TDX Module ABI Specification: Updates for Non-Blocking TD Export

DRAFT – WORK IN PROGRESS

Updated Text is Highlighted in Red

Draft

June 2025

Copyright © 2025 Intel Corporation. All rights reserved.

Table of Contents

1. Updated: Data Types 4

1.1. Updated: Interface Function Completion Status 4
1.1.1. New Status Codes (to be added to the spreadsheet) 4
1.1.2. New Operand IDs (to be added to the spreadsheet) 4

1.2. Updated: TDX Module Configuration, Enumeration, Initialization and Lifecycle Types 5
1.2.1. Updated: Global-Scope (TDX Module) Metadata 5
1.2.1.1. Updated: TDX Features Enumeration 5

1.3. Updated: TD Parameter Types 5
1.3.1. Updated: CONFIG_FLAGS 5

1.4. Updated: TD Private Memory Management Data Types: Secure EPT 5
1.4.1. Updated: Secure EPT Entry Information as Returned by TDX Module Functions 6
1.4.1.1. Updated: Returned L1 Secure EPT Entry Content 6
1.4.1.2. Returned L2 Secure EPT Entry Content 6
1.4.1.3. Updated: Additional Returned Secure EPT Information 7

1.5. Updated: Migration Types 10
1.5.1. Updated: GPA List 10
1.5.1.1. Updated: Overview 10
1.5.1.2. Updated: GPA_LIST_INFO: HPA, First and Last Entries of a GPA List 11
1.5.1.3. Updated: GPA List Entry 12
1.5.1.4. Updated: GPA List Entry Details 13
1.5.1.5. TD Migration Protocol Version Compatibility 16

2. Updated: TD Metadata (Non-Memory State) 17

2.1. Updated: TD-Scope Metadata 17

3. Updated: Interface Functions 18

3.1. Updated: Host-Side (SEAMCALL) Interface Functions 18
3.1.1. Updated: TDH.EXPORT.ABORT Leaf 18
3.1.1.1. Inputs 18
3.1.1.2. Outputs 18
3.1.1.3. Leaf Function Description 18
3.1.1.4. Operands Information 19
3.1.1.5. Completion Status Codes 19
3.1.2. Updated: TDH.EXPORT.BLOCKW Leaf 21
3.1.3. Updated: TDH.EXPORT.MEM Leaf 22
3.1.3.1. Inputs 22
3.1.3.2. Outputs 23
3.1.3.3. Leaf Function Description 23
3.1.3.4. Operands Information 26
3.1.3.5. Completion Status Codes 27
3.1.4. Updated: TDH.EXPORT.PAUSE Leaf 28
3.1.4.1. Leaf Function Description 28
3.1.4.2. Completion Status Codes 28
3.1.5. Updated: TDH.EXPORT.STATE.IMMUTABLE Leaf 30
3.1.5.1. Input Operands 30
3.1.5.2. Output Operands 31
3.1.5.3. Leaf Function Description 31
3.1.5.4. Operands Information 31
3.1.5.5. Completion Status Codes 32
3.1.6. Updated: TDH.EXPORT.TRACK Leaf 34
3.1.6.1. Inputs 34
3.1.6.2. Outputs 34
3.1.6.3. Leaf Function Description 35
3.1.6.4. Operands Information 35

3.1.6.5. Completion Status Codes 36
3.1.7. Updated: TDH.EXPORT.UNBLOCKW Leaf 37
3.1.8. Updated: TDH.IMPORT.MEM Leaf 38
3.1.9. Updated: TDH.MEM.PAGE.ADD Leaf 38
3.1.10. Updated: TDH.MEM.PAGE.AUG Leaf 39
3.1.11. Updated: TDH.MEM.PAGE.DEMOTE Leaf 40
3.1.11.1. Leaf Function Description 40
3.1.11.2. Completion Status Codes 40
3.1.12. Updated: TDH.MEM.PAGE.REMOVE Leaf 41
3.1.13. Updated: TDH.MEM.RANGE.BLOCK Leaf 41
3.1.13.1. Inputs 41
3.1.13.2. Outputs 41
3.1.13.3. Leaf Function Description 42
3.1.13.4. Operands Information 42
3.1.13.5. Completion Status Codes 43
3.1.14. New: TDH.MEM.SCAN.COMP/RANGE – Common 44
3.1.14.1. GPA List-of-Lists Processing 44
3.1.15. New: TDH.MEM.SCAN.COMP Leaf 47
3.1.15.1. Inputs 47
3.1.15.2. Outputs 48
3.1.15.3. Leaf Function Description 48
3.1.15.4. Operands Information 51
3.1.15.5. Completion Status Codes 52
3.1.16. New: TDH.MEM.SCAN.CONFIG Leaf 53
3.1.16.1. Inputs 53
3.1.16.2. Outputs 54
3.1.16.3. Leaf Function Description 54
3.1.16.4. Operands Information 55
3.1.16.5. Completion Status Codes 55
3.1.17. New: TDH.MEM.SCAN.RANGE Leaf 57
3.1.17.1. Input Operands 57
3.1.17.2. Output Operands 57
3.1.17.3. Leaf Function Description 58
3.1.17.4. Operands Information 59
3.1.17.5. Completion Status Codes 60
3.1.18. New: TDH.MEM.SCAN.RESET Leaf 62
3.1.18.1. Inputs 62
3.1.18.2. Outputs 62
3.1.18.3. Leaf Function Description 62
3.1.18.4. Operands Information 62
3.1.18.5. Completion Status Codes 63
3.1.19. Updated: TDH.SYS.CONFIG Leaf 64
3.1.20. Updated: TDH.SYS.UPDATE Leaf 66
3.1.21. Updated: TDH.VP.ENTER Leaf 67

3.2. Updated: Guest-Side (TDCALL) Interface Functions 68
3.2.1. Updated: TDG.MEM.PAGE.ACCEPT Leaf 68
3.2.2. Updated: TDG.MEM.PAGE.ATTR.WR Leaf 68
3.2.3. Updated: TDG.MEM.PAGE.RELEASE Leaf 68

1. Updated: Data Types

1.1. Updated: Interface Function Completion Status

1.1.1. New Status Codes

Table 1.1: New Status Codes Definition

Completion Status Code	Description
TDX_BLOCKED_MEMORY_EXISTS	Operation failed because some of the TD's private memory is blocked. The host VMM may read BLOCKED_COUNT and PENDING_BLOCKED_COUNT using TDH.MNG.RD. The host VMM may retry the operation when no more memory is blocked.
TDX_INTERRUPTED_LIST_FULL	Operation was interrupted because the output list is full. The host VMM may resume the operation with an updated output list.
TDX_MEM_RANGE_SCAN_SUCCESS	Comprehensive scan of the current GPA range completed successfully
TDX_MEM_SCAN_SUCCESS	Comprehensive scan of the whole TD private GPA address space completed successfully
TDX_MEM_SCAN_FAILED_BLOCKED_RANGE	The comprehensive GPA space scan failed because a blocked memory range was detected
TDX_MEM_SCAN_FAILED_OTHER_THREAD	A comprehensive GPA space scan failure was detected by TDH.MEM.SCAN running on another thread
TDX_MEM_SCAN_CONFIG_REQUIRED	Comprehensive GPA space scan has not been configured by TDH.COMP.SCAN.CONFIG
TDX_MEM_SCAN_RESET_REQUIRED	The comprehensive GPA space scan state needs to be reset after a previous scan, by calling TDH.COMP.SCAN.RESET
TDX_UNEXPORTED_MEMORY_REMAINS	Operation failed because some of the TD's private memory has not been exported. The host VMM may read MEM_COUNT and MIG_COUNT using TDH.MNG.RD. The host VMM may export the remaining memory and retry the operation.
TDX_MEM_SCAN_CONIG_ALREADY_DONE	Comprehensive GPA space scan has already been configured by TDH.COMP.SCAN.CONFIG

5

1.1.2. New Operand IDs

Table 1.2: New Operand IDs Definition

Completion Status Code	Description
MEM_SCAN_CONTEXT	Memory scan context
MEM_SCAN_STATE	Memory scan internal state

1.2. Updated: TDX Module Configuration, Enumeration, Initialization and Lifecycle Types

1.2.1. Updated: Global-Scope (TDX Module) Metadata

1.2.1.1. Updated: TDX Features Enumeration

Table 1.3: Updated: TDX_FEATURES0 Definition

Bit(s)	Name	Description
…	…	…
41	NON_BLOCKING_EXPORT	The TDX module supports TD migration using non-blocking export, based on Secure EPT entries' Dirty bit.
…	…	…
43	SCAN_EXPORT_RESTORE	The TDX module supports the EXPORT_RESTORE operation of TDH.MEM.SCAN.
…	…	…

1.3. Updated: TD Parameter Types

1.3.1. Updated: CONFIG_FLAGS

CONFIG_FLAGS is a set of TD configuration flags.

Table 1.4: Updated: TD_PARAMS_STRUCT.CONFIG_FLAGS Definition

Bits	Name	Description
…	…	…
5	**TDX_CONNECT**	Enables TDX Connect for the current TD: 0: TDX Connect is disabled. 1: TDX Connect is enabled. TDX_CONNECT may not be set if ATTRIBUTES.MIGRATABLE is 1 and the TDX module has been configured for write-blocking based export. **Enumeration:** Availability of TDX_CONNECT is enumerated by TDX_FEATURES0.TDX_CONNECT (bit 6) and by CONFIG_FLAGS_FIXED0/1, readable using TDH.SYS.RD*.
6	**PAGE_RELEASE**	Enables TDG.MEM.PAGE.RELEASE for the current TD. If TDX_CONNECT above is set to 1, PAGE_RELEASE is implicitly 1. **Enumeration:** Availability of PAGE_RELEASE is enumerated by TDX_FEATURES0.PAGE_RELEASE (bit 38) and by CONFIG_FLAGS_FIXED0/1, readable using TDH.SYS.RD*.
…	…	…

1.4. Updated: TD Private Memory Management Data Types: Secure EPT

…

1.4.1. Updated: Secure EPT Entry Information as Returned by TDX Module Functions

…

1.4.1.1. *Updated: Returned L1 Secure EPT Entry Content*

The returned L1 secure EPT entry format is detailed below. It may be different that the actual Secure EPT format as maintained by the TDX module in memory.

Table 1.5: Updated: L1 Secure EPT Entry Content as Returned by TDX Interface Functions

L1 Secure EPT Entry Field						Value Returned in RCX (per Class of Entry State Returned in RDX)		
MSB	**LSB**	**Size**	**Short Name**	**Full Name**	**Enabled**	**Leaf**	**Non-Leaf**	**Free**
0	0	1	R	Read	N/A	R	R	0
1	1	1	W	Write	N/A	W	W	0
2	2	1	X / Xs	Execute	N/A	X	X	0
5	3	3	MT	Memory Type	N/A	MT	0	0
6	6	1	IPAT	Ignore PAT	N/A	IPAT	0	0
7	7	1	PS	Leaf	N/A	1	0	0
8	8	1	A	Accessed	No	0	0	0
9	9	1	D	Dirty	No	0	0	0
10	10	1	Xu	Execute (User)	No	0	0	0
11	11	1	Ignored	Ignored	N/A	0	0	0
51	12	40	HPA[51:12]	Host Physical Address [51:12]	N/A	HPA[51:12]	HPA[51:12]	0
57	57	1	VGP	Verify Guest Paging	No	0	0	0
58	58	1	PWA	Paging-Write Access	No	0	0	0
59	59	1	Ignored	Ignored	N/A	0	0	0
60	60	1	SSS	Supervisor Shadow Stack	No	0	0	0
61	61	1	SPP	Check Sub-Page Permissions	No	0	0	0
62	62	1	Ignored	Ignored	N/A	0	0	0
63	63	1	SVE	Suppress #VE	Yes	SVE	0	1

For L1 SEPT entries, the R, W and X access permission bits' values depend on the SEPT entry state:

- For leaf entries in the MAPPED and EXPORTED_DIRTY states, and non-leaf entries in the NL_MAPPED state, RWX = 111.
- For leaf entries in the BLOCKED, PENDING* and REMOVED states, non-leaf entries in the NL_BLOCKED state and FREE entries, RWX = 000.
- For leaf entries in the *BLOCKEDW* states, RWX = 101.

1.4.1.2. *Returned L2 Secure EPT Entry Content*

The returned L2 secure EPT entry format is detailed below. It may be different than the actual L2 Secure EPT format as maintained by the TDX module in memory.

Table 1.6: Updated: L2 Secure EPT Entry Content as Returned by TDX Interface Functions

L2 Secure EPT Entry Field						Value Returned in RCX (per Class of Entry State Returned in RDX)		
MSB	**LSB**	**Size**	**Short Name**	**Full Name**	**Enabled**	**Leaf**	**Non-Leaf**	**Free**
0	0	1	R	Read	N/A	R	R	0
1	1	1	W	Write	N/A	W	W	0
2	2	1	Xs	Execute	N/A	Xs	Xs	0
5	3	3	MT	Memory Type	N/A	MT	0	0
6	6	1	IPAT	Ignore PAT	N/A	IPAT	0	0
7	7	1	PS	Leaf	N/A	1	0	0
8	8	1	A	Accessed	No	0	0	0
9	9	1	D	Dirty	No	0	0	0
10	10	1	Xu	Execute (User)	No	Xu	Xu	0
11	11	1	Ignored	Ignored	N/A	0	0	0
51	12	40	HPA[51:12]	Host Physical Address [51:12]	N/A	HPA[51:12]	HPA[51:12]	0
57	57	1	VGP	Verify Guest Paging	No	0 / VGP	0	0
58	58	1	PWA	Paging-Write Access	No	0 / PWA	0	0
59	59	1	Ignored	Ignored	N/A	0	0	0
60	60	1	SSS	Supervisor Shadow Stack	No	0 / SSS	0	0
61	61	1	SPP	Check Sub-Page Permissions	No	0	0	0
62	62	1	Ignored	Ignored	N/A	0	0	0
63	63	1	SVE	Suppress #VE	Yes	SVE	0	1

For L2 SEPT entries, the R, W, Xs and Xu access permission bits' values depend on the L2 SEPT entry state and on the TD's ATTRIBUTE.DEBUG value:

- For leaf entries in the L2_MAPPED state:
  - If ATTRIBUTES.DEBUG is 0, then RWXsXu = 1111 and VGP, PWA and SSS are cleared to 0.
  - Else, the real values of RWXsXu and of VGP, PWA and SSS are returned.
- For leaf entries in the L2_BLOCKED state:
  - If ATTRIBUTES.DEBUG is 0, then RWXsXu = 0000 and VGP, PWA and SSS are cleared to 0.
  - Else, then RWXsXu = 0000 and GP, PWA and SSS are returned.
- For non-leaf entries in the L2_NL_MAPPED state, RWXsXu = 1111.
- For non-leaf entries in the L2_NL_BLOCKED state and L2_FREE entries, RWXsXu = 0000.

