Security of Data on NVMe over Fabrics, The Armored Truck Way

Live Webcast
May 12, 2021
10:00 am PT / 1:00 pm ET
Today’s Presenters

- John Kim
  SNIA NSF Chair
  NVIDIA

- Claudio DeSanti
  Dell Technologies

- Nishant Lodha
  Marvell

- Hrishikesh Sathawane
  Samsung

- Eric Hibbard
  Samsung
SNIA-at-a-Glance

180 industry leading organizations

2,500 active contributing members

50,000 IT end users & storage pros worldwide

Learn more: snia.org/technical  
@SNIA
SNIA Legal Notice

- The material contained in this presentation is copyrighted by the SNIA unless otherwise noted.
- Member companies and individual members may use this material in presentations and literature under the following conditions:
  - Any slide or slides used must be reproduced in their entirety without modification
  - The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA.
- Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.
Storage Security Overview

Nishant Lodha
What do customers expect from NVMe over Fabrics?

- Operational simplicity
- Performance
- Applications
- Scale
- Security
- Cost
Data Protection vs. Security

There is often confusion between storage/data security and protection.

**Data security** refers specifically to measures taken to protect the integrity of the data itself.

**Data security** primarily involves keeping private information out of the hands of anyone not authorized to see or modify it.

Unauthorized access and access control, auditing
Intentional or accidental loss/corruption of sensitive data

**Data Security** measures include encryption of data both at rest and in motion, physical and logical access controls that prevent unauthorized access etc.

**Data Protection**, refers to the mechanism of making copies of your data to restore in the event of a loss or corruption.
## Datacenter Security and Standards

### Incidents and Security Standards

<table>
<thead>
<tr>
<th>Security Incidents</th>
<th>Storage Security Standards</th>
<th>Standards Regulations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Aurora 2010</td>
<td>TCG Opal 2.01 2015</td>
<td>FIPS 197 2015</td>
</tr>
<tr>
<td>Supermicro Hack 2015</td>
<td>OCP Security 2018</td>
<td>Common Criteria 2017</td>
</tr>
<tr>
<td>Lojax Malware 2018</td>
<td>Google OpenTitan 2019</td>
<td>FIPS 140-3 2019</td>
</tr>
<tr>
<td>Intel IceLake</td>
<td></td>
<td>NIST SP800 Storage 2020</td>
</tr>
</tbody>
</table>

- Standards, Security threats growing in past 10 yrs.
- New Security Standards organizations emerged

### Datacenter Security Considerations

- **Data in Flight**: Network security (especially applicable for shared infrastructure)
- **Data at Rest**: Against theft of data or keys, and ransomware (esp. SSD media and key encryption with SED)
- **HW Root of Trust**: Dedicated security engine to ensure Secure Boot, Secure FW, and Key Management across all peripherals
Storage Fabrics: Where to Secure – 10K View

Intra Rack / Smaller Scale
- NVMe/RoCE
- GPU
- NVMe/RoCE/IB
- Storage Array

Cross Rack / Large Cluster
- NVMe/TCP/RoCE/FC
- Storage Controller
- JBOF

Remote Datacenter(s)
- NVMe/TCP
- NVMe/TCP/RoCE/FC
- Compute
- NVMe/RoCE/IB
- GPU
Drivers for Storage Security

Implementing a “ZERO Trust” framework requires advanced technologies

Healthcare
Finance
Defense
Government

Sensitive Verticals

Multi-tenancy
DR/Cloud Storage
Malicious Insiders

New Deployment Use Cases

HIPAA
GDPR
ISO270001

Regulatory

Sensitive Verticals
Potential DC NVMe-oF Security Threats

- Sniffing Storage Traffic
- Storage Masquerading
- Ransomware
- Session Hijacking

Must secure NVMe payloads in flight and rest
Securing Storage Area Networks
Focus on NVMe-oF

