Confidential Compute: Extending Confidentiality to Data Storage

Chandra Nelogal

Agenda

- Overview – context and problem statement
- Data storage and confidential compute
- Overview of SEDs (detour)
- Solution
    - MEK mapping
- Industry standards status and details
- Conclusion

Overview

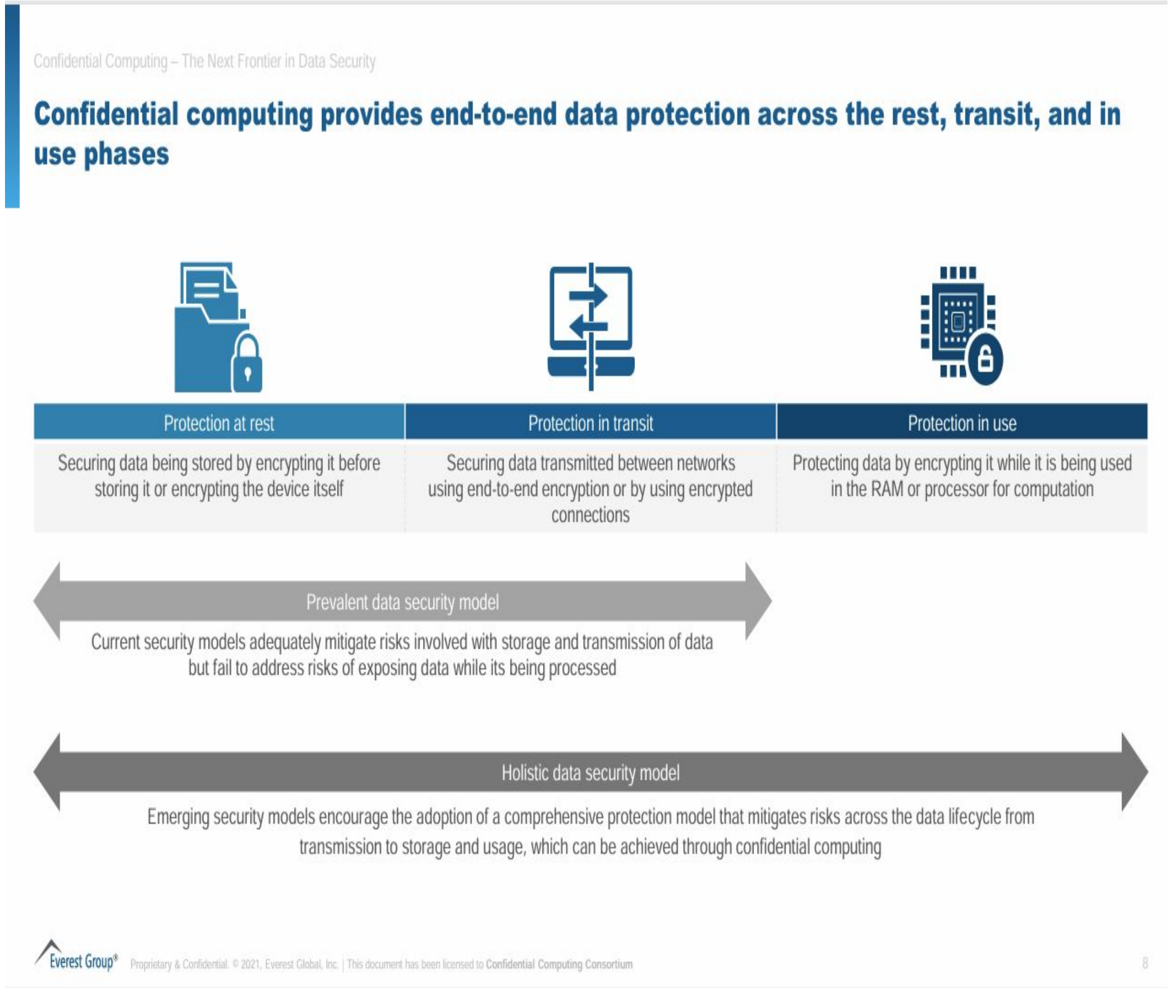
Context:

This presentation focuses on data storage at rest.

