802.1AR Annex-C Discussion

Paul Congdon

Boris Balachef

Objectives of Annex C

- To demonstrate how to implement .1AR using a TPM.
- To discuss how to support the goals and specific requirements of .1AR using the TPM
- TPM specs available at: www.trustedcomputinggroup.org/specs/TPM

Goals of .1AR

- Shipping with an initial credential
- Protecting the DevID and the binding to the device
- Ability for the owner of a device to add a locally significant DevID
- Ability to use a DevID in an authentication exchange

Requirements of .1AR-d1

- Storage for secret and credential
- RSA-2048 and/or ECC
- Hashing with SHA-256
- RNG with 128-bit strength
- Operations
  - Initialize (with added RNG entropy)
  - Enable/Disable
  - Key pair manipulation (create, add, delete)
  - Credential manipulation (insert, delete)
  - Signing

Goal: Shipping with an initial credential

- TPM ships with an Endorsement Key (EK) and Endorsement Credential, typically from manufacturer
- However EK can not be used as an identity for authentication (i.e. signing)
- To include a digital signature identity key at manufacturing time the manufacturer would have to take ownership of the TPM and generate and install another identity credential (i.e. IDevID).
- Control of the TPM would be maintained by the device operating environment and not exposed to the device owner.
- NOTE: TPM offers a means to install an IDevID post manufacturing by relying upon the trust of the EK

Goal: Protecting the DevID and the binding to the device

- A TPM can generate and/or protect cryptographic keys and control their use for digital signature operations.
- A TPM can prevent generated and/or protected keys from ever being available outside of the TPM itself.
- The cryptographic identity is bound to the device the TPM is physically embedded in.

Goal: Ability to add a locally significant DevID

- A TPM can generate an unlimited number of cryptographic keys and can protect and control the usage of those keys as well as generic credential data in an external memory

Goal: Ability to use the credential in an authentication exchange

- Usage of a TPM key in an authentication protocol requires compatibility of the cryptographic algorithms used.
- TPMs are being used in a number of authentication protocols today
- TPM supports:
  - RSA up to 2048 bit key sizes
  - SHA-1 hashing
  - RNG

Req: Storage for secret and credential

- This is a core function of the TPM. It supports the creation, import, and controlled usage of an unlimited number of asymmetric cryptographic keys in a single TPM

Req: RSA-2048 and/or ECC

- The TPM requires the support of the RSA algorithm up to 2048 bit key sizes.
- A TPM does not currently support ECC algorithms.

Req: Hashing with SHA-256

- A TPM currently supports SHA-1 as a required hash algorithm.
- Investigating if external SHA-256 hash may be passed to a RSASSAPKCS1v15_DER signature scheme

- Note: this may represent a compatibility issue

Req: RNG with 128-bit strength

- The TPM predates the NIST specifications quoted in the 802.1AR standard on 128-bit strength.
- The TPM includes a command that allows the addition of entropy to its RNG engine, allowing the seeding of the RNG to meet required strength.

Req Operation: Initialize

- TPM initialization includes the setting of a TPM ownership authorization data, used to control key TPM commands and functions.
- TPM initialization should only be done once at manufacturing time to avoid deleting the IDevID.
- Initialization can also include the input of RNG entropy.

Req Operation: Enable/Disable

- The TPM can be enabled and disabled

Req Operation: Key Pair Manipulation

- The TPM supports the creation, addition, and deletion of RSA key pairs up to 2048 bits in length
- The TPM supports the import of key pairs that could be generated securely outside of the TPM itself

Req Operation: Credential Manipulation

- Credentials may be inserted or deleted with or without the direct use of a TPM.
- The TPM provides minimal non-volatile storage, as such credentials and certificates will typically be stored on external storage media
- The TPM can be used to protect credentials stored on external media.

Req Operation: Signing

- The TPM specifies the use of RSA PKCS1v1.5SSA signature algorithm with SHA-1.
- The TPM further specifies how to format the input of the signature functions, which is key to achieving interoperability.
- Three signature schemes are defined:
  - TPM_SS_RSASSAPKCS1v15_SHA1
  - TPM_SS_RSASSAPKCS1v15_DER
  - TPM_SS_RSASSAPKCS1v15_INFO
- Investigating if TPM_SS_RSASSAPKCS1v15_DER is the appropriate function for the current .1AR signature function.