1.4.1.3. *Updated: Additional Returned Secure EPT Information*

…

Table 1.7: Updated: Secure L1 EPT Entry TDX State Returned by TDX Interface Functions

L1 SEPT Entry State Name	Public State Number	Class: Free, Non-Leaf or Leaf	Description	TDX_FEATURES0 Enumeration
FREE	0	Free	L1 Secure EPT entry does not map a GPA range.	N/A
REMOVED	5	Free	L1 Secure EPT entry is of a removed page	TD_MIGRATION

L1 SEPT Entry State Name	Public State Number	Class: Free, Non-Leaf or Leaf	Description	TDX_FEATURES0 Enumeration
NL_MAPPED	132	Non-Leaf	L1 Secure EPT entry maps a private GPA range which is accessible by the guest TD.	N/A
NL_BLOCKED	129	Non-Leaf	L1 Secure EPT entry maps a private GPA range, but new address translations to that range are blocked.	N/A
MAPPED	4	Leaf	L1 Secure EPT entry maps a private GPA page which is accessible by the guest TD.	N/A
BLOCKED	1	Leaf	L1 Secure EPT entry maps a private GPA page but new address translations to that range are blocked.	N/A
REMOVE_IN_PROGRESS	6	Free	L1 Secure EPT entry maps a private page that is being removed (TDH.MEM.PAGE.REMOVE has been interrupted).	ACT
BLOCKEDW	8	Leaf	Write-Blocking Export: L1 Secure EPT entry maps a private GPA page, but new address translations for write operations to that range are blocked.	TD_MIGRATION
EXPORTED_BLOCKEDW	9	Leaf	Write-Blocking Export: L1 Secure EPT entry maps a private page that has been blocked for writing and exported.	TD_MIGRATION or S4
EXPORTED_DIRTY	11	Leaf	Write-Blocking Export: L1 Secure EPT entry maps a private page that was exported, but is not blocked for writing and its content and/or attributes may have since been modified.	TD_MIGRATION
EXPORTED_DIRTY_ BLOCKEDW	12	Leaf	Write-Blocking Export: L1 Secure EPT entry maps a private page that was previously exported, its content and/or attributes may have since been modified and then it was blocked for writing.	TD_MIGRATION
EXPORTED	24	Leaf	Non-Blocking Export: The page has been exported. This state indicates that any change in the page content or attributes would require a re-export.	N/A
EXPORTED_MODIFIED	25	Leaf	Non-Blocking Export: The page was exported and later either identified as dirty based on the Dirty bit (which was cleared by the same operation that checked its value) or some memory management operation changed the page attributes or state. This state indicates that the page must be re-exported as a REMIGRATE operation.	NON_BLOCKING_ EXPORT

L1 SEPT Entry State Name	Public State Number	Class: Free, Non-Leaf or Leaf	Description	TDX_FEATURES0 Enumeration
EXPORTED_BLOCKED	26	Leaf	Non-Blocking Export: The page was exported and later blocked by TDH.MEM.RANEG.BLOCK. This state indicates that the page must be re-exported as a CANCEL operation.	NON_BLOCKING_ EXPORT
EXPORTED_REMOVED	27	Free	Non-Blocking Export: The page was exported and later removed by TDH.MEM.PAGE.REMOVE. This state indicates that the page should be re-exported, as a CANCEL operation.	NON_BLOCKING_ EXPORT
EXPORTED_REMOVE_ IN_PROGRESS	28	Free	Non-Blocking Export: The page was exported and later partially removed by TDH.MEM.PAGE.REMOVE. This state indicates that the page should be re-exported, as a CANCEL operation.	NON_BLOCKING_ EXPORT and ACT
PENDING	2	Leaf	L1 Secure EPT entry maps a 4KB or a 2MB page that has been dynamically added to the guest TD using TDH.MEM.PAGE.AUG and is pending acceptance by the guest TD using TDG.MEM.PAGE.ACCEPT. This page is not yet accessible by the guest TD.	N/A
PENDING_BLOCKED	3	Leaf	L1 Secure EPT entry is both pending and blocked.	N/A
PENDING_BLOCKEDW	16	Leaf	Write-Blocking Export: L1 Secure EPT entry is both pending and blocked for writing.	TD_MIGRATION
PENDING_EXPORTED_ BLOCKEDW	17	Leaf	Write-Blocking Export: L1 Secure EPT entry is both pending and exported.	TD_MIGRATION or S4
PENDING_EXPORTED_ DIRTY	19	Leaf	Write-Blocking Export: L1 Secure EPT entry is both pending and exported, and is not blocked for writing.	TD_MIGRATION
PENDING_EXPORTED_ DIRTY_BLOCKEDW	20	Leaf	L1 Secure EPT entry is both pending and exported, and is blocked for writing.	TD_MIGRATION
PENDING_EXPORTED	29	Leaf	Non-Blocking Export: The page has been exported. This state indicates that any change in the page attributes would require a re-export.	NON_BLOCKING_ EXPORT
PENDING_EXPORTED_ MODIFIED	30	Leaf	Non-Blocking Export: The page was exported and later identified as dirty based on the Dirty bit (which was atomically cleared by the same operation that checked its value). This state indicates that the page must be re-exported as a REMIGRATE operation.	NON_BLOCKING_ EXPORT

L1 SEPT Entry State Name	Public State Number	Class: Free, Non-Leaf or Leaf	Description	TDX_FEATURES0 Enumeration
PENDING_EXPORTED_BLOCKED	31	Leaf	Non-Blocking Export: The page was exported and later blocked by TDH.MEM.RANEG.BLOCK. This state indicates that the page must be re-exported as a CANCEL operation.	NON_BLOCKING_EXPORT
MMIO_MAPPED	32	Leaf	L1 Secure EPT entry maps a private MMIO page which is accessible by the guest TD.	TDX_CONNECT
MMIO_BLOCKED	33	Leaf	L1 Secure EPT entry maps a private MMIO page, but new address translations to that page are blocked.	TDX_CONNECT
MMIO_PENDING	34	Leaf	L1 Secure EPT entry maps a 4KB, 2MB or 1GB MMIO page that is pending acceptance by the guest TD using TDG.MMIO.ACCEPT. This page is not yet accessible by the guest TD.	TDX_CONNECT

Table 1.8: Updated: Secure L2 EPT Entry TDX State Returned by TDX Interface Functions

L2 SEPT Entry Sta3te Name	Public State Number	Class: Free, Leaf or Non-Leaf	Description	TDX_FEATURES Enumeration
L2_FREE	64	Free	L2 Secure EPT entry does not map a GPA range.	TD_PARTITIONING
L2_NL_MAPPED	196	Non-Leaf	L2 Secure EPT entry maps a private GPA range which is accessible by the L2 VM.	TD_PARTITIONING
L2_NL_BLOCKED	193	Non-Leaf	L2 Secure EPT entry maps a private GPA range, but new address translations to that range are blocked.	TD_PARTITIONING
L2_MAPPED	68	Leaf	L2 Secure EPT entry maps a private GPA page which is accessible by the L2 VM.	TD_PARTITIONING
L2_BLOCKED	65	Leaf	L2 Secure EPT entry maps a private GPA page but new address translations to that range are blocked.	TD_PARTITIONING
L2_MMIO_MAPPED	96	Leaf	L2 Secure EPT entry maps a private MMIO page which is accessible by the L2 VM.	TD_PARTITIONING and TDX_CONNECT
L2_MMIO_BLOCKED	97	Leaf	L2 Secure EPT entry maps a private MMIO page, but new address translations to that page are blocked.	TD_PARTITIONING and TDX_CONNECT

…

1.5. Updated: Migration Types

1.5.1. Updated: GPA List

1.5.1.1. Updated: Overview

A GPA list specifies a list of 4KB-aligned GPAs, each with associated attributes, required operation and status. ~~to be migrated by TDH.EXPORT.MEM and TDH.IMPORT.MEM, blocked for writing by TDH.EXPORT.BLOCKW or reset to their~~

~~original SEPT entry state by TDH.EXPORT.RESTORE.~~ A single GPA list may have up to 512 entries, is contained in a single 4KB page and must be aligned on 4KB. The GPA list may contain null entries, as indicated by the OPERATION field's value set to 0 (NOP).

Interface functions that process up to 512 GPAs, such as TDH.EXPORT.MEM, use a single GPA list page. The GPA list is specified by GPA_LIST_INFO, which contains the HPA of the GPA list page and the index of the first entry and last entry to be processed.

Interface functions that process up to 512^2 GPAs, such as TDH.MEM.SCAN, use multiple GPA list pages. The list of pages is specified by a page of GPA_LIST_INFO entries. The GPA list info page is specified by a PAGE_LIST_INFO, which contains the HPA of the GPA list info page and the index of the first entry and last entry to be processed.

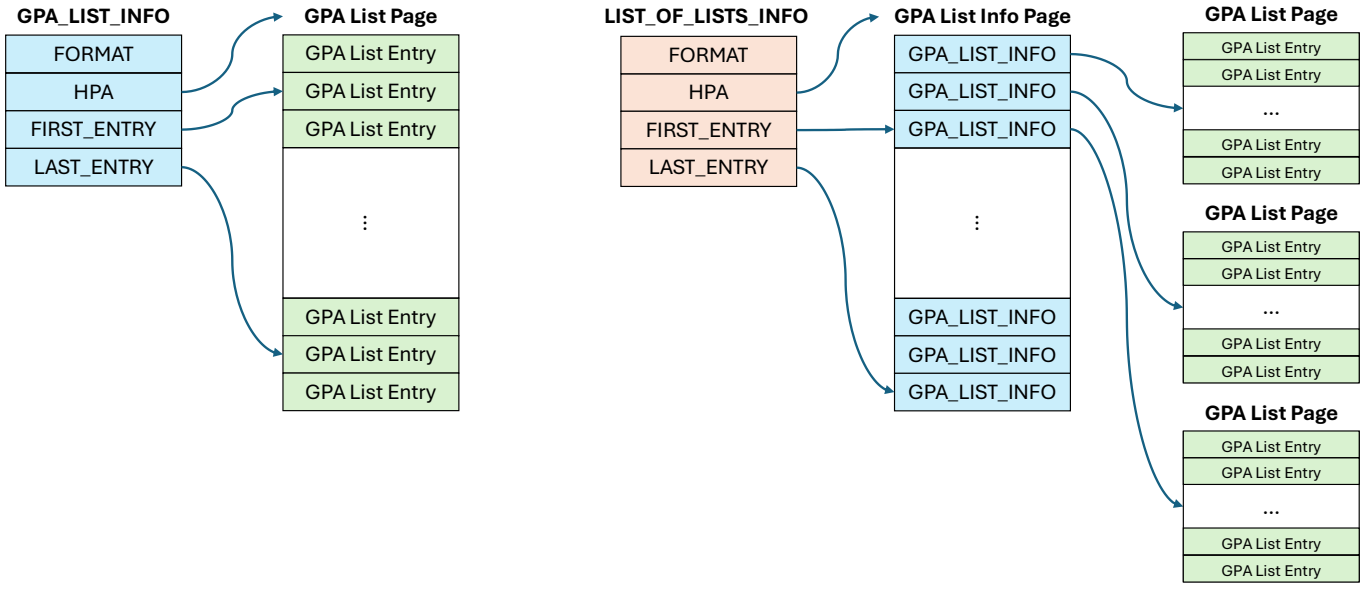

Figure 1.1: New: GPA List and List of Lists

1.5.1.2. *Updated: GPA_LIST_INFO: HPA, First and Last Entries of a GPA List*

GPA_LIST_INFO is a 64b structure used as a GPR input and output operand of multiple migration interface functions, e.g., TDH.EXPORT.MEM. It provides the HPA of the GPA list page in shared memory, and the index of the first entry and last entries to be processed. When used with some interface functions, it may provide list-of-lists information.

Table 1.9: GPA_LIST_INFO

Bits	Name	Description		
2:0	**FORMAT**	GPA list format		
		Value	**Name**	**Description**
		0	GPA_ONLY	A GPA list page is provided
		1	GPA_AND_L2_ATTR	GPA list and L2 page attributes list pages are provided. This format is only used by TDH.EXPORT.MEM and TDH.IMPORT.MEM. It is mandatory for migrating partitioned TDs (which contain one or more L2 VMs). TDX module support of this feature is enumerated by TDX_FEATURES0.PARTITIONED_TD_MIGRATION, readable by TDH.SYS.RD* (see 1.2.1.1).
		2	LIST_OF_LISTS	A list-of-lists page is provided
		Other	RESERVED	Reserved
11:3	**FIRST_ENTRY**	Index of the first entry of the list to be processed		

Bits	Name	Description
51:12	**HPA**	Bits 51:12 of the host physical address (including HKID) of the GPA list page, which must be a shared HPA
54:52	**RESERVED**	Reserved: must be 0
63:55	**LAST_ENTRY**	Index of the last entry in the GPA list

The following special values indicate an empty list. This is used as the output of TDH.MEM.SCAN.COMP/RANGE.

- FIRST_ENTRY set to all-1 (511)
- LAST_ENTRY set to 0

1.5.1.3. *Updated: GPA List Entry*

Table 1.10 below shows the format of a GPA list entry as used. The GPA list entry format is designed so that the output of TDH.EXPORT.BLOCKW and TDH.MEM.SCAN can be used directly with TDH.EXPORT.MEM, and the output of TDH.EXPORT.MEM can be used directly with TDH.IMPORT.MEM.

Table 1.10: GPA List Entry Definition

Bit(s)	Size	Name	Description	TDH.MEM.SCAN	TDH.EXPORT.BLOCKW		TDH.EXPORT.MEM		TDH.IMPORT.MEM		TDH.EXPORT.RESTORE	
				Out	In	Out	In	Out	In	Out	In	Out
1:0	2	LEVEL	Mapping level (size)	0 (4KB), 1 (2MB) or 2 (1GB)	Must be 0 (4KB)	Unmod.	Must be 0 (4KB)	Unmod.	Must be 0 (4KB)	Unmod.	Must be 0 (4KB)	Unmod.
2	1	PENDING	See below	Yes	Ignored	Unmod.	Ignored	Yes	Yes	Unmod.	Ignored	Unmod.
4:3	2	STATE	See below	Yes	Must be 0	Unmod.	Ignored	0	Must be 0	Unmod.	Must be 0	Unmod.
6:5	2	RESERVED	Reserved	0	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.
9:7	3	L2_MAP	See below	0	Ignored	Unmod.	Ignored	Yes	Yes	Unmod.	Ignore	Unmod.
11:10	2	MIG_TYPE	See below	0	Yes	Unmod.	Yes	Unmod.	Yes	Unmod.	Yes	Unmod.
51:12	40	GPA	Guest Physical Address bits 51:12	Yes	Yes	Unmod.	Yes	Unmod.	Yes	Unmod.	Yes	Unmod.
53:52	2	OPERATION	See below	Yes[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
55:54	2	RESERVED	Reserved	0	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.
60:56	5	STATUS	See below	Yes[2]	Ignored	Yes	Ignored	Yes	Ignored	Yes	Ignored	Yes
63:61	3	RESERVED	Reserved	0	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.	Must be 0	Unmod.

[1] OPERATION value for TDH.MEM.SCAN.RANGE(DSCAN) and TDH.MEM.SCAN.COMP(DCHECK) is always MIGRATE.

[2] STATUS value for TDH.MEM.SCAN.RANGE(DSCAN) and TDH.MEM.SCAN.COMP(DCHECK) is always SUCCESS.

1.5.1.4. *Updated: GPA List Entry Details*

1.5.1.4.1. Updated: LEVEL

LEVEL specifies the GPA mapping size for this entry, as shown in the table below.

Table 1.11: LEVEL Definition

Level	GPA Mapping Size
0	4KB
1	2MB
2	1GB
3	Reserved

For TDH.EXPORT.BLOCKW, TDH.EXPORT.MEM, TDH.EXPORT.RESTORE and TDH.IMPORT.MEM only level 0 (4KB) is supported.

