

SCMS Proof-of-Concept (POC) System

The SCMS POC was formally closed in October 2018, however the key concepts, systems architectures and interfaces developed during the POC development are still utilized by SCMS providers today.

Secure Credential Management System (SCMS) Proof of Concept

The initial SCMS Proof-of-Concept (POC) system was developed jointly through US DOT and the Crash Avoidance Metrics Partnership (CAMP). The SCMS POC was formally closed in October 2018, however the key concepts, systems architectures and interfaces developed during the POC development are still utilized by SCMS providers today. The following sections provide the systems architectures, SCMS processes and interface descriptions that are still relevant and useful to any organization that is implementing connected vehicle devices that interface with an SCMS.

SCMS PoC Environment

The SCMS POC software operated in three different environments (locations) for three different purposes. All environments have their own, independent Root CA.

[Environment Documentation](#)

Design Concepts, Policies and Procedures

The following information was developed throughout the life of the SCMS POC system. This covers SCMS design concepts as well as SCMS policies and procedures. This information is still relevant and covers technical functions and interfaces that operational SCMS systems still utilize.

- Key SCMS Concepts
- SCMS Use Cases
- Software Design Documentation
- Test Vectors
- Policies and Procedures

Design Concepts, Policies and Procedures

The following information was developed throughout the life of the SCMS POC system. This covers SCMS design concepts as well as SCMS policies and procedures. This information is still relevant and covers technical functions and interfaces that operational SCMS systems still utilize.

- Key SCMS Concepts

The following are key concepts of an SCMS System:

[SCMS Supported V2X Applications](#)

[SCMS Certificate Types](#)

[Device Hardware, Software and OS Security Requirements](#)

[Elector-based Root Management](#)

[Cryptography](#)

[End-Entity \(EE\) to Request Authority \(RA\) Communications – General Guidance](#)

[EE to SCMS Core Communications](#)

- SCMS Use Cases

The following pages are a hierarchy of requirements sorted by SCMS use cases.

[OBE, RSE, Common and Backend Use Cases](#)

[Use Case 2: OBE Bootstrapping](#)

The manual process for RSE bootstrapping is exactly the same as for OBE for at least the first year of CV pilot SCMS POC operations.

[Use Case 3: OBE Pseudonym Certificate Provisioning](#)

[Use Case 5: Misbehavior Reporting](#)

[Use Case 6: CRL Download](#)

[Use Case 8: OBE Pseudonym Certificate Revocation](#)

[Use Case 11: Backend Management](#)

[Use Case 13: RSE Application Certificate Provisioning](#)

[Use Case 16: RSE Application and OBE identification Certificate Revocation](#)

[Use Case 18: Provide and Enforce Technical Policies](#)

[Use Case 19: OBE Identification Certificate Provisioning](#)

[Use Case 20: EE Re-Enrollment](#)

● Software Design Documentation

● [Common Service View](#)

The Service View shows the architecture from the perspective of the services exposed by each component. Components interact with each other via service invocation. The service is therefore the point of contact between two collaborating components. A Service is a discrete piece of functionality that is made accessible for other components to consume over the network. While there are other types of services, we chose to design our solution in the style of [RESTful](#) Web Services, which provides the following advantages:

- Relatively simple connector implementation with plenty of tooling and libraries to support development
- Easier to introduce proxies, gateways, caches, and load balancers without affecting involved components
- Wide adoption by the industry at large

However, the implementation will differ from a pure RESTful style in the following significant details:

1. Payloads are not JSON or XML (as commonly used in most RESTful services) but binary messages generated from custom ASN.1 grammar.
2. Cacheability of responses may not be used due to the nature of the problem domain
3. Uniform Interface architectural constraint is not fully observed due to adherence to ASN.1-defined protocol and to keep payload sizes to the minimum possible

● [MA and RA Services Views](#)

● Test Vectors

The following are test vectors for cryptographic functions and python scripts outlining the mathematical steps involved in those functions that are needed for encryption and signing/verification:

[Crypto Test Vectors](#)

[1602.2 & SCMS ASN.1 Objects](#)

[ECIES Encryption](#)

● Policies and Procedure

- Root Management Policy:

QA SCMS PoC Root Management Policy

The policy for management of the QA SCMS PoC Root CA can be found at the SCMS Wiki at: <https://wiki.campllc.org/display/SCP/Root+Management+and+Revocation+Recovery>. As the QA SCMS PoC system will be updated over time, the QA SCMS Root Management policy will remain in the SCMS Wiki so it can adapt to future changes if necessary.

Production SCMS PoC Root Management Policy

The policy for management of the Production SCMS PoC Root CA, and the policies for connecting an ICA to the Root CA can be found in:

- INTEGRITY Security Services, Inc. V2X Root Certificate Authority (CA) Subscriber Agreement for Intermediate CA Certificate
- INTEGRITY Security Services, Inc. Certificate Policy

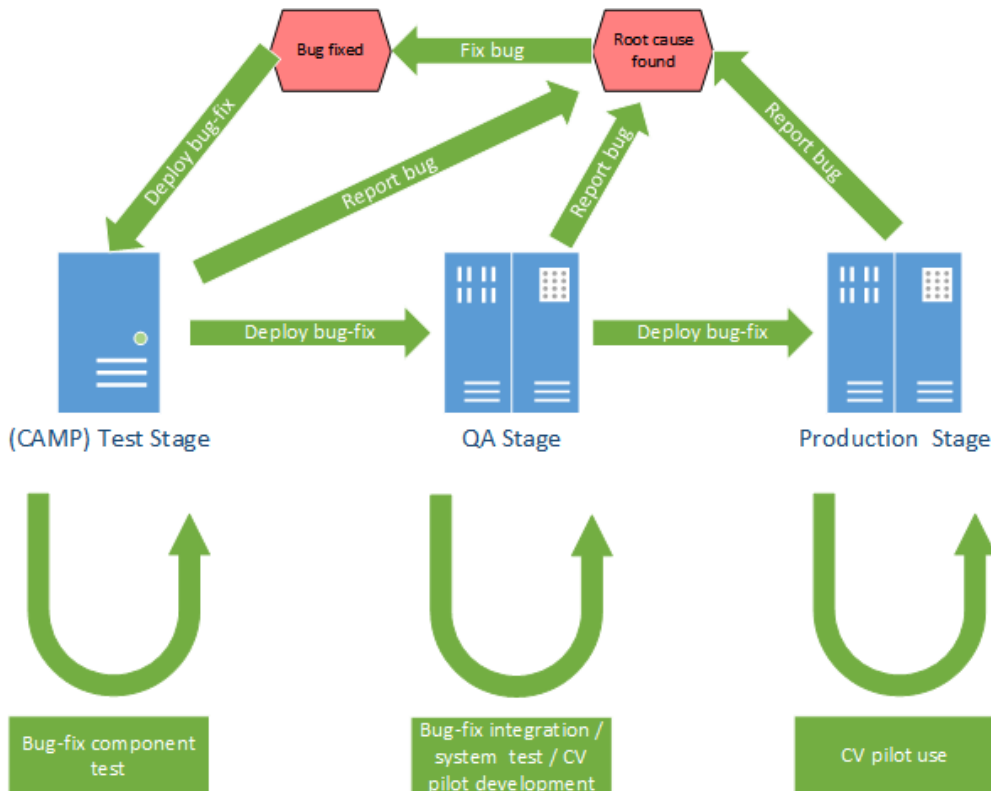
The INTEGRITY Security Services, Inc. Certificate Policy can be found at:

<https://www.ghsiss.com/v2xca>

- [SCMS Policies](#)

Environment Documentation

SCMS Proof-of-Concept Connected Vehicle Pilot Environments Overview



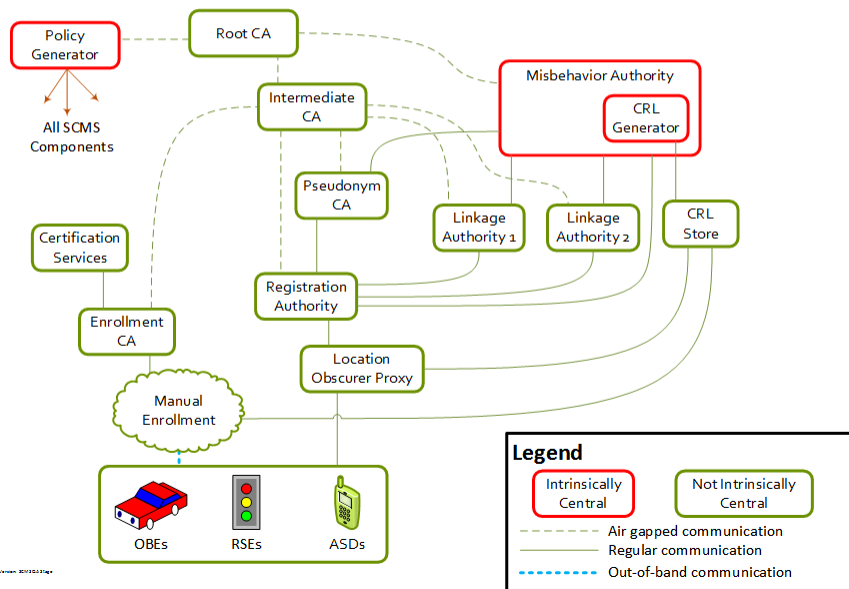
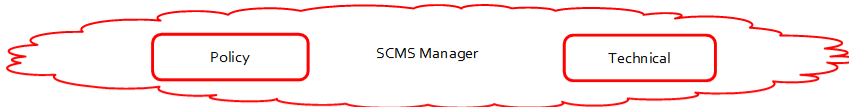
The picture above shows these three environments and how they relate to each other:

1. (CAMP) Test Stage (TEST): This environment is internal to CAMP and is not available to any outside stakeholders and is used for SCMS development and testing purposes.
2. QA Stage (QA): This environment is publicly available via Internet IPv6 and IPv4 connections. It is used to evaluate new SCMS software versions, as well as bug fixes and enhancements. The environment provides device developers with a working system that they can use to develop and test their devices. The level of security, as well as the security requirements for devices using certificates, is lower than the Production stage.
3. Production Stage (PROD): This environment has the highest level of security, uses a production grade offline Root CA (including storing the CA's certificate in an HSM) and is strictly used for production devices only. These production devices are more specifically US DOT approved CV Pilot participants. Approved Devices that handle certificates issued by this system must implement all security requirements as outlined in Use Case 2: OBE Bootstrapping (Manual), Secure Environment for Device Enrollment and Hardware, Software and OS Security Requirements.

If any bugs are detected (in any of the stages) the SCMS software team will analyze the error, respectively create a new version of the SCMS POC software and then apply the following deployment cycle:

1. The new version is deployed to TEST and tested internally at CAMP.
2. After successful testing and assured stability, the software will be deployed to QA.
3. Following a few of weeks of monitoring the new software in the QA stage, and considering any feedback from the development community, the new version will eventually be deployed to PROD.

SCMS Proof-of-Concept Connected Vehicle Pilot QA Environment

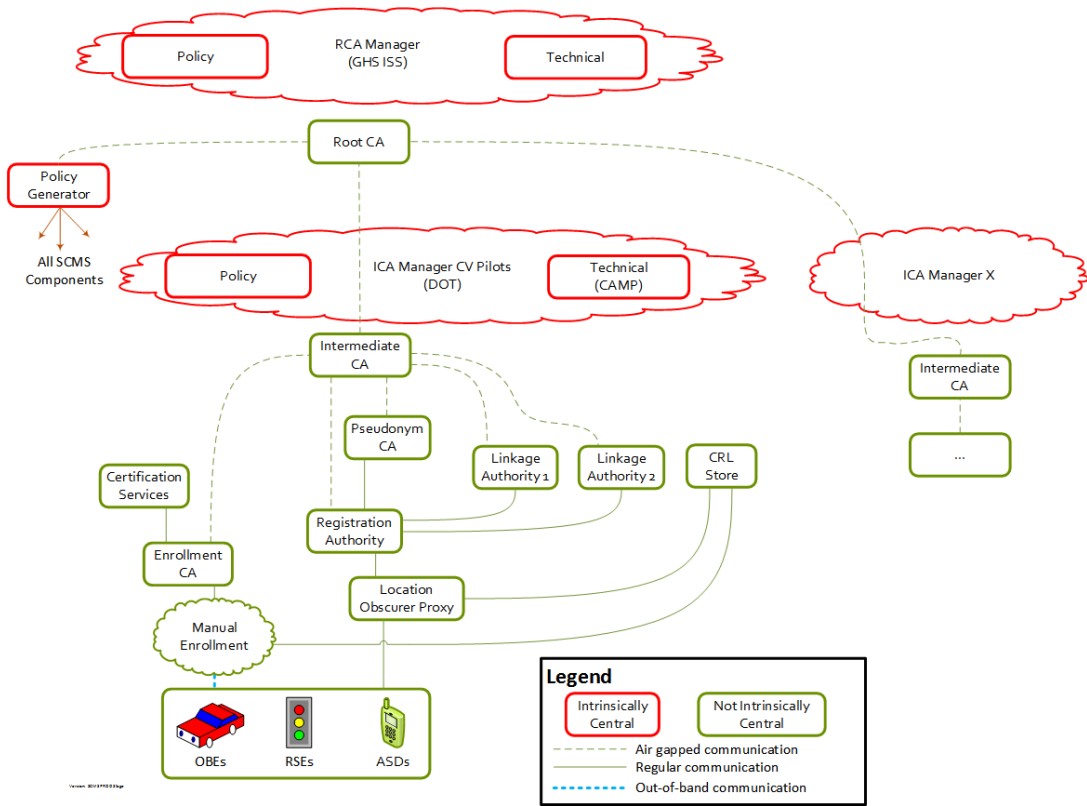


The QA environment has the capability to revoke certificates, however only manual revocation is supported. Bootstrapping is implemented with a manual enrollment as documented in Use Case 2: OBE Bootstrapping (Manual).

Features to be added at a later:

- Global Misbehavior Detection will be implemented to provide an (semi-)automatic way of revoking certificates based on misbehavior reports
- Automatic enrollment for selected device suppliers / operators
- Re-enrollment as documented in Use Case 20: EE Re-Enrollment
- Electors as documented in Elector-based Root Management

SCMS Proof-of-Concept Connected Vehicle Pilot PROD Environment



Initially the PROD environment will not have a MA, and therefore will not have the capability to receive or handle misbehavior reports. To achieve the expected security levels, the PROD stage uses a commercially available Root CA. The overall SCMS system has multiple levels of management as seen in the SCMS PKI hierarchy:

- As a governance body there is a Root CA Manager that sits above the system and is seen as the policy and technical arm. It is responsible to run and protect the Root CA and issue a PG, a CRLG and ICA certificates. Stakeholders that get an ICA must follow the Root CA policies, e.g. the Certificate Policy.
- In the SCMS PKI hierarchy below the Root CA Manager there can be multiple ICA Managers. The USDOT is considered an ICA Manager and will manage an ICA with the help of its policy and technical arm. The SCMS design can support many ICA Managers.

Given a single shared Root CA it's important to note that for certain SCMS features to work all of the ICA Managers have to cooperate with the Root CA Manager.

SCMS Supported V2X Applications

This is the list of supported V2X Applications for PoC and Pilot Deployment.

	Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
1	Basic Safety Message (BSM)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		Support multiple V2V safety applications	SSP: absent (default permissions)	NYC THEA Wyoming	OBE	Pseudonym Certificate
2	Vehicle Turning Right in Front of Bus Warning	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		Assumes specific application in bus to analyze the received BSMs and determine if a warning should be provided to the bus driver	SSP: absent (default permissions)		OBE	Pseudonym Certificate
3	Intelligent Traffic Signal System (I-SIG) In-Vehicle Information Potential	Current assumption is BSM inputs only (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)	Difficult to know the SSP requirements until the application design is more complete	Difficult to know if there are other application messaging requirements until the application design is more complete	SSP: absent (default permissions)			Pseudonym Certificate
4	Forward Collision Warning (FCW)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)			SSP: absent (default permissions)		OBE	Pseudonym Certificate
5	Emergency Electronic Brake Light (EEBL)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)			SSP: absent (default permissions)		OBE	Pseudonym Certificate
6	Blind Spot Warning (BSW)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1			SSP: absent (default permissions)		OBE	Pseudonym Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
			(application specification)						
7	Lane Change Warning/Assist (LCA)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		SSP: absent (default permissions)		OBE	Pseudonym Certificate
8	Intersection Movement Assist	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		SSP: absent (default permissions)		OBE	Pseudonym Certificate
9	Stationary Vehicle Ahead (SVA)	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		SSP: absent (default permissions)		OBE	Pseudonym Certificate
10	Do Not Pass Warning	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735 (data structure) / SAE J945/1 (application specification)		SSP: absent (default permissions)		OBE	Pseudonym Certificate
11	Probe Enabled Traffic Monitoring	BSM inputs (BSM PSID)	0p20 (0x20) (32)	SAE J2735	Detailed application description not available.	Either RSE just collects BSMs or RSE sends WSA with probe request and then vehicle uses IP service to send requested information or establish two-way communications. In the case of probe request it isn't clear whether the probe request PSID		OBE	Pseudonym Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
					needs to appear in a certificate.				
12	WAVE security management	Support	0p23 (0x23) (35)	IEEE 1609.2		SSP: absent (default permissions)			
13	Misbehavior Reporting for Common Applications	Support	0p26 (0x26) (38)	Crash Avoidance Metrics Partners LLC	Detailed application description not available.	NOTE: This PSID is also used for event data recording in NYC, because it already appears in the BSM certificate and because event data reporting is very similar to misbehavior reporting. But this is a bit of a hack.	SSP: absent (default permissions)	OBE	Pseudonym Certificate
14	Vulnerable Road Users Safety Application	Vulnerable Road Users Safety PSID	0p27 (0x27) (39)	SAE J2735 (data structure) / SAE J945/9 (application specification – draft)	Detailed application description not available.				Application Certificate
15	Differential GPS Corrections, Uncompressed	Support	0p80-00 (0x80) (128)	SAE J2735 (data structure)	Detailed application description not available			TMC App	Application Certificate
16	Differential GPS Corrections, Compressed	Support	0p80-01 (0x81) (129)	SAE J2735 (data structure)	Detailed application description not available				Application Certificate
17	Red Light Violation Warning / RSE	3 - Signal Violation Warning (Intersection Safety and Awareness PSID. SPaT &	0p80-02 (0x82) (130)	SAE J2735 (data structure) SAE J2945/2 (SSP framework) SAE J2945/10 (application	Detailed application description not available. SAE subgroup has been tasked with developing SSP specification for SPAT; draft available.			RSE	Application Certificate

	Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
		MAP use message ID to distinguish message type)		specification, in progress) RSU requirements document (partial application specification)						
18	Pedestrian in Signalized Crosswalk Warning / RSE	16 - Pedestrian Warnings (Intersection Safety and Awareness PSID. SPaT & MAP use message ID to distinguish message type)	0p80-02 (0x82) (130)	SAE J2735 (data structure) SAE J2945/2 (SSP framework) SAE J2945/10 (application specification, in progress) RSU requirements document (partial application specification)	SAE subgroup has been tasked with developing SSP specification for SPAT; draft available May require SSP field to indicate that RSU is equipped with ability to detect pedestrians.				RSE	Application Certificate
19	Mobile Accessible Pedestrian Signal System (PED-SIG)	SRM	SRM – which PSID?	SAE J2735 (Data structure) SAE J2945/11 (Application specification, in progress)	Detailed application description being developed within Pilot Deployment projects. No SSP definition currently known.					Application Certificate
20	Transit Signal Priority/ Special Vehicles	1 - Signal Pre-emption/Priority (SignalRequest Message)	SRM – which PSID?	SAE J2735 (Data structure) SAE J2945/11 (Application specification, in progress)	Detailed application description being developed within Pilot Deployment projects. No SSP definition currently known.					Identification Certificate
21	Modified Eco-Speed	2 - Speed Harmonization	0p80-03 (0x83) (131)	SAE J2735 (Data Structure)	SAE subgroup has been tasked with developing	Not clear that this message will ever be signed by RSE;			TMC App	Application Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
Harmonization / RSE	(Traveler Information and Roadside Signage PSID)		SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SSP specification for TIM . Draft available.	NYC and Wyoming approaches assume all TIMs are signed by TMC			or RSU	
22 Modified Eco-Speed Harmonization / TMC	2 - Speed Harmonization (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available. SSP may differentiate between different speed harmonization categories (e.g., eco-, light vehicles, freight, transit)	NYC and Wyoming approaches assume all TIMs are signed by TMC			TMC App or RSU	Application Certificate
23 Curve Speed Warning	8 - Curve Speed Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.				TMC App or RSU	Application Certificate
24 Reduced Speed / Work Zone Warning / RSE	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC, even for situations like this where they contain local information.			TMC App or RSU	Application Certificate

	Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
25	Reduced Speed / Work Zone Warning / TMC	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC, even for situations like this where they contain local information.			TMC App or RSU	Application Certificate
26	Spot Specific Weather Warnings / RSE	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC, even for situations like this where they contain local information.			TMC App or RSU	Application Certificate
27	Spot Specific Weather Warnings / TMC	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC, even for situations like this where they contain local information.			TMC App or RSU	Application Certificate
28	Variable Speed Limits / RSE	10 - Speed Zone (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate

	Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
29	Variable Speed Limits / TMC	10 - Speed Zone (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
30	Speed Harmonization / RSE	2 - Speed Harmonization	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
31	Speed Harmonization / TMC	2 - Speed Harmonization (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
32	Work Zone Alerts / RSE	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
33 Work Zone Alerts / TMC	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
34 Truck Restrictions / RSE	11 - Special Vehicle Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
35 Truck Restrictions / TMC	11 - Special Vehicle Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
36 Automatic Alerts for First Responders	11 - Special Vehicle Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available. Specific SSPs may need to be designated for messages to special vehicles	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
37 CV-enabled Weather-Responsive Variable Speed Limits	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
38 Road Weather Advisories for Trucks and Vehicles	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available. Different SSPs may be needed to differentiate messages for different categories of vehicles	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
39 Emergency Communications and Evacuation (EVAC)	9 - Temporary Situation Warning (Traveler Information and Roadside Signage PSID)	0p80-03 (0x83) (131)	SAE J2735 (Data Structure) SAE J2945/4 (Application specification, in progress) CVPD site-specific documents	SAE subgroup has been tasked with developing SSP specification for TIM . Draft available.	NYC is assuming that TIMs are signed by the TMC.			TMC App or RSU	Application Certificate
40 Probe Data Collection	BSM and alert event data collection for researchers Regional extension of PDM / PVD Alternatively: IPv6	0p80-04 (0x84) (132)	SAE J2735 (data structures) SAE J2945/12 (application specification) CVPD site-specific documents		Not clear that this appears in certificates – may appear only in WSAs?	SSP: absent (default permissions)			

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
41	WAVE Service Advertisement	Support	0p80-07 (0x87) (135)	IEEE 1609.3		SSP: absent (default permissions)			
42	Peer-to-peer distribution of Security Management Information	Peer-to-peer Certificate Distribution Psid	0p80-08 (0x88) (136)	Crash Avoidance Metrics Partners LLC		Doesn't appear in certificates			
43	Certificate Revocation List Application	Support	0p80-80 (0x100) (256)	IEEE 1609.2 (Data structures / SSP) / CAMP (application specification)		Only appears in appPermissions of the CRLG	SSP: specific to CRL as specified in 1609.2 CRL ASN.1 module		
44	Vehicle initiated distress notification		0pC0-00-02 (0x40-82) (16,514)	Wyoming DOT	<p>UPER-encoded J2945/2 DSRC-SSP containing a SSPentry. Version is set to 1, allowedSSPs is a SEQUENCE containing exactly one SSPentry field, as follows:</p> <pre> SSPentry ::= SEQUENCE { index INTEGER(msg-travelerInformation), - - 31 constraint (SSPconstraintAll) -- Boolean: True } </pre>		SSP (DSRC-SSP UPER encoded in hex): 00 80 01 F0 40 Entire 1609.2 PsidSsp structure as an OPAQUE OCTET STRING in COER: 80 02 40 82 80 05 00 80 01 F0 40	Wyoming	Identification Certificate
45	Transcore software update	Support	0pE0-00-00-03 (0x20-40-83) (2,113,667)	Transcore, Inc					Application Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
46 Over-the-air File Broadcast	Support	0pE0-00-00-08 (0x20-40-88) (2,113,672)	Siemens Industry, Inc.						Application Certificate
47 Data Log Transfer	Support	0pE0-00-00-09 (0x20-40-89) (2,113,673)	Siemens Industry, Inc.						Application Certificate
48 CV Pilot Application 3		0pE0-00-00-0A (0x20-40-8A) (2,113,674)	Crash Avoidance Metrics Partners LLC						
49 CV Pilot Application 4		0pE0-00-00-0B (0x20-40-8B) (2,113,675)	Crash Avoidance Metrics Partners LLC						
50 CV Pilot Application 5		0pE0-00-00-0C (0x20-40-8C) (2,113,676)	Crash Avoidance Metrics Partners LLC						
51 CV Pilot Application 6		0pE0-00-00-0D (0x20-40-8D) (2,113,677)	Crash Avoidance Metrics Partners LLC						
52 CV Pilot Application 7		0pE0-00-00-0E (0x20-40-8E) (2,113,678)	Crash Avoidance Metrics Partners LLC						
53 CV Pilot Application 8		0pE0-00-00-0F	Crash Avoidance						

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
		(0x20-40-8F) (2,113,679)	Metrics Partners LLC						
54 CV Pilot Application 9		0pE0-00-00-10 (0x20-40-90) (2,113,680)	Crash Avoidance Metrics Partners LLC						
55 CV Pilot Application 10		0pE0-00-00-11 (0x20-40-91) (2,113,681)	Crash Avoidance Metrics Partners LLC						
56 CV Pilot Application 11		0pE0-00-00-12 (0x20-40-92) (2,113,682)	Crash Avoidance Metrics Partners LLC						
57 CV Pilot Application 12		0pE0-00-00-13 (0x20-40-93) (2,113,683)	Crash Avoidance Metrics Partners LLC						
58 CV Pilot Application 13		0pE0-00-00-14 (0x20-40-94) (2,113,684)	Crash Avoidance Metrics Partners LLC						
59 CV Pilot traffic signal priority status	SSM	0pE0-00-00-15 (0x20-40-95) (2,113,685)	US Department of Transportation					RSU	Application Certificate
60 CV Pilot traffic signal request	SRM	0pE0-00-00-16 (0x20-40-	US Department of Transportation					OBE	Identification Certificate

Application	Application Category	PSID (Hex value) (Decimal value)	Organization/ Documentation	SSP Notes	Comments	SSP Value to Appear in End Entity Certificates for CV Pilots	CVPD Sites That Use It (to be completed)	EE Type	Certificate Type
		96) (2,113,686)							
61 CV Pilot MAP distribution	MAP	0pE0-00-00-17 (0x20-40-97) (2,113,687)	US Department of Transportation						Application Certificate

SCMS Certificate Types

The V2X system uses several types of certificates. SCMS components generate these and in many cases can also revoke them. All certificate lifetimes and renewal periods are listed separately for [PoC](#) and CV Pilot [Test, QA](#), and [Prod](#) stages. All the EE certificates are of **implicit** type to save storage space and over-the-air bytes. All the SCMS component certificates are of **explicit** type.

On-Board Equipment (OBE)

Road-side Equipment (RSE)

SCMS Components

EE Certificate Type Features

Requirements

PoC Certificates Expiration Timeline

CV Pilot PROD Certificates Expiration Timeline

CV Pilot QA + Test Certificates Expiration Timeline

CV Pilot PROD Certificates Expiration Timeline

Device Hardware, Software and OS Security Requirements

Overview and Goals

This section describes hardware, software, and operating system security for systems that run DSRC applications and use cryptographic private keys and certificates in the format specified by IEEE Standard 1609.2-2016 and that are issued by the Security Credentials Management System (SCMS).

The security requirements apply to two logically distinct sets of functional blocks:

- **Privileged applications:** These applications run autonomously (i.e., do not require human intervention to start running) and either send or receive signed messages. They run on the **host processor**.
- **Cryptographic operations:** These operations use secret keys from symmetric cryptographic algorithms, or private keys from asymmetric cryptographic algorithms. They run on the **Hardware Security Module (HSM)**.

The goals of these requirements are:

1. Different privileged applications can have different sets of keys such that:
 - a. A privileged application is able to sign with its own keys
 - b. A privileged application is not able to sign with keys reserved for use by a different privileged application
 - c. Non-privileged applications do not have any access to keys that are reserved for use by privileged applications
2. No application has read access to key material – all key material is execute- or write-only
3. Keys used for verification are protected against unauthorized replacement
4. The system supports software/firmware update in such a way that the above properties continue to hold

This document does not address processes for certifying that systems meet the requirements. Its purpose is simply to state the requirements.

Architecture

The requirements below cover three architectures.

- **Integrated architecture:** The host processor and the HSM are the same processor
- **Connected architecture:** The host processor and the HSM are different, but they are physically connected using a connector that connects only those two processors. The only way to read or write data flowing between the two processors is by physically tapping into that connector.
- **Networked architecture:** The host processor and the HSM are different and connected over a network or bus that has other processors connected to it

Architecture-specific Requirements

- **Integrated architecture:** An integrated processor has no additional requirements beyond the ones identified above.
- **Connected architecture:** A Connected processor has no additional requirements beyond the ones identified above.
- **Networked architecture:** In Addition to the requirements identified above, the host processor must authenticate itself to the HSM with an authentication mechanism based in hardware with the same physical security level as the HSM itself.

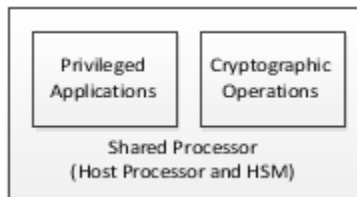


Figure 1 Integrated Architecture

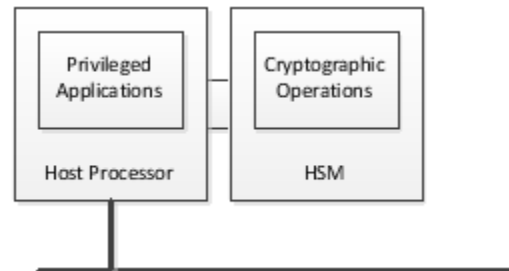


Figure 2 Connected Architecture

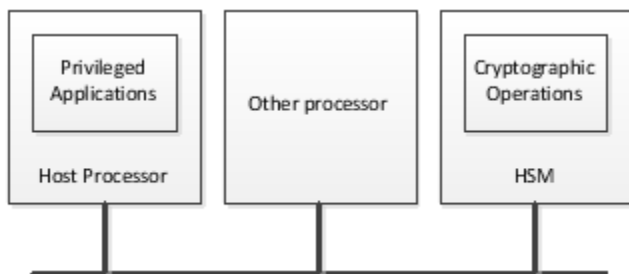


Figure 3 Networked Architecture

Host Processor

HSM

Secure Environment for Device Enrollment and Storage Consideration

Elector Based Root Management

After a root CA certificate's validity period ends or a revocation was necessary and a new root CA certificate has been established for replacement, how can an EE start trusting this new root CA certificate? The trust in an initial root CA certificate is implicit, as it is installed in a [secure environment](#) with out-of-band communication during bootstrapping of the device. One option would be to get the device back to that secure environment and use out-of-band communication to install the new root CA certificate. However, this is suboptimal due to the required effort and will render the overall V2X system partly out-of-order until all devices have installed the new certificate.

To manage the root CA certificate over time and gain resilience against compromises on any level, the SCMS needs the ability to heal itself, which means to bring itself into a state where it can endure another single compromise or end of the validity period of a Root CA. This recovery should occur while keeping the devices operational whenever possible, that is, capable of sending, receiving and validating BSMs, and be able to restore the system hierarchy without requiring physical access to devices. Elector-based Root Management is the solution that provides those means by installing a distributed management schema on top of the SCMS Root CAs.

Distributed Management & Electors

A distributed management scheme, like a democracy, contains within itself the power to replace an established hierarchy and does not succumb to a single failure. The concept of **Electors**, which together have the power to change and manage the trust relationships of the system, adds such a scheme to the SCMS design. Within a system like the SCMS, the number of electors should be $2n+1$, where n is the number of simultaneous elector expiration/compromises that the SCMS can tolerate.

Like in a democracy the Elector-based Root Management introduces a **Ballot** with **Endorsements**. The electors cast **Votes** by signing an endorsement of a given root CA or elector certificate. A ballot aggregates all these endorsements. When a quorum of valid elector endorsements is on the ballot, any component in the system can trust the ballot.

The electors are not part of the PKI hierarchy, and therefore they can use a different crypto-system than the SCMS PKI. In fact, each of them can use a different one. This raises the probability that in case of a root CA or elector certificate compromise due to a broken cryptography, the system is still able to heal itself.

The resulting system may have multiple, self-signed root CA certificates, each of which operates at the top of their trust chain. Each root CA's certificate is endorsed by a ballot with at least a quorum of votes from non-revoked electors. Devices need to verify the trust chain up to a root CA certificate, at which point they must verify that a quorum of non-revoked electors has endorsed that root CA certificate.

Ballots & Endorsements

Electors operate by signing endorsements. A ballot can include the following basic types of endorsements:

- Add root CA certificate
- Add elector certificate
- Revoke root CA certificate
- Revoke elector certificate

Each ballot contains only one type of endorsement. SCMS components, including devices, receive ballots adding a certificate via a certificate chain file distributed by the PG. They receive ballots removing a certificate via the CRL distributed by the CRL store.

All components know the quorum and the certificates of the initial set of electors and therefore can validate the endorsements contained in the ballot. Once the ballot is validated, the component can follow the endorsed action to add or remove the ballot's certificate from its trust store.

The SCMS Manager will coordinate the production of the ballot messages.

Structure of Ballots

Revocation/Endorsement Impact on Devices

Structure of Ballots

The ballot which aggregates all independent elector endorsements is an ASN.1 structure. This structure contains the following elements:

1. The certificate of the root CA or elector to be endorsed
2. A sequence of endorsements, each containing:
 - a. The type of endorsement
 - b. The hash id of the certificate to be endorsed
 - c. The generation time of the endorsement
 - d. A signature of the elector.

Note that the validity period of a ballot is implicitly given by the validity period of the endorsed certificate.

Revocation/Endorsement Impact on Devices

A key consideration in the design of the root management system is to maintain secure operation of devices without requiring recall or manual re-enrollment of individual devices. The following table outlines the status of devices through the addition or revocation of Electors and Root CAs.

Operation	Elector Model Implementation
Revoking an Elector	As long as there are at least three electors with a quorum of two, then one elector may be removed without impacting operation: The remaining electors are still a quorum and their endorsements of the root CA certificate would still be valid. A single revoked elector would not stop operations of any device. A replacement elector may then be added back to the system to return to a state with three valid electors. A larger number of electors may be used to improve the system's resilience to compromise or failure of these top-level trust anchors.
Revoking a Root CA	Revoking a root CA certificate would stop operations of devices that possess certificates chaining up to the revoked root CA certificate. Those devices would need to re-enroll and be re-provisioned with a different root CA before they could be trusted by other devices.
Adding an Elector	A new self-signed elector certificate that is endorsed by a quorum of valid electors can be trusted by devices and other SCMS components without the need of returning them to a secure environment. In addition, this new elector can endorse existing root CA certificates without the need for any updates of the existing valid certificates, including the device's pseudonym certificates.
Adding a Root CA	A new, self-signed root CA certificate that is endorsed by a quorum of valid electors can be trusted by devices and other SCMS components without the need of returning them to a secure environment. Devices can immediately begin to trust messages that chain up to the new root CA.

Table 1 EE Status through Addition/Revocation of Electors and Root CAs

Effect of Voting Schemes on the GCCF

The Global Certificate Chain File (GCCF) contains all the trust chains needed by the SCMS (including EEs), including the Root CA certificates. With the elector model, the Root CA certificates are also accompanied with elector endorsements. The Root CA certificates in the GCCF will be supplied in the form of the "Add Root CA" ballots. The trust chain for certificates under a Root CA will be recorded in the GCCF as a list of IEEE 1609.2 certificates.

Structure of the Trust Hierarchy

The diagram below shows how the SCMS-specific implementation of the elector-based scheme (shown in green) can be implemented in parallel with a standard PKI hierarchy, which supports all SCMS components and EEs. Note that all of the structures shown here can be implemented with standard IEEE 1609.2 certificates without modification. A significant advantage of the elector-based scheme is that, as new Electors are added at level 0, an existing root CA can receive new endorsements from an elector without having to change their certificates.

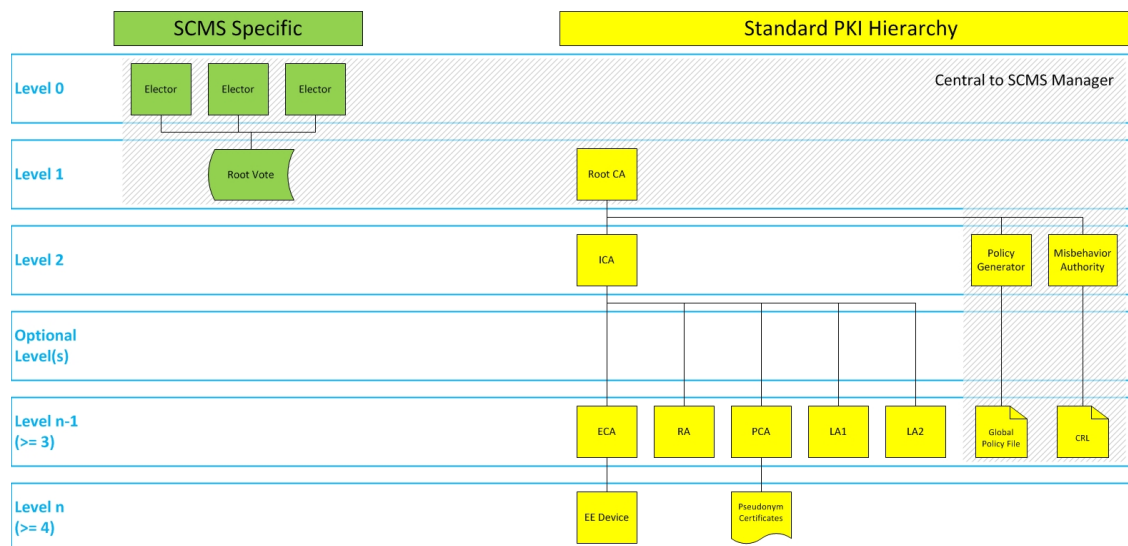


Figure 4 Endorsement Method Details

Impact on EE Storage

The implementation of the elector scheme will affect how EE storage is used.

1. An EE must be able to store securely a number of elector IEEE 1609.2 self-signed certificates. In the PoC, three electors will be operational. Storage for four electors and elector endorsements must be available. In deployment, perhaps nine will be operational, and storage for ten is assumed.
2. An EE must be able to store securely a number of Root CA self-signed certificates. In the PoC, there will be at most two (to allow for testing of Root replacement). In deployment, storage for ten is assumed. If the EE will check the votes on these Root CA self-signed certificates each time, then these need not be stored in the secure trust store.
3. EEs must have secure software used to update the trust store through the correct processing of ballots. This also involves protection for basic parameters under which votes are acted upon, the *quorum*, which is an assumed number less than ten.

Note that all EEs (and other SCMS components) must have a secure method for storing and recovering Root CA certificates. Developers of EE hardware and software may choose from a variety of methods for managing secure storage, but their chosen approach must be approved through an EE certification process. To demonstrate some of the various options that are available, three methods are suggested and described in the following diagram:

- Suggestion 1: Store the Root CA certificate directly in tamper-evident storage. This approach allows the EE to quickly access the Root CA certificate with no further validation (EE must validate it only once before it is placed in secure storage).
- Suggestion 2: The EE may store the endorsement message signed by the electors in secure storage to support peer-to-peer certificate learning of root CA certificates.

- Suggestion 3: The EE may validate the root CA certificate once and then store a hash of the certificate in tamper evident storage. Note that this is effectively the same as Suggestion 2 since the endorsement itself will contain a hash of the root CA certificate, but the EE may choose to use a different hashing algorithm to optimize for speed or to reduce storage.

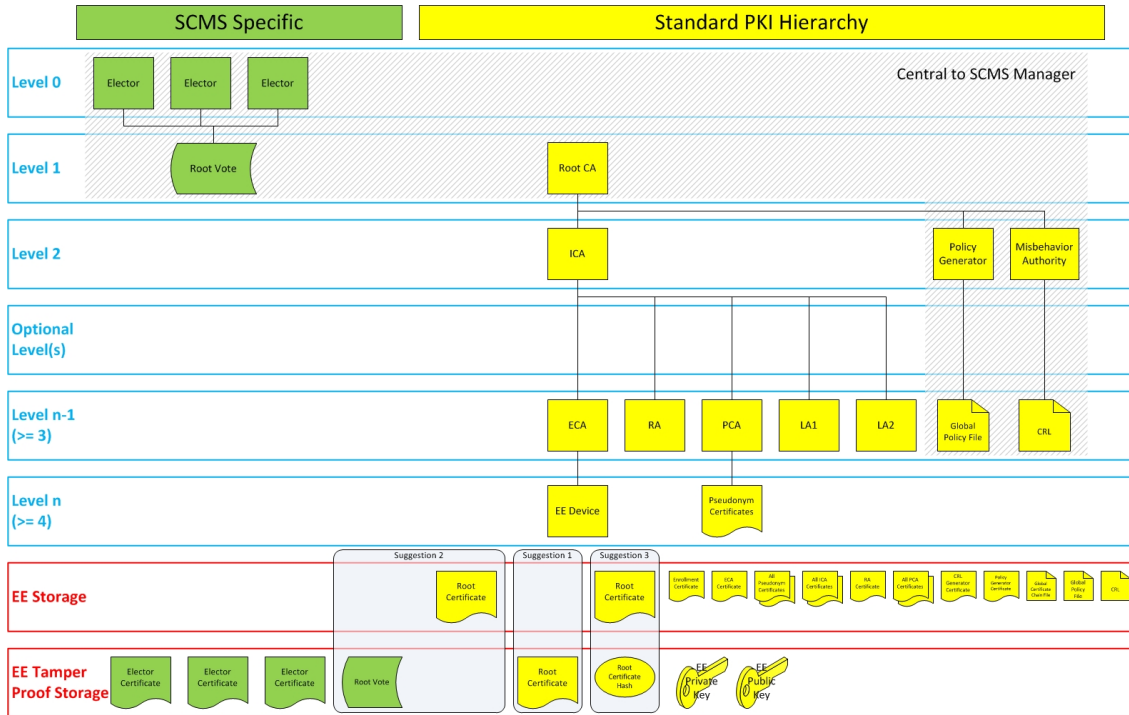


Figure 5 EE Storage Requirements

Root Management and Disaster Recovery in Action

Root Management and Disaster Recovery Actions

This section aims to show root management and disaster recovery in action. As a result, the revocation and replacement of an elector is described below:

- Day 1: Typical SCMS Operations
- Day 2: Revoking an Elector
- Day 3: SCMS Operating with 2 Electors Only
- Day 4: Replacing a Elector
- Day 5: SCMS Returning to Typical Operation

The diagrams given below (and in sections above) are high-level summaries only, and do not contain all requirements for the SCMS components or the EEs.

Day 1: Typical SCMS Operations

SCMS Root CA & Elector Trust Relationships

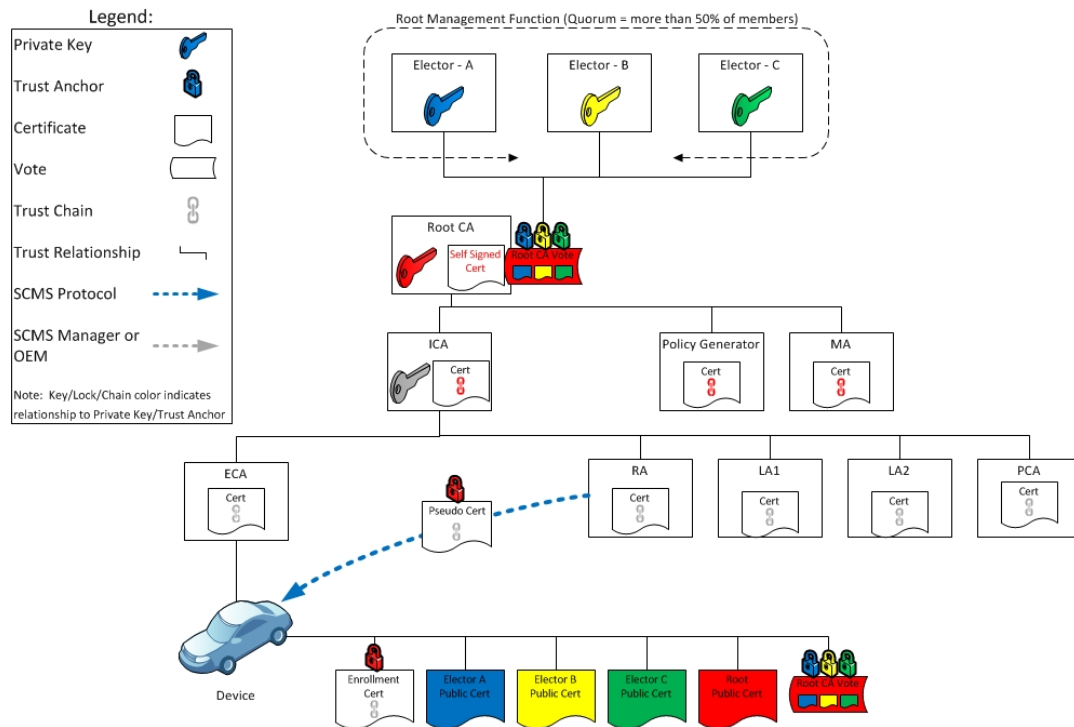


Figure 6 Day 1: Typical SCMS Operations

Day 2: Revoking an Elector

Elector A Revocation Process

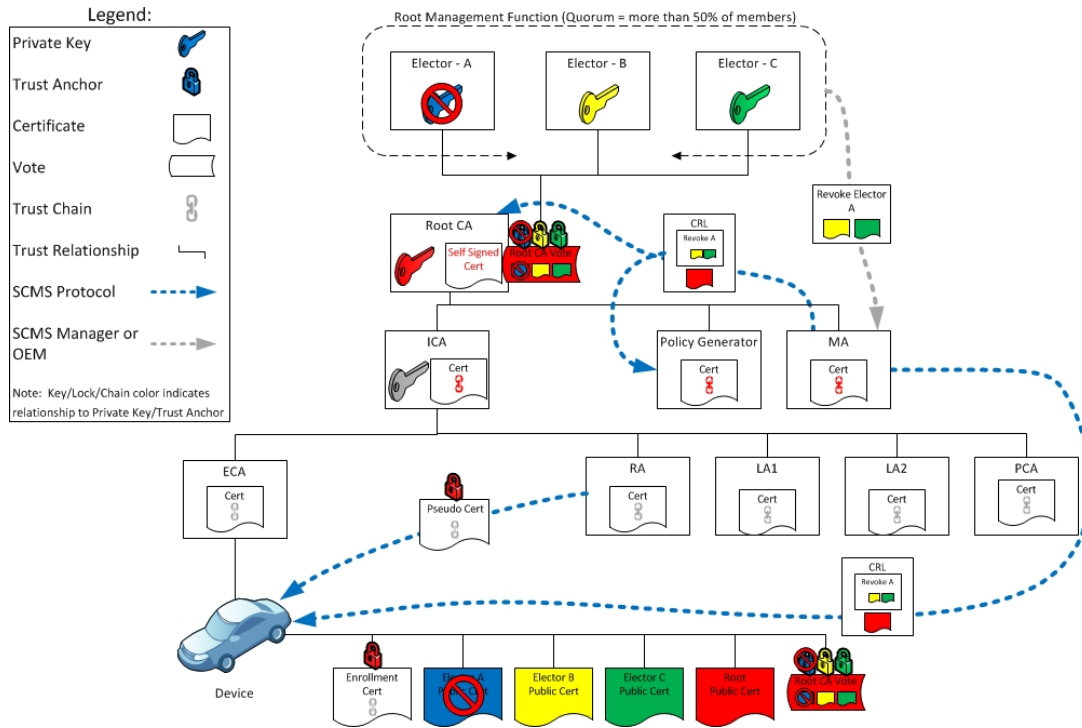


Figure 7 Day 2: Revoking an Elector

At Day 2, an elector has been revoked by votes from m electors (here $m=2$). These votes are included in the CRL. The CRL is distributed to all SCMS components and EEs. The SCMS is still operational.

Day 3: SCMS operating with two electors only

SCMS Operational with Electors B & C Only

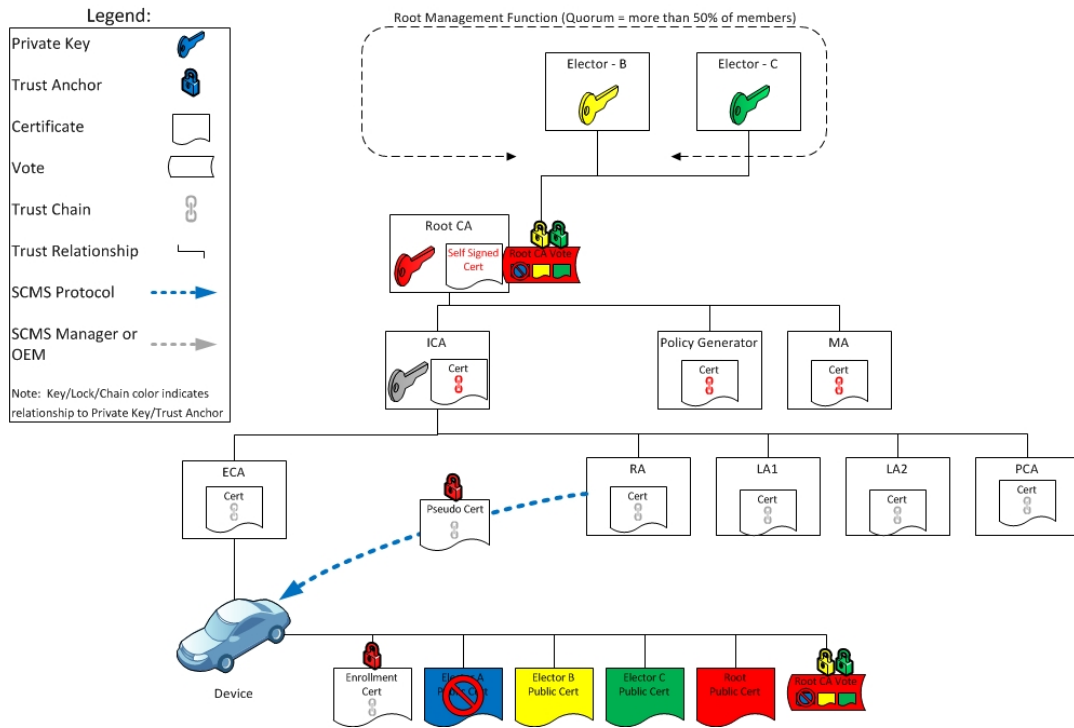


Figure 8 Day 3: SCMS Operating with Two Electors Only

In Day 3, the SCMS is operational with only two, non-revoked electors. Pseudonym certificates continue to validate and EEs to operate.