Claudio DeSanti
# SAN Protocols - Security Mechanism Comparison

<table>
<thead>
<tr>
<th>Storage Endpoint Authentication</th>
<th>iSCSI</th>
<th>Fibre Channel</th>
<th>NVMe over Fabrics/IP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CHAP (strong secret)</td>
<td>DH-CHAP (strong secret)</td>
<td>DH-HMAC-CHAP (strong secret)</td>
</tr>
<tr>
<td></td>
<td>SRP (weak secret, e.g., password) [not used in practice]</td>
<td>FCPAP (weak secret, e.g., password)</td>
<td></td>
</tr>
<tr>
<td>Secure Channel (authenticated encryption &amp; cryptographic integrity)</td>
<td>IPsec (e.g., in security gateway)</td>
<td>FC ESP_Header (IPsec-like)</td>
<td>TLS (pre-shared key) – for TCP only (not usable with RoCEv2 or iWARP)</td>
</tr>
<tr>
<td></td>
<td>FC EAP (strong secret)</td>
<td>FCAP (certificates)</td>
<td>IPsec is also an option</td>
</tr>
</tbody>
</table>

Secure Channel (authenticated encryption & cryptographic integrity)
NVMe-oF Authentication Example

1. A TCP session is established
2. The Connect exchange is performed to set up NVMe queue and associate host to controller
3. The host performs an authentication transaction with the controller to authenticate the end-points
4. Queue is ready for subsequent operations
NVMe-oF Authentication: DH-HMAC-CHAP Protocol

- Defined in TP 8006
- Based on keys that need to be different for each NQN
- Challenge/response protocol: CHAP
  - the authenticator sends a challenge $C$;
  - the responder computes a response $R = \text{Hash}(C \mid\mid \text{key}_{\text{responder}} \mid\mid \text{other things})$;
  - the authenticator verifies the response (or delegates verification)
- DH-HMAC-CHAP: Strengthened version of CHAP
  - DH: Diffie-Hellman, adds (optional) key exchange to frustrate eavesdroppers
  - HMAC: Hashed MAC, uses secure hash twice to improve security
- Bidirectional authentication
DH-HMAC-CHAP Protocol: Bidirectional Authentication

Unidirectional challenge/response protocol
- Controller C sends a challenge \( C_1 \)
- Host H computes a response \( R_1 = \text{HMAC}(K_h, C_1 || \text{other things}) \)
- Controller C verifies the response
- Unidirectional authentication (controller authenticates host)

Getting bidirectional authentication
- H sends a challenge \( C_2 \)
- C computes a response \( R_2 = \text{HMAC}(K_c, C_2 || \text{other things}) \)
- H verifies the response
- Unidirectional authentication (host authenticates controller)

Verification:
- Controller computes \( R_1' \) and check if it matches the received \( R_1 \)
- Host computes \( R_2' \) and check if it matches the received \( R_2 \)

**AUTH_Negotiate**
(T_ID, SC_C, AuthID, HashIDList, DHgIDList)

**DH-HMAC-CHAP_Challenge**
(T_ID, HashID, DHgID, S_1, C_1, g^x mod p)

**DH-HMAC-CHAP_Reply**
(T_ID, R_1, g^y mod p, [S_2, C_2])

**DH-HMAC-CHAP_Success1**
(T_ID, [R_2])

**[DH-HMAC-CHAP_Success2]**
(T_ID)

**DH-HMAC-CHAP_Success2**
(T_ID)

**Authentication Responses**
- \( K_S = H((g^x \mod p) \cdot g^y \mod p) \)
- \( C_{a1} = (DHgID == 0) \) ? \( C_1 : \text{HMAC}(K_S, C_1) \)
- \( R_1 = \text{HMAC}(K_h, C_{a1} || S_1 || T_{ID} || SC_C || \text{"HostHost"} || NQN_h || 00h || NQN_h) \)
- \( K_S = H((g^y \mod p) \cdot g^x \mod p) \)
- \( C_{a2} = (DHgID == 0) \) ? \( C_2 : \text{HMAC}(K_S, C_2) \)
- \( R_2 = \text{HMAC}(K_c, C_{a2} || S_2 || T_{ID} || SC_C || \text{"Controller"} || NQN_c || 00h || NQN_c) \)
## SAN Protocols - Security Mechanism Comparison

<table>
<thead>
<tr>
<th>Storage Endpoint Authentication</th>
<th>iSCSI</th>
<th>Fibre Channel</th>
<th>NVMe over Fabrics/IP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Storage Endpoint Authentication</strong></td>
<td>CHAP (strong secret)</td>
<td>DH-CHAP (strong secret)</td>
<td>DH-HMAC-CHAP (strong secret)</td>
</tr>
<tr>
<td></td>
<td>SRP (weak secret, e.g., password) [not used in practice]</td>
<td>FCPAP (weak secret, e.g., password)</td>
<td></td>
</tr>
<tr>
<td><strong>Secure Channel</strong></td>
<td>IPsec (e.g., in security gateway)</td>
<td>FC ESP_Header (IPsec-like)</td>
<td>TLS (pre-shared key) – for TCP only (not usable with RoCEv2 or iWARP)</td>
</tr>
<tr>
<td>(authenticated encryption &amp; cryptographic integrity)</td>
<td></td>
<td></td>
<td>IPsec is also an option</td>
</tr>
</tbody>
</table>
Secure Channel: TLS

