedef struct TPM\_DIGEST TPM\_DIGEST
- 4.169.2.23 typedef UINT32 TPM\_DIRINDEX
- 4.169.2.24 typedef TPM\_DIGEST TPM\_DIRVALUE
- 4.169.2.25 typedef UINT16 TPM\_ENC\_SCHEME
- 4.169.2.26 typedef TPM\_AUTHDATA TPM\_ENCAUTH
- 4.169.2.27 typedef UINT16 TPM\_ENTITY\_TYPE
- 4.169.2.28 typedef UINT32 TPM\_HANDLE
- 4.169.2.29 typedef TPM\_DIGEST TPM\_HMAC
- 4.169.2.30 typedef struct TPM\_KEY TPM\_KEY
- 4.169.2.31 typedef UINT32 TPM\_KEY\_FLAGS
- 4.169.2.32 typedef TPM\_HANDLE TPM\_KEY\_HANDLE
- 4.169.2.33 typedef struct TPM\_KEY\_PARMS TPM\_KEY\_PARMS
- 4.169.2.34 typedef UINT16 TPM\_KEY\_USAGE
- 4.169.2.35 typedef UINT16 TPM\_MIGRATE\_SCHEME
- 4.169.2.36 typedef struct TPM\_NONCE TPM\_NONCE
- 4.169.2.37 typedef BYTE TPM\_PAYLOAD\_TYPE
- 4.169.2.38 typedef struct TPM\_PCR\_COMPOSITE TPM\_PCR\_COMPOSITE

4.169.2.39 `typedef struct TPM_PCR_INFO TPM_PCR_INFO`

4.169.2.40 `typedef struct TPM_PCR_SELECTION TPM_PCR_SELECTION`

4.169.2.41 `typedef UINT32 TPM_PCRINDEX`

4.169.2.42 `typedef TPM_DIGEST TPM_PCRVALUE`

4.169.2.43 `typedef UINT16 TPM_PHYSICAL_PRESENCE`

4.169.2.44 `typedef UINT16 TPM_PROTOCOL_ID`

4.169.2.45 `typedef struct TPM_PUBKEY TPM_PUBKEY`

4.169.2.46 `typedef UINT32 TPM_RESOURCE_TYPE`

4.169.2.47 `typedef UINT32 TPM_RESULT`

4.169.2.48 `typedef struct TPM_RSA_KEY_PARMS TPM_RSA_KEY_PARMS`

4.169.2.49 `typedef TPM_AUTHDATA TPM_SECRET`

4.169.2.50 `typedef UINT16 TPM_SIG_SCHEME`

4.169.2.51 `typedef UINT16 TPM_STARTUP_TYPE`

4.169.2.52 `typedef struct TPM_STORE_PUBKEY TPM_STORE_PUBKEY`

4.169.2.53 `typedef struct TPM_STORED_DATA TPM_STORED_DATA`

4.169.2.54 `typedef UINT16 TPM_TAG`

4.169.2.55 `typedef struct TPM_VERSION TPM_VERSION`

4.169.2.56 `typedef unsigned short UINT16`

4.169.2.57 `typedef unsigned int UINT32`

4.169.2.58 `typedef unsigned long long UINT64`

## 4.170 xen/tools/vtpm\_manager/vtpmmgrtalk/vtpmmgrtalk.c File Reference

```
#include <unistd.h>
#include <fcntl.h>
#include <sys/file.h>
#include <stdio.h>
#include <errno.h>
#include <string.h>
#include <stdint.h>
#include "../manager/vtpm_manager.h"
```

### Functions

- `int main` (int argc, char \*\*argv)

## 4.170.1 Function Documentation

4.170.1.1 `int main ( int argc, char ** argv )`

## 4.171 xen/tools/xenstore/xenstored\_core.c File Reference

```
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/socket.h>
#include <sys/select.h>
#include <sys/un.h>
#include <sys/time.h>
#include <time.h>
#include <unistd.h>
#include <fcntl.h>
#include <stdbool.h>
#include <stdio.h>
#include <stdarg.h>
#include <stdlib.h>
#include <syslog.h>
#include <string.h>
#include <errno.h>
#include <dirent.h>
#include <getopt.h>
#include <signal.h>
#include <assert.h>
#include <setjmp.h>
#include "utils.h"
#include "list.h"
#include "talloc.h"
#include "xs_lib.h"
#include "xenstored_core.h"
#include "xenstored_watch.h"
#include "xenstored_transaction.h"
#include "xenstored_domain.h"
#include "xenctrl.h"
#include "tdb.h"
#include "hashtable.h"
```

## Macros

- `#define log(...)`
- `#define TDB_FLAGS 0`

## Functions

- `LIST_HEAD` (connections)
- `TDB_CONTEXT * tdb_context` (struct connection \*conn)
- `bool replace_tdb` (const char \*newname, TDB\_CONTEXT \*newtdb)
- `void trace` (const char \*fmt,...)
- `void trace_create` (const void \*data, const char \*type)
- `void trace_destroy` (const void \*data, const char \*type)
- `bool is_child` (const char \*child, const char \*parent)
- `struct node * get_node` (struct connection \*conn, const char \*name, enum xs\_perm\_type perm)

- unsigned int [get\\_strings](#) (struct buffered\_data \*data, char \*vec[], unsigned int num)
- void [send\\_reply](#) (struct connection \*conn, enum xsd\_sockmsg\_type type, const void \*data, unsigned int len)
- void [send\\_ack](#) (struct connection \*conn, enum xsd\_sockmsg\_type type)
- void [send\\_error](#) (struct connection \*conn, int error)
- bool [is\\_valid\\_nodename](#) (const char \*node)
- char \* [canonicalize](#) (struct connection \*conn, const char \*node)
- bool [check\\_event\\_node](#) (const char \*node)
- struct connection \* [new\\_connection](#) (connwritefn\_t \*write, connreadfn\_t \*read)
- void [dump\\_conn](#) (struct connection \*conn)
- int [main](#) (int argc, char \*argv[])

## Variables

- [xc\\_evtchn](#) \* [xce\\_handle](#)
- int [quota\\_nb\\_entry\\_per\\_domain](#) = 1000
- int [quota\\_nb\\_watch\\_per\\_domain](#) = 128
- int [quota\\_max\\_entry\\_size](#) = 2048
- int [quota\\_max\\_transaction](#) = 10

## 4.171.1 Macro Definition Documentation

### 4.171.1.1 #define log( ... )

#### Value:

```
do {
    char *s = talloc\_asprintf(NULL, \_\_VA\_ARGS\_\_);
    trace("s\n", s);
    syslog(LOG_ERR, "s", s);
    talloc\_free(s);
} while (0)
```

### 4.171.1.2 #define TDB\_FLAGS 0

## 4.171.2 Function Documentation

### 4.171.2.1 char\* canonicalize ( struct connection \* conn, const char \* node )

### 4.171.2.2 bool check\_event\_node ( const char \* node )

### 4.171.2.3 void dump\_conn ( struct connection \* conn )

### 4.171.2.4 struct node\* get\_node ( struct connection \* conn, const char \* name, enum xs\_perm\_type perm )

If it fails, returns NULL and sets errno.

### 4.171.2.5 unsigned int get\_strings ( struct buffered\_data \* data, char \* vec[], unsigned int num )

Break input into vectors, return the number, fill in up to num of them. Always returns the actual number of nuls in the input. Stores the positions of the starts of the nul-terminated strings in vec. Callers who use this and then rely only on vec[] will ignore any data after the final nul.

### 4.171.2.6 bool is\_child ( const char \* child, const char \* parent )

Is child a subnode of parent, or equal?

4.171.2.7 `bool is_valid_nodename ( const char * node )`

4.171.2.8 `LIST_HEAD ( connections )`

4.171.2.9 `int main ( int argc, char * argv[] )`

4.171.2.10 `struct connection* new_connection ( connwritefn_t * write, connreadfn_t * read )`

4.171.2.11 `bool replace_tdb ( const char * newname, TDB_CONTEXT * newtdb )`

4.171.2.12 `void send_ack ( struct connection * conn, enum xsd_sockmsg_type type )`

Some routines (write, mkdir, etc) just need a non-error return

4.171.2.13 `void send_error ( struct connection * conn, int error )`

4.171.2.14 `void send_reply ( struct connection * conn, enum xsd_sockmsg_type type, const void * data, unsigned int len )`

4.171.2.15 `TDB_CONTEXT* tdb_context ( struct connection * conn )`

4.171.2.16 `void trace ( const char * fmt, ... )`

4.171.2.17 `void trace_create ( const void * data, const char * type )`

4.171.2.18 `void trace_destroy ( const void * data, const char * type )`

### 4.171.3 Variable Documentation

4.171.3.1 `int quota_max_entry_size = 2048`

4.171.3.2 `int quota_max_transaction = 10`

4.171.3.3 `int quota_nb_entry_per_domain = 1000`

4.171.3.4 `int quota_nb_watch_per_domain = 128`

4.171.3.5 `xc_evtchn* xce_handle`

## 4.172 xen/tools/xenstore/xenstored\_domain.c File Reference

```
#include <stdio.h>
#include <sys/mman.h>
#include <unistd.h>
#include <stdlib.h>
#include <stdarg.h>
#include <xenctrl.h>
#include "utils.h"
#include "talloc.h"
#include "xenstored_core.h"
#include "xenstored_domain.h"
#include "xenstored_transaction.h"
#include "xenstored_watch.h"
```

## Data Structures

- struct [domain](#)

## Macros

- `#define DOMID_DOMT 1`
- `#define DOMID_DOMT 1`

## Functions

- void [handle\\_event](#) (void)
- bool [domain\\_can\\_read](#) (struct connection \*conn)
- bool [domain\\_is\\_unprivileged](#) (struct connection \*conn)
- bool [domain\\_can\\_write](#) (struct connection \*conn)
- void [do\\_introduce](#) (struct connection \*conn, struct buffered\_data \*in)
- void [do\\_set\\_target](#) (struct connection \*conn, struct buffered\_data \*in)
- void [do\\_release](#) (struct connection \*conn, const char \*domid\_str)
- void [do\\_resume](#) (struct connection \*conn, const char \*domid\_str)
- void [do\\_get\\_domain\\_path](#) (struct connection \*conn, const char \*domid\_str)
- void [do\\_is\\_domain\\_introduced](#) (struct connection \*conn, const char \*domid\_str)
- const char \* [get\\_implicit\\_path](#) (const struct connection \*conn)
- void [restore\\_existing\\_connections](#) (void)
- void [domain\\_init](#) (void)
- void [domain\\_entry\\_inc](#) (struct connection \*conn, struct node \*node)
- void [domain\\_entry\\_dec](#) (struct connection \*conn, struct node \*node)
- void [domain\\_entry\\_fix](#) (unsigned int domid, int num)
- int [domain\\_entry](#) (struct connection \*conn)
- void [domain\\_watch\\_inc](#) (struct connection \*conn)
- void [domain\\_watch\\_dec](#) (struct connection \*conn)
- int [domain\\_watch](#) (struct connection \*conn)

## Variables

- [xc\\_evtchn](#) \* [xce\\_handle](#) = NULL

### 4.172.1 Macro Definition Documentation

4.172.1.1 `#define DOMID_DOMT 1`

4.172.1.2 `#define DOMID_DOMT 1`

### 4.172.2 Function Documentation

4.172.2.1 void [do\\_get\\_domain\\_path](#) ( struct connection \* *conn*, const char \* *domid\_str* )

4.172.2.2 void [do\\_introduce](#) ( struct connection \* *conn*, struct buffered\_data \* *in* )

domid, mfn, evtchn, path



4.172.2.3 void do\_is\_domain\_introduced ( struct connection \* *conn*, const char \* *domid\_str* )

4.172.2.4 void do\_release ( struct connection \* *conn*, const char \* *domid\_str* )

domid

4.172.2.5 void do\_resume ( struct connection \* *conn*, const char \* *domid\_str* )

4.172.2.6 void do\_set\_target ( struct connection \* *conn*, struct buffered\_data \* *in* )

4.172.2.7 bool domain\_can\_read ( struct connection \* *conn* )

4.172.2.8 bool domain\_can\_write ( struct connection \* *conn* )

4.172.2.9 int domain\_entry ( struct connection \* *conn* )

4.172.2.10 void domain\_entry\_dec ( struct connection \* *conn*, struct node \* *node* )

4.172.2.11 void domain\_entry\_fix ( unsigned int *domid*, int *num* )

4.172.2.12 void domain\_entry\_inc ( struct connection \* *conn*, struct node \* *node* )

4.172.2.13 void domain\_init ( void )

4.172.2.14 bool domain\_is\_unprivileged ( struct connection \* *conn* )

4.172.2.15 int domain\_watch ( struct connection \* *conn* )

4.172.2.16 void domain\_watch\_dec ( struct connection \* *conn* )

4.172.2.17 void domain\_watch\_inc ( struct connection \* *conn* )

4.172.2.18 const char\* get\_implicit\_path ( const struct connection \* *conn* )

Returns the implicit path of a connection (only domains have this)

4.172.2.19 void handle\_event ( void )

4.172.2.20 void restore\_existing\_connections ( void )

Restore existing connections.

## 4.172.3 Variable Documentation

4.172.3.1 xc\_evtchn\* xce\_handle = NULL

## 4.173 xen/tools/xenstore/xenstored\_transaction.c File Reference

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/wait.h>
#include <sys/time.h>
#include <time.h>
#include <assert.h>
#include <stdarg.h>
#include <stdlib.h>
#include <fcntl.h>
#include <unistd.h>
#include "talloc.h"
#include "list.h"
#include "xenstored_transaction.h"
#include "xenstored_watch.h"
#include "xenstored_domain.h"
#include "xs_lib.h"
#include "utils.h"
```

### Data Structures

- struct [changed\\_node](#)
- struct [changed\\_domain](#)
- struct [transaction](#)

### Functions

- TDB\_CONTEXT \* [tdb\\_transaction\\_context](#) (struct [transaction](#) \*trans)
- void [add\\_change\\_node](#) (struct [transaction](#) \*trans, const char \*node, bool recurse)
- struct [transaction](#) \* [transaction\\_lookup](#) (struct connection \*conn, uint32\_t id)
- void [do\\_transaction\\_start](#) (struct connection \*conn, struct buffered\_data \*in)
- void [do\\_transaction\\_end](#) (struct connection \*conn, const char \*arg)
- void [transaction\\_entry\\_inc](#) (struct [transaction](#) \*trans, unsigned int domid)
- void [transaction\\_entry\\_dec](#) (struct [transaction](#) \*trans, unsigned int domid)
- void [conn\\_delete\\_all\\_transactions](#) (struct connection \*conn)

### Variables

- int [quota\\_max\\_transaction](#)

### 4.173.1 Function Documentation

#### 4.173.1.1 void add\_change\_node ( struct transaction \* trans, const char \* node, bool recurse )

Callers get a change node (which can fail) and only commit after they've finished. This way they don't have to unwind eg. a write.

4.173.1.2 void conn\_delete\_all\_transactions ( struct connection \* *conn* )

4.173.1.3 void do\_transaction\_end ( struct connection \* *conn*, const char \* *arg* )

4.173.1.4 void do\_transaction\_start ( struct connection \* *conn*, struct buffered\_data \* *in* )

4.173.1.5 TDB\_CONTEXT\* tdb\_transaction\_context ( struct transaction \* *trans* )

Return tdb context to use for this connection.

4.173.1.6 void transaction\_entry\_dec ( struct transaction \* *trans*, unsigned int *domid* )

4.173.1.7 void transaction\_entry\_inc ( struct transaction \* *trans*, unsigned int *domid* )

4.173.1.8 struct transaction\* transaction\_lookup ( struct connection \* *conn*, uint32\_t *id* )

## 4.173.2 Variable Documentation

4.173.2.1 int quota\_max\_transaction

## 4.174 xen/tools/xenvfsd/xenvfsd.c File Reference

```
#include <err.h>
#include <errno.h>
#include <error.h>
#include <fcntl.h>
#include <getopt.h>
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <termios.h>
#include <unistd.h>
#include <sys/stat.h>
#include <sys/ioctl.h>
#include <sys/mman.h>
#include <assert.h>
#include <sys/types.h>
#include <libxlutil.h>
#include <xenctrl.h>
#include <xs.h>
#include <vfs_intf.h>
```

## Macros

- #define [\\_GNU\\_SOURCE](#)
- #define [DOMID\\_Z](#) 0
- #define [DOMID\\_T](#) 1
- #define [SET\\_OTHEREND](#)(a)

## Functions

- int [read\\_arg](#) (struct xs\_handle \*xsh, char \*\*target, int num)

- int [read\\_arg\\_int](#) (struct xs\_handle \*xsh, int \*itarget, int num)
- int [xs\\_write\\_int](#) (struct xs\_handle \*h, xs\_transaction\_t t, const char \*path, int data)
- int [read\\_to\\_gntref](#) (xc\_gnttab \*xcg, int fd, uint32\_t gref, size\_t nbyte)
- int [write\\_from\\_gntref](#) (xc\_gnttab \*xcg, int fd, uint32\_t gref, size\_t nbyte)
- int [stat\\_to\\_gntref](#) (xc\_gnttab \*xcg, char \*path, uint32\_t gref)
- void \* [map\\_gref\\_map](#) (xc\_gnttab \*xcg, domid\_t otherend, uint32\_t gref, uint32\_t \*num)
- int [chk\\_map\\_gref\\_map](#) (uint8\_t \*\*p, size\_t nbyte, xc\_gnttab \*xcg, domid\_t otherend, uint32\_t gref, uint32\_t \*num)
- domid\_t [read\\_dyn\\_domid](#) (void)
- int [main](#) (int argc, char \*\*argv)

## Variables

- const char \* [path\\_req](#) = VFS\_PATH "/req"
- const char \* [path\\_rsp](#) = VFS\_PATH "/rsp"
- const char \* [path\\_arg1](#) = VFS\_PATH "/arg1"
- const char \* [path\\_arg2](#) = VFS\_PATH "/arg2"
- const char \* [path\\_arg3](#) = VFS\_PATH "/arg3"
- const char \* [path\\_arg4](#) = VFS\_PATH "/arg4"
- int [dynamic](#) = 0
- char \* [dyn\\_file](#) = NULL

## 4.174.1 Macro Definition Documentation

4.174.1.1 `#define _GNU_SOURCE`

4.174.1.2 `#define DOMID_T 1`

4.174.1.3 `#define DOMID_Z 0`

4.174.1.4 `#define SET_OTHEREND( a )`

**Value:**

```
do {
    if (!dynamic)
        a = DOMID_T;
    else
        a = read_dyn_domid();
} while(0)
```

## 4.174.2 Function Documentation

4.174.2.1 int [chk\\_map\\_gref\\_map](#) ( uint8\_t \*\* p, size\_t nbyte, xc\_gnttab \* xcg, domid\_t otherend, uint32\_t gref, uint32\_t \* num )

4.174.2.2 int [main](#) ( int argc, char \*\* argv )

4.174.2.3 void \* [map\\_gref\\_map](#) ( xc\_gnttab \* xcg, domid\_t otherend, uint32\_t gref, uint32\_t \* num )

4.174.2.4 int [read\\_arg](#) ( struct xs\_handle \* xsh, char \*\* target, int num )

4.174.2.5 int [read\\_arg\\_int](#) ( struct xs\_handle \* xsh, int \* itarget, int num )

4.174.2.6 domid\_t [read\\_dyn\\_domid](#) ( void )

4.174.2.7 int read\_to\_gntref ( xc\_gnttab \* *xcg*, int *fd*, uint32\_t *gref*, size\_t *nbyte* )

4.174.2.8 int stat\_to\_gntref ( xc\_gnttab \* *xcg*, char \* *path*, uint32\_t *gref* )

4.174.2.9 int write\_from\_gntref ( xc\_gnttab \* *xcg*, int *fd*, uint32\_t *gref*, size\_t *nbyte* )

4.174.2.10 int xs\_write\_int ( struct xs\_handle \* *h*, xs\_transaction\_t *t*, const char \* *path*, int *data* )

### 4.174.3 Variable Documentation

4.174.3.1 char\* dyn\_file = NULL

4.174.3.2 int dynamic = 0

4.174.3.3 const char \* path\_arg1 = VFS\_PATH "/arg1"

4.174.3.4 const char \* path\_arg2 = VFS\_PATH "/arg2"

4.174.3.5 const char \* path\_arg3 = VFS\_PATH "/arg3"

4.174.3.6 const char \* path\_arg4 = VFS\_PATH "/arg4"

4.174.3.7 const char\* path\_req = VFS\_PATH "/req"

4.174.3.8 const char \* path\_rsp = VFS\_PATH "/rsp"

## 4.175 xen/xen/arch/x86/domain\_build.c File Reference