Day 4: Replacing an Elector

Introduce Elector D

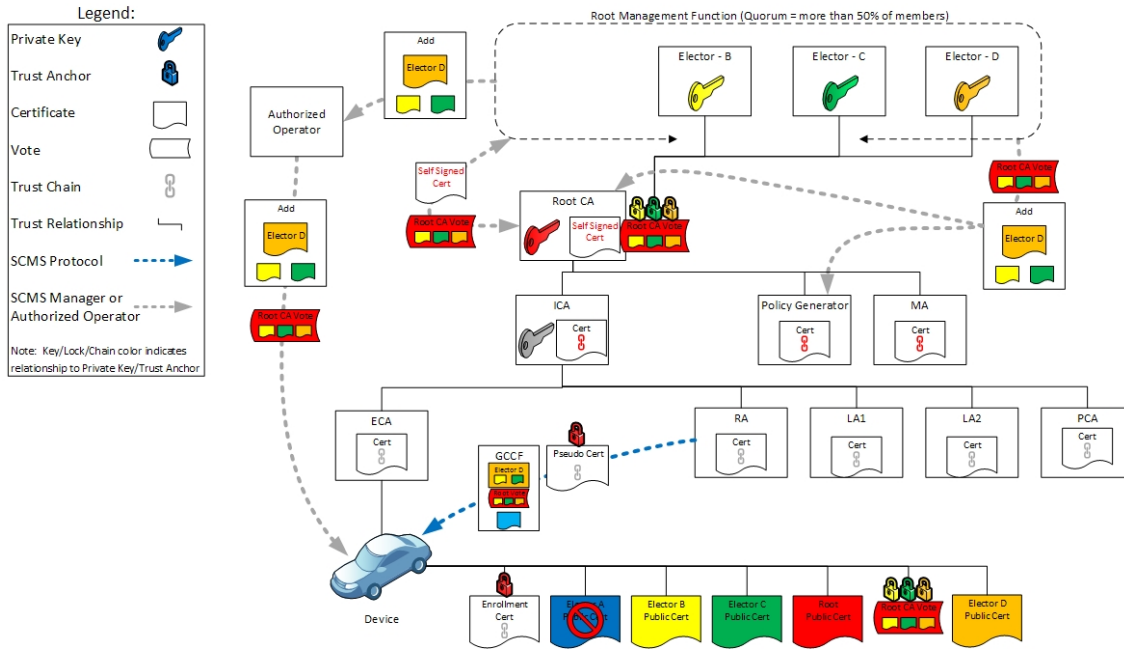


Figure 9 Day 4: Replacing an Elector

In Day 4, the SCMS Manager introduces a new elector through votes endorsing the new elector obtained from the two, remaining, non-revoked electors. Existing devices that do not recognize the new elector continue to operate. The SCMS Manager adds a new elector through a *Ballot* inserted into the Global Certificate Chain File (GCCF), which it then provides through the Policy Generator to RAs. The root management message includes votes from the electors, which the SCMS components and EEs will need to validate before performing the root management operation (adding the elector to the trust store). The SCMS Manager provides a new vote from the new elector for the existing root CA certificate and adds it to the GCCF as well. Even with the addition of the new elector, pseudonym certificates continue to validate and EEs to operate.

Day 5: SCMS Returning to Typical Operation

SCMS Trust Relationships with Elector D

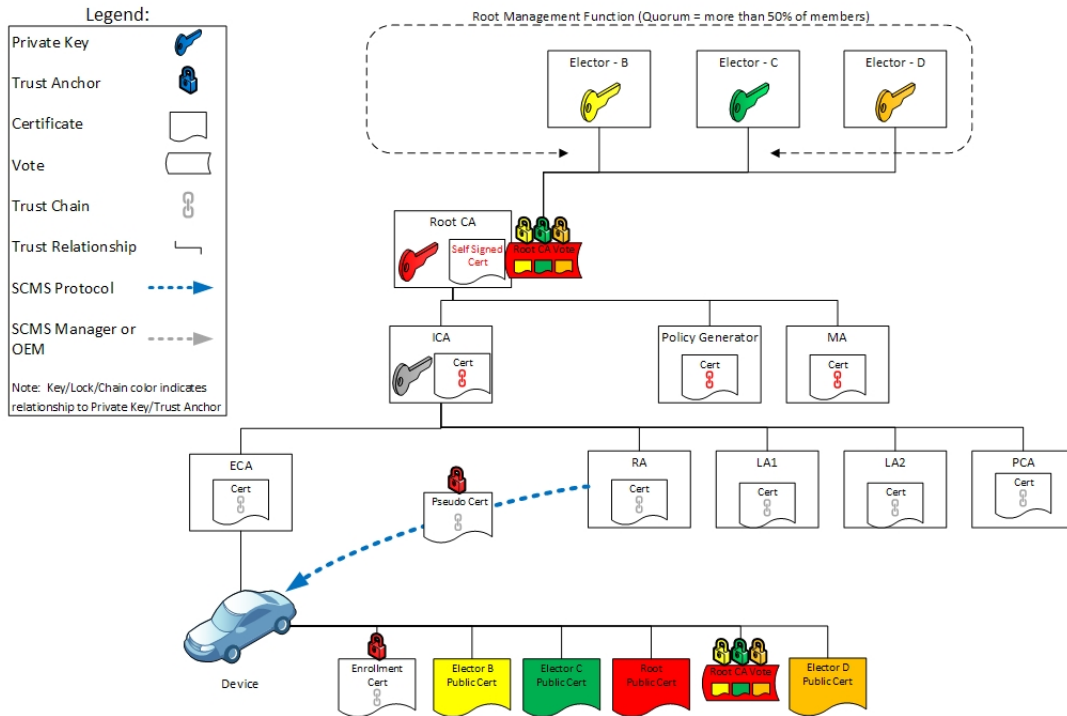


Figure 10 Day 5: SCMS Returning to Typical Operation

In Day 5, the SCMS has been returned to an equivalent of the Initial State of Day 1 with a replacement elector.

The following describes the revocation and replacement of a root CA certificate:

- Day 1: Typical SCMS operations
- Day 2: Standing up a new root CA certificate
- Day 3: Putting the SCMS backend trust relationships in place for the new root CA certificate
- Day 4: Revoking the existing and adding the new root CA certificate
- Day 5: Revoked root CA certificate, system non-functional
- Day 6: System functionality restored

Day 1: Typical SCMS operations

SCMS Root CA & Elector Trust Relationships

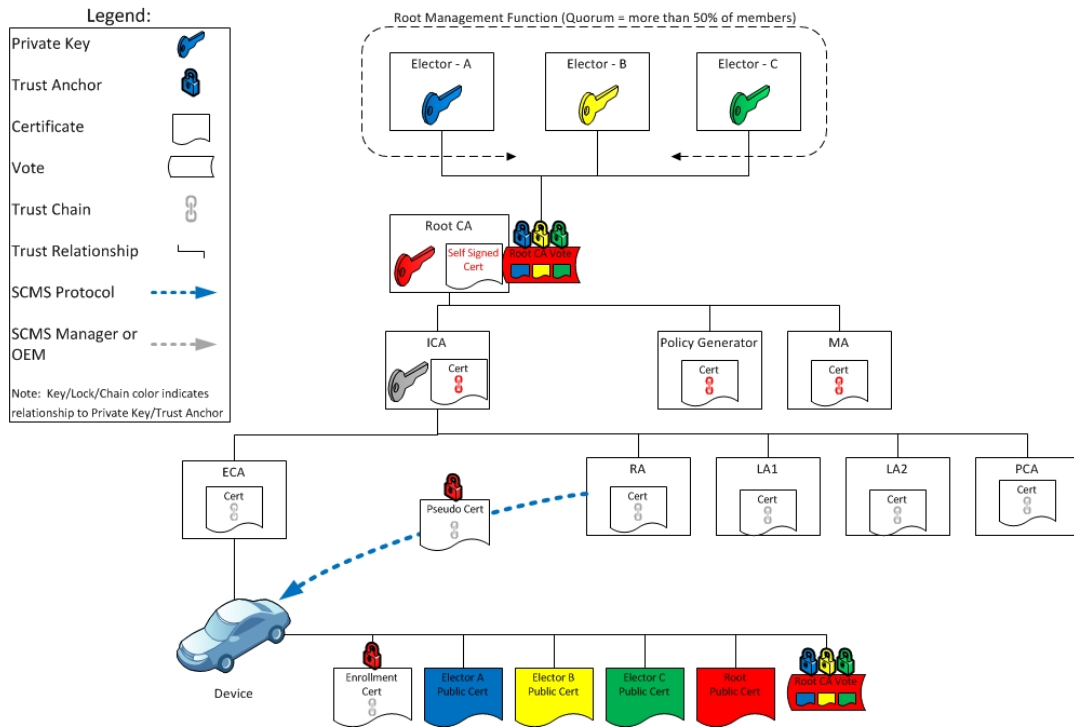


Figure 11 Day 1: Typical SCMS Operations

Day 2: Standing up a new root CA certificate

Create Replacement Root CA & Distribute to SCMS Servers

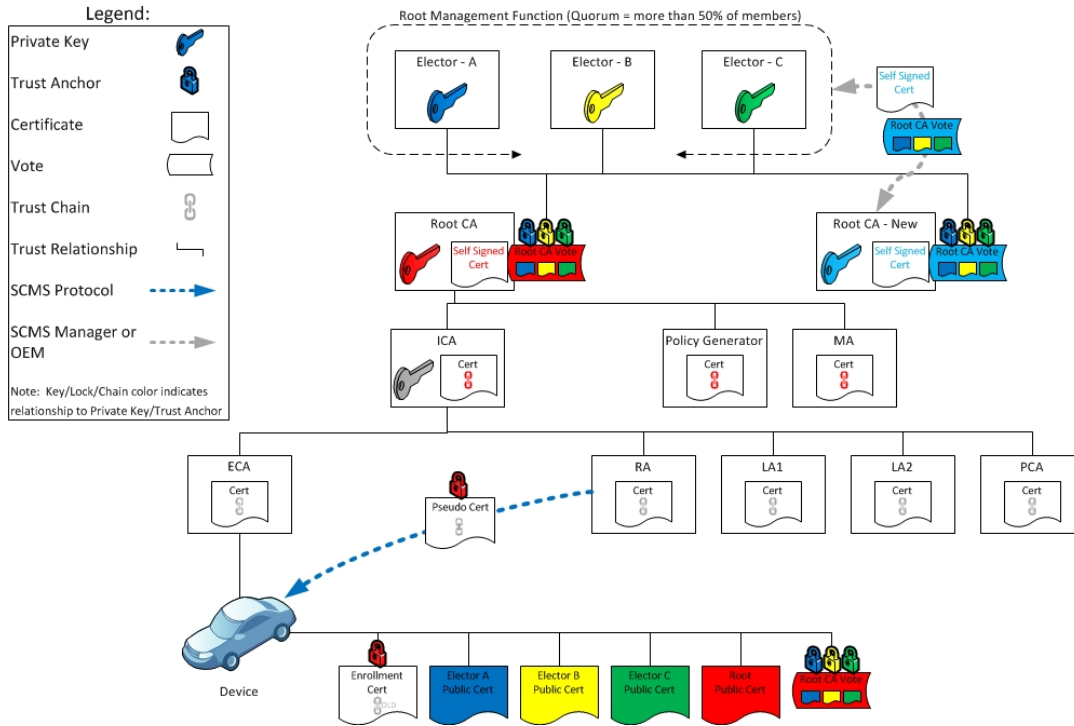


Figure 12 Day 2: Standing Up a New Root CA Certificate

In Day 2, the new root CA certificate is established and endorsed but is not used by the SCMS.

Day 3: Putting the SCMS backend trust relationships in place for the new root CA certificate

Introduce Replacement Root CA Before Revoking Current Root CA

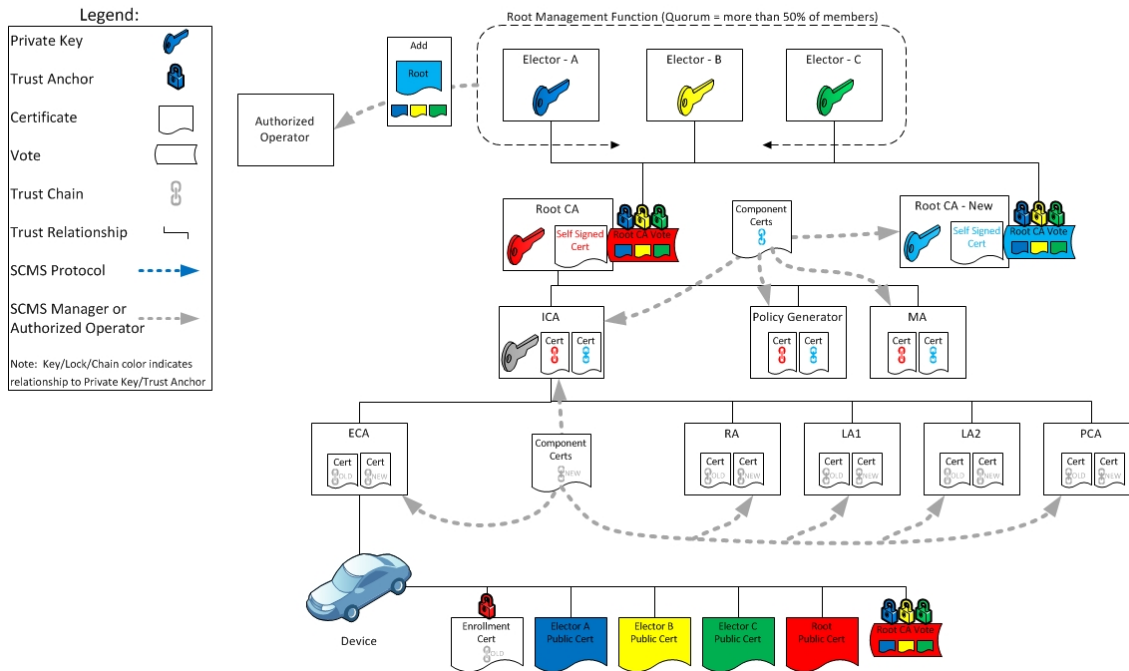


Figure 13 Day 3: Putting the SCMS Backend Trust Relationships in Place for the New Root CA Certificate

On Day 3, all of the background tasks of generating new certificates for SCMS components is performed, but these are not made active. The new root management operation, "Add Root CA," is distributed to all the authorized operators to prepare them for distribution.

Day 4: Revoking the existing and adding the new root CA certificate

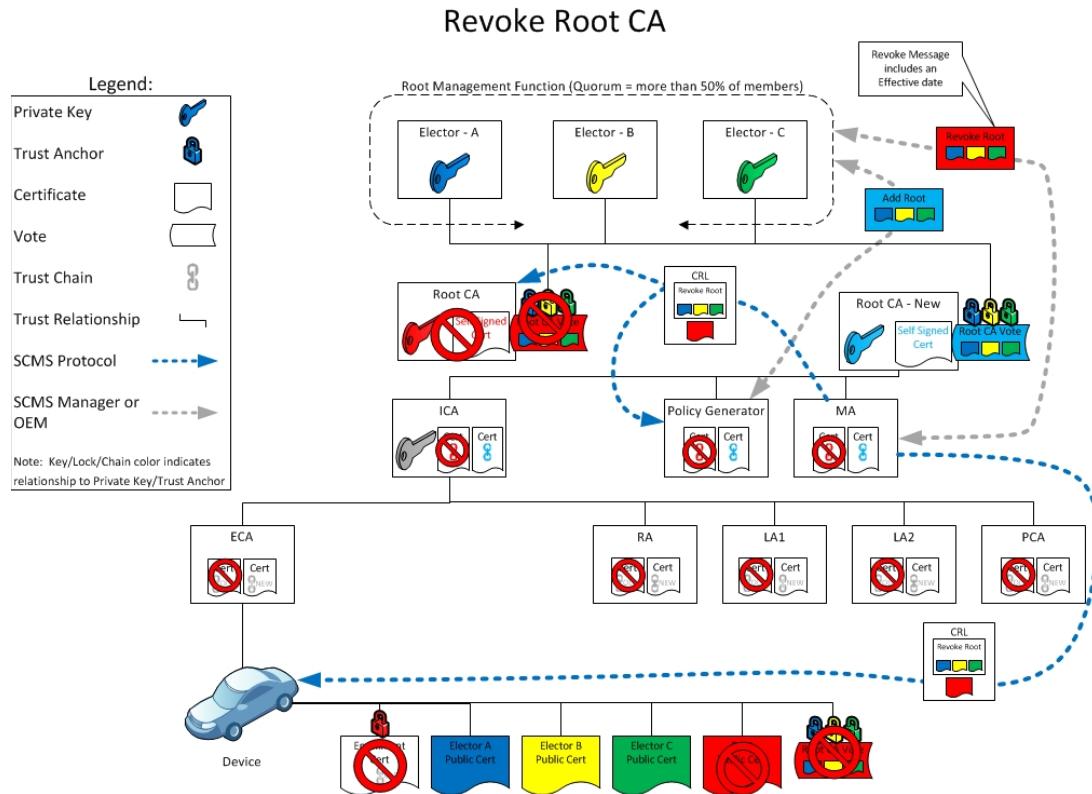


Figure 14 Day 4: Revoking the Existing and Adding the New Root CA Certificate

On Day 4, the old root CA is revoked and the new root CA is added simultaneously to all SCMS components (not EEs). The EEs only receive the revoke message. The GCCF needs to be reset with the new trust structure, which was created on Day 3. All the SCMS components start using the certificates, which chain to the new root CA certificate.

Day 5: Revoked root CA certificate, system non-functional

Root Revoked – System Non-functional

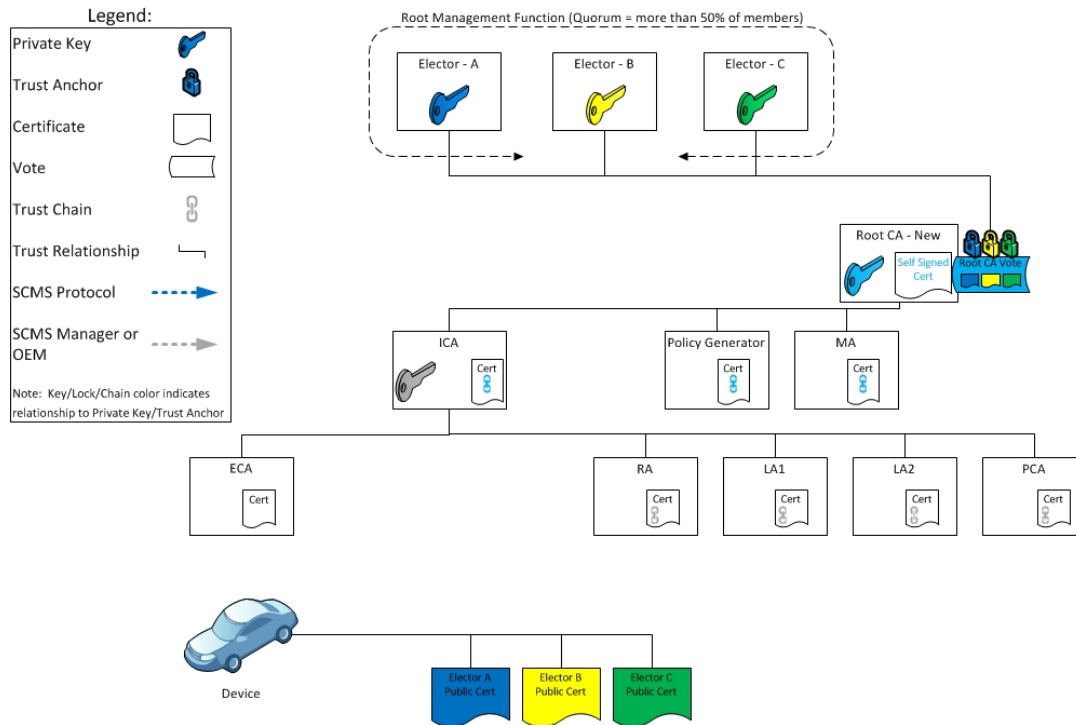


Figure 15 Day 5: Revoked Root CA, System Non-Functional

On Day 5, all of the existing pseudonym and enrollment certificates are no longer valid. This means that from an EE point of view, the SCMS is not functioning. The CRL also needs to be reset: any certificate without linkage values can be removed. The handling of the linkage values on the CRL will depend on if the linkage values are continued. Those that are continued will need to remain on the CRL.

Day 6: System functionality restored

Update EEs with new certificates

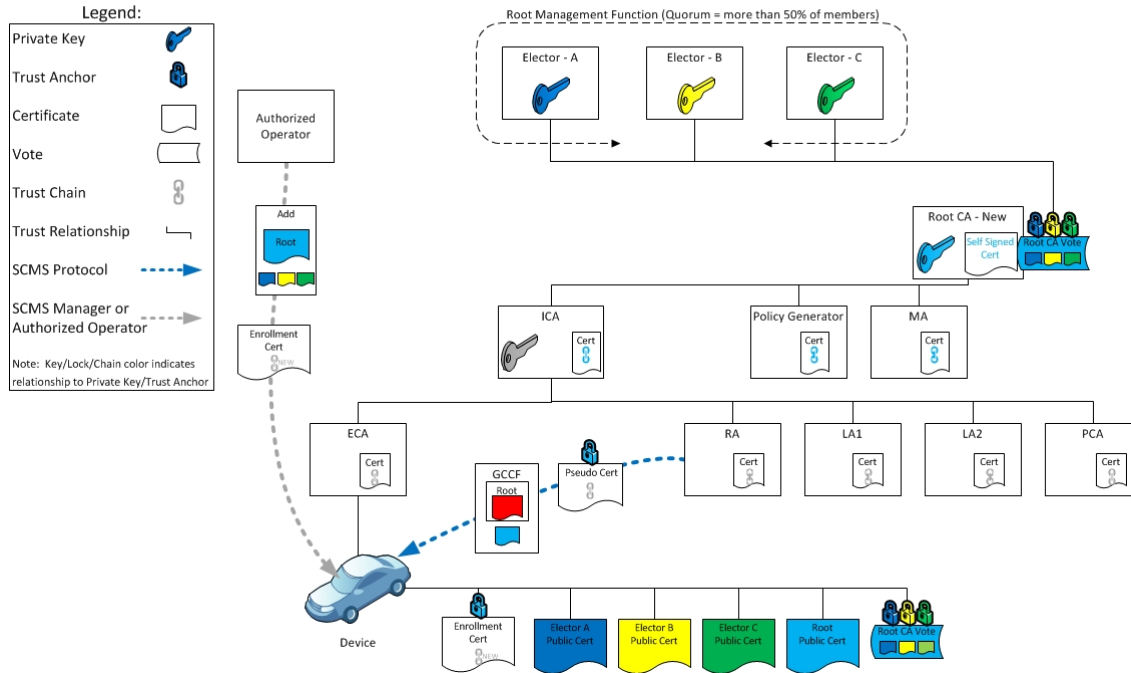


Figure 16 Day 6: System Functionality Restored

On Day 6, the authorized operators will issue new enrollment certificates to the EEs. All EE certificates, including pseudonym certificates, are generated. The EEs require new enrollment certificates to authenticate themselves to their RA. The SCMS does not yet specify the mechanism used to provide new enrollment certificates to EEs; a later release will support this. Once an EE receives its new enrollment certificate, it can download the policy file, the GCCF, and new pseudonym Certificates. The EEs now become operational again.

Cryptography

Approved Cryptographic Algorithms

The following algorithms are approved for use as specified in [IEEE 1609.2-2016](#):

- **Signing:** ECDSA over NIST P-256
- **Public key encryption:** ECIES over NIST P-256
- **Hash:** SHA-256
- **Symmetric Encryption:** AES-CCM with 128-bit keys

See IEEE 1609.2 for normative references to the definitions of the algorithms.

Approved Random Number Generators

This is a non-exhaustive list of recommended software random number generators. Generally, hardware random number generators are preferred. Both types should follow the requirements specified in [CB2: Types of Cryptographic Algorithms](#).

Based on java documentation, a random number can be generated using SecureRandom. This class provides a cryptographically strong random number generator (RNG).

```
public class SecureRandom extends Random
/*
A cryptographically strong random number minimally complies with the statistical random number generator
tests specified in FIPS 140-2, Security Requirements for Cryptographic Modules. SecureRandom must produce
non-deterministic output. SecureRandom is acceptable only if seeding/entropy source is provable sufficiently secure
*/

public static void main(String[] args) t..... {
    SecureRandom ranGen = new SecureRandom();
}
```

Implementation

- A software based RNG solution shall be sufficient through CV pilot until hardware based solutions are identified and accepted.
- Java SecureRandom running on a virtual machine is only acceptable if the host machine entropy is accessible and used by the VM. This can be accomplished by employing utilities such as virtio-rng. Please check your desired VM implementation for support of such a feature.

Testing

- The used RNG shall be tested using the NIST SP800-22b statistical test suite "sts-2.1.1". A description of the test suite ([NIST Special Publication 800-22rev1a](#), dated April 2010) and the NIST statistical test suite software sts-2.1.1 are available at http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html.
- The NIST test suite allows testing an input file of RNG output with various tests. The following tests shall be performed. All tests shall use sufficiently sized input files to the NIST test suite.
 - a. *Test Randomness:* Generate random output of SecureRandom on the VM and run *all* tests of the NIST test suite.
 - b. *Test Seeding:* Generate random output o_1 of SecureRandom on the VM at time t (relative to start-up time). Restart the VM and generate random output o_2 of SecureRandom on the VM at time t (relative to start-up time). Combine o_1 and o_2 in a single file, and run the full NIST test suite.

- c. *Test nonce and reconstruction values*: While an SCMS component operates normally, store the output of SecureRandom in a file and run the full NIST test suite.
- A third party description of proper RNG testing can be found at http://www.st.com/web/en/resource/technical/document/application_note/DM00073853.pdf (cp. sections 2 and 3).

[Cryptography Background](#)

[Special Cryptographic Primitives in SCMS](#)

End-Entity (EE) to Request Authority (RA) Communications – General Guidance

The following is provided as general guidance for EE-RA messaging. For specific messaging, refer to the [RA - Services View](#).

EE initiates all communication between EE and RA. All communications between EE and RA fall into one of two categories: 1) (Non-)Authenticated Download Requests 2) SCMS Protocol Messages.

EE-RA Authentication and RA-EE Authentication

1. EE establishes a secure server-authenticated TLS connection with RA (RA authenticates to EE).
2. EE then digitally signs the current time of type IEEE 1609.2 Time32 with EE's enrollment certificate.
3. EE uses POST to include the IEEE 1609.2 enrollment certificate, the current time of type IEEE 1609.2 Time32, the digital signature over the current time, and the ASN.1 request. Note that this payload is TLS protected.
4. RA validates the enrollment certificate against the internal blacklist, and then verifies the enrollment certificate.
5. RA validates the time-stamp against a configurable time tolerance (default value is defined in [SCMS-1203](#)), and then digitally verifies the signature of the current time.
6. RA grants access to the file to download, if all verifications were successful. Otherwise, RA closes the connection.

A simplified version is displayed in the diagram below. Note that the diagram does not include the digitally signed time-stamp of Step 2 and the verification of Step 5.

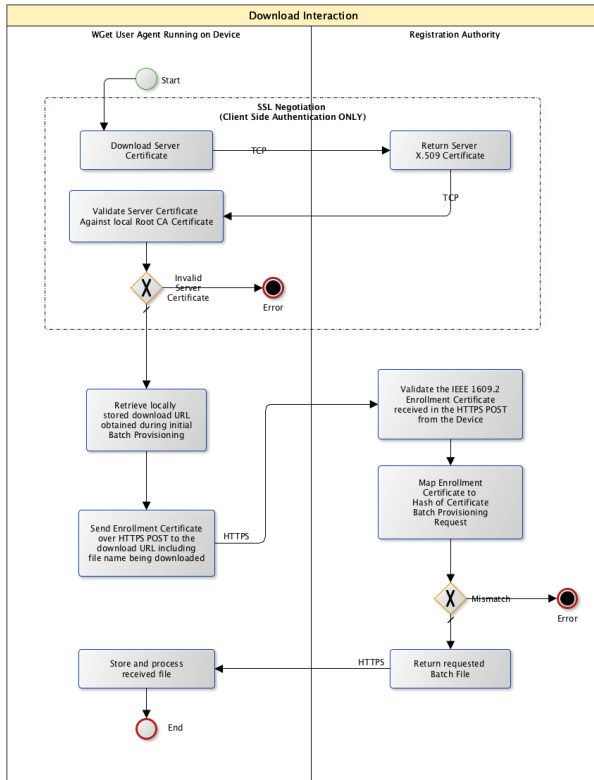


Figure 17 EE-RA Download Interaction

RA Revocation

An X.509 root CA certificate that EEs install during bootstrapping issues RA's X.509 certificate. EE will perform the following check before Step 2 in above EE-RA mutual authentication:

- EE validates whether the X.509 root CA issued RA's X.509 certificate, and whether RA's X.509 certificate is valid.

In order to revoke an RA, the operator will modify the DNS entry for the RA (e.g. ra.ra-host.com) to point to the new RA (or RA's load-balancer/firewall, depending on RA's architecture). Attacks might be still possible; an attacker can compromise the RA X.509 certificate, implement DNS spoofing, and compromise the LOP. However, the adversary's gain is limited to learning enrollment certificates. Therefore, the RA may or may not support a revocation mechanism for RA's TLS certificate (e.g. the certificate status request extension, colloquially known as OCSP stapling and specified in [RFC 6066](http://RFC6066), Section 8). The EE (both OBE and RSE) may or may not support the TLS revocation mechanism.

Download Request

Download requests are used by the EE to download a file from the RA.

The EE uses HTTP GET to make download requests. There are two different kind of download requests: authenticated and non-authenticated:

- In order to provide IEEE 1609.2 based authentication from EE to RA for authenticated download requests an APDU named SignedAuthenticatedDownloadRequest is included in the request. The filename of the file EE is attempting to download and the current time timestamp is included in the SignedAuthenticatedDownloadRequest. The EE uses its enrollment certificate's signing key to create the signature in the SignedAuthenticatedDownloadRequest. A HTTP header with Base64 encoded ASN.1 serialized SignedAuthenticatedDownloadRequest APDU is included in the HTTP GET message.

- Non-authenticated download are plain HTTP GET messages with an optional HTTP Header 'If-None-Match' to identify the version of an already downloaded file.

The HTTP GET Range option may be used to request a partial download for the purposes of resuming a previously interrupted download.

SCMS Protocol Messages

SCMS protocol messages are used by the EE to send SCMS protocol APDU messages to RA. The EE uses HTTP POST to send the SCMS protocol APDU to RA. The EE ASN.1 serializes the APDU and sends it as the HTTP POST Message Body in binary form.

Requirements

- Download requests include requests from EE to RA for the following files:
 - .info
 - [Global Policy File \(GPF\)](#)/[Local Policy File \(LPF\)](#)
 - [Global Certificate Chain File \(GCCF\)](#)/[Local Certificate Chain File \(LCCF\)](#)
 - [OBE pseudonym certificate batch file](#)
 - [RSE application certificate files](#)
 - [OBE identification certificate files](#)
- Download requests shall be sent from EE to RA via HTTP GET.
- Authenticated download requests shall include a HTTP Header with value equal to an ASN1 serialized Base64 encoded SignedAuthenticatedDownloadRequest message.
- APDUs sent from EE to RA via HTTP POST shall include:
 - SecuredRACertRequest
 - SecuredPseudonymCertProvisioningRequest
 - SecuredIdCertProvisioningRequest
- APDUs other than SignedAuthenticatedDownloadRequest shall be sent from EE to RA via HTTP POST.
- APDUs sent from EE to RA via HTTP POST shall sent Content-Type header equal to application/octet-stream.
- APDUs sent from EE to RA via HTTP POST shall be sent in the HTTP Message Body in binary ASN.1 serialized form.

End-Entity (EE) to SCMS Core Communications

Goals

- The goal of the EE-SCMS Core Communication Requirements section is to define all requirements that an EE must follow whenever establishing a connection to the SCMS.
- Individual requirements shall be labeled with their respective use case(s).
- In cases where a specific use case has a conflicting requirement, that use case shall define the new requirement and reference which core requirement is being overridden.

Background and Strategic Fit

IP Address Translation

- Prevent SCMS component (RA, CRL Store, etc.) from learning location information based on the IP address of the EE.
- LOP & SCMS Component must have adequate separation.

TLS Connection

- Provide a means to verify the identity of the SCMS component by using x.509 1-way authentication.
- Encryption is an added privacy preserving enhancement but not a core requirement.

IEEE 1609.2 Encrypting and/or Signing

- Provides application layer security and privacy.

