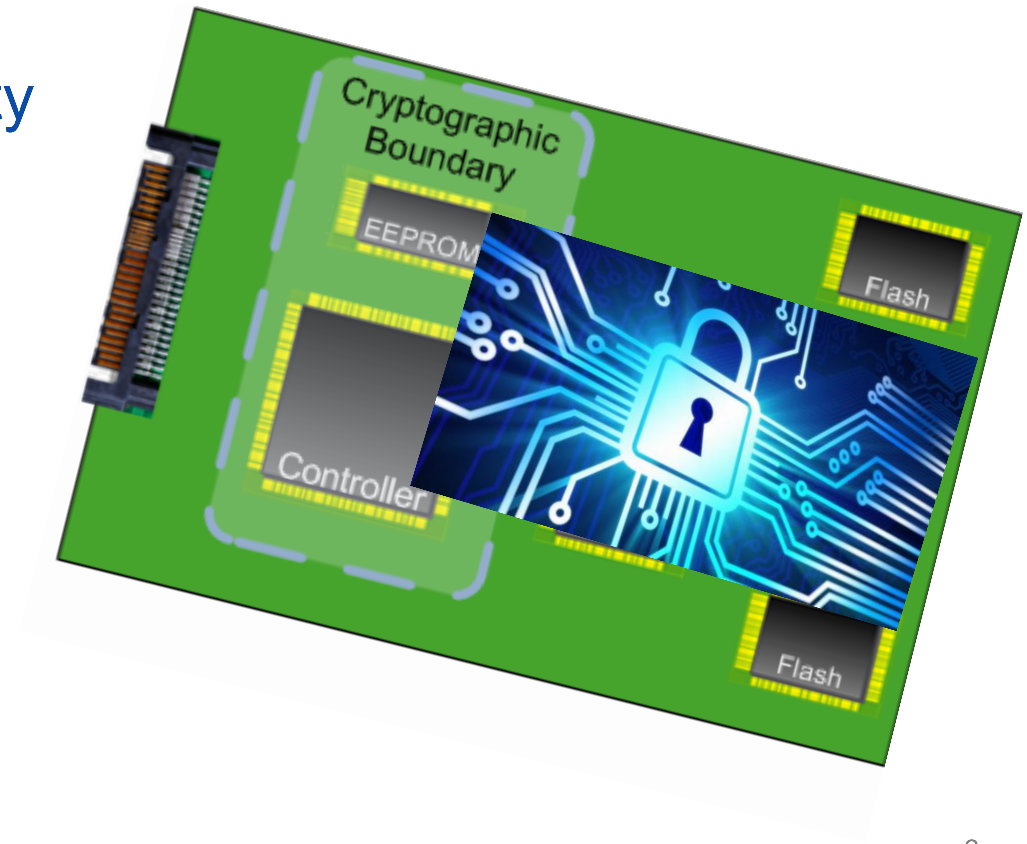
Securing the SSDs – NVMe Controller Encryption