```
#include <xen/config.h>
#include <xen/init.h>
#include <xen/lib.h>
#include <xen/ctype.h>
#include <xen/sched.h>
#include <xen/sched-if.h>
#include <xen/smp.h>
#include <xen/delay.h>
#include <xen/event.h>
#include <xen/console.h>
#include <xen/kernel.h>
#include <xen/domain.h>
#include <xen/version.h>
#include <xen/iocap.h>
#include <xen/bitops.h>
#include <xen/compat.h>
#include <xen/libelf.h>
#include <asm/regs.h>
#include <asm/system.h>
#include <asm/io.h>
#include <asm/processor.h>
#include <asm/desc.h>
#include <asm/i387.h>
#include <asm/paging.h>
#include <asm/p2m.h>
#include <asm/e820.h>
#include <asm/acpi.h>
#include <asm/setup.h>
#include <asm/bzimage.h>
#include <asm/io_apic.h>
#include <public/version.h>
```

### Macros

- `#define CAAS_DEVELOP`
- `#define round_pgup(_p) (((_p)+(PAGE_SIZE-1))&PAGE_MASK)`
- `#define round_pgdown(_p) ((_p)&PAGE_MASK)`
- `#define REVERSE_START ((v_end - v_start) >> PAGE_SHIFT)`
- `#define pfn (nr_pages - 1 - (pfn - (alloc_epfn - alloc_spfn)))`
- `#define TPM_LOC0 0xFED40000 >> PAGE_SHIFT`
- `#define TPM_LOCALL 0xFED44000 >> PAGE_SHIFT`

### Functions

- `integer_param` ("caas\_enforcing", caas\_enforcing)
- `custom_param` ("dom0\_mem", parse\_dom0\_mem)
- `integer_param` ("dom0\_max\_vcpus", opt\_dom0\_max\_vcpus)
- `integer_param` ("domt\_max\_vcpus", opt\_domt\_max\_vcpus)
- `struct vcpu * __init alloc_dom0_vcpu0` (void)
- `struct vcpu * __init alloc_domt_vcpu0` (void)
- `boolean_param` ("dom0\_shadow", opt\_dom0\_shadow)
- `string_param` ("dom0\_ioports\_disable", opt\_dom0\_ioports\_disable)

- `int __init construct_dom (enum domtype t, struct domain *d, const module_t *image, unsigned long image_headroom, module_t *initrd, void *(*bootstrap_map)(const module_t *), char *cmdline)`

### 4.175.1 Macro Definition Documentation

4.175.1.1 `#define CAAS_DEVELOP`

4.175.1.2 `#define pfn (nr_pages - 1 - (pfn - (alloc_epfn - alloc_spfn)))`

4.175.1.3 `#define REVERSE_START ((v_end - v_start) >> PAGE_SHIFT)`

4.175.1.4 `#define round_pgdown( _p ) (( _p ) & PAGE_MASK)`

4.175.1.5 `#define round_pgup( _p ) ((( _p ) + (PAGE_SIZE - 1)) & PAGE_MASK)`

4.175.1.6 `#define TPM_LOC0 0xFED40000 >> PAGE_SHIFT`

4.175.1.7 `#define TPM_LOCALL 0xFED44000 >> PAGE_SHIFT`

### 4.175.2 Function Documentation

4.175.2.1 `struct vcpu* __init alloc_dom0_vcpu0 ( void )`

4.175.2.2 `struct vcpu* __init alloc_domt_vcpu0 ( void )`

4.175.2.3 `boolean_param ( "dom0_shadow", opt_dom0_shadow )`

4.175.2.4 `int __init construct_dom ( enum domtype t, struct domain * d, const module_t * image, unsigned long image_headroom, module_t * initrd, void (*)(const module_t *) bootstrap_map, char * cmdline )`

4.175.2.5 `custom_param ( "dom0_mem", parse_dom0_mem )`

4.175.2.6 `integer_param ( "caas_enforcing", caas_enforcing )`

4.175.2.7 `integer_param ( "dom0_max_vcpus", opt_dom0_max_vcpus )`

4.175.2.8 `integer_param ( "domt_max_vcpus", opt_domt_max_vcpus )`

4.175.2.9 `string_param ( "dom0_ioports_disable", opt_dom0_ioports_disable )`

## 4.176 xen/xen/arch/x86/setup.c File Reference

```
#include <xen/config.h>
#include <xen/init.h>
#include <xen/lib.h>
#include <xen/sched.h>
#include <xen/sched-if.h>
#include <xen/domain.h>
#include <xen/serial.h>
#include <xen/softirq.h>
#include <xen/acpi.h>
#include <xen/console.h>
#include <xen/trace.h>
#include <xen/multiboot.h>
#include <xen/domain_page.h>
#include <xen/version.h>
#include <xen/gdbstub.h>
#include <xen/percpu.h>
#include <xen/hypercall.h>
#include <xen/keyhandler.h>
#include <xen/numa.h>
#include <xen/rcupdate.h>
#include <xen/vga.h>
#include <xen/dmi.h>
#include <xen/nodemask.h>
#include <public/version.h>
#include <asm/bitops.h>
#include <asm/smp.h>
#include <asm/processor.h>
#include <asm/mpspec.h>
#include <asm/apic.h>
#include <asm/desc.h>
#include <asm/paging.h>
#include <asm/e820.h>
#include <xsm/acm/acm_hooks.h>
#include <xen/kexec.h>
#include <asm/edd.h>
#include <xsm/xsm.h>
#include <asm/tboot.h>
#include <asm/bzimage.h>
#include <asm/mach-generic/mach_apic.h>
#include <asm/setup.h>
#include <xen/cpu.h>
```

### Data Structures

- struct [boot\\_video\\_info](#)

### Macros

- #define [EARLY\\_FAIL](#)(f, a...)
- #define [BOOTSTRAP\\_MAP\\_BASE](#) (16UL << 20)
- #define [BOOTSTRAP\\_MAP\\_LIMIT](#) (1UL << L3\_PAGETABLE\_SHIFT)



## Functions

- [boolean\\_param](#) ("nosmp", opt\_nosmp)
- [integer\\_param](#) ("maxcpus", max\_cpus)
- [boolean\\_param](#) ("watchdog", opt\_watchdog)
- [invbool\\_param](#) ("smep", disable\_smep)
- [custom\\_param](#) ("acpi", parse\_acpi\_param)
- [boolean\\_param](#) ("acpi\_skip\_timer\_override", acpi\_skip\_timer\_override)
- [boolean\\_param](#) ("noapic", skip\_ioapic\_setup)
- [boolean\\_param](#) ("cpuidle", xen\_cpuidle)
- [DEFINE\\_PER\\_CPU](#) (struct tss\_struct, init\_tss)
- [char \\_\\_attribute\\_\\_](#) ((\_\_section\_\_(".bss.stack\_aligned"))) )
- [unsigned long \\_\\_init initial\\_images\\_nrpages](#) (void)
- [void \\_\\_init discard\\_initial\\_images](#) (void)
- [void \\_\\_devinit srat\\_detect\\_node](#) (int cpu)
- [void set\\_pdx\\_range](#) (unsigned long smfn, unsigned long emfn)
- [void init\\_done](#) (void)
- [void \\_\\_init \\_\\_start\\_xen](#) (unsigned long mbi\_p)
- [void arch\\_get\\_xen\\_caps](#) (xen\_capabilities\_info\_t \*info)
- [int xen\\_in\\_range](#) (unsigned long mfn)

## Variables

- [u16 boot\\_edid\\_caps](#)
- [u8 boot\\_edid\\_info](#) [128]
- [struct boot\\_video\\_info boot\\_vid\\_info](#)
- [s8 \\_\\_read\\_mostly xen\\_cpuidle](#) = -1
- [bool\\_t \\_\\_read\\_mostly early\\_boot](#) = 1
- [cpumask\\_t \\_\\_read\\_mostly cpu\\_present\\_map](#)
- [unsigned long \\_\\_read\\_mostly xen\\_phys\\_start](#)
- [unsigned long \\_\\_read\\_mostly mmu\\_cr4\\_features](#) = X86\_CR4\_PSE | X86\_CR4\_PGE | X86\_CR4\_PAE
- [bool\\_t \\_\\_initdata acpi\\_disabled](#)
- [bool\\_t \\_\\_initdata acpi\\_force](#)
- [char \\_\\_init\\_begin](#) []
- [char \\_\\_init\\_end](#) []
- [char \\_\\_bss\\_start](#) []

### 4.176.1 Macro Definition Documentation

4.176.1.1 [#define BOOTSTRAP\\_MAP\\_BASE](#) (16UL << 20)

4.176.1.2 [#define BOOTSTRAP\\_MAP\\_LIMIT](#) (1UL << L3\_PAGETABLE\_SHIFT)

4.176.1.3 [#define EARLY\\_FAIL](#)( f, a... )

#### Value:

```
do {
    printk( f, ## a );
    for ( ; ; ) halt();
} while (0)
```

## 4.176.2 Function Documentation

- 4.176.2.1 `char __attribute__ ( ( __section__( ".bss.stack_aligned" ) )`
- 4.176.2.2 `void __init __start_xen ( unsigned long mbi_p )`
- 4.176.2.3 `void arch_get_xen_caps ( xen_capabilities_info_t * info )`
- 4.176.2.4 `boolean_param ( "nosmp", opt_nosmp )`
- 4.176.2.5 `boolean_param ( "watchdog", opt_watchdog )`
- 4.176.2.6 `boolean_param ( "acpi_skip_timer_override", acpi_skip_timer_override )`
- 4.176.2.7 `boolean_param ( "noapic", skip_ioapic_setup )`
- 4.176.2.8 `boolean_param ( "cpuidle", xen_cpuidle )`
- 4.176.2.9 `custom_param ( "acpi", parse_acpi_param )`
- 4.176.2.10 `DEFINE_PER_CPU ( struct tss_struct, init_tss )`
- 4.176.2.11 `void __init discard_initial_images ( void )`
- 4.176.2.12 `void init_done ( void )`
- 4.176.2.13 `unsigned long __init initial_images_nrpages ( void )`
- 4.176.2.14 `integer_param ( "maxcpus", max_cpus )`
- 4.176.2.15 `invbool_param ( "smep", disable_smep )`
- 4.176.2.16 `void set_pdx_range ( unsigned long smfn, unsigned long emfn )`
- 4.176.2.17 `void __devinit srat_detect_node ( int cpu )`
- 4.176.2.18 `int xen_in_range ( unsigned long mfn )`

## 4.176.3 Variable Documentation

- 4.176.3.1 `char __bss_start[]`
- 4.176.3.2 `char __init_begin[]`
- 4.176.3.3 `char __init_end[]`
- 4.176.3.4 `bool_t __initdata acpi_disabled`
- 4.176.3.5 `bool_t __initdata acpi_force`
- 4.176.3.6 `u16 boot_edid_caps`
- 4.176.3.7 `u8 boot_edid_info[128]`
- 4.176.3.8 `struct boot_video_info boot_vid_info`

4.176.3.9 `cpumask_t __read_mostly cpu_present_map`

4.176.3.10 `bool_t __read_mostly early_boot = 1`

4.176.3.11 `unsigned long __read_mostly mmu_cr4_features = X86_CR4_PSE | X86_CR4_PGE | X86_CR4_PAE`

4.176.3.12 `s8 __read_mostly xen_cpuidle = -1`

4.176.3.13 `unsigned long __read_mostly xen_phys_start`

## 4.177 xen/xen/common/domain.c File Reference

```
#include <xen/config.h>
#include <xen/compat.h>
#include <xen/init.h>
#include <xen/lib.h>
#include <xen/ctype.h>
#include <xen/errno.h>
#include <xen/sched.h>
#include <xen/domain.h>
#include <xen/mm.h>
#include <xen/event.h>
#include <xen/time.h>
#include <xen/console.h>
#include <xen/softirq.h>
#include <xen/tasklet.h>
#include <xen/domain_page.h>
#include <xen/rangeset.h>
#include <xen/guest_access.h>
#include <xen/hypercall.h>
#include <xen/delay.h>
#include <xen/shutdown.h>
#include <xen/percpu.h>
#include <xen/multicall.h>
#include <xen/rcupdate.h>
#include <xen/wait.h>
#include <acpi/cpufreq/cpufreq.h>
#include <asm/debugger.h>
#include <public/sched.h>
#include <public/sysctl.h>
#include <public/vcpu.h>
#include <xsm/xsm.h>
#include <xen/trace.h>
#include <xen/tmem.h>
```

### Data Structures

- struct [migrate\\_info](#)

### Macros

- #define [DOMAIN\\_HASH\\_SIZE](#) 256
- #define [DOMAIN\\_HASH](#)(\_id) ((int)(\_id)&(DOMAIN\_HASH\_SIZE-1))

## Functions

- [boolean\\_param](#) ("dom0\_vcpus\_pin", opt\_dom0\_vcpus\_pin)
- [custom\\_param](#) ("cpufreq", setup\_cpufreq\_option)
- [DEFINE\\_SPINLOCK](#) (domlist\_update\_lock)
- [DEFINE\\_RCU\\_READ\\_LOCK](#) (domlist\_read\_lock)
- [int current\\_domain\\_id](#) (void)
- [struct vcpu \\* alloc\\_vcpu](#) (struct [domain \\*d](#), unsigned int vcpu\_id, unsigned int cpu\_id)
- [custom\\_param](#) ("extra\_guest\_irqs", parse\_extra\_guest\_irqs)
- [struct domain \\* domain\\_create](#) (domid\_t domid, unsigned int domcr\_flags, ssidref\_t ssidref)
- [void domain\\_update\\_node\\_affinity](#) (struct [domain \\*d](#))
- [struct domain \\* get\\_domain\\_by\\_id](#) (domid\_t dom)
- [struct domain \\* rcu\\_lock\\_domain\\_by\\_id](#) (domid\_t dom)
- [int rcu\\_lock\\_target\\_domain\\_by\\_id](#) (domid\_t dom, struct [domain \\*\\*d](#))
- [int rcu\\_lock\\_remote\\_target\\_domain\\_by\\_id](#) (domid\_t dom, struct [domain \\*\\*d](#))
- [int domain\\_kill](#) (struct [domain \\*d](#))
- [void \\_\\_domain\\_crash](#) (struct [domain \\*d](#))
- [void \\_\\_domain\\_crash\\_synchronous](#) (void)
- [void domain\\_shutdown](#) (struct [domain \\*d](#), u8 reason)
- [void domain\\_resume](#) (struct [domain \\*d](#))
- [int vcpu\\_start\\_shutdown\\_deferral](#) (struct [vcpu \\*v](#))
- [void vcpu\\_end\\_shutdown\\_deferral](#) (struct [vcpu \\*v](#))
- [void domain\\_pause\\_for\\_debugger](#) (void)
- [void domain\\_destroy](#) (struct [domain \\*d](#))
- [void vcpu\\_pause](#) (struct [vcpu \\*v](#))
- [void vcpu\\_pause\\_nosync](#) (struct [vcpu \\*v](#))
- [void vcpu\\_unpause](#) (struct [vcpu \\*v](#))
- [void domain\\_pause](#) (struct [domain \\*d](#))
- [void domain\\_unpause](#) (struct [domain \\*d](#))
- [void domain\\_pause\\_by\\_systemcontroller](#) (struct [domain \\*d](#))
- [void domain\\_unpause\\_by\\_systemcontroller](#) (struct [domain \\*d](#))
- [int boot\\_vcpu](#) (struct [domain \\*d](#), int vcpuid, [vcpu\\_guest\\_context\\_u](#) ctxt)
- [void vcpu\\_reset](#) (struct [vcpu \\*v](#))
- [long do\\_vcpu\\_op](#) (int cmd, int vcpuid, XEN\_GUEST\_HANDLE(void) arg)
- [long vm\\_assist](#) (struct [domain \\*p](#), unsigned int cmd, unsigned int type)
- [int continue\\_hypercalle\\_on\\_cpu](#) (unsigned int cpu, long(\*func)(void \*data), void \*data)

## Variables

- unsigned int [xen\\_processor\\_pmbits](#) = XEN\_PROCESSOR\_PM\_PX
- enum [cpufreq\\_controller](#) [cpufreq\\_controller](#) = [FREQCTL\\_xen](#)
- struct [domain \\* domain\\_list](#)
- struct [domain \\* dom0](#)
- struct [domain \\* domt](#)
- struct [vcpu \\*idle\\_vcpu](#)[NR\_CPUS] [\\_\\_read\\_mostly](#)
- [vcpu\\_info\\_t](#) [dummy\\_vcpu\\_info](#)

### 4.177.1 Macro Definition Documentation

4.177.1.1 `#define DOMAIN_HASH( _id ) ((int)(_id)&(DOMAIN_HASH_SIZE-1))`

4.177.1.2 `#define DOMAIN_HASH_SIZE 256`

### 4.177.2 Function Documentation

4.177.2.1 `void __domain_crash ( struct domain * d )`

Mark specified domain as crashed. This function always returns, even if the caller is the specified domain. The domain is not synchronously descheduled from any processor.

4.177.2.2 `void __domain_crash_synchronous ( void )`

Mark current domain as crashed and synchronously deschedule from the local processor. This function never returns.

4.177.2.3 `struct vcpu* alloc_vcpu ( struct domain * d, unsigned int vcpu_id, unsigned int cpu_id )`

4.177.2.4 `boolean_param ( "dom0_vcpus_pin", opt_dom0_vcpus_pin )`

4.177.2.5 `int boot_vcpu ( struct domain * d, int vcpuid, vcpu_guest_context_u ctxt )`

4.177.2.6 `int continue_hypcall_on_cpu ( unsigned int cpu, long(*)(void *data) func, void * data )`

Continue the current hypercall via *func(data)* on specified *cpu*. If this function returns 0 then the function is guaranteed to run at some point in the future. If this function returns an error code then the function has not been and will not be executed.

4.177.2.7 `int current_domain_id ( void )`

4.177.2.8 `custom_param ( "cpufreq", setup_cpufreq_option )`

4.177.2.9 `custom_param ( "extra_guest_irqs", parse_extra_guest_irqs )`

4.177.2.10 `DEFINE_RCU_READ_LOCK ( domlist_read_lock )`

4.177.2.11 `DEFINE_SPINLOCK ( domlist_update_lock )`

4.177.2.12 `long do_vcpu_op ( int cmd, int vcpuid, XEN_GUEST_HANDLE(void) arg )`

4.177.2.13 `struct domain* domain_create ( domid_t domid, unsigned int domcr_flags, ssidref_t ssidref )`

4.177.2.14 `void domain_destroy ( struct domain * d )`

Release resources belonging to task .

4.177.2.15 `int domain_kill ( struct domain * d )`

4.177.2.16 `void domain_pause ( struct domain * d )`

4.177.2.17 `void domain_pause_by_systemcontroller ( struct domain * d )`

4.177.2.18 void domain\_pause\_for\_debugger ( void )

4.177.2.19 void domain\_resume ( struct domain \* d )

4.177.2.20 void domain\_shutdown ( struct domain \* d, u8 reason )

4.177.2.21 void domain\_unpause ( struct domain \* d )

4.177.2.22 void domain\_unpause\_by\_systemcontroller ( struct domain \* d )

4.177.2.23 void domain\_update\_node\_affinity ( struct domain \* d )

4.177.2.24 struct domain\* get\_domain\_by\_id ( domid\_t dom )

4.177.2.25 struct domain\* rcu\_lock\_domain\_by\_id ( domid\_t dom )

[rcu\\_lock\\_domain\\_by\\_id\(\)](#) is more efficient than [get\\_domain\\_by\\_id\(\)](#). This is the preferred function if the returned domain reference is short lived, but it cannot be used if the domain reference needs to be kept beyond the current scope (e.g., across a softirq). The returned domain reference must be discarded using [rcu\\_unlock\\_domain\(\)](#).

4.177.2.26 int rcu\_lock\_remote\_target\_domain\_by\_id ( domid\_t dom, struct domain \*\* d )

As [rcu\\_lock\\_target\\_domain\\_by\\_id\(\)](#), but will fail EPERM rather than resolve to local domain. Successful return always resolves to a remote domain that the local domain is privileged to control.

4.177.2.27 int rcu\_lock\_target\_domain\_by\_id ( domid\_t dom, struct domain \*\* d )

As above function, but accounts for current domain context:

- Translates target DOMID\_SELF into caller's domain id; and
- Checks that caller has permission to act on the target domain.

4.177.2.28 void vcpu\_end\_shutdown\_deferral ( struct vcpu \* v )

4.177.2.29 void vcpu\_pause ( struct vcpu \* v )

4.177.2.30 void vcpu\_pause\_nosync ( struct vcpu \* v )

4.177.2.31 void vcpu\_reset ( struct vcpu \* v )

4.177.2.32 int vcpu\_start\_shutdown\_deferral ( struct vcpu \* v )

4.177.2.33 void vcpu\_unpause ( struct vcpu \* v )

4.177.2.34 long vm\_assist ( struct domain \* p, unsigned int cmd, unsigned int type )

### 4.177.3 Variable Documentation

4.177.3.1 struct vcpu\* idle\_vcpu[NR\_CPUS] \_\_read\_mostly

4.177.3.2 enum cpufreq\_controller cpufreq\_controller = FREQCTL\_xen

4.177.3.3 struct domain\* dom0

4.177.3.4 struct domain\* domain\_list

4.177.3.5 struct domain\* domt

4.177.3.6 vcpu\_info\_t dummy\_vcpu\_info

4.177.3.7 unsigned int xen\_processor\_pmbits = XEN\_PROCESSOR\_PM\_PX

## 4.178 xen/xen/common/domctl.c File Reference