Diagrams of Communications Methods

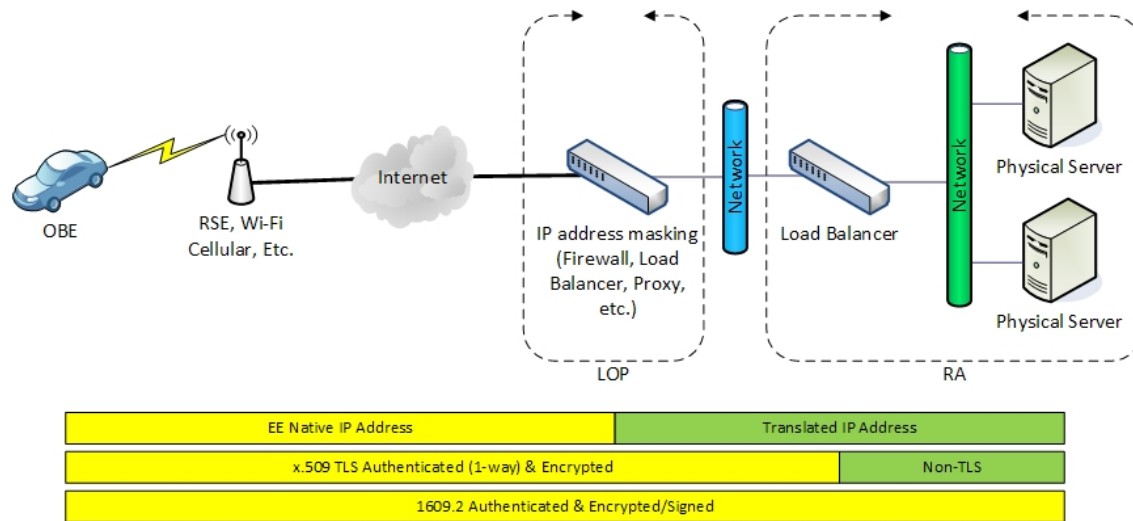


Figure 18 Overview of Methods

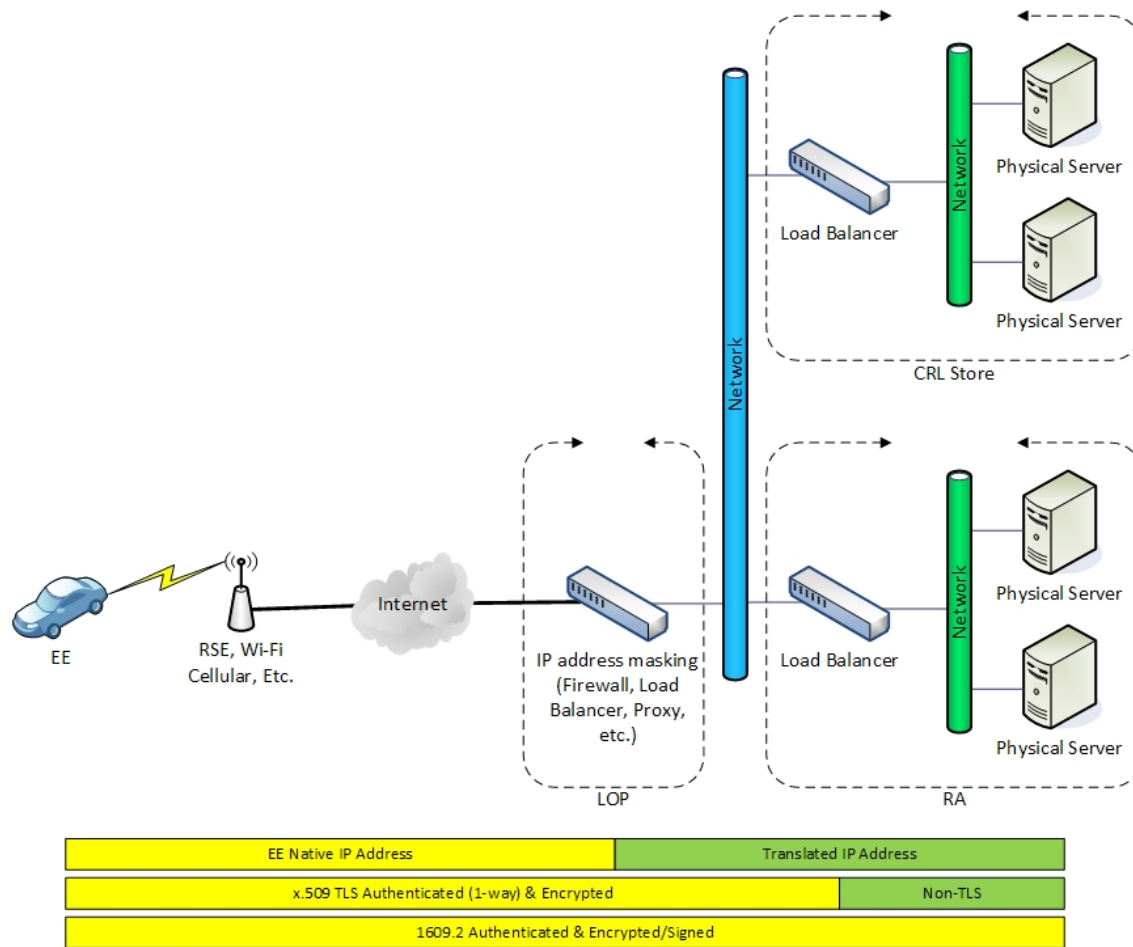


Figure 19 Overview of Multiple SCMS Components Served by Single LOP

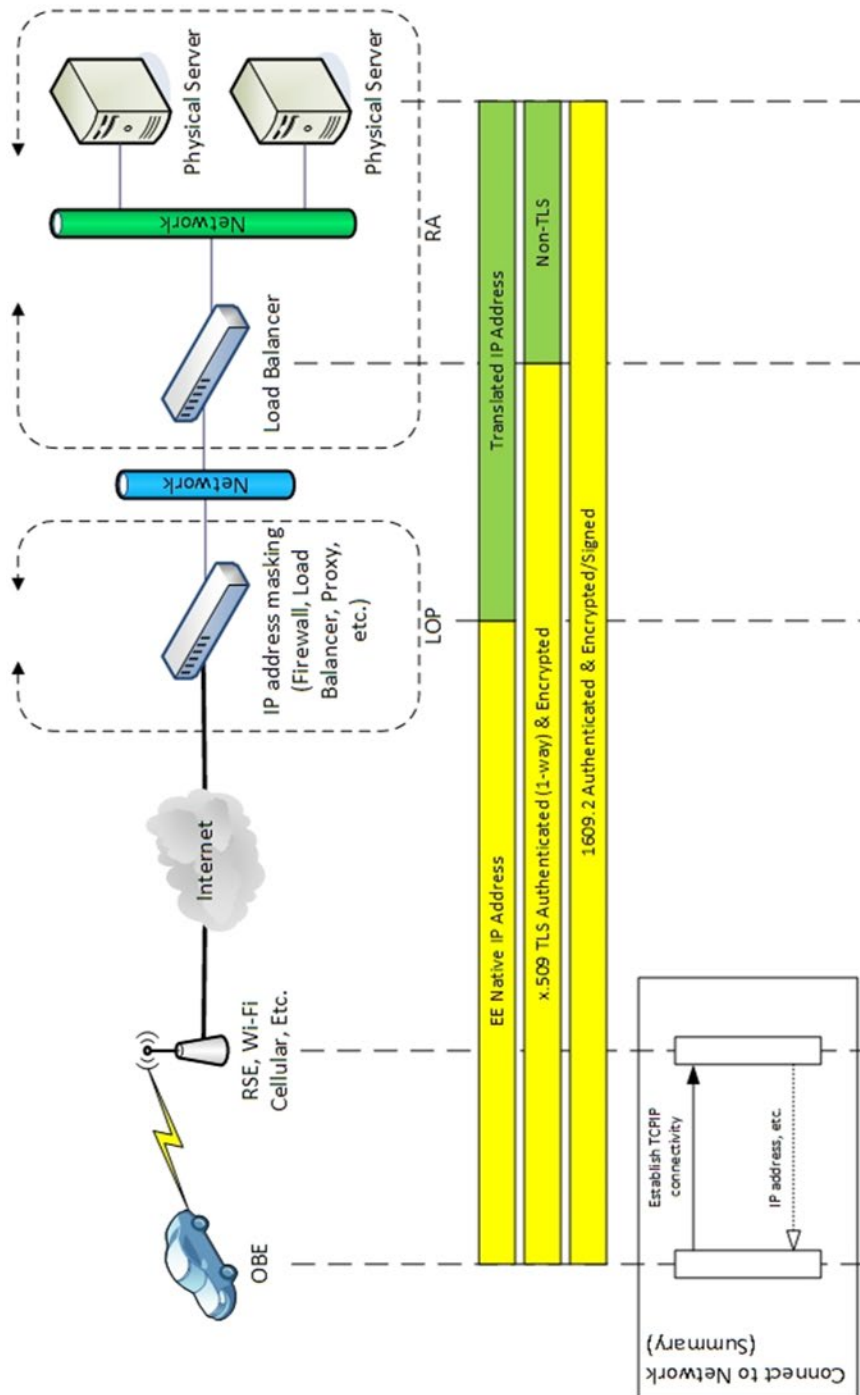


Figure 20 Universal SCMS Handshake Processes, 1 of 5

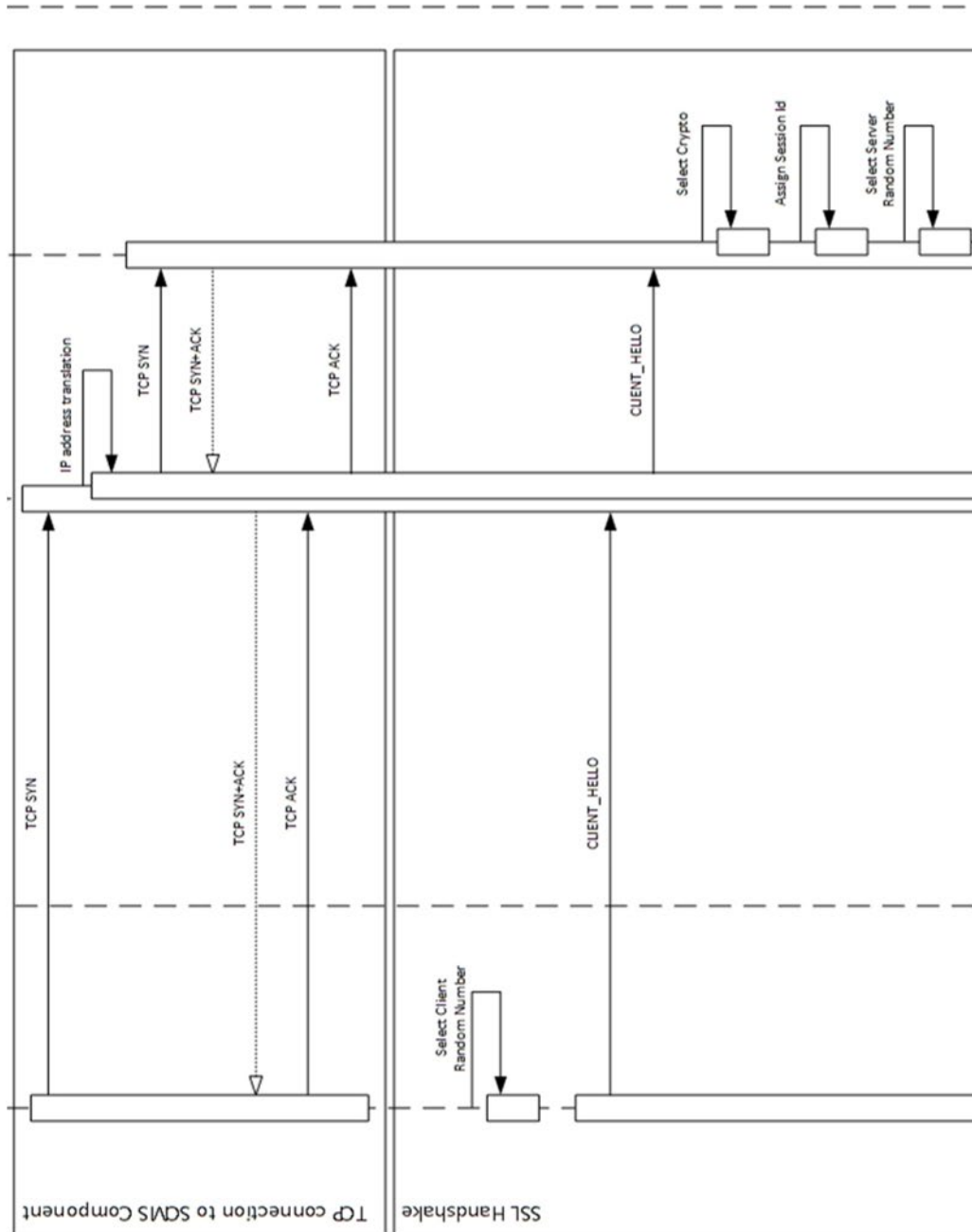


Figure 21 Universal SCMS Handshake Processes, 2 of 5

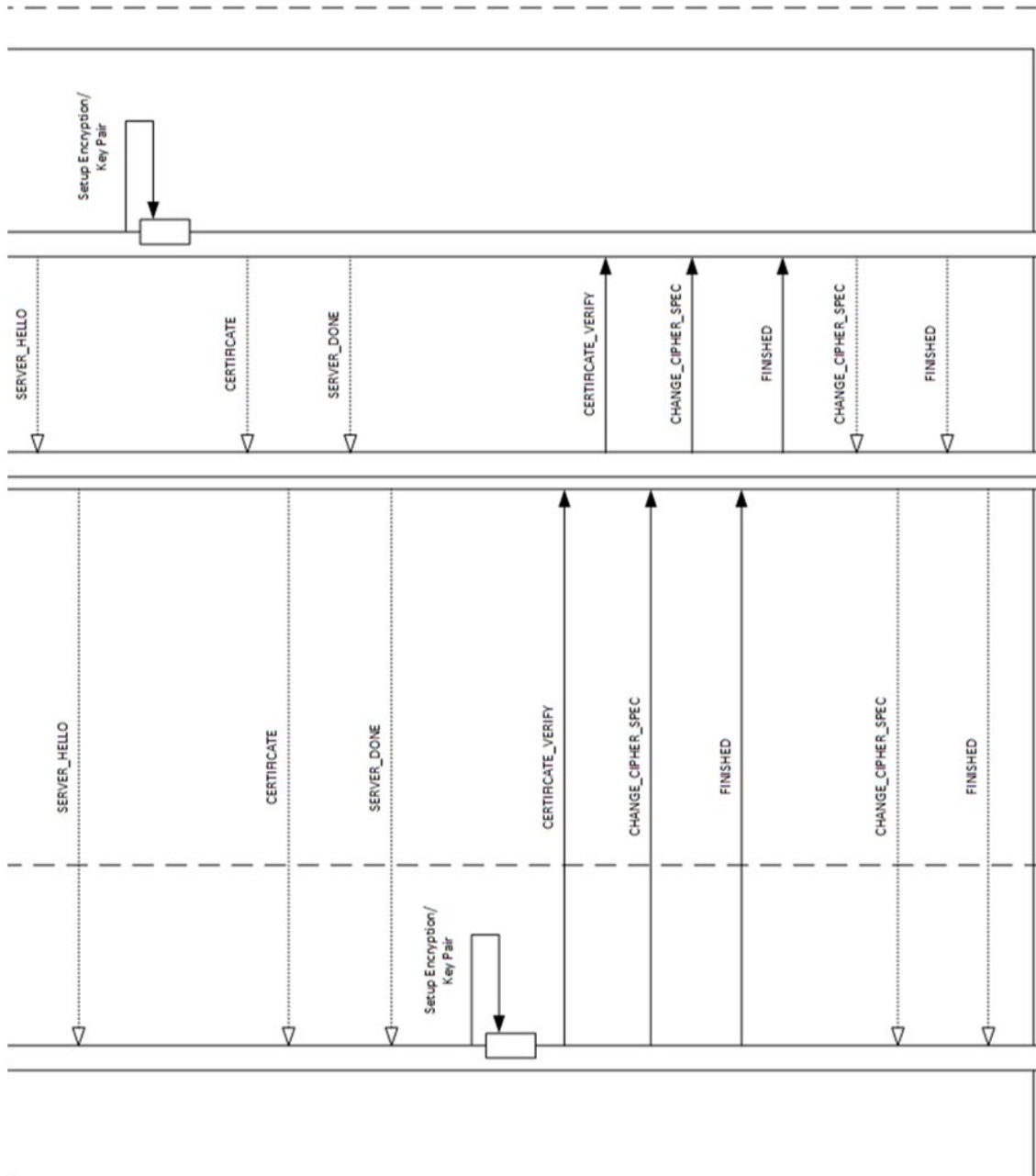


Figure 22 Universal SCMS Handshake Processes, 3 of 5

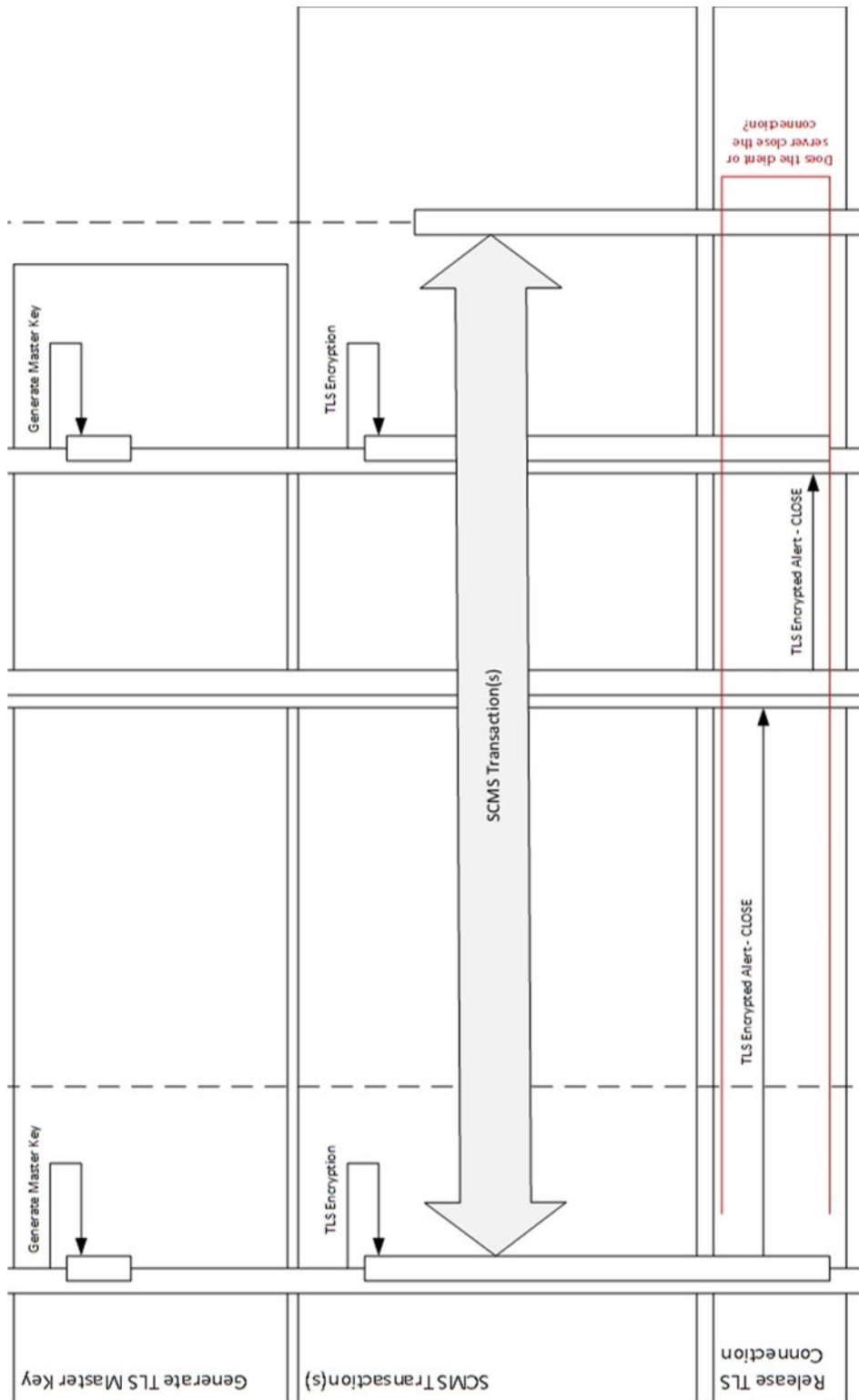


Figure 23 Universal SCMS Handshake Processes, 4 of 5

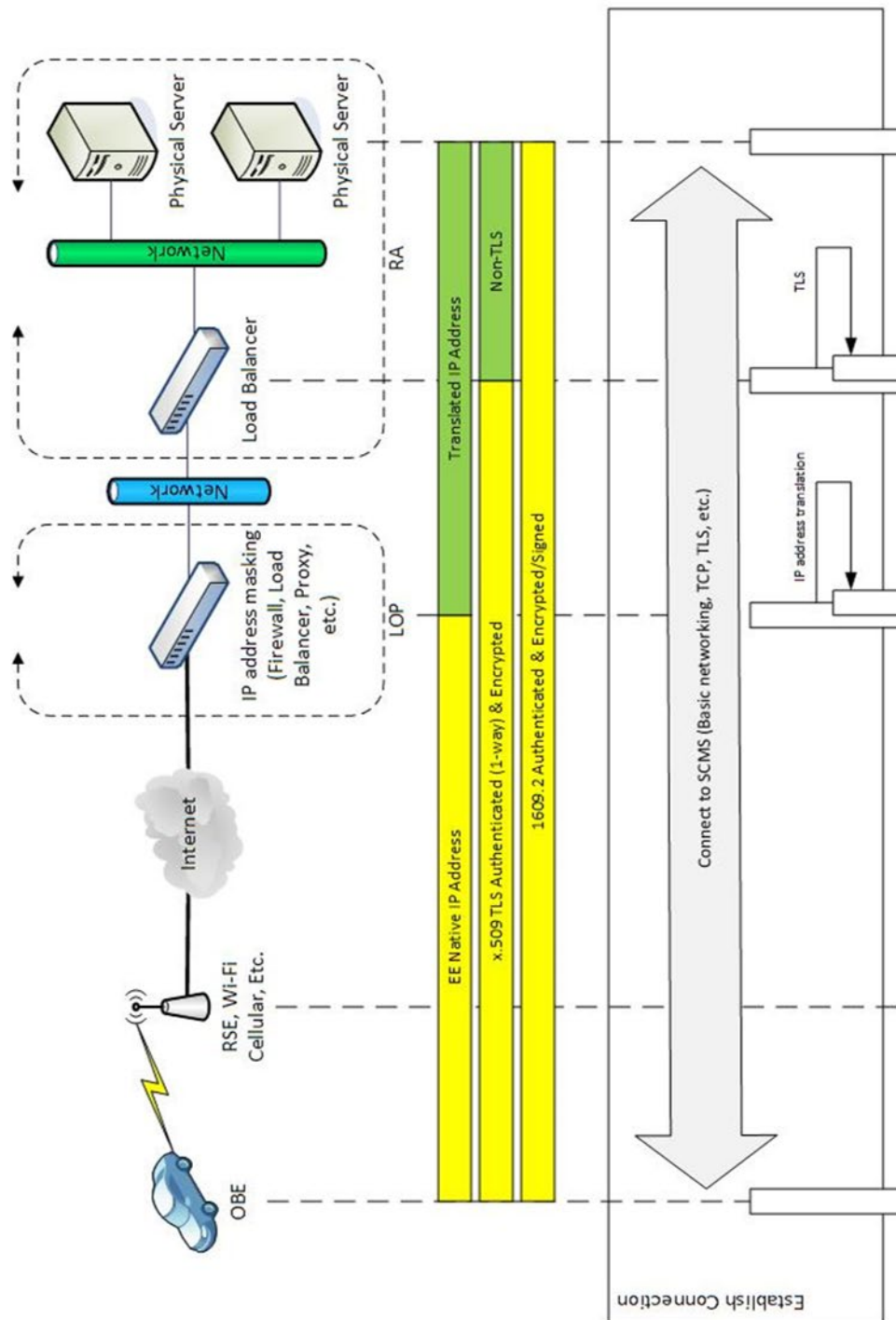


Figure 25 Common Process for File Download Operations, 1 of 3

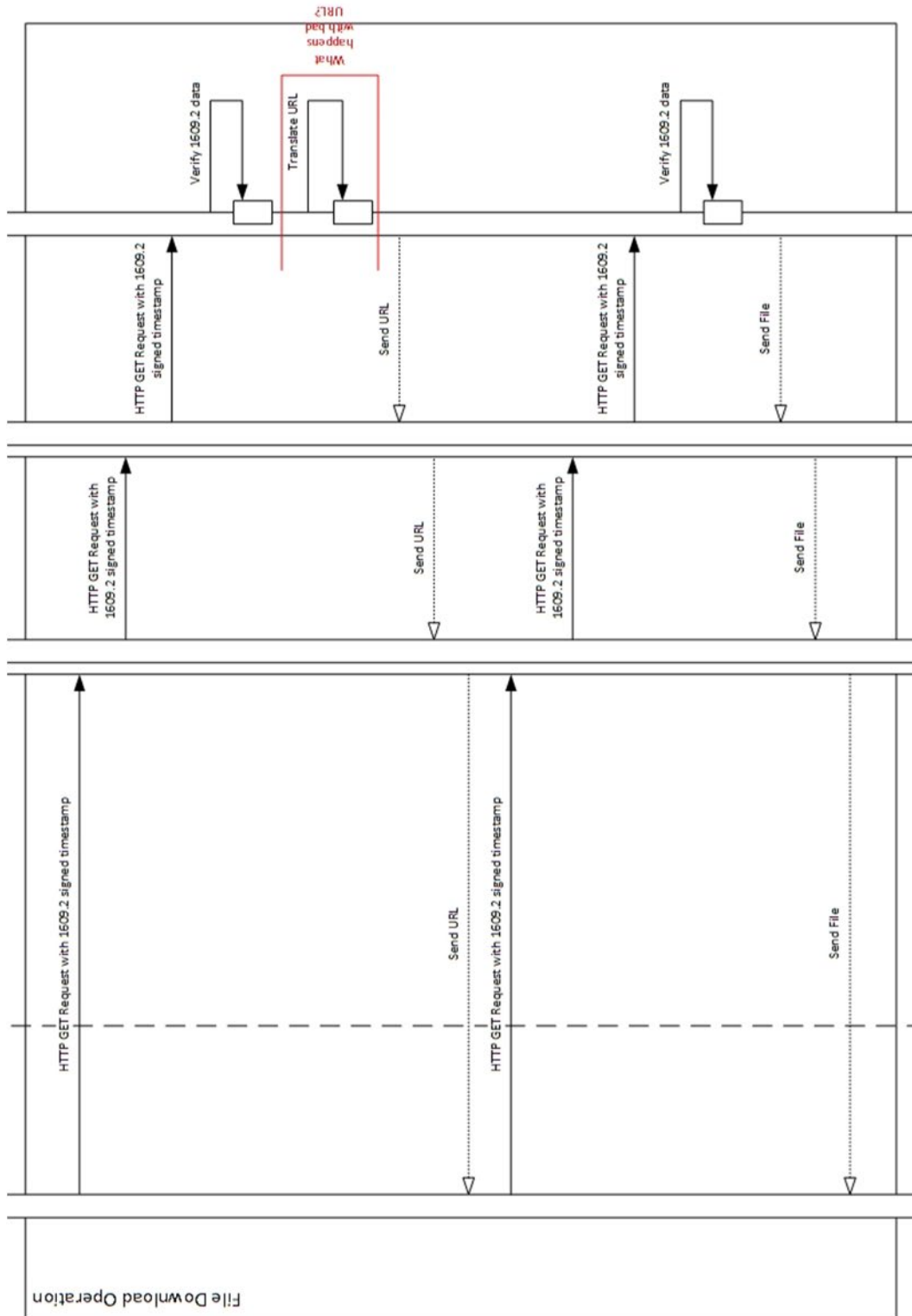


Figure 26 Common Process for File Download Operations, 2 of 3

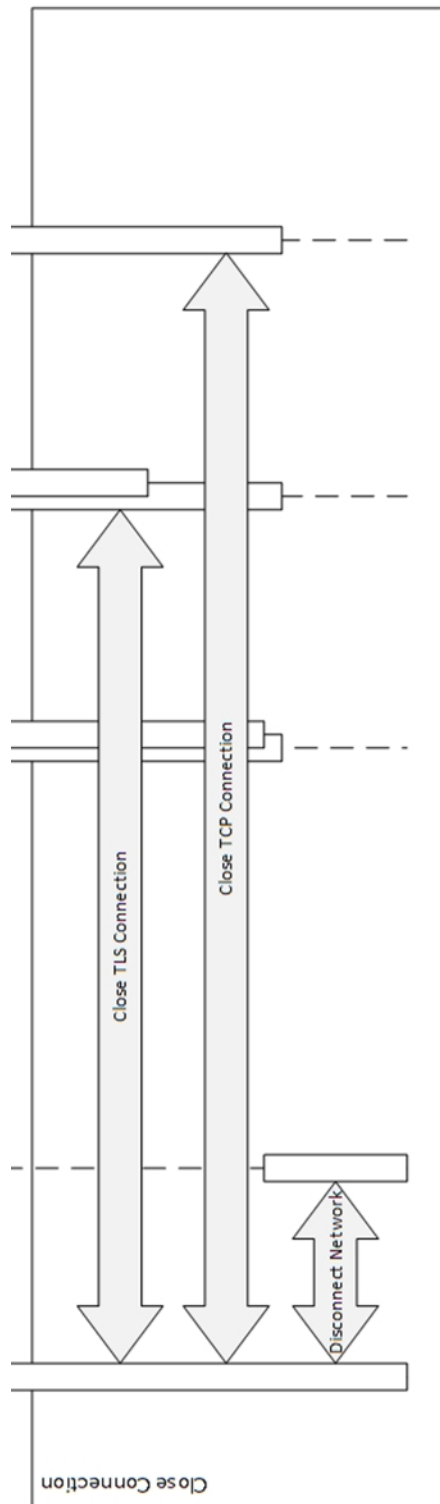


Figure 27 Common Process for File Download Operations, 3 of 3

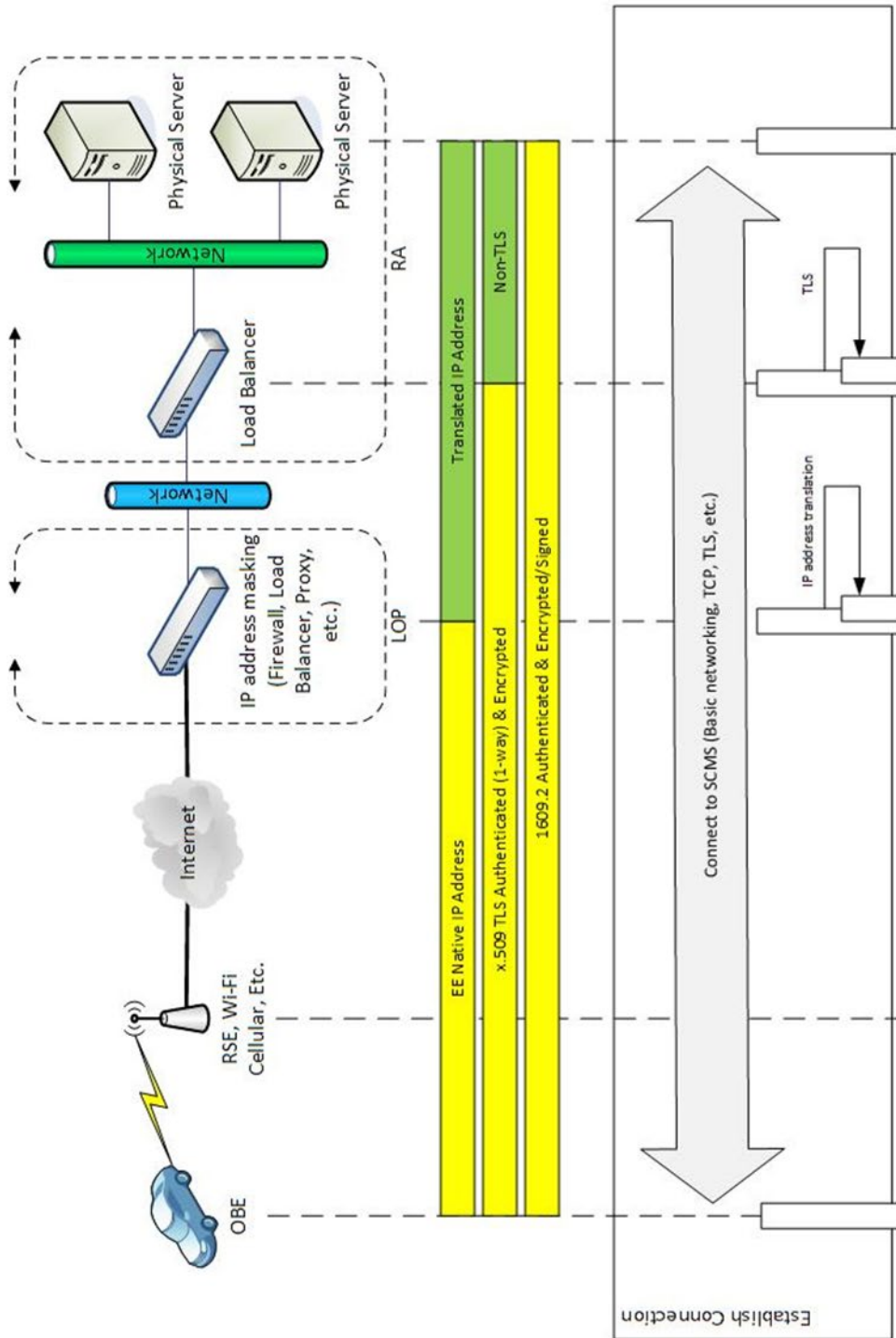


Figure 28 Common Process for Sending SCMS Messages, 1 of 2

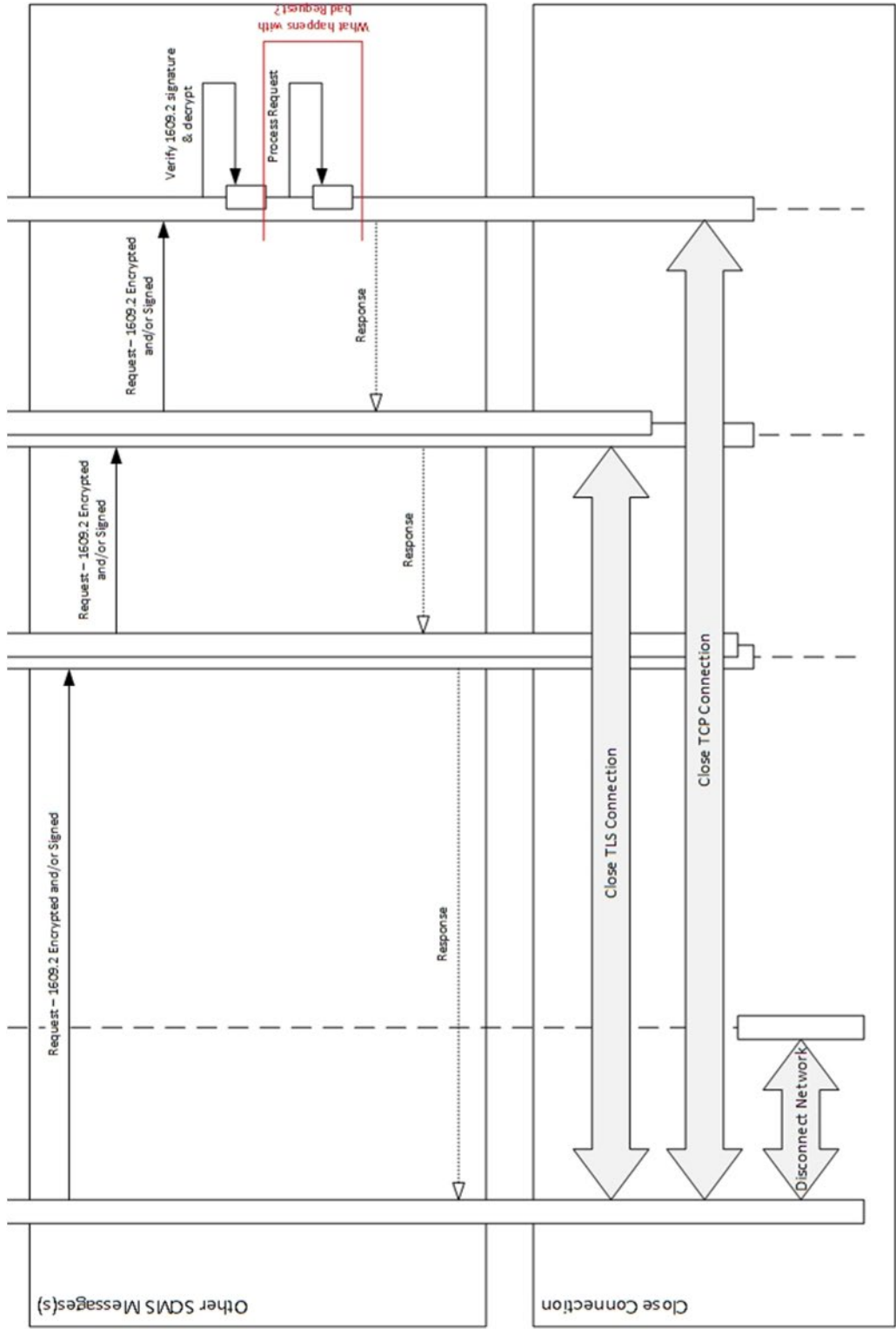


Figure 29 Common Process for Sending SCMS Messages, 2 of 2

OBE, RSE, Common and Backend Uses Cases

The following pages are a hierarchy of requirements sorted by SCMS use cases. A use case contains all requirements that must be implemented from an end entities ([EE](#)) perspective to fulfill a major feature of the SCMS. A use case might comprehend multiple steps from a system's architecture perspective that can be run without interference with each other to return a partial result of the overall use case. In general, steps need to be executed in the given order to fulfill the use case. For example, [Use Case 3: OBE Pseudonym Certificates Provisioning](#) describes all necessary processes to equip an OBE with pseudonym certificates. It comprehends five steps that are coherent but self-contained:

- [Step 3.1: Request for Pseudonym Certificates](#)
- [Step 3.2: Pseudonym Certificate Generation](#)
- [Step 3.3: Initial Download of Pseudonym Certificates](#)
- [Step 3.4: Schedule Generation of Subsequent Batch of Pseudonym Certificates](#)
- [Step 3.5: Top-off Pseudonym Certificates](#)

This format supports end-to-end implementation as well as testing better than a pure listing of requirements.

On-board Equipment (OBE) Use Cases

The following chapters are about OBE requirements. These are the main use cases for OBEs, but there are requirements throughout all chapters for OBEs. For example, in [11. Backend Management](#) are requirements about what an OBE needs to do if a root CA is revoked or a new root CA is introduced to the system.

- [Use Case 2: OBE Bootstrapping \(Manual\)](#)
- [Use Case 3: OBE Pseudonym Certificates Provisioning](#)
- [Use Case 8: Global Misbehavior Detection and Revocation](#)
- [Use Case 19: OBE Identification Certificate Provisioning](#)

Road-side Equipment (RSE) Use Cases

The following chapters are about RSE requirements. These are the main use cases for RSEs, but there are requirements throughout all chapters for RSEs. For example, in [11. Backend Management](#) are requirements about what an RSE needs to do if a root CA is revoked or a new root CA is introduced to the system.

- [Use Case 12: RSE Bootstrapping \(Manual\)](#)
- [Use Case 13: RSE Application Certificate Provisioning](#)
- [Use Case 16: RSE Application and OBE Identification Certificate Revocation](#)

Common EE Use Cases

Both EE types should implement the following chapters:

- [Use Case 5: Misbehavior Reporting](#)
- [Use Case 6: CRL Download](#)
- [Use Case 11: Backend Management](#) (CA compromise recover strategy)
- [Use Case 18: Provide and Enforce Technical Policies](#)

- [Use Case 20: EE Re-Enrollment](#)

Backend Use Cases

Features specific only to the SCMS (no relevance to end entities) as well as deployment and management requirements are listed in the following use cases:

- [Use Case 1: SCMS Component Setup](#)
- [Use Case 7: CRL Broadcast](#)
- [Use Case 11: Backend Management](#)

Requirement Status

All requirements are listed with all details including their status of implementation (e.g., [SCMS-500](#) - Firewall whitelist [SCMS POC OUT OF SCOPE](#)) and a [JIRA](#) link is given for traceability reasons. Statuses given are:

Status	Description
Review	Requirement is currently under review by the Software Team
In Implementation	Requirement is currently in implementation by the Software Team
Implemented	Software Team finished the implementation as well as the unit tests
Ready for Testing	Test Team created test cases as well as test scripts for this requirement and the requirement is ready to be tested with the next test run
Tests Passed	All tests of the given requirement were successful within the latest test run
Tests Failed	One or more tests of the given requirement failed during the latest test run
Closed	Requirement is implemented and successfully tested
Manual Process	Requirement is meant to be manually executed within the PoC software and will not be implemented in software
SCMS PoC Out Of Scope	Requirement will neither be implemented in the PoC software nor executed manually. This applies especially to EE requirements or SCMS production requirements that are listed but out of scope for implementation during the PoC project.

Table 2 Document Header and Status

Use Case 2: OBE Bootstrapping (Manual)

[Note: The manual process for RSE bootstrapping is exactly the same as [Use Case 2: OBE Bootstrapping \(Manual\)](#) for at least the first year of CV pilot SCMS POC operations.]

Table of Contents

- [Background and Goals](#)
- [Assumptions and Preconditions](#)
- [Process Steps](#)
 - [Manual Bootstrapping Process - QA Environment](#)
 - [Manual Bootstrapping Process - PROD Environment](#)
 - [Enrollment certificate request checks](#)
 - [OBE Bootstrap Process Logging Requirement](#)
- [Enrollment Certificate Request Example](#)
- [Requirements](#)
- [Additional Reference Information](#)

- [ASN.1 Specification](#)

Background and Goals

The bootstrap process enables the OBE to interact with the SCMS.

Bootstrapping is executed at the start of the OBE's lifecycle. At the start of bootstrapping, the OBE has no SCMS certificates and no knowledge of how to contact the SCMS. At the end of bootstrapping the OBE has the following:

- Certificates and information that allows an OBE to trust the SCMS:
 - The required [Root CA certificate](#)(s), optional Intermediate CA and Pseudonym CA certificates to allow it to verify received messages. The OBE can learn unknown PCA and ICA certificates in ongoing operation as defined in IEEE 1609.2 P2P CD. At minimum, any EE needs the certificate chain of the PCA that issued certificates to it.
 - The latest CRL (includes the CRL Generator certificate, which in turn includes the FQDN of the CRL store)
 - The MA certificate to encrypt misbehavior reports, before submitting them to the RA
- Credentials and information allowing an OBE to communicate with the SCMS:
 - A correctly issued enrollment certificate, private key reconstruction value, and ECA certificate.
 - The RA certificate (which includes the FQDN of the RA).

Bootstrapping must protect the OBE from getting incorrect information, and the ECA from issuing a certificate to an unauthorized OBE. Any bootstrapping process is acceptable, that results in secure placement of this information on an OBE device.

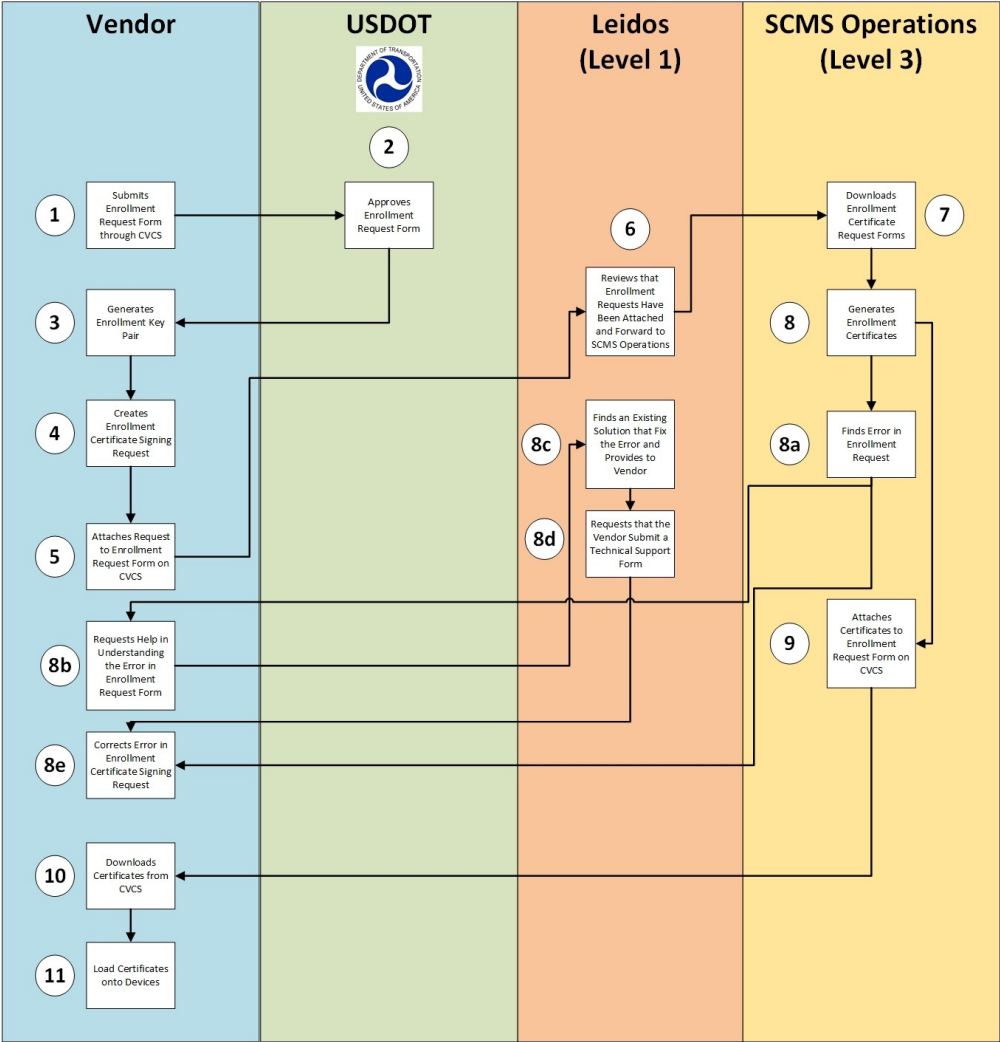
Assumptions and Preconditions

- A documented procedure for performing the enrollment process.
- A “secure environment” as defined in [Secure Environment for Device Enrollment](#), ensures that the OBE is under control of the operator running the bootstrapping operation.
- One or more authorized devices (computers) for managing the enrollment process.
- An activity log or recording of the enrollment operations performed.
- A user account at the USDOT workflow tool.

Process Steps

Manual Bootstrapping Process - QA Environment

The CV Pilot will initially use a manual bootstrapping process that combines device initialization and enrollment. The following process applies to the SCMS QA stage. The vendor will initiate this process by requesting device initialization information and enrollment certificate from a **DOT Workflow Approval** tool, as depicted in this process:



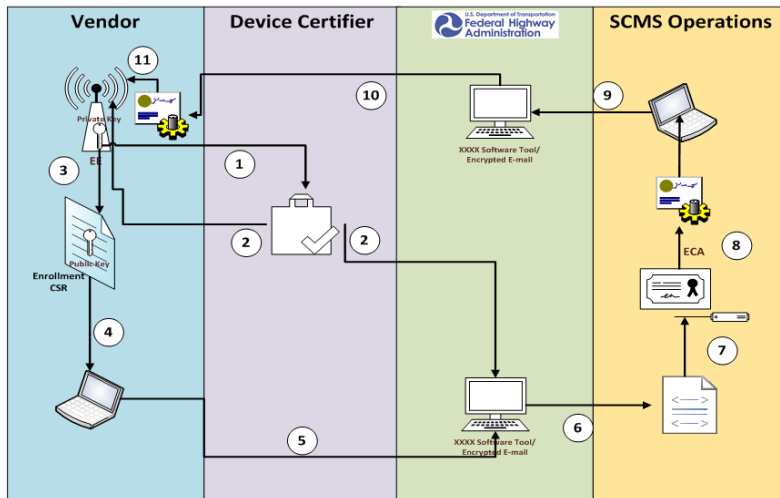
Step	Actor	Description	Status	Assignee
1	Vendor	Logs into CVCS Samanage , initiates an enrollment certificate request. There is a dedicated form for that.	New	USDOT
2	USDOT	<p>Logs into CVCS Samanage and reviews the enrollment certificate request form. They ensure that:</p> <ul style="list-style-type: none"> The vendor is on the list of known vendors for CV device manufacture. If the request is not correct, USDOT will deny the request, and the vendor will need to correct the request and resubmit through Step 3. <p>USDOT Personnel approve the request, if it meets the above criteria, and USDOT sends the request back to the Vendor for them add the enrollment certificate signing request.</p>	Awaiting Customer Input	Leidos
3	Vendor	<p>The vendor in his secure environment generates in each OBE a verification key pair (see Public Key Algorithms in CB2: Types of Cryptographic Algorithms). The private key is used to sign the enrollment certificate request (CSR) in step 4. The public key is added to the request and used by the ECA subsequently as input to calculating the public value within the implicit certificate, issued at end of this process.</p> <p>NOTE: The verification key pair must be generated using an algorithm approved for use (see Approved Cryptographic Algorithms, Approved Random Number Generators). Best practice is to generate the verification key pair inside the EE's HSM and the private key never leaves the EE.</p>	Awaiting Customer Input	Leidos
4	Vendor	<p>The vendor in a secure environment creates an enrollment certificate signing request for each device, a signed structure called SignedEeEnrollmentCertRequest. The CSR includes the verification public key to use to create the public key reconstruction value in the enrollment certificate. The enrollment certificate request permissions (PSIDs, SSPs, Geographic Region) and lifetime are stated in the CSR as well. The vendor signs the CSR with the device's private key, and writes the CSR to a file with filename format <enrollment pub hex>.oer in OER encoding. The vendor then collects multiple CSRs, places them in a flat directory and zips the directory. The directory structure within the zip file should look identical to the following example. IMPORTANT: DUE TO AUTOMATED PROCESSING OF REQUESTS, DEVIATIONS FROM THIS ZIPFILE AND DIRECTORY STRUCTURE WILL RESULT IN REQUESTS FAILING TO BE PROCESSED.</p> <pre style="border: 1px dashed black; padding: 5px;"> + 4A2...BC1.oer + 61C...E1F.oer + ... + ... + 23B1...5FF.oer </pre> <p>Code Block 1 Enrollment Request Zip File Example</p>	Awaiting Customer Input	Leidos
5	Vendor	Vendor logs into CVCS Samanage and attaches the enrollment request zip file to the previous enrollment request form.	Awaiting Customer Input	Leidos
6	Leidos	<p>Reviews Enrollment Request Form and ensures files have been attached and manually verifies the following fields:</p> <ul style="list-style-type: none"> PSID Region 	Assigned	SCMS Operations

Step	Actor	Description	Status	Assignee
7	SCMS Operations	Logs into CVCS Samanage and downloads the enrollment certificate request zip file.	Work in Progress	SCMS Operations
8	SCMS Operations	<p>Executes their enrollment requests script to create enrollment certificates. If successful move to Step 9. The ECA generates and returns an enrollment certificate for each individual request. The response is a signed structure called SignedEeEnrollmentCertResponse. The SCMS operator collects all ECA responses, creates a directory structure that includes bootstrapping information as well as one directory per CSR using the filename of the CSR as directory name. Each of those directories contains the RA certificate to be used by the OBE to communicate with the SCMS, the certificate of the ECA that signed the enrollment certificate, as well as the enrollmentCert itself and the privKeyReconstruction. The SCMS operator zips all files into a single zip file. Following the example in step 4, the directory structure within the zip file would look like this (please be aware that the Root CA certificate is explicitly given in the file root.oer):</p> <pre> + root.oer: IEEE 1609.2 root CA certificate encoded as OER + LCCF.oer: current Local Certificate Chain File including ICA and PCA certificates. + LPF.oer: current Local Policy File + CRL.oer: current Certificate Revocation List + root.tls: TLS (X.509) root certificate RA's TLS cert chains to + 4A2...BC1 (dir) +RA.oer: RA's 1609.2 certificate +ECA.oer: ECA's 1609.2 certificate +enrollment.oer: (EE's enrollment certificate, see enrollmentCert as part of the ECA response SignedEeEnrollmentCertResponse) +enrollment.s: (EE's Private key reconstruction value, see privKeyReconstruction as part of the ECA response SignedEeEnrollmentCertResponse) + 61C...E1F (dir) +RA.oer +ECA.oer +enrollment.oer +enrollment.s + ... + ... + 23B1...5FF (dir) +RA.oer +ECA.oer +enrollment.oer +enrollment.s </pre> <p>Code Block 2 Enrollment Resonse Zip File Example</p>	Work in Progress	SCMS Operations
8a	SCMS Operations	If SCMS Operations finds an error within the request, SCMS Operations will send the Error Response to the Vendor through the CVCS enrollment request.	Awaiting Customer Input	SCMS Operator

Step	Actor	Description	Status	Assignee
8b	Vendor	Requests help/clarification in understanding the error found in the enrollment certificate signing request as a comment to the Enrollment Request Form.	Work in Progress	Leidos
8c	Vendor	Looks for an existing solution that will fix the vendors error. If they find a solution they provide it to the vendor.	Awaiting Customer Input	SCMS Operator
8d	Vendor	If an existing solution cannot be found, Leidos requests the vendor submit the Technical Support form and sends the Vendor the link.	Awaiting Customer Input	SCMS Operator
8e	Vendor	Corrects the error and reattaches the enrollment certificate signing request to the Enrollment Request Form.	Awaiting Customer Input	SCMS Operator
9	SCMS Operator	Logs into the CVCS Samanage and creates an enrollment certificate response for the appropriate vendor and attaches the enrollment response zip file .	Resolved	Vendor
10	Vendor	Vendor logs into CVCS Samanage and downloads their device enrollment certificates in their secure environment .	Resolved	Vendor
11	Vendor	The vendor loads the appropriate enrollment certificate onto the appropriate device, in their secure environment .	Resolved	Vendor

Manual Bootstrapping Process - PROD Environment

The CV Pilot will initially use a manual Bootstrap Process that combines device initialization and enrollment. The process on the SCMS PROD stage is essentially the same as for QA (see QA process above) with the exception that the vendor must first submit their OBE device to a certification lab for certification before requesting the device enrollment certificate. The complete process is described below:



1. Vendor submits their device to one of the device certification companies for certification. Vendor logs into DOT Workflow Approval tool and creates a device certification request, for a specific model of device, selecting the appropriate device certification company.
2. Device certification company conducts device certification testing. After successful completion of certification, device certification company notifies DOT Workflow Approval tool of certification for the specific device model, and attaches certification documentation. DOT Workflow Approval tool notifies the vendor and USDOT of the approval, and maintains device certification documentation in database of certified devices.
3. to 11. Same as step 1-9 in QA

Enrollment certificate request checks

The following checks have to be done in step 6:

- The CSR only contains PSID from [SCMS PoC Supported V2X Applications](#)
- The CSR only contains PSIDs the device is eligible to
- The CSR contains the right SSP values for the requested PSID
- The CSR only contains SSP values the device is eligible to
- The CSR only contains Region USA
- The CSR does not contain a public key that was used with a previous enrollment cert request
- The CSR does have a validity period that fits the ECA's validity period
- The CSR contains the correct cracald
- The CSR contains the correct crlSeries
- The CSR contains a useful Certificateld

OBE Bootstrap Process Logging Requirement

The following bootstrap operation information must be logged and maintained by the organization performing the PROD bootstrapping process, for each unique device, and for each enrollment certificate, if multiple enrollment certificates are requested for a single device.

- OBE serial number or unique unit identifier
- Initial Bootstrap Start Date
- Bootstrap LCCF file version identifier
- Bootstrap LPF file version identifier
- Enrollment cert
- Bootstrap Complete Date

Enrollment Certificate Request Example

The following clear text is an example for an enrollment certificate request that we provide in an [OER encoded version](#), as it is supposed to be sent during manual enrollment.

```
value ScmsPDU ::= {
  version 1,
  content eca-ee : eeEcaCertRequest : {
    version 1,
    currentTime 431026272,
    tbsData {
      id name : "obeenr",
      cracald '000000'H,
      crlSeries 4,
      validityPeriod {
        start 431026272,
        duration hours : 4320
      },
      region identifiedRegion : {
        countryOnly : 124,
        countryOnly : 484,
        countryOnly : 840
      },
      certRequestPermissions {
        {
          subjectPermissions explicit : {
            {
              psid 32,
              sspRange opaque : {}
            },
            {
              psid 38,
              sspRange opaque : {}
            }
          },
          minChainDepth 0,
          chainDepthRange 0,
          eeType {app}
        }
      },
      verifyKeyIndicator verificationKey : ecdsaNistP256 : compressed-y-1 :
      '8751D2FDC5D7BF8CCE4A7FACE5E5AD7B92FA6B8CA0B202FBC93CBC08412AA934'H
    }
  }
}
```

Code Block 3 Clear Text Before Signing/Encrypting


```
value SecuredScmsPDU ::= {  
  protocolVersion 3,  
  content signedCertificateRequest :  
  '00018180000119B0F0604481066F6265656E72000000000419B0F0608410E083010380'H -- truncated --  
}
```

Code Block 4 Textual After Signing/Encrypting (SecuredScmsPDU Layer)

```
038381a500018180000119b0f0604481066f6265656e72000000000419b0f0608410e083010380007c8001e48003480101808  
0010280012080010080012680010001008080838751d2fdc5d7bf8cce4a7face5e5ad7b92fa6b8ca0b202fbc93cbc08412aa934  
828080301d57f8d01e98c685428c49328be8164bae24e18d46030048911c5fd4275df73121b89c7919fd75d7ab411cfb254a446  
60997f7b1ae9235f2d0f1949198826
```

Code Block 5 Binary (Hexadecimal) After Signing/Encrypting

```

value SignedCertificateRequest ::= {
  hashId sha256,
  tbsRequest {
    version 1,
    content eca-ee : eeEcaCertRequest : {
      version 1,
      currentTime 431026272,
      tbsData {
        id name : "obeenr",
        cracald '000000'H,
        criSeries 4,
        validityPeriod {
          start 431026272,
          duration hours : 4320
        },
        region identifiedRegion : {
          countryOnly : 124,
          countryOnly : 484,
          countryOnly : 840
        },
        certRequestPermissions {
          {
            subjectPermissions explicit : {
              {
                psid 32,
                sspRange opaque : {}
              },
              {
                psid 38,
                sspRange opaque : {}
              }
            },
            minChainDepth 0
          }
        },
        verifyKeyIndicator verificationKey : ecdsaNistP256 : compressed-y-1 :
'8751D2FDC5D7BF8CCE4A7FACE5E5AD7B92FA6B8CA0B202FBC93CBC08412AA934'H
      }
    },
    signer self : NULL,
    signature ecdsaNistP256Signature : {
      r x-only : '301D57F8D01E98C685428C49328BE8164BAE24E18D46030048911C5FD4275DF7'H,
      s '3121B89C7919FD75D7AB411CFB254A44660997F7B1AE9235F2D0F19491988265'H
    }
  }
}

value ScmsPDU ::= {
  version 1,
  content eca-ee : eeEcaCertRequest : {
    version 1,
    currentTime 431026272,
    tbsData {
      id name : "obeenr",
      cracald '000000'H,
      criSeries 4,
      validityPeriod {
        start 431026272,
        duration hours : 4320
      },
      region identifiedRegion : {
        countryOnly : 124,
        countryOnly : 484,
        countryOnly : 840
      },
      certRequestPermissions {
        {

```

```
subjectPermissions explicit : {
  {
    psid 32,
    sspRange opaque : {}
  },
  {
    psid 38,
    sspRange opaque : {}
  }
},
minChainDepth 0
},
verifyKeyIndicator verificationKey : ecdsaNistP256 : compressed-y-1 :
'8751D2FDC5D7BF8CCE4A7FACE5E5AD7B92FA6B8CA0B202FBC93CBC08412AA934'H
}
```

Code Block 6 Textual After Signing/Encrypting (SignedCertificateRequest Layer)

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-364	MANUAL PROCESS	DCM Configuration of EEs After Component Revocation	DCM shall not configure new EEs with credentials of revoked SCMS component.	The SCMS Manager will manage the transition of devices after the revocation of a component.	In the PoC this will occur by a manual process. The DCM will provision EEs with valid certificates for SCMS components including one or more ICA and one or more RA. When the DCM learns that any component is revoked, it shall no longer provision new EEs with that revoked certificate.	DCM
SCMS-486	MANUAL PROCESS	DCM shall acquire the current CRL	The DCM shall acquire the current CRL from the CRL Store.	The DCM will provide the latest CRL to newly provisioned EEs. This saves the EE from having to get the CRL right away.	The DCM will request these from the CRL Store and will provide these to the EE.	DCM
SCMS-557	EE REQUIREMENT	Secure chain of custody	EE shall get firmware, enrollment certificates, etc. injected within a secure chain of custody.	Documented and audited processes are crucial to the security of EEs.	See Secure Environment for Device Enrollment for guidelines on physical security for device provisioning. Not software testable, procedural	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-562	CLOSED	RA certificate and FQDN	DCM shall provide the EE with the RA certificate and the FQDN for the RA.	The EE will need to communicate securely with the RA (e.g., to request new certificates).		DCM
SCMS-563	CLOSED	ECA certificate and FQDN	DCM shall provide the EE with the ECA certificate and the FQDN for the ECA.	The EE will need to communicate securely with the ECA.		DCM

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-564	CLOSED	MA certificate and FQDN	DCM shall provide the EE with the MA certificate.	The EE will need to communicate securely with the MA (e.g., in order to download CRLs)		DCM
SCMS-565	CLOSED	ICA certificates	DCM shall provide the EE with its own ICA certificate. Optionally, include other existing ICA certificates.	The EE needs its ICA certificate, e.g., to provide this to other EE in peer-to-peer certificate updates.		DCM
SCMS-566	CLOSED	PCA certificates	DCM shall provide the EE with its own PCA certificate. Optionally, include other existing PCA certificates.	The EE needs its PCA certificate, e.g., to provide this to other EE in peer-to-peer certificate updates.		DCM
SCMS-567	CLOSED	CRL	DCM shall provide the EE with the latest CRL and contact information for the CRL (CRACA certificate is part of the CRL).	The EE will be provided with the current CRL so as to reject communication from invalidated devices.		DCM
SCMS-568	CLOSED	X.509 certificate	DCM shall provide the EE with the Root X.509 TLS certificate.	The EE will need to communicate securely, at the TLS level, with the RA (e.g., in order to download certificates) and the MA (to upload misbehavior reports).	Revocation status shall be available online, e.g., via OCSP.	DCM
SCMS-570	SCMS POC OUT OF SCOPE	Certification Services	Certification Services shall utilize a secure connection to provide attestation to the ECA that the EE is of a type it certified	So that valid EEs are certified and uncertified EEs cannot get enrollment certificates.	Does not apply to POC. For PoC every EE requesting an enrollment certificate is assumed to be certified.	Certification Service
SCMS-573	EE REQUIREMENT	Secure Key Injection	EE shall generate the private key for the enrollment certificate or the DCM shall use a secure key injection mechanism to provide it to the EE.	To maintain confidentiality of private keys	Does not apply to POC	DCM, On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-946	CLOSED	Root CA certificates	DCM shall provide the EE with all Root CA certificates.	The Root CA will have signed the current ICA certificate as well as the centralized components, the Policy Generator and the Misbehavior Authority.		DCM
SCMS-948	CLOSED	Bootstrap: Local Certificate Chain File	DCM shall provide the EE with the latest Local Certificate Chain File.	The EE will use this in the verification process of SCMS certificates.		DCM
SCMS-949	EE REQUIREMENT	Error code: eeInitCertProvFailed	EE shall log this error code, if the Initialization process fails at completing a certificate provisioning of any of the certificates	The EE must signal an error, if any, in the provisioning of any of the certificates.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-950	EE REQUIREMENT	Error code: eeInitCRLProvError	EE shall log this error code, if the Initialization process fails at completing the CRL provisioning.	The EE must signal an error, if any, in the provisioning of the CRL.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1095	CLOSED	RSE Enrollment	RSE enrollment shall be the same as OBE enrollment as specified in Step 2.2: Enrollment (Bootstrapping)	RSE enrollment is the same in terms of process and the resulting certificate.		ECA
SCMS-1160	EE REQUIREMENT	EE securely stores Root CA certificates	EE shall store all root CA certificates in tamper-resistant (or equivalent) storage.	Root CA certificates must be protected against manipulation. It is public and no read protection is required, however, it must be stored in secure storage so that it can only be updated when the proper root (elector) Management authentication mechanisms have been satisfied.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1174	EE REQUIREMENT	EE stores the Policy Generator certificate	EE shall store the Policy Generator certificate.	The EE requires this to validate the signature on Policy Files.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1176	EE REQUIREMENT	EE stores the CRLG certificate	EE shall store the Certificate Revocation List Generator certificate.	The EE requires this to validate the signature on the CRL.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1205	CLOSED	Policy Generator certificate	DCM shall provide the EE with the Policy Generator certificate.	The EE requires this to validate the signature on Policy Files.		DCM
SCMS-1206	CLOSED	Certificate Revocation List Generator certificate	DCM shall provide the OBE with the Certificate Revocation List Generator (CRLG) certificate.	The OBE requires this to validate the signature on the CRL.		DCM
SCMS-1207	EE REQUIREMENT	EE securely stores Certificate Revocation List	EE shall store the Certificate Revocation List in tamper-resistant (or equivalent) storage.	The EE will be provided with the current CRL so as to reject communication from invalidated devices.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1208	EE REQUIREMENT	EE securely stores X.509 root certificate	EE shall store the X.509 root certificate in tamper-resistant (or equivalent) storage.	The EE will need to communicate securely, at the TLS level, with the RA (e.g., in order to download pseudonym certificates) and the MA (to upload misbehavior reports).	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1209	EE REQUIREMENT	EE securely stores Local Certificate Chain File	EE shall store the Local Certificate Chain File in tamper-resistant (or equivalent) storage.	EE will use Local Certificate Chain File during verification of SCMS certificates	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1210	EE REQUIREMENT	EE Secure Key Storing	EE shall store the following keys in tamper-resistant (or equivalent) storage: <ul style="list-style-type: none"> Private enrollment key Butterfly key parameters (seed + expansion) 	To avoid extraction of private keys via software-based attacks.	This is out of scope since it defines EE's behavior. It is highly recommended to protect the content encryption key by a TPM-like mechanism that offers secure boot and that protects the keys	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
			<p>function parameter)</p> <ul style="list-style-type: none"> All private keys (e.g., of OBE application certificates and private keys calculated from the Butterfly key parameters) 		<p>against software-based attacks. Additional details are listed in Hardware, Software and OS Security</p>	
SCMS-1305	CLOSED	PSID in enrollment certificate	ECA shall assign each Enrollment Certificate at least one PSID.	Each enrollment certificate is associated with a particular application that is represented by a PSID/SSP combination. Enrollment certificates cannot have an empty PSID field.		ECA
SCMS-1306	REVIEW	ECA: Not more than one enrollment certificate with same PSID/SSP combination	ECA shall not issue more than one enrollment certificate per requested public key.	A clear mapping is required for proper administration.	In cases where an enrollment certificate has more than one PSID, the corresponding apps are expected to be similar in nature. Such groupings of PSIDs in an enrollment certificate are likely to be related to policy decisions to be made by the SCMS Manager.	ECA
SCMS-1411	SCMS POC OUT OF SCOPE	CV pilots: DCM keep track of generated enrollment certificates	The Single Point of Contact (SPOC) of the DCMs shall keep track of all issued enrollment certificates for the CV pilot deployment.	To be able to revoke all devices from a supplier that was not able to securely handle his enrollment certificates/part of the enrollment process.	This is out of scope for PoC as it defines a manual process for CV pilot operations that is not part of the SCMS PoC project.	DCM
SCMS-1419	CLOSED	ECA issues implicit certificates	ECA shall issue implicit OBE and RSE enrollment certificates	To save storage space and over-the-air bytes		ECA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1441	SCMS POC OUT OF SCOPE	DCM: Not more than one enrollment certificate per PSID/SSP	DCM shall not allow that a single EE requests more than one enrollment certificate associated with the same PSID/SSP values.	To avoid that an EE can receive multiple sets of certificates via different enrollment certificates for a single application (PSID/SSP).	This is enforced by policy mechanisms (e.g., audit). There are no technical means for ECA to validate that an EE didn't request several enrollment certificates for the same PSID/SSP.	DCM
SCMS-1600	CLOSED	Enrollment certificate lifetime	ECA shall issue Enrollment Certificates with an expiration date on or before 00:00:00 UTC January 1, 2025.	To avoid any need to update enrollment certificates during the CV-Pilot project.	Maximum life span 1,084 sixtyHours. This is for CV-Pilot only.	ECA
SCMS-1906	EE REQUIREMENT	Enrollment certificate corresponds to the private key	The enrollment key-pair generator (OBE, RSE, or DCM) shall check that the enrollment certificate corresponds to the private key	This is necessary because otherwise the device won't be able to use the enrollment certificate for requesting pseudonym/identification/application certificates.	If re-enrolling, no DCM is available and this check must be done by the EE.	DCM, On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1907	EE REQUIREMENT	Enrollment certificate verification	The enrollment key-pair generator (OBE, RSE, or DCM) shall check that the enrollment certificate correctly verifies, including building a chain back to the root CA.	This is necessary because otherwise the device won't be able to use the enrollment certificate for requesting pseudonym/identification/application certificates.		DCM, On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1910	EE REQUIREMENT	Verification key pair generation algorithm	EE shall generate the verification key pair using an algorithm approved for use within the SCMS.	Because only those algorithms will be supported by the SCMS.	See Approved Cryptographic Algorithms This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

[35 issues](#)

Additional Reference Information

- [CB2: Types of Cryptographic Algorithms](#)
- [Approved Cryptographic Algorithms](#)
- [Approved Random Number Generators](#)

ASN.1 Specification

- [scms-protocol.asn](#)
- [eca-ee.asnhttps://stash.campllc.org/projects/SCMS/repos/scms-asn/browse/dcm-ee.asn](https://stash.campllc.org/projects/SCMS/repos/scms-asn/browse/dcm-ee.asn)
- [scms-policy.asnhttps://stash.campllc.org/projects/SCMS/repos/scms-asn/browse/dcm-ee-errors.asn](https://stash.campllc.org/projects/SCMS/repos/scms-asn/browse/dcm-ee-errors.asn)

Use Case 3: OBE Pseudonym Certificate Provisioning

Goals

The goal is to provide a freshly bootstrapped OBE with the very first batch of pseudonym certificates that it can use in applications like Basic Safety Message (BSM).

