• The information in this presentation represents a snapshot of work in progress within the DMTF.
• This information is subject to change without notice. The standard specifications remain the normative reference for all information.
• For additional information, see the DMTF website.
• This information is a summary of the information that will appear in the specifications. See the specifications for further details.
SPDM’s Overall Goals

• All SPDM features fall into at least one of these main goals:
  • Device Attestation
  • Securing Communication over the Wire

• Device Attestation
  • The ability to attest various aspect of a device such as firmware integrity and device identity.

• Securing Communication over the Wire
  • Provide the transport the ability to secure communication of any data over that transport.
SPDM Summary

• Version 1.0:
  • Measurement Support
  • Device Authentication

• Version 1.1:
  • Secure Session
    • Public Key Exchange
    • Symmetric Key Exchange
  • Mutual Authentication
SPDM 1.2 Feature Additions

- **Provisioning**
  - Allows installation of device certificate in manufacturing.

- **Certificates**
  - Allows for alias leaf certificates derived from device certificates.

- **Message Fragmentation**
  - Send large SPDM messages in chunks.

- **Miscellaneous:**
  - Added SM2, SM3, SM4 algorithms to supported list.
  - New OIDs added.
SPDM 1.2 Feature Deprecation

- Deprecating Basic Mutual Authentication
  - Removing mutual authentication in CHALLENGE and CHALLENGE_AUTH.
SPDM 1.2 Change Awareness

• Statement of Backwards Compatibility:
  • SPDM message format will maintain bit-wise and semantic compatibility for existing fields.
    • SPDM may append new fields to an existing message.
    • SPDM may make use of reserved values.
    • SPDM may deprecate a valid value.
  • SPDM may make operational changes to fix a security issue or strengthen the security posture of the operation even if they are technically incompatible.

• Therefore, SPDM 1.2 contains changes that may be deemed technically incompatible with prior versions.
  • Please see change notes at the end of DSP0274 1.2 for details.
Provisioning

- Allows for a device certificate (i.e., certificate slot 0) to be installed in a secured environment (e.g., manufacturing).

- New Request / Response
  - SET_CERTIFICATE / CERTIFICATE_RSP
    - Installs a certificate chain to the specified slot.
  - GET_CSR / CSR
    - Generates a certificate signing request to be signed by a certificate signing infrastructure.
Alias Certificates Support

• What is an Alias Certificate or Certificate chain?
  • They are dynamically generated, usually, on each device reset.
  • They are chained to the device certificate.
  • They are mutable.

• New Feature
  • Devices can generate alias certificate dynamically usually on device boot.
  • Alias certificates will be used as the leaf certificate instead of device certificates in all existing device authentication flow (i.e., CHALLENGE, KEY_EXCHANGE, GET_MEASUREMENT, etc...).
  • Device Certificates are usually static, immutable and hardware anchored.
SPDM Certificate Models

DeviceCert Model

Root CA

Intermediate CA

...

Device Certificate

AttestationCert Model

Root CA

Intermediate CA

...

Device Certificate CA

Alias Intermediate CA

...

Alias Certificate
Message Fragmentation – Chunks Transfer

- Allows a large SPDM message to be transferred in fragments (called chunks) to account for the receiving buffer size.

- New Request / Response:
  - CHUNK_SEND / CHUNK_SEND_ACK
    - Send a large SPDM Request in fragments.
  - CHUNK_GET / CHUNK_RESPONSE
    - Retrieves a large SPDM Response in fragments
**Send Large SPDM Request Flow**

**Requester**

- **SPDM Header**
  - Chunk 0
  - Chunk 1
  - Chunk 2
  - Chunk 3

**Responder**

- **CHUNK_SEND**
  - Handle 5
  - Chunk Sequence 0
  - Chunk Size 250 Bytes
  - Large Message Size 800 Bytes
  - Chunk 0 Data

- **CHUNK_SEND_ACK**
  - Handle 5
  - Chunk Sequence 0

- **CHUNK_SEND**
  - Handle 5
  - Chunk Sequence 1
  - Chunk Size 254 Bytes
  - Chunk 1 Data

- **CHUNK_SEND_ACK**
  - Handle 5
  - Chunk Sequence 1

- **CHUNK_SEND**
  - Handle 5
  - Chunk Sequence 3
  - Last Chunk
  - Chunk Size 42 Bytes
  - Chunk 3 Data

- **CHUNK_SEND_ACK**
  - Handle 5
  - Chunk Sequence 3
  - Last Chunk
  - Response: SET_CERTIFICATE_RSP

---

**Large SET_CERTIFICATE Message**

Total Message Size = 800

- Chunk Size is DataTransferSize minus the sizes of all the fields (except for SPDMchunk) of CHUNK_SEND_REQUEST. Thus, 266 - 12 = 254 bytes.

- Chunk Sequence 0 contains an extra field. Thus, the Chunk Size for the first chunk is 266 - 16 = 250 bytes.
Responder creates the MEASUREMENTS response with a total size of 1000 bytes. This is > 312 bytes.

Chunk Size is DataTransferSize minus the sizes of all the fields (except for SPDMchunk) of CHUNK_RESPONSE. Thus, 312 - 12 = 300 bytes.

Chunk Sequence 0 contains an extra field. Thus, the Chunk Size for the first chunk is 312 - 16 = 296 bytes.