1.5.1.4.2. PENDING

Indicates that the page is Pending. Note that the actual SEPT entry state may be one of multiple PENDING* states.

~~**Table 1.12: PENDING Values Definition**~~

~~Value~~	~~Name~~	~~Description~~
~~0~~	~~MAPPED~~	~~SEPT entry is MAPPED~~
~~1~~	~~PENDING~~	~~SEPT entry is PENDING~~

1.5.1.4.3. New: STATE

Provides a hint about the page state.

Table 1.13: New: STATE Values Definition for TDH.MEM.SCAN*

Value	Output	
	Name	Description
0	NOT_EXPORTED	Page is a candidate for initial export.
1	EXPORTED_MODIFIED	Page is a candidate for re-export due to updated content or attributes.
2	EXPORTED_BLOCKED	Page is a candidate for re-export due to being blocked. Blocking is an interim state followed by some other memory management operation; the host VMM may decide to defer the re-export.
3	EXPORTED_REMOVED	Page is a candidate for re-export due to being removed.

1.5.1.4.4. L2_MAP

A bitmap which indicates whether the page is mapped in one or more L2 VMs. This field is provided as part of the GPA list entry to enable the host VMM to prepare L2 SEPT pages before invoking TDH.IMPORT.MEM.

1.5.1.4.5. Updated: OPERATION / STATE

The following tables describe the meaning of OPERATION, as used for each applicable interface function. Note that the OPERATION definitions for TDH.EXPORT.BLOCKW, TDH.EXPORT.MEM and TDH.IMPORT.MEM are designed to be compatible, so that the same GPA list can be used for all of them.

OPERATION Values for TDH.EXPORT.BLOCKW

Table 1.14: OPERATION Values Definition for TDH.EXPORT.BLOCKW

Value	Input		Output	
	Name	Description	Name	Description
0	NOP	No operation	NOP	Not blocked for writing
1	BLOCKW	Block for writing	BLOCKW	Blocked for writing
2	NOP	No operation	NOP	Not blocked for writing
3	BLOCKW	Block for writing	BLOCKW	Blocked for writing

New: OPERATION Values for TDH.MEM.SCAN*

Table 1.15: New: OPERATION / STATE Values Definition for TDH.MEM.SCAN*

Value	Output	
	Name	Description
0	RESERVED	Reserved
1	MIGRATE	Export
2	RESERVED	Reserved
3	RESERVED	Reserved

OPERATION Values for TDH.EXPORT.MEM (For Write Blocking Based Export)

Note that on input, the MIGRATE operation is represented by two possible values. This is done for direct compatibility with the TDH.EXPORT.BLOCKW output.

Table 1.16: OPERATION Values Definition for TDH.EXPORT.MEM (For Write Blocking Based Export)

Value	Input		Output	
	Name	Description	Name	Description
0	NOP	No operation	NOP	Not exported
1	MIGRATE	Export	MIGRATE	Initial export during this migration session or following a CANCEL
2	CANCEL	Cancel previous export	CANCEL	Cancellation of a previous export Not applicable for S4 hibernation.
3	MIGRATE	Export	REMIGRATE	Re-export of updated content or attributes

OPERATION Values for TDH.EXPORT.MEM (For Non-Blocking Export)

Note that on input, the MIGRATE operation is represented by two possible values. This is done for direct compatibility with the TDH.MEM.SCAN(DSCAN/DCHECK) output.

Table 1.17: New: OPERATION Values Definition for TDH.EXPORT.MEM (For Non-Blocking Export)

Value	Input		Output	
	Name	Description	Name	Description
0	NOP	No operation	NOP	Not exported

Value	Input		Output	
	Name	Description	Name	Description
1	MIGRATE	Export	MIGRATE	Initial export during this migration session or following a CANCEL
2	RESERVED	Reserved	CANCEL	Cancellation of a previous export Not applicable for S4 hibernation.
3	RESERVED	Reserved	REMIGRATE	Re-export of updated content or attributes

OPERATION Values for TDH.IMPORT.MEM

Table 1.18: OPERATION Values Definition for TDH.IMPORT.MEM

Value	Input		Output	
	Name	Description	Name	Description
0	NOP	No operation	NOP	Not imported
1	MIGRATE	Initial import during this migration session or following a CANCEL	MIGRATE	Imported
2	CANCEL	Cancel previous import	CANCEL	Removed previous import Not applicable for S4 resumption.
3	REMIGRATE	Re-import of updated page content or attributes	REMIGRATE	Imported Not applicable for S4 resumption.

OPERATION Values for TDH.EXPORT.RESTORE

Table 1.19: OPERATION Values Definition for TDH.EXPORT.RESTORE

Value	Input		Output	
	Name	Description	Name	Description
0	NOP	No operation	NOP	Not restored
1	RESTORE	Restore SEPT entry to non-migration state	RESTORE	Restored
2	NOP	Reserved	NOP	Not restored
3	RESTORE	Restore SEPT entry to non-migration state	RESTORE	Restored

1.5.1.4.6. MIG_TYPE

Table 1.20: MIG_TYPE Values Definition

Value	Name	Description
0	PAGE_4K	4KB private memory page
Other	RESERVED	Reserved for future types

1.5.1.4.7. Updated: STATUS

Table 1.21: STATUS Values Definition

Value	Name	Description
0	SUCCESS	GPA list entry was processed successfully
1	SKIPPED	GPA list entry was skipped because NOP was requested
2	SEPT_WALK_FAILED	Secure EPT walk failed for the requested GPA
3	SEPT_ENTRY_BUSY_HOST_PRIORITY	Secure EPT entry was busy. The host VMM should retry the operation until successful.
4	SEPT_ENTRY_STATE_INCORRECT	Secure EPT entry state was incorrect for the requested operation and the TD's OP_STATE
5	TLB_TRACKING_NOT_DONE	TLB tracking was not done for the requested GPA
6	OP_STATE_INCORRECT	The TD's OP_STATE was incorrect for the requested operation and Secure EPT entry state
7	MIGRATED_IN_CURRENT_EPOCH	Requested GPA has already been migrated during the current migration epoch
8	MIG_BUFFER_NOT_AVAILABLE	Required migration buffer was not provided
9	NEW_PAGE_NOT_AVAILABLE	Required new TD page was not provided
10	INVALID_PAGE_MAC	Page MAC was invalid
11	DISALLOWED_IMPORT_OVER_REMOVED	Page import over a removed page is not allowed
12	TD_PAGE_BUSY_HOST_PRIORITY	TD page was busy. The host VMM should retry the operation until successful.
13	L2_SEPT_WALK_FAILED	L2 Secure EPT walk failed for the requested GPA
14	ATTR_LIST_ENTRY_INVALID	The L2 attributes list entry is invalid
15	GPA_LIST_ENTRY_INVALID	The GPA list entry is invalid
16	INVALID_MIGRATION_BUFFER_HPA	The provided migration buffer HPA is not valid
17	PAGE_DIRTY	Page not exported because the SEPT entry's Dirty bit was set
31-18	Reserved	Reserved

1.5.1.5. *TD Migration Protocol Version Compatibility*

The table below specifies the TD migration protocol versions for which the above GPA List definition is appliable.

Table 1.22: GPA List Compatibility with TD Migration Versions

TD Migration Version	Minimum	Maximum
Export version	0	0
Import version	0	0

2. Updated: TD Metadata (Non-Memory State)

2.1. Updated: TD-Scope Metadata

3. Updated: Interface Functions

3.1. Updated: Host-Side (SEAMCALL) Interface Functions

3.1.1. Updated: TDH.EXPORT.ABORT Leaf

TDH.EXPORT.ABORT aborts an export session and allows the source TD to resume normal operation, depending on export state and an abort token received from the destination platform.

3.1.1.1. *Inputs*

Table 3.1: TDH.EXPORT.ABORT Input Operands Definition

Operand	Description		
RAX	SEAMCALL instruction leaf number and version, see **Error! Reference source not found.**		
	Bits	**Field**	**Description**
	15:0	Leaf Number	Selects the SEAMCALL interface function: 64
	23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
	63:24	Reserved	Must be 0
RCX	HPA of the source TD's TDR page (HKID bits must be 0)		
R8	If an abort token is available, R8 provides the HPA and size of memory of an MBMD structure in memory, as described below. Otherwise, R8's value must be 0.		
	Bits	**Name**	**Description**
	51:0	HPA	Bits 51:0 of the host physical address (including HKID bits)
	63:52	Size	Size of the memory buffer containing MBMD, in bytes
R10	Migration stream index:		
	Bits	**Name**	**Description**
	15:0	MIGS_INDEX	Migration stream index – must be 0
	63:16	RESERVED	Reserved: must be 0

3.1.1.2. *Outputs*

Table 3.2: TDH.EXPORT.ABORT Output Operands Definition

Operand	Description
RAX	SEAMCALL instruction return code, see **Error! Reference source not found.**
Other	Unmodified

3.1.1.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

3.1.1.3.1. Overview

TDH.EXPORT.ABORT aborts an export session. If successful, i.e., the target TD does not run, the source TD becomes runnable. If called during the out-of-order phase, an abort token received from the destination platform is required.

If the TDX module is configured for non-blocking export, TDH.EXPORT.ABORT resets the internal state held by the TDX module for comprehensive memory scans of the specified TD's GPA address space.

3.1.1.3.2. Enumeration

Availability of TDH.EXPORT.ABORT is enumerated by TDX_FEATURES0.TD_MIGRATION (bit 0), readable by TDH.SYS.RD* (see 1.2.1.1). If not supported, calling TDH.EXPORT.ABORT returns a TDX_OPERAND_INVALID(RAX) status.

3.1.1.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.3: TDH.EXPORT.ABORT Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	HPA	MBMD buffer	MBMD	R	Shared	128B	None	None	None
Explicit	R10	Index	Mig. Stream context	Mig. Stream context	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	Migration context	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	Comprehensive scan state	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A

3.1.1.5. *Completion Status Codes*

Table 3.4: TDH.EXPORT.ABORT Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_INCORRECT_MBMD_MAC	
TDX_INVALID_MBMD	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	Operation is successful
TDX_SYS_NOT_READY	

Completion Status Code	Description
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.2. Updated: TDH.EXPORT.BLOCKW Leaf

…

Leaf Function Description

…

Enumeration: Availability of TDH.EXPORT.BLOCKW is enumerated by TDX_FEATURES0.TD_MIGRATION (bit 0), readable by TDH.SYS.RD* (see 1.2.1.1). If not supported, calling TDH.EXPORT.BLOCKW returns a TDX_OPERAND_INVALID(RAX) status.

Export Mode: TDH.EXPORT.BLOCKW is only available if the TDX module is configured for write-blocking based export. If not available, calling TDH.EXPORT.BLOCKW returns a TDX_OPERAND_INVALID(RAX) status.

List Entry Error: If a page can't be blocked for writing, TDH.EXPORT.BLOCKW marks its GPA list entry as unsuccessful. List processing is not aborted; it continues to the next entry, if applicable. The return status in RAX indicates the number of such cases encountered during operation in the lower 32 bits.

Interruptibility: TDH.EXPORT.BLOCKW is interruptible. If a pending interrupt is detected during operation, TDH.EXPORT.BLOCKW returns with a TDX_INTERRUPTED_RESUMABLE status in RAX. RCX is updated with the next list entry index to process, so the host VMM may re-invoke TDH.EXPORT.BLOCKW immediately after handling the interrupt.

3.1.3. Updated: TDH.EXPORT.MEM Leaf

TDH.EXPORT.MEM exports a list of TD private pages contents and/or cancellation requests and prepares a migration bundle in shared memory.

3.1.3.1. *Inputs*

Table 3.5: TDH.EXPORT.MEM Input Operands Definition

Operand	Name	Description		
RAX	Leaf and Version	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RCX	GPA_LIST_INFO	HPA of a GPA list page in shared memory, and first and last entries to process, as defined in 1.5.1 On a new invocation, FIRST_ENTRY must be 0. On a resumed invocation, FIRST_ENTRY must be the index of the next GPA list entry to export.		
RDX	TDR	HPA of the source TD's TDR page (HKID bits must be 0)		
R8	MBMD	HPA and size of memory of a memory buffer to use for MBMD:		
		Bits	**Name**	**Description**
		51:0	HPA	Bits 51:0 of the host physical address (including HKID bits)
		63:52	Size	Size of the memory buffer containing MBMD, in bytes
R9	MIG_BUFF_LIST	HPA (including HKID bits) of a migration buffer list in shared memory, corresponding to the GPA list pointed by RCX – see [ref].		
R10	MIG_STREAM	Migration stream and resume flag:		
		Bits	**Name**	**Description**
		15:0	MIGS_INDEX	Migration stream index If N migration streams have been created by TDH.MIG.STREAM.CREATE, then MIGS_INDEX must be lower than N-1.
		62:16	RESERVED	Reserved: must be 0
		63	RESUME	0: This is a new invocation 1: This is resumption of a previously interrupted operation
R11	MAC_LIST_0	HPA (including HKID bits) of a MAC list in shared memory, corresponding to the first 256 entries of the GPA list pointed by RCX – see [ref]. If GPA_LIST_INFO.FIRST_ENTRY >= 256, then MAC_LIST_0 is ignored.		

Operand	Name	Description
R12	MAC_LIST_1	HPA (including HKID bits) of a MAC list in shared memory, corresponding to the last 256 entries of the GPA list pointed by RCX – see [ref]. If GPA_LIST_INFO.LAST_ENTRY < 256, then MAC_LIST_1 is ignored.
R14	ATTRIB_LIST	If GPA_LIST_INFO.FORMAT is GPA_AND_L2_ATTR, then R14 contains the HPA (including HKID bits) of a page attributes list in shared memory – see [ref]. Else, R14 is ignored. An ATTRIB_LIST is mandatory for exporting partitioned TDs (which contain one or more L2 VMs). **Enumeration:** Support of this feature is enumerated by TDX_FEATURES0.PARTITIONED_TD_MIGRATION, readable by TDH.SYS.RD* (see 1.2.1.1).

3.1.3.2. *Outputs*

Table 3.6: TDH.EXPORT.MEM Output Operands Definition

Operand	Name	Description
RAX	STATUS	SEAMCALL instruction return code, see [ref]
RCX	GPA_LIST_INFO	Same as the input value, except that FIRST_ENTRY is updated to the index of the next entry to be processed. If all entries have been processed, FIRST_ENTRY is updated to (LAST_ENTRY + 1) Modulo 512.
RDX	NUM_EXPORTED	If TDH.EXPORT.MEM is successful, RDX returns the number of exported 4KB migration buffers, including: • The GPA list page • Attributes list page (if GPA_LIST_INFO.FORMAT is GPA_AND_L2_ATTR) • One or two MAC pages (depending on GPA_LIST_INFO.FIRST_ENTRY and GPA_LIST_INFO.LAST_ENTRY • Up to 512 encrypted memory pages If TDH.EXPORT.MEM is not successful, RDX is unmodified.
AVX, AVX2 and AVX512 state		May be reset to the architectural RESET state
Other		Unmodified

3.1.3.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

3.1.3.3.1. Overview

TDH.EXPORT.MEM exports a list of up to 512 TD private 4KB pages as a migration bundle, which includes an MBMD, set of 4KB pages encrypted with the migration session key, a 4KB page containing the GPA list, an optional 4KB page containing attributes list, and two 4KB pages containing page MACs.