```
#include <xen/config.h>
#include <xen/types.h>
#include <xen/lib.h>
#include <xen/mm.h>
#include <xen/sched.h>
#include <xen/sched-if.h>
#include <xen/domain.h>
#include <xen/event.h>
#include <xen/domain_page.h>
#include <xen/trace.h>
#include <xen/console.h>
#include <xen/iocap.h>
#include <xen/rcupdate.h>
#include <xen/guest_access.h>
#include <xen/bitmap.h>
#include <xen/paging.h>
#include <asm/current.h>
#include <public/domctl.h>
#include <xsm/xsm.h>
```

### Functions

- long [arch\\_do\\_domctl](#) (struct [xen\\_domctl](#) \*op, XEN\_GUEST\_HANDLE([xen\\_domctl\\_t](#)) u\_domctl)
- int [cpumask\\_to\\_xenctl\\_cpumap](#) (struct [xenctl\\_cpumap](#) \*xenctl\_cpumap, cpumask\_t \*cpumask)
- int [xenctl\\_cpumap\\_to\\_cpumask](#) (cpumask\_t \*cpumask, struct [xenctl\\_cpumap](#) \*xenctl\_cpumap)
- void [getdomaininfo](#) (struct [domain](#) \*d, struct [xen\\_domctl\\_getdomaininfo](#) \*info)
- bool\_t [domctl\\_lock\\_acquire](#) (void)
- void [domctl\\_lock\\_release](#) (void)
- long [do\\_domctl](#) (XEN\_GUEST\_HANDLE([xen\\_domctl\\_t](#)) u\_domctl)

### 4.178.1 Function Documentation

4.178.1.1 long [arch\\_do\\_domctl](#) ( struct [xen\\_domctl](#) \* op, XEN\_GUEST\_HANDLE([xen\\_domctl\\_t](#)) u\_domctl )

4.178.1.2 int [cpumask\\_to\\_xenctl\\_cpumap](#) ( struct [xenctl\\_cpumap](#) \* [xenctl\\_cpumap](#), cpumask\_t \* [cpumask](#) )

4.178.1.3 long [do\\_domctl](#) ( XEN\_GUEST\_HANDLE([xen\\_domctl\\_t](#)) u\_domctl )

4.178.1.4 bool\_t [domctl\\_lock\\_acquire](#) ( void )

4.178.1.5 void [domctl\\_lock\\_release](#) ( void )

4.178.1.6 void [getdomaininfo](#) ( struct [domain](#) \* d, struct [xen\\_domctl\\_getdomaininfo](#) \* info )

4.178.1.7 `int xenctl_cpumap_to_cpumask ( cpumask_t * cpumask, struct xenctl_cpumap * xenctl_cpumap )`

## 4.179 xen/xen/common/memory.c File Reference

```
#include <xen/config.h>
#include <xen/types.h>
#include <xen/lib.h>
#include <xen/mm.h>
#include <xen/perfc.h>
#include <xen/sched.h>
#include <xen/event.h>
#include <xen/paging.h>
#include <xen/iocap.h>
#include <xen/guest_access.h>
#include <xen/hypercall.h>
#include <xen/errno.h>
#include <xen/tmem.h>
#include <xen/tmem_xen.h>
#include <asm/current.h>
#include <asm/hardirq.h>
#include <xen/numa.h>
#include <public/memory.h>
#include <xsm/xsm.h>
#include <xen/trace.h>
```

### Data Structures

- struct [memop\\_args](#)

### Functions

- int [guest\\_remove\\_page](#) (struct [domain](#) \*d, unsigned long gmfn)
- long [do\\_memory\\_op](#) (unsigned long cmd, XEN\_GUEST\_HANDLE(void) arg)

### 4.179.1 Function Documentation

4.179.1.1 `long do_memory_op ( unsigned long cmd, XEN_GUEST_HANDLE(void) arg )`

4.179.1.2 `int guest_remove_page ( struct domain * d, unsigned long gmfn )`

## 4.180 xen/xen/include/asm-x86/setup.h File Reference

```
#include <xen/multiboot.h>
```

### Enumerations

- enum [domtype](#) { [DOM0](#), [DOMT](#) }



## Functions

- void [init\\_done](#) (void)
- void [early\\_cpu\\_init](#) (void)
- void [early\\_time\\_init](#) (void)
- void [early\\_page\\_fault](#) (void)
- int [intel\\_cpu\\_init](#) (void)
- int [amd\\_init\\_cpu](#) (void)
- int [cyrix\\_init\\_cpu](#) (void)
- int [nsc\\_init\\_cpu](#) (void)
- int [centaur\\_init\\_cpu](#) (void)
- int [transmeta\\_init\\_cpu](#) (void)
- void [numa\\_initmem\\_init](#) (unsigned long start\_pfn, unsigned long end\_pfn)
- void [arch\\_init\\_memory](#) (void)
- void [subarch\\_init\\_memory](#) (void)
- void [init\\_IRQ](#) (void)
- void [vesa\\_init](#) (void)
- void [vesa\\_mtrr\\_init](#) (void)
- int [construct\\_dom](#) (enum [domtype](#) t, struct [domain](#) \*d, const module\_t \*kernel, unsigned long kernel\_headroom, module\_t \*initrd, void \*(\*bootstrap\_map)(const module\_t \*), char \*cmdline)
- unsigned long [initial\\_images\\_nrpages](#) (void)
- void [discard\\_initial\\_images](#) (void)

## Variables

- bool\_t [early\\_boot](#)
- unsigned long [xenheap\\_initial\\_phys\\_start](#)

### 4.180.1 Enumeration Type Documentation

#### 4.180.1.1 enum domtype

Enumerator

***DOM0***

***DOMT***

### 4.180.2 Function Documentation

4.180.2.1 int [amd\\_init\\_cpu](#) ( void )

4.180.2.2 void [arch\\_init\\_memory](#) ( void )

4.180.2.3 int [centaur\\_init\\_cpu](#) ( void )

4.180.2.4 int [construct\\_dom](#) ( enum [domtype](#) t, struct [domain](#) \* d, const module\_t \* *kernel*, unsigned long *kernel\_headroom*, module\_t \* *initrd*, void \*(\*)(const module\_t \*) *bootstrap\_map*, char \* *cmdline* )

4.180.2.5 int [cyrix\\_init\\_cpu](#) ( void )

4.180.2.6 void [discard\\_initial\\_images](#) ( void )

4.180.2.7 void [early\\_cpu\\_init](#) ( void )

4.180.2.8 void early\_page\_fault ( void )

4.180.2.9 void early\_time\_init ( void )

4.180.2.10 void init\_done ( void )

4.180.2.11 void init\_IRQ ( void )

4.180.2.12 unsigned long initial\_images\_nrpages ( void )

4.180.2.13 int intel\_cpu\_init ( void )

4.180.2.14 int nsc\_init\_cpu ( void )

4.180.2.15 void numa\_initmem\_init ( unsigned long *start\_pfn*, unsigned long *end\_pfn* )

4.180.2.16 void subarch\_init\_memory ( void )

4.180.2.17 int transmeta\_init\_cpu ( void )

4.180.2.18 void vesa\_init ( void )

4.180.2.19 void vesa\_mtrr\_init ( void )

#### 4.180.3 Variable Documentation

4.180.3.1 bool\_t early\_boot

4.180.3.2 unsigned long xenheap\_initial\_phys\_start

### 4.181 xen/xen/include/public/domctl.h File Reference