Background and Strategic Fit

The initial provisioning of pseudonym certificates is the process by which an OBE receives its very first batch of pseudonym certificates. This use case also acts as a trigger for subsequent provisioning of pseudonym certificates. The OBE does not need to make any more requests, the RA automatically does everything necessary (such as doing the butterfly key expansion, getting pre-linkage values from the LAs, making individual certificate requests to the PCA, etc.) for the next batches of certificates.

Due to the time constraints imposed by the OEMs, shuffling requirements for the initial provisioning may be relaxed.

This use case involves the following SCMS components:

- Linkage Authorities (LAs)
- Location Obscurer Proxy (LOP)
- Pseudonym Certificate Authority (PCA)
- Registration Authority (RA)

At the start of this use case, the OBE has no pseudonym certificates. At the end of this use case, the OBE has three years' worth of pseudonym certificates, and the RA has everything it needs from the OBE for generating and providing subsequent pseudonym certificate batches for the OBE.

Assumptions

In order to facilitate the certificate request process, an OBE must meet the following prerequisites:

- OBE has a valid enrollment certificate
- OBE has Root CA, RA and PCA certificates installed
- OBE knows the FQDN of the RA

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campllc.org/browse/SCMS-859 , SCMS-504) and the X.509 CRL (https://jira.campllc.org/browse/SCMS-405)	RA
SCMS-510	CLOSED	Keep interactions as independent as possible	RA shall keep the interactions with the device, the LAs, and the PCA as independent as possible	so that organizational separation is maintained	Not software testable, but should be checked in code review. RA should simply follow the protocol.	RA

Table 3 Use Case 3 - Requirements

[2 issues](#)

Design

The following flow chart documents the general flow of steps an OBE needs to carry out in the given order to obtain Pseudonym certificates. It is not a 100% accurate description of the process. Please refer to the use case's steps and their requirements in the following subsections for a complete description of the process.

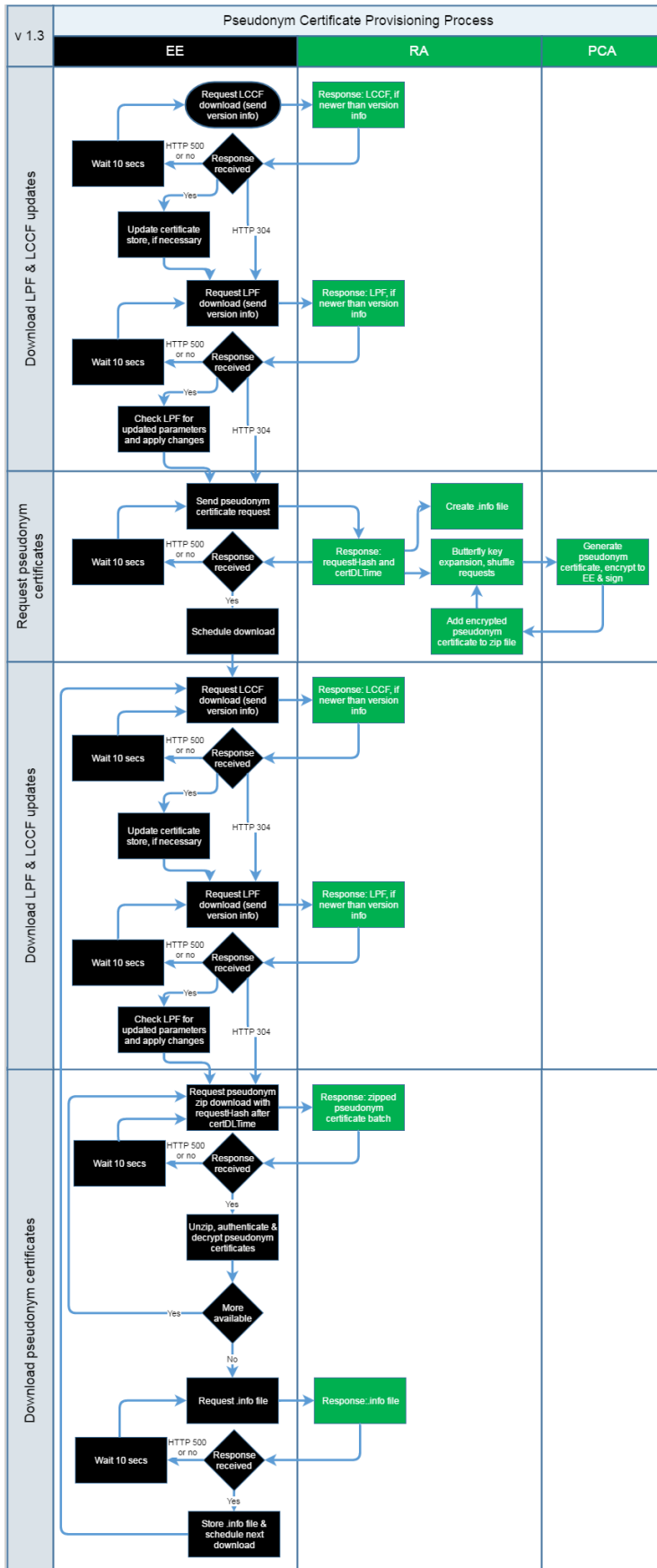


Figure 30 Pseudonym Certificate Provisioning Process

At a high level, three steps are relevant towards an OBE:

1. [Request for Pseudonym Certificates](#)
2. [Initial Download of Pseudonym Certificates](#)
3. [Top-off Pseudonym Certificates](#)

Having determined which RA to submit the request to, the OBE creates a request, signs it with the enrollment certificate, encrypts the signed request for the RA and sends it to the LOP/RA. The LOP strips any IP information that could be used to determine the OBE's location and forwards it to the RA. The RA checks to make sure that the certificate request is correct and authorized and sends back a download location (*requestHash*) and time (*certDLTime*). The RA performs butterfly key expansion on the request to create a batch of public keys to be certified. The RA then merges the certificate request information with linkage information from the LAs to create a series of individual certificate requests. RA then sends those requests to the PCA, mixing the certificate requests with certificate requests generated for other OBEs to provide privacy against insiders at the PCA. The PCA signs the pseudonym certificates, encrypts them for the OBE, signs the encrypted version of the certificate, and returns the encrypted and signed pseudonym certificates to the RA. The RA does not remove any of the named signatures or encryptions, adds them to a zip file and stores them for download by the OBE. The OBE starts downloading the zip files at *certDLTime*.

Step 3.1: Request for Pseudonym Certificates

Goals

The goal of this use case is to define the messages and actions which allow a device to request new pseudonym certificates from the RA. An initial request is for 3,000 (3,120 to be exact) certificates and is assumed to be the default for a batch request. (20 pseudonym certificates per week x 52 weeks per year x 3 years). Note: 20 pseudonym certificates is minimum number of certificates per week. Each OEM can decide to have more certificates per week. The number of requested certificates per week changes the number of request towards PCA and, therefore, requires more computational and storage capacity at the PCA.

Background and Strategic Fit

Whenever the SCMS Manager decides to change technical policies for the SCMS, all participating devices must be updated. Therefore, the RA provides a [Local Policy File \(LPF\)](#) based on the [Global Policy File \(GPF\)](#) generated and signed by the Policy Generator. The Policy Generator as well signs the LPF. The OBE must download the [LPF](#) and [Local Certificate Chain File \(LCCF\)](#) before sending any subsequent request or any certificate download every time it connects to the RA.

The OBE must request pseudonym certificates from its RA within the overall policy set by the SCMS Manager in the LPF. The OBE will be preconfigured during [Use Case 2: OBE Bootstrapping \(Manual\)](#) with the FQDN of the RA to which it submits the certificate batch request.

Assumptions

OBE has successfully completed [Use Case 2: OBE Bootstrapping \(Manual\)](#).

Process Steps

The OBE should follow the steps outlined below to request pseudonym certificates. Neither order nor fulfillment of all steps is enforced, but highly recommended.

1. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#) using the API documented in [RA - Download local policy file](#) and [RA - Download Local Certificate Chain File](#)

- a. The OBE applies all changes to its trust-store (necessary for PCA Certificate Validations) if there is an updated LCCF
 - b. The OBE applies those changes if there is an updated LPF
2. The OBE creates the request, signs it with the enrollment certificate, encrypts the signed request to the RA and sends it via LOP to the RA using the API documented in [RA - Request Pseudonym Certificate Batch Provisioning](#)
3. The LOP strips any information that could be used to determine the OBE's location and forwards it to the RA
4. The RA ensures that the certificate batch request is correct and authorized before it starts [Step 3.2: Pseudonym Certificate Generation](#)

Error Handling

1. The OBE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in errors
2. The OBE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The OBE will need to execute the certification/bootstrap process again to exit a revoked state.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SCMS-341	<div style="border: 1px solid green; padding: 2px; display: inline-block;">EE REQUIREMENT</div>	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SCMS-411	<div style="border: 1px solid green; padding: 2px; display: inline-block;">EE REQUIREMENT</div>	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
SCMS-459	<div style="border: 1px solid green; padding: 2px; display: inline-block;">CLOSED</div>	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section	RA																					

Key	Status	Summary	Description	Justification	Notes	Component/s
				updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 , SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SCMS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available, and that configuration shall be current.	Note that LPF might have the same content as the global policy file (GPF).	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SCMS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption (https://jira.campllc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campllc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SCMS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SCMS-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the location of EEs.		RA
SCMS-520	EE REQUIREMENT	Request only initial set	OBE shall make a certificate provisioning request only for the initial set of pseudonym and application certificates or when the certificate parameters change	Because top-up certificates are generated	This is out of scope as it defines OBE behavior.	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				automatically by the RA.		
SCM S-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SCM S-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-529	CLOSED	Store enrollment certificate and butterfly parameters	RA shall store enrollment certificate and butterfly parameters for each OBE for its lifetime.	so that OBE can be revoked properly. Arbitrary number based on historical trends for vehicle ownership. For example, collector vehicles that are kept on the road for	PoC will only store 3 years	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				longer than typical vehicles.		
SCMS-534	CLOSED	Certificate Batch	RA shall store certificates to be downloaded by EE in the folder provided in the ack message to the provisioning request.	Certificate batch is the basis for receiving pseudonym certificates. The use-case objective is to transfer certificate batches from RA to EE.		RA
SCMS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Communications - General Guidance		
SCMS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SCMS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Technical Policies.		
SCM S-754	EE REQUIREMENT	Sign certificate request	The EE shall sign certificate requests with its enrollment certificate.	So that RA can verify that the certificate request was not been modified in transit and to verify that the certificate request is originating from a valid EE	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information,		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.		
SCMS-776	EE REQUIREMENT	Encrypt certificate request	The EE shall encrypt the request using the RA certificate.	So that the request is shared confidentially between the EE and RA.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				error handling whenever the EE is not able to get the latest version of that file.		
SCM S-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid giving potential attackers relevant information	This is not in ASN.1 but http 404	RA
SCMS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SCMS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant information.		RA
SCMS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SCMS-987	TESTS FAILED	Error code: raWrongParameters	RA shall log "Error code: raWrongParameters", if a device sends request with wrong parameters.	To enable server side diagnostics and to avoid giving potential attackers relevant information		RA
SCMS-988	TESTS FAILED	Error code: raRetries	RA shall log "Error code: raRetries", if the EE retries within the time specified in SCMS-522 .	To enable server side diagnostics and to avoid giving potential attackers relevant information. Retry not allowed within 2 seconds.		RA
SCMS-990	TESTS FAILED	Error code: raMoreThanAllowedTries	RA shall return status code HTTP 500, if the EE violates SCMS-523 , and log "Error code: raMoreThanAllowedTries".	To avoid DoS attacks		RA
SCMS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to	RA response to EE shall follow SCMS-1397	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				the EE to avoid giving a potential attacker sensitive information.		
SCMS-1070	CLOSED	Error code: raDuplicateRequestReceived	The RA shall log "Error code: raDuplicateRequestReceived" as well as identifying information of the EE, if EE sent a duplicate request.	This error code catches duplicate requests.	Consider this for MA integration at a later stage.	RA
SCMS-1076	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of the local policy file.	As the local policy file contains security relevant configuration, it is essential to verify if a recently downloaded version of that file is coming from a trustworthy source.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1082	CLOSED	Error code: raInvalidSignature	The RA shall log "Error code: raInvalidSignature", if the EE does not sign the certificate request with its enrollment certificate or if the signature is invalid.	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unsigned request might be an indication for misbehavior.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1083	CLOSED	Error code: raRequestNotEncrypted	The RA shall log "Error code: raRequestNotEncrypted", if the EE does not encrypt the certificate request using the RA's 1609 certificate.	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unencrypted certificate request might be an indication for misbehavior.	RA
SCMS-1084	CLOSED	Error code: raInvalidCredentials	The RA shall log "Error code: raInvalidCredentials", if the EE has invalid credentials (blacklisted, expired, unauthorized)	To enable server side diagnostics and to avoid giving potential attackers relevant information	A request with invalid credentials might be an indication for misbehavior.	RA
SCMS-1085	TESTS FAILED	Error code: raUnauthorizedRequest	RA shall log "Error code: raUnauthorizedRequest", if an EE makes an unauthorized request (invalid permissions)	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unauthorized request might be an indication for misbehavior.	RA
SCMS-1086	TESTS FAILED	Error code: raMalformedRequest	RA shall log "Error code: raMalformedRequest", if an EE makes a malformed request not captured in https://jira.campllc.org/browse/SCMS-1082 , https://jira.campllc.org/browse/SCMS-1083 , https://jira.campllc.org/browse/SCMS-1084 SCMS-1085 .	To enable server side diagnostics and to avoid giving potential attackers relevant information.	A malformed request might be an indication for misbehavior.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1087	CLOSED	Error code: raMismatch	The RA shall log "Error code: raMismatch", if this RA does not service the requesting EE.	To enable server side diagnostics and to avoid giving potential attackers relevant information.	A request from an EE that is not serviced by the requested RA might be an indication for misbehavior.	RA
SCMS-1088	CLOSED	Error code: raInvalidTimeReceived	The RA shall return status code HTTP 500, if the EE has send an invalid system time, and log "Error code: raInvalidTimeReceived".	To avoid EEs using the invalid certificates		RA
SCMS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SCMS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					out enrollment certificates to the IBLM.	
SCMS-1210	EE REQUIREMENT	EE Secure Key Storing	<p>EE shall store the following keys in tamper-resistant (or equivalent) storage:</p> <ul style="list-style-type: none"> Private enrollment key Butterfly key parameters (seed + expansion function parameter) All private keys (e.g., of OBE application certificates and private keys calculated from the Butterfly key parameters) 	To avoid extraction of private keys via software-based attacks.	This is out of scope since it defines EE's behavior. It is highly recommended to protect the content encryption key by a TPM-like mechanism that offers secure boot and that protects the keys against software-based attacks. Additional details are listed in Hardware, Software and OS Security	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EEs need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate the certificate chain.	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1377	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to	<ul style="list-style-type: none"> Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				malicious actors		
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SCMS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1512	EE REQUIREMENT	Generating Butterfly Key seeds and expansion function	The EE shall generate butterfly key seeds and expansion function.	Protect privacy of data during transfer by not extracting the keys.	For OBE pseudonym certificates, OBE will generate Butterfly key parameters for the certificate signature keys and the response encryption key. For OBE identification certificates, OBE will generate Butterfly key parameters for the certificate signature keys, and	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					optionally for certificate encryption keys and response encryption keys.	
SCMS-1625	TESTS FAILED	RA-EE Certificate Request Ack Message	<p>RA-EE Certificate Request Ack Message shall contain the following information:</p> <p>Case: Certificate Provisioning Request Accept</p> <ul style="list-style-type: none"> Version Low order 8-bytes of the SHA-256 hash of the encoded "ToBeSigned" certificate request from the device Time at which the first certificate batches will be available for download (represented by IEEE 1609.2 Time32) URL of the certificate repository (common for all devices serviced by a specific RA) <p>Case: Certificate Provisioning Request Reject</p> <ul style="list-style-type: none"> HTTP 500 error code 	As the EE needs to know, when and where it can go to download certificates.		RA
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				and may change at any time.		
SCM S-2612	REVIEW	Store butterfly parameters	RA shall store butterfly parameters for each OBE for the estimated functional lifetime of the OBE.	So that the certificate pre-generation and revocation can function properly. Arbitrary number based on historical trends for vehicle ownership. For example, collector vehicles that are kept on the road for longer than typical vehicles.		RA

Table 4 Use Case 3.1 - Requirements

[62 issues](#)

Design

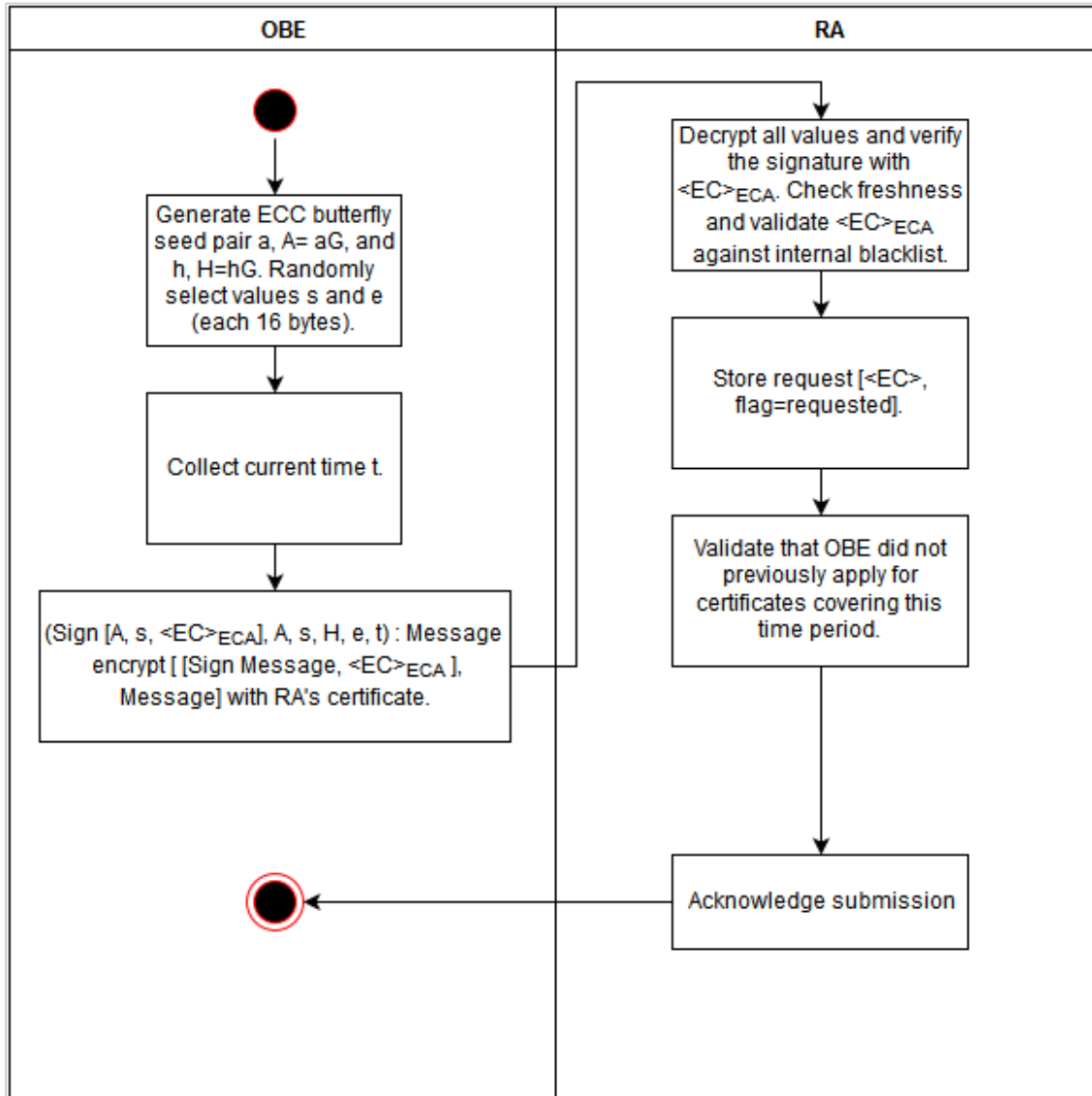


Figure 31 OBE-RA Communication

EE Request

EE initiates the certificate request message to provide the RA with critical information (key parameters, current time, etc.) necessary for certificate batch generation. New devices may experience some delay between the initial request and the time that the first certificate batches are available for download to accommodate provisioning processes such as shuffling, certificate generation, and certificate encryption. The RA will store information from the initial certificate provisioning request message and use it for ongoing certificate pre-generation until:

- The device is blacklisted at the RA due to misbehavior or malfunction

The Certificate Provisioning Request message is sent only once for each unique request and no subsequent Certificate Provisioning Request is necessary to acquire new certificate batches.

Security / Privacy

The Certificate Provisioning Request message uses signing and encryption to ensure:

- The request has not been modified in transit
- The RA can verify the message came from EE
- The request is shared confidentially between EE and RA

The EE signs the request with the Enrollment Certificate. The EE also encrypts the request using the RA certificate.

Message Contents

The EE uses the ASN.1 defined for creating the Request Certificate message. Details can be found at [RA - Request Pseudonym Certificate Batch Provisioning](#). In order for a request to be validated by the RA, the EE includes the following information in the Certificate Provisioning Request message:

- Version
- EE enrollment certificate
- Butterfly public seed / expansion function (see [SCP1: Butterfly Keys](#) for details) parameters for:
 - certificate signing key
 - response encryption key (to encrypt the created certificate towards EE)
- Current device time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)
- Requested certificate start time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)

RA Response

The RA response to the Certificate Provisioning Request message is either *accept* (indicated by a Request Acknowledgement) or *reject* (indicated by a HTTP 500). Specific error codes will be hidden from EEs in production to avoid providing useful information to malicious actors. RA logs the specific error for future investigation.

RA - EE Request Acknowledgement

The Request Acknowledge message is initiated by the RA in response to a Certificate Provisioning Request message successfully received from the EE. If the EE request is received and processed without triggering an error (invalid signature, blacklisted, etc.), the RA processes the certificate request and begins certificate pre-generation. The Request Acknowledge message provides the EE with the URL and the time where and at which the first certificates batches will be available for download.

Security / Privacy

The Request Acknowledge message use signing to ensure:

- The request has not been modified in transit
- The EE can verify the message came from the RA

The RA signs the Request Acknowledge message using the RA certificate.

Message Contents

The RA uses the ASN.1 defined for creating the Request Acknowledge message, which can be found at [RA - Request Pseudonym Certificate Batch Provisioning](#).

EE Response

If the RA provides a positive acknowledgement (*accept*) to a Certificate Provisioning Request, the EE moves forward with the certificate batch download process using the provided URL and time both given in the acknowledge message.

If the EE does not receive an acknowledgement from the RA in response to the request within defined time, EE should retry. Several conditions may necessitate the EE sending the request more than once. This may be due to:

- Request lost in transit (no TCP ack)
- RA offline, unavailable or RA network address has changed (EE must query DNS for latest RA network information)
- EE possesses an invalid RA certificate and cannot establish secure communications
- EE received HTTP-500 Error Code

The EE should not attempt to transmit the Request Certificate message without having completed the prerequisites.

ASN.1 Specification

- [ee-ra.asn](#)
- [scms-protocol.asn](#)
- [scms-base-types.asn](#)
- [scms-error.asn](#)
- [scms-policy.asn](#)
- [scms-common-errors.asn](#)
- [1609dot2-schema.asn](#)
- [1609dot2-base-types.asn](#)

Step 3.3: Initial Download of Pseudonym Certificates

Goals

The goal is to provide a reliable, secure and timely method for certified devices to download credentials, while maintaining a minimum level of privacy that is expected by the end user. The solution should prevent a certified device (that has not been revoked) from running out of credentials required for critical safety systems to operate to the greatest extent possible.

Background and Strategic Fit

The purpose of this use case is to provide a defined method that a certified OBE can use to download batches of credentials. These credentials will be used to certify the device during transmission of critical safety messages, submission of misbehavior reports, and other critical system functions. The download will include:

1. Files that include batches of certificates (each file holds certificates worth a week)
2. The .info file that includes the time when the next batch of certificates will be available for download
3. A local certificate chain file containing all PCA certificate chains required to validate the pseudonym certificates
4. The local policy file

Assumptions

1. The OBE has successfully completed [Step 3.1: Request for Pseudonym Certificates](#)
2. The RA retrieved from PCA the issued certificates, zipped, and stored them in a folder for OBE to download

Process Steps

The OBE should follow the following steps to download the initial batch of pseudonym certificates. Neither order nor fulfillment of all steps is enforced, but highly recommended.

1. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#), as before in [Step 3.1: Request for Pseudonym Certificates](#)
 - a. The OBE applies all changes to its trust-store (necessary for PCA Certificate Validations) if there is an updated LCCF
 - b. The OBE applies those changes if there is an updated LPF
2. The OBE downloads the pseudonym certificate batches using the API documented in [RA - Download Pseudonym Certificate Batch](#)
3. The OBE downloads the .info file using the API documented in [RA - Download .info File](#)

Error Handling

1. The OBE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in errors
2. The OBE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The OBE will execute the certification/bootstrap process again to exit a revoked state.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SC MS-341	EE REQUIREMENT	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SC MS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SC MS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SC MS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration	Note that LPF might have the same content as the global policy file (GPF).	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				available, and that configuration shall be current.		
SC MS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SC MS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption for the download process. (https://jira.campllc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campllc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SC MS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SC MS-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				location of EEs.		
SC MS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SC MS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-534	CLOSED	Certificate Batch	RA shall store certificates to be downloaded by EE in the folder provided in the ack message to the provisioning request.	Certificate batch is the basis for receiving pseudonym certificates. The use-case objective is to transfer certificate batches from RA to EE.		RA
SC MS-537	CLOSED	RA-to-EE encryption	The RA-to-EE communication shall be encrypted.	To avoid that an adversary is able to read EE's enrollment certificate (protect location privacy) or,	For pseudonym certificates, this counters a somewhat exotic attack: if an attacker eavesdrops all individually encrypted pseudonym certificates (encrypted by PCA to EE), and then later extracts the Butterfly keys (e.g., after the car arrived on the junk yard), the attacker is able to track the	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				in case of pseudonym certificates, that an adversary is able to read PCA-encrypted pseudonym certificates.	target vehicle in a retrofit manner assuming that attacker has access to a large database of tracking data. For other certificates, this is just an add-on security layer.	
SC MS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-543	CLOSED	Individual certificate downloads	RA shall support individual certificate batch, or certificate file, downloads by EEs.	The design allows download of individual certificate batches, or files, to avoid that an EE needs to download all certificates each time. This also allows easier resume of a download.		RA
SC MS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SC MS-547	CLOSED	Available certificate batches	The number of certificate batches, or certificate files, available for download shall be configurable (e.g. 3	This might change during the		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			years) as defined by the configuration option max_available_cert_supply in the global policy.	lifetime of the SCMS. It might even vary for different EEs.		
SC MS-548	CLOSED	X.info file	RA shall provide an .info file for download by EE.	The .info file provides information when new pseudonym certificates, or identification certificates, can be downloaded.	In order for the EE to determine the earliest time which new certificate batches will be available for download, the RA shall maintain a file in each device specific repository. This file will contain a timestamp at which the RA is predicted to update certificate batches in the device repository. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004). The file shall be named according to the following format: X.info Where X is the lower 8-bytes of the SHA-256 hash of device request in hexadecimal	RA
SC MS-549	CLOSED	Keep Certificates	The RA shall allow the EE to download certificates that have previously been downloaded, so long as the devices credentials are still valid and the certificates are not expired.	to recover from a loss of certificates at the device level (e.g., disk corruption).		RA
SC MS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside

Key	Status	Summary	Description	Justification	Notes	Component/s
				latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce Technical Policies.		Equipment (RSE)
SC MS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information , as appropriate		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.		
SC MS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to get the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-964	CLOSED	Error code: raCertFileUnavailable	RA shall return status code HTTP 500 to EE, if certificate batch is not available and log "Error code: raCertFileUnavailable.	to enable EE side		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				error handling.		
SC MS-965	EE REQUIREMENT	Error code: eeCertFileDownloadFailed	If OBE is not able to download pseudonym or identification certificate files (e.g., because there is none or it is corrupted), OBE shall implement OEM defined error handling and store the error code in OBE's error log file.	To enable EE side diagnostics .	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SC MS-967	EE REQUIREMENT	Error code: eeCertFileVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of an encrypted certificate.	To enable EE side diagnostics .	This is out of scope since it defines EE behavior. This is for a single-issue certificate that has been encrypted and digitally signed by PCA.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-969	EE REQUIREMENT	Error code: eeCertificateFileDecryptionFailed	EE shall log this error code, if EE is not able to decrypt an encrypted certificate.	To enable EE side diagnostics .	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-971	EE REQUIREMENT	Error code: eeCertificateVerificationFailed	EE shall log this error code, if EE is not able to verify a certificate.	This is to verify the issued certificate.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-973	EE REQUIREMENT	Error code: eeCertContentFalse	EE shall log this error code, if EE is not able to parse a certificate, or if the certificate has wrong content.	To enable EE side diagnostics .	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid	This is not in ASN.1 but http 404	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				giving potential attackers relevant information		
SC MS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SC MS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant information .		RA
SC MS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics .	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SC MS-982	CLOSED	X.info file update period	RA shall update the .info file at least on a weekly basis.	The .info file is updated regularly to provide timely updates to EE		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-984	EE REQUIREMENT	Error code: obelInfoFileDownloadFailed	OBE shall log this error code, if it is not able to download the .info file (e.g. because there is none or it is corrupted).	to enable EE side diagnostics .	This is out of scope since it defines EE behavior.	On-board Equipment (OBE)
SC MS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid giving a potential attacker sensitive information .	RA response to EE shall follow SCMS-1397	RA
SC MS-1076	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of the local policy file.	As the local policy file contains security relevant configuration, it is essential to verify if a recently downloaded version of that file is coming from a trustworthy source.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1090	CLOSED	Error code: raTcpErrors	RA shall return standard TCP error codes if TCP errors occur and log "Error code: raTcpErrors" and the encountered TCP error.	in order to enable client side error handling.		RA
SC MS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1201	EE REQUIREMENT	EE certificate download via HTTPS over TCP/IP	EE shall use HTTPS (TLS) over TCP/IP to download files from the SCMS.	In order to use standard internet technology .	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SC MS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					out enrollment certificates to the IBLM.	
SC MS-1214	EE REQUIREMENT	OBE downloads .info file	OBE shall download the .info file each time OBE downloaded pseudonym or identification certificates.	EE requires the information to learn when certificates will be available for download.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SC MS-1215	EE REQUIREMENT	EE contacts RA for certificate download	EE shall try to download certificates any time after the time provided by the time-stamp in the .info file that has been recovered last time EE tried to download, or downloaded, certificates.	To avoid wasting resources by trying to download certificates before they are available.	This is out of scope since it defines EE behavior. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004).	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1279	EE REQUIREMENT	Error code: eeCertificateDecryptionFailed	EE shall log this error if certificate decryption failed at EE.	To allow error reaction and investigation.	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1280	EE REQUIREMENT	Error code: eeCertificateNotReadable	EE shall log this error if any certificate is not readable.	To enable error reaction and investigation.		On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1282	EE REQUIREMENT	Error code: eeDecompressionError	EE shall log "Error code: eeDecompressionError", if it is not able to decompress the received certificate.	To allow error reaction and investigation.	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1303	EE REQUIREMENT	Verification of certificate validity	EE shall verify the validity of a received certificate against IEEE 1609.2-v3-D12, clause 5.1 and 5.3.	To verify if the certificate is issued by a trustworthy source and therefore messages signed by this certificate can be trusted.	This is for testing that SCMS issued valid and proper certificates. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate.	On-board Equipment (OBE), RA, Roadside

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificate chain.	This is out of scope since it defines EE behavior	Equipment (RSE)
SCMS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EES need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate the certificate chain.	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1377	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful	<ul style="list-style-type: none"> Standard TCP (https://jira.campplc.org/browse/SCMS-1090) and TLS (https://jira.campplc.org/browse/SCMS-977) errors shall be reported to EEs 	CRL Store, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				information from being provided to malicious actors	<ul style="list-style-type: none"> All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	
SC MS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SC MS-1420	SCMS POC OUT OF SCOPE	Keep track of which authorization (pseudonym, ID, application) certificates are downloaded	The RA shall keep track of how far into the future each device has downloaded its certificates.	This allows revoked devices to be removed from the CRL (or deprioritized for inclusion on the CRL) once they exhaust the set of certificates that they have downloaded.	This is an optimization for CRL handling and therefore out of scope for PoC implementation.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-142 1	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-145 4	CLOSED	Pseudonym certificate batch filename	RA shall name pseudonym certificate batch files according to the following format: <ul style="list-style-type: none"> • X_Y.zip • Where X is the lower 8-bytes of the SHA-256 hash of device request in hexadecimal in uppercase • Where Y is the i-value in hexadecimal in uppercase • Where the extension is .zip in lowercase 	File names must be predefined to allow OBEs to make valid download requests.	Example file name: 2AFC55B22CFDBE3E_3C.zip	RA
SC MS-145 6	CLOSED	Certificate file content	RA shall organize individual certificates contained within the certificate batch according to the following format: <ul style="list-style-type: none"> • X_Y • Where X is the i-value in hexadecimal in uppercase • Where Y is a sequence of "j" values from j = 0 to j = j_max-1 in hexadecimal in uppercase • Where there is no extension 	File content must be predefined to allow EEs to process the contents.	For example: <ul style="list-style-type: none"> • 0_0 • 0_1 • ... • 0_<j_max-1> 	RA
SC MS-163 9	EE REQUIREMENT	Download certificate batches	OBE shall not attempt to download certificate batches for i-value periods more than max_available_cert_supply in the future	To reduce resource usage by not attempting to download	<ul style="list-style-type: none"> • This is out of scope as it defines EE behavior. • This is the OBE counterpart of https://jira.campilc.org/browse/SCMS-547 	

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificate batches that do not exist.		
SC MS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Table 5 Use Case 3.3 - Requirements

[67 issues](#)

Step 3.5: Top-off Pseudonym Certificates

Goals

The goal is to provide a reliable, secure and timely method for certified devices to download credentials. The solution should prevent a certified device (that has not been revoked) from running out of credentials required for critical safety systems to operate to the greatest extent possible.

Background and Strategic Fit

The purpose of this use-case is to provide a defined method that a certified OBE can use to download new batches of credentials. These credentials will be used to certify the device during transmission of critical safety messages, submission of misbehavior reports, and other critical system functions. The download will include:

1. Files that include batches of certificates (each file holds certificates worth a week)
2. The .info file that includes the time when the next batch of certificates will be available for download
3. A local certificate chain file containing all PCA certificate chains required to validate the pseudonym certificates
4. The local policy file

The step at hand is to top-up pseudonym certificates. It is similar to [Step 3.3: Initial Download of Pseudonym Certificates](#) and differences are documented in this section. Also, see [Step 3.4: Schedule Generation of Subsequent Batch of Pseudonym Certificates](#) for full details of the process to schedule certificate pre-generation.

Assumptions

- OBE has successfully completed [Step 3.1: Request for Pseudonym Certificates](#)
- OBE has successfully completed [Step 3.3: Initial Download of Pseudonym Certificates](#)
- RA retrieved the issued certificates from PCA, zipped, and stored them in a folder on RA for OBE to download

Process Steps

The OBE should follow the following steps to download the initial batch of pseudonym certificates. Neither order nor fulfillment of all steps is enforced, but highly recommended.

1. The OBE checks that, and if necessary waits until, the current time matches or is after the timestamp given in the .info file
2. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#), as before in [Step 3.1: Request for Pseudonym Certificates](#)
 - a. If there is an updated LCCF, the OBE applies all changes to its trust-store (necessary for PCA Certificate Validations)
 - b. If there is an updated LPF, the OBE applies those changes
3. The OBE downloads pseudonym certificate batches
4. The OBE downloads .info file using the API documented in [RA - Download .info File](#)

Error Handling

1. The OBE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in critical errors

2. The OBE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The OBE will need to execute the certification/bootstrap process again to exit a revoked state.
3. The OBE may terminate the certificate batch download process if sufficient storage is not available for subsequent batches

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SC MS-341	EE REQUIREMENT	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SC MS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SC MS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SC MS-509	CLOSED	Stop pre-generating pseudonym and OBE identification certificates for revoked device	RA shall stop pre-generating pseudonym and OBE identification certificates for a device that has been revoked by the MA, i.e., for a device that appears on RA's internal blacklist.	so that computing resources are not wasted by		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				generating certificates for revoked devices		
SCMS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available, and that configuration shall be current.	Note that LPF might have the same content as the global policy file (GPF).	RA
SCMS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SCMS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption (https://jira.campllc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campllc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SCMS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	
SC MS-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the location of EEs.		RA
SC MS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SC MS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-534	CLOSED	Certificate Batch	RA shall store certificates to be downloaded by EE in the folder provided in the ack message to the provisioning request.	Certificate batch is the basis for receiving pseudonym certificates. The use-case		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				objective is to transfer certificate batches from RA to EE.		
SC MS-537	CLOSED	RA-to-EE encryption	The RA-to-EE communication shall be encrypted.	To avoid that an adversary is able to read EE's enrollment certificate (protect location privacy) or, in case of pseudonym certificates, that an adversary is able to read PCA-encrypted pseudonym certificates.	For pseudonym certificates, this counters a somewhat exotic attack: if an attacker eavesdrops all individually encrypted pseudonym certificates (encrypted by PCA to EE), and then later extracts the Butterfly keys (e.g., after the car arrived on the junk yard), the attacker is able to track the target vehicle in a retrofit manner assuming that attacker has access to a large database of tracking data. For other certificates, this is just an add-on security layer.	RA
SC MS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance		
SC MS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-543	CLOSED	Individual certificate downloads	RA shall support individual certificate batch, or certificate file, downloads by EEs.	The design allows download of individual certificate batches, or files, to avoid that an EE needs to download all certificates		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				each time. This also allows easier resume of a download.		
SC MS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SC MS-547	CLOSED	Available certificate batches	The number of certificate batches, or certificate files, available for download shall be configurable (e.g. 3 years) as defined by the configuration option max_available_cert_supply in the global policy.	This might change during the lifetime of the SCMS. It might even vary for different EEs.		RA
SC MS-548	CLOSED	X.info file	RA shall provide an .info file for download by EE.	The .info file provides information when new pseudonym certificates, or identification certificates, can be downloaded.	In order for the EE to determine the earliest time which new certificate batches will be available for download, the RA shall maintain a file in each device specific repository. This file will contain a timestamp at which the RA is predicted to update certificate batches in the device repository. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004). The file shall be named according to the following format: X.info Where X is the lower 8-bytes of	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					the SHA-256 hash of device request in hexadecimal	
SC MS-549	CLOSED	Keep Certificates	The RA shall allow the EE to download certificates that have previously been downloaded, so long as the devices credentials are still valid and the certificates are not expired.	to recover from a loss of certificates at the device level (e.g., disk corruption).		RA
SC MS-576	CLOSED	Update .info file	The RA shall update .info files for all EEs even if no new certificate batches are created.	The EE uses the .info file to determine when the earliest the next download is allowed to happen.	Timestamp in .info file is dynamically calculated based on system load. PoC scope will be to update .info file for non-revoked EEs only.	RA
SC MS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Technical Policies.		
SC MS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information , as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				will include a newer LCCF if available.		
SC MS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to get the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside

Key	Status	Summary	Description	Justification	Notes	Component/s
				correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.		Equipment (RSE)
SC MS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-964	CLOSED	Error code: raCertFileUnavailable	RA shall return status code HTTP 500 to EE, if certificate batch is not available and log "Error code: raCertFileUnavailable.	to enable EE side error handling.		RA
SC MS-965	EE REQUIREMENT	Error code: eeCertFileDownloadFailed	If OBE is not able to download pseudonym or identification certificate files (e.g., because there is none or it is corrupted), OBE shall implement OEM defined error handling and store the error code in OBE's error log file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SC MS-967	EE REQUIREMENT	Error code: eeCertFileVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of an encrypted certificate.	To enable EE side diagnostics.	This is out of scope since it defines EE behavior. This is for a single-issue certificate that has been encrypted and digitally signed by PCA.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-969	EE REQUIREMENT	Error code: eeCertificateFileDecryptionFailed	EE shall log this error code, if EE is not able to decrypt an encrypted certificate.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-971	EE REQUIREMENT	Error code: eeCertificateVerificationFailed	EE shall log this error code, if EE is not able to verify a certificate.	This is to verify the issued certificate.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-973	EE REQUIREMENT	Error code: eeCertContentFalse	EE shall log this error code, if EE is not able to parse a certificate, or if the certificate has wrong content.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid giving potential attackers relevant information	This is not in ASN.1 but http 404	RA
SC MS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SC MS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				giving potential attackers relevant information.		
SC MS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SC MS-982	CLOSED	X.info file update period	RA shall update the .info file at least on a weekly basis.	The .info file is updated regularly to provide timely updates to EE		RA
SC MS-984	EE REQUIREMENT	Error code: obeInfoFileDownloadFailed	OBE shall log this error code, if it is not able to download the .info file (e.g. because there is none or it is corrupted).	to enable EE side diagnostics.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE)
SC MS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid	RA response to EE shall follow SCMS-1397	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				giving a potential attacker sensitive information.		
SC MS-1070	CLOSED	Error code: raDuplicateRequestReceived	The RA shall log "Error code: raDuplicateRequestReceived" as well as identifying information of the EE, if EE sent a duplicate request.	This error code catches duplicate requests.	Consider this for MA integration at a later stage.	RA
SC MS-1076	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of the local policy file.	As the local policy file contains security relevant configuration, it is essential to verify if a recently downloaded version of that file is coming from a trustworthy source.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1090	CLOSED	Error code: raTcpErrors	RA shall return standard TCP error codes if TCP errors occur and log "Error code: raTcpErrors" and the encountered TCP error.	in order to enable client side error handling.		RA
SC MS-1164	EE REQUIREMENT	OBE next download timing	OBE shall use the stored .info file to schedule the next download attempt.	The .info file contains the timestamp when the next batch	This is out of scope since it defines EE's behavior. <ul style="list-style-type: none"> If no pseudonym certificates are available on the OBE for the current i_period (week), the OBE is allowed to make a 	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				of certificates (pseudonym or identification) will be available for download. This timestamp is the earliest the OBE is allowed to connect to the RA for the next download. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004).	<p>download attempt at any time.</p> <ul style="list-style-type: none"> If no pseudonym certificates are available on the OBE for the next <i>i</i>_period (week), the OBE is allowed to make a download attempt at any time. If no identification certificate is available on the OBE for the current or next time period, the OBE is allowed to make a download attempt at any time. 	
SC MS-116 Z	EE REQUIREMENT	Expired Certificate Batches	The OBE shall only download pseudonym certificate batches for the current and future i_period .	Only download certificates that are valid at the current time or in the future.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Certificates that are already expired should not be downloaded.		
SC MS-1168	EE REQUIREMENT	OBE_pseudonym_certificate_duplicate_downloads	OBE shall not download pseudonym certificate batches that are already verified and stored on the device.	During top-up downloads, the OBE shall only download pseudonym certificate batches that are not currently verified and stored on the device. This is to prevent repeated downloads of the same content.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SC MS-1171	EE REQUIREMENT	EE revoked	EEs that are revoked shall not attempt to download LCCF, LPF, pseudonym certificates, identification certificates or file misbehavior reports. Exceptions to this are: <ul style="list-style-type: none"> • EE is unable to determine its revocation status • EE has no pseudonym or identification certificates available in local storage • EE is attempting to perform a re-enrollment operation 	To avoid unnecessary load at the RA.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-118 9	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-120 1	EE REQUIREMENT	EE certificate download via HTTPS over TCP/IP	EE shall use HTTPS (TLS) over TCP/IP to download files from the SCMS.	In order to use standard internet technology	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-120 3	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SC MS-120 4	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	RA
SC MS-121 4	EE REQUIREMENT	OBE downloads .info file	OBE shall download the .info file each time OBE downloaded pseudonym or identification certificates.	EE requires the information	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				to learn when certificates will be available for download.		
SC MS-1215	EE REQUIREMENT	EE contacts RA for certificate download	EE shall try to download certificates any time after the time provided by the time-stamp in the .info file that has been recovered last time EE tried to download, or downloaded, certificates.	To avoid wasting resources by trying to download certificates before they are available.	This is out of scope since it defines EE behavior. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004).	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1270	EE REQUIREMENT	Network connection	EES shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1282	EE REQUIREMENT	Error code: eeDecompressionError	EE shall log "Error code: eeDecompressionError", if it is not able to decompress the received certificate.	To allow error reaction and investigation.	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
						Equipment (RSE)
SC MS-135 3	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Roadside Equipment (RSE)
SC MS-135 6	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EEs need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate the certificate chain.	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-137 7	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download	Whitelist defined in SCMS-1371	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificates or files.		
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to malicious actors	<ul style="list-style-type: none"> Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SCMS-1420	SCMS POC OUT OF SCOPE	Keep track of which authorization (pseudonym, ID, application) certificates are downloaded	The RA shall keep track of how far into the future each device has downloaded its certificates.	This allows revoked devices to be removed from the CRL (or deprioritized for inclusion on the	This is an optimization for CRL handling and therefore out of scope for PoC implementation.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				CRL) once they exhaust the set of certificates that they have downloaded.		
SC MS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1454	CLOSED	Pseudonym certificate batch filename	RA shall name pseudonym certificate batch files according to the following format: <ul style="list-style-type: none"> • X_Y.zip • Where X is the lower 8-bytes of the SHA-256 hash of device request in hexadecimal in uppercase • Where Y is the i-value in hexadecimal in uppercase • Where the extension is .zip in lowercase 	File names must be predefined to allow OBEs to make valid download requests.	Example file name: 2AFC55B22CFDBE3E_3C.zip	RA
SC MS-1456	CLOSED	Certificate file content	RA shall organize individual certificates contained within the certificate batch according to the following format: <ul style="list-style-type: none"> • X_Y • Where X is the i-value in hexadecimal in uppercase • Where Y is a sequence of "j" values from j = 0 to j = j_max-1 in hexadecimal in uppercase 	File content must be predefined to allow EEs to process the contents.	For example: <ul style="list-style-type: none"> • 0_0 • 0_1 • ... • 0_<j_max-1> 	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			<ul style="list-style-type: none"> Where there is no extension 			
SCMS-1639	EE REQUIREMENT	Download certificate batches	OBE shall not attempt to download certificate batches for i-value periods more than max_available_cert_supply in the future	To reduce resource usage by not attempting to download certificate batches that do not exist.	<ul style="list-style-type: none"> This is out of scope as it defines EE behavior. This is the OBE counterpart of https://jira.campllc.org/browse/SCMS-547 	
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EES shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 6 Use Case 3.5 - Requirements

[72 issues](#)

Design Notes

- See [Step 3.3: Initial Download of Pseudonym Certificates](#) for full details of the batch download process. Differences are documented in this section.
- From the SCMS point of view, the basic process for "top-up" certificate downloads is the same as that used for initial provisioning as detailed in [Step 3.3: Initial Download of Pseudonym Certificates](#). However, this is an incremental download, not a full download of all available certificate files. The number of files downloaded shall be factored in system sizing requirements.
- From the OBE's point of view, the process is slightly different from the process for initial provisioning
- See [Step 3.4: Schedule Generation of Subsequent Batch of Pseudonym Certificates](#) for full details of the process to schedule certificate pre-generation
- The RA will record the last time an OBE established a connection. This last connection time will be used to stop pre-generating pseudonym certificates if there is no activity for a period of time.
- The RA will automatically resume pre-generating pseudonym certificates when an OBE reestablishes a connection. The new certificates will be available for download at the time specified in the .info file.

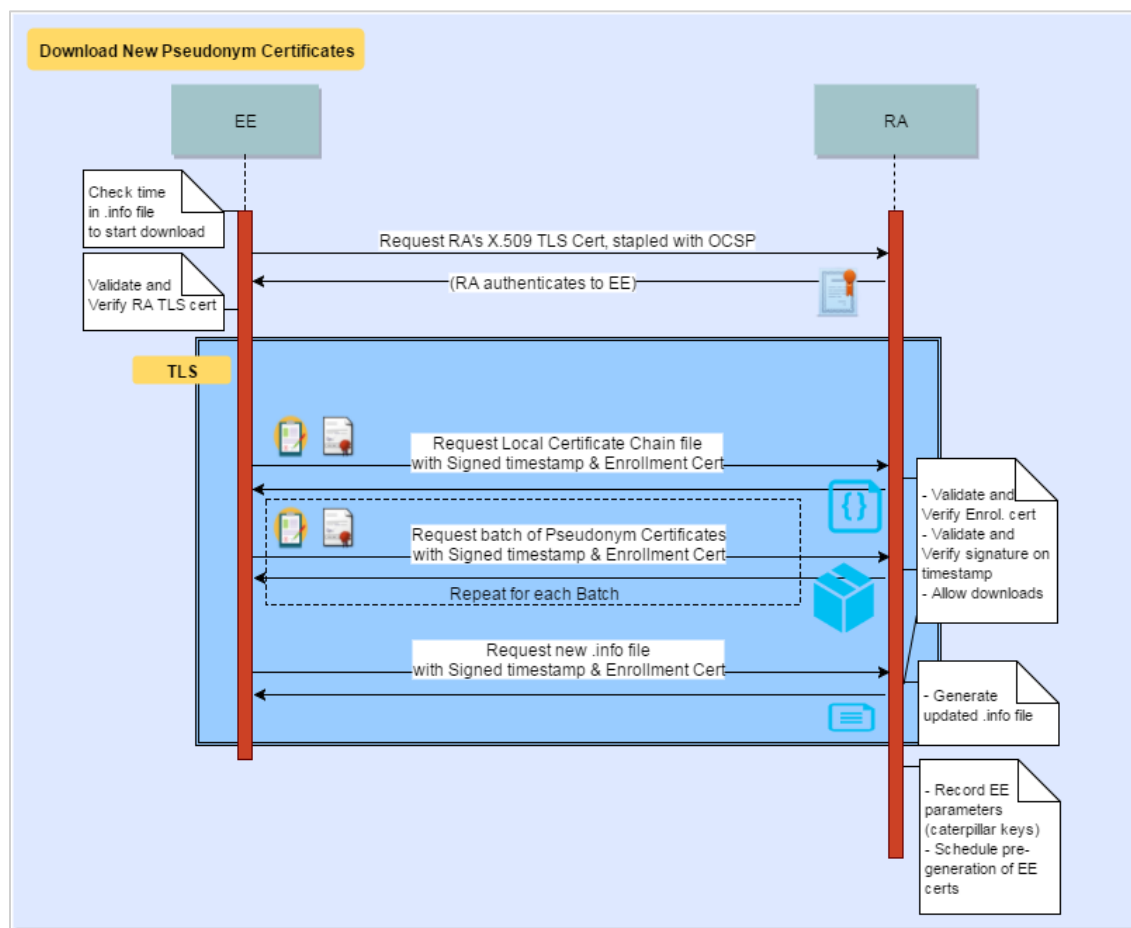


Figure 32 Download New Pseudonym Certificates

Not Doing

- Stopping of pre-generation of pseudonym certificates if an OBE has not contacted the RA for a period of time

Use Case 5: Misbehavior Reporting

Misbehavior Reporting will be integrated with the ongoing "Misbehavior Authority Integration" sub project as SCMS PoC release 2.0. Until then misbehavior reports will not be received and the previous misbehavior report format as described further down can and will change.