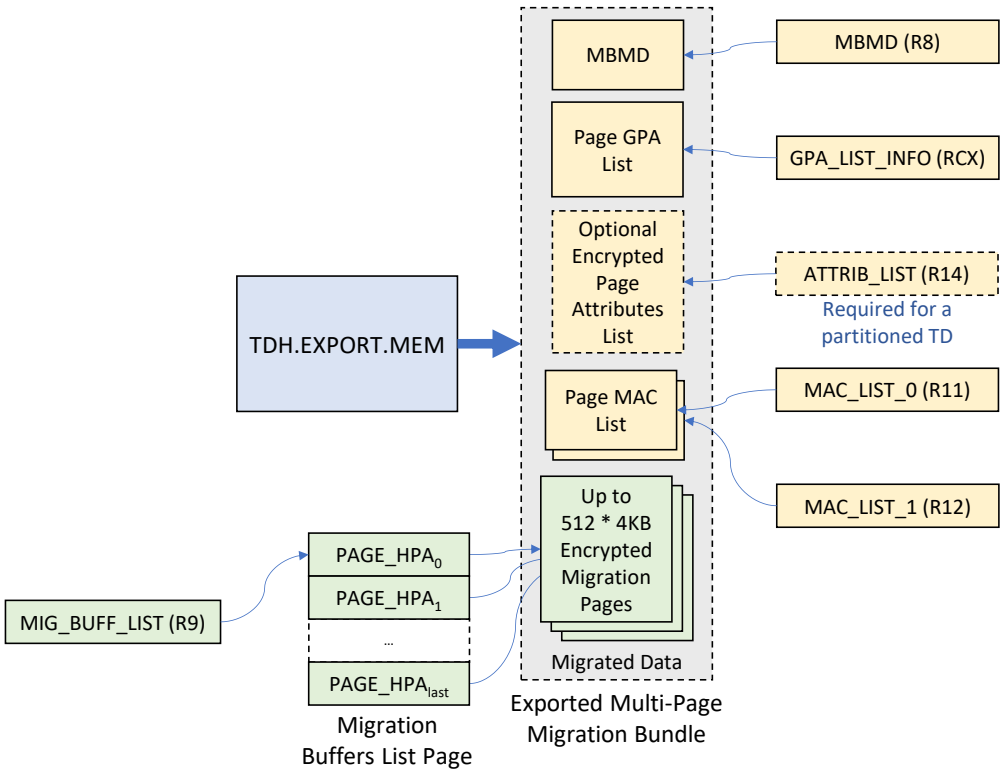

Figure 3.1: TDH.IMPORT.MEM Inputs and Outputs

3.1.3.3.2. Enumeration

Availability of TDH.EXPORT.MEM is enumerated by either TDX_FEATURES0.TD_MIGRATION (bit 0) or TDX_FEATURES0.S4 (bit 13), readable by TDH.SYS.RD* (see 1.2.1.1), being set to 1. If not supported, calling TDH.EXPORT.MEM returns a TDX_OPERAND_INVALID(RAX) status.

TDX_FEATURES0.PARTITIONED_TD_MIGRATION (bit 21) enumerates TDX module support of migrating partitioned TDs (which contain one or more L2 VMs).

Interruption due to yielding to concurrent functions that try to acquire an exclusive lock on the SEPT trees (see below) is supported if TDX_FEATURES0.NON_BLOCKING_EXPORT (bit 41) is set to 1.

3.1.3.3.3. GPA List

A GPA list is provided as an input and is updated by TDH.EXPORT.MEM. The GPA list format is described in 1.5.1. Only 4KB pages are supported.

For write-blocking based export, the requested operation for each page in the list may be one of the following:

MIGRATE: Export the page. TDH.EXPORT.MEM decides, based on the SEPT entry state, whether the operation is an export or a re-export of a previously exported page. Note that the MIGRATE operation is represented by two possible values.

CANCEL: Cancel a previous page export.

NOP: No operation.

For non-blocking export, the requested operation for each page in the list may be one of the following:

MIGRATE: Export the page. TDH.EXPORT.MEM decides, based on the SEPT entry state, whether the operation is export, re-export of a previously exported page, or cancellation of a previous export. Note that the MIGRATE operation is represented by two possible values.

NOP: No operation.

TDH.EXPORT.MEM updates the GPA list with the actual OPERATION code (MIGRATE, REMIGRATE, CANCEL or NOP) and STATUS; the host VMM is expected to send the GPA list as part of the migration bundle, to be imported on the destination platform by TDH.IMPORT.MEM.

3.1.3.3.4. S4 Hibernation

If TDH.EXPORT.MEM is called as part of an S4 hibernation, it only supports write-blocking based export (though no write blocking is actually required) and the out-of-order export phase. As a result, the GPA list may not contain a CANCEL operation. In addition, blocking and TLB tracking is not required.

3.1.3.3.5. Migration Buffers List

A list of 4KB page buffers is provided as an input and is updated by TDH.EXPORT.MEM. In case no data is exported (PENDING page, page cancellation or some state error) TDH.EXPORT.MEM marks the applicable list entry as invalid.

3.1.3.3.6. Write-Blocking and TLB Tracking (Write-Blocking based Export)

The following applies if the TDX module is configured for write-blocking based export:

If the TD may be running, the exported pages must be blocked for writing by TDH.EXPORT.BLOCKW and TLB tracked (TDH.MEM.TRACK followed by IPI to all the LPs running the TD VCPUs) to be exported. Unlike memory management operations such as TDH.MEM.PAGE.REMOVE, the TLB tracking is not page-specific; it should be done after the last TDH.EXPORT.BLOCKW of any page has been called for the current export round.

Else (e.g., the TD has been paused for export), no blocking and tracking is required.

3.1.3.3.7. Scanning and TLB Tracking (Non-Blocking Export)

The following applies if the TDX module is configured for non-blocking export:

If the TD may be running, the exported pages must have been scanned by TDH.MEM.SCAN(DSCAN) and TLB tracked to be exported. Unlike write-blocking based export, TLB tracking is page-specific, allowing TDH.MEM.SCAN(DSCAN) to run concurrently with TDH.EXPORT.MEM.

Else (e.g., the TD has been paused for export), no tracking is required.

3.1.3.3.8. Page-Specific Errors

If a page can't be exported for a reason that is specific to that page, TDH.EXPORT.MEM marks its GPA list entry as unsuccessful, but does not abort. It continues to the next entry, if applicable. Eventually, if TDH.EXPORT.MEM completes successfully, the TDX_SUCCESS return status indicates the number of such cases encountered during operation in RAX[31:0].

3.1.3.3.9. Concurrency

TDH.EXPORT.MEM may be called concurrently on multiple LPs (each specifying a separate migration stream) and may run concurrently with other functions (e.g., TDH.MEM.SCAN.*).

SEPT Tree Concurrency

TDH.EXPORT.MEM acquires a shared lock on the SEPT trees, to prevent blocking or changes to the tree structure (e.g., by TDH.MEM.RANGE.BLOCK or TDH.MEM.SEPT.REMOVE) while it runs. If supported, TDH.EXPORT.MEM may prevent starvation of concurrent memory management functions by detecting that they failed to acquire an exclusive lock on the SEPT trees. In this case, TDH.EXPORT.MEM yields and returns with a TDX_INTERRUPTED_BUSY status in RAX. The host VMM is expected to resume TDH.EXPORT.MEM. See the discussion on interruptibility below.

SEPT Entry Concurrency

Failure to acquire a lock on an SEPT entry is handled as a page-specific error, as described above. TDH.EXPORT.MEM skips the busy page.

3.1.3.3.10. Interruption and Resumption

TDH.EXPORT.MEM is interruptible. An interruption occurs in the following cases:

- If a pending interrupt is detected during operation, TDH.EXPORT.MEM returns with a TDX_INTERRUPTED_RESUMABLE status in RAX.
- If supported and if TDH.EXPORT.MEM detects that a concurrent function (such as TDH.MEM.PAGE.PROMOTE) has failed to acquire an exclusive lock on the SEPT trees, it may yield and return with a TDX_INTERRUPTED_BUSY status in RAX.

If a pending interrupt is detected during operation, TDH.EXPORT.MEM returns with a TDX_INTERRUPTED_RESUMABLE status in RAX. RCX is updated with the next list entry index to process.

The host VMM should re-invoke TDH.EXPORT.MEM after handling the interrupt, keeping the same inputs (and updated value in RCX) except setting R10.RESUME to 1. Failing to resume TDH.EXPORT.MEM and to export the generated migration bundle when it completes successfully will result in the migration protocol going out of sync; this will be detected by the destination side, resulting in an import failure.

The host VMM should not transmit the generated migration bundle to the destination side until TDH.EXPORT.MEM is completed successfully. Doing so will result in the migration protocol going out of sync; this will be detected by the destination side, resulting in an import failure.

3.1.3.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.7: TDH.EXPORT.MEM Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	GPA List page	GPA_LIST	RW	Shared	4KB	None	None	None
Explicit	RDX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	HPA	Memory to use for MBMD	MBMD	RW	Shared	128B	None	None	None
Explicit	R9	HPA	Migration buffer list	PAGE_LIST	RW	Shared	4KB	None	None	None
Explicit	R10	Index	Mig. Stream context	Mig. Stream context	RW	Opaque	N/A	Exclusive	N/A	N/A
Explicit	R11	HPA	MAC list page 1	MAC list	RW	Shared	4KB	None	None	None
Explicit	R12	HPA	MAC list page 2	MAC list	RW	Shared	4KB	None	None	None
Explicit	R14	HPA	attributes list page	page attributes	R	Shared	4KB	None	None	None
Explicit	N/A	GPA	TD private pages (via GPA list)	Blob	R	Private	4KB	None	None	None
Explicit	N/A	HPA	Migration buffer pages (via page list)	Blob	RW	Shared	4KB	None	None	None
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	R	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	N/A	Migration context	N/A	RW	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT tree	N/A	RW	Private	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT entry	SEPT Entry	RW	Private	N/A	Exclusive(h)	N/A	N/A

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Implicit	N/A	GPA	L2 Secure EPT trees	N/A	RW	Private	N/A	Shared(i)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT entries	SEPT Entry	RW	Private	N/A	Exclusive(i)	N/A	N/A

3.1.3.5. *Completion Status Codes*

Table 3.8: TDH.EXPORT.MEM Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_INTERRUPTED_RESUMABLE	
TDX_INVALID_RESUMPTION	
TDX_INTERRUPTED_RESUMABLE	
TDX_MIGRATION_STREAM_STATE_INCORRECT	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	Operation is successful. **Note:** Processing of some GPA list entries may have encountered errors, but this did not cause an abort of the overall operation. The number such errors is reported in the lower 32 bits of the completion status.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.4. Updated: TDH.EXPORT.PAUSE Leaf

…

3.1.4.1. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

TDH.EXPORT.PAUSE starts the Live Migration Blackout period on the source platform.

Enumeration: Availability of TDH.EXPORT.PAUSE is enumerated by either TDX_FEATURES0.TD_MIGRATION (bit 0) or TDX_FEATURES0.S4 (bit 13), readable by TDH.SYS.RD* (see 1.2.1.1), being set to 1. If not supported, calling TDH.EXPORT.PAUSE returns a TDX_OPERAND_INVALID(RAX) status.

Preconditions:
- An export session must be in progress and must be in the LIVE_EXPORT state.
- If the TDX module is configured for non-blocking export, no TD private page may be blocked.
- If the TD is configured for TDX Connect, no TDIs may be bound.

…

Table 3.9: TDH.EXPORT.PAUSE Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	TDR page	TDR	R	Opaque	4KB	Exclusive	Shared	Shared
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Exclusive(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Exclusive(i)	N/A	N/A
Implicit	N/A	N/A	Migration context	N/A	RW	Opaque	N/A	Exclusive(i)	N/A	N/A
Implicit	N/A	N/A	TDCS Epoch Tracking Fields	N/A	RW	Opaque	N/A	Exclusive(i)	N/A	N/A

3.1.4.2. *Completion Status Codes*

Table 3.10: TDH.EXPORT.PAUSE Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_BLOCKED_PAGES_EXIST	There are memory pages that have been blocked by TDH.MEM.RANGE.BLOCK. This is not permitted if the TDX module is configured for non-blocking export.
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	Operation is successful
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	

Completion Status Code	Description
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	
TODO: Error code for bound TDIs.	

3.1.5. Updated: TDH.EXPORT.STATE.IMMUTABLE Leaf

TDH.EXPORT.STATE.IMMUTABLE starts a new export session and exports the TD's immutable state as a multi-page migration bundle.

TDH.EXPORT.STATE.IMMUTABLE is also used for starting a new S4 hibernation session.

3.1.5.1. *Input Operands*

Table 3.11: TDH.EXPORT.STATE.IMMUTABLE Input Operands Definition

Operand	Name	Description		
RAX	Leaf and Version	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function: 72
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RCX	TDR	Source TD handle and flags		
		Bits	**Name**	**Description**
		0	EXPORT_TYPE	0: TD Export 1: S4 Hibernation
		11:1	Reserved	Must be 0
		51:12	TDR HPA	HPA[51:12] of the source TD's TDR page (HKID bits must be 0)
		63:52	Reserved	Must be 0
R8	MBMD	HPA and size of memory of a memory buffer to use for MBMD:		
		Bits	**Name**	**Description**
		51:0	HPA	Bits 51:0 of the host physical address (including HKID bits)
		63:52	Size	Size of the memory buffer containing MBMD, in bytes
R9	PAGE_LIST_INFO	Migration buffers list information – see [ref]		
R10	MIG_STREAM	Migration stream and resume flag:		
		Bits	**Name**	**Description**
		15:0	MIGS_INDEX	Migration stream index – must be 0
		62:16	RESERVED	Reserved: must be 0
		63	RESUME	0: This is a new invocation 1: This is resumption of a previously interrupted operation

3.1.5.2. *Output Operands*

Table 3.12: TDH.EXPORT.STATE.IMMUTABLE Output Operands Definition

Operand	Name	Description
RAX	STATUS	SEAMCALL instruction return code, see [ref]
RDX	NUM_EXPORTED	Number of exported 4KB migration buffers
AVX, AVX2 and AVX512 state		May be reset to the architectural RESET state
Other		Unmodified

3.1.5.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

TDH.EXPORT.STATE.IMMUTABLE starts a new export session. It exports the TD's immutable state as a migration bundle, which includes an MBMD and a set of 4KB pages, encrypted with the migration session key. The migration bundle is protected by a MAC that is stored in the MBMD.

TDH.EXPORT.STATE.IMMUTABLE is also used for starting a new S4 hibernation session.

Enumeration: Availability of TDH.EXPORT.STATE.IMMUTABLE is enumerated by either TDX_FEATURES0.TD_MIGRATION (bit 0) or TDX_FEATURES0.S4 (bit 13), readable by TDH.SYS.RD* (see 1.2.1.1), being set to 1. If not supported, calling TDH.EXPORT.STATE.IMMUTABLE returns a TDX_OPERAND_INVALID(RAX) status.

Interruptibility: TDH.EXPORT.STATE.IMMUTABLE is interruptible. The host VMM is expected to invoke it in a loop until it returns with either a success indication or with a non-recoverable error indication.

