Command Security via SAs

r1 - revised based on discussions during CAP Security working group

T10/07-149r1
Goals

★ Per-Command Security

★ Fine-grained Reservations and/or Access Controls
  + Tied to TBD Entities Inside Application Client
  + Greater Command-Access Flexibility

★ Consideration for OS Performance
Securing Commands

(Comparison of Approaches)

07-069

Initiator

CDB

Capability & ICV

Target

122 bytes

07-149

Initiator

CDB

ID

ICV

Target

8 bytes

64 bytes
ID Options
(ICV is ICV …)

✔ SA Identifier
  ✴ AC_SAI + DS_SAI
  ✴ DS_SAI is actually enough
  (shrinks ID size to 4 bytes)

✔ OS-Specific
  ✔ Setup as Synonym for SA ID
    (during SA Creation, i.e., validated)
  ✔ Tied to Program Running on OS
OS-Specific ID
(Conceptual Protocol)

Program | OS | Target

Request/Return ID

Set up SA with ID as Synonym (Passthrough)

Non-Passthrough I/O

OS Adds ID to CDB

Net Result – Program is Identified

✔ When the OS stops adding a given ID is the crucial success factor
OS-Specific ID
(What Might Work)

- Process ID
- Image Count

☆ OS must deliver I/O completion to right image
☆ This kind of information seems likely to be available in some parts of the driver stack
☆ Your mileage may vary
SA Extensions
(Extensions to SA Creation)

- Authentication Required (usage based)
- Synonym Setup

☆ Commands Controls … 3 Lists
→ Allowed When SA ID Present
→ Allowed for Others SAs
→ Allowed for Everything Else

✓ Checked Against Permissions for Authenticated SA Creator
Command Controls
(Preliminary List Format Ideas)

🌟 Allowed Bit Mask (1 bit for each OP code)
🌟 Exceptions Descriptors
  ➔ Service Actions
  ➔ Mode Page Codes
  ➔ Mode Page Changeable
  ➔ Log Page Codes
  ➔ Diagnostic Page Codes
  ➔ Reservations Modes
  ➔ …
Command Controls
(Preliminary Format Ideas)

🌟 Prohibit All MAINTENANCE OUT except SET IDENTIFYING INFORMATION

🌟 Allow All MODE SELECT(10) except Control Mode Page

🌟 Prohibit All Reservations except All Registrants
Inventive Enough?

Too Much?

→ Is putting the command selection burden on the Initiator right?

→ Is the Allow/Prohibit Model Flexible Enough?

→ Is there more that can be done for multiple concurrent SAs?

→ How do ICVs fit into this picture?
Capabilities Too?

★ Could SA Extension be Capability?
(instead of bulky bit/exceptions format)

✦ Somehow push to 1 Authentication
  (the Security Manager one)
✦ I_T Nexus ID?
✦ ICV?
Two Usage Models
(Good Reasons for Each)

- **Initiator**
  - Usage validation in Target
  - Decentralized Security
  - More Smarts in Target
  - Small Configurations

- **Target**

- **Initiator**
  - Usage validation in SM
  - Centralized Security
  - Less Smarts in Target
  - Large Configurations
Help!