Data in transit and in use are protected by confidential compute frameworks.

Problem Statement:

- Typically the data at rest is protected using SEDs (Self Encrypting Drives).
- Data access is controlled by authentication keys. Scope for data access can be the full drive or statically configured encryption bands



Data storage in Confidential Computing Environments

- Most well known use cases for confidential compute involve virtual machines running on Compute and Memory complex
- Data stored on the storage devices can be encrypted and protected
- Typical mechanisms involve SEDs (Self Encrypting Drives)
  - Host needs to authenticate to the drive to gain access to data
- The authentication keys for SEDs can be served from remote Key Management Servers
  - Local key management options available as well

<u>The Case for Confidential Computing (linuxfoundation.org)</u>

<u>The_Next_Frontier_in_Data_Security</u>

SED overview

- SED - Self Encrypting Drives
- Drives encrypt data on the recording media using media encryption keys
- The media encryption keys are managed by the drive
  - Generate, Change, Delete
- Authentication keys are managed by the host
  - Provision, Unlock (power event), Rekey, Revert
- Encryption bands set up
  - One band for all addressable media
  - Fixed number of bands
- Encryption bands are typically are set up at provisioning time
- Problem Statement: Typical Enterprise: Key scope is entire drive

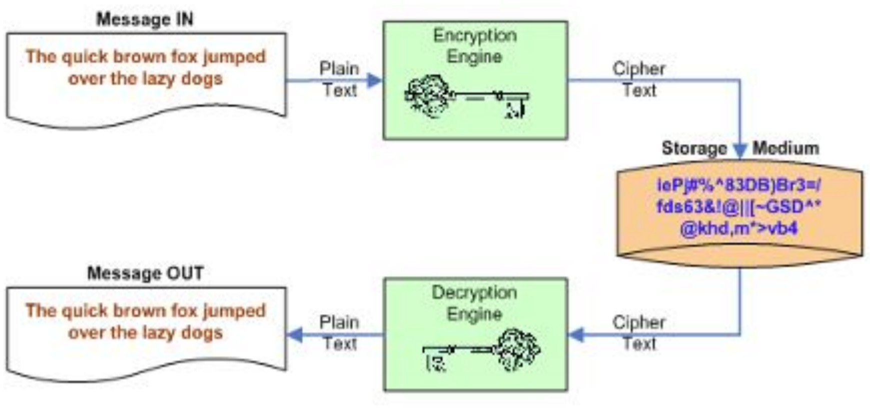


Figure 1. Encryption and decryption

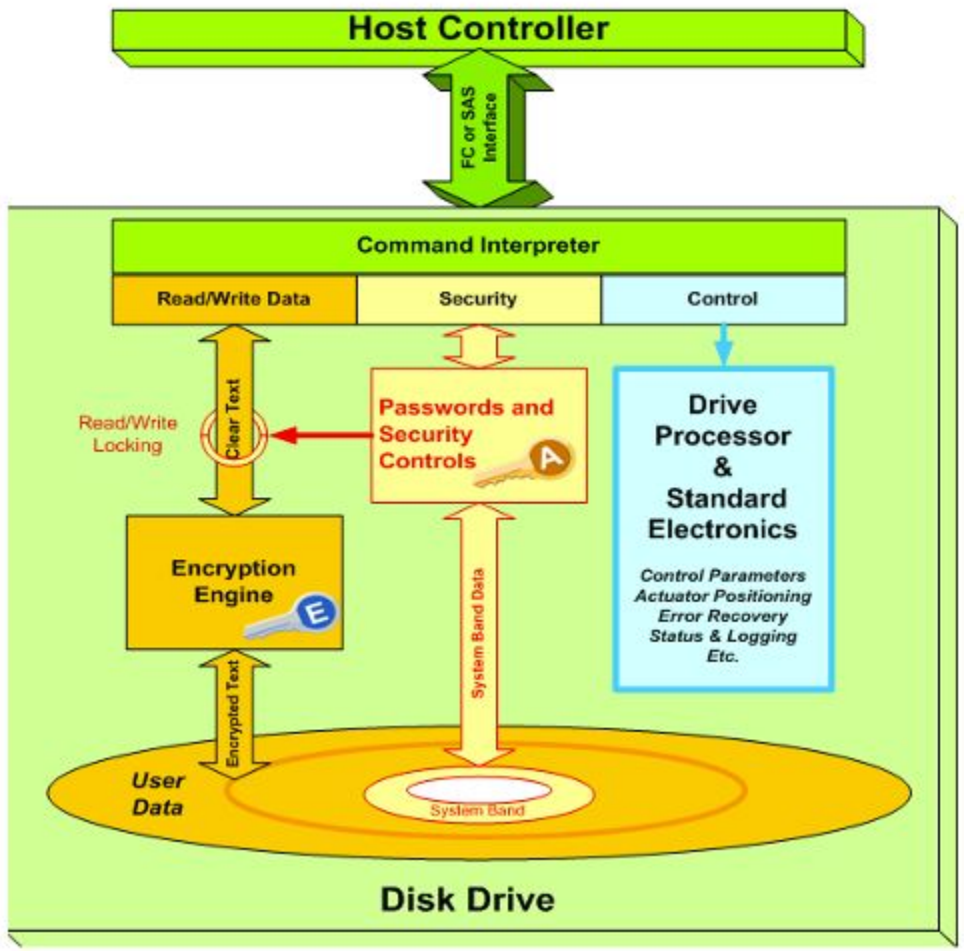


Figure 6. SED major components

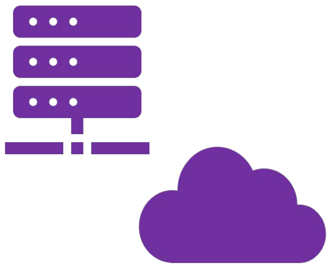
https://www.seagate.com/files/staticfiles/support/docs/manual/Interface%20manuals/100515636b.pdf