3.1.5.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.13: TDH.EXPORT.STATE.IMMUTABLE Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	HPA	Memory to use for MBMD	MBMD	RW	Shared	128B	None	None	None
Explicit	R9	HPA	Page list	PAGE_LIST	RW	Shared	4KB	None	None	None
Explicit	R10	N/A	Mig. Stream context	Mig. Stream context	RW	Opaque	N/A	Exclusive	N/A	N/A
Explicit	N/A	HPA	Destination pages (via page list)	Blob	RW	Shared	4KB	None	None	None
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	Migration context	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	PL.S4_STATE	N/A	RW	Hidden	N/A	Exclusive	N/A	N/A

3.1.5.5. *Completion Status Codes*

Table 3.14: TDH.EXPORT.STATE.IMMUTABLE Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_INTERRUPTED_RESUMABLE	
TDX_INVALID_RESUMPTION	
TDX_IOMMU_IOTLB_TRACKING_NOT_DONE	Applicable only if TDX Connect is supported
TDX_MAX_EXPORTS_EXCEEDED	
TDX_METADATA_LIST_OVERFLOW	
TDX_MIGRATION_SESSION_KEY_NOT_SET	
TDX_MIGRATION_STREAM_STATE_INCORRECT	
TDX_MIN_MIGS_NOT_CREATED	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_PREVIOUS_EXPORT_CLEANUP_INCOMPLETE	
TDX_RND_NO_ENTROPY	Failed to generate a random migration encryption key. This is typically caused by an entropy error of the CPU's random number generator, and may be impacted by RDSEED, RDRAND or PCONFIG executing on other LPs. The operation should be retried.
TDX_SUCCESS	Operation is successful
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_HAS_ATTACHED_DEVICES	Applicable only if TDX Connect is supported
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TD_NOT_MIGRATABLE	

Completion Status Code	Description
TDX_TDCS_NOT_ALLOCATED	

3.1.6. Updated: TDH.EXPORT.TRACK Leaf

TDH.EXPORT.TRACK ends the current export epoch and starts a new one. It either starts a new in-order phase epoch or start the out-of-order phase. In both cases, TDH.EXPORT.TRACK generates an epoch token to be exported to the destination platform.

3.1.6.1. *Inputs*

Table 3.15: TDH.EXPORT.TRACK Input Operands Definition

Operand	Description		
RAX	SEAMCALL instruction leaf number and version, see [ref]		
	Bits	**Field**	**Description**
	15:0	Leaf Number	Selects the SEAMCALL interface function: 71
	23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
	63:24	Reserved	Must be 0
RCX	HPA of the source TD's TDR page (HKID bits must be 0)		
R8	HPA and size of memory of a memory buffer to use for MBMD:		
	Bits	**Name**	**Description**
	51:0	HPA	Bits 51:0 of the host physical address (including HKID bits)
	63:52	Size	Size of the memory buffer containing MBMD, in bytes
R10	Migration stream and flags:		
	Bits	**Name**	**Description**
	15:0	MIGS_INDEX	Migration stream index – must be 0
	62:16	RESERVED	Reserved: must be 0
	63	IN_ORDER_DONE	Indicates that the in-order export phase is done, and a start token should be generated

3.1.6.2. *Outputs*

Table 3.16: TDH.EXPORT.TRACK Output Operands Definition

Operand	Description
RAX	SEAMCALL instruction return code, see [ref]
AVX, AVX2 and AVX512 state	May be reset to the architectural RESET state
Other	Unmodified

3.1.6.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

Enumeration

Availability of TDH.EXPORT.TRACK is enumerated by either TDX_FEATURES0.TD_MIGRATION (bit 0) or TDX_FEATURES0.S4 (bit 13), readable by TDH.SYS.RD* (see 1.2.1.1), being set to 1. If not supported, calling TDH.EXPORT.TRACK returns a TDX_OPERAND_INVALID(RAX) status.

Epoch Token Generation

TDH.EXPORT.TRACK generates an epoch token, to be exported. The epoch token, once successfully imported on the destination side, indicates that a new migration epoch has started.

Start Token Generation

An R10.IN_ORDER_DONE value of 1 indicates that TDH.EXPORT.TRACK is requested to end the in-order export phase, start the out-of-order phase and generate a start token to be exported. A start token is a private case of an epoch token. Once successfully imported on the destination side, the start token enables the host VMM to commit the migration and start running the TD on the destination.

TDH.EXPORT.TRACK checks the completeness of memory export:

- TDH.EXPORT.TRACK checks that the TD has been paused by TDH.EXPORT.PAUSE.
- If the TDX module is configured for non-blocking export, TDH.EXPORT.TRACK checks that a comprehensive scan was successfully done using TDH.MEM.SCAN.COMP(DCHECK).
- TDH.EXPORT.TRACK checks that no TD private memory exported so far needs to be re-exported.
- If either the TD is configured for TDX Connect or the TDX module does not support post-copy (this is the case if the TDX module is configured for non-blocking export), TDH.EXPORT.TRACK checks that all the TD private memory has been exported.

When called as part of S4 hibernation, R10.IN_ORDER_DONE must be 1.

3.1.6.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.17: TDH.EXPORT.TRACK Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	HPA	Memory to use for MBMD	MBMD	RW	Shared	128B	None	None	None
Explicit	R10	Index	Mig. Stream context	Mig. Stream context	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Exclusive	N/A	N/A
Implicit	N/A	N/A	Migration context	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A

3.1.6.5. *Completion Status Codes*

Table 3.18: TDH.EXPORT.TRACK Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_EXPORTED_DIRTY_PAGES_REMAIN	
TDX_MEM_SCAN_DCHECK_NOT_DONE	
TDX_MIGRATION_EPOCH_OVERFLOW	
TDX_UNEXPORTED_MEMORY_REMAINS	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	Operation is successful
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.7. Updated: TDH.EXPORT.UNBLOCKW Leaf

…

Leaf Function Description

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

TDH.EXPORT.UNBLOCKW finds the write blocked Secure EPT entry for the given GPA and level. It verifies that the entry has been blocked for writing and TLB tracking has been done, then marks the entry as non-blocked for writing (MAPPED, PENDING, EXPORTED_DIRTY or PENDING_EXPORTED_DIRTY as appropriate). If the page has any L2 mappings, TDH.EXPORT.UNBLOCKW unblocks them.

Enumeration: Availability of TDH.EXPORT.UNBLOCKW is enumerated by TDX_FEATURES0.TD_MIGRATION (bit 0), readable by TDH.SYS.RD* (see 1.2.1.1). If not supported, calling TDH.EXPORT.UNBLOCKW returns a TDX_OPERAND_INVALID(RAX) status.

Export Mode: TDH.EXPORT.UNBLOCKW is only available if the TDX module is configured for write-blocking based export. If not available, calling TDH.EXPORT.UNBLOCKW returns a TDX_OPERAND_INVALID(RAX) status.

…

3.1.8. Updated: TDH.IMPORT.MEM Leaf

…

Re-Import

Re-import is only allowed during the in-order import phase. The imported pages replace an older version of the same pages, if the SEPT entry state is compatible:

- If the old SEPT state is PENDING, it may be overwritten by a new version that is either PENDING or MAPPED.
- If the old SEPT state is MAPPED, then if the TDX module supports page release by the guest TD, as enumerated by TDX_FEATURES0.PAGE_RELEASE, it may be overwritten by a new version that is either MAPPED or PENDING. Else, it may only be overwritten by a new version that is MAPPED.

Page attributes (e.g., RWX etc.) of a new page version may be different than those of a previously imported version.

If the out-of-order import phase, the imported pages may not overwrite an older version of the same pages.

…

3.1.9. Updated: TDH.MEM.PAGE.ADD Leaf

…

6. Walk the Secure EPT based on the GPA operand and find the leaf EPT entry for the 4KB page.
   6.1. If the TDX module is configured for non-blocking export, set the Dirty bit in all non-leaf SEPT entries during the walk as an indication that the GPA range contains memory pages.
   For optimization, this is done here even though the operation may fail, and the page may not be added.

…

11. Increment TDCS.MEM_COUNT.
12. Increment TDR.CHLDCNT.

…

3.1.10. Updated: TDH.MEM.PAGE.AUG Leaf

…

6. Walk the Secure EPT based on the GPA operand and find the leaf EPT entry for the 4KB or 2MB page.
   6.1. If the TDX module is configured for non-blocking export, set the Dirty bit in all non-leaf SEPT entries during the walk as an indication that the GPA range contains memory pages.
   For optimization, this is done here even though the operation may fail, and the page may not be added.

…

8. Atomically increment TDCS.MEM_COUNT by 1 (for a 4KB page) or by 512 (for a 2MB page).
9. Atomically increment TDR.CHLDCNT by 1 (for a 4KB page) or by 512 (for a 2MB page).

…

3.1.11. Updated: TDH.MEM.PAGE.DEMOTE Leaf

…

3.1.11.1. *Leaf Function Description*

…

3.1.11.1.1. New: Interaction with Non-Blocking Export

If the page state before the demote operation was MAPPED and a non-blocking export session is in the LIVE_EXPORT phase, TDH.MEM.PAGE.DEMOTE sets all the small pages' Dirty bits. Otherwise, it clears them.

…

3.1.11.2. *Completion Status Codes*

Table 3.19: TDH.MEM.PAGE.DEMOTE Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_EPT_ENTRY_STATE_INCORRECT	
TDX_EPT_WALK_FAILED	
TDX_GPA_RANGE_NOT_BLOCKED	
TDX_INTERRUPTED_RESTARTABLE	TDH.MEM.PAGE.DEMOTE's operation has been interrupted by an external event; it may be restarted (from its beginning) by calling it again.
TDX_L2_SEPT_PAGE_NOT_PROVIDED	
TDX_L2_SEPT_WALK_FAILED	
TDX_MISSING_PAMT_PAGE_PAIR	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_PAGE_NOT_FREE	
TDX_SUCCESS	TDH.MEM.PAGE.DEMOTE is successful.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	
TDX_TLB_TRACKING_NOT_DONE	

3.1.12. Updated: TDH.MEM.PAGE.REMOVE Leaf

…

3.1.13. Updated: TDH.MEM.RANGE.BLOCK Leaf

Block a TD private GPA range (i.e., a Secure EPT page or a TD private page) at any level (4KB, 2MB, 1GB, 512GB, 256TB, etc.) from creating new GPA-to-HPA address translations.

3.1.13.1. *Inputs*

Table 3.20: TDH.MEM.RANGE.BLOCK Input Operands Definition

Operand	Description		
RAX	SEAMCALL instruction leaf number and version, see [ref]		
	Bits	**Field**	**Description**
	15:0	Leaf Number	Selects the SEAMCALL interface function: 7
	23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
	63:24	Reserved	Must be 0
RCX	EPT mapping information:		
	Bits	**Name**	**Description**
	2:0	Level	Level of the Secure EPT entry that maps the GPA range to be blocked – see [ref] Level must between 0 and 3 for a 4-level EPT or between 0 and 4 for a 5-level EPT.
	11:3	Reserved	Reserved: must be 0
	51:12	GPA	Bits 51:12 of the GPA range to be blocked Depending on the level, the following least significant bits must be 0: Level 0 (EPTE): None Level 1 (EPDE): Bits 20:12 Level 2 (EPDPTE): Bits 29:12 Level 3 (EPML4E): Bits 38:12 Level 4 (EPML5E): Bits 47:12
	63:52	Reserved	Reserved: must be 0
RDX	Host physical address of the parent TDR page (HKID bits must be 0)		

3.1.13.2. *Outputs*

Table 3.21: TDH.MEM.RANGE.BLOCK Output Operands Definition

Operand	Description
RAX	SEAMCALL instruction return code – see [ref]

Operand	Description
RCX	Extended error information part 1 In case of EPT walk error, Secure EPT entry architectural content where the error was detected – see **1.4.1** The architectural content represents how the Secure EPT maps a private memory page or a Secure EPT page and may be different than the actual contents of the Secure EPT entry. Software should consult the Secure EPT information returned in RDX. In other cases, RCX returns 0.
RDX	Extended error information part 2 In case of EPT walk error, Secure EPT entry level and state where the error was detected – see **1.4.1** In other cases, RDX returns 0.
Other	Unmodified

3.1.13.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

TDH.MEM.RANGE.BLOCK finds the Secure EPT entry for the given GPA and level, and it marks it as blocked (BLOCKED or PENDING_BLOCKED as appropriate). It records the current TD's TLB epoch in the PAMT entry of the physical Secure EPT page or TD private page mapped by the blocked Secure EPT entry.

Interaction with TD Migration

If the TDX module is configured for write-blocking based export, TDH.MEM.RANGE.BLOCK cannot be used to block a page that has been exported. To block an exported page, the host VMM must first cancel the page export by calling TDH.EXPORT.MEM with a CANCEL operation for that page.

Else (the TDX module is configured for non-blocking export), TDH.MEM.RANGE.BLOCK cannot block any memory while an export session is in the blackout phase, i.e., after TDH.EXPORT.PAUSE is called and before either TDH.EXPORT.TRACK(DONE) or TDH.EXPORT.ABORT is called.

Interaction with TDX Connect

If the TD is configured with TDX Connect enabled, the following conditions apply:

- If the GPA range is mapped by a leaf SEPT entry, TDH.MEM.RANGE.BLOCK of MAPPED pages is only allowed if no TDIs are attached. TDH.MEM.RANGE.BLOCK of PENDING pages is allowed unconditionally.
- Else (the GPA range is mapped by a non-leaf SEPT entry), TDH.MEM.RANGE.BLOCK is only allowed if either no TDIs are attached, or if all 512 entries of the SEPT child page are FREE.

3.1.13.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.22: TDH.MEM.RANGE.BLOCK Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	GPA and Level	Secure EPT page or TD private page	Blob	None	Private	$2^{12+9*Level}$ Bytes	None	None	None

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RDX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT tree	N/A	RW	Private	N/A	Exclusive	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT entry	SEPT Entry	RW	Private	N/A	Exclusive(h)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT trees	N/A	RW	Private	N/A	Exclusive(i)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT entries	SEPT Entry	RW	Private	N/A	Exclusive(i)	N/A	N/A

3.1.13.5. *Completion Status Codes*

Table 3.23: TDH.MEM.RANGE.BLOCK Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_BLOCKING_DISALLOWED	
TDX_EPT_ENTRY_STATE_INCORRECT	
TDX_EPT_WALK_FAILED	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	TDH.MEM.RANGE.BLOCK is successful.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.14. New: TDH.MEM.SCAN.COMP/RANGE – Common

This section contains definitions that are common to TDH.MEM.SCAN.COMP and TDH.MEM.SCAN.RANGE.

3.1.14.1. *GPA List-of-Lists Processing*

3.1.14.1.1. Overview

GPA list-of-lists is defined in 1.5.1.