- **TLS (Transport Layer Security):** Widely used secure channel protocol
  - Secure channel = authentication, confidentiality, cryptographic integrity (primary properties)
  - Typical (web) usage: Server uses certificate with TLS, client authenticates after TLS setup (e.g., TLS-protected HTTP)

- **TLS versions:**
  - TLS 1.0 and 1.1: Obsolete - should not be used
  - TLS 1.2: Baseline TLS version, widely implemented and used, getting replaced by TLS 1.3
  - TLS 1.3: New version, complete protocol redesign (TLS 2.0 in practice), usage rolling out
    - Support available in some security libraries (e.g. OpenSSL & LibreSSL), expanding to more

- **TLS 1.3 specification for NVMe-oF/TCP:** completed in TP 8011
  - TLS not specified for other NVMe-oF IP-based protocol (i.e., RDMA, e.g., RoCEv2)
  - Based on pre-shared keys (PSKs)

- **NVMe-oF/TCP TLS 1.2:** discouraged by TP 8011
  - Usage was specified by NVMe-oF 1.1 standard
Why TLS 1.3?

**TLS 1.2**

- IANA TLS registry has 300+ cipher suite code points
  - Uncertain security properties, difficult interoperability
- Encryption starts late in the handshake
  - Client cert and target site are sent in the clear
  - Poor privacy
- Many features with known security flaws

**TLS 1.3**

- 5 cipher suites, all with PFS and modern algorithms
  - Consistent security properties
- Encryption starts as early as possible, hiding content length
  - Minimal set of cleartext protocol bits on the wire
  - Less user information visible to the network
- All those features are omitted from 1.3
TLS 1.3

1. A TCP/TLS session negotiation is performed and a secure channel is established
2. The Connect exchange is performed to set up NVMe queue and associate host to controller
3. Secure channel and queue are set up, ready for subsequent operations
TLS Credentials

- TLS secure channel for NVMe-oF/TCP is based on pre-shared keys (PSKs)
  - In order to authenticate and establish a secure channel between themselves, two NVMe entities need to be configured with the same PSK
  - This can lead to a deployment option called ‘group PSK’: all NVMe entities share the same PSK
  - Big security concern (compromising a single node may allow an attacker to access all secure channels)
  - The proper way would be to have a PSK per each pair of entity that can communicate ($n^2$ problem)

- Authentication protocols to the rescue
  - Upon successful completion of an authentication exchange, the two involved NVMe entities generate an ephemeral shared session key (e.g., a ‘PSK’ computed on the fly)
  - The TLS negotiation can then be performed using a PSK derived from that shared key
    - No more need for ‘group PSK’
  - Implementation result: the TCP connection begins unsecured and then transitions to secured
    - Opportunistic TLS
  - Linear problem (not anymore $n^2$): need just one secret per entity
Authentication Followed by TLS

1. A TCP session is established
2. The Connect exchange is performed to set up NVMe queue and associate host to controller
3. The host performs an authentication transaction with the controller, transaction that generates a pre-shared key PSK between host and controller
4. The pre-shared key PSK is used to perform a TLS negotiation and to establish a secure channel
5. Secure channel and queue are set up, ready for subsequent operations
NVMe-oF Security in Action
A Use Case in E-SSDs

Hrishikesh Sathawane and Eric Hibbard
NVMe Over Fabrics Architectures for Disaggregated Storage

**JBOF with x86**
- **Pros**: Current production, Established Ecosystem, PCIe SSDs in production
- **Cons**: BW bottleneck, Added PCIe latency, High power

**JBOF with SmartNIC**
- **Pros**: Finding some use cases, Emerging Ecosystem, PCIe SSDs in production, Lower Power
- **Cons**: Added PCIe latency, Needs SmartNIC on both sides