Radjendirane Codandaramane
Microsemi Corporation

August 9th, 2016

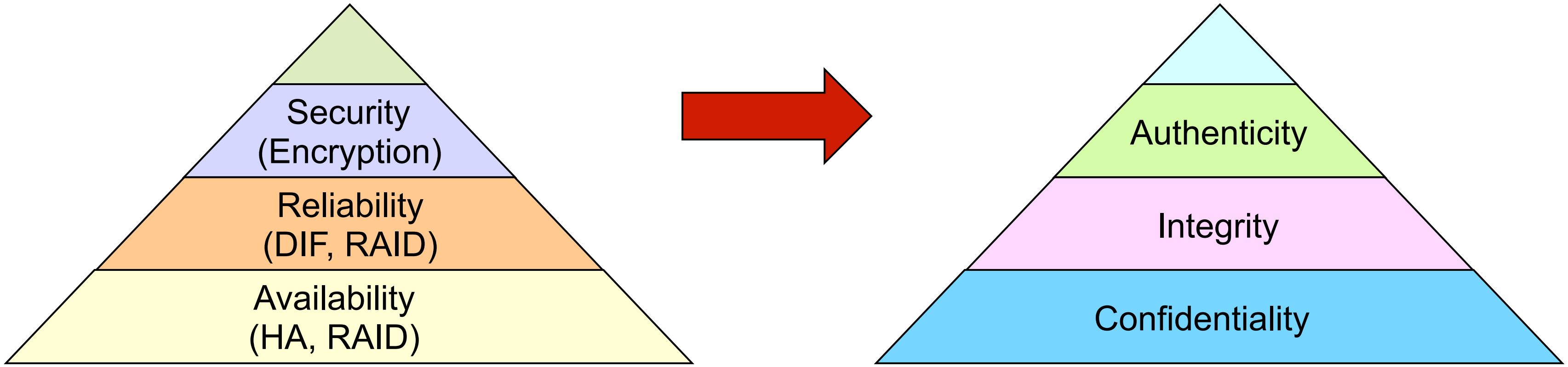
Contents

- Enterprise storage security needs
- Data-at-Rest encryption
- Encryption solution space
- Security features
- NVMe SSD example
- FIPS and design considerations



Enterprise Storage Requirements

Data Protection Requirements | Security Requirements

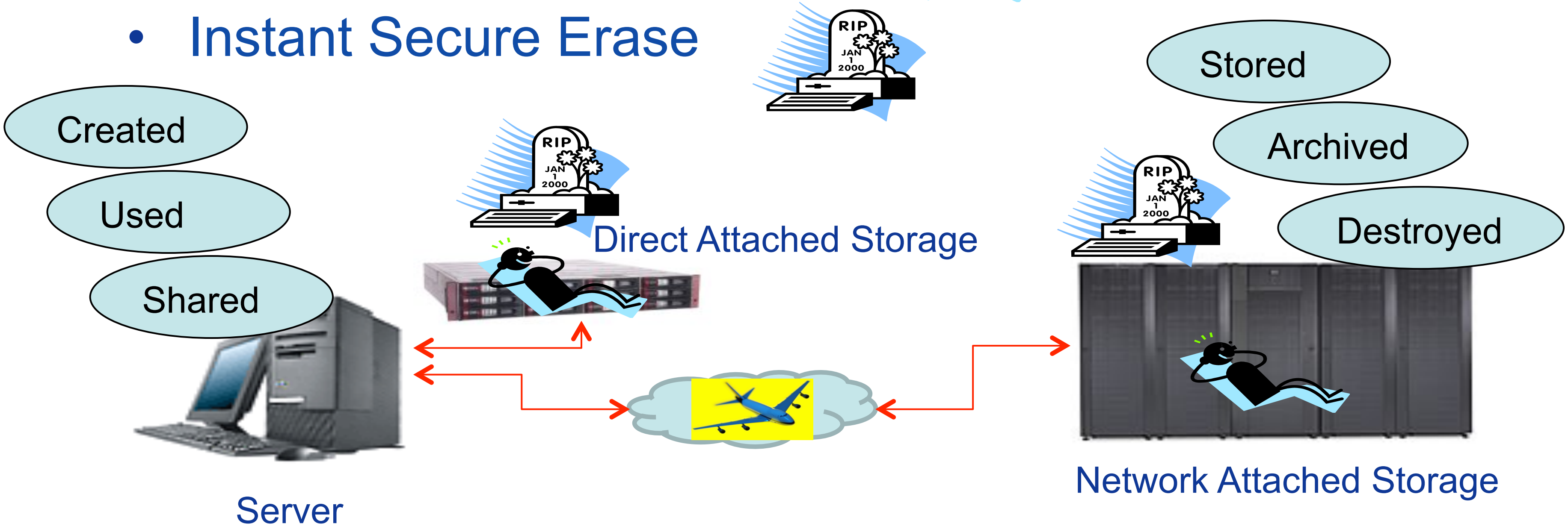


Performance, Port Density, Power, Price

- Protection against path failures through RAID, High Availability
- Protection against data corruption through RAID, DIF
- Protection against data mishandling through encryption

Encryption Types

- Data-in-Flight/Data-in-Transit Protection 
- Data-at-Rest Protection 
  - Instant Secure Erase