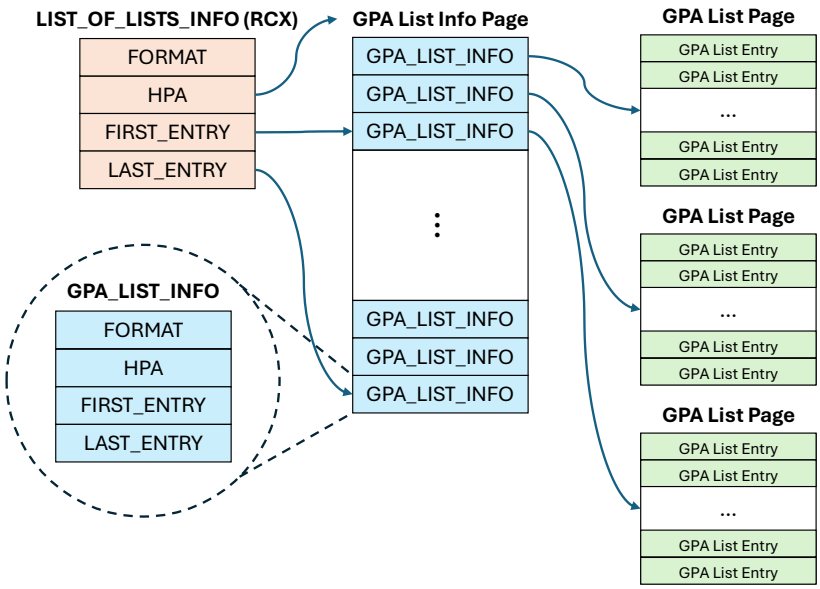

Figure 3.2: GPA List of Lists

The setting of LIST_OF_LIST_INFO and GPA_LIST_INFO entries by the host VMM, and their update by TDH.MEM.SCAN.*, is designed such that if TDH.MEM.SCAN.* is interrupted, it can be easily resumed with no need to modify the setting. On successful termination, or when the lists are full, the LIST_OF_LIST_INFO and GPA_LIST_INFO entries contain the proper information for the GPA lists, as updated by TDH.MEM.SCAN.*. This is summarized in the table below and detailed in the following sections.

Table 3.24: TDH.MEM.SCAN.* List Indices on Input and Output

	LIST_OF_LIST_INFO (in RCX)		GPA_LIST_INFO	
	FIRST_ENTRY	**LAST_ENTRY**	**FIRST_ENTRY**	**LAST_ENTRY**
Initial input	Next entry to process (should be 0)	Last valid GPA_LIST_INFO in page	Next entry to process (should be 0)	Ignored
Output on interrupt	Next entry to process	Unmodified	Next entry to process	Index of last processed GPA list entry
Input on resumption	Next entry to process	Last valid GPA_LIST_INFO in page	Next entry to process	Ignored
Output on completion of each GPA list page	N/A	N/A	0	Index of last processed GPA list entry (511)
Output on scan completion (at least one GPA list entry is valid) or on list full condition	0	Index of last processed GPA_LIST_INFO	0	Index of last processed GPA list entry

	LIST_OF_LIST_INFO (in RCX)		GPA_LIST_INFO	
	FIRST_ENTRY	LAST_ENTRY	FIRST_ENTRY	LAST_ENTRY
Output on scan completion (no GPA list entry is valid)	511	0	N/A	N/A

3.1.14.1.2. List-of-Lists on Input

Empty Lists (On Initial Invocation or on Resumption after an Interruption due to a Full List)

The host VMM is expected to set up a list-of-lists structure, consisting of a GPA list info page and one or more GPA list pages in shared memory, when calling TDH.MEM.SCAN.* in either of the following cases:

- R8.RESUME set to 0, or
- R8.RESUME set to 1 after TDH.MEM.SCAN.* returned with a TDX_INTERRUPTED_LIST_FULL status in RAX

The host VMM should set LIST_OF_LISTS_INFO input in RCX as follows:

- HPA must be a shared HPA (including HKID) pointing to the GPA list info page.
- FORMAT must be LIST_OF_LISTS (2).
- FIRST_ENTRY should be set to 0.
- LAST_ENTRY may be set to any value between FIRST_ENTRY and 511.

Note: FIRST_ENTRY is not checked by TDH.MEM.SCAN.* to be 0. Failure to set it to 0 may result in an incorrectly formatted GPA list.

The host VMM should set each GPA_LIST_INFO entry in the GPA list info page as follows:

- HPA must be a shared HPA (including HKID) pointing to the respective GPA list page.
- FORMAT must be GPA_ONLY (0).
- FIRST_ENTRY should be set to 0.
- LAST_ENTRY is ignored by the TDX module.

Note: FIRST_ENTRY is not checked by TDH.MEM.SCAN.* to be 0. Failure to set it to 0 may result in an incorrectly formatted GPA list.

Resumption After an Interruption other than List Full

When calling TDH.MEM.SCAN.* with R8.RESUME set to 1, resuming it after it returned with TDX_INTERRUPTED_RESUMABLE or TDX_INTERRUPTED_BUSY, the host VMM should not modify LIST_OF_LISTS_INFO input in RCX nor any of the GPA_LIST_INFO entries in the GPA list info page.

3.1.14.1.3. List-of-Lists on Output

GPA list pages are filled in the order of scan, i.e., ascending GPA order. Note that each concurrent instance of TDH.MEM.SCAN.* writes its own output lists; there is no specific order between lists written by different instances.

GPA_LIST_INFO entries (in the GPA list info page) which have been fully processed are updated as follows:

- FIRST_ENTRY is set to 0.
- LAST_ENTRY is set to the index of the last updated entry on the page.

GPA_LIST_INFO entries (in the GPA list info page) which have not yet been processed at all are unmodified.

Interruption other than List Full

When TDH.MEM.SCAN.* is interrupted with a TDX_INTERRUPTED_RESUMABLE or a TDX_INTERRUPTED_BUSY status, it updates the GPA list-of-list fields such that the host VMM can call it to resume the operation.

LIST_OF_LISTS_INFO in RCX is updated as follows:

- FIRST_ENTRY is set to the index of the entry to be processed on resumption.

The GPA_LIST_INFO entry (in the GPA list info page) which was being processed at the time of interruption is updated as follows:

- FIRST_ENTRY is set to index of the entry to be processed on resumption.

Successful Completion or Interruption due to a Full List

When TDH.MEM.SCAN.* completed the range scan successfully, or is interrupted with a TDX_INTERRUPTED_LIST_FULL status

LIST_OF_LISTS_INFO in RCX is updated as follows:

- FIRST_ENTRY is set to 0.
- LAST_ENTRY is set to the index of the last entry processed.

The GPA_LIST_INFO entry (in the GPA list info page) which was being processed at the time of interruption is updated as follows:

- FIRST_ENTRY is set to the index of the next entry to process.
- LAST_ENTRY is set to index of the last entry processed.

3.1.15. New: TDH.MEM.SCAN.COMP Leaf

Do a comprehensive scan of the TD's private GPA space and perform the requested operation.

3.1.15.1. *Inputs*

Table 3.25: TDH.MEM.SCAN.COMP Input Operands Definition

Operand	Name	Description		
RAX	Leaf and Version	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function: 93
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RCX	LIST_OF_LISTS_ INFO	HPA of a list-of-lists in shared memory, and first and last entries to process. The list contains up to 512 PAGE_LIST_INFO entries, each pointing to a GPA list containing up to 512 entries. For details, see the description below and 1.5.1. FORMAT must be LIST_OF_LISTS (2). If the requested OPERATION does not return a page list (e.g., EXPORT_RESTORE), LIST_OF_LIST_INFO is ignored.		
RDX	TDR	HPA of the source TD's TDR page (HKID bits must be 0)		
R8	CONTROLS	Controls fields:		
		Bits	**Field**	**Description**
		7:0	OPERATION	Identifies the requested operation – see the table below **Enumeration:** See below for details.
		15:8	QUALIFIER	Additional qualification of the requested operation
		31:16	RESERVED	Reserved, must be 0
		47:32	CONTEXT_ID	Identifies the context of this invocation, used when a comprehensive scan of the GPA range is requested.
		55:48	RANGE_ID	Identifies the GPA range (previously configured by TDH.SCAN.CONFIG) for this invocation
		62:56	RESERVED	Reserved, must be 0
		63	RESUME	0: This is a new invocation of TDH.MEM.SCAN.COMP 1: This is resumption of a previously interrupted TDH.MEM.SCAN.COMP operation

3.1.15.2. *Outputs*

Table 3.26: TDH.MEM.SCAN.COMP Output Operands Definition

Operand	Name	Description
RAX	STATUS	SEAMCALL instruction return code, see [ref]
RCX	LIST_OF_LISTS_INFO	Similar to the input value, except that FIRST_ENTRY and LAST_ENTRY are updated. See the description below for details.
Other		Unmodified

3.1.15.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

3.1.15.3.1. Overview

TDH.MEM.SCAN.COMP scans the whole TD's GPA range and performs the requested operation. Most operations fill page information in the returned GPA lists.

3.1.15.3.2. Enumeration

TDH.MEM.SCAN.COMP is supported if any of its operations are enumerated as available. See the operation modes table below for details. If not supported, calling TDH.MEM.SCAN.COMP returns a TDX_OPERAND_INVALID(RAX) status.

The supported number of memory scan contexts in enumerated by NUM_MEM_SCAN_CONTEXTS.

3.1.15.3.3. Operation Modes and Qualifiers

TDH.MEM.SCAN.COMP may support multiple operation modes as detailed below. Operation modes may have qualifiers.

Table 3.27: TDH.MEM.SCAN.COMP Operation Modes

OPERATION Value	Name	Description
1	DCHECK	Do a comprehensive scan of the TD's GPA range for a final list of export candidates. DCHECK is used in the final round of export, after the TD has been paused and as a prerequisite to exporting the up-to-date memory image before calling TDH.EXPORT.TRACK(DONE). DCHECK is only available if the TDX module is configured for non-blocking export. An export session must be in progress, and the TD must have been paused by TDH.EXPORT.PAUSE. **Enumeration:** DCHECK support is enumerated by TDX_FEATURES0.NON_BLOCKING_EXPORT (bit 41), readable by TDH.SYS.RD* (see 1.2.1.1). DCHECK has two sub modes, selected by the QUALIFIER input: **0: EXPORT:** Scan and return a GPA list of memory pages that either have not been exported or need to be re-exported. **1: REEXPORT:** Scan and return a GPA list of memory pages that need to be re-exported. Pages that have not been exported are still counted by UNEXPORT_COUNT. This mode is useful for implementing post-copy (if supported). **Other:** Reserved
Other	RESERVED	Reserved

3.1.15.3.4. Comprehensive Scan Details

The whole private GPA space is scanned by multiple, possibly concurrent, invocations of TDH.MEM.SCAN.COMP. While a comprehensive scan is in progress, no changes are allowed to the TD's GPA space structure.

3.1.15.3.4.1. Multiple GPA Ranges

To allow optimization for NUMA configurations, multiple GPA ranges may be configured by the host VMM using TDH.MEM.SCAN.CONFIG.

The host VMM can invoke concurrent scanning TDH.MEM.SCAN.COMP threads of the GPA ranges, typically executed on LPs that are close (in the NUMA sense) to the memory range being scanned.

3.1.15.3.4.2. Scan Concurrency within a GPA Range

Within each GPA range, the host VMM can invoke multiple concurrent scanning TDH.MEM.SCAN.COMP threads. The threads balance the work among themselves.

1. When invoked, TDH.MEM.SCAN.COMP allocates a sub-range to scan. The sub-range size is configured by the host VMM using TDH.MEM.SCAN.CONFIG.
2. When done with that sub-range, if there is still unscanned memory in the range, TDH.MEM.SCAN.COMP allocates a new sub-range to scan and loops to step 1 above.
3. If there is no more memory to scan, TDH.MEM.SCAN.COMP returns to the host VMM. If there are no more concurrent scans running in the current range, it indicates a successful scan of the range.

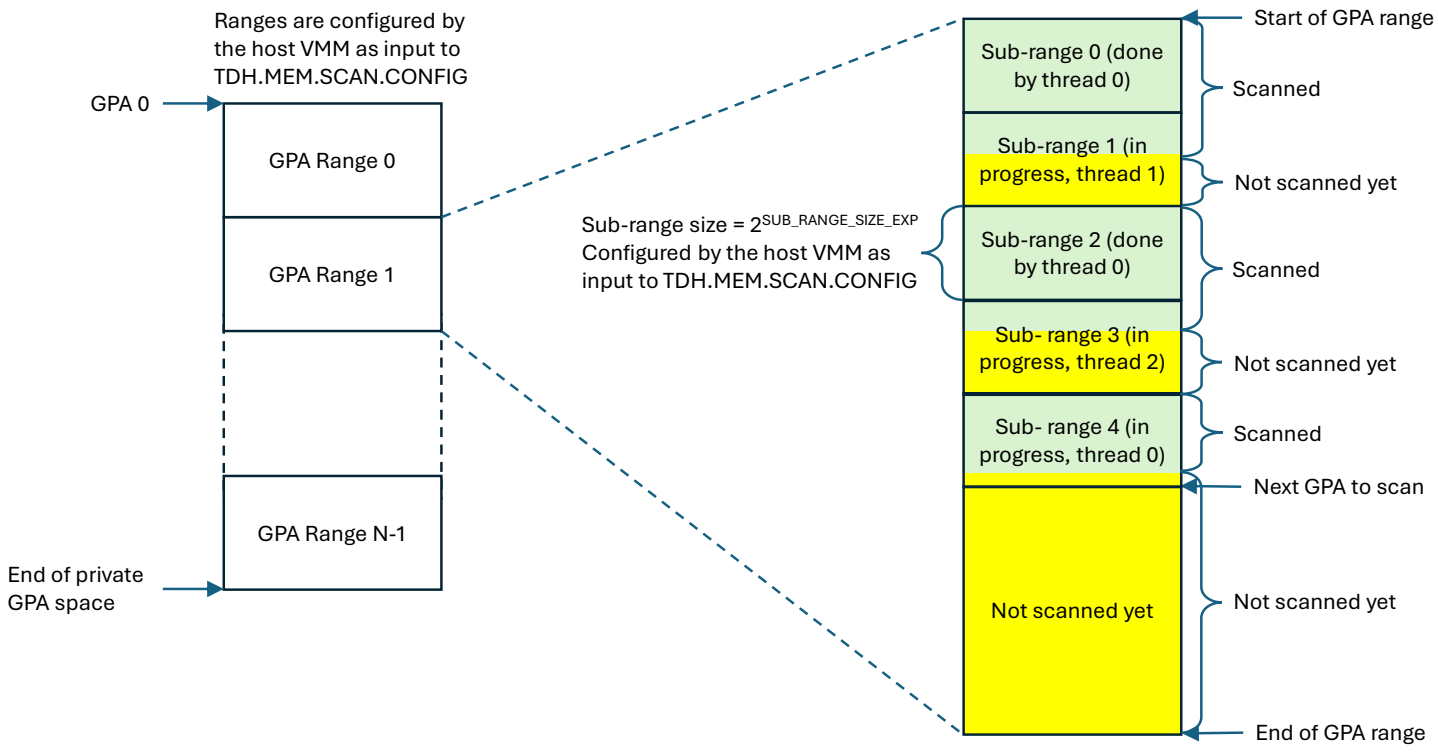

Figure 3.3: Concurrent Scan within a GPA Range

3.1.15.3.4.3. Comprehensive Scan Start

A new comprehensive scan starts when the host VMM calls TDH.MEM.SCAN.COMP to perform a comprehensive scan, and either of the following conditions is true:

- This is the first invocation of TDH.MEM.SCAN.COMP since the TD was created or imported, or
- This is the first invocation of TDH.MEM.SCAN.COMP since TDH.MEM.SCAN.RESET was called.

3.1.15.3.4.4. Comprehensive Scan Context

To support multiple concurrent comprehensive scan threads, the TDX module holds multiple instances of scan contexts per TD, allocated by TDH.MEM.SCAN.CONFIG. The CONTEXT_ID parameter is used as a handle for the internally held context, to allow resumption after an incomplete operation or an interruption.