Goals

- Maintain the trust in the system
- Identify and remove bad actors

Background and Strategic Fit

EES send misbehavior reports to the MA via the RA. The format of a misbehavior report is not defined yet but a report potentially includes reported BSMs as well as the reporter's pseudonym certificate and the reporter's signature. Reports may include random BSMs (casual report), suspicious BSMs, and alert-related BSMs. The report is encrypted by the EE for the MA. Note: The EEs' misbehavior detection algorithms (also called local misbehavior detection) are not defined yet.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SCMS-341	<div style="border: 1px solid green; padding: 2px;">EE REQUIREMENT</div>	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7251</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7251</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251																									
SCMS-411	<div style="border: 1px solid green; padding: 2px;">EE REQUIREMENT</div>	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Road-side Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SCMS-515	CLOSED	RA requires EE	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
		authentication		can send requests, download certificates or files.	authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	
SCMS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SCMS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-684	EE REQUIREMENT	Encryption	EE shall encrypt misbehavior reports with the Misbehavior Authority's public key before sending.	To avoid unauthorized parties	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE),

Key	Status	Summary	Description	Justificati on	Notes	Compone nt/s
				getting access to the misbehaving report.		Road-side Equipment (RSE)
SCMS-685	CLOSED	Privacy	LOP shall remove all information in the IP layer that can be used to identify the location of the OBE before forwarding the misbehavior report to the RA.	to protect the identity of the sending OBE.	This does not apply to misbehavior reports send by RSEs.	LOP, RA
SCMS-686	TESTS PASSED	Shuffle	RA shall shuffle misbehavior reports before sending them to MA.	Shuffling ensures MA cannot be sure that two misbehaviors or reports coming at the same time are from same OBE.	Amount of shuffle and/or maximum delay decided by SCMS manager.	RA
SCMS-765	TESTS PASSED	Shuffle Threshold	RA shall use a shuffle threshold of 10,000 misbehavior reports or one day whichever is reached first.	Shuffling ensures MA cannot be sure that two misbehaviors or reports coming at the same time are from same RSE.		RA
SCMS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificate		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				<p>s without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information, as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer</p>		

Key	Status	Summary	Description	Justification	Notes	Component/s
				LCCF if available.		
SCMS-1171	EE REQUIREMENT	EE revoked	EEs that are revoked shall not attempt to download LCCF, LPF, pseudonym certificates, identification certificates or file misbehavior reports. Exceptions to this are: <ul style="list-style-type: none"> • EE is unable to determine its revocation status • EE has no pseudonym or identification certificates available in local storage • EE is attempting to perform a re-enrollment operation 	To avoid unnecessary load at the RA.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Road-side Equipment (RSE)
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being	<ul style="list-style-type: none"> • Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs • All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				provided to malicious actors		
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS component	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				ts are not guaranteed to be static and may change at any time.		Equipment (RSE)

Table 7 Use Case 5 - Requirements

[22 issues](#)

Design

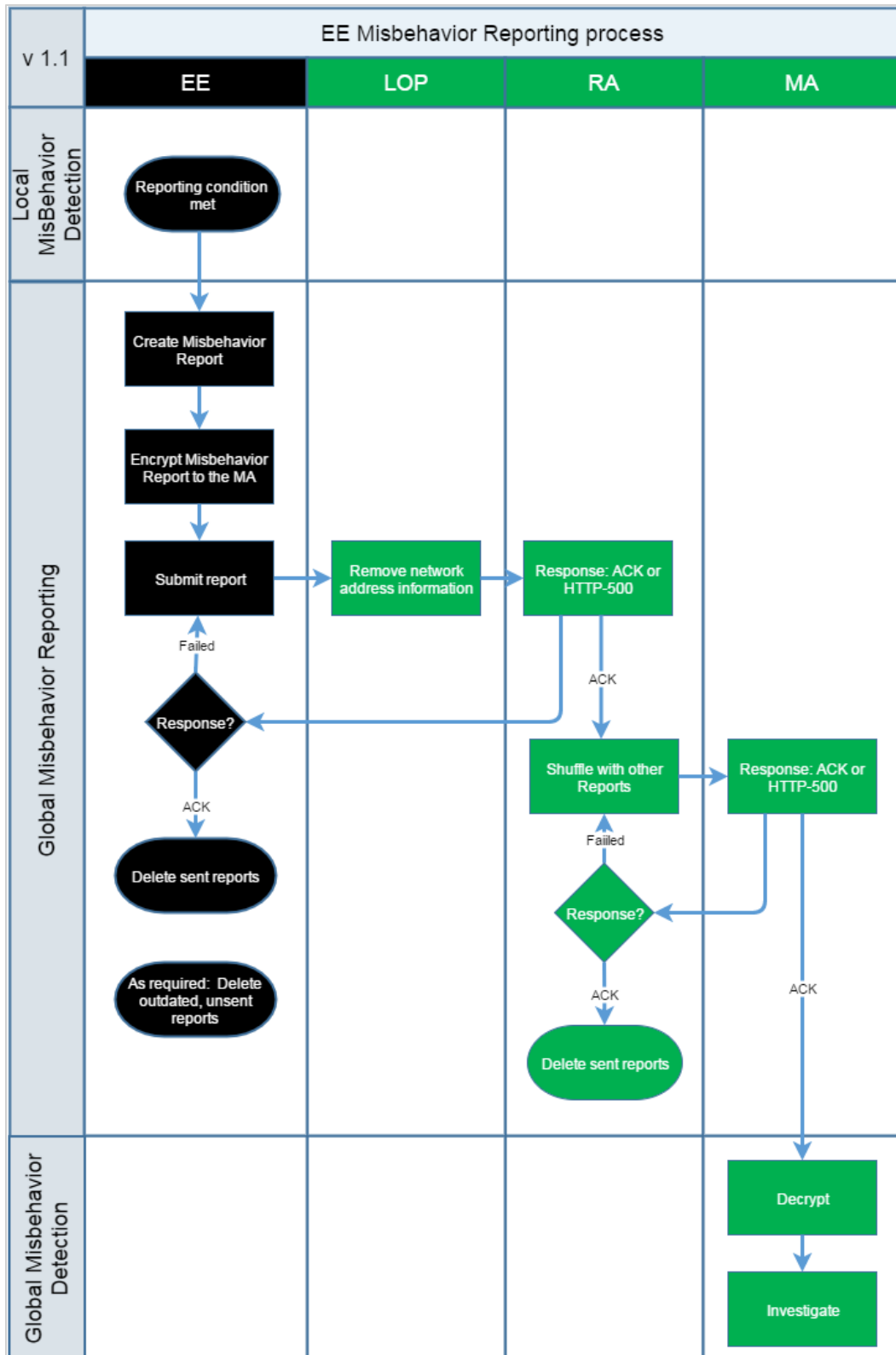


Figure 33 EE Misbehavior Reporting Process

The following steps are executed:

- Step 1: Reporting condition met
- Step 2: EE creates a misbehavior report and signs with a pseudonym certificate
- Step 3: EE encrypts report to the MA
- Step 4: EE submits it to the RA
 - Step 4.1: The LOP removes any identifiers from the encrypted misbehavior report (e.g., MAC address and IP address) and forwards the encrypted report to RA
 - Step 4.2: RA shuffles misbehavior reports and sends to MA individually. Shuffle threshold is 10,000 misbehavior reports or one day whichever is reached first. (Note: This shuffle threshold is for POC only, needs to be re-evaluated by SCMS manager for production)
- Step 5: Unsent misbehavior reports older than one week may be deleted by the EE if insufficient memory exists

ASN.1 Specification

ASN.1 interface specifications for misbehavior reports will be finalized with the to-be-awarded "Misbehavior Authority Integration" sub project. Until then the interface given is to be handled as draft.

- [ee-ma.asn](#)

Use Case 6: CRL Download

Goals

The goal is to provide the CRL file from the CRL Store (a component of the MA) to the EE when requested.

Background and Strategic Fit

The EE must be aware of revoked entities.

Assumptions

- One or more CRLs have been generated, signed by the CRL Generator, put into a CRL file, and has been made available to the CRL Store
- The CRL Store is able to validate cryptographically the signature on the CRL file prior to making it available for download
The EE is able to download the CRL by issuing a CRL HTTP get request to the CRL Store.
- The CRL Store will not authenticate the EE, i.e., CRL Store will not require that EE sends its enrollment certificate for authentication purposes
- OBE has successfully executed [Use Case 2: OBE Bootstrapping \(Manual\)](#)

Process Steps

1. OBE downloads the CRL using the API documented in [MA - Download CRL](#)

Requirements

Key	Status	Summary	Description	Justification	Notes	Components																					
SCM S-335	EE REQUIREMENT	EE issues a CRL Request	An EE shall issue a HTTP get to the CRL Store to obtain the latest CRL.	EE needs to be provided with current CRL so that the EE can be informed of revoked components.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)																					
SCM S-340	CLOSED	CRL availability	The CRL Store shall provide a CRL for download at any given time.	To ensure that an EE can always download a CRL.		CRL Store																					
SCM S-341	EE REQUIREMENT	EE TLS Cipher Suite	<p>The EE shall support at least the following TLS cipher suites for all communications to SCMS components:</p> <table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xA</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7251</td> </tr> <tr> <td>0xC0,0xA</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7251</td> </tr> </tbody> </table>	IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xA	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251	0xC0,0xA	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
IANA Value	Description	Reference																									
0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																									
0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																									
0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																									
0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																									
0xC0,0xA	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251																									
0xC0,0xA	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251																									
SCM S-342	EE REQUIREMENT	CRL Store Authentication	The EE shall authenticate the CRL Store through usual SSL/TLS means.	This will provide a level of trust for the CRL Store. An impostor	The CRL Store does not authenticate EE before download	On-board Equipment (OBE), Road-side Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s	
					CRL Store might distribute only old CRLs (which would be valid).	of the CRL starts, i.e., EE does not authenticate by using its enrollment certificate to CRL Store but EE can download the CRL without authentication to CRL Store. This is out of scope as it defines EE behavior.	
SCM S-786	CLOSED	CRL download	CRLG shall provide CRL via CRL store.	so that EEs can check revocation status	CRL entries may be dependent on the type of certificate	CRLG	
SCM S-991	EE REQUIREMENT	Error code: eeCRLStoreAuthenticationFailed	EE shall log "Error code: eeCRLStoreAuthenticationFailed", if it cannot authenticate the CRL Store.	EE cannot authenticate the CRL Store.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)	
SCM S-994	EE REQUIREMENT	Error code: eeCRLDownloadFailed	EE shall log "Error code: eeCRLDownloadFailed", if EE is not able to download the CRL file.	EE cannot download CRL file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)	

Key	Status	Summary	Description	Justification	Notes	Component/s
SCM S-995	EE REQUIREMENT	Error code: eeCRLVerificationFailed	EE shall log "Error code: eeCRLVerificationFailed", if verification of the CRL signature fails.	In order to enable client side error handling.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-1264	CLOSED	CRL Store download resume	CRL Store shall support byte-wise resume of CRLs by EE.	This will improve reliability of the download process and reduce communication cost.		CRL Store
SCM S-1270	EE REQUIREMENT	Network connection	EES shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-2523	EE REQUIREMENT	EE TLS version	The EE shall at minimum support TLS version 1.2 as defined in RFC5246 for all communications to SCMS components.	To avoid known security issues in older versions of the protocol.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 8 Use Case 6 - Requirements

[12 issues](#)

Use Case 8: OBE Pseudonym Certificate Revocation

OBE Revocation will be integrated with the ongoing "Misbehavior Authority Integration" sub project as SCMS PoC release 2.0. Until then every reported pseudonym certificate leads automatically to a revocation of all pseudonym certificates belonging to this OBE for testing purposes.

Goals

Perform misbehavior investigation and eventually revocation of OBEs.

Step 8.4: OBE CRL Check

Goals

The OBE needs to perform several computational steps to check whether a received Basic Safety Message (BSM) has been sent by a revoked EE. This document lists the corresponding requirements.

Assumptions

The OBE received a CRL as defined in [Use Case 6: CRL Download](#).

Process Steps

1. The OBE expands the CRL and calculates the linkage values for the current i-period based on the CRL entries (linkage seeds) of the CRL pseudonym certificate section
2. Whenever the OBE receives a new, unknown pseudonym certificate, it checks whether the linkage value of that unknown certificate is listed in the OBE's expanded CRL (from Step 1)
 - a. If yes, then the OBE discards the received certificate
 - b. Otherwise, the OBE accepts the received certificate as verified
3. Whenever the OBE receives a new, unknown OBE identification certificate, the OBE will calculate the certificate digest of that unknown certificate and will check whether the CRL lists it
 - a. If yes, then the OBE discards the received certificate
 - b. Otherwise, the OBE accepts the received certificate as verified
4. Before the end of each i-period, the OBE will:
 - a. Update its expanded CRL and calculate the linkage value for the next i-period
 - b. Remove entries from the expanded CRL that belong to revoked devices that ran out of certificates, if a CRL entry indicated that the revoked device does not have any more valid certificates. Note that the OBE may not immediately remove such entries, but add a safety buffer.
5. If the OBE recognizes itself on the CRL, the OBE will stop sending over-the-air DSRC messages related to the indicated PSID/SSP. This also applies if the OBE recognizes that the [Enrollment CA](#) that issued the OBE's enrollment certificate, the [Pseudonym CA](#) that issued the OBE's certificates, any [Intermediate CA](#) that is in the chain between its ECA or PCA up to the Root CA, or the [Root CA](#) itself has been revoked.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
-----	--------	---------	-------------	---------------	-------	-------------

Table 9 Use Case 8.4 - Requirements

[No issues found](#)

Use Case 11: Backend Management

Goals

- The goal of backend management is the addition and removal of SCMS components
- The provisioning and initial setup requirements for all backend components is defined in [Use Case 1: SCMS Component Setup](#)

Background and Strategic Fit

As the SCMS system evolves, it is necessary that SCMS components can be added and removed.

This includes Root CAs. For the PoC, there will be only one Root CA. To manage Root CAs, (e.g., to add and remove them) the SCMS will employ a multi-Elector system. In this scheme, there are a number of electors. These entities are trust anchors but also vote to manage Root CAs, i.e., to remove or add a new Root CA. The SCMS Manager coordinates the electors. An operation on a Root CA (addition or revocation) will require a message signed by some given number of electors. The exact number of electors needed to perform addition or revocation is a fixed quorum m . The public keys of the electors will be installed into the trust stores of every SCMS component, including the OBEs. In the PoC, electors will be implemented to be manual processes, and the Root Management messages signed by electors will be generated by manual means for testing the management of the Roots CAs.

Assumptions

- SCMS components need to be added and revoked but not removed and rolled-over
- More requirements specific to each operation and component will occur in the subsections

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-406	CLOSED	TLS Cipher Suite	All TLS communication between SCMS components shall employ a cipher suite that uses cryptographic mechanisms and key lengths that are at least as strong as the following cipher suite for each individual cryptographic mechanism of the suite: TLS_ECDHE_ECDSA_WITH_AES_128_CM (as defined in RFC7251)	Forward security, ECC with defined security level as minimum, or better.	Refer to NIST recommendations to evaluate whether a selected cryptographic mechanism is stronger than the defined minimum security level. A proper starting point is "NIST, Recommendation for Key Management, Special Publication 800-57 Part 1 Rev. 3, 07/2012."	CRL Store, CRLG, DCM, ECA, IBLM, LA, MA, PCA, PG, RA
SCMS-407	IMPLEMENTED	TLS Mutual Authentication	Communication transactions between SCMS components shall be client-server mutually authenticated and encrypted with TLS (except EE-RA).	Authentication between servers with mutual TLS is an additional security layer.	For PoC, TLS shall be used for inter component communication with mutual authentication, and an OCSP service at the technical component of the SCMS manager shall be deployed for revocation. See https://jira.campllc.org/browse/SCMS-406 for information about the cipher suite, SCMS-1016 for information about OCSP, and https://jira.campllc.org/browse/SCMS-938 for TLS certificate management.	CRL Store, CRLG, DCM, ECA, IBLM, LA, MA, PCA, PG, RA
SCMS-1017	SCMS POC OUT OF SCOPE	Robustness against catastrophic failure of the components	SCMS components shall be robust against catastrophic failure.	The SCMS must be robust enough to handle trials, and to build the basis of the robustness of the eventual production system.	For PoC only minimal robustness is required. This requirement is for the production system.	
SCMS-1018	SCMS POC OUT OF SCOPE	Data encryption in geographically diverse locations	Databases should be encrypted and authenticated before sent to geographically diverse locations.	so that no unauthorized entity can gain access to the data.	For PoC only minimal robustness is required. This requirement is for the production system.	

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1019	SCMS POC OUT OF SCOP E	HSM replication	HSMs holding private keys shall be replicated and stored securely, to be able to recover encrypted and authenticated backups, as well as any operational secrets.	so that backups can be used for recovery when necessary.	For PoC only minimal robustness is required. This requirement is for the production system.	
SCMS-1020	SCMS POC OUT OF SCOP E	Replicated HSM storage	Replicated HSMs shall be stored securely, with protections at least at the same level as are provided to the production system.	to avoid security breaches via replicated HSMs.	For PoC only minimal robustness is required. This requirement is for the production system.	
SCMS-1021	SCMS POC OUT OF SCOP E	Hot standby	Hot standbys in geographically diverse locations shall be provisioned, and cold standbys similarly created to be able to accept replicated HSMs and reconstruct production materials from backup.	so that a failing SCMS component can be replaced immediately and without operational interruption of the overall system.	For PoC only minimal robustness is required. This requirement is for the production system.	
SCMS-1022	SCMS POC OUT OF SCOP E	Redundancy	The system hardware components, such as servers, switches, UPSs, etc. shall be deployed redundantly	so that failure of a single such component does not take the system offline; even more redundancy can be considered where multiple such components can fail and the system remains operational.	For PoC only minimal robustness is required. This requirement is for the production system. PoC system will have some sort of redundancy - compare the PoC hardware design.	

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1023	CLOSED	Root CA Trust Store	The SCMS Component shall be provisioned with the self-signed SCMS certificates of the Root CAs.	Every SCMS component will need to manage Root CA update automatically. To authenticate the Root CA management messages, the Root CA must be trusted, and therefore require that their Certificates be in the SCMS component trust stores.		CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1314	MANUAL PROCESS	SCMS component certificate types (implicit vs. explicit)	The SCMS component shall have a certificate of explicit type.	Implicit: OBE Enrollment, RSE Enrollment, Pseudonym, Application, Identification Explicit (Self Signed): RootCA, Elector Explicit: Everything else PCA, ICA, Root CA, and elector certificates need to be of explicit type in order to	Details discussed in certificate types	CRL Store, CRLG, DCM, IBLM, ICA, LA, PCA, PG, RA, RCA

Key	Status	Summary	Description	Justification	Notes	Component/s
				support P2P distribution. All the EE certificates are of implicit type to save storage space and over-the-air bytes, and all the SCMS Component certificates are of explicit type.		
SCMS-1315	MANUAL PROCESS	Only 1 certificate valid at a time	Each SCMS component shall have only 1 valid and in-use certificate at a time.	There are no privacy concerns for SCMS components that would justify more than one certificate valid at a given time. At the same time, it is desirable to keep complexity low and have maximum control over components, hence allowing exactly one certificate at a given time.		CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-1316	SCMS POC OUT OF SCOPE	Additional SCMS component	Each SCMS component shall be allowed to request and receive a certificate that is valid for the next time period at a time defined by	To allow continuity of secure	The additional certificate is likely requested by the SCMS component	CRL Store, CRLG, DCM, ECA, IBLM,

Key	Status	Summary	Description	Justification	Notes	Component/s
		certificate for the next time period	the certificate policy given by the SCMS Manager.	communication between SCMS components.	towards the end of the current time period.	ICA, LA, PCA, PG, RA
SCMS-2522	CLOSED	TLS version	All TLS communication shall use TLS version 1.2 as specified in RFC5246 or higher	to avoid known security issues in older versions of the protocol.		CRL Store, CRLG, DCM, ECA, IBLM, LA, MA, PCA, PG, RA

Table 10 Use Case 11 - Requirements

[13 issues](#)

Summary Showing Trust Anchor Relationships Only

SCMS Root CA Trust Anchor Relationships - Summary

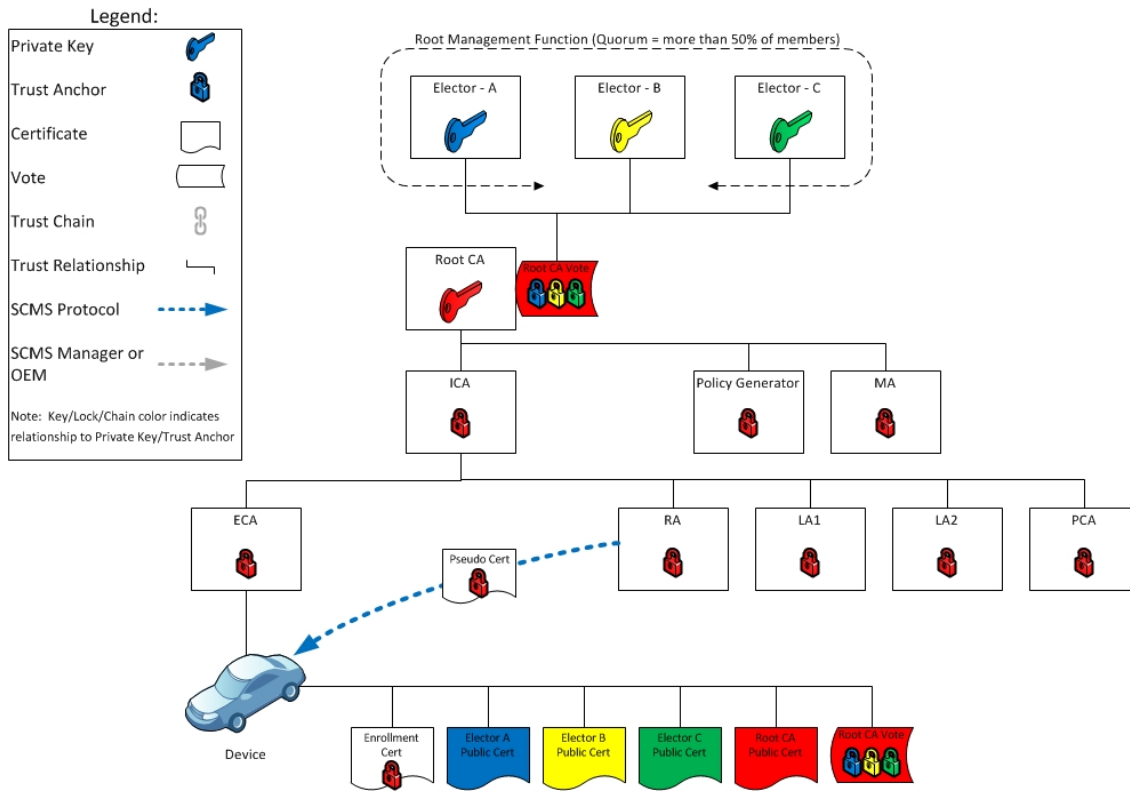


Figure 35 SCMS Root CA Trust Anchor Relationships - Overview

Typical SCMS Operations

Day 1: Typical SCMS System Operations

SCMS Root CA & Elector Trust Relationships

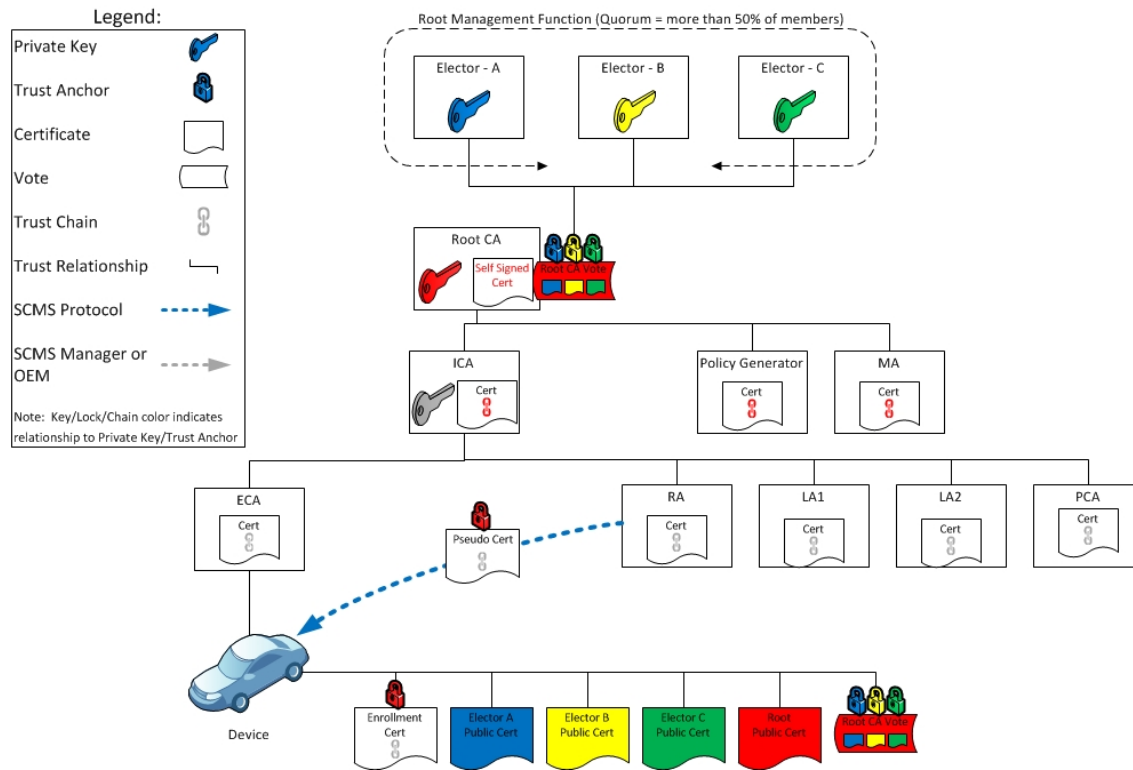


Figure 36 SCMS Root CA & Elector Trust Relationships

Scenario 1: Life Cycle of Elector (Level 0) Revocation and Replacement

Scenario 1, Day 2: Process to Revoke an Elector while Maintaining Functionality

Elector A Revocation Process

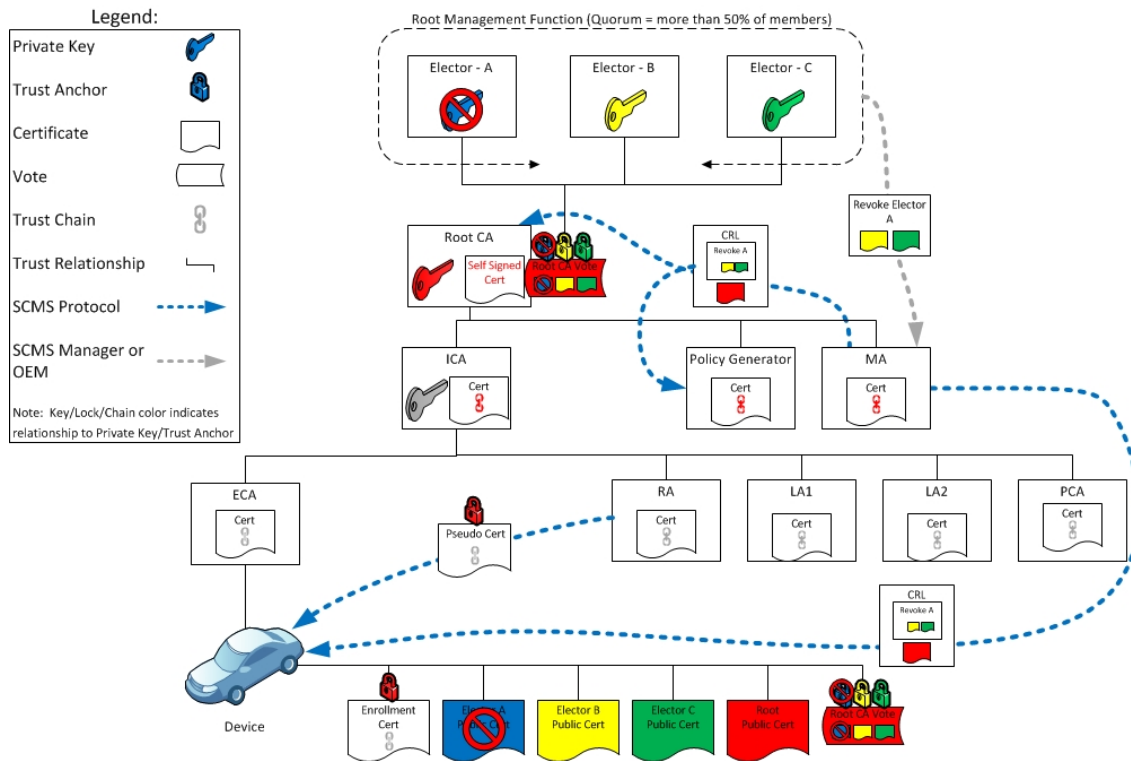


Figure 37 Elector A Revocation Process

Scenario 1, Day 3: System Functional for Period of Time with Two Root Endorsers

SCMS Operational with Electors B & C Only

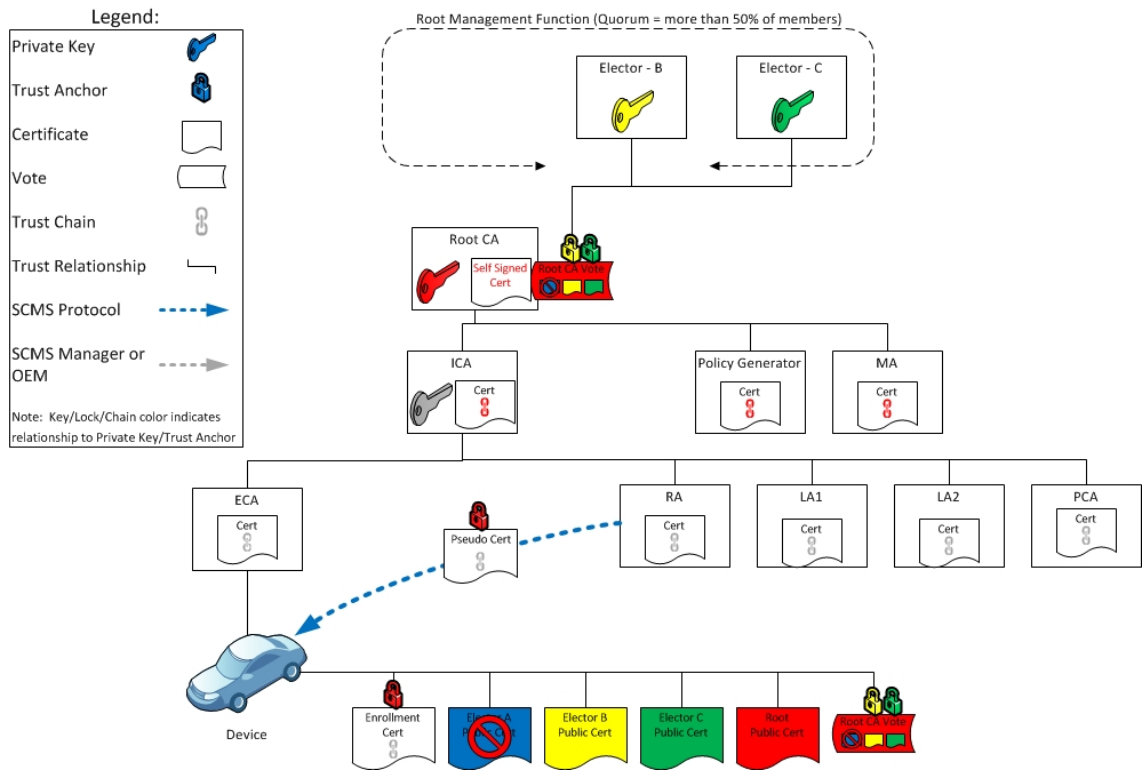


Figure 38 SCMS Operational with Electors B & C Only
Scenario 1, Day 4: Introduction of Replacement Elector

Introduce Elector D

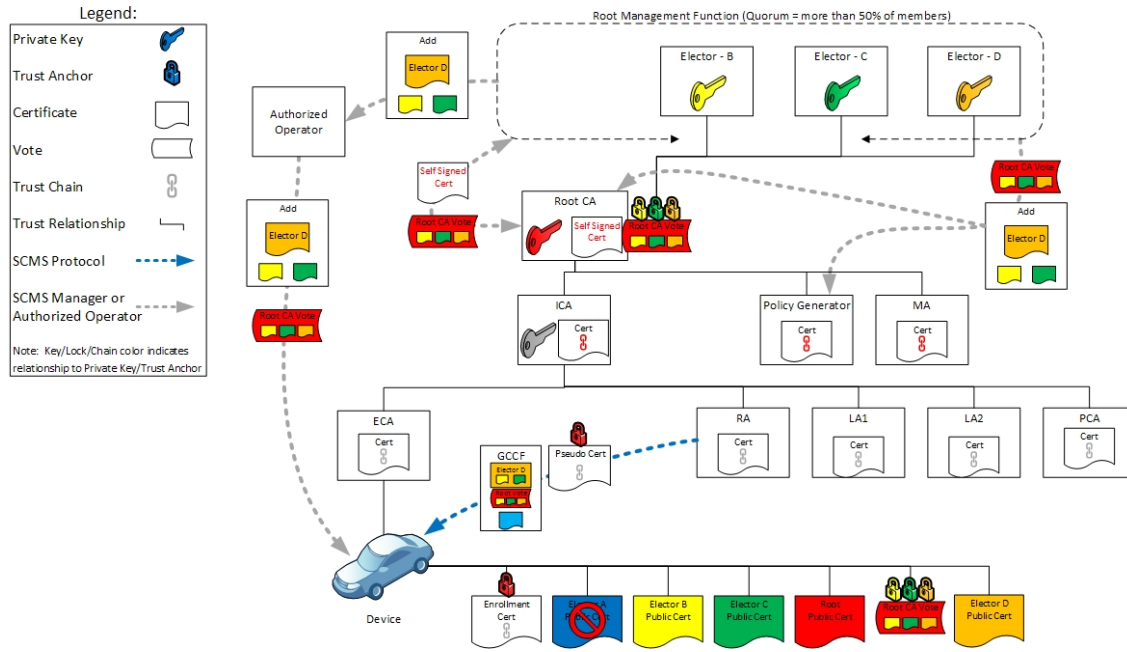


Figure 39 Introduce Elector D

Scenario 1, Day 5: Steady State Operations after the Introduction of Replacement Elector

SCMS Trust Relationships with Elector D

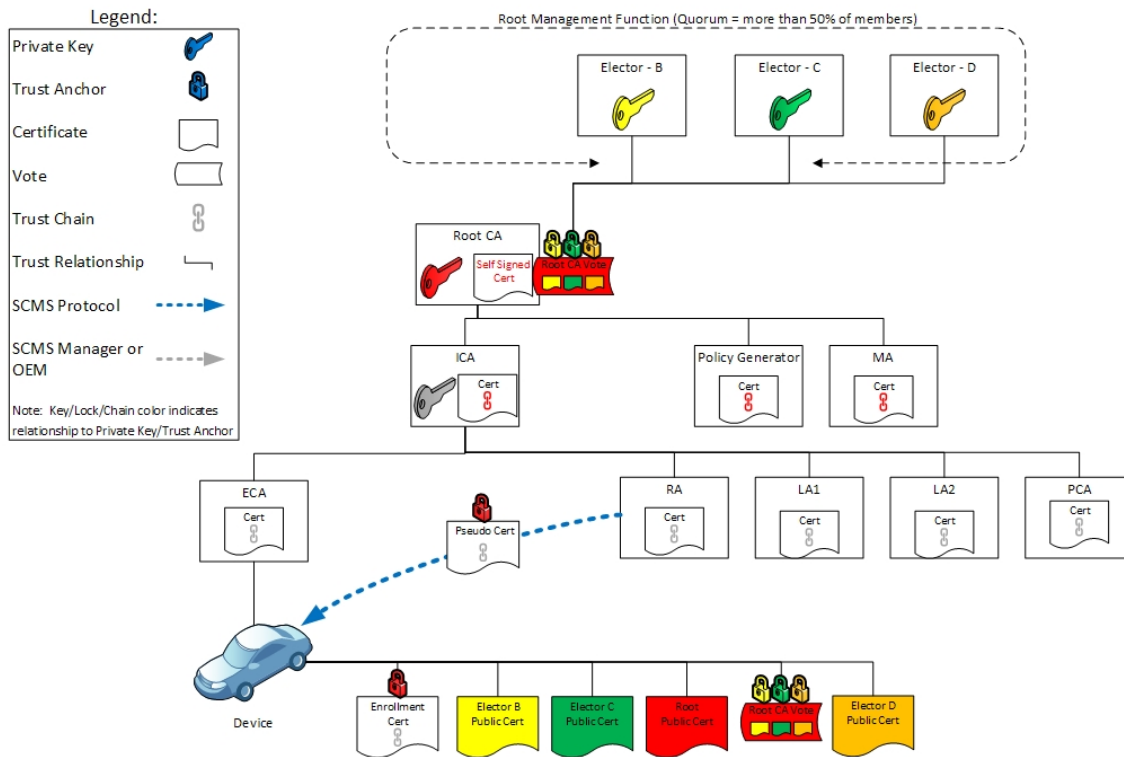


Figure 40 SCMS Trust Relationships with Elector D

Scenario 2: Life Cycle of Root CA (Level 1) Revocation and Replacement

Scenario 2, Day 2: Prepare New Root CA

Create Replacement Root CA & Distribute to SCMS Servers

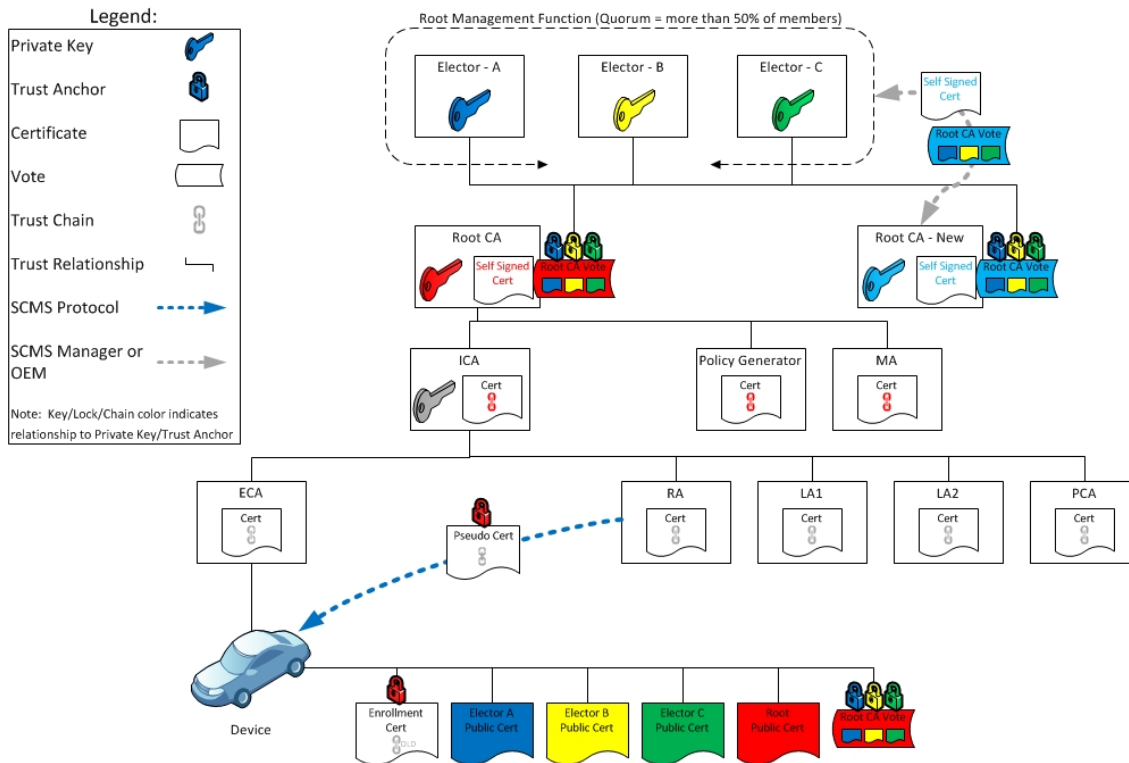


Figure 41 Create Replacement Root CA & Distribute to SCMS Servers

Scenario 2, Day 3: Generate New Certificates for all SCMS Components & Distribute

Introduce Replacement Root CA Before Revoking Current Root CA

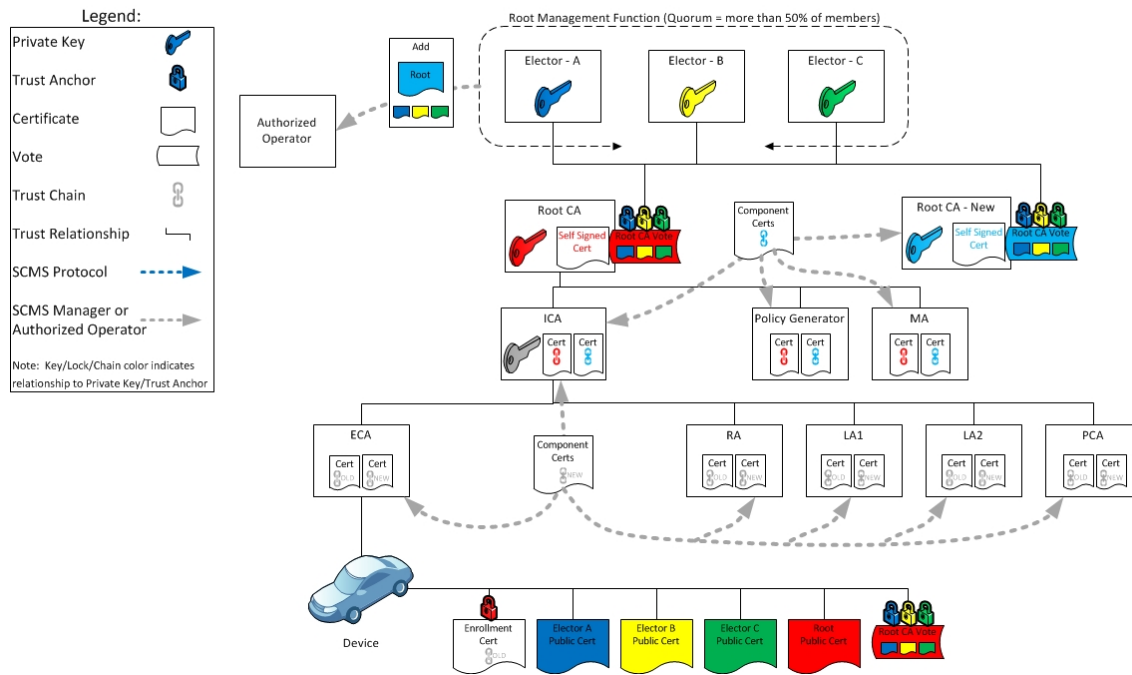


Figure 42 Introduce Replacement Root CA before Revoking Current Root CA
Scenario 2, Day 4: Revoke Root CA

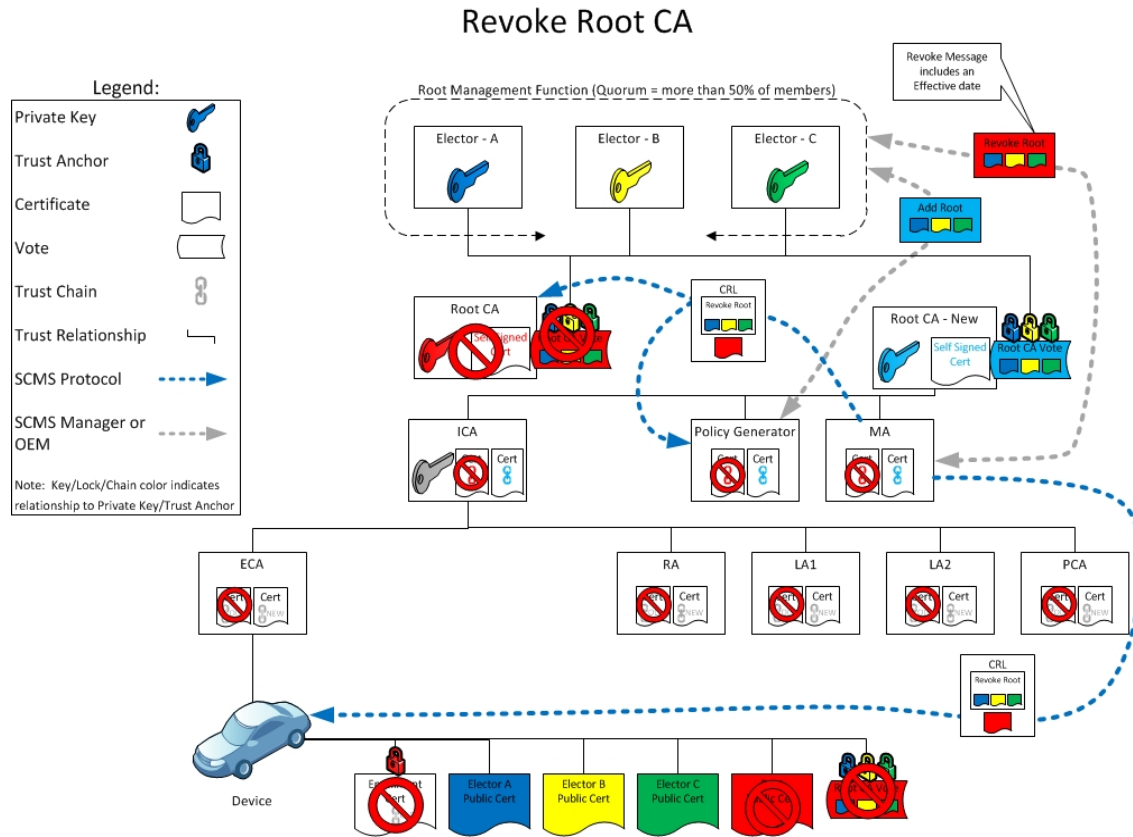


Figure 43 Revoke Root CA

Scenario 2, Day 5: Condition of SCMS while Root CA is Revoked

Root Revoked – System Non-functional

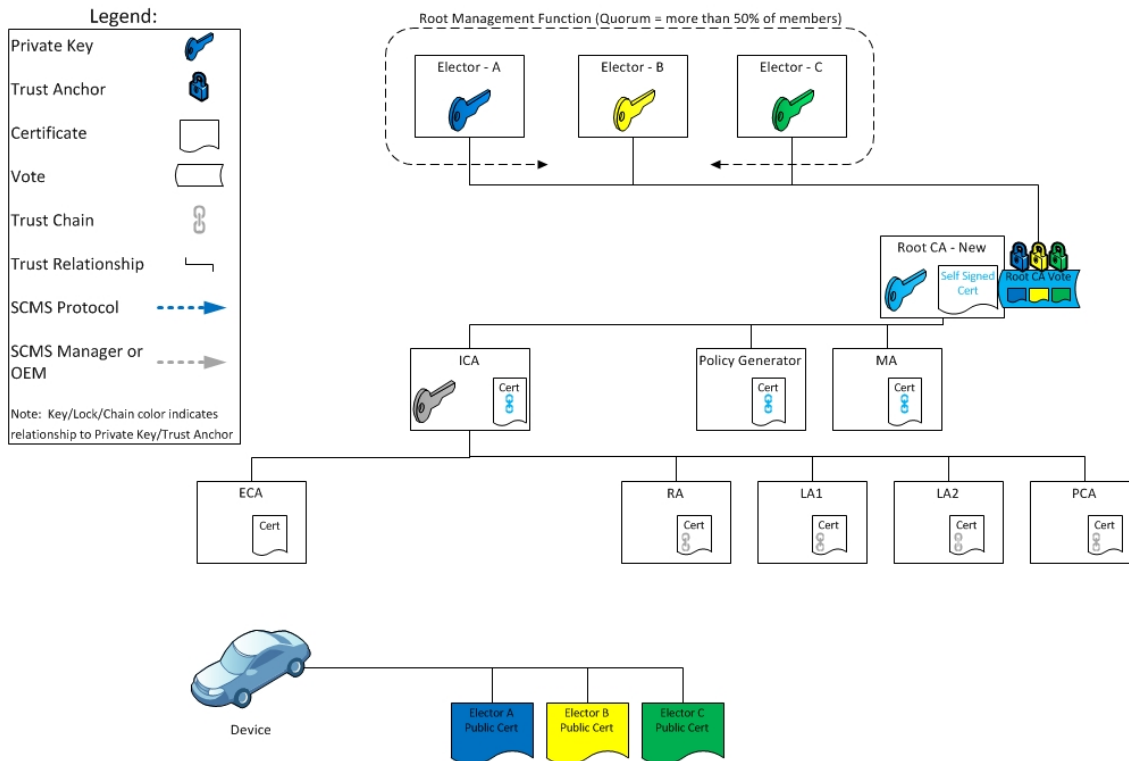


Figure 44 Root Revoked - System Non-functional

Scenario 2, Day 6: EEs Updated with New Root Certificate, New Enrollment Certificate and New Pseudonym Certificates

Update EEs with new certificates

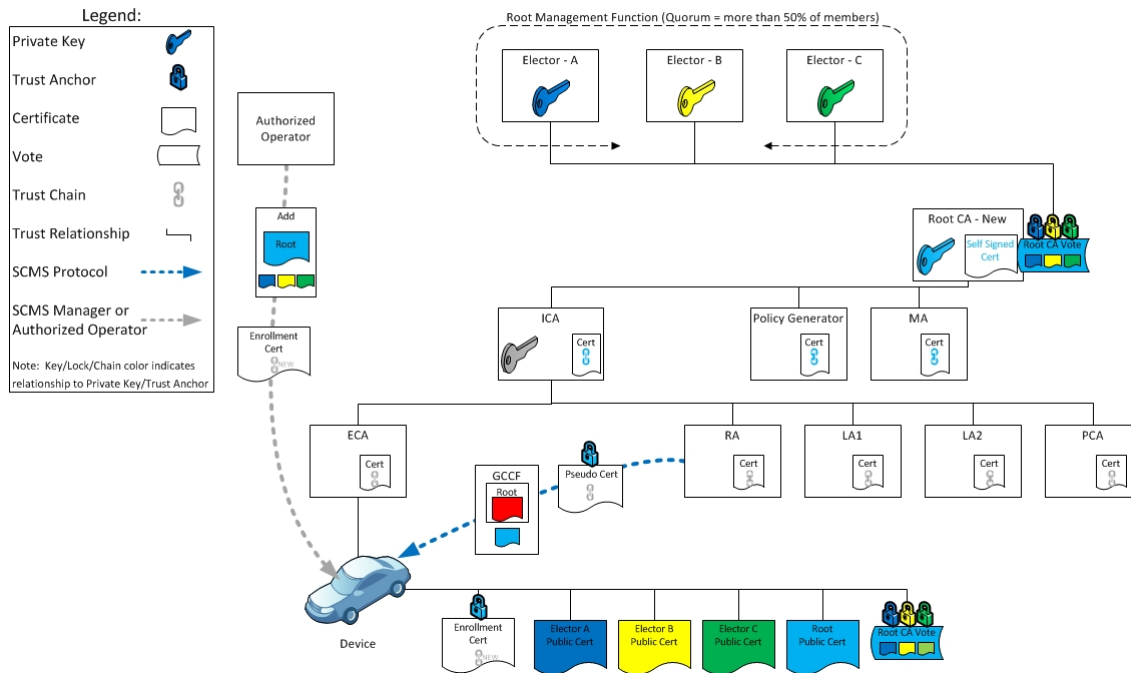


Figure 45 Update EEs with New Certificates

Step 11.1: Add SCMS Component

Goals

The goal of this use case with its collection of subsequent steps is to describe the procedures for adding backend SCMS components to the system. In all cases, before a component can be added it must first be setup correctly using the appropriate [Component Setup](#) use case.