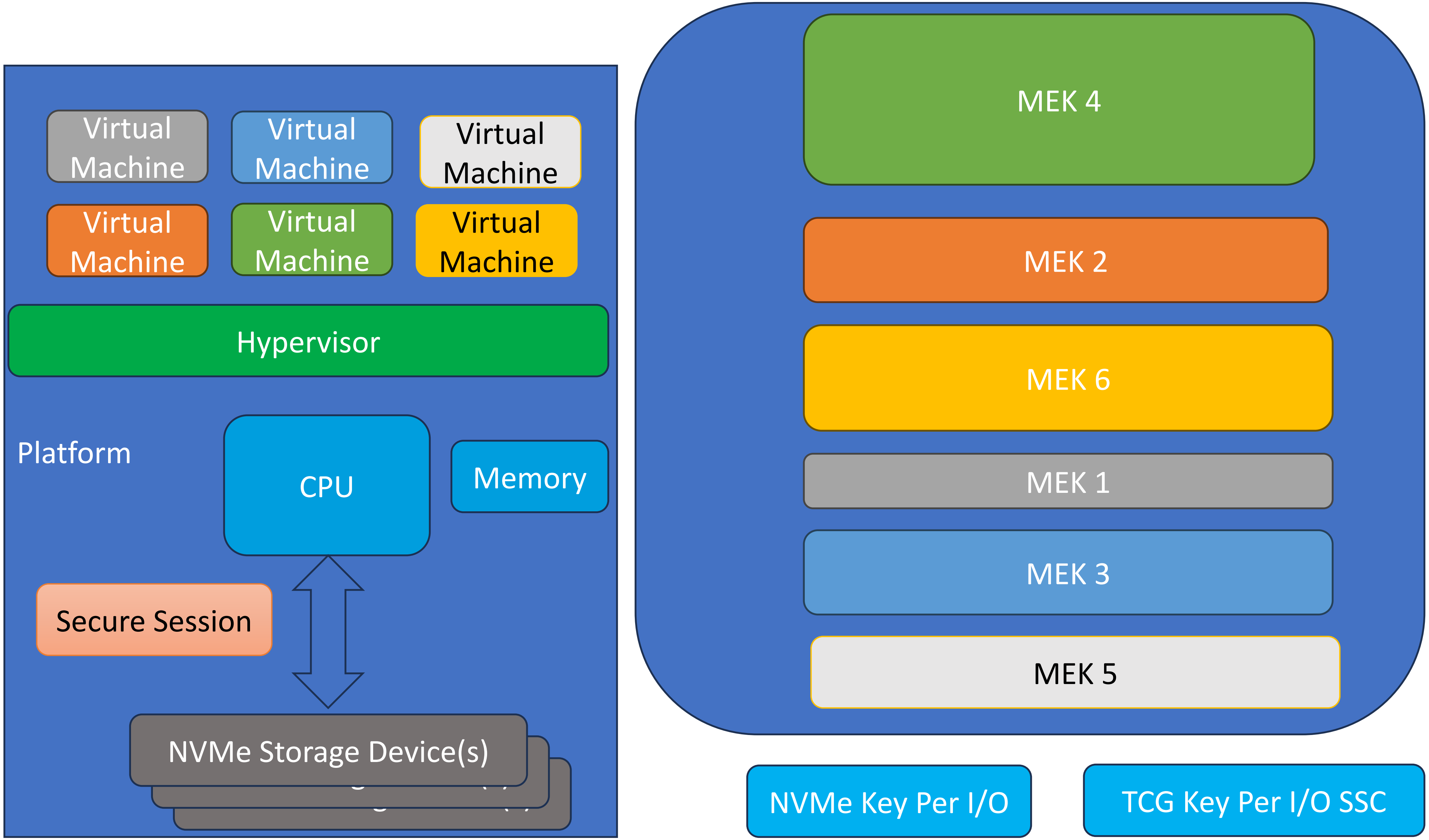
Solution

- There are a couple of updates to the specifications
  - NVMe – Support for Key Per I/O
  - TCG Storage – Key Per IO SSC (Security Subsystem class)
- Main points of key per I/O
  - The media encryption key is provisioned from the host
  - The media encryption key is _not_ stored persistently on the drive
- Hence, data access is not possible with host providing the media access key
- MEK scope is on a per read/write basis
- Allows for dynamic data range on a per VM or a per container basis
  - Per file as well
- Data can be considered erased by just deleting the key at the key store
  - Or anywhere in the storage stack
  - No Risk of data exposure if the drive is lost or mis handled (Decommissioning, transfers, etc.)
  - Reverse circulation possible

Media Encryption Keys (MEKs) – 1:1 per VM

Key Management Server





Example can be extended to containers

NVMe command extensions for KPIO

A controller that supports the Key Per I/O capability shall set the KPIOS bit to '1' in the Identify Controller data structure (refer to Figure 312).

A namespace that supports the Key Per I/O capability shall set the KPIOSNS bit to '1' in the I/O Command Set Independent Identify Namespace data structure (refer to Figure 319).

The Key Per I/O capability uses the Command Extension Type (CETYPE) and Command Extension Value (CEV) fields in all read and write commands. Definition of the CETYPE fields are shown in Figure 620.