3.1.15.3.4.5. Comprehensive Scan Success

If TDH.MEM.SCAN.COMP executes successfully, the return status is RAX indicates the following:

TDX_SUCCESS	The current invocation of TDH.MEM.SCAN.COMP completed successfully, but there are concurrent instances of TDH.MEM.SCAN.COMP still scanning the current GPA range.
TDX_MEM_RANGE_SCAN_SUCCESS	Comprehensive scan of the current GPA range completed successfully, but there are concurrent instances of TDH.MEM.SCAN.COMP threads still scanning other GPA ranges. This status is returned at most once per GPA range.
TDX_MEM_SCAN_SUCCESS	Comprehensive scan of the entire GPA space completed successfully. This status is returned at most once per comprehensive scan.

3.1.15.3.4.6. Comprehensive Scan Failure

There might be cases where a TDH.MEM.SCAN.COMP instance fails in a way that fails the whole comprehensive scan. For example, this happens when a blocked SEPT entry that is discovered by TDH.MEM.SCAN.COMP(DCHECK). In such cases, TDH.MEM.SCAN.COMP returns a TDH_MEM_SCAN_FAILED status. Concurrently executing TDH.MEM.SCAN.COMP instances consequently also fail, returning a TDH_MEM_SCAN_FAILED(OTHER_THREAD_FAILED) status.

After a comprehensive scan failed, the host VMM may start a new comprehensive scan. To do that, it first needs to call TDH.COMP.SCAN.RESET to reset the internal comprehensive scan state.

3.1.15.3.4.7. Backward Compatibility

TDH.MEM.SCAN.COMP supports scanning the memory of a TD that was created by an older TDX module that didn't support TDH.MEM.SCAN.COMP, and the TDX module was later updated using a TD-preserving update. This scan may be somewhat less efficient since some information collected during the TD lifetime may not exist.

3.1.15.3.5. Concurrency

TDH.MEM.SCAN.COMP may be called concurrently on multiple LPs and may run concurrently with other functions (e.g., TDH.EXPORT.MEM). This is especially important for the DCHECK operation, which is done during the export blackout time and thus should be completed as fast as possible.

3.1.15.3.5.1. Memory Page Concurrency

For the DCHECK operation, concurrently changing page state in a way that impacts export correctness is prevented by the architectural restrictions on the allowed memory management operations.

3.1.15.3.5.2. SEPT Tree Concurrency

TDH.MEM.SCAN.COMP acquires a shared lock on the SEPT trees, to prevent blocking or changes to the tree structure (e.g., by TDH.MEM.RANGE.BLOCK or TDH.MEM.SEPT.REMOVE) while TDH.MEM.SCAN.COMP is running. However, this lock does not prevent SEPT tree structure modification when TDH.MEM.SCAN is not executing (e.g., it was interrupted).

For the DCHECK operation, the non-blocking export architecture prevents SEPT tree structure change while the comprehensive scan is in progress, even if no instance of TDH.MEM.SCAN.COMP is running, by not allowing the applicable memory management functions to run.

3.1.15.3.5.3. SEPT Entry Concurrency

If TDH.MEM.SCAN.COMP fails to acquire a lock on an SEPT entry, it returns with a TDX_INTERRUPTED_BUSY status in RAX. The host VMM is expected to resume TDH.MEM.SCAN.COMP; see the discussion on interruptibility below.

3.1.15.3.6. GPA List-of-Lists Processing

Processing of the GPA list-of-list by TDH.MEM.SCAN.COMP and by TDH.MEM.SCAN.RANGE is similar; thus it is described in 3.1.14.1.

3.1.15.3.7. Interruption and Resumption

TDH.MEM.SCAN.COMP is interruptible and resumable. An interruption occurs in the following cases:

- If a pending interrupt is detected during operation, TDH.MEM.SCAN.COMP returns with a TDX_INTERRUPTED_RESUMABLE status in RAX.

- If TDH.MEM.SCAN.COMP fails to acquire a lock on an SEPT entry, it returns with a TDX_INTERRUPTED_BUSY Status in RAX.
- If TDH.MEM.SCAN.COMP detects that a concurrent function (such as TDH.EXPORT.ABORT) has failed to acquire an exclusive lock on the comprehensive memory scan state, it yields and returns with a TDX_INTERRUPTED_BUSY status in RAX.

In all cases, RCX and the page list in memory are updated with the next list entry index to process, so the host VMM may re-invoke TDH.MEM.SCAN.COMP soon after handling the interrupt.

If an interrupted TDH.MEM.SCAN.COMP is resumed after an error was detected by a TDH.MEM.SCAN.COMP running in another context, it immediately returns with a TDX_MEM_SCAN_FAILED_OTHER_THREAD status.

3.1.15.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.28: TDH.MEM.SCAN.COMP Memory Operands Information

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	GPA list info page	HPA	RW	Shared	4KB	None	None	None
Explicit	N/A	N/A	GPA list pages (via GPA list info page)	HPA	RW	Shared	4KB	None	None	None
Explicit	RDX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	Unsigned Integer	Comprehensive scan context	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A
Explicit	R9	Unsigned Integer	GPA range	N/A	RW	Opaque	4KB	None	None	None
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	R	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT tree	N/A	RW	Private	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT entry	SEPT Entry	RW	Private	N/A	Exclusive(h)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT trees	N/A	RW	Private	N/A	Shared(i)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT entries	SEPT Entry	RW	Private	N/A	Exclusive(i)	N/A	N/A
Implicit	N/A	N/A	Comprehensive scan state	N/A	RW	Opaque	N/A	Shared	N/A	N/A

3.1.15.5. *Completion Status Codes*

Table 3.29: TDH.MEM.SCAN.COMP Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_BLOCKED_PAGES_EXIST	There are memory pages that have been blocked by TDH.MEM.RANGE.BLOCK. This is not permitted if the TDX module is configured for non-blocking export.
TDX_MEM_SCAN_FAILED_BLOCKED_RANGE	
TDX_MEM_SCAN_FAILED_OTHER_THREAD	
TDX_MEM_SCAN_SUCCESS	
TDX_INTERRUPTED_LIST_FULL	The output GPA lists are full.
TDX_INVALID_RESUMPTION	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_SUCCESS	Operation is successful.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	
TDX_TDCS_PAGES_REQUIRED	

3.1.16. New: TDH.MEM.SCAN.CONFIG Leaf

Configure memory scan and add control structure pages.

3.1.16.1. *Inputs*

Table 3.30: TDH.MEM.SCAN.CONFIG Input Operands Definition

Operand		Description		
RAX	LEAF_AND_VERSION	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function: 94
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RCX	RANGE_LIST_INFO	The shared physical address (including HKID) of a of a page where memory scan control page will be created		
		Bits	**Field**	**Description**
		8:0	NUM_RANGES	Number of GPA ranges Must be between 1 and MAX_MEM_SCAN_RANGES, readable by TDH.SYS.RD*.
		11:9	RESERVED	Must be 0
		51:12	RANGE_LIST_HPA	Shared HPA (Incl. HKID) of a range list page
		63:52	RESERVED	Must be 0
RDX	TD_HANDLE	The physical address of the owner TDR page (HKID bits must be 0)		
R8	CX0_HPA	The physical address where memory scan control page 0 will be created (HKID bits must be 0)		
R9	CX1_HPA	The physical address where memory scan control page 1 will be created (HKID bits must be 0) A valid HPA is required if the value of MEM_SCAN_CONFIG_PAGES, readable by TDH.SYS.RD*, is 2 or higher. Otherwise, R9 is ignored.		
R10	CX2_HPA	The physical address where memory scan control page 2 will be created (HKID bits must be 0) A valid HPA is required if the value of MEM_SCAN_CONFIG_PAGES, readable by TDH.SYS.RD*, is 3 or higher. Otherwise, R10 is ignored.		
R11	CX3_HPA	The physical address where memory scan control page 3 will be created (HKID bits must be 0) A valid HPA is required if the value of MEM_SCAN_CONFIG_PAGES, readable by TDH.SYS.RD*, is 4 or higher. Otherwise, R11 is ignored.		

3.1.16.2. *Outputs*

Table 3.31: TDH.MEM.SCAN.CONFIG Output Operands Definition

Operand	Name	Description
RAX	Status	SEAMCALL instruction return code – see [ref]
Other		Unmodified

3.1.16.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

TDH.MEM.SCAN.CONFIG adds a page where scan control information used by TDH.MEM.SCAN will reside. This function can be invoked at any time after the TDCS pages have been allocated.

TDH.MEM.SCAN.CONFIG can only be successfully invoked if no comprehensive memory scan session is in progress.

3.1.16.3.1. Enumeration

TDH.MEM.SCAN.CONFIG is supported if TDH.MEM.SCAN is supported. If not supported, calling TDH.MEM.SCAN.CONFIG returns a TDX_OPERAND_INVALID(RAX) status.

The required number of memory scan control pages is enumerated by NUM_MEM_SCAN_CONTROL_PAGES, readable by TDH.SYS.RD*.

The maximum number of GPA ranges is enumerated by MAX_MEM_SCAN_RANGES.

3.1.16.3.2. Range List Page

The range list page holds a list of GPA ranges, to be used by TDH.MEM.SCAN. The list is formatted as an array of 64-bit entries, with entry N specifying the start of a GPA range N. The number of valid entries is specified by RCX.NUM_RANGES. See the description of TDH.MEM.SCAN.COMP in 3.1.15.3.4 for details.

Table 3.32: Range List Entry

Bits	Name	Description
20:0	RESERVED	Must be 0
50:21	RANGE_START	Bits 50:21 of the range start GPA RANGE_START must be a valid private GPA, lower than the maximum valid private GPA allowed for the TD – see the [TDX Module Base Spec] section on GPA Space Size Configuration and Virtualization. RANGE_START must be aligned on SUB_RANGE_SIZE, i.e., all bits lower than SUB_RANGE_SIZE_EXT must be 0. RANGE_START of entry 0 must be 0.
51	RESERVED	Must be 0
57:52	SUB_RANGE_SIZE_EXP	Specifies the sub-range size to be used by TDH.MEM.SCAN.COMP within the current range: $SUB\_RANGE\_SIZE = 2^{SUB\_RANGE\_SIZE\_EXP}$ SUB_RANGE_SIZE_EXP must be at least 21 (SUB_RANGE_SIZE of 2MB). Sub ranges are used by TDH.MEM.SCAN for multithreaded scanning. A SUB_RANGE_SIZE value equal or higher than the range size forces the range scan to be single threaded.
63:58	RESERVED	Must be 0

3.1.16.3.3. Dynamic PAMT

If the TDX module is configured for dynamic PAMT, the PAMT hierarchy can be built on demand. A TDX_MISSING_PAMT_PAGE_PAIR status indicates that a PAMT page pair is missing for a control page. The host VMM may add it using TDH.PHYMEM.PAMT.ADD and retry the operation.

3.1.16.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.33: TDH.MEM.SCAN.CONFIG Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	Range list page	N/A	R	Shared	4KB	N/A	N/A	N/A
Explicit	RDX	HPA	TDR page	TDR	RW	Opaque	4KB	Shared	Shared	Shared
Explicit	R8	HPA	Control page 0	N/A	RW	Opaque	4KB	Exclusive	Shared	Shared
Explicit	R9	HPA	Control page 1	N/A	RW	Opaque	4KB	Exclusive	Shared	Shared
Explicit	R10	HPA	Control page 2	N/A	RW	Opaque	4KB	Exclusive	Shared	Shared
Explicit	R11	HPA	Control page 3	N/A	RW	Opaque	4KB	Exclusive	Shared	Shared
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	4KB	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Shared(h)	N/A	N/A
Implicit	N/A	N/A	Comprehensive scan state	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A

3.1.16.5. *Completion Status Codes*

Table 3.34: TDH.MEM.SCAN.CONFIG Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_MEM_SCAN_CONFIG_ALREADY_DONE	
TDX_MEM_RANGE_SCAN_SUCCESS	
TDX_MISSING_PAMT_PAGE_PAIR	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_OPERAND_PAGE_METADATA_INCORRECT	
TDX_SUCCESS	Operation is successful.

Completion Status Code	Description
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.17. New: TDH.MEM.SCAN.RANGE Leaf

Scan a range of the TD's private GPA space and perform the requested operation.

3.1.17.1. *Input Operands*

Table 3.35: TDH.MEM.SCAN.RANGE Input Operands Definition

Operand	Name	Description		
RAX	Leaf and Version	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function: 92
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RCX	LIST_OF_LISTS_ INFO	HPA of a list-of-lists in shared memory, and first and last entries to process. The list contains up to 512 PAGE_LIST_INFO entries, each pointing to a GPA list containing up to 512 entries. For details, see the description below and 1.5.1. FORMAT must be LIST_OF_LISTS (2). If the requested OPERATION does not return a page list (e.g., EXPORT_RESTORE), LIST_OF_LIST_INFO is ignored.		
RDX	TDR	HPA of the source TD's TDR page (HKID bits must be 0)		
R8	CONTROLS	Controls fields:		
		Bits	**Field**	**Description**
		7:0	OPERATION	Identifies the requested operation – see the table below **Enumeration:** See below for details.
		15:8	QUALIFIER	Additional qualification of the requested operation
		62:16	RESERVED	Reserved, must be 0
		63	RESUME	0: This is a new invocation of TDH.MEM.SCAN.RANGE 1: This is resumption of a previously interrupted TDH.MEM.SCAN.RANGE operation
R9	RANGE_START	The start address of the private GPA range to scan Must be a valid private GPA, aligned on 4KB.		
R10	RANGE_SIZE	The size of the GPA range to scan Must be a multiple of 4KB. Bits 63:52 must be 0.		

3.1.17.2. *Output Operands*

Table 3.36: TDH.MEM.SCAN.RANGE Output Operands Definition

Operand	Name	Description
RAX	STATUS	SEAMCALL instruction return code, see [ref]

Operand	Name	Description
RCX	LIST_OF_LISTS_INFO	Similar to the input value, except that FIRST_ENTRY and LAST_ENTRY are updated. See the description below for details.
R9	NEXT_START	The first GPA for next scan, aligned on 4KB
R10	REMAINING_SIZE	The remaining size of the GPA range to scan, in multiples of 4KB
Other		Unmodified

3.1.17.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

3.1.17.3.1. Overview

TDH.MEM.SCAN.RANGE scans the requested TD's GPA range and performs the requested operation. Most operations fill page information in the returned GPA lists.

3.1.17.3.2. Enumeration

TDH.MEM.SCAN.RANGE is supported if any of its operations are enumerated as available. See the operation modes table below for details. If not supported, calling TDH.MEM.SCAN returns a TDX_OPERAND_INVALID(RAX) status.

3.1.17.3.3. Operation Modes and Qualifiers

TDH.MEM.SCAN.RANGE supports multiple operation modes as detailed below. Some TDH.MEM.SCAN.RANGE operation modes are available only if the TDX module is configured for non-blocking export. Operation modes may have qualifiers.

Table 3.37: TDH.MEM.SCAN.RANGE Operation Modes

OPERATION Value	Name	Description
0	DSCAN	Scan for export candidates. DSCAN is only available if the TDX module is configured for non-blocking export. **Enumeration:** DSCAN support is enumerated by TDX_FEATURES0.NON_BLOCKING_EXPORT (bit 41), readable by TDH.SYS.RD* (see 1.2.1.1). DSCAN has two sub modes, selected by the QUALIFIER input: **0: EXPORT:** Scan and return a GPA list of memory pages that either have not been exported or need to be re-exported. **1: REEXPORT:** Scan and return a GPA list of memory pages that need to be re-exported. **Other:** Reserved
2	EXPORT_RESTORE	Restore SEPT entries state after an export session has been aborted by TDH.EXPORT.ABORT. EXPORT_RESTORE does not return a page list; LIST_OF_LISTS_INFO is ignored. **Enumeration:** EXPORT_RESTORE support is enumerated by TDX_FEATURES0.SCAN_EXPORT_RESTORE (bit 43), readable by TDH.SYS.RD* (see 1.2.1.1).
Other	RESERVED	Reserved

3.1.17.3.4. Concurrency

TDH.MEM.SCAN.RANGE may be called concurrently on multiple LPs and may run concurrently with other functions (e.g., TDH.EXPORT.MEM).