```
#include "xen.h"
#include "grant_table.h"
```

#### Data Structures

- struct [xen\\_domctl\\_createdomain](#)
- struct [xen\\_domctl\\_getdomaininfo](#)
- struct [xen\\_domctl\\_getmemlist](#)
- struct [xen\\_domctl\\_getpageframeinfo](#)
- struct [xen\\_domctl\\_getpageframeinfo2](#)
- struct [xen\\_domctl\\_getpageframeinfo3](#)
- struct [xen\\_domctl\\_shadow\\_op\\_stats](#)
- struct [xen\\_domctl\\_shadow\\_op](#)
- struct [xen\\_domctl\\_max\\_mem](#)
- struct [xen\\_domctl\\_vcpucontext](#)
- struct [xen\\_domctl\\_getvcpuinfo](#)
- struct [xen\\_domctl\\_vcpuaffinity](#)
- struct [xen\\_domctl\\_max\\_vcpus](#)
- struct [xen\\_domctl\\_scheduler\\_op](#)
- struct [xen\\_domctl\\_setdomainhandle](#)
- struct [xen\\_domctl\\_setdebugging](#)

- struct [xen\\_domctl\\_irq\\_permission](#)
- struct [xen\\_domctl\\_iomem\\_permission](#)
- struct [xen\\_domctl\\_ioport\\_permission](#)
- struct [xen\\_domctl\\_hypercall\\_init](#)
- struct [xen\\_domctl\\_arch\\_setup](#)
- struct [xen\\_domctl\\_settimeoffset](#)
- struct [xen\\_domctl\\_hvmcontext](#)
- struct [xen\\_domctl\\_address\\_size](#)
- struct [xen\\_domctl\\_real\\_mode\\_area](#)
- struct [xen\\_domctl\\_sendtrigger](#)
- struct [xen\\_domctl\\_assign\\_device](#)
- struct [xen\\_domctl\\_get\\_device\\_group](#)
- struct [xen\\_domctl\\_bind\\_pt\\_irq](#)
- struct [xen\\_domctl\\_memory\\_mapping](#)
- struct [xen\\_domctl\\_ioport\\_mapping](#)
- struct [xen\\_domctl\\_pin\\_mem\\_cacheattr](#)
- struct [xen\\_domctl\\_ext\\_vcpucontext](#)
- struct [xen\\_domctl\\_set\\_opt\\_feature](#)
- struct [xen\\_domctl\\_set\\_target](#)
- struct [xen\\_domctl\\_subscribe](#)
- struct [xen\\_domctl\\_debug\\_op](#)
- struct [xen\\_domctl\\_hvmcontext\\_partial](#)
- struct [xen\\_domctl\\_disable\\_migrate](#)
- struct [xen\\_guest\\_tsc\\_info](#)
- struct [xen\\_domctl\\_tsc\\_info](#)
- struct [xen\\_domctl\\_gdbsx\\_memio](#)
- struct [xen\\_domctl\\_gdbsx\\_pauseunp\\_vcpu](#)
- struct [xen\\_domctl\\_gdbsx\\_domstatus](#)
- struct [xen\\_domctl\\_mem\\_event\\_op](#)
- struct [xen\\_domctl\\_mem\\_sharing\\_op](#)
- struct [xen\\_domctl\\_set\\_access\\_required](#)
- struct [xen\\_domctl](#)

## Macros

- #define [XEN\\_DOMCTL\\_INTERFACE\\_VERSION](#) 0x00000007
- #define [\\_XEN\\_DOMCTL\\_CDF\\_hvm\\_guest](#) 0
- #define [XEN\\_DOMCTL\\_CDF\\_hvm\\_guest](#) (1U<<\_XEN\_DOMCTL\_CDF\_hvm\_guest)
- #define [\\_XEN\\_DOMCTL\\_CDF\\_hap](#) 1
- #define [XEN\\_DOMCTL\\_CDF\\_hap](#) (1U<<\_XEN\_DOMCTL\_CDF\_hap)
- #define [\\_XEN\\_DOMCTL\\_CDF\\_s3\\_integrity](#) 2
- #define [XEN\\_DOMCTL\\_CDF\\_s3\\_integrity](#) (1U<<\_XEN\_DOMCTL\_CDF\_s3\_integrity)
- #define [\\_XEN\\_DOMCTL\\_CDF\\_oos\\_off](#) 3
- #define [XEN\\_DOMCTL\\_CDF\\_oos\\_off](#) (1U<<\_XEN\_DOMCTL\_CDF\_oos\_off)
- #define [\\_XEN\\_DOMCTL\\_CDF\\_priv](#) 5
- #define [XEN\\_DOMCTL\\_CDF\\_priv](#) (1U<<\_XEN\_DOMCTL\_CDF\_priv)
- #define [\\_XEN\\_DOMINF\\_dying](#) 0
- #define [XEN\\_DOMINF\\_dying](#) (1U<<\_XEN\_DOMINF\_dying)
- #define [\\_XEN\\_DOMINF\\_hvm\\_guest](#) 1
- #define [XEN\\_DOMINF\\_hvm\\_guest](#) (1U<<\_XEN\_DOMINF\_hvm\_guest)
- #define [\\_XEN\\_DOMINF\\_shutdown](#) 2
- #define [XEN\\_DOMINF\\_shutdown](#) (1U<<\_XEN\_DOMINF\_shutdown)
- #define [\\_XEN\\_DOMINF\\_paused](#) 3
- #define [XEN\\_DOMINF\\_paused](#) (1U<<\_XEN\_DOMINF\_paused)

- #define `_XEN_DOMINF_blocked` 4
- #define `XEN_DOMINF_blocked` (1U<<\_XEN\_DOMINF\_blocked)
- #define `_XEN_DOMINF_running` 5
- #define `XEN_DOMINF_running` (1U<<\_XEN\_DOMINF\_running)
- #define `_XEN_DOMINF_debugged` 6
- #define `XEN_DOMINF_debugged` (1U<<\_XEN\_DOMINF\_debugged)
- #define `XEN_DOMINF_shutdownmask` 255
- #define `XEN_DOMINF_shutdownshift` 16
- #define `XEN_DOMCTL_PFINFO_LTAB_SHIFT` 28
- #define `XEN_DOMCTL_PFINFO_NOTAB` (0x0U<<28)
- #define `XEN_DOMCTL_PFINFO_L1TAB` (0x1U<<28)
- #define `XEN_DOMCTL_PFINFO_L2TAB` (0x2U<<28)
- #define `XEN_DOMCTL_PFINFO_L3TAB` (0x3U<<28)
- #define `XEN_DOMCTL_PFINFO_L4TAB` (0x4U<<28)
- #define `XEN_DOMCTL_PFINFO_LTABTYPE_MASK` (0x7U<<28)
- #define `XEN_DOMCTL_PFINFO_LPINTAB` (0x1U<<31)
- #define `XEN_DOMCTL_PFINFO_XTAB` (0xfU<<28) /\* invalid page \*/
- #define `XEN_DOMCTL_PFINFO_PAGETAB` (0x8U<<28)
- #define `XEN_DOMCTL_PFINFO_LTAB_MASK` (0xfU<<28)
- #define `XEN_DOMCTL_SHADOW_OP_OFF` 0
- #define `XEN_DOMCTL_SHADOW_OP_ENABLE` 32
- #define `XEN_DOMCTL_SHADOW_OP_CLEAN` 11
- #define `XEN_DOMCTL_SHADOW_OP_PEEK` 12
- #define `XEN_DOMCTL_SHADOW_OP_GET_ALLOCATION` 30
- #define `XEN_DOMCTL_SHADOW_OP_SET_ALLOCATION` 31
- #define `XEN_DOMCTL_SHADOW_OP_ENABLE_TEST` 1
- #define `XEN_DOMCTL_SHADOW_OP_ENABLE_LOGDIRTY` 2
- #define `XEN_DOMCTL_SHADOW_OP_ENABLE_TRANSLATE` 3
- #define `XEN_DOMCTL_SHADOW_ENABLE_REFCOUNT` (1 << 1)
- #define `XEN_DOMCTL_SHADOW_ENABLE_LOG_DIRTY` (1 << 2)
- #define `XEN_DOMCTL_SHADOW_ENABLE_TRANSLATE` (1 << 3)
- #define `XEN_DOMCTL_SHADOW_ENABLE_EXTERNAL` (1 << 4)
- #define `XEN_SCHEDULER_SEDF` 4
- #define `XEN_SCHEDULER_CREDIT` 5
- #define `XEN_SCHEDULER_CREDIT2` 6
- #define `XEN_SCHEDULER_ARINC653` 7
- #define `XEN_DOMCTL_SCHEDOP_putinfo` 0
- #define `XEN_DOMCTL_SCHEDOP_getinfo` 1
- #define `_XEN_DOMAINSETUP_hvm_guest` 0
- #define `XEN_DOMAINSETUP_hvm_guest` (1UL<<\_XEN\_DOMAINSETUP\_hvm\_guest)
- #define `_XEN_DOMAINSETUP_query` 1 /\* Get [parameters](#) (for save) \*/
- #define `XEN_DOMAINSETUP_query` (1UL<<\_XEN\_DOMAINSETUP\_query)
- #define `_XEN_DOMAINSETUP_sioemu_guest` 2
- #define `XEN_DOMAINSETUP_sioemu_guest` (1UL<<\_XEN\_DOMAINSETUP\_sioemu\_guest)
- #define `XEN_DOMCTL_SENDTRIGGER_NMI` 0
- #define `XEN_DOMCTL_SENDTRIGGER_RESET` 1
- #define `XEN_DOMCTL_SENDTRIGGER_INIT` 2
- #define `XEN_DOMCTL_SENDTRIGGER_POWER` 3
- #define `XEN_DOMCTL_SENDTRIGGER_SLEEP` 4
- #define `DPCI_ADD_MAPPING` 1
- #define `DPCI_REMOVE_MAPPING` 0
- #define `XEN_DOMCTL_MEM_CACHEATTR_UC` 0
- #define `XEN_DOMCTL_MEM_CACHEATTR_WC` 1
- #define `XEN_DOMCTL_MEM_CACHEATTR_WT` 4
- #define `XEN_DOMCTL_MEM_CACHEATTR_WP` 5

- #define [XEN\\_DOMCTL\\_MEM\\_CACHEATTR\\_WB](#) 6
- #define [XEN\\_DOMCTL\\_MEM\\_CACHEATTR\\_UCM](#) 7
- #define [XEN\\_DOMCTL\\_DEBUG\\_OP\\_SINGLE\\_STEP\\_OFF](#) 0
- #define [XEN\\_DOMCTL\\_DEBUG\\_OP\\_SINGLE\\_STEP\\_ON](#) 1
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_ENABLE](#) 0
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_DISABLE](#) 1
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_PAGING](#) 1
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_PAGING\\_NOMINATE](#) 0
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_PAGING\\_EVICT](#) 1
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_PAGING\\_PREP](#) 2
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_PAGING\\_RESUME](#) 3
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_ACCESS](#) 2
- #define [XEN\\_DOMCTL\\_MEM\\_EVENT\\_OP\\_ACCESS\\_RESUME](#) 0
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_CONTROL](#) 0
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_NOMINATE\\_GFN](#) 1
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_NOMINATE\\_GREF](#) 2
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_SHARE](#) 3
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_RESUME](#) 4
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_DEBUG\\_GFN](#) 5
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_DEBUG\\_MFN](#) 6
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_OP\\_DEBUG\\_GREF](#) 7
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_S\\_HANDLE\\_INVALID](#) (-10)
- #define [XEN\\_DOMCTL\\_MEM\\_SHARING\\_C\\_HANDLE\\_INVALID](#) (-9)
- #define [XEN\\_DOMCTL\\_createdomain](#) 1
- #define [XEN\\_DOMCTL\\_destroydomain](#) 2
- #define [XEN\\_DOMCTL\\_pausedomain](#) 3
- #define [XEN\\_DOMCTL\\_unpausedomain](#) 4
- #define [XEN\\_DOMCTL\\_getdomaininfo](#) 5
- #define [XEN\\_DOMCTL\\_getmemlist](#) 6
- #define [XEN\\_DOMCTL\\_getpageframeinfo](#) 7
- #define [XEN\\_DOMCTL\\_getpageframeinfo2](#) 8
- #define [XEN\\_DOMCTL\\_setvcpuaffinity](#) 9
- #define [XEN\\_DOMCTL\\_shadow\\_op](#) 10
- #define [XEN\\_DOMCTL\\_max\\_mem](#) 11
- #define [XEN\\_DOMCTL\\_setvcpucontext](#) 12
- #define [XEN\\_DOMCTL\\_getvcpucontext](#) 13
- #define [XEN\\_DOMCTL\\_getvcpuinfo](#) 14
- #define [XEN\\_DOMCTL\\_max\\_vcpus](#) 15
- #define [XEN\\_DOMCTL\\_scheduler\\_op](#) 16
- #define [XEN\\_DOMCTL\\_setdomainhandle](#) 17
- #define [XEN\\_DOMCTL\\_setdebugging](#) 18
- #define [XEN\\_DOMCTL\\_irq\\_permission](#) 19
- #define [XEN\\_DOMCTL\\_iomem\\_permission](#) 20
- #define [XEN\\_DOMCTL\\_ioport\\_permission](#) 21
- #define [XEN\\_DOMCTL\\_hypcall\\_init](#) 22
- #define [XEN\\_DOMCTL\\_arch\\_setup](#) 23
- #define [XEN\\_DOMCTL\\_settimeoffset](#) 24
- #define [XEN\\_DOMCTL\\_getvcpuaffinity](#) 25
- #define [XEN\\_DOMCTL\\_real\\_mode\\_area](#) 26
- #define [XEN\\_DOMCTL\\_resumedomain](#) 27
- #define [XEN\\_DOMCTL\\_sendtrigger](#) 28
- #define [XEN\\_DOMCTL\\_subscribe](#) 29
- #define [XEN\\_DOMCTL\\_gethvmcontext](#) 33
- #define [XEN\\_DOMCTL\\_sethvmcontext](#) 34
- #define [XEN\\_DOMCTL\\_set\\_address\\_size](#) 35

- `#define XEN_DOMCTL_get_address_size` 36
- `#define XEN_DOMCTL_assign_device` 37
- `#define XEN_DOMCTL_bind_pt_irq` 38
- `#define XEN_DOMCTL_memory_mapping` 39
- `#define XEN_DOMCTL_ioport_mapping` 40
- `#define XEN_DOMCTL_pin_mem_cacheattr` 41
- `#define XEN_DOMCTL_set_ext_vcpucontext` 42
- `#define XEN_DOMCTL_get_ext_vcpucontext` 43
- `#define XEN_DOMCTL_set_opt_feature` 44
- `#define XEN_DOMCTL_test_assign_device` 45
- `#define XEN_DOMCTL_set_target` 46
- `#define XEN_DOMCTL_deassign_device` 47
- `#define XEN_DOMCTL_unbind_pt_irq` 48
- `#define XEN_DOMCTL_set_cpuid` 49
- `#define XEN_DOMCTL_get_device_group` 50
- `#define XEN_DOMCTL_set_machine_address_size` 51
- `#define XEN_DOMCTL_get_machine_address_size` 52
- `#define XEN_DOMCTL_suppress_spurious_page_faults` 53
- `#define XEN_DOMCTL_debug_op` 54
- `#define XEN_DOMCTL_gethvmcontext_partial` 55
- `#define XEN_DOMCTL_mem_event_op` 56
- `#define XEN_DOMCTL_mem_sharing_op` 57
- `#define XEN_DOMCTL_disable_migrate` 58
- `#define XEN_DOMCTL_gettscinfo` 59
- `#define XEN_DOMCTL_settscinfo` 60
- `#define XEN_DOMCTL_getpageframeinfo3` 61
- `#define XEN_DOMCTL_setvcpuextstate` 62
- `#define XEN_DOMCTL_getvcpuextstate` 63
- `#define XEN_DOMCTL_set_access_required` 64
- `#define XEN_DOMCTL_gdbsx_guestmemio` 1000
- `#define XEN_DOMCTL_gdbsx_pausevcpu` 1001
- `#define XEN_DOMCTL_gdbsx_unpausevcpu` 1002
- `#define XEN_DOMCTL_gdbsx_domstatus` 1003

## Typedefs

- `typedef struct`  
`xen_domctl_createdomain xen_domctl_createdomain_t`
- `typedef struct`  
`xen_domctl_getdomaininfo xen_domctl_getdomaininfo_t`
- `typedef struct`  
`xen_domctl_getmemlist xen_domctl_getmemlist_t`
- `typedef struct`  
`xen_domctl_getpageframeinfo xen_domctl_getpageframeinfo_t`
- `typedef struct`  
`xen_domctl_getpageframeinfo2 xen_domctl_getpageframeinfo2_t`
- `typedef struct`  
`xen_domctl_shadow_op_stats xen_domctl_shadow_op_stats_t`
- `typedef struct xen_domctl_shadow_op xen_domctl_shadow_op_t`
- `typedef struct xen_domctl_max_mem xen_domctl_max_mem_t`
- `typedef struct`  
`xen_domctl_vcpucontext xen_domctl_vcpucontext_t`
- `typedef struct`  
`xen_domctl_getvcpuinfo xen_domctl_getvcpuinfo_t`

- typedef struct  
    [xen\\_domctl\\_vcpuaffinity](#) [xen\\_domctl\\_vcpuaffinity\\_t](#)
- typedef struct [xen\\_domctl\\_max\\_vcpus](#) [xen\\_domctl\\_max\\_vcpus\\_t](#)
- typedef struct  
    [xen\\_domctl\\_scheduler\\_op](#) [xen\\_domctl\\_scheduler\\_op\\_t](#)
- typedef struct  
    [xen\\_domctl\\_setdomainhandle](#) [xen\\_domctl\\_setdomainhandle\\_t](#)
- typedef struct  
    [xen\\_domctl\\_setdebugging](#) [xen\\_domctl\\_setdebugging\\_t](#)
- typedef struct  
    [xen\\_domctl\\_irq\\_permission](#) [xen\\_domctl\\_irq\\_permission\\_t](#)
- typedef struct  
    [xen\\_domctl\\_iomem\\_permission](#) [xen\\_domctl\\_iomem\\_permission\\_t](#)
- typedef struct  
    [xen\\_domctl\\_ioport\\_permission](#) [xen\\_domctl\\_ioport\\_permission\\_t](#)
- typedef struct  
    [xen\\_domctl\\_hypercall\\_init](#) [xen\\_domctl\\_hypercall\\_init\\_t](#)
- typedef struct  
    [xen\\_domctl\\_arch\\_setup](#) [xen\\_domctl\\_arch\\_setup\\_t](#)
- typedef struct  
    [xen\\_domctl\\_settimeoffset](#) [xen\\_domctl\\_settimeoffset\\_t](#)
- typedef struct  
    [xen\\_domctl\\_hvmcontext](#) [xen\\_domctl\\_hvmcontext\\_t](#)
- typedef struct  
    [xen\\_domctl\\_address\\_size](#) [xen\\_domctl\\_address\\_size\\_t](#)
- typedef struct  
    [xen\\_domctl\\_real\\_mode\\_area](#) [xen\\_domctl\\_real\\_mode\\_area\\_t](#)
- typedef struct  
    [xen\\_domctl\\_sendtrigger](#) [xen\\_domctl\\_sendtrigger\\_t](#)
- typedef struct  
    [xen\\_domctl\\_assign\\_device](#) [xen\\_domctl\\_assign\\_device\\_t](#)
- typedef struct  
    [xen\\_domctl\\_get\\_device\\_group](#) [xen\\_domctl\\_get\\_device\\_group\\_t](#)
- typedef enum [pt\\_irq\\_type\\_e](#) [pt\\_irq\\_type\\_t](#)
- typedef struct  
    [xen\\_domctl\\_bind\\_pt\\_irq](#) [xen\\_domctl\\_bind\\_pt\\_irq\\_t](#)
- typedef struct  
    [xen\\_domctl\\_memory\\_mapping](#) [xen\\_domctl\\_memory\\_mapping\\_t](#)
- typedef struct  
    [xen\\_domctl\\_ioport\\_mapping](#) [xen\\_domctl\\_ioport\\_mapping\\_t](#)
- typedef struct  
    [xen\\_domctl\\_pin\\_mem\\_cacheattr](#) [xen\\_domctl\\_pin\\_mem\\_cacheattr\\_t](#)
- typedef struct  
    [xen\\_domctl\\_ext\\_vcpucontext](#) [xen\\_domctl\\_ext\\_vcpucontext\\_t](#)
- typedef struct  
    [xen\\_domctl\\_set\\_opt\\_feature](#) [xen\\_domctl\\_set\\_opt\\_feature\\_t](#)
- typedef struct  
    [xen\\_domctl\\_set\\_target](#) [xen\\_domctl\\_set\\_target\\_t](#)
- typedef struct [xen\\_domctl\\_subscribe](#) [xen\\_domctl\\_subscribe\\_t](#)
- typedef struct [xen\\_domctl\\_debug\\_op](#) [xen\\_domctl\\_debug\\_op\\_t](#)
- typedef struct  
    [xen\\_domctl\\_hvmcontext\\_partial](#) [xen\\_domctl\\_hvmcontext\\_partial\\_t](#)
- typedef struct  
    [xen\\_domctl\\_disable\\_migrate](#) [xen\\_domctl\\_disable\\_migrate\\_t](#)
- typedef struct [xen\\_guest\\_tsc\\_info](#) [xen\\_guest\\_tsc\\_info\\_t](#)
- typedef struct [xen\\_domctl\\_tsc\\_info](#) [xen\\_domctl\\_tsc\\_info\\_t](#)

- typedef struct [xen\\_domctl\\_mem\\_event\\_op](#) [xen\\_domctl\\_mem\\_event\\_op\\_t](#)
- typedef struct [xen\\_domctl\\_mem\\_sharing\\_op](#) [xen\\_domctl\\_mem\\_sharing\\_op\\_t](#)
- typedef struct [xen\\_domctl\\_set\\_access\\_required](#) [xen\\_domctl\\_set\\_access\\_required\\_t](#)
- typedef struct [xen\\_domctl](#) [xen\\_domctl\\_t](#)

## Enumerations

- enum [pt\\_irq\\_type\\_e](#) { [PT\\_IRQ\\_TYPE\\_PCI](#), [PT\\_IRQ\\_TYPE\\_ISA](#), [PT\\_IRQ\\_TYPE\\_MSI](#), [PT\\_IRQ\\_TYPE\\_MSI-  
\\_TRANSLATE](#) }

## Functions

- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_createdomain\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_getdomaininfo\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_getmemlist\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_getpageframeinfo\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_getpageframeinfo2\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_shadow\\_op\\_stats\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_shadow\\_op\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_max\\_mem\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_vcpucontext\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_getvcpuinfo\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_vcpuaffinity\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_max\\_vcpus\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_scheduler\\_op\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_setdomainhandle\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_setdebugging\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_irq\\_permission\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_iomem\\_permission\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_ioport\\_permission\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_hypervisor\\_init\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_arch\\_setup\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_settimeoffset\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_hvmcontext\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_address\\_size\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_real\\_mode\\_area\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_sendtrigger\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_assign\\_device\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_get\\_device\\_group\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_bind\\_pt\\_irq\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_memory\\_mapping\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_ioport\\_mapping\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_pin\\_mem\\_cacheattr\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_ext\\_vcpucontext\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_set\\_opt\\_feature\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_set\\_target\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_subscribe\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_debug\\_op\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_hvmcontext\\_partial\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_guest\\_tsc\\_info\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_mem\\_event\\_op\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_mem\\_sharing\\_op\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_set\\_access\\_required\\_t](#))
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) ([xen\\_domctl\\_t](#))



### 4.181.1 Macro Definition Documentation

4.181.1.1 `#define _XEN_DOMAINSETUP_hvm_guest 0`

4.181.1.2 `#define _XEN_DOMAINSETUP_query 1 /* Get parameters (for save) */`

4.181.1.3 `#define _XEN_DOMAINSETUP_sioemu_guest 2`

4.181.1.4 `#define _XEN_DOMCTL_CDF_hap 1`

4.181.1.5 `#define _XEN_DOMCTL_CDF_hvm_guest 0`

4.181.1.6 `#define _XEN_DOMCTL_CDF_oos_off 3`

4.181.1.7 `#define _XEN_DOMCTL_CDF_priv 5`

4.181.1.8 `#define _XEN_DOMCTL_CDF_s3_integrity 2`

4.181.1.9 `#define _XEN_DOMINF_blocked 4`

4.181.1.10 `#define _XEN_DOMINF_debugged 6`

4.181.1.11 `#define _XEN_DOMINF_dying 0`

4.181.1.12 `#define _XEN_DOMINF_hvm_guest 1`

4.181.1.13 `#define _XEN_DOMINF_paused 3`

4.181.1.14 `#define _XEN_DOMINF_running 5`

4.181.1.15 `#define _XEN_DOMINF_shutdown 2`

4.181.1.16 `#define DPCI_ADD_MAPPING 1`

4.181.1.17 `#define DPCI_REMOVE_MAPPING 0`

4.181.1.18 `#define XEN_DOMAINSETUP_hvm_guest (1UL<<_XEN_DOMAINSETUP_hvm_guest)`

4.181.1.19 `#define XEN_DOMAINSETUP_query (1UL<<_XEN_DOMAINSETUP_query)`

4.181.1.20 `#define XEN_DOMAINSETUP_sioemu_guest (1UL<<_XEN_DOMAINSETUP_sioemu_guest)`

4.181.1.21 `#define XEN_DOMCTL_arch_setup 23`

4.181.1.22 `#define XEN_DOMCTL_assign_device 37`

4.181.1.23 `#define XEN_DOMCTL_bind_pt_irq 38`

4.181.1.24 `#define XEN_DOMCTL_CDF_hap (1U<<_XEN_DOMCTL_CDF_hap)`

4.181.1.25 `#define XEN_DOMCTL_CDF_hvm_guest (1U<<_XEN_DOMCTL_CDF_hvm_guest)`

4.181.1.26 `#define XEN_DOMCTL_CDF_oos_off (1U<<_XEN_DOMCTL_CDF_oos_off)`

4.181.1.27 `#define XEN_DOMCTL_CDF_priv (1U<<_XEN_DOMCTL_CDF_priv)`

4.181.1.28 `#define XEN_DOMCTL_CDF_s3_integrity (1U<<_XEN_DOMCTL_CDF_s3_integrity)`

4.181.1.29 `#define XEN_DOMCTL_createdomain 1`

4.181.1.30 `#define XEN_DOMCTL_deassign_device 47`

4.181.1.31 `#define XEN_DOMCTL_debug_op 54`

4.181.1.32 `#define XEN_DOMCTL_DEBUG_OP_SINGLE_STEP_OFF 0`

4.181.1.33 `#define XEN_DOMCTL_DEBUG_OP_SINGLE_STEP_ON 1`

4.181.1.34 `#define XEN_DOMCTL_destroydomain 2`

4.181.1.35 `#define XEN_DOMCTL_disable_migrate 58`

4.181.1.36 `#define XEN_DOMCTL_gdbsx_domstatus 1003`

4.181.1.37 `#define XEN_DOMCTL_gdbsx_guestmemio 1000`

4.181.1.38 `#define XEN_DOMCTL_gdbsx_pausevcpu 1001`

4.181.1.39 `#define XEN_DOMCTL_gdbsx_unpausevcpu 1002`

4.181.1.40 `#define XEN_DOMCTL_get_address_size 36`

4.181.1.41 `#define XEN_DOMCTL_get_device_group 50`

4.181.1.42 `#define XEN_DOMCTL_get_ext_vcpucontext 43`

4.181.1.43 `#define XEN_DOMCTL_get_machine_address_size 52`

4.181.1.44 `#define XEN_DOMCTL_getdomaininfo 5`

4.181.1.45 `#define XEN_DOMCTL_gethvmcontext 33`

4.181.1.46 `#define XEN_DOMCTL_gethvmcontext_partial 55`

4.181.1.47 `#define XEN_DOMCTL_getmemlist 6`

4.181.1.48 `#define XEN_DOMCTL_getpageframeinfo 7`

4.181.1.49 `#define XEN_DOMCTL_getpageframeinfo2 8`

4.181.1.50 `#define XEN_DOMCTL_getpageframeinfo3 61`

4.181.1.51 `#define XEN_DOMCTL_gettscinfo 59`

4.181.1.52 `#define XEN_DOMCTL_getvcpuaffinity 25`

4.181.1.53 `#define XEN_DOMCTL_getvcpucontext 13`

4.181.1.54 `#define XEN_DOMCTL_getvcpuextstate 63`

4.181.1.55 `#define XEN_DOMCTL_getvcpuinfo 14`

- 4.181.1.56 `#define XEN_DOMCTL_hypercall_init 22`
- 4.181.1.57 `#define XEN_DOMCTL_INTERFACE_VERSION 0x00000007`
- 4.181.1.58 `#define XEN_DOMCTL_iomem_permission 20`
- 4.181.1.59 `#define XEN_DOMCTL_ioport_mapping 40`
- 4.181.1.60 `#define XEN_DOMCTL_ioport_permission 21`
- 4.181.1.61 `#define XEN_DOMCTL_irq_permission 19`
- 4.181.1.62 `#define XEN_DOMCTL_max_mem 11`
- 4.181.1.63 `#define XEN_DOMCTL_max_vcpus 15`
- 4.181.1.64 `#define XEN_DOMCTL_MEM_CACHEATTR_UC 0`
- 4.181.1.65 `#define XEN_DOMCTL_MEM_CACHEATTR_UCM 7`
- 4.181.1.66 `#define XEN_DOMCTL_MEM_CACHEATTR_WB 6`
- 4.181.1.67 `#define XEN_DOMCTL_MEM_CACHEATTR_WC 1`
- 4.181.1.68 `#define XEN_DOMCTL_MEM_CACHEATTR_WP 5`
- 4.181.1.69 `#define XEN_DOMCTL_MEM_CACHEATTR_WT 4`
- 4.181.1.70 `#define XEN_DOMCTL_mem_event_op 56`
- 4.181.1.71 `#define XEN_DOMCTL_MEM_EVENT_OP_ACCESS 2`
- 4.181.1.72 `#define XEN_DOMCTL_MEM_EVENT_OP_ACCESS_RESUME 0`
- 4.181.1.73 `#define XEN_DOMCTL_MEM_EVENT_OP_DISABLE 1`
- 4.181.1.74 `#define XEN_DOMCTL_MEM_EVENT_OP_ENABLE 0`
- 4.181.1.75 `#define XEN_DOMCTL_MEM_EVENT_OP_PAGING 1`
- 4.181.1.76 `#define XEN_DOMCTL_MEM_EVENT_OP_PAGING_EVICT 1`
- 4.181.1.77 `#define XEN_DOMCTL_MEM_EVENT_OP_PAGING_NOMINATE 0`
- 4.181.1.78 `#define XEN_DOMCTL_MEM_EVENT_OP_PAGING_PREP 2`
- 4.181.1.79 `#define XEN_DOMCTL_MEM_EVENT_OP_PAGING_RESUME 3`
- 4.181.1.80 `#define XEN_DOMCTL_MEM_SHARING_C_HANDLE_INVALID (-9)`
- 4.181.1.81 `#define XEN_DOMCTL_mem_sharing_op 57`
- 4.181.1.82 `#define XEN_DOMCTL_MEM_SHARING_OP_CONTROL 0`
- 4.181.1.83 `#define XEN_DOMCTL_MEM_SHARING_OP_DEBUG_GFN 5`

4.181.1.84 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_DEBUG\_GREF 7

4.181.1.85 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_DEBUG\_MFN 6

4.181.1.86 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_NOMINATE\_GFN 1

4.181.1.87 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_NOMINATE\_GREF 2

4.181.1.88 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_RESUME 4

4.181.1.89 #define XEN\_DOMCTL\_MEM\_SHARING\_OP\_SHARE 3

4.181.1.90 #define XEN\_DOMCTL\_MEM\_SHARING\_S\_HANDLE\_INVALID (-10)

4.181.1.91 #define XEN\_DOMCTL\_memory\_mapping 39

4.181.1.92 #define XEN\_DOMCTL\_pausedomain 3

4.181.1.93 #define XEN\_DOMCTL\_PFINFO\_L1TAB (0x1U<<28)

4.181.1.94 #define XEN\_DOMCTL\_PFINFO\_L2TAB (0x2U<<28)

4.181.1.95 #define XEN\_DOMCTL\_PFINFO\_L3TAB (0x3U<<28)

4.181.1.96 #define XEN\_DOMCTL\_PFINFO\_L4TAB (0x4U<<28)

4.181.1.97 #define XEN\_DOMCTL\_PFINFO\_LPINTAB (0x1U<<31)

4.181.1.98 #define XEN\_DOMCTL\_PFINFO\_LTAB\_MASK (0xfU<<28)

4.181.1.99 #define XEN\_DOMCTL\_PFINFO\_LTAB\_SHIFT 28

4.181.1.100 #define XEN\_DOMCTL\_PFINFO\_LTABTYPE\_MASK (0x7U<<28)

4.181.1.101 #define XEN\_DOMCTL\_PFINFO\_NOTAB (0x0U<<28)

4.181.1.102 #define XEN\_DOMCTL\_PFINFO\_PAGEDTAB (0x8U<<28)

4.181.1.103 #define XEN\_DOMCTL\_PFINFO\_XTAB (0xfU<<28) /\* invalid page \*/

4.181.1.104 #define XEN\_DOMCTL\_pin\_mem\_cacheattr 41

4.181.1.105 #define XEN\_DOMCTL\_real\_mode\_area 26

4.181.1.106 #define XEN\_DOMCTL\_resumedomain 27

4.181.1.107 #define XEN\_DOMCTL\_SCHEDOP\_getinfo 1

4.181.1.108 #define XEN\_DOMCTL\_SCHEDOP\_putinfo 0

4.181.1.109 #define XEN\_DOMCTL\_scheduler\_op 16

4.181.1.110 #define XEN\_DOMCTL\_sendtrigger 28

4.181.1.111 #define XEN\_DOMCTL\_SENDTRIGGER\_INIT 2

- 4.181.1.112 `#define XEN_DOMCTL_SENDTRIGGER_NMI 0`
- 4.181.1.113 `#define XEN_DOMCTL_SENDTRIGGER_POWER 3`
- 4.181.1.114 `#define XEN_DOMCTL_SENDTRIGGER_RESET 1`
- 4.181.1.115 `#define XEN_DOMCTL_SENDTRIGGER_SLEEP 4`
- 4.181.1.116 `#define XEN_DOMCTL_set_access_required 64`
- 4.181.1.117 `#define XEN_DOMCTL_set_address_size 35`
- 4.181.1.118 `#define XEN_DOMCTL_set_cpuid 49`
- 4.181.1.119 `#define XEN_DOMCTL_set_ext_vcpucontext 42`
- 4.181.1.120 `#define XEN_DOMCTL_set_machine_address_size 51`
- 4.181.1.121 `#define XEN_DOMCTL_set_opt_feature 44`
- 4.181.1.122 `#define XEN_DOMCTL_set_target 46`
- 4.181.1.123 `#define XEN_DOMCTL_setdebugging 18`
- 4.181.1.124 `#define XEN_DOMCTL_setdomainhandle 17`
- 4.181.1.125 `#define XEN_DOMCTL_sethvmcontext 34`
- 4.181.1.126 `#define XEN_DOMCTL_settimeoffset 24`
- 4.181.1.127 `#define XEN_DOMCTL_settscinfo 60`
- 4.181.1.128 `#define XEN_DOMCTL_setvcpuaffinity 9`
- 4.181.1.129 `#define XEN_DOMCTL_setvcpucontext 12`
- 4.181.1.130 `#define XEN_DOMCTL_setvcpuextstate 62`
- 4.181.1.131 `#define XEN_DOMCTL_SHADOW_ENABLE_EXTERNAL (1 << 4)`
- 4.181.1.132 `#define XEN_DOMCTL_SHADOW_ENABLE_LOG_DIRTY (1 << 2)`
- 4.181.1.133 `#define XEN_DOMCTL_SHADOW_ENABLE_REFCOUNT (1 << 1)`
- 4.181.1.134 `#define XEN_DOMCTL_SHADOW_ENABLE_TRANSLATE (1 << 3)`
- 4.181.1.135 `#define XEN_DOMCTL_shadow_op 10`
- 4.181.1.136 `#define XEN_DOMCTL_SHADOW_OP_CLEAN 11`
- 4.181.1.137 `#define XEN_DOMCTL_SHADOW_OP_ENABLE 32`
- 4.181.1.138 `#define XEN_DOMCTL_SHADOW_OP_ENABLE_LOGDIRTY 2`
- 4.181.1.139 `#define XEN_DOMCTL_SHADOW_OP_ENABLE_TEST 1`