Assumptions

- All components that will be added to the SCMS have already been configured using the appropriate [Component Setup](#) use case
- All components that will be added to the SCMS have been certified and approved through a process defined by the SCMS Manager
- The addition of any new SCMS component is coordinated and managed by an authorized agent of the SCMS Manager or local ICA Manager
- Many of the steps in the component add procedure are defined as "manual" and are not fully specified or defined in SCMS requirements. The details for these procedures will be defined by individual implementations. The goal of the SCMS requirements and these use cases are to preserve the security and integrity of the SCMS system and ensure compatibility among individual SCMS components while granting significant latitude for diverse implementations.

Conditions

A new SCMS component may be added under five conditions. In many cases, these conditions require the same add procedure, but there are situations where the procedure is very different. The five conditions are defined here. Individual component use cases will describe the procedure for adding the new component or managing the transition to the component's parameters.

1. Net New

- a. This is the case where a net-new component is being added to the SCMS. This new component will be configured to receive and process messages from other components.
 - b. New components are assumed to have internal storage that is prepared to store new data, but that is in a state that is initially cleared.
2. SCMS Certificate Retired and Re-Issued
 - a. Most SCMS components have an SCMS certificate that has a useful life that is shorter than the expiration time for the certificate. At the end of this useful life, the old certificate is retired and an ICA or root CA will issue a new certificate.
 - b. When a certificate is retired, previous signatures issued by that component may still be trusted, so normal operation may resume without the need to re-certify any sub-components.
3. Component Decommissioned and Replaced
 - a. An SCMS component may be securely decommissioned and replaced. At a high level, this implies that the private key is securely destroyed or the physical device is put into secure storage. When this happens, the SCMS Manager or local ICA Manager may determine that the component's SCMS certificate does not need to be revoked.
 - b. In this situation, a replacement component may share the same network address as the original component and it may be possible to transfer securely the internal storage of the original component to the new device. However, the replacement component will have a new SCMS certificate.
 - c. Note that this condition is very similar to a retired SCMS certificate, but in this case, the component is being replaced with a new device and it may happen prior to the planned end of useful life for the original SCMS certificate.
4. SCMS Certificate Revoked, Component Replaced or Re-Certified
 - a. When a component's certificate is revoked, it may be necessary to replace or re-certify the component.
 - b. In this situation, the replacement (or re-certified) component is assumed to have the same network address (but it will require a new TLS certificate) as the original component but it will have a new key pair, new SCMS certificate, and the component's internal memory will be cleared.
5. Certifying SCMS Certificate Revoked, Component Re-Certified
 - a. When any higher level CA in the chain that issued a component's SCMS certificate is revoked, then the component's SCMS certificate shall also be treated as untrusted and implicitly revoked.
 - b. When this happens, the SCMS Manager or local ICA Manager may determine that the impacted components can be re-certified and re-used.
 - c. As in the case where the device itself is revoked, the component may retain the same network address (and possibly the same TLS certificate), but it will have a new SCMS certificate and the internal storage may be cleared.
 - d. Note that this condition is very similar to the case where the SCMS certificate of the component is revoked and is treated as equivalent in most of the component add use cases.

Step 11.1.1: Add Non-Root SCMS Component

Goals

Provide a process for adding an SCMS component other than root CA, e.g. intermediate CA, PCA, RA, etc.

Assumptions

Use cases and more specific requirements are in the subsections under this heading.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-2842	REVIEW	Estimated project expiration	Certificates shall expire on or before 12:00:00 UTC January 3, 2025.	To ensure no certificates are valid beyond the defined project period.	Due to the 1609.2 sixtyHours unit of time, the actual certificate expiration may be up to 60 hours after the estimated project expiration of 00:00:00 UTC January 1, 2025. This is for CV-Pilot only.	CRLG, DCM, ICA, LA, MA, PG, RA
SCMS-1422	SCMS POC OUT OF SCOPE	Renewal of component certificate	A SCMS component shall request rollover IEEE 1609.2 certificates no sooner than 3 months prior to the end of the In-use life of the current certificate. A SCMS component shall not issue rollover IEEE 1609.2 certificates prior 3 months to the end of the In-use life of the current certificate.	To prevent the existence of certificates that are not valid until a significant time in the future.	Does not apply to component compromise/revoked situations. For the PoC & CV-Pilot, 3 months is being used. This should be re-evaluated for other deployments.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA, RCA
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other	In the PoC, this will occur by a manual process.	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
				component via GCCF.		
SCMS-1319	MANUAL PROCESS	Component certificate expiration	The component shall request a certificate with a validity of 3 years and 1 week.	Use 3 years for standard SCMS components	This is for POC & CV-Pilot only.	CRLG, DCM, LA, MA, PG, RA
SCMS-1035	SCMS POC OUT OF SCOPE	Error code: tcIssuedCertComponentPublished	TCotSCMSM shall log "Error code: tcIssuedCertComponentPublished", if the certificate issued to the new component could not be published to another SCMS component (e.g. DCM).	Issued certificate cannot be published can be tracked		TCotSCMSM
SCMS-1032	SCMS POC OUT OF SCOPE	Error code: tcComponentUnreachable	TCotSCMSM shall log "Error code: tcComponentUnreachable", if testing the addressing info of the new component resulted in a failure.	Unreachable components can be tracked		TCotSCMSM
SCMS-1031	SCMS POC OUT OF SCOPE	Error code: tcComponentAddressingInfoInvalid	TCotSCMSM shall log "Error code: tcComponentAddressingInfoInvalid", if the addressing info required to locate the new component on the network supplied by the new component is invalid.	Invalid component address can be tracked		TCotSCMSM
SCMS-1027	SCMS POC OUT OF SCOPE	Error Code: authCAUnauthorizedAdd	A SCMS component shall log "Error Code: authCAUnauthorizedAdd", if the authorizing Root CA or ICA has refused to issue a certificate to the new component due to it being unauthorized.	So unauthorized add can be logged		
SCMS-1026	MANUAL PROCESS	Error code: authCAAuthenticationFailed	A SCMS component shall log "Error code: authCAAuthenticationFailed", if the authorizing Root CA or ICA designated by the TCotSCMSM to the new component cannot be authenticated.	Authentication failure error		CRL Store, CRLG, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-784	MANUAL PROCESS	TCotSCMSM inform ECA of DCMs	The local ICA Manager shall update the ECA when a DCM is added and provide the DCM's SCMS certificate and TLS certificate.	The ECA will need to authenticate with DCMs, and hence will need to	In the PoC this will occur by a manual process.	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
				be aware of their identity		
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM

Table 11 Use Case 11.1.1 - Requirements

[11 issues](#)

Step 11.1.1 - Add CRLG

Goals

The CRL Generator (CRLG) is an SCMS component that signs and publishes updated Certificate Revocation Lists (CRLs). In normal operation, the CRLG receives commands from the Misbehavior Authority (MA) or the TCotSCMSM to add revoked certificates to the current CRL. The CRLG adds revocation information of the certificates to the current CRL file, signs the new file, and publishes the new CRL. The CRLG does not directly receive messages from any other SCMS back-end components. The updated CRL is published to the CRL Store.

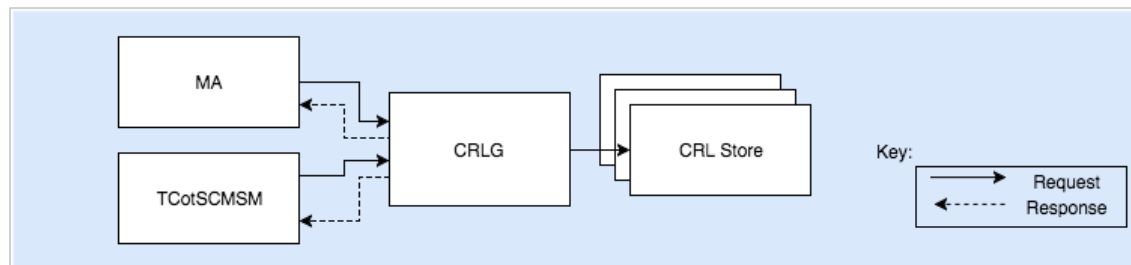


Figure 46 CRLG Messaging Diagram

The figure shows that the CRLG will receive messages from the MA and from the TCotSCMSM. It must also be able to publish a new CRL to one or more CRL Stores.

Process

To add a new CRLG to the SCMS, the TCotSCMSM must enable communication from the MA to the CRLG. It must also enable the CRLG to publish updated CRLs to one or more CRL Stores.

Specifically, the new CRLG must be configured with the following information:

1. The FQDN and TLS certificate of one or more CRL Store
2. The TLS certificate of the MA
3. Security credentials needed to authenticate the TCotSCMSM (this may be certificate based, user name and password, or secured through privileged access to the CRLG internal storage)

When a new CRLG is added, the MA must be updated with the following information:

1. The FQDN and TLS certificate of the CRLG

When a new CRLG is added, all CRL stores must be updated with the following information:

1. The TLS certificate of the CRLG

End State

After completing this use case, the CRLG will be configured with the following connection information:

CRLG Value	Notes
CRL Store FQDN and TLS certificate	The CRLG requires the network address of one (or more) CRL Store. For the PoC, there will be only one CRL Store.
MA TLS Certificate	The CRLG requires the MA's TLS certificate for authentication.

Table 12 CRLG Values

After completing this use case, the MA will be configured with the following connection information:

MA Value	Notes
CRLG FQDN and TLS certificate	The MA requires the network address of one CRLG. For the PoC, there will only be one active CRLG.

Table 13 MA Values

After completing this use case, the CRL store will be configured with the following connection information:

CRL Store Value	Notes
CRLG TLS certificate	CRL store requires the TLS certificate of one or more CRLG. For the PoC, there will only be one active CRLG.

Table 14 CRL Store Values

Special Cases

The procedure described above shall be used when configuring a new CRLG. The following details define how to deal with special cases of replacing a previous CRLG component.

- If the CRLG's SCMS certificate has retired and a new certificate is issued, there is no need for a special procedure to add the new certificate. It will be learned by all SCMS components when they load the latest CRL and validate the CRLG signature. The CRLG can continue to use the same network address and TLS certificate as before.
- If the CRLG has decommissioned and replaced, it will be necessary to update the internal memory of the replacement component with the last known state of the CRL. This may be done through secure transfer to the new component or by loading and validating the last published CRL. No other configuration changes are needed (provided that the replacement component has the same network address and TLS certificate as the prior CRLG).
- If the CRLG's SCMS certificate has been revoked, or if the root CA's certificate has been revoked, then the SCMS Manager will have to perform an investigation to validate the contents of the latest CRL state prior to re-certifying a replacement CRLG. Note that once a CRL is published, none of the contents can be removed from the list, even if they were added incorrectly (i.e., you cannot un-revoke a component even if you realize that the component was never compromised).

Assumptions

- The CRLG has been set up as described in the [Setup CRL Generator](#) use case
- The root CA issues the CRLG's SCMS certificate
- SCMS components and EEs can learn and validate the SCMS certificate when they download the latest CRL. There is no need to distribute the CRLG certificate to all components.
- The CRLG periodically publishes updated CRLs to the CRL Store
- The TCotSCMSM can trigger an immediate CRL update if necessary
- The CRLG will provide an interface to allow the addition or removal of CRL Stores from the list of sites that receive new CRL updates. This interface will require that there is always at least one active CRL Store. The mechanism for adding and removing CRL Store addresses in the CRLG is implementation specific and is not defined here.
- For the PoC there will be only one CRLG in the SCMS
- The CRLG will need to incorporate root and elector revocation commands on the CRL. These commands will be assembled by the TCotSCMSM and delivered to the CRLG through the communications mechanism established in this use case.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS policy and/or based on the expiration and In-use lifetime of subordinate certificates.	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA
SCMS-1581	SCMS POC OUT OF SCOPE	Component certificate in-use period	The SCMS component shall use its certificate for an in-use period of 3 years.	Use 3 years for standard SCMS components	Out of scope as this needs to be implemented as operational policy. This is for POC & CV-Pilot only.	CRLG, DCM, LA, MA, PG, RA
SCMS-1725	MANUAL PROCESS	Component certificate FQDN match	The SCMS component shall have a certificate with a certificateld field that matches the FQDN of the component.	FQDN of each component must match the official ID of the component.		CRLG, DCM, ECA, LA, MA, PCA, PG, RA

Table 15 Step 11.1.1 Add CRLG - Requirements

[4 issues](#)

Step 11.1.1 - Add ECA

Goals

The Enrollment Certificate Authority (ECA) is an SCMS backend component that signs enrollment certificates for End Entity (EE) devices. In normal operation, the ECA receives and responds to requests from one or more Device Configuration Managers (DCMs). The addition of an ECA to the SCMS requires that the ECA is informed of the DCMs that will be sending requests and that the network is set up to enable those requests to reach the ECA.

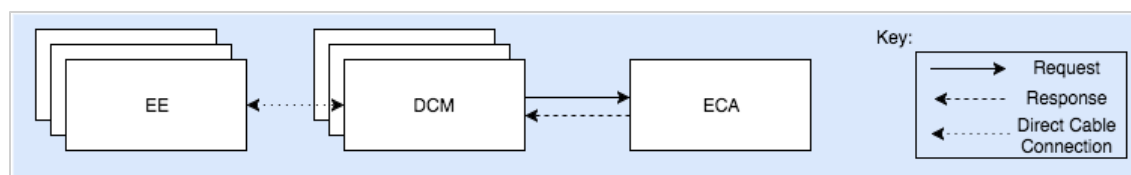


Figure 47 ECA Messaging Diagram

The figure shows that the ECA will receive messages initiated by one or more DCM. It is recommended, but not required, that each DCM work with a single ECA.

The SCMS PoC design requires that each ECA maintain a list of DCMs that are authorized to send it enrollment requests. It does this by maintaining a list of authorized DCM TLS certificates.

Each ECA will be assigned to one RA, which will only trust this ECA's enrollment certificates.

Process

To add an ECA to the SCMS, the local ICA Manager must provide a list of TLS certificates to the ECA for all DCMs that are authorized to send it requests. The ECA must also configure its network to allow communication from the DCMs to the ECA.

In order to access the new ECA, each DCM that will send it requests must be provided with the following:

1. The FQDN of the new ECA
2. The ECA's TLS certificate

The local ICA Manager must inform the designated RA of the newly added ECA's SCMS certificate.

End State

After completing this use case, the ECA will be configured with the following values:

ECA Value	Notes
One or more DCM TLS certificates	The ECA requires a list of authorized DCMs that can send it signature requests. The ECA shall maintain a table of allowed DCM TLS certificates.

Table 16 ECA Values

After completion of this use case, each DCM that is authorized to communicate with the newly added ECA will be configured with the following values:

DCM Value	Notes
FQDN and TLS certificate of the ECA	Each DCM requires the FQDN and TLS Certificate of one (or more) ECA to process enrollment requests.

Table 17 DCM Values

After completion of this use case, the designated RA will have the following information:

RA Value	Notes
SCMS certificate of the newly added ECA	One RA must be configured to accept enrollment certificates from the new ECA.

Table 18 RA Values

Special Cases

The procedure described here can be used when adding a net-new ECA to the SCMS.

Other conditions must be managed as follows:

- ECA SCMS Certificate Retired and Re-Issued - When this happens, the RA that the ECA is assigned to must be informed of the new ECA certificate. The local ICA Manager will perform this update.
- ECA Decommissioned and Replaced - If an ECA is securely decommissioned, enrollment certificates that were previously issued may continue to be trusted. The local ICA Manager must instruct all DCMs that they can no longer send requests to the decommissioned ECA. A new replacement ECA may be added using the procedure described here as if it were a net-new ECA to the SCMS.
- ECA Revoked: see [Step 11.2.1 - Revoke ECA](#)

Assumptions

- The ECA must be configured using the [Setup ECA](#) use case before it can be added
- The ECA will support a mechanism for adding and removing authorized DCM certificates from its internal table. The method of updating this table is implementation specific and is not part of the SCMS design.
- Each DCM will maintain a list of one (or more) active ECAs that it may use for signing enrollment certificates
- Each RA will maintain a list of ECAs whose enrollment certificates it may trust
- Each ECA will be assigned to a single RA

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must authorize the new component.	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-784	MANUAL PROCESS	TCotSCMSM inform ECA of DCMs	The local ICA Manager shall update the ECA when a DCM is added and provide the DCM's SCMS certificate and TLS certificate.	The ECA will need to authenticate with DCMs, and hence will need to be aware of their identity	In the PoC this will occur by a manual process.	TCotSCMSM
SCMS-1039	SCMS POC OUT OF SCOPE	Error Code: tcNotifyECAofDCMFailure	TCotSCMSM shall log "Error Code: tcNotifyECAofDCMFailure", if the TCotSCMSM cannot notify the ECA of the DCMs with which it will communicate.			TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other component via GCCF.	In the PoC, this will occur by a manual process.	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS policy and/or based on the expiration and In-use lifetime of subordinate certificates.	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA
SCMS-1591	CLOSED	ECA certificate validity	ECA shall request an ECA certificate with a validity of 11 years.	To support issuing of subordinate certificates.	This is for POC only.	ECA
SCMS-1602	SCMS POC OUT OF SCOPE	ECA certificate in-use period	ECA shall use its ECA certificate for an in-use period of 3 years.	Use 3 years for Enrollment SCMS components	Out of scope as this needs to be implemented as operational policy. This is for CV-Pilot only.	ECA
SCMS-1605	MANUAL PROCESS	ECA certificate validity	ECA shall request an ECA certificate with a maximum validity of 8 years +/- 1 week.	To support issuing of subordinate certificates.	1st generation: Start = 428,630,405, Duration = 1,084 sixtyHours This is for CV-Pilot only.	ECA

Table 19 Use Case 11.1. Add ECA - Requirements

[9 issues](#)

Step 11.1.1 - Add ICA

Goals

The Intermediate Certificate Authority (ICA) is a non-central, backend component of the SCMS. There may be many instances of ICAs within the system. The ICA authorizes all other non-central components including ECAs, PCAs, RAs, LAs, or additional ICAs. Adding a new ICA to the system makes the new ICA available to authorize new components.

An ICA is intended to be an offline component, meaning that it should be configured with no direct network access or address. A local ICA Manager operates the ICA manually. The specific details of how the operator presents messages to the ICA is implementation-specific and subject to review by a certification procedure approved by the SCMS Manager.

Procedure

The procedure required for adding an ICA to the system depends on whether the new ICA is replacing a previously revoked or removed ICA or if it is a net-new component.

New ICA

A new ICA must be properly set-up using the process described in the [Setup ICA](#) use case. Since the ICA operates offline, there are no network addresses or other parameters to configure when adding the ICA.

Note that if the new ICA issues a certificate for a PCA or RA, then the [Add PCA](#) use case will cause the ICA to be registered with the Policy Generator (PG) for inclusion in future updates to the Global Certificate Chain File (GCCF). There is no need to register the ICA with the PG until a new PCA or RA is added. All other components that are issued certificates by the ICA will make the ICA certificate available to recipients of their messages when required.

Re-Certified ICA

An ICA certificate has a limited useful life that is shorter than the expiration period of the certificate. When an ICA certificate is retired, the current private key must be deleted, a new key pair must be generated, and a new certificate must be issued. There are no additional actions needed to add or enable the new ICA certificate. As with the procedure for adding a new ICA (above), there is no need to communicate the new ICA certificate to the PG or any other components.

Replacement ICA

When replacing an ICA that was previously removed or revoked, the new component must first be set up using the [Setup ICA](#) use case. The local ICA Manager must then use the new component to re-issue certificates to all of the components that were previously authorized under the ICA that was removed or revoked, see [Step 11.2.1 - Revoke ICA](#)

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-715	CLOSED	Provide Elector Certificates	The TCotSCMSM shall provision all SMCMS components with the self-signed certificates of all electors that are valid at the time of the component setup.	Root Management messages require signatures from the Electors to be validated, so authentic root CA Certificates are also required.	When receiving Root Management messages through the GCCF or LCCF, the authenticity of the messages will be validated by counting valid Elector signatures on the message and ensuring that at least that number (quorum) required in the Global Policy is present, after which the Root Management message can be processed.	TCotSCMSM
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must authorize the new component.	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other component via GCCF.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			policy and/or based on the expiration and In-use lifetime of subordinate certificates.			
SCMS-1596	CLOSED	ICA certificate validity	ICA shall request an ICA certificate with a validity of 13 years.	To support issuing of subordinate certificates.	This is for POC only.	ICA
SCMS-1597	SCMS POC OUT OF SCOPE	ICA certificate in-use period	ICA shall use its ICA certificate for an in-use period of 4 years.	The in-use period shall be short to minimize impact, if revocation is required.	Out of scope as this needs to be implemented as operational policy. This is for POC only.	ICA
SCMS-1603	MANUAL PROCESS	ICA certificate validity	ICA shall request an ICA certificate with a maximum validity of 8 years +/- 1 week.	To support issuing of subordinate certificates.	Start = 410,313,605 Duration = 1,169 sixtyHours This is for CV-Pilot only.	ICA
SCMS-1604	SCMS POC OUT OF SCOPE	ICA certificate in-use period	ICA shall use its ICA certificate for the entire validity period of the certificate.	The In-use period shall be short to minimize impact if revocation is required.	Out of scope as this needs to be implemented as operational policy. This is for CV-Pilot only.	ICA

Table 20 Use Case 11.1.1 Add ICA - Requirements

[9 issues](#)

Step 11.1.1 - Add MA

Goals

The Misbehavior Authority (MA) is an intrinsically central SCMS component that performs multiple functions to manage risk in the SCMS like receiving misbehavior reports from EEs, investigating potential misbehavior, and blacklisting or revoking components. As a central component, there will only be one MA instance.

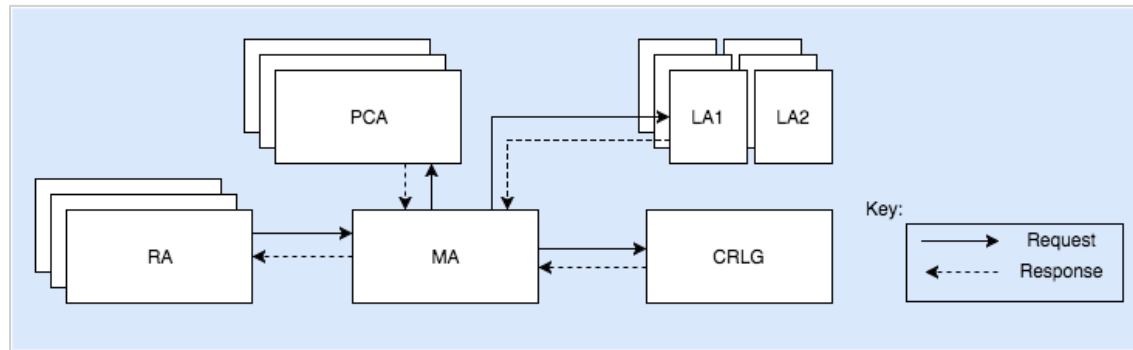


Figure 48 MA Messaging Diagram

The figure shows that the MA receives requests from one or more RAs and it sends out requests to PCAs, pairs of LAs, and the CRLG.

EEs must encrypt misbehavior reports to be sent to the MA. Therefore, all EEs will need the current MA certificate, which they obtain during enrollment from the DCM or during operation from their assigned RA.

Procedure

Components that communicate with an added MA must be properly configured.

End States

After completing this use case, the MA will be configured with the following values:

MA Value	Notes
List of RA TLS certificates	The MA must maintain a list of TLS certificates for all RA's that will forward misbehavior reports on behalf of EEs.
List of PCA FQDN and TLS certificates	The MA must maintain a list of all PCA network addresses and TLS certificates.
List of LA FQDN and TLS certificates	The MA must maintain a list of all LA's and their TLS certificates.
CRLG FQDN and TLS certificate	The MA must be able to send revocation requests to the CRLG.

Table 21 MA Values

After completing this use case, RAs will be configured with the following values:

RA Value	Notes
MA FQDN and TLS certificate	Each RA must be able to establish a secure connection to the MA.
MA's SCMS certificate	Each RA must provide MA's certificate to its EEs.

Table 22 RA Values

After completing this use case, DCMs will be configured with the following values:

DCM Value	Notes
MA's SCMS certificate	Each DCM must provide the current MA's certificate to EEs during enrollment.

Table 23 DCM Values

All RAs, PCAs, LAs, and the CRLG will need a copy of the new MA's TLS certificate so that they can establish secure communication. These components can learn the MA's SCMS certificate by validating any signed message from the MA and chaining it up to the SCMS root certificate (which they already have).

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-715	CLOSED	Provide Elector Certificates	The TCotSCMSM shall provision all SMCMS components with the self-signed certificates of all electors that are valid at the time of the component setup.	Root Management messages require signatures from the Electors to be validated, so authentic root CA Certificates are also required.	When receiving Root Management messages through the GCCF or LCCF, the authenticity of the messages will be validated by counting valid Elector signatures on the message and ensuring that at least that number (quorum) required in the Global Policy is present, after which the Root Management message can be processed.	TCotSCMSM
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must authorize the new component.	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other component via GCCF.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			policy and/or based on the expiration and In-use lifetime of subordinate certificates.			
SCMS-1581	SCMS POC OUT OF SCOPE	Component certificate in-use period	The SCMS component shall use its certificate for an in-use period of 3 years.	Use 3 years for standard SCMS components	Out of scope as this needs to be implemented as operational policy. This is for POC & CV-Pilot only.	CRLG, DCM, LA, MA, PG, RA
SCMS-1725	MANUAL PROCESS	Component certificate FQDN match	The SCMS component shall have a certificate with a certificateId field that matches the FQDN of the component.	FQDN of each component must match the official ID of the component.		CRLG, DCM, ECA, LA, MA, PCA, PG, RA

Table 24 Use Case 11.1.1 Add MA - Requirements

[7 issues](#)

Step 11.1.1 - Add PCA

Goals

The Pseudonym Certificate Authority (PCA) is an intrinsically non-central component of the SCMS. It issues pseudonym, identification, and application certificates for End Entities (EEs). There may be multiple PCAs in the SCMS. Each PCA is associated with a single RA and a pair of LAs to perform its core functions. The PCA responds to requests from the MA to investigate potential misbehavior.

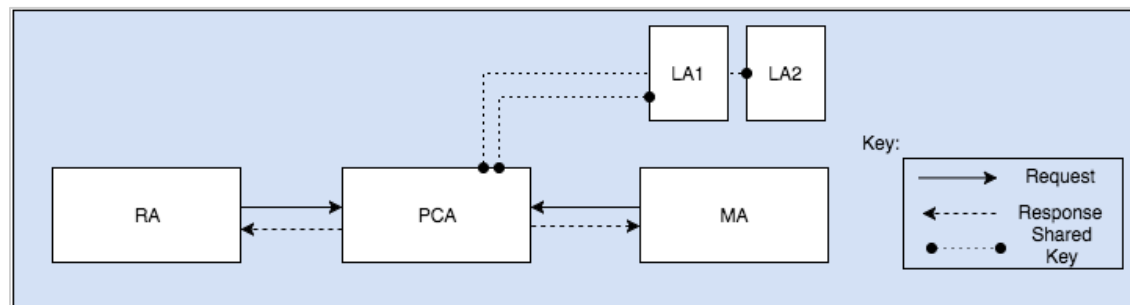


Figure 49 PCA Messaging Diagram

The figure shows that the PCA responds to requests from both the RA and the MA. It also requires shared symmetric encryption/decryption between the LAs and the PCA, although, there is no direct communication between them. The PCA also maintains a secure database containing all pre-linkage values, certificates, and a hash of the certificate request that it received from the RA.

Procedure

To add a new PCA to the SCMS, the local ICA Manager will select an RA and a pair of LAs to associate with the new PCA. It must then coordinate the generation and installation of the shared symmetric encryption/decryption key between the PCA and each of the LAs. It will also inform the RA and the central MA of the PCA's FQDN.

End States

After completing this use case, the PCA will be configured with the following values:

PCA Value	Notes
LA1-PCA shared key	Symmetric encryption key shared with LA1
LA1 ID	A globally unique identifier associated with LA1
LA2-PCA shared key	Symmetric encryption key shared with LA2
LA2 ID	A globally unique identifier associated with LA2

Table 25 PCA Values

After completion of this use case, the designated RA will have the following information:

RA Value	Notes
PCA FQDN	RA will use this address to send certificate signing requests to the PCA. RA signs each request.

Table 26 RA Values

After completion of this use case, the MA will have the following information:

MA Value	Notes
PCA FQDN	The MA must be able to contact the PCA to retrieve linkage values to support misbehavior investigation and blacklisting or revocation.

Table 27 MA Values

After completion of this use case, the designated LAs will have the following information:

LA1/2 Value	Notes
LA1/2-PCA shared key	Each LA (i.e., LA1 and LA2) stores the shared key that was exchanged with the PCA.

Table 28 LA Values

Special Cases

The procedure described above shall be used when adding a new PCA to the SCMS. The following details define how to deal with special cases of replacing a previous PCA component.

- If the PCA's SCMS certificate has been retired and a new certificate is issued, there is no need for a special procedure to add the new certificate. The PCA can continue to use the same FQDN and TLS certificate as before. The RA and MA should be able to learn the new PCA certificate.
- If the PCA has been securely decommissioned and replaced, the local ICA Manager may transfer the contents of the PCA database to the new component. The replacement PCA may use the same network address as the decommissioned device. The RA and MA should be able to learn the new PCA certificate.
- If the PCA's SCMS certificate has been revoked, then in addition to adding the new PCA, all certificates that were previously issued by that PCA will need to be removed by the EEs to which they were issued. This process will be triggered by the presence of the PCA's certificate on the CRL, which is distributed to all EEs (see the [Revoke PCA](#) use case for details on how to revoke a PCA). EEs that become inoperative or are at risk of jeopardizing their privacy because of this action will need to contact their RA to request new certificates or take OEM specific action to recover.
- If an ICA in the PCA's certificate chain or the root CA has been revoked and replaced, then the PCA must generate a new key pair and receive a new SCMS certificate from a re-certified or replaced ICA. As in the case of PCA revocation, affected EEs will need to request new certificates or follow an OEM specified procedure to recover.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-244	CLOSED	PCA Availability	The Local ICA Manager shall make the new PCA's certificate and information to locate it on the network available to any RAs that will forward certificate requests to it.	The PCA must be integrated correctly into the SCMS system.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-715	CLOSED	Provide Elector Certificates	The TCotSCMSM shall provision all SMCMS components with the self-signed certificates of all electors that are valid at the time of the component setup.	Root Management messages require signatures from the Electors to be validated, so authentic root CA Certificates are also required.	When receiving Root Management messages through the GCCF or LCCF, the authenticity of the messages will be validated by counting valid Elector signatures on the message and ensuring that at least that number (quorum) required in the Global Policy is present, after which the Root Management message can be processed.	TCotSCMSM
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must authorize the new component.	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain	In the PoC, this will occur by a manual process.	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	must be available to any other component via GCCF.		
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS policy and/or based on the expiration and In-use lifetime of subordinate certificates.	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA
SCMS-1594	CLOSED	PCA certificate expiration	PCA shall request a certificate with a validity of 4 years.	The expiration must be sufficiently long to issue pseudonym certificates for 3 years in the future.	This is for POC only.	PCA
SCMS-1595	SCMS POC OUT OF SCOPE	PCA certificate in-use period	PCA shall use its certificate for an in-use period of 1 years.	The In-use period shall be short to minimize impact if revocation is required.	Out of scope as this needs to be implemented as operational policy. This is for POC & CV-Pilot only.	PCA
SCMS-1806	MANUAL PROCESS	Register non-central component with the central MA	The Local ICA Manager shall update the MA with the FQDN, TLS certificate, and SCMS certificate of any newly added PCA, LA, or RA.	The MA must know about all PCAs, LAs, and RAs in the system so that it can execute misbehavior investigations and revocation procedures.	In PoC, this will occur by a manual process. When a new PCA, LA, or RA is added, the local ICA manager will notify the TCotSCMSM about the newly added component (this is a manual process). The TCotSCMSM will then update the central MA with the necessary information about the newly added component. It is expected (but not required) that a PCA, RA, and pair of LAs will typically be added as a complete set.	TCotSCMSM
SCMS-2608	MANUAL PROCESS	Map PCA IssuerIdentifier to PCA FQDN	The TCotSCMSM shall associate each PCA IssuerIdentifier (the	A mapping between PCAs' IssuerIdentifier and PCA hostname is needed. During an	For the PoC, the SCMS operator will manually configure the mapping of PCA	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			HashId8 of the PCA signing certificate) with the FQDN of the PCA that has the certificate.	investigation, the MA (GMBD) will receive a cert, extract linkage value, extract IssuerIdentifier, and then ask the PCA that issued the cert for the linkage value. For that step, MA must be able to map the IssuerIdentifier to the PCA hostname. This mapping is maintained by the TCotSCMSM and configured in the MA as needed, or this mapping may be maintained as a built-in feature of the MA.	IssuerIdentifiers with the FQDN of the corresponding PCA.	
SCMS-2843	MANUAL PROCESS	PCA certificate expiration	PCA shall request a certificate with a maximum validity of 4 years +/- 2 weeks.	The expiration must be sufficiently long to issue pseudonym certificates for 3 years in the future.	1st generation: Start = 428,630,405, Duration = 1,084 sixtyHours This is for CV-Pilot only.	PCA

Table 29 Use Case 11.1.1 Add PCA - Requirements

[11 issues](#)

Step 11.1.1 - Add PG

Goals

The Policy Generator (PG) is an intrinsically central SCMS component that maintains and signs updates to the [Global Policy File](#) (GPF) and the [Global Certificate Chain File](#) (GCCF). In addition, the PG is required to sign [Local Policy Files](#) (LPFs) at the request of RAs who want to set local policy values or reduce the volume of information that they distribute to their EEs. When signing LPFs, the PG is responsible for validating that critical global information has not been removed and that all local policy adjustments comply with the global policy.

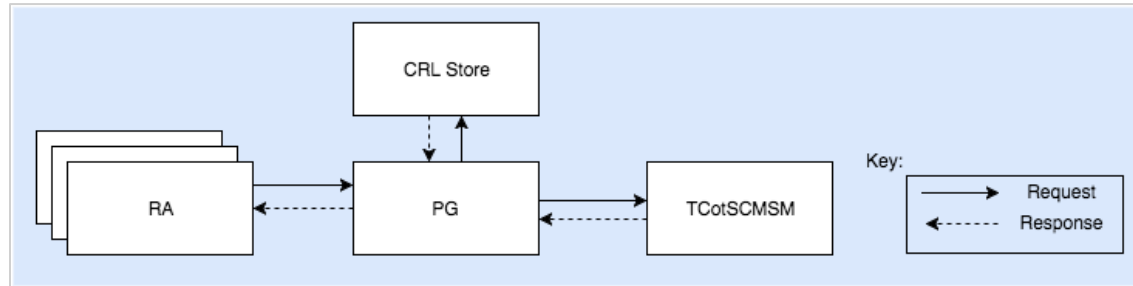


Figure 50 PG Messaging Diagram

The figure shows the request-response relationships of the PG. This diagram explicitly includes the TCotSCMSM, which is the only authority that is able to define changes to global policy, which in turn will be distributed through the GPF. The TCotSCMSM is also the conduit through which new PCA certificate chains can be communicated for addition to the GCCF. Updates to the CRL downloaded from the CRL store might trigger updates to the GCCF in case it contains a revoked certificate.

Procedure

The PG is an intrinsically central component, so there will only be one instance of the PG in the SCMS. When adding or replacing the PG, the TCotSCMSM must ensure that all RAs are aware of the FQDN of the PG and that they are allowed to access to the PG. This will likely be done in cooperation with local ICA Managers who operate each RA.

Prior to initiating this process, the new PG must be set up according to the [Setup Policy Generator](#) use case.

End State

After completing this use case, the PG will be configured with the following values:

PG Value	Notes
CRL Store FQDN	The PG needs to download the latest CRL on a regular basis in order to remove revoked certificates from the GCCF.

Table 30 PG Values

After completing this use case, RAs will be configured with the following values:

RA Value	Notes
PG FQDN	Every RA in the SCMS must be able to contact the PG to request signatures on LPFs and to download the latest GPF and GCCF.

Table 31 RA Values

Special Cases

The procedure defined above applies when a new PG is initially added to the SCMS. Changes required for replacing a PG are required based on the reason for the replacement.

- The PG's SCMS certificate has a useful life that is shorter than the certificate expiration date. When the PG's SCMS certificate is retired, the current private key must be deleted, a new key pair must be generated, and a new SCMS certificate can be installed in the PG. Other SCMS components can learn the new certificate by reading it from the signed updates to the GPF or GCCF and validating that the Root CA signed it. There is no need to communicate the new SCMS certificate directly to any other SCMS components.
- If the PG is securely decommissioned and replaced, the new component must be issued a new SCMS certificate, which can be learned as described above. The current state of the Global Policy and the current GCCF can be securely copied to the replacement component or it can load these files from the last signed copies that were published.
- If a PG is revoked, then it must be re-certified or replaced. The TCotSCMSM must determine if the latest published version of the GPF is reliable for loading into the new component or it can re-create a current Global Policy definition. Similarly, the TCotSCMSM can import a reliable copy of the GCCF or it can collect PCA cert chains and reproduce the GCCF.
- If the root CA is revoked causing implicit revocation of the PG, the TCotSCMSM must re-create the Global Policy and replace or re-certify the PG. In this situation, the GCCF should be re-created by collecting PCA certificate chains to ensure consistency with all newly issued root CA or ICA certificates (if an ICA has been revoked, validated certificate chains for PCAs that were not impacted may be copied from the previous GCCF).

Assumptions

- A new PG must be setup using the [Setup Policy Generator](#) use case
- The interface between the TCotSCMSM and the PG is not defined. It is assumed that updates to the GPF or GCCF will be encoded using the same format as the published files (i.e., using the same ASN.1 message structure up to the "to be signed" structure).
- The method for the TCotSCMSM to authenticate to the PG is not defined. It is assumed that a secure process will manage and log updates to global policy and certificate chain files.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-715	CLOSED	Provide Elector Certificates	The TCotSCMSM shall provision all SMCMS components with the self-signed certificates of all electors that are valid at the time of the component setup.	Root Management messages require signatures from the Electors to be validated, so authentic root CA Certificates are also required.	When receiving Root Management messages through the GCCF or LCCF, the authenticity of the messages will be validated by counting valid Elector signatures on the message and ensuring that at least that number (quorum) required in the Global Policy is present, after which the Root Management message can be processed.	TCotSCMSM
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must authorize the new component.	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other component via GCCF.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			policy and/or based on the expiration and In-use lifetime of subordinate certificates.			
SCMS-1581	SCMS POC OUT OF SCOPE	Component certificate in-use period	The SCMS component shall use its certificate for an in-use period of 3 years.	Use 3 years for standard SCMS components	Out of scope as this needs to be implemented as operational policy. This is for POC & CV-Pilot only.	CRLG, DCM, LA, MA, PG, RA
SCMS-1725	MANUAL PROCESS	Component certificate FQDN match	The SCMS component shall have a certificate with a certificateId field that matches the FQDN of the component.	FQDN of each component must match the official ID of the component.		CRLG, DCM, ECA, LA, MA, PCA, PG, RA

Table 32 Use Case 11.1.1 Add PG - Requirements

[7 issues](#)

Step 11.1.1 - Add RA

Goals

The Registration Authority (RA) is an intrinsically, non-central component of the SCMS. There may be multiple RAs active at any given time in the SCMS.

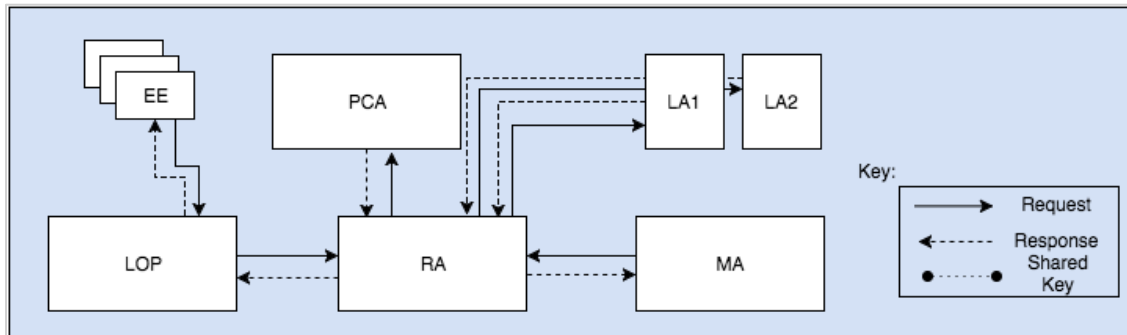


Figure 51 RA Messaging Diagram

The figure shows that each RA supports the following connections:

- The RA can receive and respond to requests from EEs through the LOP which masks the source IP address and route of the EE from the RA. Only EEs that have enrollment certificates from ECAs that are authorized to use the RA will be accepted. Each EE is configured to contact only one RA.
- The RA can initiate certificate requests to a PCA to generate certificates. Each PCA is associated with a pair of LAs (LA1 and LA2) that generate pre-linkage values for pseudonym certificates, which are used in EE revocation.
- The RA initiates requests to both LAs to obtain pre-linkage values
- The RA must respond to requests from the central MA to add EEs to its internal blacklist and to support misbehavior investigation

Not shown in Figure 1 is the association of the RA with one or more ECA. While there is no direct communication between an ECA and the RA, the RA must maintain a white list of ECA certificates such that only EEs with enrollment certificates signed by authorized ECAs can access the RA. In addition, the RA maintains extensive logs of transaction history and an internal blacklist, which identifies EEs that are disallowed to request or download new certificates.

Procedure

The addition of a new RA to the SCMS must begin with a certified RA component that has been setup according to the [Setup RA](#) use case.

The following actions are required to add the new RA:

- The MA must be updated with the Fully Qualified Domain Name (FQDN) of the new RA. This requires the local ICA Manager to inform the TCotSCMSM and request that the new RA be added to the MA.
- The RA must receive the FQDN of the PCA
- The RA must receive the FQDNs of both LAs and their LA IDs
- The RA must receive at least one ECA certificate, which will be added to the RA's white list of authorized ECAs

All of these steps are manual processes that are carried out by the local ICA Manager.

End State

After completing this use case, the RA will be configured with the following values:

RA Value	Notes
PCA FQDN	The RA must initiate communication with the PCA to request certificates.
LA1/2 FQDN	The RA requires the network address of LA1 and LA2.
LA1/2 ID	The RA requires the globally unique LA ID for LA1 and LA2.
ECA certificate	The RA must have a valid SCMS certificate from at least one active ECA which will configure EEs to contact the RA for certificates.

Table 33 RA Values

After completing this use case, the DCM will be configured with the following values:

DCM Value	Notes
RA FQDN	The DCM requires the network address of the RA that it is authorized to use when configuring new EEs.

Table 34 DCM Values

After completing this use case, the MA will be configured with the following values:

MA Value	Notes
RA FQDN	The MA must be able to contact the RA to update the RA's internal blacklist or to support misbehavior investigation.

Table 35 MA Values

Special Cases

The general procedure described above applies when adding a new RA to the SCMS. There are variations to the process when a replacement RA is introduced.

- If the RA certificate has been retired and the same RA now has a new certificate, the RA may continue to operate using the same network address and internal storage status. All DCMs that are authorized to use the RA shall obtain the new RA certificate for use in configuring new EEs.
- If the RA hardware were securely decommissioned, the internal memory of the prior RA may be transferred to a new device. As in the previous case, all DCMs that are authorized to configure EEs for the RA shall receive the new RA certificate.
- If the RA has been revoked and replaced, the local ICA Manager must decide if any pre-existing state information can be securely transferred to the replacement component.
- If a component in the RA's certificate chain (an ICA or the Root CA) is revoked and replaced, the RA will be implicitly revoked and need to be replaced. Here too, the local ICA manager may decide if any pre-linkage values from prior transactions can be saved. If not, then past values shall be purged.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-264	CLOSED	SCMS Notify RA Add	The TCotSCMSM shall inform the new RA of PCAs available to receive certificate requests, making available those PCAs' certificates and necessary information for locating them on the network.	The RA must be integrated correctly into the SCMS system.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-266	MANUAL PROCESS	DCM configure RA	The Technical Component of the SCMS Manager shall communicate the FQDN of RA to the DCM.	The RA must be integrated correctly into the SCMS system. Logical RA.	The relevant DCMs configure their end-entity devices to communicate with an RA to request pseudonym certificates. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-715	CLOSED	Provide Elector Certificates	The TCotSCMSM shall provision all SMCMS components with the self-signed certificates of all electors that are valid at the time of the component setup.	Root Management messages require signatures from the Electors to be validated, so authentic root CA Certificates are also required.	When receiving Root Management messages through the GCCF or LCCF, the authenticity of the messages will be validated by counting valid Elector signatures on the message and ensuring that at least that number (quorum) required in the Global Policy is present, after which the Root Management message can be processed.	TCotSCMSM
SCMS-770	MANUAL PROCESS	Create CSR	The to-be-added component shall create a CSR, which shall be forwarded to the authorizing root CA or ICA in order to obtain its SCMS identity certificate.	Most communications in the system are authenticated. A root CA or ICA must	In the PoC, this will occur by a manual process.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				authorize the new component.		
SCMS-774	CLOSED	Distribute the new component certificate	The TCotSCMSM shall forward signed certificates to the component that generated the corresponding CSR.	The result of the CSR is the new components identity in the system. This will be used to authenticate itself to other entities in the system.	After the authorizing root CA or ICA signs and returns a 1609.2 certificate to the TCotSCMSM, the new certificate must be delivered to the new component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1049	MANUAL PROCESS	Error code: tcNotifyRAofPCAListFailure	The TCotSCMSM shall log "Error code: tcNotifyRAofPCAListFailure", if the TCotSCMSM cannot notify the new RA of the list of available PCAs			TCotSCMSM
SCMS-1050	SCMS POC OUT OF SCOPE	Error code: tcNotifyDCMFailure	The TCotSCMSM shall log "Error code: tcNotifyDCMFailure", if it cannot notify the DCM of a newly added RA.			TCotSCMSM
SCMS-1386	MANUAL PROCESS	Add component's certificate to GCCF	The TCotSCMSM shall forward required information to the Policy Generator in order to add the certificate of the newly created SCMS component to the Global Certificate Chain File (GCCF).	For a newly added component to be a valid SCMS component, its certificate must chain back to the SCMS root CA and its chain must be available to any other component via GCCF.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-1412	MANUAL PROCESS	Destroy certificate's private key	The certificate's private key shall be destroyed at the end of the "In-use" life of a certificate. The in-use lifetime of certificates shall be defined either by SCMS policy and/or based on the expiration and In-use lifetime of subordinate certificates.	To prevent the usage of certificates that have reached the end of defined In-use lifetime.	Out of scope as this needs to be implemented as operational policy.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1581	SCMS POC OUT OF SCOPE	Component certificate in-use period	The SCMS component shall use its certificate for an in-use period of 3 years.	Use 3 years for standard SCMS components	Out of scope as this needs to be implemented as operational policy. This is for POC & CV-Pilot only.	CRLG, DCM, LA, MA, PG, RA
SCMS-1725	MANUAL PROCESS	Component certificate FQDN match	The SCMS component shall have a certificate with a certificateId field that matches the FQDN of the component.	FQDN of each component must match the official ID of the component.		CRLG, DCM, ECA, LA, MA, PCA, PG, RA
SCMS-1806	MANUAL PROCESS	Register non-central component with the central MA	The Local ICA Manager shall update the MA with the FQDN, TLS certificate, and SCMS certificate of any newly added PCA, LA, or RA.	The MA must know about all PCAs, LAs, and RAs in the system so that it can execute misbehavior investigations and revocation procedures.	In PoC, this will occur by a manual process. When a new PCA, LA, or RA is added, the local ICA manager will notify the TCotSCMSM about the newly added component (this is a manual process). The TCotSCMSM will then update the central MA with the necessary information about the newly added component. It is expected (but not required) that a PCA, RA, and pair of LAs will typically be added as a complete set.	TCotSCMSM

Table 36 Use Case 11.1.1 Add RA - Requirements

[12 issues](#)

Step 11.1.2: Add Root CA

Goals

The goal is to define the procedures and requirements to add and manage root CA certificates in the SCMS.

Background and Strategic Fit

The SCMS root CA is the root of trust for all SCMS certificates and digital signatures. The root CA private key is stored in a high-integrity component that is accessed through offline messages and that are managed by the Technical Component of the SCMS Manager (TCotSCMSM). Adding a new root CA to the SCMS and distributing the CA's certificate is necessary to maintain the integrity of the SCMS certificate hierarchy when a previous root CA certificate expires or when a previous root CA is revoked or securely decommissioned.

There shall be only one active root CA in the SCMS at any time. Specifically, it is the responsibility of the TCotSCMSM to ensure that only one root CA can be used to sign new messages. However, the design allows SCMS components to continue to trust certificates signed by previous root CAs until their certificates expire. This mechanism allows older root CAs to be retired (i.e., cease to be used for signing new certificates) without invalidating or revoking all of the component certificates that they signed in the past. This use case describes the mechanism for introducing a new root CA to all SCMS components.

Procedure

Before a new root CA can be added to the SCMS, it must first be setup using the process defined in the [Setup Root CA](#) use case. The new root CA must then be endorsed by a quorum of existing electors and the signed root endorsement must be distributed to all SCMS components. The message will be distributed through inclusion in the Global Certificate Chain File (GCCF) and any local copies (LCCFs) that are created and distributed by an RA.