**E-BOF**
- **Pros**: BOM cost savings, Lower latency/power, True disaggregation possible
- **Cons**: No Ecosystem yet, E-SSDs are in PoC

---

**Production**

**Engage-Short Term**

**Engage-Long Term**
E-SSD and the Hype Cycle

E-SSD is in the early phase of the hype cycle.
Ethernet SSD Introduction and Value Proposition

- Ethernet SSD can be one of the solution for scalable performance of JBOF
  - Also targeting for TCO saving through replacement from high CPU & BOM to Ethernet switch
  - Samsung is also continuing to study the architecture, ecosystem and benefits of Ethernet SSD

---

**JBoF**
- Limited performance
- High CPU power / cost

**Ethernet BOF (EBOF)**
- Scalable performance
- Storage service should be moved to upward

---

**NVMe-oF Capable JBOF**

- Remote Host
- Compute
- TOR
- NIC
- x86
- PCIe Switch
- PCIe SSD
- Performance Bottleneck
- Storage Service

**Ethernet BOF (EBOF)**

- Remote Host
- Compute
- TOR
- E-SSD
- Ethernet Switch
- Scalable & Flexible Performance
- Native NVMe-oF Capable
- High B/W for AI/ML applications
Advantage of NVMe-oF SSD

- NVMeoF JBOF can solve performance, scalability, and flexibility
  - Scalable and flexible data center solution through Ethernet-only infrastructure

**Conventional SSD System**

- Compute
- Non-scalable Storage Controller
  - PCIe single root constraint
  - Limited Bandwidth
- Limited Bandwidth
- PCIe & Ethernet constraint

**NVMeoF SSD System**

- Compute
- Virtualized Server Pool
- NVMeoF E-BOF
- Dual-port E-SSD

![Diagram comparing conventional and NVMe-oF SSD systems](image)

PCIe Switch

Dual-port NVMe SSD

E-SSD

Ethernet Switch

TOR

ToR Switch

x86

Compute

Application Server

Virtualized Server Pool

E-SSD

Ethernet

PCIe Switch

PCIe SSD

NIC

TOR

ToR Switch

x86

Compute

Non-scalable Storage Controller

PCIe single root constraint

Limited Bandwidth

PCIe & Ethernet constraint

Conventional SSD System

NVMeoF SSD System
Disaggregated Architecture

- NIC card's essential features are offloaded to E-SSD.
- Storage controller can communicate with multiple E-BOFs as it is done with JBOFs.
  - Connection and discovery are added to maintain connection in NVMe-oF specification.
  - NVMe I/O command processing is equal as PCIe based NVMe SSD for E-SSD.
  - Additional target configuration may not be needed, since configured information are saved in E-SSD.
- SW modification
  - Schema of JBOF for Redfish can be changed.

NIC + CPU
- IP address configuration
- Manage Connections
- Network Management

PCIe SSD
- NVMe I/O Processing

BMC (Expected)
- E-SSD IP address configuration

E-SSD (offloaded)
- IP address configuration (manual)
- Manage Connections
- Network Management
- NVMe I/O Processing
Disaggregated Architecture

- NIC card’s essential features are offloaded to E-SSD.
- Storage controller can communicate with multiple E-BOFs as it is done with JBOFs.
  - Connection and discovery are added to maintain connection in NVMe-oF specification.
  - NVMe I/O command processing is equal as PCIe based NVMe SSD for E-SSD.
  - Additional target configuration may not be needed, since configured information are saved in E-SSD.
- SW modification
  - Schema of JBOF for Redfish can be changed.

NIC + CPU
- IP address configuration
- Manage Connections
- Network Management

PCIe SSD
- NVMe I/O Processing

BMC (Expected)
- E-SSD IP address configuration

E-SSD (offloaded)
- IP address configuration (manual)
- Manage Connections
- Network Management
- NVMe I/O Processing
Questions
Other Resources

- Webcast: NVMe over Fabrics: Looking Beyond Performance Hero Numbers
- Multiple resources: SNIA Geek Out on NVMe-oF
- Blog: NVMe over Fabrics for Absolute Beginners
After this Webcast

- Please rate this webcast and provide us with your feedback
- This webcast and a copy of the slides will be available at the SNIA Educational Library https://www.snia.org/educational-library
- A Q&A from this webcast, including answers to questions we couldn’t get to today, will be posted on our blog at https://sniansfblog.org/
- Follow us on Twitter @SNIANSF
Thank You