Objectives

• Provide a secure bi-directional mechanism for exchange of sensitive values (e.g., auth tokens) in the context of Capability Advertisements and Configuration Metadata

• Allow this mechanism to function without any required out-of-band communication via embedded cryptographic messages

• But also support reference to external services (e.g., HashiCorp Vault) in place of the in-band mechanism
MI.SecretStore

• Define the configuration for associated MI.SecretValue

  • An embedded message in CMS (RFC5652) format OR

  • A cleartext value for development and test environments OR

  • Either a reference to an external service

```json
{
  "secret-store-id": "store-1",
  "secret-store-type": "MI.SecretStoreTypeEmbedded",
  "secret-store-certificate-id": "cert-1",
  "secret-store-config": {
    "format": "cms"
  }
}

{
  "secret-store-id": "store-1",
  "secret-store-type": "MI.SecretStoreTypeEmbedded",
  "secret-store-config": {
    "format": "cleartext"
  }
}

{
  "secret-store-id": "store-2-vaultv2",
  "secret-store-type": "MI.SecretStoreTypeVault",
  "secret-store-config": {
    "endpoint": "https://vault.example.com/v1.secret",
    "version": 2,
    "namespace": "customer-1"
  }
}
```
**MI.SecretValue**

- The object container for embedded CMS values or for references to secrets stored on an external service

- Consists of a reference to the MI.SecretStore configuration and either a "secret-value" or "secret-path" property

- Used as a property value inside other MI or FCI objects which contain secret data
Communicates a PEM format X.509 cert to the other party for use in a MI.SecretStore of type MI.SecretStoreTypeEmbedded

```
{ "certificate-id" : "cert-1", "certificate-value" : "MIIDZTCCAK2gAIBA14UFokJ2xAzQxGxSbBhBuSbLpwMSLwADQYJKoZIhlvcaNAQELBQAwQgELMA4AUEwMCVwEDAOgBqGBAGMD0dU1nAExITTAT4FgqVBAaAMEIGldU6YbWV9TFD0ZGdyM4dUHR5TExG0DAwFovyM2 AZM5YMA0MDwNDQxMwFgXyAYMCWwEII1yIOZMNEIzAIcJ38vBvAATyA1TMwR4vDwVQnDIzDwHdJoyZy5l5hMSm5h5wVQQ7DQw58 JbnRlcn5ldCBxwRnoXRzI8F88e5BmG9OwggEiMA0GCSqGSIb3DQEBAUAA4LBmAqggAKAoIBAQCT110ybedJnjiqXO sLTz5HtpIysahNpKiCE1K8ByZ5l15cS9Hga2KzQ3V3Ntbj31samasHbgynsqqqs5KGuJzidB3/LJIAOB2z1YVFJEdLYQ QF33Do6pJz2RQcJFZ/BLQwEFZ5N5hrz/JVWB3jzpT1u3c5KXUCNfX3Jjk6gX7qVv5jHfcRf58d85HflgMxJpLs1w7YQgYDd5lXnP9cX5GQJYD6 VxkQ+/EX5C+EeGD0Q7WwOMADwM5E1FamilyBDJvYbJoysw7VM06Gy2QAZP1x82wzrzwLxvDHMLHnkM6Lc1e6b0vYB4PnDO1mIcL g9USi/y66581htx5CnmoZO7/y/gGquOi/2ywnm75n9F2OdAgM4AAGU1+UMR0Z6A1dDgDwQpO5b5b3jNlVGpE5r3bIE7id58 g535uAfeyVHSMDGAY5QzoD5YyLEqErlebiei65Bgs53uATChgNVRM88A8f8ETADAQfMA0GCSqGSIb3DQEBCwUA 4IBAQBUrn3yHbVWw/u/xj2k8pdTt8ekB0k0105lYeH55sRa6n31t3QYF9jOMBQ81nxFX0izGaUZDQXa0C0QwLbJwv unlimited use in a MI.SecretCertificate

Certificate:

Version: 3 (0x2)
Signature Algorithm: sha256WithRSAEncryption
Issuer: C = US, ST = Georgia, O = Internet Wigitz Pty Ltd
Validity Not Before: Jan 23 20:36:03 2023 GMT
Not After: Feb 22 20:36:03 2023 GMT
Subject: C = US, ST = Georgia, O = Internet Wigitz Pty Ltd
Subject Public Key Info:

Public Key Algorithm: rsaEncryption
RSA Public-Key: (2048 bit)

0a:86:6e:55:14:33:17:67:0b:38:04:44:50:08:00:
8b:0e:cf:f5:65:06:1d:db:ce:0e:93:06:0b:07:76:
c6:58:4b:a7:dc:oa:77:2a:c4:18:0f:00:64:3c:06:
43:97:77:ac:64:dc:e1:17:e5:51:01:00:e8:08:08:
e2:1f:2a:be:98:05:1e:00:2e:03:29:0a:58:0a:06:
3c:8f:22:00:04:08:04:23:67:0c:00:79:94:9d:79:
5c:dd
Exponent: 65537 (0x10001)

5898:3 extensions:
5898:3 Authority Key Identifier:


MI.SecretCertificate
```
FCI Wrappers

Allow use of MI.SecretStore and MI.SecretStoreCertificate from the Advertisement side for bi-directional exchange of secrets

```json
{
  "capabilities": [
    {
      "capability-type": "FCI.SecretStore",
      "capability-value": {
        "cert-id": "store-1",
        "secret-store-type": "MI.SecretStoreTypeEmbedded",
        "secret-store-config": {
          "format": "csm"
        }
      }
    }
  ]
}
```
Changes from Revision 00

• No longer marked as updating 8006 or 8008
• 1 Fix text incorrectly referencing a "subpart", reference to external documentation for HashiCorp Vault, workflow overview text cleanup
• 3.4 Added 'timeout' property to MI.SecretValue
  • The longest interval for which a secret may be cached before it must be retrieved again from the Store.
• General text cleanup, consistency (thanks Alfonso)