To implement this process, an authorized agent of the SCMS manager will perform the following actions:

1. In a secure environment, command the root CA to create a self-signed certificate. See the [Setup Root CA](#) use case for details on the certificate parameters.
2. Present the root CA certificate to all existing, valid SCMS electors and request that they produce a digitally signed copy of the certificate. The collection of all independent signatures from existing electors is then assembled into one root endorsement message with the sequence of elector signatures attached (note that each elector's signature contains a copy of the original message that was signed). The number of elector signatures must be greater than or equal to the value of 'quorum' defined in the current GPF. This is a manual process to be implemented by the TCotSCMSM.
3. The complete root endorsement message with signatures is then delivered to the PG for inclusion in future updates to the GCCF. Note that the PG signature is not necessary for the root endorsement to be validated by SCMS components. The role of the PG in this case is to assemble updates to the GCCF with all active root endorsement messages included. The RAs will be required to include all root endorsement messages in any LCCF files that they derive from the GCCF.
4. SCMS components (including EEs) that receive a GCCF or LCCF with one or more root endorsement messages attached must check to see if they have already added the new root to their trust store. If they have not, they must validate the message by checking the attached signatures and confirming it has non-expired certificates for at least a 'quorum' of the existing electors that signed the message. Once the message is validated, the SCMS component must add the new root CA certificate to their trust store. When validating a message to add a root, an entity must check that the signed data is identical in each elector signature and that the 'type' element of the signed data has the value "addRoot."

Special Cases

The procedure described above is sufficient to establish a new root CA distribute its certificate to SCMS components. The following special considerations must be applied based on the reason for adding a new root.

- If the current root CA certificate is due to be retired, then the defined procedure is sufficient to distribute a new replacement root. The activation time for the new root should be set such that it does not overlap with the active life of the current root's useful life. The private key associated with the current root certificate shall be securely deleted or destroyed when the new root certificate becomes active. There is no need to remove the previous root certificate or to re-certify components.
- If the current root CA is to be securely decommissioned and replaced, the same procedure can be used as described for root certificate retirement. There is no need to remove the previous root certificate or to re-certify components.
- If the current root CA has been compromised or otherwise needs to be revoked, then the TCotSCMSM may follow the procedure described above with the activation time for the new root, set the current time, or the activation time, defined in the current root removal message. In addition, the TCotSCMSM must initiate the process of re-certifying all components in the SCMS with the new root CA. Specifically, the MA, CRGL, PG, and all ICAs that were certified with the previous root, must be re-certified. See the component "add" use cases for details on how to cascade the impact of re-certification throughout all other SCMS components.

Assumptions

- The SCMS Manager has the power to set policies for what conditions a new root CA must fulfill in order to be an accredited part of the system
- The root CA went through the setup process defined in [Step 1.8: Setup Root CA](#)
- Root Management is performed according to the elector scheme outlined in: [Elector-based Root Management](#)
- The Global Policy File (GPF) will define the current value for root management quorum, which is the minimum number of valid electors that need to endorse a root management message for it to be accepted by SCMS components. The value of quorum may be set independent of the current number of electors defined. (For the PoC, the value of quorum will be set to 2, meaning that a minimum of two elector signatures are needed for a root management command to take effect.)
- When the PG receives a valid "add root" message, it will continue to include that message on all future GCCF files that it produces until one of the following conditions occur:
 - The certificate of the root CA that is added in the message expires
 - The certificates of the endorsing electors expire resulting in fewer than 'quorum' valid signatures on the message
 - The value of 'quorum' is increased and distributed through an update to the GPF causing the "add elector" message to be invalid
 - The PG receives a valid "remove root" message that removes the endorsed root (effectively revoking the root that was added in the original message)
 - The PG receives a valid "remove elector" message that removes one of the endorsing electors reducing the number of valid electors to be less than the current value of quorum defined in the GPF, thereby rendering the message invalid

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-178	MANUAL PROCESS	Add a Root	The TCotSCMSM shall access each of the Electors to sign an "Add Root CA" message for the new Root CA.	New Root CAs must be authenticated with multiple signatures.	In the PoC, a manual process will produce the "Add Root CA" message. In the PoC, n is set to 3, and m is set to 2. The add root message will be distributed to all SCMS components and EEs as part of the GCCF/LCCF files.	TCotSCMSM
SCMS-1024	CLOSED	Root CA Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Root CA" and "Revoke Root CA" messages, which will be signed by Electors, and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage root CA updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1055	EE REQUIREMENT	EE verify "Add Root CA" message	The EE shall add the new root CA certificate to its trust store only after verifying the validity of the "Add Root CA" message. The validation of this message shall be carried out securely in the EE's secure execution environment or HSM.	A quorum of Electors must authorize a new root CA	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1094	CLOSED	Verify "Add Root CA" message	All SCMS Backend Components shall add the new root CA certificate to their trust stores only after verifying the validity of the "Add Root CA" message. The validation of this message shall be carried out securely in the component's HSM.	A quorum of Electors must authorize a new root CA		CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1200	MANUAL PROCESS	Distribute "Add Root CA" messages	The Technical Component of the SCMS Manager (TCotSCMSM) shall communicate the multi-signed	SCMS components and EEs need to be aware of a newly added root CA. They get this information through an update		TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			"Add Root CA" message to the Policy Generator to be included in a new Global Certificate Chain File (GCCF) which will be distributed to SCMS components and EEs to inform them of the new root CA.	to the Global Certificate Chain File (GCCF), respectively, the Local Certificate Chain File (LCCF) which contains a section for trust management messages (add a root or elector). The generation and distribution to the PG of this message is done manually whereas the distribution to other SCMS components and EEs is done automatically via GCCF/LCCF available at the RA.		
SCMS-1318	CLOSED	Root CA certificate validity	The root CA certificate validity period shall be set to 17 years.	Root CA certificates must have an expiration date. The root CA certificate must be valid at least as long as the longest issued enrollment certificate.	Certificate types and expiration periods are defined in the Certificate Types common requirements section. This is for PoC only.	RCA
SCMS-1332	MANUAL PROCESS	Root CA certificate overlap	Root CA certificates shall have an overlap of 9 years (an in-use period of 8 years).	The overlap is necessary to allow rollover.	This is for POC & CV-Pilot only.	RCA
SCMS-1409	CLOSED	Elector Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Elector" and "Revoke Elector" messages, which will be signed by Electors and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage Elector updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1422	SCMS POC OUT OF SCOPE	Renewal of component certificate	A SCMS component shall request rollover IEEE 1609.2 certificates no sooner than 3 months prior to the end of the In-use life of the current certificate. A SCMS component shall not issue rollover IEEE 1609.2 certificates prior 3	To prevent the existence of certificates that are not valid until a significant time in the future.	Does not apply to component compromise/revoked situations. For the PoC & CV-Pilot, 3 months is being used. This should be re-evaluated for other deployments.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA, RCA

Key	Status	Summary	Description	Justification	Notes	Component/s
			months to the end of the In-use life of the current certificate.			

Table 37 Use Case 11.1.2 Add Root CA - Requirements

[9 issues](#)

Design

The detailed design for the elector-based root management process is described in the [Elector-based Root Management](#) section.

Diagrams

Create Replacement Root CA & Distribute to SCMS Servers

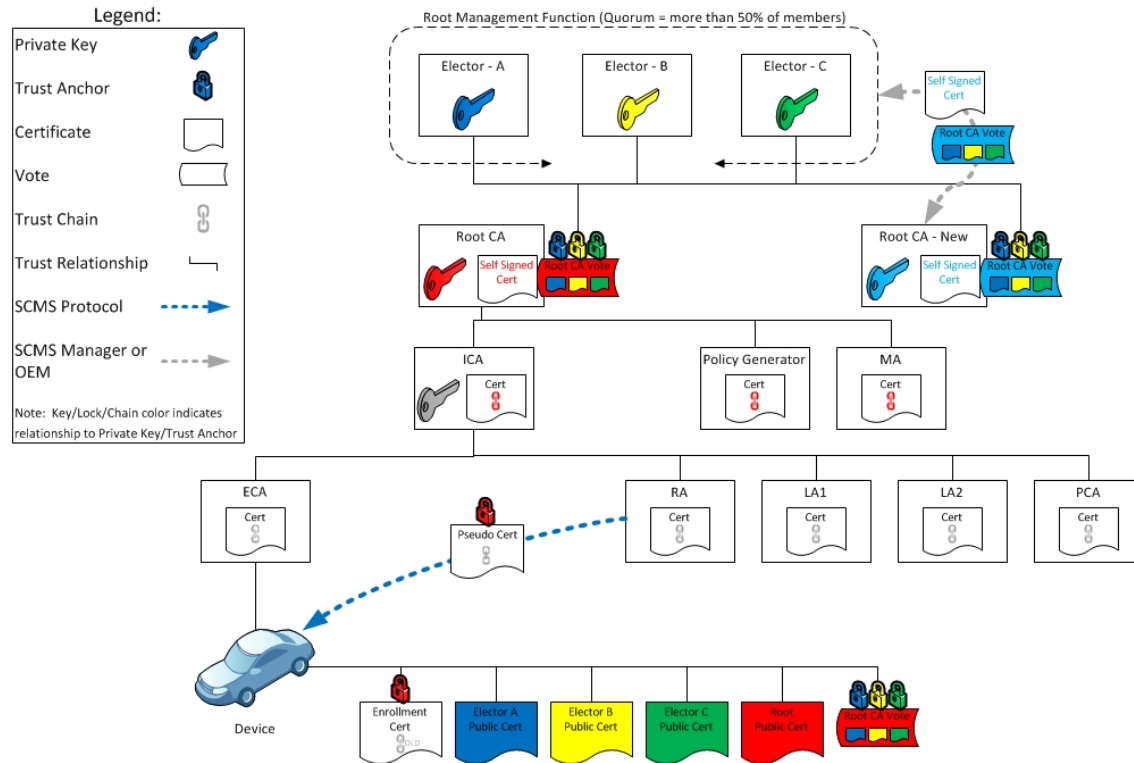


Figure 52 Create Replacement Root CA & Distribute to SCMS Servers

Introduce Replacement Root CA Before Revoking Current Root CA

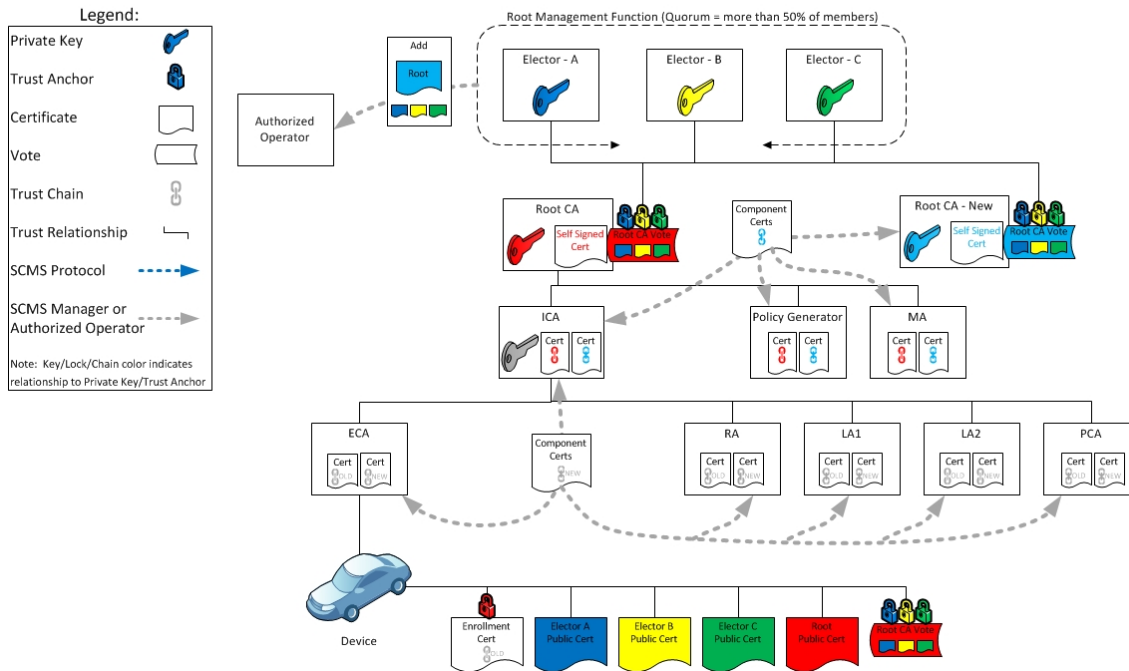


Figure 53 Introduce Replacement Root CA Before Revoking Current Root CA

Step 11.1.3: Add Elector

Goals

The goal is to define the procedures and requirements to add and manage root management electors in the SCMS.

Background and Strategic Fit

Electors are a collection of highly trusted backend components in the SCMS, which are used to certify root management messages. Specifically, a message that commands all SCMS components to add or remove a root CA certificate from their trust store will be trusted only if it is signed by a quorum of electors. The value of quorum is defined in the Global Policy File (GPF). Root management messages are distributed as part of the Global Certificate Chain File (GCCF) or a local copy of the chain file (an LCCF).

Procedure

Before a new elector can be added to the SCMS, it must first be setup using the process defined in the [Setup Elector](#) use case. The new elector must then be endorsed by a quorum of existing electors and the signed "add elector" message must be distributed to all SCMS components. The message will be distributed through inclusion in the Global Certificate Chain File (GCCF) and any local copies (LCCFs) that are created and distributed by an RA.

To implement this process, an authorized agent of the SCMS manager will perform the following actions:

1. Command the new elector to create a self-signed certificate of the new elector (the elector certificate is created in the [Setup Elector](#) use case) in a secure environment.
2. Present the new elector certificate to all existing, valid SCMS electors and request that they produce a digitally signed copy of the certificate. The collection of all independent signatures from existing electors is then assembled into one elector endorsement message with the sequence of existing

elector signatures attached. The number of elector signatures must be greater than or equal to the value of 'quorum' defined in the current GPF. This is a manual process to be implemented by the TCotSCMSM.

3. Deliver the complete elector endorsement message with signatures to the PG for inclusion in future updates to the GCCF. Note that the PG signature is not necessary for the elector endorsement to be validated by SCMS components. The role of the PG in this case is to assemble updates to the GCCF with all active elector endorsement messages included. RAs will be required to include all elector endorsement messages in any LCCF files that they derive from the GCCF.

SCMS components (including EEs) that receive a GCCF or LCCF with one or more elector endorsement message attached must validate the message by checking the attached signatures and confirming it has non-expired certificates for at least a 'quorum' of the existing electors that signed the message. Once the message is validated, the SCMS component must add the new elector certificate to their trust store. When validating an elector endorsement message, an entity must check that the signed data is identical in each elector signature and that the ballotType element of the signed data has the value "addElector."

Assumptions

- An initial set of electors and self-signed elector certificates will be created as part of ceremony (or sequence of ceremonies) at the launch of an SCMS infrastructure. The SCMS Manager will define policies and procedures to ensure the integrity of this initial set of electors.
- Once an SCMS is in operation and the initial set of electors has been installed in all existing SCMS components, new electors may be added using the process defined here.
- The existing electors that sign an "add elector" message must have valid, non-expired SCMS certificates at the time when they sign the message. SCMS components that process an "add elector" message must confirm that the endorsing elector certificates are not expired at the time when the message is being processed. Once the message is validated, the SCMS components will add the new elector to their trust store and it will remain there even if one or more of the endorsing elector certificates expire. As long as that expiration happens after the message was validated and processed, the new elector remains trusted.
- When the PG receives a valid, "add elector" message, it will continue to include that message on all future GCCF files that it produces until one of the following conditions occur:
 - The certificate of the elector that is added in the message expires
 - The certificates of the endorsing electors expire resulting in fewer than 'quorum' valid signatures on the message
 - The value of 'quorum' is increased and distributed through an update to the GPF causing the "add elector" message to be invalid
 - The PG receives a "remove elector" message that removes the endorsed elector or removes the endorsing electors rendering the message invalid

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1024	CLOSED	Root CA Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Root CA" and "Revoke Root CA" messages, which will be signed by Electors, and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage root CA updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1380	MANUAL PROCESS	Distribute "Add Elector" message	The Technical Component of the SCMS Manager (TCotSCMSM) shall communicate the multi-signed "Add Elector" message to the Policy Generator to be included in a new Global Certificate Chain File (GCCF), which will be distributed to SCMS components and EEs to inform them of the new Elector.	SCMS components and EEs need to be aware of a newly added Elector. They get this information through an update to the Global Certificate Chain File (GCCF), respectively, the Local Certificate Chain File (LCCF), which contains a section for trust management messages (add a root or elector). The generation and distribution to the PG of this message is done manually whereas the distribution to other SCMS components and EEs is done automatically via GCCF/LCCF.		TCotSCMSM
SCMS-1382	MANUAL PROCESS	Add an Elector	The TCotSCMSM, in cooperation with the new and existing Electors, shall produce an "Add Elector" message for the new Elector.	New Electors must be authenticated with signatures of a quorum of non-revoked Electors.	In the PoC, this will message will be produced by a manual process. In the PoC, the number of electors is 3, and the quorum is set to 2.	TCotSCMSM
SCMS-1384	EE REQUIREMENT	EE verify "Add Elector" message	The EE shall add the new Elector certificate to its trust store only after verifying the validity of the "Add Elector" message. The validation of this message shall be carried out securely in the EE's	A quorum of electors must authorize a new elector	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
			secure execution environment or HSM.			
SCMS-1385	CLOSED	Verify "Add Elector" message	All SCMS Backend Components shall add the new Elector certificate to their trust stores only after verifying the validity of the "Add Elector" message. The validation of this message shall be carried out securely in the component's HSM.	A quorum of Electors must authorize a new Elector		CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1409	CLOSED	Elector Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Elector" and "Revoke Elector" messages, which will be signed by Electors and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage Elector updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1414	MANUAL PROCESS	Added Elector endorses current Root CAs	The added Elector shall endorse all current root CAs by signing the existing "Add root CA" message.	To avoid a situation where a revoked Elector would enforce the revocation of an existing and valid root CA.		Elector
SCMS-1422	SCMS POC OUT OF SCOPE	Renewal of component certificate	A SCMS component shall request rollover IEEE 1609.2 certificates no sooner than 3 months prior to the end of the In-use life of the current certificate. A SCMS component shall not issue rollover IEEE 1609.2 certificates prior 3 months to the end of the In-use life of the current certificate.	To prevent the existence of certificates that are not valid until a significant time in the future.	Does not apply to component compromise/revoked situations. For the PoC & CV-Pilot, 3 months is being used. This should be re-evaluated for other deployments.	CRL Store, CRLG, DCM, ECA, ICA, LA, MA, PCA, PG, RA, RCA
SCMS-1423	MANUAL PROCESS	Elector Certificate Expiration	The Technical Component of the SCMS Manager (TCotSCMSM) shall issue Elector certificates with an expiration of 12 years.	Component 1609 certificates shall have a defined expiration.	In the case of the certificate being revoked, the new certificate may have a different expiration to align with predefined replacement schedules (if any exist).	Elector

Key	Status	Summary	Description	Justification	Notes	Component/s
					For the initial system deployment, 1 of the 3 Electors shall have a certificate expiration of 4 years, another one a certificate expiration of 8 years, to prevent multiple elector certificates from expiring at the same time. These durations are for the SCMS PoC and CV-Pilot only. For other SCMS instances, this duration should be reevaluated.	
SCMS-1590	SCMS POC OUT OF SCOPE	Elector Certificate In-Use period	The Elector certificate In-Use period shall be the same as the Expiration period.	Out of scope as this needs to be implemented as operational policy. To maintain a fixed number of valid Elector at all times.		Elector
SCMS-1809	CLOSED	Elector certificate validity	Elector certificates validity period shall be set to 12 years.	Elector certificates must have an expiration date.	Certificate types and expiration periods are defined in the Certificate Types common requirements section. This is for PoC and CV-Pilot only.	Elector

Table 38 Use Case 11.1.3: Add Elector - Requirements

[11 issues](#)

Design

The detailed design for the elector-based, root management process is described in the [Elector-based Root Management](#) section.

Step 11.2: Revoke SCMS Component

Goals

Revoke procedure for each SCMS component.

Assumptions

- As the SCMS system evolves, it will become necessary that components be revoked in the case of compromise or obsolescence
- Actual requirements will be outlined in subsections

CRL Series

This is the CRL series diagram for POC / Pilot Deployments.

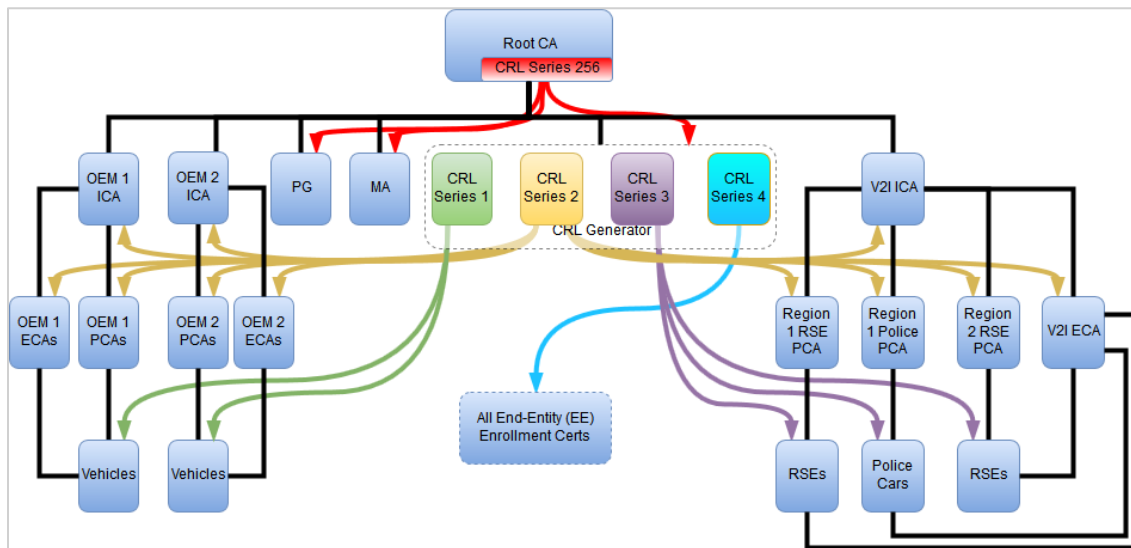


Figure 54 CRL Series Diagram

Step 11.2.1: Revoke non-Root SCMS Component

Goals

- Provide a mechanism for revoking an SCMS component other than root CA.
- Define procedures to enable continued SCMS operations after the revocation.

Assumptions

- The technical component of the SCMS Manager will coordinate the revocation of SCMS components to enable continued operations.
- When an SCMS component is revoked, the function that it provided will be taken over by a peer device with sufficient privileges and capabilities to continue operations. The peer device may be a pre-existing device that is taking on additional work or a newly added (i.e., replacement) device.
- This use case specifies the requirements that are common among all non-root SCMS component revocations. Individual use cases will provide specific details that are unique to each component type.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-314	CLOSED	Revocation of a Non-Root SCMS component	Root CA and CRL Generator (CRLG) shall revoke non-root SCMS components.	SCMS components that are compromised should not be trusted for operation and are revoked starting at time T, not simply removed.	Revocation of a component at a time T dictates that from this time onward all certificate chains that chain back to this component are to not be trusted. Root CA revokes PG, MA, and CRLG. CRLG revokes ICA, PCA, RA, and EE. All CRLs are available by CRLG, and CRLG shall provide the SCMS components CRL upon request.	CRLG, RCA
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA. All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> In sending requests to that component 	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					<ul style="list-style-type: none"> In trusting certificate chains chaining to that component's certificate In trusting messages signed using this component's certificate 	
SCMS-1379	CLOSED	Provide x.509 SCMS component OSCP stapling	The Technical Component of the SCMS shall provide an OSCP service that supports stapling for x.509 SCMS component certificates with a validity of 24 hours.	to ensure that SCMS components do not establish communication to other SCMS components that are revoked.		TCotSCMSM
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM
SCMS-1598	CLOSED	Regenerate CRL	MA shall regenerate the CRL from internal data, if the CRLG is revoked.	Because the old CRL cannot be trusted anymore if CRLG is revoked.	Older versions of the CRL signed by the revoked CRLG would be discarded.	MA
SCMS-2461	CLOSED	SCMS components regularly pull SCMS component CRL	The SCMS component shall download the SCMS component CRL from CRLG regularly, at an interval of at least every 60 minutes	To ensure that revoked SCMS components are excluded from the system within a short time period.		DCM, ECA, LA, MA, PCA, PG, RA

[7 issues](#)

Step 11.2.1 - Revoke CRLG

Goals

The goal is to revoke a CRLG certificate from the SCMS System.

Background and Strategic Fit

A CRL Generator (CRLG) can only be revoked by a root CA. In a situation where a CRLG has been compromised or has failed, the TCotSCMSM must activate a root CA and use it to sign a Series 256 CRL listing the compromised CRLG as revoked. This file must then be copied to the CRL Store for distribution to all components.

On receipt of a CRL signed by a root CA and listing a CRLG as revoked, the CRL Store must create a new composite CRL that contains:

1. All non-expired, Elector-signed, root-management messages
2. The new root CA signed CRL listing a CRLG as revoked
3. Any other non-expired root CA signed CRLs
4. Any CRLs signed by other, no-revoked, non-expired CRLGs

This new composite CRL shall be distributed to all components.

The MA shall no longer the use revoked CRLG to sign CRLs ([the procedure for adding a new CRLG](#) will update the MA with the address and TLS certificate of the new CRLG).

Assumptions

- In the SCMS design, there may be more than one CRLG. However, for the Proof of Concept (PoC) deployment, there will be a single, central CRLG.
- The SCMS requires a valid CRLG in order to sustain operation. If the only active CRLG is revoked, the TCotSCMSM must initiate the process of adding a new CRLG (or re-certifying the existing CRLG) using the procedure described in the [Add CRLG](#) use case.
- After receipt of the new CRL signed by a root CA listing a CRLG as revoked, all components and EE shall cease to process any CRL signed by the revoked CRLG.
- Components will have no reliable way to know the sequence in which valid or fraudulent revocation messages were created. Therefore, there is no effective way to "un-revoke" components previously placed on the CRL by a compromised CRLG. All previously revoked components will need to be re-certified with new certificates in order to restore trust.
- The procedure for interacting with the CRL Store and assembling a new composite CRL is implementation specific. There are no standard SCMS messages or procedures for performing this function.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	<p>Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA.</p> <p>In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA.</p> <p>All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1598	CLOSED	Regenerate CRL	MA shall regenerate the CRL from internal data, if the CRLG is revoked.	Because the old CRL cannot be trusted anymore if CRLG is revoked.	Older versions of the CRL signed by the revoked CRLG would be discarded.	MA
SCMS-1606	EE REQUIREMENT	EE shall store ValidityPeriod.start of last valid CRLG Certificate	The EE shall store the ValidityPeriod.start value of the last CRLG Certificate that passes validation.	In order to prevent the following attack sequence: 1) A CRLG Certificate is compromised by attacker 2) A new valid CRLG Certificate is used to sign a CRL revoking the compromised CRLG certificate 3) The CRL Store makes the new valid CRL available for download 4) The attacker downloads the new valid CRL 5) Attacker creates a fraudulent CRL signed by the compromised certificate which revokes the new CRLG certificate 6) Attacker distributes the new fraudulent CRL via collaborative distribution before all devices have downloaded the new valid CRL 7) Repeat steps 2-6		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1607	EE REQUIREMENT	EE shall check CRLG Certificate Validity.start time	Upon receiving a new CRL, the EE shall reject the CRL and CRLG Certificate if the ValidityPeriod.start value of the CRLG certificate used to sign the newly received CRL is chronologically earlier than the stored ValidityPeriod.start value of	In order to prevent the following attack sequence: 1) A CRLG Certificate is compromised by attacker 2) A new valid CRLG Certificate is used to		On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
			the previously received valid CRLG Certificate.	sign a CRL revoking the compromised CRLG certificate 3) The CRL Store makes the new valid CRL available for download 4) The attacker downloads the new valid CRL 5) Attacker creates a fraudulent CRL signed by the compromised certificate which revokes the new CRLG certificate 6) Attacker distributes the new fraudulent CRL via collaborative distribution before all devices have downloaded the new valid CRL 7) Repeat steps 2-6		

Table 39 Use Case 11.2.1 Revoke CRLG - Requirements

[5 issues](#)

Step 11.2.1 - Revoke ECA

Goals

The goal is to revoke an ECA certificate from the SCMS System.

Background and Strategic Fit

The technical component of the SCMS Manager (or a local ICA Manager in cooperation with the TCotSCMSM) determines that an Enrollment Certificate Authority (ECA) needs to be revoked. It contacts the CRLG and instructs it to add the ECA certificate to the CRL.

All components and entities that receive the updated CRL will cease to trust any enrollment certificate issued by the ECA and stop communicating with the ECA. All end-entity devices whose enrollment certificate chains back to the revoked ECA should obtain a new enrollment certificate as soon as possible (the SCMS Manager may set performance requirements for how quickly this must happen).

Procedure

- The local ICA Manager responsible for the revoked ECA must contact all DCMs that are configured to use the revoked component and remove it from their list of trusted ECAs for use in generating enrollment certificates. The ICA manager might reconfigure the DCMs to use a different ECA or stand up a new ECA following the procedures defined in the [Add ECA](#) use case.
- The ICA manager must also inform the RA that has the impacted ECA in its list of trusted ECAs and inform it to remove the revoked component. The RA will cease to pre-generate pseudonym certificates for any EE enrolled by that ECA and cease to accept any new requests from EEs certified by that ECA.
- EEs must have a proprietary mechanism to re-enroll in order to recover from the revocation of the ECA that signed their enrollment certificate. Once they are re-enrolled and associated with an RA, each impacted EE will have to request new pseudonym, application, or identification certificates.

Assumptions

- Authorized managers of EEs must provide a trusted (and certified by an agent of the SCMS Manager) method for re-enrolling EEs under their jurisdiction that are impacted by a revoked ECA
- A compromised DCM will require that all ECAs that were used with that DCM shall be revoked. All local ICA Managers will be required to record which ECAs were used in issuing enrollment certificates for every DCM.
- The procedure requires that all DCMs provide a proprietary mechanism (i.e., there are no SCMS messages defined for this step) to remove a revoked ECA from the list of ECAs that they use for enrolling new EEs. Note that a DCM should remove by default an ECA from the list of components that they use upon receipt of the updated CRL listing the ECA as revoked. However, the proprietary mechanism described in the use case assumes that ICA Managers will want a mechanism to remove pro-actively a revoked ECA.
- The procedure requires that all RAs provide a proprietary mechanism (i.e., there are no SCMS messages defined for this step) to remove a revoked ECA from the list of ECAs whose enrollment certificates they will trust. All RAs shall remove by default an ECA from the list of components that they trust as soon as they receive the updated CRL listing the ECA as revoked. However, the proprietary mechanism described in the use case assumes that ICA Managers will want a mechanism to remove pro-actively a revoked ECA.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			component shall be removed.		<p>detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA. All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			revoked component and all its certificate chains from the Global Certificate Chain File.			
SCMS-1587	EE REQUIREMENT	EE shall cease to trust the revoked CA	EEs receiving and validating a CRL shall remove all revoked CA certificates from their trust store, remove all cached certificate chains that roll up to the revoked CA, and stop sending immediately in case EE's enrollment, pseudonym, application, or identification certificate chains up to the revoked CA.	EE should not use the revoked component's certificate to trust it. If it chains include the revoked component, they need to receive new certificates with a new certificate chain.	EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> In sending requests to that component In trusting certificate chains chaining to that component's certificate In trusting messages signed using this component's certificate In sending messages signed with certificates that chain up to this component's certificate This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1589	EE REQUIREMENT	EE receive new enrollment certificate after CA revocation	EE shall get back to the secure	EE should not use the revoked CA's certificate and all certificates that chain		On-board Equipment

Key	Status	Summary	Description	Justification	Notes	Component/s
			environment used during their bootstrapping process and be re-bootstrapped after its RCA, ICA or ECA was revoked.	up to that CA certificate to trust it or to use it in communication. That includes enrollment certificates that chain up to the revoked CA certificate.		(OBE), Roadside Equipment (RSE)
SCMS-1593	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after CA revocation	EE shall request new pseudonym, application, or identification certificates after it was re-bootstrapped due to revocation of its RCA, ICA, or ECA.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.		On-board Equipment (OBE), Roadside Equipment (RSE)

Table 40 Use Case 11.2.1 Revoke ECA - Requirements

[7 issues](#)

Step 11.2.1 - Revoke ICA

Goals

The goal is to revoke an ICA certificate from the SCMS System.

Background and Strategic Fit

The Technical Component of the SCMS Manager (TCotSCMSM), a local ICA Manager, or the Misbehavior Authority, determines that an Intermediate Certificate Authority (ICA) needs to be revoked. The TCotSCMSM contacts the appropriate CRLG (as indicated in the ICA certificate, see the [CRL Series Diagram](#) for details) and adds the impacted ICA to the CRL. On receiving and validating the new CRL, all components will cease to trust the ICA and any certificates that chain back to the ICA.

Impacted components may include ECA, RA, PCA, LA and any EEs enrolled through an impacted ECA. All end-entity devices (EE) whose enrollment or application certificates chain back to the revoked ICA should obtain new enrollment or application certificates as soon as possible (the SCMS Manager may set performance requirements for how quickly this must happen). The SCMS will provide re-enrollment processes at a later stage.

All EEs whose pseudonym, application, or identification certificates chain back to the impacted ICA will cease to use those certificates. They shall request new certificates.

The TCotSCMSM will inform the Policy Generator (PG) to update the GCCF and remove all component certificates that chain back to the revoked ICA. The new GCCF will be distributed to all un-revoked RAs, which will incorporate the new lists in the next LCCF that they issue.

Assumptions

- The local ICA Manager will coordinate with the TCotSCMSM when revoking an ICA
- If the MA determines that an ICA shall be revoked, it will notify the TCotSCMSM. This will not be an automated process.
- The TCotSCMSM will inform the local ICA Manager when revoking an ICA

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			component shall be removed.		<p>detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA. All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			revoked component and all its certificate chains from the Global Certificate Chain File.			
SCMS-1587	EE REQUIREMENT	EE shall cease to trust the revoked CA	EEs receiving and validating a CRL shall remove all revoked CA certificates from their trust store, remove all cached certificate chains that roll up to the revoked CA, and stop sending immediately in case EE's enrollment, pseudonym, application, or identification certificate chains up to the revoked CA.	EE should not use the revoked component's certificate to trust it. If it chains include the revoked component, they need to receive new certificates with a new certificate chain.	EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate • In sending messages signed with certificates that chain up to this component's certificate This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1589	EE REQUIREMENT	EE receive new enrollment certificate after CA revocation	EE shall get back to the secure	EE should not use the revoked CA's certificate and all certificates that chain		On-board Equipment

Key	Status	Summary	Description	Justification	Notes	Component/s
			environment used during their bootstrapping process and be re-bootstrapped after its RCA, ICA or ECA was revoked.	up to that CA certificate to trust it or to use it in communication. That includes enrollment certificates that chain up to the revoked CA certificate.		(OBE), Road-side Equipment (RSE)
SCMS-1593	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after CA revocation	EE shall request new pseudonym, application, or identification certificates after it was re-bootstrapped due to revocation of its RCA, ICA, or ECA.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1608	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after PCA revocation	EE shall request new pseudonym, application, or identification certificates whenever it's certificates chain up to a PCA certificate that is invalidated due to a RCA, ICA, or PCA revocation.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.	EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					<ul style="list-style-type: none"> In sending messages signed with certificates that chain up to this component's certificate <p>This is out of scope as it defines EE behavior.</p>	

Table 41 Use Case 11.2.1 Revoke ICA - Requirements

[8 issues](#)

Step 11.2.1 - Revoke MA

Goals

The goal is to revoke an MA certificate from the SCMS System.

Background and Strategic Fit

The Technical Component of the SCMS Manager (TCotSCMSM) determines that the Misbehavior Authority (MA) needs to be revoked. It will request that the CRLG add the MA certificate to CRL and immediately issue a new CRL. The CRLG will cease to accept new requests signed by the MA. On receipt of the new CRL, all PCAs, RAs, and LAs will cease to accept new requests signed by the revoked MA.

The TCotSCMSM activates the replacement MA as described in [Step 11.1.1 - Add MA](#).

Assumptions

The TCotSCMSM will recover information on any active investigations underway when the MA was revoked. Trusted data will be copied to a replacement MA and those investigations will continue.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA. All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> In sending requests to that component 	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					<ul style="list-style-type: none"> In trusting certificate chains chaining to that component's certificate In trusting messages signed using this component's certificate 	
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM

Table 42 Use Case 11.2.1 Revoke MA - Requirements

[4 issues](#)

Step 11.2.1 - Revoke PCA

Goals

The goal is to revoke a PCA certificate from the SCMS System.

Background and Strategic Fit

The Technical Component of the SCMS Manager (or a local ICA Manager) determines that a Pseudonym Certificate Authority (PCA) needs to be revoked.

Procedure

- The TCotSCMSM contacts the CRLG and adds the certificate of the impacted PCA to the CRL. On receipt of the new CRL, all components will cease to trust pseudonym certificates issued by the PCA.
- The local ICA Manager will contact any RA that was configured to use the impacted PCA and instruct it to send new pseudonym certificate requests to a different PCA or it will stand up a new PCA (see the [Add PCA](#) use case).
- The LAs that share a secret key with the impacted PCA will delete the shared key and await configuration information from the local ICA Manager to establish a key with a new PCA.
- All end-entity devices whose pseudonym certificates were signed by the revoked PCA should obtain a new batch of pseudonym certificates as soon as possible (the SCMS Manager may set performance requirements for how quickly this must happen). If they have certificates from other non-revoked PCAs for the current time period, they may continue to operate using those certificates until a replacement batch can be downloaded.

Assumptions

- All RAs will destroy any stored batches of pseudonym certificates proactively generated by the impacted PCA
- Any misbehavior investigations that relied on the PCA will be stopped

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-358	EE REQUIREMENT	Discard Certificate Batches Signed by a Revoked PCA	OBE shall discard all pseudonym certificates that were issued by a PCA upon validating that this PCA has been revoked.	PCA generates batches of pseudonym certificates and an OBE device cannot know when the pseudonym certificates were signed, therefore all such certificates from the revoked PCA must be untrusted even if the PCA's certificate was verified previously.	This is out of scope as it defines OBE behavior.	On-board Equipment (OBE)
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
			<p>component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.</p>		<p>encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA.</p> <p>In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA.</p> <p>All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM
SCMS-1587	EE REQUIREMENT	EE shall cease to trust the revoked CA	EEs receiving and validating a CRL shall remove all revoked CA certificates from their trust store, remove all cached certificate chains that roll up to the revoked CA, and stop sending immediately in case EE's enrollment, pseudonym, application, or identification certificate chains up to the revoked CA.	EE should not use the revoked component's certificate to trust it. If it chains include the revoked component, they need to receive new certificates with a new certificate chain.	EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate • In sending messages signed with certificates that chain up to this 	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					<p>component's certificate</p> <p>This is out of scope as it defines EE behavior.</p>	
SCMS-1608	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after PCA revocation	EE shall request new pseudonym, application, or identification certificates whenever it's certificates chain up to a PCA certificate that is invalidated due to a RCA, ICA, or PCA revocation.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.	<p>EES receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate • In sending messages signed with certificates that chain up to this component's certificate <p>This is out of scope as it defines EE behavior.</p>	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2608	MANUAL PROCESS	Map PCA IssuerIdentifier to PCA FQDN	The TCotSCMSM shall associate	A mapping between PCAs' IssuerIdentifier and PCA hostname is	For the PoC, the SCMS operator will	TCotSCMSM

Key	Status	Summary	Description	Justification	Notes	Component/s
			each PCA IssuerIdentifier (the HashId8 of the PCA signing certificate) with the FQDN of the PCA that has the certificate.	needed. During an investigation, the MA (GMBD) will receive a cert, extract linkage value, extract IssuerIdentifier, and then ask the PCA that issued the cert for the linkage value. For that step, MA must be able to map the IssuerIdentifier to the PCA hostname. This mapping is maintained by the TCotSCMSM and configured in the MA as needed, or this mapping may be maintained as a built-in feature of the MA.	manually configure the mapping of PCA IssuerIdentifiers with the FQDN of the corresponding PCA.	

Table 43 Use Case 11.2. Revoke PCA - Requirements

[8 issues](#)

Step 11.2.1 - Revoke PG

Goals

Revoke a Policy Generator certificate from the SCMS System.

Background and Strategic Fit

The Technical Component of the SCMS Manager (TCotSCMSM) determines that a Policy Generator (PG) needs to be revoked. The TCotSCMSM access the CRL generator for CRL series 256 (either the root CA or a central CRLG) and causes the PG to be added to the composite CRL, which is made available to all SCMS components and EEs. On receipt of the new CRL, all SCMS components and EEs shall mark the affected Policy Generator as untrusted. Components and EEs must request a new policy file signed by a new PG as soon as it is available.

Procedure

1. The TCotSCMSM contacts the series 256 CRL generator (see the [CRL Series Diagram](#) for details) and instructs it to add the current PG certificate to the CRL. The CRLG assembles and signs an updated CRL, which is made available to all components and EEs through the CRL store or via collaborative distribution.
2. The TCotSCMSM shall configure a new PG and issue a new GPF as described in the [Add PG](#) use case. The GPF will be made available to all RAs and back-end components. On receipt of the new GPF, each RA will assemble an updated LPF and submit the custom portion of the local policy to be signed by the new PG.
3. Upon receipt of the updated CRL, all SCMS components and EEs shall cease to trust the current policy or any new policy files signed by the revoked PG. They shall all resort to a set of pre-configured "default" policy values and attempt to download an updated policy file signed by a new PG as soon as it is available.
4. EEs shall contact their RA to download a new policy file signed by a replacement PG. They shall switch to a pre-defined set of "default" policy values until the new file is available.
5. SCMS components shall attempt to download a new policy file signed by a replacement PG. They will use a pre-defined set of "default" policy values until the new file is available.

Assumptions

- Backend components and EEs will be pre-programmed with a set of "default" policy values that can maintain some level of system operation while the new PG is established and new policy files are distributed.
- Each RA will need to receive the new GPF when it is available, assemble their own custom section of their LPF and submit it to the PG to be signed. Local ICA Managers will implement a manual process to push the new GPF out to their RAs. The TCotSCMSM may implement network management practices to limit traffic to the replacement PG.
- There may be a time delay before a new policy file is available from an RA for EEs to download. OEMs shall define an implementation specific mechanism to manage EE messaging to the RA. OEMs that have alternate mechanisms to push content out to their EEs may use these mechanisms to distribute a new signed policy file as soon as it is available.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA. All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> In sending requests to that component 	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s																				
					<ul style="list-style-type: none"> In trusting certificate chains chaining to that component's certificate In trusting messages signed using this component's certificate 																					
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM																				
SCMS-1685	EE REQUIREMENT	EEs shall use default policy when PG is revoked	<p>EEs shall switch to the following set of pre-defined default policy values upon receipt of a CRL that revokes the Policy Generator (PG) that signed the most recently accepted policy update.</p> <table border="1"> <thead> <tr> <th>Identifier</th> <th>Poc Default Value</th> </tr> </thead> <tbody> <tr> <td>scms_version</td> <td>1</td> </tr> <tr> <td>global_cert_chain_file_id</td> <td>2 bytes</td> </tr> <tr> <td>overdue_CRL_tolerance</td> <td>2 weeks</td> </tr> <tr> <td>(OBE only) i_period</td> <td>1 week</td> </tr> <tr> <td>(OBE only) min_certs_per_i_period</td> <td>20</td> </tr> <tr> <td>(OBE only) cert_validity_model</td> <td>concurrent</td> </tr> <tr> <td>(OBE only) max_available_cert_supply</td> <td>3 years</td> </tr> <tr> <td>(RSE only) rse_application_cert_validity</td> <td>1 week + 1 hour</td> </tr> <tr> <td>(RSE only) rse_application_cert_overlap</td> <td>1 hour</td> </tr> </tbody> </table>	Identifier	Poc Default Value	scms_version	1	global_cert_chain_file_id	2 bytes	overdue_CRL_tolerance	2 weeks	(OBE only) i_period	1 week	(OBE only) min_certs_per_i_period	20	(OBE only) cert_validity_model	concurrent	(OBE only) max_available_cert_supply	3 years	(RSE only) rse_application_cert_validity	1 week + 1 hour	(RSE only) rse_application_cert_overlap	1 hour	When the current PG is revoked, EEs can no longer trust the currently active policy values. Rather than operate with potentially invalid values, they shall switch to a set of pre-programmed default values that are deemed suitable to maintain safe operation.	This requires that EE software contain default values, which will be used when the current PG is revoked. It also implies that each EE keep track of the identity of the PG that signed the most recent policy update that the EE accepted. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
Identifier	Poc Default Value																									
scms_version	1																									
global_cert_chain_file_id	2 bytes																									
overdue_CRL_tolerance	2 weeks																									
(OBE only) i_period	1 week																									
(OBE only) min_certs_per_i_period	20																									
(OBE only) cert_validity_model	concurrent																									
(OBE only) max_available_cert_supply	3 years																									
(RSE only) rse_application_cert_validity	1 week + 1 hour																									
(RSE only) rse_application_cert_overlap	1 hour																									

Table 44 Use Case 11.2.1 Revoke PG - Requirements

[5 issues](#)

Step 11.2.1 - Revoke RA

Goals

Revoke an RA certificate from the SCMS System.

Background and Strategic Fit

The Technical Component of the SCMS Manager (TCotSCMSM) determines that an Registration Authority (RA) needs to be revoked, generates a certificate revocation message listing the RA certificate, and distributes it to all affected components. Relevant PCAs are instructed to mark the affected RA as untrusted.

The TCotSCMSM must ensure that those PCAs have at least one other RA from which to receive individual certificate requests for pseudonym certificates.

DCM(s) must no longer configure new end-entity devices to contact that RA to request pseudonym certificates.

All components and entities that receive the revocation notification also must cease to trust immediately any future message that was signed by the RA.

All end-entity devices that normally contact the revoked RA should obtain another RA's certificate and address from which to request pseudonym certificates as soon as possible (the SCMS Manager may set performance requirements for how quickly this must happen).

Assumptions

- Messages and procedures need to be defined, potentially using existing data structures defined in IEEE 1609.2 for CA revocation.
- These data structures only support the case where the CRL is signed by a single signer.
- If components use any other authentication mechanism, such as symmetric authentication or multiple signatures, the data structures, if used, would have to be redefined.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-364	MANUAL PROCESS	DCM Configuration of EEs After Component Revocation	DCM shall not configure new EEs with credentials of revoked SCMS component.	The SCMS Manager will manage the transition of devices after the revocation of a component.	In the PoC this will occur by a manual process. The DCM will provision EEs with valid certificates for SCMS components including one or more ICA and one or more RA. When the DCM learns that any component is revoked, it shall no longer provision new EEs with that revoked certificate.	DCM
SCMS-771	MANUAL PROCESS	Invoke Revocation of non-Root SCMS component	The Technical Component of the SCMS Manager shall interact with the CRLG to list the certificate of the component to be revoked in the components CRL and have the CRLG sign it.	An authenticated message from the SCMS Manager is required to revoke a component.	In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-772	MANUAL PROCESS	Standing up a SCMS component replacing the revoked component	The Technical Component of the SCMS Manager shall issue a replacement component certificate to the revoked certificate, if the revoked certificate belongs to a central component.	Upon revoking a component, a replacement component is needed to carry on its tasks and responsibilities.	This could be a new component or an existing one that will carry on the same tasks and responsibilities of the revoked component. In the PoC, this will occur by a manual process.	TCotSCMSM
SCMS-859	CLOSED	All relevant components cease to trust the revoked component	All SCMS components receiving and validating a CRL shall remove all revoked component certificates from their trust store. All cached certificate chains that roll up to a revoked component shall be removed.	The relevant components should not use the revoked component's certificate to trust it. If their chains include the revoked component, they should receive new certificate chains.	Particularly, in the case of LA revocation, the RA needs to be informed in order to stop requesting encrypted PLVs from the revoked LA. The MA needs to be informed in order to stop requesting linkage information (i.e., for misbehavior detection) from the revoked LA. In the case of RA revocation, the LAs need to be informed in order to stop sending encrypted PLVs to the revoked RA.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					<p>All SCMS components and EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately:</p> <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate 	
SCMS-1387	MANUAL PROCESS	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMSM

Table 45 Use Case 11.2.1 Revoke RA - Requirements

[5 issues](#)

Step 11.2.2: Revoke Root CA

Goals

To produce the "Revoke Root CA" message, signed by at least the required number (m below) of non-revoked Electors, which SCMS components and EEs must receive and act on.

Assumptions

Root Management is performed according to the Elector scheme outlined in [Elector-based Root Management](#).

Background and Strategic Fit

The SCMS Manager determines that a root CA is to be revoked. The SCMS Manager employs the non-revoked Electors to authenticate the revocation of a root CA. The SCMS Manager forms the bare message indicating the revocation of the root CA, including the root CA's certificate and has this message signed by at least m non-revoked Electors. The SCMS Manager instructs each Elector that it desires to sign this message, authenticating the removal of the root CA. These signatures on the message are accumulated into a final message. In this way, the SCMS Manager controls the production of the "Revoke root CA" message, signed by at least m non-revoked Electors of n . This message is delivered to all affected SCMS Components via the CRL and by proprietary messaging. To validate the "Revoke root CA" message, components or EEs must verify at least m non-revoked Electors signatures.

The MA, relevant CAs, RAs, and CRLG(s) are instructed to remove the affected root CA from their list of trusted roots. The OEMs will ensure that new end-entity devices will not be provisioned with the revoked root CA certificate.

All components and entities that receive the revocation notification also cease to trust any other affected certificate.

All end-entity devices whose certificates chain back to the revoked root CA should obtain new certificates as soon as possible (the SCMS Manager may set performance requirements for how quickly this must happen, and will coordinate).

All CAs, the MA, RAs, and CRLG(s), whose certificates chain back to the revoked root CA should cease issuing certificates with their old certificates immediately and obtain new certificates as quickly as possible (the SCMS Manager may set performance requirements for how quickly this must happen, and will coordinate). The SCMS Manager will manage the recovery from the root CA revocation and establishing a new trust hierarchy.

Procedure

To implement this process, an authorized agent of the SCMS manager will perform the following actions:

1. Obtain a copy of the root CA certificate that is to be revoked. The SCMS Manager shall define procedures for validating that the correct certificate is being revoked.
2. An agent of the SCMS Manager will present the root CA certificate to all existing, valid SCMS electors and request that they produce a digitally signed "removeRoot" ballot. The collection of all independent signed ballots from existing electors is then assembled into one root endorsement message with the sequence of elector signatures attached. The number of elector signatures must be greater than or equal to the value of 'quorum' defined in the current GPF. This is a manual process to be implemented by the TCotSCMSM.
3. The complete root removal message with signatures is then delivered to the CRLG for inclusion in an updated composite CRL file. Note that the CRLG signature is not necessary for the root removal to be validated by SCMS components. The role of the CRLG in this case is to assemble updates to the composite CRL with all active root removal messages included.