3.1.17.3.4.1. Memory Page Concurrency

TDH.MEM.SCAN.RANGE may not detect pages that are concurrently added to the TD, or whose state or attributes (including the Dirty bit) are concurrently modified, while scan is in progress.

3.1.17.3.4.2. SEPT Tree Concurrency

TDH.MEM.SCAN.RANGE acquires a shared lock on the SEPT trees, to prevent blocking or changes to the tree structure (e.g., by TDH.MEM.RANGE.BLOCK or TDH.MEM.SEPT.REMOVE) while TDH.MEM.SCAN.RANGE is in progress. Note that this lock does not prevent SEPT tree structure modification when TDH.MEM.SCAN.RANGE is not executing (e.g., it was interrupted).

TDH.MEM.SCAN.RANGE may prevent starvation of concurrent memory management functions by detecting that they failed to acquire an exclusive lock on the SEPT trees. In this case, TDH.MEM.SCAN.RANGE yields and returns with a TDX_INTERRUPTED_BUSY Status in RAX. The host VMM is expected to resume TDH.MEM.SCAN.RANGE. See the discussion on interruptibility below.

3.1.17.3.4.3. SEPT Entry Concurrency

If TDH.MEM.SCAN.RANGE fails to acquire a lock on an SEPT entry, behavior depends on the requested operation:

- For a DSCAN operation, TDH.MEM.SCAN.RANGE skips the busy SEPT entry.
  Note: Non-blocking export is designed to ensure that such entries will be scanned at least once by DCHECK. For details, see the [TD Migration Spec].
- For an EXPORT_RESTORE operation, TDH.MEM.SCAN.RANGE returns with a TDX_INTERRUPTED_BUSY Status in RAX. The host VMM is expected to resume TDH.MEM.SCAN.RANGE. See the discussion on interruptibility below.

3.1.17.3.5. GPA List-of-Lists Processing

Processing of the GPA list-of-list by TDH.MEM.SCAN.COMP and by TDH.MEM.SCAN.RANGE is similar; thus it is described in 3.1.14.1.

3.1.17.3.6. TLB Tracking

For the DSCAN operation, whenever TDH.MEM.SCAN.RANGE clears the Dirty bit of an SEPT entry, it records the current TD epoch (in the PAMT entry for that page). This allows TDH.EXPORT.MEM to check TLB tracking to help ensure that EPT translation caches have been flushed before the page is exported. Unlike write-blocking based export, TLB tracking is page-specific, allowing TDH.MEM.SCAN.RANGE(DSCAN) of a certain GPA range to run concurrently with TDH.EXPORT.MEM of another range.

TD epoch is not recorded for the EXPORT_RESTORE operation.

3.1.17.3.7. Interruption and Resumption

TDH.MEM.SCAN.RANGE is interruptible and resumable. An interruption occurs in the following cases:

- If a pending interrupt is detected during operation, TDH.MEM.SCAN.RANGE returns with a TDX_INTERRUPTED_RESUMABLE status in RAX.
- If TDH.MEM.SCAN.RANGE fails to acquire a lock on an SEPT entry during an EXPORT_RESTORE operation, it returns with a TDX_INTERRUPTED_BUSY Status in RAX.
- If TDH.MEM.SCAN.RANGE detects that a concurrent function (such as TDH.MEM.PAGE.PROMOTE) has failed to acquire an exclusive lock on the SEPT trees, it may yield and return with a TDX_INTERRUPTED_BUSY Status in RAX.

In all cases, RCX and the page list in memory are updated with the next list entry index to process, so the host VMM may re-invoke TDH.MEM.SCAN soon after handling the interrupt.

3.1.17.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.38: TDH.MEM.SCAN.RANGE Memory Operands Information

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	GPA list info page	HPA	RW	Shared	4KB	None	None	None
Explicit	N/A	N/A	GPA list pages (via GPA list info page)	HPA	RW	Shared	4KB	None	None	None
Explicit	RDX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared
Explicit	R9	GPA	GPA range	GPA	R	Private	4KB	None	None	None
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	R	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT tree	N/A	RW	Private	N/A	Shared	N/A	N/A
Implicit	N/A	GPA	L1 Secure EPT entry	SEPT Entry	RW	Private	N/A	Exclusive(h)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT trees	N/A	RW	Private	N/A	Shared(i)	N/A	N/A
Implicit	N/A	GPA	L2 Secure EPT entries	SEPT Entry	RW	Private	N/A	Exclusive(i)	N/A	N/A

3.1.17.5. *Completion Status Codes*

Table 3.39: TDH.MEM.SCAN.RANGE Completion Status Codes (Returned in RAX) Definition

Completion Status Code	Description
TDX_MEM_SCAN_FAILED_BLOCKED_RANGE	
TDX_MEM_SCAN_FAILED_OTHER_THREAD	
TDX_MEM_SCAN_FAILED_PREPARE_REQUIRED	
TDX_MEM_SCAN_SUCCESS	
TDX_INTERRUPTED_LIST_FULL	The output GPA lists are full.
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_SUCCESS	Operation is successful.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	

Completion Status Code	Description
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	
TDX_TDCS_PAGES_REQUIRED	

3.1.18. New: TDH.MEM.SCAN.RESET Leaf

Reset the TDX module's comprehensive memory scan internal state for the specified TD.

3.1.18.1. *Inputs*

Table 3.40: TDH.MEM.SCAN.RESET Input Operands Definition

Operand	Name	Description		
RAX	Leaf and Version	SEAMCALL instruction leaf number and version, see [ref]		
		Bits	**Field**	**Description**
		15:0	Leaf Number	Selects the SEAMCALL interface function: 95
		23:16	Version Number	Selects the SEAMCALL interface function version Must be 0
		63:24	Reserved	Must be 0
RDX	TDR	HPA of the source TD's TDR page (HKID bits must be 0)		

3.1.18.2. *Outputs*

Table 3.41: TDH.MEM.SCAN.RESET Output Operands Definition

Operand	Name	Description
RAX	STATUS	SEAMCALL instruction return code, see [ref]
Other		Unmodified

3.1.18.3. *Leaf Function Description*

Note: The description below is provided at a high level. Actual details, order of checks, returned status codes, etc. may vary.

3.1.18.3.1. Overview

TDH.MEM.SCAN.RESET resets the internal state held by the TDX module for comprehensive memory scans of the specified TD's GPA address space. It is typically used if a scan by TDH.MEM.SCAN.COMP failed, before attempting a new scan.

3.1.18.3.2. Enumeration

TDH.MEM.SCAN.RESET is supported if TDH.MEM.SCAN.COMP is supported. If not supported, calling TDH.MEM.SCAN.RESET returns a TDX_OPERAND_INVALID(RAX) status.

3.1.18.4. *Operands Information*

To understand the table and text below, please refer to the [TDX Module Base Spec] chapter discussing general aspects of the Intel TDX Module API.

Table 3.42: TDH.MEM.SCAN.RANGE Memory Operands Information

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RDX	HPA	TDR page	TDR	R	Opaque	4KB	Shared	Shared	Shared

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	R9	GPA	GPA range	GPA	R	Private	4KB	None	None	None
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i)	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	R	Opaque	N/A	Shared	N/A	N/A
Implicit	N/A	N/A	Comprehensive scan state	N/A	RW	Opaque	N/A	Exclusive	N/A	N/A

3.1.18.5. *Completion Status Codes*

Table 3.43: TDH.MEM.SCAN.RESET Completion Status Codes (Returned in RAX) Definition TO BE COMPLETED

Completion Status Code	Description
TDX_MEM_SCAN_CONFIG_REQUIRED	
TDX_MEM_SCAN_IN_PROGRESS	
TDX_OP_STATE_INCORRECT	
TDX_OPERAND_ADDR_RANGE_ERROR	
TDX_OPERAND_BUSY	Operation encountered a busy operand, indicated by the lower 32 bits of the status. In many cases, this can be resolved by retrying the operation.
TDX_OPERAND_INVALID	
TDX_SUCCESS	Operation is successful.
TDX_SYS_NOT_READY	
TDX_SYS_SHUTDOWN	
TDX_TD_FATAL	
TDX_TD_KEYS_NOT_CONFIGURED	
TDX_TDCS_NOT_ALLOCATED	

3.1.19. Updated: TDH.SYS.CONFIG Leaf

Globally configure the Intel TDX module.

Table 3.44: Updated: TDH.SYS.CONFIG Input Operands Definition

Operand	Description		
RAX	SEAMCALL instruction leaf number and version, see [ref]		
	Bits	**Field**	**Description**
	15:0	Leaf Number	Selects the SEAMCALL interface function
	23:16	Version Number	Selects the SEAMCALL interface function version Versions 0 and 1 are supported. See enumeration details below.
	63:24	Reserved	Must be 0
RCX	The physical address of an array of pointers, each containing the physical address of a single TDMR_INFO entry (see [ref]). The pointer array must be sorted such that TDMR base addresses (TDMR_INFO.TDMR_BASE) are sorted from the lowest to the highest base address, and TDMRs do not overlap with each other.		
RDX	The number of pointers in the above buffer, between 1 and 64		
R8	**Bits**	**Name**	**Description**
	15:0	HKID	TDX module's global private HKID value
	16	DYNAMIC_PAMT	Selects whether the TDX module will use a static or dynamic allocation of PAMT 0: Static PAMT 1: Dynamic PAMT **Enumeration:** Support of Dynamic PAMT is enumerated by TDX_FEATURES0.DYNAMIC_PAMT (bit 36), readable by TDH.SYS.RD*.
	63:17	Reserved	Reserved: must be 0
R9	If the requested version in RAX is 1 or higher, R9 specifies TDX module feature enabling flags, formatted similarly to TDX_FEATURES0, readable by TDH.SYS.RD*. A bit may be set to 1 if the corresponding TDX_FEATURES0 bit is 1. Else, R9 is ignored.		
	6	TDX_CONNECT	Enables TDX Connect
	41	NON_BLOCKING_EXPORT	Controls TD export mode: 0: Write-blocking based export 1: Non-blocking export
	Bits that are 0 in TDX_FEATURES0		Must be 0
	Other bits		Ignored
R10	If the requested version in RAX is 1 or higher, R10 is reserved for additional TDX module feature enabling flags. It must be 0. Else, R10 is ignored.		

…

Enumeration: Support of dynamic PAMT is enumerated by TDX_FEATURES0.DYNAMIC_PAMT (bit 36), readable by the host VMM using TDH.SYS.RD*.

TDH.SYS.CONFIG version 1 is supported if the value of either TDX_FEATURES0.TDX_CONNECT (bit 6) or TDX_FEATURES0.NON_BLOCKING_EXPORT (bit 41) (listed as configurable in R9) is 1.

…

3.1.20. Updated: TDH.SYS.UPDATE Leaf

Populate Intel TDX module internal variables from the handoff data prepared by the previous Intel TDX module.

Table 3.45: TDH.SYS.UPDATE Input Operands Definition

Operand	Description		
RAX	SEAMCALL instruction leaf number and version, see [ref]		
	Bits	**Field**	**Description**
	15:0	Leaf Number	Selects the SEAMCALL interface function: 53
	23:16	Version Number	Selects the SEAMCALL interface function version Versions 0 and 1 are supported. See enumeration details below.
	63:24	Reserved	Must be 0
R9	If the requested version in RAX is 1 or higher, R9 specifies TDX module feature enabling flags, formatted similarly to TDX_FEATURES0, readable by TDH.SYS.RD*. A bit may be set to 1 if the corresponding TDX_FEATURES0 bit is 1. There are limitations of how features can be enabled vs. the previous TDX modules that ran on the platform since the initial TDX module installation, as detailed below. Else, R9 is ignored.		
	6	TDX_CONNECT	Enables TDX Connect. TDX Connect cannot be disabled if it was enabled for the pre-update TDX module.
	41	NON_BLOCKING_EXPORT	Controls TD export mode: 0: Write-blocking based export. This value is valid only if the pre-update TDX module was configured for write-blocking based export. 1: Non-blocking export. This value is valid only if write-blocking based export session has never happened during the lifecycle of any of the previous TDX modules from which a TD-preserving update was done.
	Bits that are 0 in TDX_FEATURES0		Must be 0
	Other bits		Ignored
R10	If the requested version in RAX is 1 or higher, R10 is reserved for additional TDX module feature enabling flags. It must be 0. Else, R10 is ignored.		

…

Enumeration: Availability of TDH.SYS.UPDATE is enumerated by TDX_FEATURES0.TD_PRESERVING (bit 1), readable by TDH.SYS.RD* (see 1.2.1.1). If not supported, calling TDH.SYS.UPDATE returns a TDX_OPERAND_INVALID(RAX) status.

TDH.SYS.UPDATE version 1 is supported if the value of either TDX_FEATURES0.TDX_CONNECT (bit 6) or TDX_FEATURES0.NON_BLOCKING_EXPORT (bit 41) (listed as configurable in R9) is 1.

…

3.1.21. Updated: TDH.VP.ENTER Leaf

…

Table 3.46: Updated: TDH.VP.ENTER Operands Information Definition

Explicit/ Implicit	Reg.	Ref Type	Resource	Resource Type	Access	Access Semantics	Align Check	Concurrency Restrictions		
								Operand	Contain. 2MB	Contain. 1GB
Explicit	RCX	HPA	TDVPR page	TDVPS	RW	Opaque	4KB	Shared(c)[3]	Shared(c)[3]	Shared(c)[3]
Implicit	N/A	HPA	TDR page	TDR	RW	Opaque	N/A	Shared(c)[3]	N/A	N/A
Implicit	N/A	N/A	TDCS structure	TDCS	RW	Opaque	N/A	Shared(i,c)[3]	N/A	N/A
Implicit	N/A	N/A	TDCS.OP_STATE	OP_STATE	RW	Opaque	N/A	Shared(t)[4]	N/A	N/A
Implicit	N/A	N/A	TDCS TLB Tracking Fields	N/A	RW	Opaque	N/A	Shared(t)[4]	N/A	N/A
Implicit	N/A	N/A	SEPT tree	N/A	R	Opaque	N/A	Shared(t)[4]	N/A	N/A

5

[3] The shared locking of TDVPS, TDR, TDCS, TDCS.OP_STATE is for the whole duration of running in TDX non-root mode; the locks are released on TD exit.

[4] The locking of OP_STATE, SEPT tree and the TLB tracking fields is until before entering TDX non-root mode; the locks are released before VM entry into the TD VCPU.

3.2. Updated: Guest-Side (TDCALL) Interface Functions

3.2.1. Updated: TDG.MEM.PAGE.ACCEPT Leaf

...

3.2.2. Updated: TDG.MEM.PAGE.ATTR.WR Leaf

...

3.2.3. Updated: TDG.MEM.PAGE.RELEASE Leaf

Release a private page, enabling the host VMM to remove it.