Figure 620: CETYPE Definition

Value	Definition	CEV Field Definition
0h	Reserved	
1h	**Key Per I/O Tag (KPIOTAG):** This command is using the Key Per I/O capability.	**Key Tag (KEYTAG)**: Specifies a namespace-specific 16-bit encryption key tag that identifies the encryption key used to encrypt or decrypt the data of the command. The same Key Tag value on different namespaces may or may not identify the same encryption key. Refer to the Maximum Key Tag field in the I/O Command Set Independent Identify Namespace data structure (refer to Figure 319) for the supported values.
2h to Eh	Reserved	
Fh	Vendor Specific	

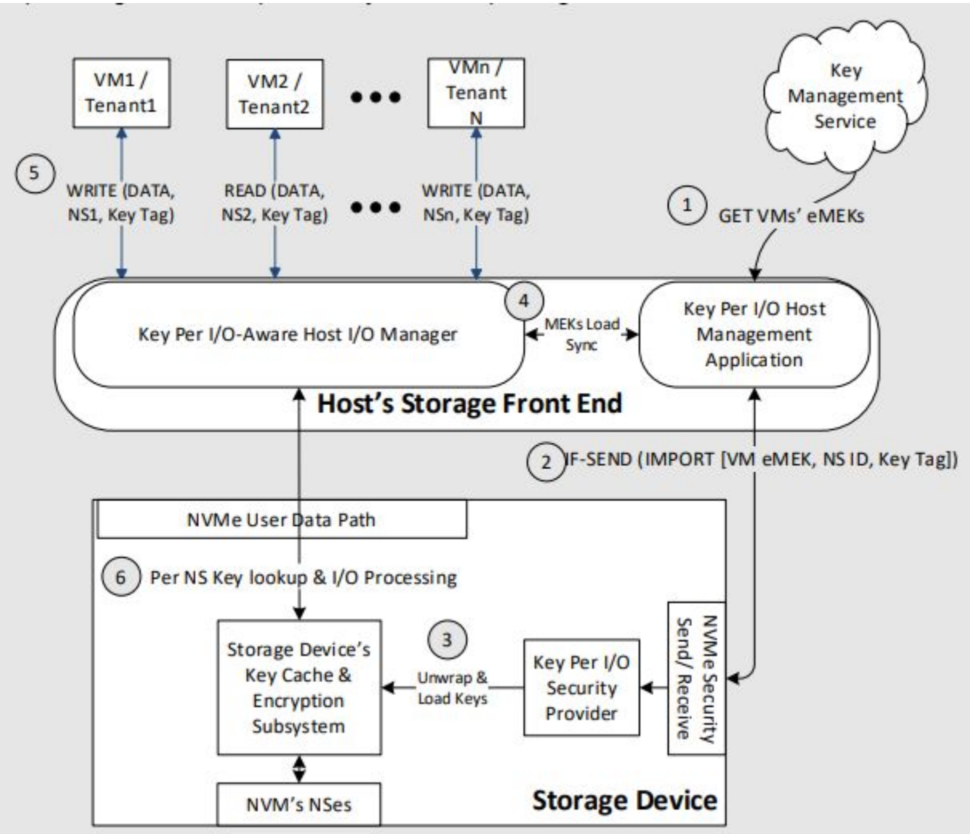


Figure 1: Example Key Per I/O's Operating Model in a Multi-Tenant Storage System

Enhancing Data Encryption Capabilities in the Data Center with the NVM Express® Key Per I/O Feature - NVM Express