```

4.181.1.140  #define XEN_DOMCTL_SHADOW_OP_ENABLE_TRANSLATE 3
4.181.1.141  #define XEN_DOMCTL_SHADOW_OP_GET_ALLOCATION 30
4.181.1.142  #define XEN_DOMCTL_SHADOW_OP_OFF 0
4.181.1.143  #define XEN_DOMCTL_SHADOW_OP_PEEK 12
4.181.1.144  #define XEN_DOMCTL_SHADOW_OP_SET_ALLOCATION 31
4.181.1.145  #define XEN_DOMCTL_subscribe 29
4.181.1.146  #define XEN_DOMCTL_suppress_spurious_page_faults 53
4.181.1.147  #define XEN_DOMCTL_test_assign_device 45
4.181.1.148  #define XEN_DOMCTL_unbind_pt_irq 48
4.181.1.149  #define XEN_DOMCTL_unpauseddomain 4
4.181.1.150  #define XEN_DOMINF_blocked (1U<<_XEN_DOMINF_blocked)
4.181.1.151  #define XEN_DOMINF_debugged (1U<<_XEN_DOMINF_debugged)
4.181.1.152  #define XEN_DOMINF_dying (1U<<_XEN_DOMINF_dying)
4.181.1.153  #define XEN_DOMINF_hvm_guest (1U<<_XEN_DOMINF_hvm_guest)
4.181.1.154  #define XEN_DOMINF_paused (1U<<_XEN_DOMINF_paused)
4.181.1.155  #define XEN_DOMINF_running (1U<<_XEN_DOMINF_running)
4.181.1.156  #define XEN_DOMINF_shutdown (1U<<_XEN_DOMINF_shutdown)
4.181.1.157  #define XEN_DOMINF_shutdownmask 255
4.181.1.158  #define XEN_DOMINF_shutdownshift 16
4.181.1.159  #define XEN_SCHEDULER_ARINC653 7
4.181.1.160  #define XEN_SCHEDULER_CREDIT 5
4.181.1.161  #define XEN_SCHEDULER_CREDIT2 6
4.181.1.162  #define XEN_SCHEDULER_SEDF 4

```

## 4.181.2 Typedef Documentation

```

4.181.2.1  typedef enum pt_irq_type_e pt_irq_type_t
4.181.2.2  typedef struct xen_domctl_address_size xen_domctl_address_size_t
4.181.2.3  typedef struct xen_domctl_arch_setup xen_domctl_arch_setup_t
4.181.2.4  typedef struct xen_domctl_assign_device xen_domctl_assign_device_t

```

- 4.181.2.5 typedef struct xen\_domctl\_bind\_pt\_irq xen\_domctl\_bind\_pt\_irq\_t
- 4.181.2.6 typedef struct xen\_domctl\_createdomain xen\_domctl\_createdomain\_t
- 4.181.2.7 typedef struct xen\_domctl\_debug\_op xen\_domctl\_debug\_op\_t
- 4.181.2.8 typedef struct xen\_domctl\_disable\_migrate xen\_domctl\_disable\_migrate\_t
- 4.181.2.9 typedef struct xen\_domctl\_ext\_vcpucontext xen\_domctl\_ext\_vcpucontext\_t
- 4.181.2.10 typedef struct xen\_domctl\_get\_device\_group xen\_domctl\_get\_device\_group\_t
- 4.181.2.11 typedef struct xen\_domctl\_getdomaininfo xen\_domctl\_getdomaininfo\_t
- 4.181.2.12 typedef struct xen\_domctl\_getmemlist xen\_domctl\_getmemlist\_t
- 4.181.2.13 typedef struct xen\_domctl\_getpageframeinfo2 xen\_domctl\_getpageframeinfo2\_t
- 4.181.2.14 typedef struct xen\_domctl\_getpageframeinfo xen\_domctl\_getpageframeinfo\_t
- 4.181.2.15 typedef struct xen\_domctl\_getvcpuinfo xen\_domctl\_getvcpuinfo\_t
- 4.181.2.16 typedef struct xen\_domctl\_hvmcontext\_partial xen\_domctl\_hvmcontext\_partial\_t
- 4.181.2.17 typedef struct xen\_domctl\_hvmcontext xen\_domctl\_hvmcontext\_t
- 4.181.2.18 typedef struct xen\_domctl\_hypercall\_init xen\_domctl\_hypercall\_init\_t
- 4.181.2.19 typedef struct xen\_domctl\_iomem\_permission xen\_domctl\_iomem\_permission\_t
- 4.181.2.20 typedef struct xen\_domctl\_ioport\_mapping xen\_domctl\_ioport\_mapping\_t
- 4.181.2.21 typedef struct xen\_domctl\_ioport\_permission xen\_domctl\_ioport\_permission\_t
- 4.181.2.22 typedef struct xen\_domctl\_irq\_permission xen\_domctl\_irq\_permission\_t
- 4.181.2.23 typedef struct xen\_domctl\_max\_mem xen\_domctl\_max\_mem\_t
- 4.181.2.24 typedef struct xen\_domctl\_max\_vcpus xen\_domctl\_max\_vcpus\_t
- 4.181.2.25 typedef struct xen\_domctl\_mem\_event\_op xen\_domctl\_mem\_event\_op\_t
- 4.181.2.26 typedef struct xen\_domctl\_mem\_sharing\_op xen\_domctl\_mem\_sharing\_op\_t
- 4.181.2.27 typedef struct xen\_domctl\_memory\_mapping xen\_domctl\_memory\_mapping\_t
- 4.181.2.28 typedef struct xen\_domctl\_pin\_mem\_cacheattr xen\_domctl\_pin\_mem\_cacheattr\_t
- 4.181.2.29 typedef struct xen\_domctl\_real\_mode\_area xen\_domctl\_real\_mode\_area\_t
- 4.181.2.30 typedef struct xen\_domctl\_scheduler\_op xen\_domctl\_scheduler\_op\_t
- 4.181.2.31 typedef struct xen\_domctl\_sendtrigger xen\_domctl\_sendtrigger\_t
- 4.181.2.32 typedef struct xen\_domctl\_set\_access\_required xen\_domctl\_set\_access\_required\_t

- 4.181.2.33 `typedef struct xen_domctl_set_opt_feature xen_domctl_set_opt_feature_t`
- 4.181.2.34 `typedef struct xen_domctl_set_target xen_domctl_set_target_t`
- 4.181.2.35 `typedef struct xen_domctl_setdebugging xen_domctl_setdebugging_t`
- 4.181.2.36 `typedef struct xen_domctl_setdomainhandle xen_domctl_setdomainhandle_t`
- 4.181.2.37 `typedef struct xen_domctl_settimeoffset xen_domctl_settimeoffset_t`
- 4.181.2.38 `typedef struct xen_domctl_shadow_op_stats xen_domctl_shadow_op_stats_t`
- 4.181.2.39 `typedef struct xen_domctl_shadow_op xen_domctl_shadow_op_t`
- 4.181.2.40 `typedef struct xen_domctl_subscribe xen_domctl_subscribe_t`
- 4.181.2.41 `typedef struct xen_domctl xen_domctl_t`
- 4.181.2.42 `typedef struct xen_domctl_tsc_info xen_domctl_tsc_info_t`
- 4.181.2.43 `typedef struct xen_domctl_vcpuaffinity xen_domctl_vcpuaffinity_t`
- 4.181.2.44 `typedef struct xen_domctl_vcpucontext xen_domctl_vcpucontext_t`
- 4.181.2.45 `typedef struct xen_guest_tsc_info xen_guest_tsc_info_t`

### 4.181.3 Enumeration Type Documentation

- 4.181.3.1 `enum pt_irq_type_e`

Enumerator

***PT\_IRQ\_TYPE\_PCI***  
***PT\_IRQ\_TYPE\_ISA***  
***PT\_IRQ\_TYPE\_MSI***  
***PT\_IRQ\_TYPE\_MSI\_TRANSLATE***

### 4.181.4 Function Documentation

- 4.181.4.1 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_createdomain_t )`
- 4.181.4.2 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_getdomaininfo_t )`
- 4.181.4.3 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_getmemlist_t )`
- 4.181.4.4 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_getpageframeinfo_t )`
- 4.181.4.5 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_getpageframeinfo2_t )`
- 4.181.4.6 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_shadow_op_stats_t )`
- 4.181.4.7 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_shadow_op_t )`
- 4.181.4.8 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_max_mem_t )`



- 4.181.4.9    `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_vcpucontext_t )`
- 4.181.4.10   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_getvcpuinfo_t )`
- 4.181.4.11   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_vcpuaffinity_t )`
- 4.181.4.12   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_max_vcpus_t )`
- 4.181.4.13   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_scheduler_op_t )`
- 4.181.4.14   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_setdomainhandle_t )`
- 4.181.4.15   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_setdebugging_t )`
- 4.181.4.16   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_irq_permission_t )`
- 4.181.4.17   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_iomem_permission_t )`
- 4.181.4.18   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_ioport_permission_t )`
- 4.181.4.19   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_hypercall_init_t )`
- 4.181.4.20   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_arch_setup_t )`
- 4.181.4.21   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_settimeoffset_t )`
- 4.181.4.22   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_hvmcontext_t )`
- 4.181.4.23   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_address_size_t )`
- 4.181.4.24   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_real_mode_area_t )`
- 4.181.4.25   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_sendtrigger_t )`
- 4.181.4.26   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_assign_device_t )`
- 4.181.4.27   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_get_device_group_t )`
- 4.181.4.28   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_bind_pt_irq_t )`
- 4.181.4.29   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_memory_mapping_t )`
- 4.181.4.30   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_ioport_mapping_t )`
- 4.181.4.31   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_pin_mem_cacheattr_t )`
- 4.181.4.32   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_ext_vcpucontext_t )`
- 4.181.4.33   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_set_opt_feature_t )`
- 4.181.4.34   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_set_target_t )`
- 4.181.4.35   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_subscribe_t )`
- 4.181.4.36   `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_debug_op_t )`

4.181.4.37 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_hvmcontext_partial_t )`

4.181.4.38 `DEFINE_XEN_GUEST_HANDLE ( xen_guest_tsc_info_t )`

4.181.4.39 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_mem_event_op_t )`

4.181.4.40 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_mem_sharing_op_t )`

4.181.4.41 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_set_access_required_t )`

4.181.4.42 `DEFINE_XEN_GUEST_HANDLE ( xen_domctl_t )`

## 4.182 xen/xen/include/public/xen.h File Reference