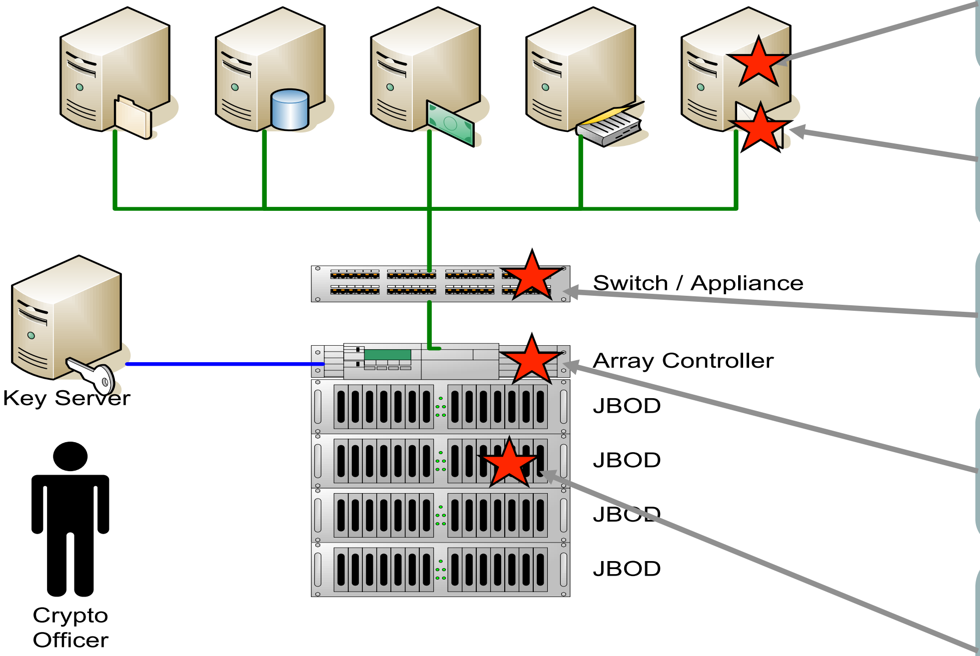
Drive For "Data-at-Rest" Encryption

Storage Drivers	Information Attacks	Regulatory Compliance
Information Sharing	Insiders	Industrial
Consolidation	End of Life Disposal	Federal
Outsourcing	Data Breaches	Local

Increasing Amount of Vulnerable Data* x *Multiple Threats* x *Increased Liability and Penalties

= Enterprise Data, Money, and Brand at Risk

Encryption Solution Space



Software Encryption

HBA Encryption

Appliance Encryption

Controller-Based Encryption (CBE)

Drive Encryption (SED)

Drive Interfaces – PCIe/NVMe, SAS, SATA

Enterprise Storage Encryption Features

Performance

- Must encrypt/decrypt data without impacting I/O performance

Cost-effective

- Affordable upgrade to existing storage installations

Flexibility

- Support for different block sizes, key granularities (drive, LUN, LBA etc.)