Specifications - NVM Express

MEK, Key Tag and usage mapping

Use Case	Key Tag	MEK
VM1	Key Tag 1	MEK1
VM2	Key Tag 2	MEK2
VM3	Key Tag 3	MEK3

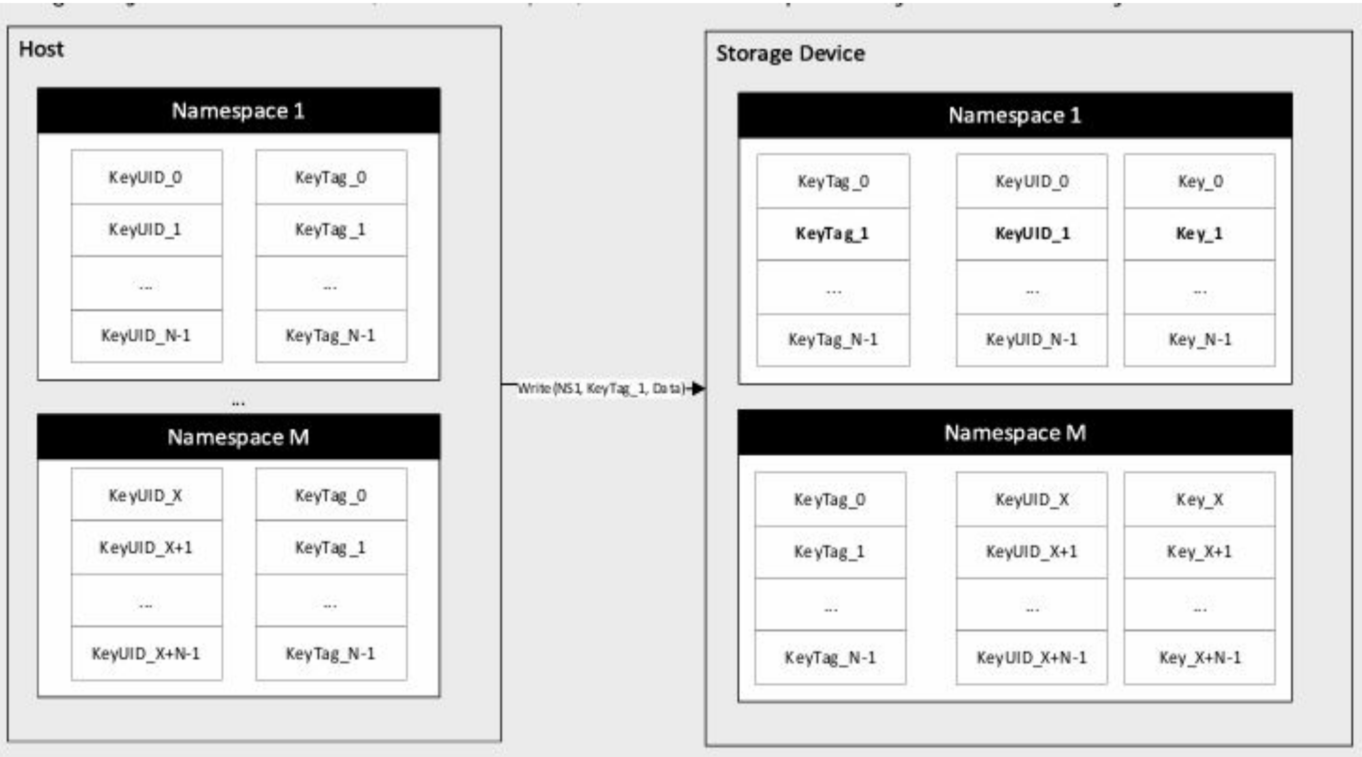
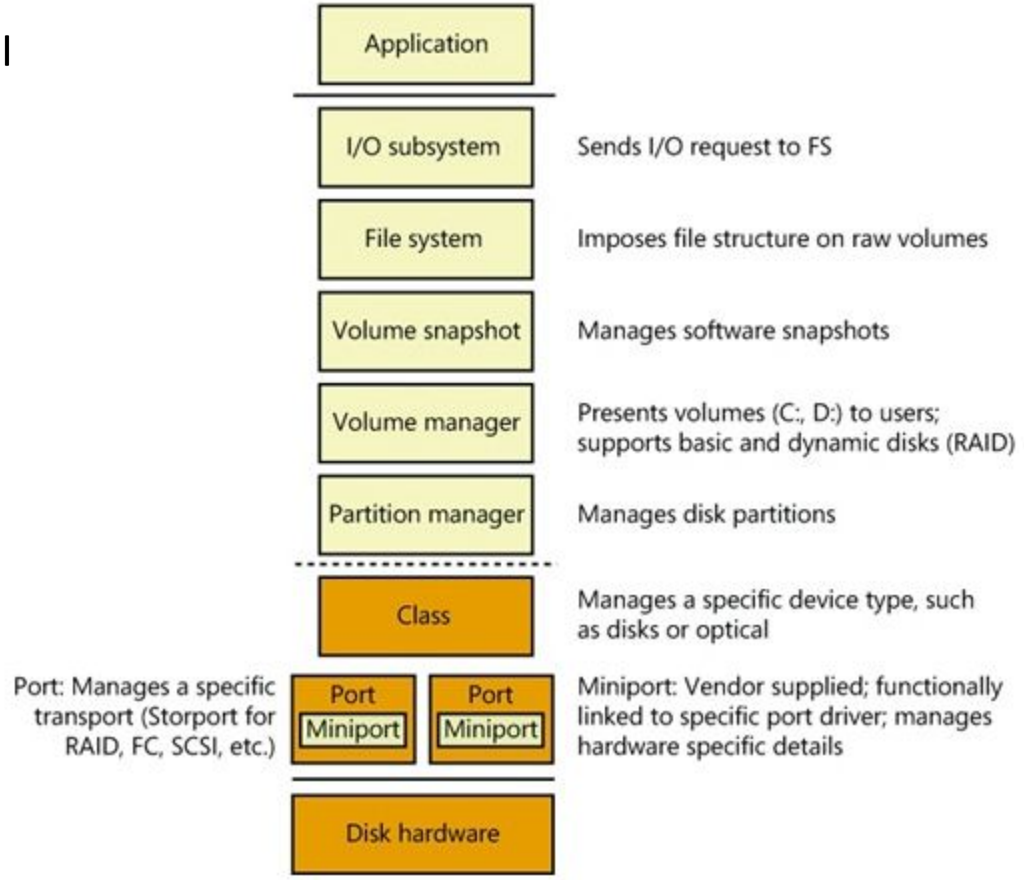