```
#include "xen-compat.h"
```

### Data Structures

- struct [mmuext\\_op](#)
- struct [mmu\\_update](#)
- struct [multicall\\_entry](#)
- struct [vcpu\\_time\\_info](#)
- struct [vcpu\\_info](#)
- struct [shared\\_info](#)
- struct [start\\_info](#)
- struct [xen\\_multiboot\\_mod\\_list](#)
- struct [dom0\\_vga\\_console\\_info](#)

### Macros

- `#define __HYPERVISOR_set_trap_table 0`
- `#define __HYPERVISOR_mmu_update 1`
- `#define __HYPERVISOR_set_gdt 2`
- `#define __HYPERVISOR_stack_switch 3`
- `#define __HYPERVISOR_set_callbacks 4`
- `#define __HYPERVISOR_fpu_taskswitch 5`
- `#define __HYPERVISOR_sched_op_compat 6 /* compat since 0x00030101 */`
- `#define __HYPERVISOR_platform_op 7`
- `#define __HYPERVISOR_set_debugreg 8`
- `#define __HYPERVISOR_get_debugreg 9`
- `#define __HYPERVISOR_update_descriptor 10`
- `#define __HYPERVISOR_memory_op 12`
- `#define __HYPERVISOR_multicall 13`
- `#define __HYPERVISOR_update_va_mapping 14`
- `#define __HYPERVISOR_set_timer_op 15`
- `#define __HYPERVISOR_event_channel_op_compat 16 /* compat since 0x00030202 */`
- `#define __HYPERVISOR_xen_version 17`
- `#define __HYPERVISOR_console_io 18`
- `#define __HYPERVISOR_physdev_op_compat 19 /* compat since 0x00030202 */`
- `#define __HYPERVISOR_grant_table_op 20`
- `#define __HYPERVISOR_vm_assist 21`
- `#define __HYPERVISOR_update_va_mapping_otherdomain 22`

- #define `__HYPERVISOR_iret` 23 /\* x86 only \*/
- #define `__HYPERVISOR_vcpu_op` 24
- #define `__HYPERVISOR_set_segment_base` 25 /\* x86/64 only \*/
- #define `__HYPERVISOR_mmuext_op` 26
- #define `__HYPERVISOR_xsm_op` 27
- #define `__HYPERVISOR_nmi_op` 28
- #define `__HYPERVISOR_sched_op` 29
- #define `__HYPERVISOR_callback_op` 30
- #define `__HYPERVISOR_xenoprof_op` 31
- #define `__HYPERVISOR_event_channel_op` 32
- #define `__HYPERVISOR_physdev_op` 33
- #define `__HYPERVISOR_hvm_op` 34
- #define `__HYPERVISOR_sysctl` 35
- #define `__HYPERVISOR_domctl` 36
- #define `__HYPERVISOR_kexec_op` 37
- #define `__HYPERVISOR_tmem_op` 38
- #define `__HYPERVISOR_arch_0` 48
- #define `__HYPERVISOR_arch_1` 49
- #define `__HYPERVISOR_arch_2` 50
- #define `__HYPERVISOR_arch_3` 51
- #define `__HYPERVISOR_arch_4` 52
- #define `__HYPERVISOR_arch_5` 53
- #define `__HYPERVISOR_arch_6` 54
- #define `__HYPERVISOR_arch_7` 55
- #define `__HYPERVISOR_sched_op __HYPERVISOR_sched_op_compat`
- #define `__HYPERVISOR_event_channel_op __HYPERVISOR_event_channel_op_compat`
- #define `__HYPERVISOR_physdev_op __HYPERVISOR_physdev_op_compat`
- #define `__HYPERVISOR_dom0_op __HYPERVISOR_platform_op`
- #define `VIRQ_TIMER` 0 /\* V. Timebase update, and/or requested timeout. \*/
- #define `VIRQ_DEBUG` 1 /\* V. Request guest to dump debug info. \*/
- #define `VIRQ_CONSOLE` 2 /\* G. (DOM0) Bytes received on emergency console. \*/
- #define `VIRQ_DOM_EXC` 3 /\* G. (DOM0) Exceptional event for some domain. \*/
- #define `VIRQ_TBUF` 4 /\* G. (DOM0) Trace buffer has records available. \*/
- #define `VIRQ_DEBUGGER` 6 /\* G. (DOM0) A domain has paused for debugging. \*/
- #define `VIRQ_XENOPROF` 7 /\* V. XenOprofile interrupt: new sample available \*/
- #define `VIRQ_CON_RING` 8 /\* G. (DOM0) Bytes received on console \*/
- #define `VIRQ_PCPU_STATE` 9 /\* G. (DOM0) PCPU state changed \*/
- #define `VIRQ_MEM_EVENT` 10 /\* G. (DOM0) A memory event has occurred \*/
- #define `VIRQ_ARCH_0` 16
- #define `VIRQ_ARCH_1` 17
- #define `VIRQ_ARCH_2` 18
- #define `VIRQ_ARCH_3` 19
- #define `VIRQ_ARCH_4` 20
- #define `VIRQ_ARCH_5` 21
- #define `VIRQ_ARCH_6` 22
- #define `VIRQ_ARCH_7` 23
- #define `NR_VIRQS` 24
- #define `MMU_NORMAL_PT_UPDATE` 0 /\* checked '\*ptr = val'. ptr is MA. \*/
- #define `MMU_MACHPHYS_UPDATE` 1 /\* ptr = MA of frame to modify entry for \*/
- #define `MMU_PT_UPDATE_PRESERVE_AD` 2 /\* atomically: \*ptr = val | (\*ptr & (A|D)) \*/
- #define `MMUEXT_PIN_L1_TABLE` 0
- #define `MMUEXT_PIN_L2_TABLE` 1
- #define `MMUEXT_PIN_L3_TABLE` 2
- #define `MMUEXT_PIN_L4_TABLE` 3
- #define `MMUEXT_UNPIN_TABLE` 4

- #define [MMUEXT\\_NEW\\_BASEPTR](#) 5
- #define [MMUEXT\\_TLB\\_FLUSH\\_LOCAL](#) 6
- #define [MMUEXT\\_INVLPG\\_LOCAL](#) 7
- #define [MMUEXT\\_TLB\\_FLUSH\\_MULTI](#) 8
- #define [MMUEXT\\_INVLPG\\_MULTI](#) 9
- #define [MMUEXT\\_TLB\\_FLUSH\\_ALL](#) 10
- #define [MMUEXT\\_INVLPG\\_ALL](#) 11
- #define [MMUEXT\\_FLUSH\\_CACHE](#) 12
- #define [MMUEXT\\_SET\\_LDT](#) 13
- #define [MMUEXT\\_NEW\\_USER\\_BASEPTR](#) 15
- #define [MMUEXT\\_CLEAR\\_PAGE](#) 16
- #define [MMUEXT\\_COPY\\_PAGE](#) 17
- #define [MMUEXT\\_FLUSH\\_CACHE\\_GLOBAL](#) 18
- #define [MMUEXT\\_MARK\\_SUPER](#) 19
- #define [MMUEXT\\_UNMARK\\_SUPER](#) 20
- #define [UVMF\\_NONE](#) (0UL<<0) /\* No flushing at all. \*/
- #define [UVMF\\_TLB\\_FLUSH](#) (1UL<<0) /\* Flush entire TLB(s). \*/
- #define [UVMF\\_INVLPG](#) (2UL<<0) /\* Flush only one entry. \*/
- #define [UVMF\\_FLUSHTYPE\\_MASK](#) (3UL<<0)
- #define [UVMF\\_MULTI](#) (0UL<<2) /\* Flush subset of TLBs. \*/
- #define [UVMF\\_LOCAL](#) (0UL<<2) /\* Flush local TLB. \*/
- #define [UVMF\\_ALL](#) (1UL<<2) /\* Flush all TLBs. \*/
- #define [CONSOLEIO\\_write](#) 0
- #define [CONSOLEIO\\_read](#) 1
- #define [VMASST\\_CMD\\_enable](#) 0
- #define [VMASST\\_CMD\\_disable](#) 1
- #define [VMASST\\_TYPE\\_4gb\\_segments](#) 0
- #define [VMASST\\_TYPE\\_4gb\\_segments\\_notify](#) 1
- #define [VMASST\\_TYPE\\_writable\\_pagetables](#) 2
- #define [VMASST\\_TYPE\\_pae\\_extended\\_cr3](#) 3
- #define [MAX\\_VMASST\\_TYPE](#) 3
- #define [DOMID\\_FIRST\\_RESERVED](#) (0x7FF0U)
- #define [DOMID\\_SELF](#) (0x7FF0U)
- #define [DOMID\\_IO](#) (0x7FF1U)
- #define [DOMID\\_XEN](#) (0x7FF2U)
- #define [DOMID\\_COW](#) (0x7FF3U)
- #define [DOMID\\_INVALID](#) (0x7FF4U)
- #define [DOMID\\_IDLE](#) (0x7FFU)
- #define [NR\\_EVENT\\_CHANNELS](#) (sizeof(unsigned long) \* sizeof(unsigned long) \* 64)
- #define [MAX\\_GUEST\\_CMDLINE](#) 1024
- #define [console\\_mfn](#) console.domU.mfn
- #define [console\\_evtchn](#) console.domU.evtchn
- #define [SIF\\_PRIVILEGED](#) (1<<0) /\* Is the [domain](#) privileged? \*/
- #define [SIF\\_INITDOMAIN](#) (1<<1) /\* Is this the initial control [domain](#)? \*/
- #define [SIF\\_MULTIBOOT\\_MOD](#) (1<<2) /\* Is mod\_start a multiboot module? \*/
- #define [SIF\\_MOD\\_START\\_PFN](#) (1<<3) /\* Is mod\_start a PFN? \*/
- #define [SIF\\_PM\\_MASK](#) (0xFF<<8) /\* reserve 1 byte for xen-pm options \*/
- #define [XEN\\_VGATYPE\\_TEXT\\_MODE\\_3](#) 0x03
- #define [XEN\\_VGATYPE\\_VESA\\_LFB](#) 0x23
- #define [xen\\_vga\\_console\\_info](#) dom0\_vga\_console\_info
- #define [xen\\_vga\\_console\\_info\\_t](#) dom0\_vga\_console\_info\_t
- #define [\\_\\_mk\\_unsigned\\_long](#)(x) x ## UL
- #define [mk\\_unsigned\\_long](#)(x) \_\_mk\_unsigned\_long(x)

## Typedefs

- typedef struct [mmuext\\_op](#) [mmuext\\_op\\_t](#)
- typedef uint16\_t [domid\\_t](#)
- typedef struct [mmu\\_update](#) [mmu\\_update\\_t](#)
- typedef struct [multicall\\_entry](#) [multicall\\_entry\\_t](#)
- typedef struct [vcpu\\_time\\_info](#) [vcpu\\_time\\_info\\_t](#)
- typedef struct [vcpu\\_info](#) [vcpu\\_info\\_t](#)
- typedef struct [shared\\_info](#) [shared\\_info\\_t](#)
- typedef struct [start\\_info](#) [start\\_info\\_t](#)
- typedef struct [dom0\\_vga\\_console\\_info](#) [dom0\\_vga\\_console\\_info\\_t](#)
- typedef uint8\_t [xen\\_domain\\_handle\\_t](#) [16]

## Functions

- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (char)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uchar, unsigned char)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (int)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint, unsigned int)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (long)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (ulong, unsigned long)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (void)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint64\_t)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (xen\_pfn\_t)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (mmuext\_op\_t)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (mmu\_update\_t)
- [DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (multicall\_entry\_t)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint8, uint8\_t)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint16, uint16\_t)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint32, uint32\_t)
- [\\_\\_DEFINE\\_XEN\\_GUEST\\_HANDLE](#) (uint64, uint64\_t)

### 4.182.1 Macro Definition Documentation

4.182.1.1 `#define __HYPERVISOR_arch_0` 48

4.182.1.2 `#define __HYPERVISOR_arch_1` 49

4.182.1.3 `#define __HYPERVISOR_arch_2` 50

4.182.1.4 `#define __HYPERVISOR_arch_3` 51

4.182.1.5 `#define __HYPERVISOR_arch_4` 52

4.182.1.6 `#define __HYPERVISOR_arch_5` 53

4.182.1.7 `#define __HYPERVISOR_arch_6` 54

4.182.1.8 `#define __HYPERVISOR_arch_7` 55

4.182.1.9 `#define __HYPERVISOR_callback_op` 30

4.182.1.10 `#define __HYPERVISOR_console_io` 18

4.182.1.11 `#define __HYPERVISOR_dom0_op __HYPERVISOR_platform_op`

4.182.1.12 `#define __HYPERVISOR_domctl 36`

4.182.1.13 `#define __HYPERVISOR_event_channel_op 32`

4.182.1.14 `#define __HYPERVISOR_event_channel_op __HYPERVISOR_event_channel_op_compat`

4.182.1.15 `#define __HYPERVISOR_event_channel_op_compat 16 /* compat since 0x00030202 */`

4.182.1.16 `#define __HYPERVISOR_fpu_taskswitch 5`

4.182.1.17 `#define __HYPERVISOR_get_debugreg 9`

4.182.1.18 `#define __HYPERVISOR_grant_table_op 20`

4.182.1.19 `#define __HYPERVISOR_hvm_op 34`

4.182.1.20 `#define __HYPERVISOR_iret 23 /* x86 only */`

4.182.1.21 `#define __HYPERVISOR_kexec_op 37`

4.182.1.22 `#define __HYPERVISOR_memory_op 12`

4.182.1.23 `#define __HYPERVISOR_mmu_update 1`

4.182.1.24 `#define __HYPERVISOR_mmuext_op 26`

4.182.1.25 `#define __HYPERVISOR_multicall 13`

4.182.1.26 `#define __HYPERVISOR_nmi_op 28`

4.182.1.27 `#define __HYPERVISOR_physdev_op 33`

4.182.1.28 `#define __HYPERVISOR_physdev_op __HYPERVISOR_physdev_op_compat`

4.182.1.29 `#define __HYPERVISOR_physdev_op_compat 19 /* compat since 0x00030202 */`

4.182.1.30 `#define __HYPERVISOR_platform_op 7`

4.182.1.31 `#define __HYPERVISOR_sched_op 29`

4.182.1.32 `#define __HYPERVISOR_sched_op __HYPERVISOR_sched_op_compat`

4.182.1.33 `#define __HYPERVISOR_sched_op_compat 6 /* compat since 0x00030101 */`

4.182.1.34 `#define __HYPERVISOR_set_callbacks 4`

4.182.1.35 `#define __HYPERVISOR_set_debugreg 8`

4.182.1.36 `#define __HYPERVISOR_set_gdt 2`

4.182.1.37 `#define __HYPERVISOR_set_segment_base 25 /* x86/64 only */`

4.182.1.38 `#define __HYPERVISOR_set_timer_op 15`

- 4.182.1.39 #define \_\_HYPERVISOR\_set\_trap\_table 0
- 4.182.1.40 #define \_\_HYPERVISOR\_stack\_switch 3
- 4.182.1.41 #define \_\_HYPERVISOR\_sysctl 35
- 4.182.1.42 #define \_\_HYPERVISOR\_tmem\_op 38
- 4.182.1.43 #define \_\_HYPERVISOR\_update\_descriptor 10
- 4.182.1.44 #define \_\_HYPERVISOR\_update\_va\_mapping 14
- 4.182.1.45 #define \_\_HYPERVISOR\_update\_va\_mapping\_otherdomain 22
- 4.182.1.46 #define \_\_HYPERVISOR\_vcpu\_op 24
- 4.182.1.47 #define \_\_HYPERVISOR\_vm\_assist 21
- 4.182.1.48 #define \_\_HYPERVISOR\_xen\_version 17
- 4.182.1.49 #define \_\_HYPERVISOR\_xenoprof\_op 31
- 4.182.1.50 #define \_\_HYPERVISOR\_xsm\_op 27
- 4.182.1.51 #define \_\_mk\_unsigned\_long( x ) x ## UL
- 4.182.1.52 #define console\_evtchn console.domU.evtchn
- 4.182.1.53 #define console\_mfn console.domU.mfn
- 4.182.1.54 #define CONSOLEIO\_read 1
- 4.182.1.55 #define CONSOLEIO\_write 0
- 4.182.1.56 #define DOMID\_COW (0x7FF3U)
- 4.182.1.57 #define DOMID\_FIRST\_RESERVED (0x7FF0U)
- 4.182.1.58 #define DOMID\_IDLE (0x7FFFU)
- 4.182.1.59 #define DOMID\_INVALID (0x7FF4U)
- 4.182.1.60 #define DOMID\_IO (0x7FF1U)
- 4.182.1.61 #define DOMID\_SELF (0x7FF0U)
- 4.182.1.62 #define DOMID\_XEN (0x7FF2U)
- 4.182.1.63 #define MAX\_GUEST\_CMDLINE 1024
- 4.182.1.64 #define MAX\_VMASST\_TYPE 3
- 4.182.1.65 #define mk\_unsigned\_long( x ) \_\_mk\_unsigned\_long(x)
- 4.182.1.66 #define MMU\_MACHPHYS\_UPDATE 1 /\* ptr = MA of frame to modify entry for \*/

4.182.1.67 `#define MMU_NORMAL_PT_UPDATE 0 /* checked '*ptr = val'. ptr is MA. */`

4.182.1.68 `#define MMU_PT_UPDATE_PRESERVE_AD 2 /* atomically: *ptr = val | (*ptr&(A|D)) */`

4.182.1.69 `#define MMUEXT_CLEAR_PAGE 16`

4.182.1.70 `#define MMUEXT_COPY_PAGE 17`

4.182.1.71 `#define MMUEXT_FLUSH_CACHE 12`

4.182.1.72 `#define MMUEXT_FLUSH_CACHE_GLOBAL 18`

4.182.1.73 `#define MMUEXT_INVLPG_ALL 11`

4.182.1.74 `#define MMUEXT_INVLPG_LOCAL 7`

4.182.1.75 `#define MMUEXT_INVLPG_MULTI 9`

4.182.1.76 `#define MMUEXT_MARK_SUPER 19`

4.182.1.77 `#define MMUEXT_NEW_BASEPTR 5`

4.182.1.78 `#define MMUEXT_NEW_USER_BASEPTR 15`

4.182.1.79 `#define MMUEXT_PIN_L1_TABLE 0`

4.182.1.80 `#define MMUEXT_PIN_L2_TABLE 1`

4.182.1.81 `#define MMUEXT_PIN_L3_TABLE 2`

4.182.1.82 `#define MMUEXT_PIN_L4_TABLE 3`

4.182.1.83 `#define MMUEXT_SET_LDT 13`

4.182.1.84 `#define MMUEXT_TLB_FLUSH_ALL 10`

4.182.1.85 `#define MMUEXT_TLB_FLUSH_LOCAL 6`

4.182.1.86 `#define MMUEXT_TLB_FLUSH_MULTI 8`

4.182.1.87 `#define MMUEXT_UNMARK_SUPER 20`

4.182.1.88 `#define MMUEXT_UNPIN_TABLE 4`

4.182.1.89 `#define NR_EVENT_CHANNELS (sizeof(unsigned long) * sizeof(unsigned long) * 64)`

4.182.1.90 `#define NR_VIRQS 24`

4.182.1.91 `#define SIF_INITDOMAIN (1<<1) /* Is this the initial control domain? */`

4.182.1.92 `#define SIF_MOD_START_PFN (1<<3) /* Is mod_start a PFN? */`

4.182.1.93 `#define SIF_MULTIBOOT_MOD (1<<2) /* Is mod_start a multiboot module? */`

4.182.1.94 `#define SIF_PM_MASK (0xFF<<8) /* reserve 1 byte for xen-pm options */`



- 4.182.1.95 `#define SIF_PRIVILEGED (1<<0) /* Is the domain privileged? */`
- 4.182.1.96 `#define UVMF_ALL (1UL<<2) /* Flush all TLBs. */`
- 4.182.1.97 `#define UVMF_FLUSHTYPE_MASK (3UL<<0)`
- 4.182.1.98 `#define UVMF_INVLPG (2UL<<0) /* Flush only one entry. */`
- 4.182.1.99 `#define UVMF_LOCAL (0UL<<2) /* Flush local TLB. */`
- 4.182.1.100 `#define UVMF_MULTI (0UL<<2) /* Flush subset of TLBs. */`
- 4.182.1.101 `#define UVMF_NONE (0UL<<0) /* No flushing at all. */`
- 4.182.1.102 `#define UVMF_TLB_FLUSH (1UL<<0) /* Flush entire TLB(s). */`
- 4.182.1.103 `#define VIRQ_ARCH_0 16`
- 4.182.1.104 `#define VIRQ_ARCH_1 17`
- 4.182.1.105 `#define VIRQ_ARCH_2 18`
- 4.182.1.106 `#define VIRQ_ARCH_3 19`
- 4.182.1.107 `#define VIRQ_ARCH_4 20`
- 4.182.1.108 `#define VIRQ_ARCH_5 21`
- 4.182.1.109 `#define VIRQ_ARCH_6 22`
- 4.182.1.110 `#define VIRQ_ARCH_7 23`
- 4.182.1.111 `#define VIRQ_CON_RING 8 /* G. (DOM0) Bytes received on console */`
- 4.182.1.112 `#define VIRQ_CONSOLE 2 /* G. (DOM0) Bytes received on emergency console. */`
- 4.182.1.113 `#define VIRQ_DEBUG 1 /* V. Request guest to dump debug info. */`
- 4.182.1.114 `#define VIRQ_DEBUGGER 6 /* G. (DOM0) A domain has paused for debugging. */`
- 4.182.1.115 `#define VIRQ_DOM_EXC 3 /* G. (DOM0) Exceptional event for some domain. */`
- 4.182.1.116 `#define VIRQ_MEM_EVENT 10 /* G. (DOM0) A memory event has occurred */`
- 4.182.1.117 `#define VIRQ_PCPU_STATE 9 /* G. (DOM0) PCPU state changed */`
- 4.182.1.118 `#define VIRQ_TBUF 4 /* G. (DOM0) Trace buffer has records available. */`
- 4.182.1.119 `#define VIRQ_TIMER 0 /* V. Timebase update, and/or requested timeout. */`
- 4.182.1.120 `#define VIRQ_XENOPROF 7 /* V. XenOprofile interrupt: new sample available */`
- 4.182.1.121 `#define VMASST_CMD_disable 1`
- 4.182.1.122 `#define VMASST_CMD_enable 0`

4.182.1.123 `#define VMASST_TYPE_4gb_segments 0`

4.182.1.124 `#define VMASST_TYPE_4gb_segments_notify 1`

4.182.1.125 `#define VMASST_TYPE_pae_extended_cr3 3`

4.182.1.126 `#define VMASST_TYPE_writable_pagetables 2`

4.182.1.127 `#define xen_vga_console_info dom0_vga_console_info`

4.182.1.128 `#define xen_vga_console_info_t dom0_vga_console_info_t`

4.182.1.129 `#define XEN_VGATYPE_TEXT_MODE_3 0x03`

4.182.1.130 `#define XEN_VGATYPE_VESA_LFB 0x23`

## 4.182.2 Typedef Documentation

4.182.2.1 `typedef struct dom0_vga_console_info dom0_vga_console_info_t`

4.182.2.2 `typedef uint16_t domid_t`

4.182.2.3 `typedef struct mmu_update mmu_update_t`

4.182.2.4 `typedef struct mmuext_op mmuext_op_t`

4.182.2.5 `typedef struct multicall_entry multicall_entry_t`

4.182.2.6 `typedef struct shared_info shared_info_t`

4.182.2.7 `typedef struct start_info start_info_t`

4.182.2.8 `typedef struct vcpu_info vcpu_info_t`

4.182.2.9 `typedef struct vcpu_time_info vcpu_time_info_t`

4.182.2.10 `typedef uint8_t xen_domain_handle_t[16]`

## 4.182.3 Function Documentation

4.182.3.1 `__DEFINE_XEN_GUEST_HANDLE ( uchar , unsigned char )`

4.182.3.2 `__DEFINE_XEN_GUEST_HANDLE ( uint , unsigned int )`

4.182.3.3 `__DEFINE_XEN_GUEST_HANDLE ( ulong , unsigned long )`

4.182.3.4 `__DEFINE_XEN_GUEST_HANDLE ( uint8 , uint8_t )`

4.182.3.5 `__DEFINE_XEN_GUEST_HANDLE ( uint16 , uint16_t )`

4.182.3.6 `__DEFINE_XEN_GUEST_HANDLE ( uint32 , uint32_t )`

4.182.3.7 `__DEFINE_XEN_GUEST_HANDLE ( uint64 , uint64_t )`

4.182.3.8 `DEFINE_XEN_GUEST_HANDLE ( char )`

- 4.182.3.9 `DEFINE_XEN_GUEST_HANDLE ( int )`
- 4.182.3.10 `DEFINE_XEN_GUEST_HANDLE ( long )`
- 4.182.3.11 `DEFINE_XEN_GUEST_HANDLE ( void )`
- 4.182.3.12 `DEFINE_XEN_GUEST_HANDLE ( uint64_t )`
- 4.182.3.13 `DEFINE_XEN_GUEST_HANDLE ( xen_pfn_t )`
- 4.182.3.14 `DEFINE_XEN_GUEST_HANDLE ( mmuext_op_t )`
- 4.182.3.15 `DEFINE_XEN_GUEST_HANDLE ( mmu_update_t )`
- 4.182.3.16 `DEFINE_XEN_GUEST_HANDLE ( multicall_entry_t )`

## 4.183 xen/xen/include/public/xsm/caas\_op.h File Reference

### Data Structures

- struct [caas\\_op](#)

### Macros

- `#define CAAS_EXEMPT 1`
- `#define CAAS_LAST CAAS_EXEMPT`

### Typedefs

- `typedef struct caas\_op caas\_op\_t`

### Functions

- `DEFINE\_XEN\_GUEST\_HANDLE (caas\_op\_t)`

#### 4.183.1 Macro Definition Documentation

- 4.183.1.1 `#define CAAS_EXEMPT 1`
- 4.183.1.2 `#define CAAS_LAST CAAS_EXEMPT`

#### 4.183.2 Typedef Documentation

- 4.183.2.1 `typedef struct caas\_op caas\_op\_t`

#### 4.183.3 Function Documentation

- 4.183.3.1 `DEFINE\_XEN\_GUEST\_HANDLE ( caas\_op\_t )`

## 4.184 xen/xen/include/xen/domain.h File Reference