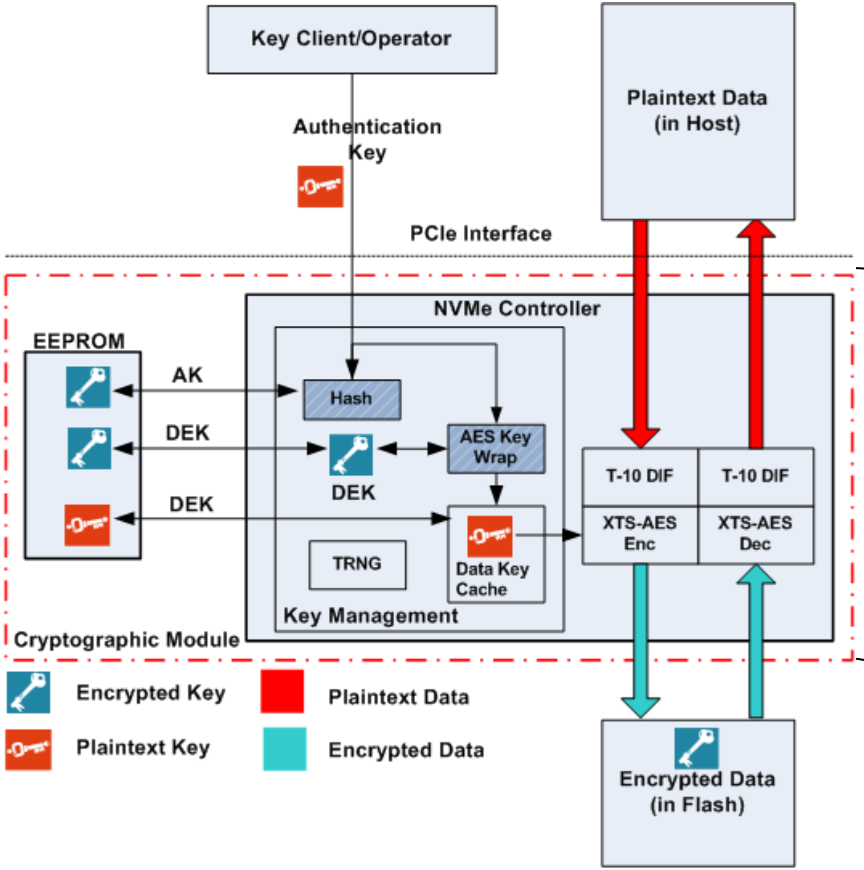
Reliability

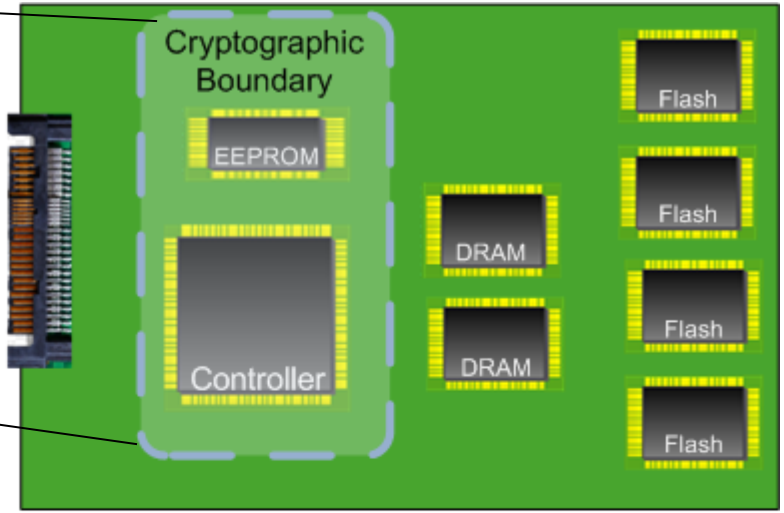
- Must provide means to ensure that data was encrypted and decrypted properly
- Must ensure data protection

Standards-compliant

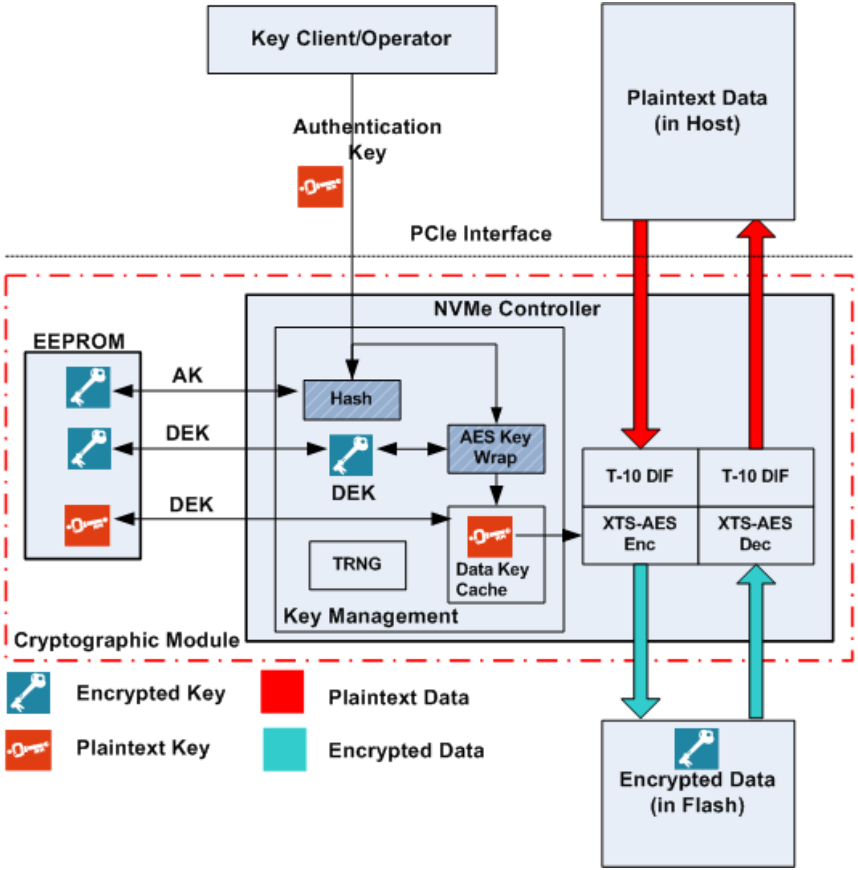
- Must meet the needs of applicable industry standards (PCI, HIPAA, etc.)
- FIPS 140–2, IEEE 1619
- TCG Enterprise, Opal, Opalite, Pyrite