Figure 8: Key Selection Example

TCG Storage Security Subsystem Class (SSC): Key Per I/O (trustedcomputinggroup.org)

OS Driver Stack

- Key Per I/O requires changes to the layers in the storage stack.
- Need the ability to use the Key Tag for every I/O request.
- Note: Similar changes in the I/O stack was implemented to use PI fields (Protection Information fields)



[Disk Drivers - Windows® Internals, Sixth Edition, Part 2 [Book] (oreilly.com)]()

Solution - Conclusion

- Concepts of confidential compute can be extended to storage devices
  - At a granular level
- Data storage technologies provide mechanisms to control media encryption keys using industry standard mechanisms
  - NVMe protocol
  - TCG Key Per IO SSC
  - Requires changes to the I/O stack
- Enhances security
  - Enhances data erasure capability
  - Granular data erase
    - Per VM
    - Per user

Backup

- Backup

Granularity of confidentiality – problem statement

- The drives are “secured” at the granularity of entire drives
  - Multi Terabyte capacity per drive
  - A single drive may store data for multiple VMs and multiple containers
- SEDs can be partitioned into multiple encryption bands and each band can be secured separately
  - NVMe drives can support multiple Name Spaces and each name space can be its own encryption band
  - However, the provisioning and access controls to the encryption bands are static in nature
- This does not provide the granular and dynamic data confidentiality requirements for data stored on storage devices

Solution

- Key Per I/O specifications from TCG and NVMe allows for the host to insert a media encryption key to the drive
- Drive does not generate the media encryption key
    - Potentially a new one on a per I/O basis

  Media encryption key at the granularity of data range for a given virtual machine
    - Media encryption keys can be specified at the granularity of LBA ranges

- The media encryption key is stored in volatile memory

- Enhancing Data Encryption Capabilities in the Data Center with the NVM Express® Key Per I/O Feature - NVM Express
- Specifications - NVM Express