```
#include <public/xen.h>
#include <asm/domain.h>
```

### Data Structures

- union [vcpu\\_guest\\_context\\_u](#)

### Functions

- struct [vcpu](#) \* [alloc\\_vcpu](#) (struct [domain](#) \*[d](#), unsigned int [vcpu\\_id](#), unsigned int [cpu\\_id](#))
- int [boot\\_vcpu](#) (struct [domain](#) \*[d](#), int [vcpuid](#), [vcpu\\_guest\\_context\\_u](#) [ctxt](#))
- struct [vcpu](#) \* [alloc\\_dom0\\_vcpu0](#) (void)
- struct [vcpu](#) \* [alloc\\_domt\\_vcpu0](#) (void)
- void [vcpu\\_reset](#) (struct [vcpu](#) \*[v](#))
- void [getdomaininfo](#) (struct [domain](#) \*[d](#), struct [xen\\_domctl\\_getdomaininfo](#) \*[info](#))
- struct [domain](#) \* [alloc\\_domain\\_struct](#) (void)
- void [free\\_domain\\_struct](#) (struct [domain](#) \*[d](#))
- struct [vcpu](#) \* [alloc\\_vcpu\\_struct](#) (void)
- void [free\\_vcpu\\_struct](#) (struct [vcpu](#) \*[v](#))
- int [vcpu\\_initialise](#) (struct [vcpu](#) \*[v](#))
- void [vcpu\\_destroy](#) (struct [vcpu](#) \*[v](#))
- int [arch\\_domain\\_create](#) (struct [domain](#) \*[d](#), unsigned int [domcr\\_flags](#))
- void [arch\\_domain\\_destroy](#) (struct [domain](#) \*[d](#))
- int [arch\\_set\\_info\\_guest](#) (struct [vcpu](#) \*, [vcpu\\_guest\\_context\\_u](#))
- void [arch\\_get\\_info\\_guest](#) (struct [vcpu](#) \*, [vcpu\\_guest\\_context\\_u](#))
- int [domain\\_relinquish\\_resources](#) (struct [domain](#) \*[d](#))
- void [dump\\_pageframe\\_info](#) (struct [domain](#) \*[d](#))
- void [arch\\_dump\\_vcpu\\_info](#) (struct [vcpu](#) \*[v](#))
- void [arch\\_dump\\_domain\\_info](#) (struct [domain](#) \*[d](#))
- void [arch\\_vcpu\\_reset](#) (struct [vcpu](#) \*[v](#))
- bool\_t [domctl\\_lock\\_acquire](#) (void)
- void [domctl\\_lock\\_release](#) (void)
- int [continue\\_hypercall\\_on\\_cpu](#) (unsigned int [cpu](#), long(\*[func](#))(void \*[data](#)), void \*[data](#))

### Variables

- unsigned int [xen\\_processor\\_pmbits](#)

#### 4.184.1 Function Documentation

4.184.1.1 struct [vcpu](#)\* [alloc\\_dom0\\_vcpu0](#) ( void )

4.184.1.2 struct [domain](#)\* [alloc\\_domain\\_struct](#) ( void )

4.184.1.3 struct [vcpu](#)\* [alloc\\_domt\\_vcpu0](#) ( void )

4.184.1.4 struct [vcpu](#)\* [alloc\\_vcpu](#) ( struct [domain](#) \* [d](#), unsigned int [vcpu\\_id](#), unsigned int [cpu\\_id](#) )

4.184.1.5 struct [vcpu](#)\* [alloc\\_vcpu\\_struct](#) ( void )

4.184.1.6 `int arch_domain_create ( struct domain * d, unsigned int domcr_flags )`

4.184.1.7 `void arch_domain_destroy ( struct domain * d )`

4.184.1.8 `void arch_dump_domain_info ( struct domain * d )`

4.184.1.9 `void arch_dump_vcpu_info ( struct vcpu * v )`

4.184.1.10 `void arch_get_info_guest ( struct vcpu *, vcpu_guest_context_u )`

4.184.1.11 `int arch_set_info_guest ( struct vcpu *, vcpu_guest_context_u )`

4.184.1.12 `void arch_vcpu_reset ( struct vcpu * v )`

4.184.1.13 `int boot_vcpu ( struct domain * d, int vcpuid, vcpu_guest_context_u ctxt )`

4.184.1.14 `int continue_hypercall_on_cpu ( unsigned int cpu, long(*)(void *data) func, void * data )`

Continue the current hypercall via func(data) on specified cpu. If this function returns 0 then the function is guaranteed to run at some point in the future. If this function returns an error code then the function has not been and will not be executed.

4.184.1.15 `int domain_relinquish_resources ( struct domain * d )`

4.184.1.16 `bool_t domctl_lock_acquire ( void )`

4.184.1.17 `void domctl_lock_release ( void )`

4.184.1.18 `void dump_pageframe_info ( struct domain * d )`

4.184.1.19 `void free_domain_struct ( struct domain * d )`

4.184.1.20 `void free_vcpu_struct ( struct vcpu * v )`

4.184.1.21 `void getdomaininfo ( struct domain * d, struct xen_domctl_getdomaininfo * info )`

4.184.1.22 `void vcpu_destroy ( struct vcpu * v )`

4.184.1.23 `int vcpu_initialise ( struct vcpu * v )`

Initialise/destroy arch-specific details of a VCPU.

- `vcpu_initialise()` is called after the basic generic fields of the VCPU structure are initialised. Many operations can be applied to the VCPU at this point (e.g., `vcpu_pause()`).
- `vcpu_destroy()` is called only if `vcpu_initialise()` previously succeeded.

4.184.1.24 `void vcpu_reset ( struct vcpu * v )`

## 4.184.2 Variable Documentation

4.184.2.1 `unsigned int xen_processor_pmbits`

## 4.185 xen/xen/include/xen/sched.h File Reference

```
#include <xen/config.h>
#include <xen/types.h>
#include <xen/spinlock.h>
#include <xen/smp.h>
#include <xen/shared.h>
#include <public/xen.h>
#include <public/domctl.h>
#include <public/sysctl.h>
#include <public/vcpu.h>
#include <public/xsm/acm.h>
#include <xen/time.h>
#include <xen/timer.h>
#include <xen/grant_table.h>
#include <xen/rangeset.h>
#include <xen/domain.h>
#include <xen/xenoprof.h>
#include <xen/rcupdate.h>
#include <xen/irq.h>
#include <xen/mm.h>
#include <xen/tasklet.h>
#include <public/mem_event.h>
#include <xen/cpumask.h>
#include <xen/nodemask.h>
#include <xen/multicall.h>
```

### Data Structures

- struct [evtchn](#)
- struct [vcpu](#)
- struct [mem\\_event\\_domain](#)
- struct [domain](#)
- struct [domain\\_setup\\_info](#)

### Macros

- #define [BITS\\_PER\\_EVTCHN\\_WORD\(d\)](#) BITS\_PER\_LONG
- #define [MAX\\_EVTCHNS\(d\)](#) (BITS\_PER\_EVTCHN\_WORD(d) \* BITS\_PER\_EVTCHN\_WORD(d))
- #define [EVTCHNS\\_PER\\_BUCKET](#) 128
- #define [NR\\_EVTCHN\\_BUCKETS](#) (NR\_EVENT\_CHANNELS / EVTCHNS\_PER\_BUCKET)
- #define [ECS\\_FREE](#) 0 /\* Channel is available for use. \*/
- #define [ECS\\_RESERVED](#) 1 /\* Channel is reserved. \*/
- #define [ECS\\_UNBOUND](#) 2 /\* Channel is waiting to bind to a remote domain. \*/
- #define [ECS\\_INTERDOMAIN](#) 3 /\* Channel is bound to another domain. \*/
- #define [ECS\\_PIRQ](#) 4 /\* Channel is bound to a physical IRQ line. \*/
- #define [ECS\\_VIRQ](#) 5 /\* Channel is bound to a virtual IRQ line. \*/
- #define [ECS\\_IPI](#) 6 /\* Channel is bound to a virtual IPI line. \*/
- #define [runstate\\_guest\(v\)](#) ((v)->runstate\_guest)
- #define [domain\\_lock\(d\)](#) spin\_lock\_recursive(&(d)->domain\_lock)
- #define [domain\\_unlock\(d\)](#) spin\_unlock\_recursive(&(d)->domain\_lock)
- #define [domain\\_is\\_locked\(d\)](#) spin\_is\_locked(&(d)->domain\_lock)
- #define [NR\\_DOMAIN\\_WATCHDOG\\_TIMERS](#) 2
- #define [PAEKERN\\_no](#) 0

- #define PAEKERN\_yes 1
- #define PAEKERN\_extended\_cr3 2
- #define PAEKERN\_bimodal 3
- #define is\_idle\_domain(d) ((d)->domain\_id == DOMID\_IDLE)
- #define is\_idle\_vcpu(v) (is\_idle\_domain((v)->domain))
- #define DOMAIN\_DESTROYED (1<<31) /\* assumes atomic\_t is >= 32 bits \*/
- #define put\_domain(\_d) if ( atomic\_dec\_and\_test(&(\_d)->refcnt) ) domain\_destroy(\_d)
- #define \_DOMCRF\_hvm 0
- #define DOMCRF\_hvm (1U<<\_DOMCRF\_hvm)
- #define \_DOMCRF\_hap 1
- #define DOMCRF\_hap (1U<<\_DOMCRF\_hap)
- #define \_DOMCRF\_s3\_integrity 2
- #define DOMCRF\_s3\_integrity (1U<<\_DOMCRF\_s3\_integrity)
- #define \_DOMCRF\_dummy 3
- #define DOMCRF\_dummy (1U<<\_DOMCRF\_dummy)
- #define \_DOMCRF\_oos\_off 4
- #define DOMCRF\_oos\_off (1U<<\_DOMCRF\_oos\_off)
- #define \_DOMCRF\_priv 5
- #define DOMCRF\_priv (1U<<\_DOMCRF\_priv)
- #define domain\_crash(d)
- #define domain\_crash\_synchronous()
- #define set\_current\_state(\_s) do { current->state = (\_s); } while (0)
- #define hypercall\_preempt\_check()
- #define for\_each\_domain(\_d)
- #define for\_each\_vcpu(\_d, \_v)
- #define \_VPF\_blocked 0
- #define VPF\_blocked (1UL<<\_VPF\_blocked)
- #define \_VPF\_down 1
- #define VPF\_down (1UL<<\_VPF\_down)
- #define \_VPF\_blocked\_in\_xen 2
- #define VPF\_blocked\_in\_xen (1UL<<\_VPF\_blocked\_in\_xen)
- #define \_VPF\_migrating 3
- #define VPF\_migrating (1UL<<\_VPF\_migrating)
- #define \_VPF\_mem\_event 4
- #define VPF\_mem\_event (1UL<<\_VPF\_mem\_event)
- #define cpu\_is\_halttable(cpu)
- #define IS\_PRIV(\_d) ((\_d)->is\_privileged)
- #define IS\_PRIV\_FOR(\_d, \_t) (IS\_PRIV(\_d) || ((\_d)->target && (\_d)->target == (\_t)))
- #define VM\_ASSIST(\_d, \_t) (test\_bit(\_t, &(\_d)->vm\_assist))
- #define is\_hvm\_domain(d) ((d)->is\_hvm)
- #define is\_hvm\_vcpu(v) (is\_hvm\_domain(v->domain))
- #define is\_pinned\_vcpu(v)
- #define need\_iommu(d) ((d)->need\_iommu)
- #define CPUPOOLID\_NONE -1
- #define num\_cpupool\_cpus(c) (cpus\_weight((c)->cpu\_valid))

## Enumerations

- enum cpufreq\_controller { FREQCTL\_none, FREQCTL\_dom0\_kernel, FREQCTL\_xen }

## Functions

- int [evtchn\\_init](#) (struct [domain](#) \*d)
- void [evtchn\\_destroy](#) (struct [domain](#) \*d)
- void [evtchn\\_destroy\\_final](#) (struct [domain](#) \*d)
- void [domain\\_update\\_node\\_affinity](#) (struct [domain](#) \*d)
- struct [domain](#) \* [domain\\_create](#) (domid\_t domid, unsigned int domcr\_flags, ssidref\_t ssidref)
- struct [domain](#) \* [rcu\\_lock\\_domain\\_by\\_id](#) (domid\_t dom)
- int [rcu\\_lock\\_target\\_domain\\_by\\_id](#) (domid\_t dom, struct [domain](#) \*\*d)
- int [rcu\\_lock\\_remote\\_target\\_domain\\_by\\_id](#) (domid\_t dom, struct [domain](#) \*\*d)
- struct [domain](#) \* [get\\_domain\\_by\\_id](#) (domid\_t dom)
- void [domain\\_destroy](#) (struct [domain](#) \*d)
- int [domain\\_kill](#) (struct [domain](#) \*d)
- void [domain\\_shutdown](#) (struct [domain](#) \*d, u8 reason)
- void [domain\\_resume](#) (struct [domain](#) \*d)
- void [domain\\_pause\\_for\\_debugger](#) (void)
- int [vcpu\\_start\\_shutdown\\_deferral](#) (struct [vcpu](#) \*v)
- void [vcpu\\_end\\_shutdown\\_deferral](#) (struct [vcpu](#) \*v)
- void [\\_\\_domain\\_crash](#) (struct [domain](#) \*d)
- void [\\_\\_domain\\_crash\\_synchronous](#) (void) [\\_\\_attribute\\_\\_\(\(noreturn\)\)](#)
- void [scheduler\\_init](#) (void)
- int [sched\\_init\\_vcpu](#) (struct [vcpu](#) \*v, unsigned int processor)
- void [sched\\_destroy\\_vcpu](#) (struct [vcpu](#) \*v)
- int [sched\\_init\\_domain](#) (struct [domain](#) \*d)
- void [sched\\_destroy\\_domain](#) (struct [domain](#) \*d)
- int [sched\\_move\\_domain](#) (struct [domain](#) \*d, struct [cpupool](#) \*c)
- long [sched\\_adjust](#) (struct [domain](#) \*, struct [xen\\_domctl\\_scheduler\\_op](#) \*)
- long [sched\\_adjust\\_global](#) (struct [xen\\_sysctl\\_scheduler\\_op](#) \*)
- int [sched\\_id](#) (void)
- void [sched\\_tick\\_suspend](#) (void)
- void [sched\\_tick\\_resume](#) (void)
- void [vcpu\\_wake](#) (struct [vcpu](#) \*d)
- void [vcpu\\_sleep\\_nosync](#) (struct [vcpu](#) \*d)
- void [vcpu\\_sleep\\_sync](#) (struct [vcpu](#) \*d)
- void [sync\\_vcpu\\_execstate](#) (struct [vcpu](#) \*v)
- void [sync\\_local\\_execstate](#) (void)
- void [context\\_switch](#) (struct [vcpu](#) \*prev, struct [vcpu](#) \*next)
- void [context\\_saved](#) (struct [vcpu](#) \*prev)
- void [continue\\_running](#) (struct [vcpu](#) \*same)
- void [startup\\_cpu\\_idle\\_loop](#) (void)
- unsigned long [hypercall\\_create\\_continuation](#) (unsigned int op, const char \*format,...)
- void [vcpu\\_unblock](#) (struct [vcpu](#) \*v)
- void [vcpu\\_pause](#) (struct [vcpu](#) \*v)
- void [vcpu\\_pause\\_nosync](#) (struct [vcpu](#) \*v)
- void [domain\\_pause](#) (struct [domain](#) \*d)
- void [vcpu\\_unpause](#) (struct [vcpu](#) \*v)
- void [domain\\_unpause](#) (struct [domain](#) \*d)
- void [domain\\_pause\\_by\\_systemcontroller](#) (struct [domain](#) \*d)
- void [domain\\_unpause\\_by\\_systemcontroller](#) (struct [domain](#) \*d)
- void [cpu\\_init](#) (void)
- struct [scheduler](#) \* [scheduler\\_get\\_default](#) (void)
- struct [scheduler](#) \* [scheduler\\_alloc](#) (unsigned int [sched\\_id](#), int \*perr)
- void [scheduler\\_free](#) (struct [scheduler](#) \*sched)
- int [schedule\\_cpu\\_switch](#) (unsigned int cpu, struct [cpupool](#) \*c)
- void [vcpu\\_force\\_reschedule](#) (struct [vcpu](#) \*v)



- int [cpu\\_disable\\_scheduler](#) (unsigned int cpu)
- int [vcpu\\_set\\_affinity](#) (struct [vcpu](#) \*v, cpumask\_t \*affinity)
- void [vcpu\\_runstate\\_get](#) (struct [vcpu](#) \*v, struct [vcpu\\_runstate\\_info](#) \*runstate)
- uint64\_t [get\\_cpu\\_idle\\_time](#) (unsigned int cpu)
- void [watchdog\\_domain\\_init](#) (struct [domain](#) \*d)
- void [watchdog\\_domain\\_destroy](#) (struct [domain](#) \*d)
- void [set\\_vcpu\\_migration\\_delay](#) (unsigned int delay)
- unsigned int [get\\_vcpu\\_migration\\_delay](#) (void)
- struct [cpupool](#) \* [cpupool\\_get\\_by\\_id](#) (int poolid)
- void [cpupool\\_put](#) (struct [cpupool](#) \*pool)
- int [cpupool\\_add\\_domain](#) (struct [domain](#) \*d, int poolid)
- void [cpupool\\_rm\\_domain](#) (struct [domain](#) \*d)
- int [cpupool\\_do\\_sysctl](#) (struct [xen\\_sysctl\\_cpupool\\_op](#) \*op)

## Variables

- struct [domain](#) \* [dom0](#)
- struct [domain](#) \* [domt](#)
- spinlock\_t [domlist\\_update\\_lock](#)
- rcu\_read\_lock\_t [domlist\\_read\\_lock](#)
- struct [vcpu](#) \* [idle\\_vcpu](#) [NR\_CPUS]
- struct [domain](#) \* [domain\\_list](#)
- bool\_t [sched\\_smt\\_power\\_savings](#)
- enum [cpufreq\\_controller](#) [cpufreq\\_controller](#)

## 4.185.1 Macro Definition Documentation

4.185.1.1 [#define \\_DOMCRF\\_dummy](#) 3

4.185.1.2 [#define \\_DOMCRF\\_hap](#) 1

4.185.1.3 [#define \\_DOMCRF\\_hvm](#) 0

4.185.1.4 [#define \\_DOMCRF\\_oos\\_off](#) 4

4.185.1.5 [#define \\_DOMCRF\\_priv](#) 5

4.185.1.6 [#define \\_DOMCRF\\_s3\\_integrity](#) 2

4.185.1.7 [#define \\_VPF\\_blocked](#) 0

4.185.1.8 [#define \\_VPF\\_blocked\\_in\\_xen](#) 2

4.185.1.9 [#define \\_VPF\\_down](#) 1

4.185.1.10 [#define \\_VPF\\_mem\\_event](#) 4

4.185.1.11 [#define \\_VPF\\_migrating](#) 3

4.185.1.12 [#define BITS\\_PER\\_EVTCHN\\_WORD\( d \)](#) BITS\_PER\_LONG

4.185.1.13 [#define cpu\\_is\\_halttable\( cpu \)](#)

## Value:

```
(!softirq_pending(cpu) &&
  cpu_online(cpu) &&
  !per_cpu(tasklet_work_to_do, cpu))
```

4.185.1.14 **#define CPUPOOLID\_NONE -1**

4.185.1.15 **#define domain\_crash( d )**

**Value:**

```
do {
  printk("domain_crash called from %s:%d\n", __FILE__, __LINE__);
  __domain_crash(d);
} while (0)
```

4.185.1.16 **#define domain\_crash\_synchronous( )**

**Value:**