NVMe SSD Example





NVMe SSD Encryption Example



- Data Encryption Key (DEK) or Range Key
  - Used to encrypt all data
  - Generated within the drive based on a TRNG
  - DEK is stored securely within the drive
- Authentication key (Range PIN):
  - Used to unlock the drive
  - Hash of this key is stored inside the drive
- At setup

  The drive generates a random range key for each range (never leaves the drive)

  Host generates a random 32B range PIN for each range and sends to the drive

  The drive wraps range key with range PIN and drive ID and stores range key blob in the drive
- At boot

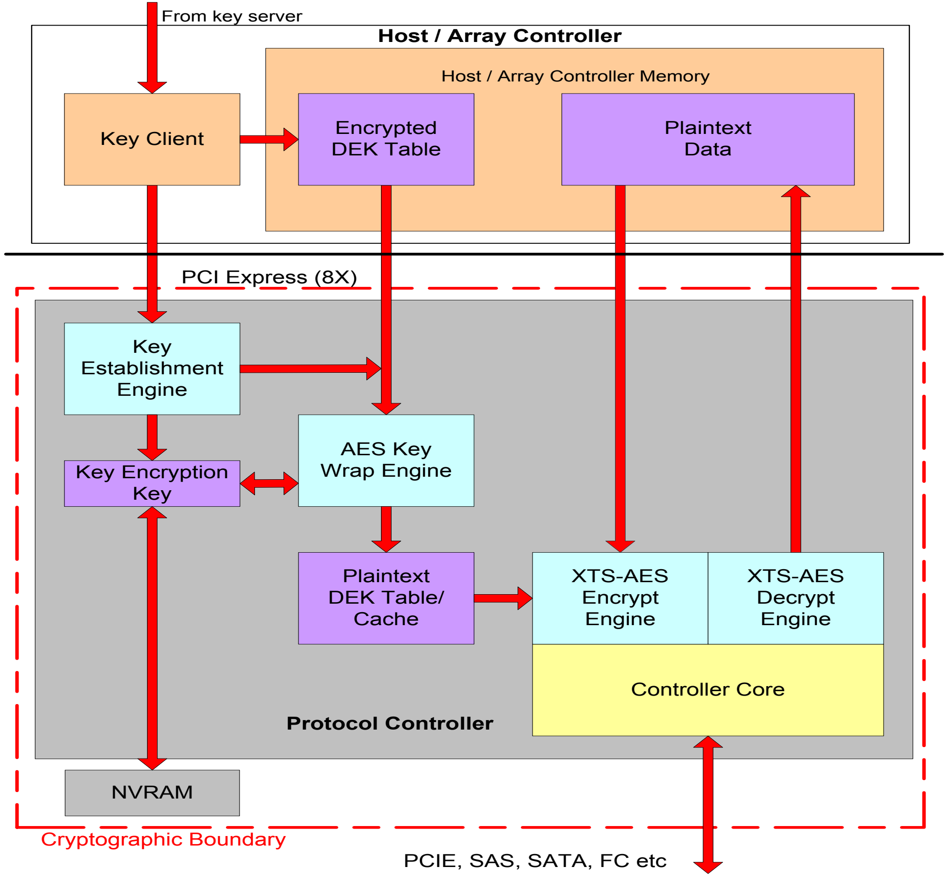
  Host sends 32B range PIN to the drive

  The drive verifies the range PIN

  If successful, then the drive is unlocked and ready

Controller-Based Encryption Example

- NVMe is evolving to fabric topology and becoming scalable like SAS/ SATA SSDs
- NVMe JBOF and RAID are on the horizon!
- Controller-based encryption is media independent



Encryption Solution Comparison

Solution	Pros	Cons
Self Encrypting Drive (SED)	• Integrated key generation	• Low security (keys and data stored in the drive) • Limited vendors and compatibility
Controller-Based Encryption (CBE)	• Encrypt any HDD/SSD • Cost-effective • High security (keys and data are separated) • Flexible key assignment (granularity of 1 key per HDD/SSD, LUN, LBA, I/O)	• Requires key manager

FIPS 140-2 Levels and Requirements

Category	Level 1	Level 2	Level 3	Level 4
Cryptographic module	Cryptographic boundary definition			
Ports and interfaces	Interfaces definition		Data paths logically separate	
Roles, services and authentication	No auth.	Roles based	ID-based authentication	
FSM	Define operational states			
Physical security	Production	Tamper evidence	Tamper response	EFP/EFT
Operational environment	Single user	EAL2 OS	EAL3 OS	EAL4 OS
Key management	Plaintext manual entry		Encrypted manual entry	
EMI/EMC	FCC Class A		FCC Class B	
Self-Tests	Power-up and conditional tests			
Design assurance	CM system	Secure dist.	High-level lang.	Extensive doc.
Mitigation of other attacks	Threats not covered by requirements			

Design for FIPS Considerations

- NIST Known-Answer-Test (KAT) vectors
- Method to prove encryption engine is working
- Self-test
  - **Power-up self-test and on-demand self-test**
    - Resetting, rebooting, and power cycling are acceptable means for the on-demand initiation of power-up tests
    - Implement a method to invoke self-tests
- Error injection
  - Method to invoke negative test cases
  - After error injection, the encryption functionality is disabled
- Physical security
  - No access to critical security parameters through debug interfaces
  - Zeroization

Summary

- Data storage security in enterprises is now a necessity
- Data-at-Rest encryption is the easiest way to safeguard data
- PCIe/NVMe SSD encryption can be implemented inside or outside the drive (SED, CBE)
- Keep in mind the design considerations for FIPS from the beginning!

Flash Memory SUMMIT

Microsemi
Power Matters.™

References and Resources

NIST: http://csrc.nist.gov/

- FIPS 197 AES Specification
- FIPS 140-2 Cryptographic Module Validation Program

IEEE 1619: http://siswg.org/

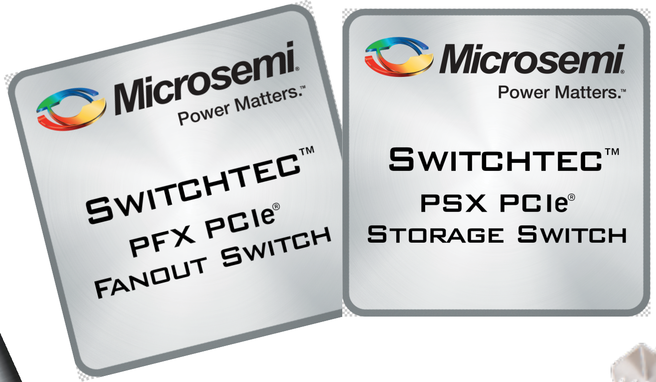
- 1619 Architecture for Encrypted Shared Storage Media (XTS-AES)

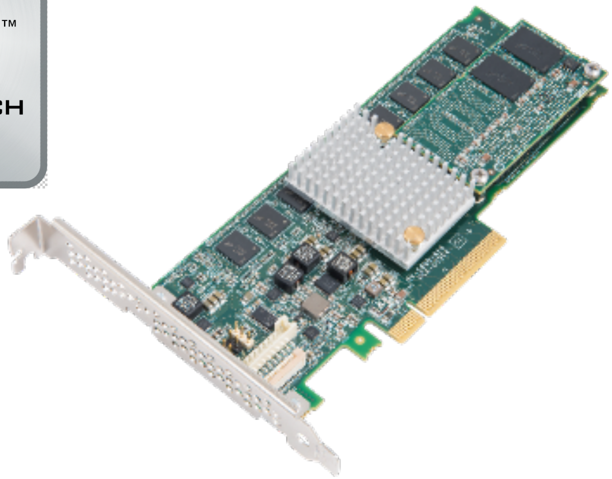
NVM Express: http://www.nvmexpress.org/

Trusted Computing Group: http://www.trustedcomputinggroup.org/

Flash Memory SUMMIT

Microsemi
Power Matters.™





THANKS!

Come and visit us at booth #213

www.microsemi.com