```
do {
  printk("domain_crash_sync called from %s:%d\n", __FILE__, __LINE__);
  __domain_crash_synchronous();
} while (0)
```

4.185.1.17 **#define DOMAIN\_DESTROYED (1<<31) /\* assumes atomic\_t is >= 32 bits \*/**

4.185.1.18 **#define domain\_is\_locked( d ) spin\_is\_locked(&(d)->domain\_lock)**

4.185.1.19 **#define domain\_lock( d ) spin\_lock\_recursive(&(d)->domain\_lock)**

4.185.1.20 **#define domain\_unlock( d ) spin\_unlock\_recursive(&(d)->domain\_lock)**

4.185.1.21 **#define DOMCRF\_dummy (1U<<\_DOMCRF\_dummy)**

4.185.1.22 **#define DOMCRF\_hap (1U<<\_DOMCRF\_hap)**

4.185.1.23 **#define DOMCRF\_hvm (1U<<\_DOMCRF\_hvm)**

4.185.1.24 **#define DOMCRF\_oos\_off (1U<<\_DOMCRF\_oos\_off)**

4.185.1.25 **#define DOMCRF\_priv (1U<<\_DOMCRF\_priv)**

4.185.1.26 **#define DOMCRF\_s3\_integrity (1U<<\_DOMCRF\_s3\_integrity)**

4.185.1.27 **#define ECS\_FREE 0 /\* Channel is available for use. \*/**

4.185.1.28 **#define ECS\_INTERDOMAIN 3 /\* Channel is bound to another domain. \*/**

4.185.1.29 **#define ECS\_IPI 6 /\* Channel is bound to a virtual IPI line. \*/**

4.185.1.30 **#define ECS\_PIRQ 4 /\* Channel is bound to a physical IRQ line. \*/**

4.185.1.31 **#define ECS\_RESERVED 1 /\* Channel is reserved. \*/**

4.185.1.32 **#define ECS\_UNBOUND 2 /\* Channel is waiting to bind to a remote domain. \*/**

4.185.1.33 **#define** ECS\_VIRQ 5 /\* Channel is bound to a virtual IRQ line. \*/

4.185.1.34 **#define** EVTCHNS\_PER\_BUCKET 128

4.185.1.35 **#define** for\_each\_domain( *\_d* )

**Value:**

```
for ( (_d) = rcu_dereference(domain_list);
      (_d) != NULL;
      (_d) = rcu_dereference((_d)->next_in_list) ) \
```

4.185.1.36 **#define** for\_each\_vcpu( *\_d*, *\_v* )

**Value:**

```
for ( (_v) = (_d)->vcpu ? (_d)->vcpu[0] : NULL;
      (_v) != NULL;
      (_v) = (_v)->next_in_list )
```

4.185.1.37 **#define** hypercall\_preempt\_check( )

**Value:**

```
(unlikely(
    softirq_pending(smp_processor_id()) |
    local_events_need_delivery()
))
```

4.185.1.38 **#define** is\_hvm\_domain( *d* ) ((*d*)->is\_hvm)

4.185.1.39 **#define** is\_hvm\_vcpu( *v* ) (is\_hvm\_domain(*v*->domain))

4.185.1.40 **#define** is\_idle\_domain( *d* ) ((*d*)->domain\_id == DOMID\_IDLE)

4.185.1.41 **#define** is\_idle\_vcpu( *v* ) (is\_idle\_domain((*v*)->domain))

4.185.1.42 **#define** is\_pinned\_vcpu( *v* )

**Value:**

```
((v)->domain->is_pinned ||
    cpus_weight((v)->cpu_affinity) == 1)
```

4.185.1.43 **#define** IS\_PRIV( *\_d* ) ((*\_d*)->is\_privileged)

4.185.1.44 **#define** IS\_PRIV\_FOR( *\_d*, *\_t* ) (IS\_PRIV(*\_d*) || ((*\_d*)->target && (*\_d*)->target == (*\_t*)))

4.185.1.45 **#define** MAX\_EVTCHNS( *d* ) (BITS\_PER\_EVTCHN\_WORD(*d*) \* BITS\_PER\_EVTCHN\_WORD(*d*))

4.185.1.46 **#define** need\_iommu( *d* ) ((*d*)->need\_iommu)

4.185.1.47 **#define** NR\_DOMAIN\_WATCHDOG\_TIMERS 2

```

4.185.1.48 #define NR_EVTCHN_BUCKETS (NR_EVENT_CHANNELS / EVTCHNS_PER_BUCKET)

4.185.1.49 #define num_cpupool_cpus( c ) (cpus_weight((c)->cpu_valid))

4.185.1.50 #define PAEKERN_bimodal 3

4.185.1.51 #define PAEKERN_extended_cr3 2

4.185.1.52 #define PAEKERN_no 0

4.185.1.53 #define PAEKERN_yes 1

4.185.1.54 #define put_domain( _d ) if ( atomic_dec_and_test(&(_d)->refcnt) ) domain_destroy(_d)

4.185.1.55 #define runstate_guest( v ) ((v)->runstate_guest)

4.185.1.56 #define set_current_state( _s ) do { current->state = (_s); } while (0)

4.185.1.57 #define VM_ASSIST( _d, _t ) (test_bit((_t), &(_d)->vm_assist))

4.185.1.58 #define VPF_blocked (1UL<<_VPF_blocked)

4.185.1.59 #define VPF_blocked_in_xen (1UL<<_VPF_blocked_in_xen)

4.185.1.60 #define VPF_down (1UL<<_VPF_down)

4.185.1.61 #define VPF_mem_event (1UL<<_VPF_mem_event)

4.185.1.62 #define VPF_migrating (1UL<<_VPF_migrating)

```

## 4.185.2 Enumeration Type Documentation

4.185.2.1 enum cpufreq\_controller

Enumerator

***FREQCTL\_none***

***FREQCTL\_dom0\_kernel***

***FREQCTL\_xen***

## 4.185.3 Function Documentation

4.185.3.1 void \_\_domain\_crash ( struct domain \* d )

Mark specified domain as crashed. This function always returns, even if the caller is the specified domain. The domain is not synchronously descheduled from any processor.

4.185.3.2 void \_\_domain\_crash\_synchronous ( void )

Mark current domain as crashed and synchronously deschedule from the local processor. This function never returns.

4.185.3.3 void context\_saved ( struct vcpu \* *prev* )

As described above, [context\\_switch\(\)](#) must call this function when the local CPU is no longer running in 's context, and 's context is saved to memory. Alternatively, if implementing lazy context switching, ensure that invoking [sync\\_vcpu\\_execstate\(\)](#) will switch and commit .

4.185.3.4 void context\_switch ( struct vcpu \* *prev*, struct vcpu \* *next* )

Called by the scheduler to switch to another VCPU. This function must call [context\\_saved\(\)](#) when the local CPU is no longer running in 's context, and that context is saved to memory. Alternatively, if implementing lazy context switching, it suffices to ensure that invoking [sync\\_vcpu\\_execstate\(\)](#) will switch and commit 's state.

4.185.3.5 void continue\_running ( struct vcpu \* *same* )

Called by the scheduler to continue running the current VCPU.

4.185.3.6 int cpu\_disable\_scheduler ( unsigned int *cpu* )

## 4.185.3.7 void cpu\_init ( void )

4.185.3.8 int cpupool\_add\_domain ( struct domain \* *d*, int *poolid* )4.185.3.9 int cpupool\_do\_sysctl ( struct xen\_sysctl\_cpupool\_op \* *op* )4.185.3.10 struct cpupool\* cpupool\_get\_by\_id ( int *poolid* )4.185.3.11 void cpupool\_put ( struct cpupool \* *pool* )4.185.3.12 void cpupool\_rm\_domain ( struct domain \* *d* )4.185.3.13 struct domain\* domain\_create ( domid\_t *domid*, unsigned int *domcr\_flags*, ssidref\_t *ssidref* )4.185.3.14 void domain\_destroy ( struct domain \* *d* )

Release resources belonging to task .

4.185.3.15 int domain\_kill ( struct domain \* *d* )4.185.3.16 void domain\_pause ( struct domain \* *d* )4.185.3.17 void domain\_pause\_by\_systemcontroller ( struct domain \* *d* )

## 4.185.3.18 void domain\_pause\_for\_debugger ( void )

4.185.3.19 void domain\_resume ( struct domain \* *d* )4.185.3.20 void domain\_shutdown ( struct domain \* *d*, u8 *reason* )4.185.3.21 void domain\_unpause ( struct domain \* *d* )4.185.3.22 void domain\_unpause\_by\_systemcontroller ( struct domain \* *d* )4.185.3.23 void domain\_update\_node\_affinity ( struct domain \* *d* )

4.185.3.24 void `evtchn_destroy` ( struct domain \* *d* )

4.185.3.25 void `evtchn_destroy_final` ( struct domain \* *d* )

4.185.3.26 int `evtchn_init` ( struct domain \* *d* )

4.185.3.27 uint64\_t `get_cpu_idle_time` ( unsigned int *cpu* )

4.185.3.28 struct domain\* `get_domain_by_id` ( domid\_t *dom* )

4.185.3.29 unsigned int `get_vcpu_migration_delay` ( void )

4.185.3.30 unsigned long `hypercall_create_continuation` ( unsigned int *op*, const char \* *format*, ... )

Creates a continuation to resume the current hypercall. The caller should return immediately, propagating the value returned from this invocation. The format string specifies the types and number of hypercall arguments. It contains one character per argument as follows: 'i' [unsigned] {char, int} 'l' [unsigned] long 'h' guest handle (XEN\_GUEST\_HANDLE(foo))

4.185.3.31 struct domain\* `rcu_lock_domain_by_id` ( domid\_t *dom* )

`rcu_lock_domain_by_id()` is more efficient than `get_domain_by_id()`. This is the preferred function if the returned domain reference is short lived, but it cannot be used if the domain reference needs to be kept beyond the current scope (e.g., across a softirq). The returned domain reference must be discarded using `rcu_unlock_domain()`.

4.185.3.32 int `rcu_lock_remote_target_domain_by_id` ( domid\_t *dom*, struct domain \*\* *d* )

As `rcu_lock_target_domain_by_id()`, but will fail EPERM rather than resolve to local domain. Successful return always resolves to a remote domain that the local domain is privileged to control.

4.185.3.33 int `rcu_lock_target_domain_by_id` ( domid\_t *dom*, struct domain \*\* *d* )

As above function, but accounts for current domain context:

- Translates target DOMID\_SELF into caller's domain id; and
- Checks that caller has permission to act on the target domain.

4.185.3.34 long `sched_adjust` ( struct domain \* , struct xen\_domctl\_scheduler\_op \* )

4.185.3.35 long `sched_adjust_global` ( struct xen\_sysctl\_scheduler\_op \* )

4.185.3.36 void `sched_destroy_domain` ( struct domain \* *d* )

4.185.3.37 void `sched_destroy_vcpu` ( struct vcpu \* *v* )

4.185.3.38 int `sched_id` ( void )

4.185.3.39 int `sched_init_domain` ( struct domain \* *d* )

4.185.3.40 int `sched_init_vcpu` ( struct vcpu \* *v*, unsigned int *processor* )

4.185.3.41 int `sched_move_domain` ( struct domain \* *d*, struct cpupool \* *c* )

- 4.185.3.42 void sched\_tick\_resume ( void )
- 4.185.3.43 void sched\_tick\_suspend ( void )
- 4.185.3.44 int schedule\_cpu\_switch ( unsigned int *cpu*, struct cpupool \* *c* )
- 4.185.3.45 struct scheduler\* scheduler\_alloc ( unsigned int *sched\_id*, int \* *perr* )
- 4.185.3.46 void scheduler\_free ( struct scheduler \* *sched* )
- 4.185.3.47 struct scheduler\* scheduler\_get\_default ( void )
- 4.185.3.48 void scheduler\_init ( void )
- 4.185.3.49 void set\_vcpu\_migration\_delay ( unsigned int *delay* )
- 4.185.3.50 void startup\_cpu\_idle\_loop ( void )
- 4.185.3.51 void sync\_local\_execstate ( void )

As above, for any lazy state being held on the local CPU.

- 4.185.3.52 void sync\_vcpu\_execstate ( struct vcpu \* *v* )

Force synchronisation of given VCPU's state. If it is currently descheduled, this call will ensure that all its state is committed to memory and that no CPU is using critical state (e.g., page tables) belonging to the VCPU.

- 4.185.3.53 void vcpu\_end\_shutdown\_deferral ( struct vcpu \* *v* )
- 4.185.3.54 void vcpu\_force\_reschedule ( struct vcpu \* *v* )
- 4.185.3.55 void vcpu\_pause ( struct vcpu \* *v* )
- 4.185.3.56 void vcpu\_pause\_nosync ( struct vcpu \* *v* )
- 4.185.3.57 void vcpu\_runstate\_get ( struct vcpu \* *v*, struct vcpu\_runstate\_info \* *runstate* )
- 4.185.3.58 int vcpu\_set\_affinity ( struct vcpu \* *v*, cpumask\_t \* *affinity* )
- 4.185.3.59 void vcpu\_sleep\_nosync ( struct vcpu \* *d* )
- 4.185.3.60 void vcpu\_sleep\_sync ( struct vcpu \* *d* )
- 4.185.3.61 int vcpu\_start\_shutdown\_deferral ( struct vcpu \* *v* )
- 4.185.3.62 void vcpu\_unblock ( struct vcpu \* *v* )
- 4.185.3.63 void vcpu\_unpause ( struct vcpu \* *v* )
- 4.185.3.64 void vcpu\_wake ( struct vcpu \* *d* )
- 4.185.3.65 void watchdog\_domain\_destroy ( struct domain \* *d* )
- 4.185.3.66 void watchdog\_domain\_init ( struct domain \* *d* )

#### 4.185.4 Variable Documentation

4.185.4.1 `enum cpufreq_controller cpufreq_controller`

4.185.4.2 `struct domain* dom0`

4.185.4.3 `struct domain* domain_list`

4.185.4.4 `rcu_read_lock_t domlist_read_lock`

4.185.4.5 `spinlock_t domlist_update_lock`

4.185.4.6 `struct domain* domt`

4.185.4.7 `struct vcpu* idle_vcpu[NR_CPUS]`

4.185.4.8 `bool_t sched_smt_power_savings`

### 4.186 xen/xen/include/xsm/xsm.h File Reference

```
#include <xen/sched.h>
#include <xen/multiboot.h>
```

#### Macros

- `#define xsm_call(fn) 0`
- `#define XSM_MAGIC 0x00000000`

#### Typedefs

- `typedef void xsm_op_t`
- `typedef u32 xsm_magic_t`

#### Functions

- `DEFINE_XEN_GUEST_HANDLE(xsm_op_t)`

#### Variables

- `struct xsm_operations * xsm_ops`

#### 4.186.1 Macro Definition Documentation

4.186.1.1 `#define xsm_call( fn ) 0`

4.186.1.2 `#define XSM_MAGIC 0x00000000`

#### 4.186.2 Typedef Documentation

4.186.2.1 `typedef u32 xsm_magic_t`



4.186.2.2 `typedef void xsm_op_t`

## 4.186.3 Function Documentation

4.186.3.1 `DEFINE_XEN_GUEST_HANDLE ( xsm_op_t )`

## 4.186.4 Variable Documentation

4.186.4.1 `struct xsm_operations* xsm_ops`

## 4.187 xen/xen/xsm/caas/hooks.c File Reference

```
#include <xen/init.h>
#include <xen/lib.h>
#include <xen/sched.h>
#include <xen/paging.h>
#include <xen/xmalloc.h>
#include <xsm/xsm.h>
#include <xen/spinlock.h>
#include <xen/cpumask.h>
#include <xen/errno.h>
#include <xen/guest_access.h>
#include <public/xen.h>
#include <public/physdev.h>
#include <public/platform.h>
#include <public/memory.h>
#include <public/xsm/caas_op.h>
```

## Macros

- `#define DOMID_DZ 0`
- `#define DOMID_DT 1`
- `#define CAAS_OK 0`
- `#define CAAS_COPY_IN`
- `#define CAAS_COPY_OUT`
- `#define CAAS_DEVELOP`
- `#define EXBUF_N 4`
- `#define CAAS_DENY(d)`
- `#define CAAS_ALLOWED_Z(d)`
- `#define CAAS_ALLOWED_Z_DUO(d1, d2)`
- `#define CAAS_ALLOWED_T(d)`
- `#define CAAS_ALLOWED_T_DUO(d1, d2)`
- `#define CAAS_ALLOWED_NONE(d) CAAS_DENY(d)`
- `#define CAAS_PASS(d)`

## Functions

- `integer_param ("caas_enforcing", caas_enforcing)`
- `xsm_initcall (caas_init)`

## Variables

- uint64\_t `exbuf` [`EXBUF_N`]
- int `exbuf_i` = 0
- struct xsm\_operations \* `original_ops` = NULL

### 4.187.1 Macro Definition Documentation

#### 4.187.1.1 #define CAAS\_ALLOWED\_NONE( d ) CAAS\_DENY(d)

#### 4.187.1.2 #define CAAS\_ALLOWED\_T( d )

##### Value:

```
do {
    if (d->domain_id != DOMID_DT) { CAAS_DENY(d); }
} while(0)
```

#### 4.187.1.3 #define CAAS\_ALLOWED\_T\_DUO( d1, d2 )

##### Value:

```
do {
    if (d1->domain_id != d2->domain_id)
        CAAS_ALLOWED_T(d1);
} while(0)
```

#### 4.187.1.4 #define CAAS\_ALLOWED\_Z( d )

##### Value:

```
do {
    if (d->domain_id != DOMID_DZ) { CAAS_DENY(d); }
} while(0)
```

#### 4.187.1.5 #define CAAS\_ALLOWED\_Z\_DUO( d1, d2 )

##### Value:

```
do {
    if (d1->domain_id != d2->domain_id)
        CAAS_ALLOWED_Z(d1);
} while(0)
```

#### 4.187.1.6 #define CAAS\_COPY\_IN

##### Value:

```
( \
    1UL<<CAAS_EXEMPT \
)
```

## 4.187.1.7 #define CAAS\_COPY\_OUT

**Value:**

```
( \
    1UL<<CAAS_OK \
)
```

## 4.187.1.8 #define CAAS\_DENY( d )

**Value:**

```
do {
    printk("CaaS: Denied access for %d in %s\n", d->domain_id, __func__); \
    if (!caas_enforcing) \
        printk("CaaS: not in enforcing mode!"); \
    else \
        return 1; \
} while(0)
```

## 4.187.1.9 #define CAAS\_DEVELOP

## 4.187.1.10 #define CAAS\_OK 0

## 4.187.1.11 #define CAAS\_PASS( d )

**Value:**

```
do {
    printk("CaaS: PASSED access for %d in %s\n", d->domain_id, __func__); \
    return 0; \
} while(0)
```

## 4.187.1.12 #define DOMID\_DT 1

## 4.187.1.13 #define DOMID\_DZ 0

## 4.187.1.14 #define EXBUF\_N 4

## 4.187.2 Function Documentation

## 4.187.2.1 integer\_param ( "caas\_enforcing", caas\_enforcing )

## 4.187.2.2 xsm\_initcall ( caas\_init )

## 4.187.3 Variable Documentation

## 4.187.3.1 uint64\_t exbuf[EXBUF\_N]

## 4.187.3.2 int exbuf\_i = 0

## 4.187.3.3 struct xsm\_operations\* original\_ops = NULL

## 4.188 xen/xen/xsm/dummy.c File Reference

```
#include <xen/sched.h>
#include <xsm/xsm.h>
```

## Macros

- #define [set\\_to\\_dummy\\_if\\_null](#)(ops, function)

## Functions

- void [xsm\\_fixup\\_ops](#) (struct xsm\_operations \*ops)

## Variables

- struct xsm\_operations [dummy\\_xsm\\_ops](#)

### 4.188.1 Macro Definition Documentation

#### 4.188.1.1 #define set\_to\_dummy\_if\_null( ops, function )

##### Value:

```
do {
    if ( !ops->function )
    {
        ops->function = dummy_##function;
        dprintk(XENLOG_DEBUG, "Had to override the " #function
            " security operation with the dummy one.\n");
    }
} while (0)
```

### 4.188.2 Function Documentation

#### 4.188.2.1 void xsm\_fixup\_ops ( struct xsm\_operations \* ops )

### 4.188.3 Variable Documentation

#### 4.188.3.1 struct xsm\_operations dummy\_xsm\_ops

## 4.189 xen/xen/xsm/xsm\_core.c File Reference

```
#include <xen/init.h>
#include <xen/errno.h>
#include <xen/lib.h>
#include <xen/hypercall.h>
#include <xsm/xsm.h>
```

## Functions

- long [do\\_xsm\\_op](#) (XEN\_GUEST\_HANDLE(xsm\_op\_t) op)

### 4.189.1 Function Documentation

#### 4.189.1.1 long do\_xsm\_op ( XEN\_GUEST\_HANDLE(xsm\_op\_t) op )