4. SCMS components (including EEs) that receive a composite CRL with one or more root removal message attached must check to see if they have already removed the root certificate from their trust store. If they have not, they must validate the root removal message by checking the attached signatures and confirming it has non-expired certificates for at least 'quorum' of the existing electors that signed the message. Once the message is validated, the SCMS component must remove the root certificate from their trust store. When validating a root removal message, an entity must check that the data, which is signed in each elector endorsement (specifically the *TbsElectorEndorsement* element), is identical and that the *EndorsementType* element of the data has the value *removeRoot*.
5. When a root is removed from a device's trust store, the device must then cease to trust any new certificates that chain back to that root.
6. When a root is removed from a device's trust store, the device must then cease to trust any certificates that chain back to that root. If a device finds that this action invalidates its own enrollment certificate or private key, it must cease operation.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-187	MANUAL PROCES S	Revoke a Root CA	The Technical Component of the SCMS Manager (TCotSCMSM) shall communicate the multi-signed "Revoke Root CA" message to CRL Generator to be included in the SCMS component CRL which will be distributed to SCMS components and EEs to inform them of the revoked Root CA.	Messages revoking Root CAs must be authenticated with m Elector signatures.	In the PoC this will message will be produced by a manual process, but automatically distributed via CRL	TCotSCMS M
SCMS-190	MANUAL PROCES S	Removing Root CA from Trust Store	MA, relevant CAs, RAs, and CRLG(s) shall validate the "Revoke Root CA" message, and if valid, the SCMS component shall remove the Root CA from its trust store. The Technical Component of the SCMS Manager shall place the "Revoke Root CA" on the CRL for distribution to EEs and other SCMS components.	Revoked Root CAs must be removed from the SCMS system with a secured message authenticated with multiple signatures.	In the PoC, a manual process will place the "Revoke Root CA" message on the CRL.	
SCMS-192	MANUAL PROCES S	SCMS Components Obtain New Certificates	All CAs, the MA, RAs, and CRLG(s), whose certificates chain back to the revoked Root CA shall cease issuing certificates with their old certificates immediately and obtain new certificates as quickly as possible.	Any operation, which use the old certificates, will no longer be trusted and will not be trusted until new certificates are obtained.	SCMS Manager may set performance requirements for how quickly this must happen	CRLG, ICA, MA
SCMS-782	MANUAL PROCES S	Root CA revocation	A quorum of Electors shall sign the Root CA revocation message to be included in the CRL.	So that SCMS components are able to verify the revocation message.		Elector
SCMS-864	EE REQUIREMEN T	EEs obtain a new LCCF upon Root CA revocation	EEs shall contact an RA to obtain a new Local Certificate Chain File (LCCF) when their current root CA has been revoked.	EE's require a valid certificate chain that can be used to validate their own pseudonym certificates and relevant SCMS component certificates.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-865	EE REQUIREMENT	EEs obtaining new Enrollment Certificates upon Root CA revocation	EEs shall obtain new Enrollment Certificates from their ECAs, if the root CA was revoked, through a re-enrollment request towards RA.	EEs need to obtain new enrollment certificates valid in the new PKI hierarchy.	Refreshed Enrollment Certificates are encrypted to the old Enrollment Certificate. The OEMs should keep a record of all Enrollment Certificates issued, so that no refreshed Enrollment Certificates are encrypted to any new Enrollment Certificate (restricting issuance of refreshed Enrollment Certificates to devices having a valid old Enrollment Certificate). This implies a strong link between the OEM and their ECA. This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-866	EE REQUIREMENT	OBES obtaining new Pseudonym/Identification Certificates upon Root CA revocation	OBES shall use the new Enrollment Certificate (cp. https://jira.campllc.org/browse/SCMS-865 to obtain new Pseudonym or	OBES need new batches of Pseudonym and Identification Certificates issued by PCAs in the new PKI hierarchy.	This requires a fresh request for butterfly keys.	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
			Identification Certificates that chain up to the new root CA.		SCMS Manager may set performance requirements for how quickly this must happen This is out of scope as it defines OBE behavior.	
SCMS-1024	CLOSED	Root CA Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Root CA" and "Revoke Root CA" messages, which will be signed by Electors, and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage root CA updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1062	MANUAL PROCESSES	Revoke Component: PG Update Global Certificate Chain File	The Policy Generator shall update the GCCF and remove all impacted certificates as soon as it receives the notification that any back-end component has been revoked.	Having an updated certificate chain file makes verification processes at EEs more efficient.	When a back-end component is revoked, it may impact the validity of other certificates on the GCCF. Specifically, when any CA is revoked, all certificates that were issued by (i.e. signed by)	PG

Key	Status	Summary	Description	Justification	Notes	Component/s
					that CA will become invalid and therefore must be removed from the GCCF. This is particularly important if a Root CA is revoked, but it applies equally to other CA revocations.	
SCMS-1170	EE REQUIREMENTS	RSEs obtain new application certificates	RSEs shall use the new Enrollment Certificate (cp. https://jira.campplc.org/browse/SCMS-865 to obtain new Application Certificates that chain up to the new root CA.	RSEs need new Application Certificates issued by PCAs in the new PKI hierarchy.	In the PoC, this will occur by a manual process. SCMS Manager may set performance requirements for how quickly this must happen	Road-side Equipment (RSE)
SCMS-1387	MANUAL PROCESSES	Remove revoked certificates from GCCF	The Technical Component of the SCMS Manager shall interact with the Policy Generator to remove the certificate of the revoked component and all its certificate chains from the Global Certificate Chain File.	Revoked certificates get invalid and therefore their certificate chains as well. They should not be available anymore via GCCF in order to save computational power during validation and bandwidth during transfer of the GCCF.		TCotSCMS M
SCMS-1409	CLOSED	Elector Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Elector" and "Revoke Elector" messages, which will be signed by Electors and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage Elector updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	
SCMS-1587	EE REQUIREMENTS	EE shall cease to trust the revoked CA	EEs receiving and validating a CRL shall remove all revoked CA certificates from their trust store, remove all cached certificate chains that roll up to the revoked CA, and stop sending immediately in case EE's enrollment, pseudonym, application, or identification certificate chains up to the revoked CA.	EE should not use the revoked component's certificate to trust it. If it chains include the revoked component, they need to receive new certificates with a new certificate chain.	EEs receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> • In sending requests to that component • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate • In sending messages signed with 	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					certificates that chain up to this component's certificate This is out of scope as it defines EE behavior.	
SCMS-1589	EE REQUIREMENT	EE receive new enrollment certificate after CA revocation	EE shall get back to the secure environment used during their bootstrapping process and be re-bootstrapped after its RCA, ICA or ECA was revoked.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes enrollment certificates that chain up to the revoked CA certificate.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1593	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after CA revocation	EE shall request new pseudonym, application, or identification certificates after it was re-bootstrapped due to revocation of its RCA, ICA, or ECA.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1608	EE REQUIREMENT	EE receive new pseudonym/application/identification certificates after PCA revocation	EE shall request new pseudonym, application, or identification certificates whenever its certificates chain up to a PCA certificate that is invalidated due to a RCA, ICA, or PCA revocation.	EE should not use the revoked CA's certificate and all certificates that chain up to that CA certificate to trust it or to use it in communication. That includes its own pseudonym/application/identification certificates that chain up to the revoked CA certificate.	EES receiving the component CRL shall mark the revoked component certificates as untrusted immediately: <ul style="list-style-type: none"> In sending requests to that component 	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					<ul style="list-style-type: none"> • In trusting certificate chains chaining to that component's certificate • In trusting messages signed using this component's certificate • In sending messages signed with certificates that chain up to this component's certificate <p>This is out of scope as it defines EE behavior.</p>	
SCMS-2461	CLOSED	SCMS components regularly pull SCMS component CRL	The SCMS component shall download the SCMS component CRL from CRLG regularly, at an interval of at least every 60 minutes	To ensure that revoked SCMS components are excluded from the system within a short time period.		DCM, ECA, LA, MA, PCA, PG, RA

Table 46 Use Case 11.2.2 Revoke Root CA - Requirements

[17 issues](#)

Design

The design for the elector-based management system is described in the [Elector-based Root Management](#) section.

Diagrams

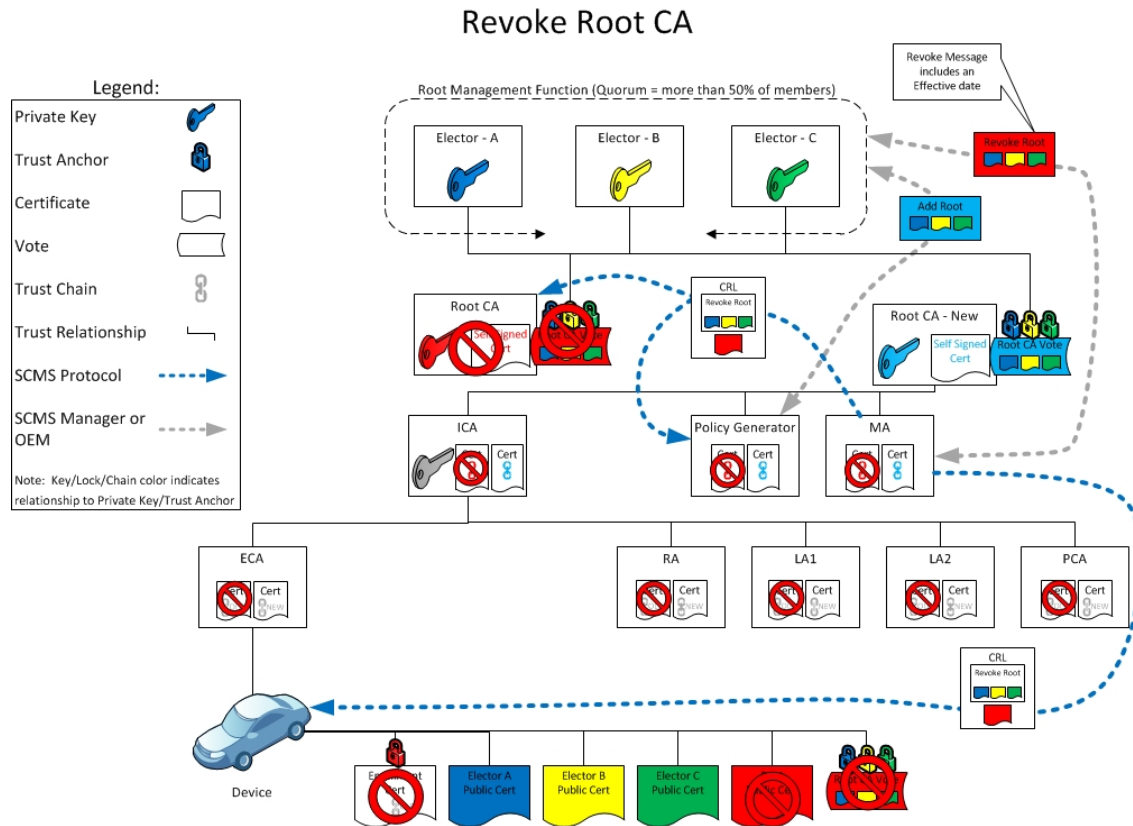


Figure 55 Revoke Root CA

Step 11.2.3: Revoke Elector

Goals

To produce the "Revoke Elector" message, signed by at least the required number (a quorum as defined in the Global Policy File) of non-revoked Electors, which SCMS Components and EEs must receive through updates to the composite CRL and act on.

Background and Strategic Fit

The SCMS Manager determines that an elector is to be revoked. The SCMS Manager employs the non-revoked electors to authenticate the revocation of the impacted elector. The SCMS Manager creates the message indicating the revocation of the elector, including the elector's certificate and has this message signed by at least "quorum" (as defined in the GPF) of non-revoked electors. The SCMS Manager instructs each elector that it desires to sign this message, authenticating the removal of the impacted elector. These signatures on the message are accumulated into a final message. In this way the SCMS Manager controls the production of the "Revoke Elector" message, signed by at least m non-revoked electors. This message is delivered to all affected SCMS Components via the CRL (proprietary messaging may also be used for faster distribution). To validate the "Revoke Elector" message, components or EEs must verify at least "quorum" non-revoked electors signatures.

The TCotSCMSM will inform the PG to create a new GCCF, removing the impacted elector signature from the root endorsement. If this causes the Root CA to be endorsed by fewer than "quorum" electors, then the TCotSCMSM must establish a new elector and have it endorse the existing root CA. The only alternative is to publish a GCCF with an un-endorsed root, which would implicitly revoke the root and cause all operations to cease.

Assumptions

- Root Management is performed according to the Elector scheme outlined in [Root Management and Revocation Recovery](#)
- Elector revocation is communicated to all SCMS components through updates to the CRL

Procedure

To implement this process, an authorized agent of the SCMS manager will perform the following actions:

1. Obtain a copy of the elector certificate that is to be revoked. The SCMS Manager shall define procedures for validating that the correct certificate is being revoked.
2. An agent of the SCMS Manager will present the elector certificate to all existing, valid SCMS electors and request that they produce a digitally signed "removeElector" ballot. The collection of all independent signed ballots from existing electors is then assembled into one elector removal message with the sequence of elector signatures attached. The number of elector signatures must be greater than or equal to the value of 'quorum' defined in the current GPF. This is a manual process to be implemented by the TCotSCMSM.
3. The complete elector removal message with signatures is then delivered to the CRLG for inclusion in an updated composite CRL file. Note that the CRLG signature is not necessary for the root removal to be validated by SCMS components. The role of the CRLG in this case is to assemble updates to the composite CRL with all active elector removal messages included.
4. SCMS components (including EEs) that receive a composite CRL with one or more elector removal messages attached must check to see if they have already removed the elector from their trust store. If they have not, they must validate the elector removal message by checking the attached signatures and confirming it has non-expired certificates for at least 'quorum' of the existing electors that signed the message. Once the message is validated, the SCMS component must remove the elector certificate from their trust store. When validating an elector removal message, an entity must check that the data, which is signed in each elector endorsement (specifically the *TbsElectorEndorsement* element) is identical and that the *EndorsementType* element of the data has the value *removeElector*.
5. When an elector is removed from a device's trust store, the device must then cease to trust any new messages endorsed by that elector.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1024	CLOSED	Root CA Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Root CA" and "Revoke Root CA" messages, which will be signed by Electors, and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage root CA updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1408	CLOSED	Elector revocation	A quorum of Electors shall sign the Elector revocation message to be included in the CRL.	So that SCMS components are able to verify the revocation message.		Elector
SCMS-1409	CLOSED	Elector Trust Store Messaging Processing	The SCMS Component shall be able to process the "Add Elector" and "Revoke Elector" messages, which will be signed by Electors and shall ensure that the number of valid signatures is at least a quorum of non-revoked Electors in its trust store.	Every SCMS component will need to manage Elector updates automatically, so therefore, every SCMS component will need to be able to process Root Management messages signed by the Electors.	For the PoC, the number of Electors will be 3 and the number of Electors required to authorize any Root Management message will be 2. Elector signatures will be produced in a manual process for the PoC.	CRL Store, CRLG, DCM, ECA, IBLM, ICA, LA, MA, PCA, PG, RA
SCMS-1413	MANUAL PROCESS	Revoke Elector: PG Update Global Certificate Chain File	The Policy Generator shall update the GCCF as soon as it receives the "Revoke Elector" message and remove the "Add Elector" message of the revoked Elector.	Having an updated certificate chain file makes verification processes at EEs more efficient.		PG

Table 47 Use Case 11.2.3 Revoke Elector - Requirements

[4 issues](#)

Design

The design for the elector-based management system is described in the [Elector-based Root Management](#) section.

Use Case 13: RSE Application Certificate Provisioning

Goals

Provide a bootstrapped RSE with an application certificate that it can use in relevant applications.

Background and Strategic Fit

The application certificate provisioning is the process by which a bootstrapped RSE receives an application certificate. As there are no location privacy or tracking concerns for RSEs, the RA is not required to shuffle the requests (unlike the case of OBEs).

This use case involves the following SCMS components:

- Pseudonym Certificate Authority (PCA)
- Registration Authority (RA)

The validity duration of application certificate is short due to the assumption that RSEs have frequent online connectivity.

Assumptions

In order to facilitate the certificate request process, a RSE must meet the following prerequisites:

- RSE has a valid enrollment certificate
- RSE has root CA, RA and PCA certificates installed
- RSE knows the FQDN of the RA

Design

The following flow chart documents the general flow of steps an RSE needs to carry out in the given order to obtain application certificates. It is not a 100% accurate description of the process. Please refer to the use case's steps and their requirements for a complete description of the process.

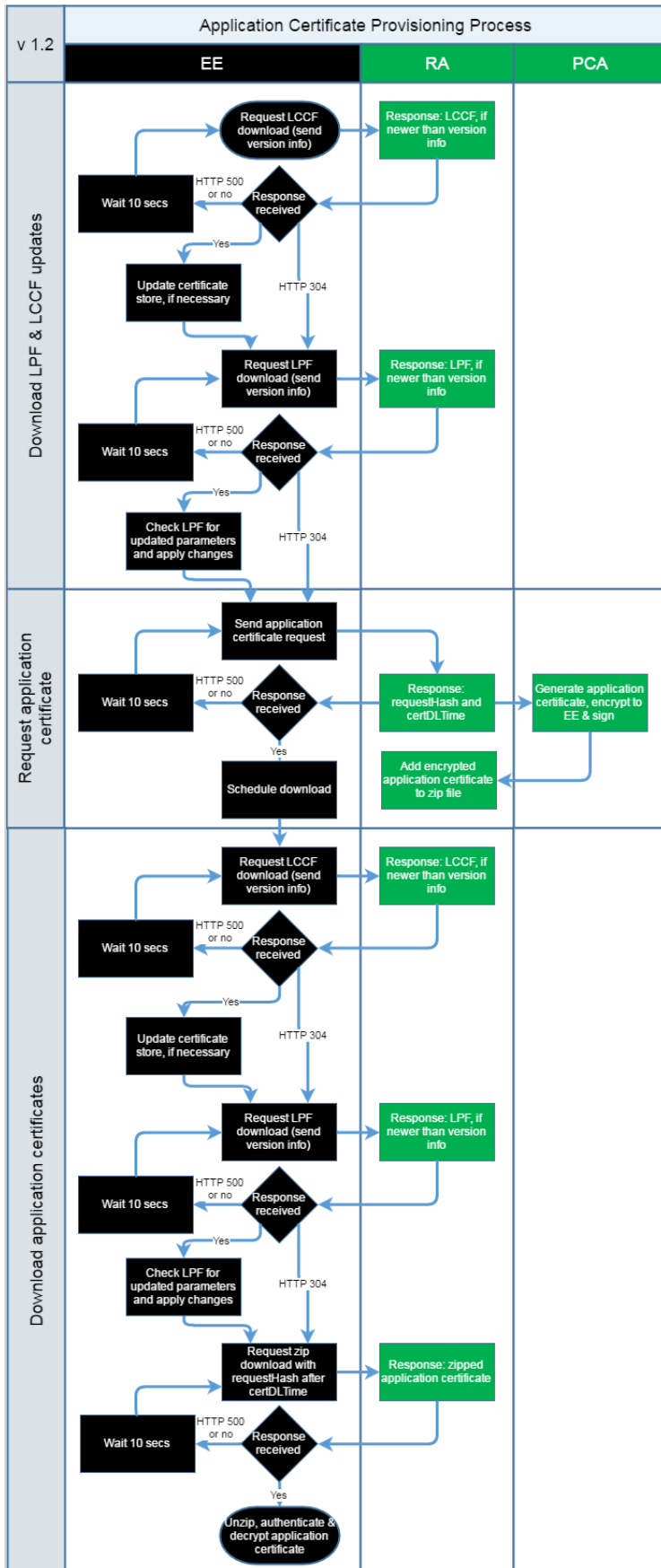


Figure 56 Application Certificate Provisioning Process

At a high level, two steps are relevant towards a RSE:

1. [Request RSE Application Certificate](#)
2. [Download RSE Application Certificate](#)

Having determined which RA to submit the request to, the RSE creates a request, signs it with the enrollment certificate, encrypts the signed request for the RA and sends it to the LOP/RA. The RA checks to make sure that the certificate request is correct and authorized, then sends back a download location (*requestHash*) and time (*certDLTime*). The RA then forwards the certificate request to the PCA. The PCA signs the application certificate, encrypts them for the RSE, signs the encrypted version of the certificate, and returns the encrypted and signed application certificate to the RA. The RA does not remove any of the named signatures or encryptions, adds them to a zip file and stores them for download by the RSE. The RSE starts downloading the zip files at *certDLTime*.

Step 13.1: Request RSE Application Certificate

Goals

The goal is to define messages and other requirements for an RSE to request an application certificate.

Background and Strategic Fit

The RSE decides to request an application certificate from its preconfigured RA.

Having determined which RA to submit the request to, the RSE creates a request, signs it with the enrollment certificate, encrypts the signed request for the RA and sends it to the RA. The RA checks to make sure that the request is correct and authorized.

RSE will attempt to download the local certificate chain file (LCCF) and the local policy file (LPF) before submitting the request. Note that any EE should download the local policy file and local certificate chain file each time it connects to RA.

Assumptions

The RSE has successfully completed [Use Case 12: RSE Bootstrapping \(Manual\)](#).

Process Steps

1. The RSE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#) using the API documented in [RA - Download local policy file](#) and [RA - Download Local Certificate Chain File](#)
 - a. If there is an updated LCCF, the RSE applies all changes to its trust-store (necessary for PCA Certificate Validations)
 - b. If there is an updated LPF, the RSE applies those changes
2. The RSE creates the request, signs it with the enrollment certificate, encrypts the signed request to the RA and sends it to the RA using the API documented in [RA - Request Application Certificate Provisioning](#)
3. The RA ensures that the certificate batch request is correct and authorized, before it starts [Step 13.2: Generate RSE Application Certificate](#)

Error Handling

1. The RSE will abandon further interactions with the RA after a certain number of failed communication attempts result in errors.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SCMS-341	<div style="border: 1px solid green; padding: 2px; display: inline-block;">EE REQUIREMENT</div>	EE TLS Cipher Suite	<p>The EE shall support at least the following TLS cipher suites for all communications to SCMS components:</p> <table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7511</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7511</td> </tr> </tbody> </table>	IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7511	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7511	<p>This is the requirement for the SSL transport tunnel.</p>	<p>This is out of scope as it defines EE behavior.</p>	<p>On-board Equipment (OBE), Road-side Equipment (RSE)</p>
			IANA Value	Description	Reference																						
			0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																						
			0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																						
			0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																						
			0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																						
			0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7511																						
			0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7511																						
<p>The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.</p>	<p>Messages from EEs to an RA must be secure against replay attacks. The signed timestamp from the EE enables the RA to validate the freshness</p>	<p>This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.</p>	<p>On-board Equipment (OBE), Road-side Equipment (RSE)</p>																								

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.camp11c.org/browse/SCMS-859 SCMS-504) and the X.509 CRL	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					https://jira.campllc.org/browse/S-CMS-405	
SCMS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available, and that configuration shall be current.	Note that LPF might have the same content as the global policy file (GPF).	RA
SCMS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SCMS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SCMS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18 :	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Provide and Enforce Technical Policies.		
SCMS-754	EE REQUIREMENT	Sign certificate request	The EE shall sign certificate requests with its enrollment certificate.	So that RA can verify that the certificate request was not been modified in transit and to verify that the certificate request is originating from a valid EE	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				<p>e Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information, as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the downloaded request to RA and the response will include a newer LCCF if available.</p>		

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-776	EE REQUIREMENT	Encrypt certificate request	The EE shall encrypt the request using the RA certificate.	So that the request is shared confidentially between the EE and RA.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant information.		RA
SCMS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SCMS-987	TESTS FAILED	Error code: raWrongParameters	RA shall log "Error code: raWrongParameters", if a device sends request with wrong parameters.	To enable server side diagnostics and to avoid giving potential attackers relevant information		RA
SCMS-988	TESTS FAILED	Error code: raRetries	RA shall log "Error code: raRetries", if the EE retries within the time specified in SCMS-522 .	To enable server		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				side diagnostics and to avoid giving potential attackers relevant information. Retry not allowed within 2 seconds.		
SCMS-990	TESTS FAILED	Error code: raMoreThanAllowedTries	RA shall return status code HTTP 500, if the EE violates SCMS-523 , and log "Error code: raMoreThanAllowedTries".	To avoid DoS attacks		RA
SCMS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid giving a potential attacker sensitive information.	RA response to EE shall follow SCMS-1397	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1066	CLOSED	RSE duplicate request check	RA shall not issue an RSE application certificates, if a request for the same PSID and an overlapping time period beyond a configurable tolerated overlap has been requested by an RSE before (identified by its enrollment certificate).	Stop misbehaving RSEs that request multiple certificates per time period.	Consider this for MA integration at a later stage.	RA
SCMS-1068	CLOSED	Error code: raRequestForMultipleCertificates	The RA shall log "Error code: raRequestForMultipleCerts" as well as identifying information of the RSE, if the RSE requested more than one certificate for the same PSID for a time period that goes beyond the tolerated overlap period.	This error code catches requests that are not duplicate but request more than one certificate per time period.		RA
SCMS-1070	CLOSED	Error code: raDuplicateRequestReceived	The RA shall log "Error code: raDuplicateRequestReceived" as well as identifying information of the EE, if EE sent a duplicate request.	This error code catches duplicate requests.	Consider this for MA integration at a later stage.	RA
SCMS-1082	CLOSED	Error code: raInvalidSignature	The RA shall log "Error code: raInvalidSignature", if the EE does not sign the certificate request with its enrollment certificate or if the signature is invalid.	To enable server side diagnostics and to avoid giving	An unsigned request might be an indication for misbehavior.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				potential attackers relevant information		
SCM S-1083	CLOSED	Error code: raRequestNotEncrypted	The RA shall log "Error code: raRequestNotEncrypted", if the EE does not encrypt the certificate request using the RA's 1609 certificate.	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unencrypted certificate request might be an indication for misbehavior.	RA
SCM S-1084	CLOSED	Error code: raInvalidCredentials	The RA shall log "Error code: raInvalidCredentials", if the EE has invalid credentials (blacklisted, expired, unauthorized)	To enable server side diagnostics and to avoid giving potential attackers relevant information	A request with invalid credentials might be an indication for misbehavior.	RA
SCM S-1085	TESTS FAILED	Error code: raUnauthorizedRequest	RA shall log "Error code: raUnauthorizedRequest", if an EE makes an unauthorized request (invalid permissions)	To enable server side diagnostics and to avoid giving potential attackers	An unauthorized request might be an indication for misbehavior.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				relevant information		
SCMS-1086	TESTS FAILED	Error code: raMalformedRequest	RA shall log "Error code: raMalformedRequest", if an EE makes a malformed request not captured in https://jira.campllc.org/browse/SCMS-1082 https://jira.campllc.org/browse/SCMS-1083 https://jira.campllc.org/browse/SCMS-1084 SCMS-1085 .	To enable server side diagnostics and to avoid giving potential attackers relevant information.	A malformed request might be an indication for misbehavior.	RA
SCMS-1087	CLOSED	Error code: raMismatch	The RA shall log "Error code: raMismatch", if this RA does not service the requesting EE.	To enable server side diagnostics and to avoid giving potential attackers relevant information.	A request from an EE that is not serviced by the requested RA might be an indication for misbehavior.	RA
SCMS-1088	CLOSED	Error code: raInvalidTimeReceived	The RA shall return status code HTTP 500, if the EE has send an invalid system time, and log "Error code: raInvalidTimeReceived".	To avoid EEs using the invalid certificates		RA
SCMS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE	To reduce resources, since RA will	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
			also shall not attempt to download a local policy file or local certificate chain file from RA.	reject request.		
SCMS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SCMS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	RA
SCMS-1210	EE REQUIREMENT	EE Secure Key Storing	EE shall store the following keys in tamper-resistant (or equivalent) storage: <ul style="list-style-type: none"> • Private enrollment key • Butterfly key parameters (seed + expansion function parameter) • All private keys (e.g., of OBE application certificates and private keys calculated from the Butterfly key parameters) 	To avoid extraction of private keys via software-based attacks.	This is out of scope since it defines EE's behavior. It is highly recommended to protect the content encryption key by a TPM-like mechanism that offers secure boot and that protects the keys against software-based attacks. Additional details are listed in Hardware, Software and OS Security	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				reachable by standard TCP/IP networking methods.		Equipment (RSE)
SCMS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Road-side Equipment (RSE)
SCMS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EEs need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				the certificate chain.		
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to malicious actors	<ul style="list-style-type: none"> Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SCMS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate	This is out of scope as it defines EEs behavior	On-board Equipment (OBE),

Key	Status	Summary	Description	Justification	Notes	Component/s
				chain update available for validating certificates and answering P2P certificate requests.		Road-side Equipment (RSE)
SCMS-1625	TESTS FAILED	RA-EE Certificate Request Ack Message	<p>RA-EE Certificate Request Ack Message shall contain the following information:</p> <p>Case: Certificate Provisioning Request Accept</p> <ul style="list-style-type: none"> • Version • Low order 8-bytes of the SHA-256 hash of the encoded "ToBeSigned" certificate request from the device • Time at which the first certificate batches will be available for download (represented by IEEE 1609.2 Time32) • URL of the certificate repository (common for all devices serviced by a specific RA) <p>Case: Certificate Provisioning Request Reject</p> <ul style="list-style-type: none"> • HTTP 500 error code 	As the EE needs to know, when and where it can go to download certificates.		RA
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				reduce the resources required and improve throughput.		
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EES shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 48 Use Case 13.1 - Requirements

[47 issues](#)

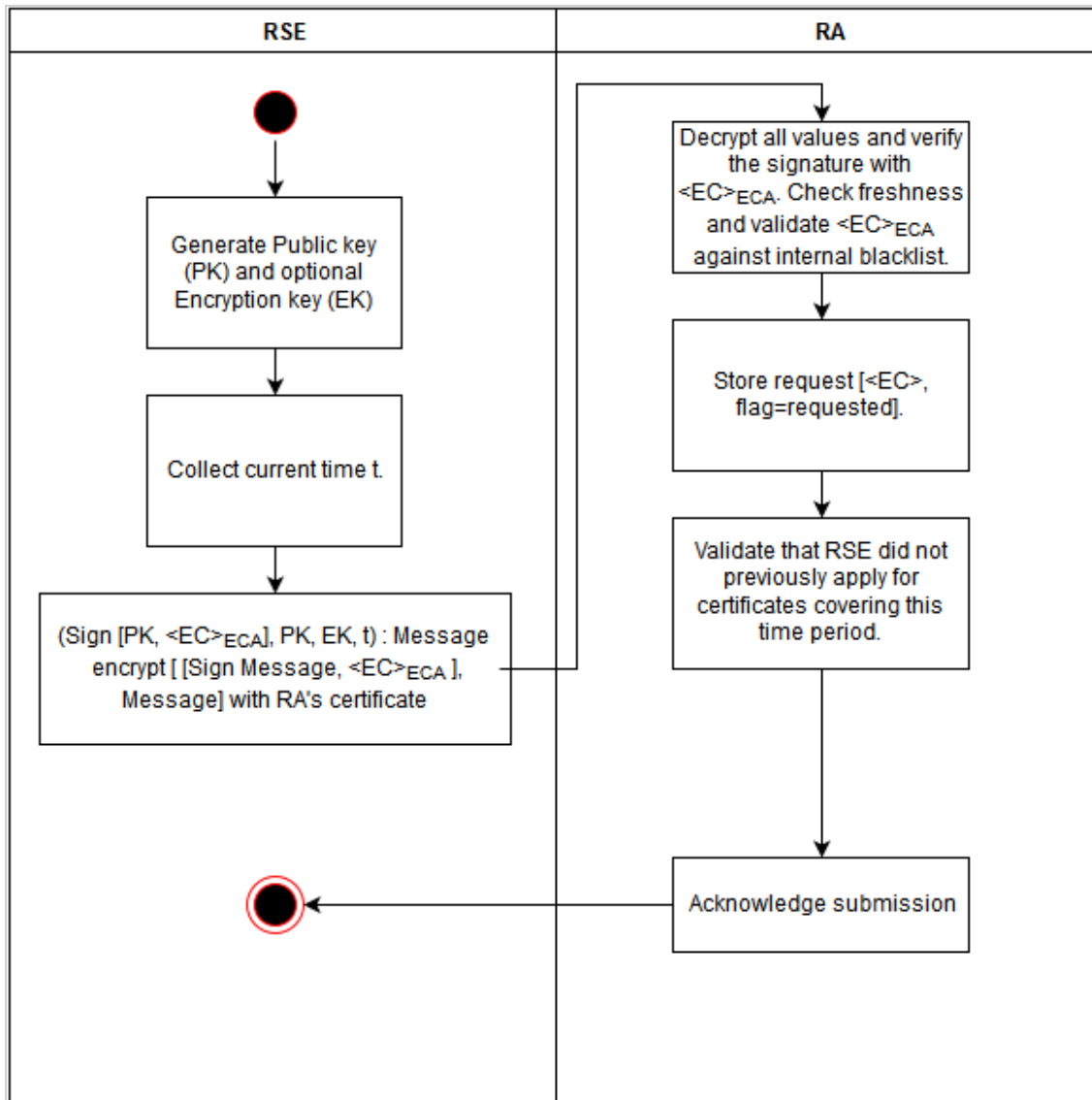


Figure 57 RSE-RA Communication

EE Request

The EE initiates the certificate request message in order to provide the RA with critical information (key parameters, current time, etc.) necessary for RSE application certificate generation. EE will send a certificate request message each time it requires a new certificate.

Security / Privacy

The Certificate Provisioning Request message shall use signing and encryption to ensure:

- The request has not been modified in transit
- The RA can verify the message came from the device
- The request is shared confidentially between the device and RA

The EE shall sign the request with the enrollment certificate. The EE shall also encrypt the request using the RA certificate and encapsulate in a 1609.2 frame of type encrypted.

Message Contents

The EE shall use the ASN.1 defined for creating the request certificate message, details can be found at [RA - Request Application Certificate Provisioning](#). In order for a request to be validated by the RA, the EE shall include the following information in the certificate provisioning request message:

- Version
- EE enrollment certificate
- A signed certificate signature key (signed with enrollment certificate)
- A response encryption key that PCA would use to encrypt the issued certificate to EE
- Optionally: a certificate encryption key that PCA would include in the issued certificate
- Current device time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)
- Requested certificate start time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)

RA Response

The RA response to the certificate provisioning request message may be *accept* (indicated by a request acknowledgement) or *reject* (indicated by a HTTP 500). In case of reject, RA shall return error code "HTTP 500" to EEs. Specific error codes should be hidden from EEs and not provide useful information to malicious actors. The RA shall log the specific error for future investigation.

RA - EE Request Acknowledgement

The request acknowledge message is initiated by the RA in response to a certificate provisioning request message successfully received from the EE. If the EE request is received and processed without triggering an error (invalid signature, blacklisted, etc.), the RA processes the certificate request and begins certificate pre-generation. The request acknowledge message provides the EE with an URL and the time where and at which the first certificate batches will be available for download.

Security / Privacy

The request acknowledge message shall use signing and encryption to ensure:

- The request has not been modified in transit
- The device can verify that the message came from the RA
- The request is shared confidentially between the device and RA

The RA shall sign and encrypt the request acknowledge message using the RA certificate and encapsulate in a 1609.2 frame of type encrypted.

Message Contents

The RA shall use the ASN.1 defined for creating the request acknowledge message in [RA - Request Application Certificate Provisioning](#) and shall include the following information:

- Case: Certificate Provisioning Request *Accept*
 - Version
 - Low order 8-bytes of the SHA-256 hash of the encoded "ToBeSigned" certificate request from the device. Returns 0 if RA cannot calculate hash of the original request
 - Time at which the first certificate file will be available for download (represented by IEEE 1609.2 Time32)

- URL of the certificate repository (common for all devices serviced by an specific RA)
- Case: Certificate Provisioning Request *Reject*
 - HTTP-500 Error Code

EE Response

If the RA provides a positive acknowledgement (*accept*) to a certificate provisioning request, the EE moves forward with the certificate download process using the provided URL given in the acknowledge message.

If the EE does not receive an acknowledgement from the RA in response to the request within defined time, EE should retry. Several conditions may necessitate the EE sending the request more than once. This may be due to:

- Request lost in transit (no TCP ack)
- RA offline, unavailable or RA network address has changed (EE must query DNS for latest RA network information)
- EE possesses an invalid RA certificate and cannot establish secure communications
- EE received HTTP-500 Error Code

The EE should not attempt to transmit the Request Certificate message without completing the prerequisites.

ASN.1 Specification

[ee-ra.asn](#)

Step 13.3: Download RSE Application Certificate

Goals

The goal is to provide a reliable, secure and timely method for RSEs to download certificates.

Background and Strategic Fit

The download will include the RSE application certificate, a local certificate chain file (LCCF), and a local policy file (LPF). The RSE will first attempt to download a LCCF (containing the PCA certificate chain required to validate the application certificate) and a LPF and process both LCCF and LPF to ensure that it is able to interpret certificates generated by the SCMS correctly. The RSE will then attempt to download the RSE application certificate.

Assumptions

- RSE has successfully executed [Step 13.1: Request RSE Application Certificate](#)
- RA retrieved the issued certificate from PCA, zipped, and stored it in a folder for RSE to download

Process Steps

1. RSE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#), as before in [Step 13.1: Request RSE Application Certificate](#)
 - a. If there is an updated LCCF, RSE applies all changes to its trust-store (necessary for PCA Certificate Validations)
 - b. If there is an updated LPF, RSE applies those changes. If those changes include changes to request parameters, RSE must skip this use case and follow [Step 13.1: Request RSE Application Certificate](#).
2. RSE downloads application certificates using the API documented in [RA - Download Application Certificate](#)

Error Handling

The RSE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in errors.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SC MS-341	EE REQUIREMENT	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SC MS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SC MS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SC MS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				download process.		
SC MS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption (https://jira.campplc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campplc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SC MS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SC MS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SC MS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-537	CLOSED	RA-to-EE encryption	The RA-to-EE communication shall be encrypted.	To avoid that an adversary is able to read EE's enrollment certificate (protect location privacy) or, in case of pseudonym certificates, that an adversary is able to read PCA-encrypted pseudonym certificates.	For pseudonym certificates, this counters a somewhat exotic attack: if an attacker eavesdrops all individually encrypted pseudonym certificates (encrypted by PCA to EE), and then later extracts the Butterfly keys (e.g., after the car arrived on the junk yard), the attacker is able to track the target vehicle in a retrofit manner assuming that attacker has access to a large database of tracking data. For other certificates, this is just an add-on security layer.	RA
SC MS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				malicious RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance		
SC MS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SC MS-590	CLOSED	Certificate availability	RA shall make application certificates available for download by the RSE.	So that proper RSEs can participate in V2I applications.		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-599	CLOSED	Keep valid application certificates	RA shall allow the RSE to download the application certificate that has previously been downloaded, so long as the RSE's credentials are still valid and the certificate is not expired.	This feature helps RSEs recover from a loss of certificates at the RSE level (e.g., disk corruption).		RA
SC MS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce Technical Policies .	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads.		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				<p>If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information, as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.</p>		
SC MS-952	<div style="border: 1px solid green; padding: 2px;">EE REQUIREMENT</div>	Error code: eePolicyFileDownloadFailed	<p>EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).</p>	<p>As the policy file is essential for the system to work</p>	<p>This is out of scope since it defines EE's behavior.</p>	<p>On-board Equipment (OBE), Road-side</p>

Key	Status	Summary	Description	Justification	Notes	Component/s
				correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to get the latest version of that file.		Equipment (RSE)
SC MS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-964	CLOSED	Error code: raCertFileUnavailable	RA shall return status code HTTP 500 to EE, if certificate batch is not available and log "Error code: raCertFileUnavailable.	to enable EE side error handling.		RA
SC MS-967	EE REQUIREMENT	Error code: eeCertFileVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of an encrypted certificate.	To enable EE side diagnostics.	This is out of scope since it defines EE behavior. This is for a single-issue certificate that has been encrypted and digitally signed by PCA.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-969	EE REQUIREMENT	Error code: eeCertificateFileDecryptionFailed	EE shall log this error code, if EE is not able to decrypt an encrypted certificate.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
						Equipment (RSE)
SC MS-971	EE REQUIREMENT	Error code: eeCertificateVerificationFailed	EE shall log this error code, if EE is not able to verify a certificate.	This is to verify the issued certificate.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-973	EE REQUIREMENT	Error code: eeCertContentFalse	EE shall log this error code, if EE is not able to parse a certificate, or if the certificate has wrong content.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid giving potential attackers relevant information	This is not in ASN.1 but http 404	RA
SC MS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SC MS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				information		
SC MS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SC MS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid giving a potential attacker sensitive information	RA response to EE shall follow SCMS-1397	RA
SC MS-1076	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of the local policy file.	As the local policy file contains security relevant configuration, it is essential to verify if a	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				recently downloaded version of that file is coming from a trustworthy source.		
SC MS-1090	CLOSED	Error code: raTcpErrors	RA shall return standard TCP error codes if TCP errors occur and log "Error code: raTcpErrors" and the encountered TCP error.	in order to enable client side error handling.		RA
SC MS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1201	EE REQUIREMENT	EE certificate download via HTTPS over TCP/IP	EE shall use HTTPS (TLS) over TCP/IP to download files from the SCMS.	In order to use standard internet technology .	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SC MS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	
SC MS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1279	EE REQUIREMENT	Error code: eeCertificateDecryptionFailed	EE shall log this error if certificate decryption failed at EE.	To allow error reaction and investigation.	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1280	EE REQUIREMENT	Error code: eeCertificateNotReadable	EE shall log this error if any certificate is not readable.	To enable error reaction and investigation.		On-board Equipment (OBE), Roadside

Key	Status	Summary	Description	Justification	Notes	Component/s
						Equipment (RSE)
SC MS-128 2	EE REQUIREMENT	Error code: eeDecompressionError	EE shall log "Error code: eeDecompressionError", if it is not able to decompress the received certificate.	To allow error reaction and investigation.	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-130 3	EE REQUIREMENT	Verification of certificate validity	EE shall verify the validity of a received certificate against IEEE 1609.2-v3-D12, clause 5.1 and 5.3.	To verify if the certificate is issued by a trustworthy source and therefore messages signed by this certificate can be trusted.	This is for testing that SCMS issued valid and proper certificates. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-135 3	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Roadside Equipment (RSE)
SC MS-137 7	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to malicious actors	<ul style="list-style-type: none"> Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA
SC MS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SC MS-1420	SCMS POC OUT OF SCOPE	Keep track of which authorization (pseudonym, ID, application) certificates are downloaded	The RA shall keep track of how far into the future each device has downloaded its certificates.	This allows revoked devices to be removed from the CRL (or deprioritized for inclusion on the CRL) once they	This is an optimization for CRL handling and therefore out of scope for PoC implementation.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				exhaust the set of certificates that they have downloaded.		
SC MS-142.1	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-169.1	CLOSED	One RSE Application certificate file per zip file	RA shall zip exactly one application certificate file per certificate download file. The content of the certificate file is the binary representation of the application certificate. <ul style="list-style-type: none"> • X • X shall be the lower 8-bytes of the SHA-256 hash of the device request in hexadecimal (case insensitive) • Where there is no extension 	There is only one RSE application certificate allowed at any given time (except for overlap) and therefore there should only be one certificate per zip file.	The device's request hash is different for each download, as application certificates need to be requested each time and are not pre-generated.	RA
SC MS-	CLOSED	RSE application certificate files	RA shall provide each application certificate to be downloaded by EE as a X.zip file in the folder provided in the ack message to the provisioning request.	This convention gives the		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
1692			<ul style="list-style-type: none"> • X.zip • Where X shall be the lower 8-bytes of the SHA-256 hash of the device request in hexadecimal (case insensitive) • Where the extension is .zip in lowercase 	RSE the ability to locate the file at the RA.		
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EES shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 49 Use Case 13.3 - Requirements

Use Case 16: RSE Application and OBE Identification Certificate Revocation

RSE Application and OBE Identification certificate revocation will be integrated with the to-be-awarded "Misbehavior Authority Integration" Project as SCMS PoC release 2.0.

Background and Strategic Fit

Misbehavior Investigation

Misbehavior investigation works as follows for RSE application certificates and OBE identification certificates (non-pseudonymous certificates without linkage values). Misbehavior investigation is further described in [16.2. RSE Application and OBE Identification Certificate Misbehavior Investigation](#).

Since MA can link certificates based on the revocation identifier field (RIF), MA now inputs the information to its global misbehavior detection algorithm. Note that misbehavior reports involving these types of certificates can be identified and directed (within the MA) to particular misbehavior investigation algorithms, based upon the PSIDs associated with the certificate information included in the misbehavior reports. Note that the considered certificates are not covered by pseudonymous considerations, such that providing the digest of the enrollment certificate does not pose a privacy concern.

Revocation

Revocation works as follows for the RSE application certificates and OBE identification certificates (non-pseudonymous certificates without linkage values). Revocation is further described in [16.3. RSE Application and OBE Identification Certificate Revocation \(CRL, blacklist\)](#).

1. MA-RA:
 - a. Using the RIF, the MA instructs the RA to add the enrollment certificate to the blacklist. Further, the MA requests a list of further valid certificates that were issued to the same enrollment certificate (e.g., all non-expired predecessors or successors).
 - b. RA adds enrollment certificate to internal blacklist
 - c. RA returns a list of all non-expired certificates that were issued to the identified enrollment certificate
2. The MA adds CertIDs (e.g., CertID8) of all non-expired certificates to the CRL

Note that revocation will be performed for all PSIDs in the reported certificate. Therefore, the CRL does not have to specify PSIDs. Further, certificates will carry all PSIDs associated with the enrollment certificate that was used to request those certificates. This implies that blacklists are not PSID-specific either.

Assumptions

Non-pseudonymous certificate types - OBE identification certificates and RSE application certificates - integrate an 8-byte revocation identifier field (RIF) that is calculated as follows:

- [LSB0-7] of RIF: the eight least significant bytes [LSB0-7] of the SHA-256 hash of the EE's enrollment certificate, i.e., $RIF_LSB0 = \text{hash_of_enrollment_cert}[LSB0]$, $RIF_LSB7 = \text{hash_of_enrollment_cert}[LSB7]$, etc.

Step 16.4: RSE CRL Check

Goals

- The RSE needs to perform several computational steps to check whether a received Basic Safety Message (BSM) has been sent by a revoked EE
- This document lists the corresponding requirements

Assumptions

The RSE received a CRL as defined in [Use Case 6: CRL Download](#).

Process Steps

1. Optional: RSE expands the CRL and calculates the linkage values for the current i-period based on the CRL entries (linkage seeds) of the CRL pseudonym certificate section. This only applies if the RSE wants to verify received BSMs.
2. Whenever RSE receives a new unknown OBE identification certificate, RSE will calculate the certificate digest of that unknown certificate and check whether the CRL lists it
 - a. If yes, then RSE discards the received certificate
 - b. Otherwise, RSE accepts the received certificate as verified
3. Optional: Whenever RSE receives a new unknown certificate, it checks whether the linkage value of that unknown certificate is listed in RSE's expanded CRL (from Step 1)
 - a. If yes, then RSE discards the received certificate
 - b. Otherwise, RSE accepts the received certificate as verified
4. Optional: Before the end of each i-period, RSE will:
 - a. Update its expanded CRL and calculate the linkage value for the next i-period
 - b. Remove entries from the expanded CRL that belong to revoked devices that ran out of certificates, if a CRL entry indicated that the revoked device does not have any more valid certificates. Note that the RSE may not immediately remove such entries, but add a safety buffer.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-786	CLOSED	CRL download	CRLG shall provide CRL via CRL store.	so that EEs can check revocation status	CRL entries may be dependent on the type of certificate	CRLG
SCMS-1171	EE REQUIREMENT	EE revoked	<p>EEs that are revoked shall not attempt to download LCCF, LPF, pseudonym certificates, identification certificates or file misbehavior reports. Exceptions to this are:</p> <ul style="list-style-type: none"> • EE is unable to determine its revocation status • EE has no pseudonym or identification certificates available in local storage • EE is attempting to perform a re-enrollment operation 	To avoid unnecessary load at the RA.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1221	EE REQUIREMENT	EE processes CRL	EE shall process the updated CRL/CRL chunk and update its CRL within 1 minute after receiving the update CRL or CRL chunk.	CRLs/CRL chunks are updated daily and EE must always update its stored CRL in a timely fashion.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1223	EE REQUIREMENT	EE checks against CRL for all certificate types	<p>EEs shall check all received relevant sender certificates, i.e., certificates of received messages that are processed, against the most recent CRL. If the sender certificate is listed, EE shall discard the received message.</p> <p>EE shall perform this check using the mechanism described in IEEE 1609.2-2016.</p>	EEs also check all relevant certificates, i.e., certificates of received messages that are processed, against the CRL. This includes OBE pseudonym, OBE identification, and RSE application certificates. It is up to EE whether it checks non-relevant certificates, i.e. certificates or received messages that are not processed, against the CRL.	<p>These checks are specified in IEEE 1609.2. Clause 5.1.3.4 describes how an EE checks whether a pseudonym certificate has been revoked by calculating the linkage values from the linkage seeds listed in the CRL, and comparing the calculated linkage value against the</p>	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					linkage value in the inspected certificate. Clause 6.4.10 and 6.4.11 contain additional information about linkage values. Clause 5.1.3.5 describes how an EE checks whether an OBE identification and RSE application certificate has been revoked by calculating the hash value of the inspected certificate, and comparing it against a CRL entry. Clause 7 contains comprehensive information about CRLs. This is out of scope since it defines EE's behavior.	
SCMS-1224	EE REQUIREMENT	EE stops sending	EE shall stop sending over-the-air DSRC messages, if it detects that itself has been listed on the CRL. This is limited to the certificates of the PSID/SSP that was revoked.	If certificates of a particular PSID/SSP have been revoked, EE stops sending all messages related to that PSID/SSP. EE might still receive DSRC messages, and send messages related to other non-revoked PSID/SSPs.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1285	EE REQUIREMENT	EE stops sending: revoked ECA for EE's	EE shall stop sending over-the-air messages, if it detects (via CRL) that its ECA, any ICA between its ECA and the root CA, or the root CA has been revoked.	In this case, EE's enrollment certificate also has been revoked.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
		enrollment certificate				Equipment (RSE)
SCMS-1286	EE REQUIREMENT	EE stops sending revoked PCA for EE's certificates	EE shall stop using all pseudonym/identification/application certificates issued by a certain PCA, if EE detects (via CRL) that anything on the pseudonym certificate chain, including PCA, ICA, or Root CA has been revoked.	If the PCA or anything in its chain was revoked, all pseudonym/identification/application certificates are also revoked.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)

Table 50 Use Case 16.4 - Requirements

[8 issues](#)

Not Doing

- For POC and CV Pilots, RSE application certificates are not listed on CRLs but revocation is enforced by not renewing certificates. At a later stage, this might be changed.
- In that case, the following requirement needs to be added: If RSE recognizes itself on the CRL, the RSE will stop sending over-the-air DSRC messages related to the indicated PSID/SSP. This also applies if RSE recognizes that the ECA, that issued RSE's enrollment certificate, or the PCA that issued RSE's certificates, has been revoked.

Use Case 18: Provide and Enforce Technical Policies

Goals

- Provide mechanisms to create an SCMS manager configuration policy
- Provide mechanisms to create policy settings and then distribute those policy settings to all SCMS components and EEs
- Provide mechanisms for individual SCMS operators to define and distribute local policies to EEs
- Provide mechanisms to distribute IEEE 1609.2 certificates to SCMS components and EEs

Background and Strategic Fit

The SCMS manager needs to set up a list of SCMS manager, technical, configuration choices and, therefore, will design technical, global policy files that are signed by the policy generator. The policy generator is an inherently centralized component.

There are local policy files affecting configurations for various SCMS components, as well as local policy files specifically for EEs. These local policy files may be signed by the appropriate SCMS component or secured through proprietary means approved by the SCMS manager. The global and local policy configuration options are displayed in [Step 18.1: Policy Configuration Options](#).

Any changes in technical global policies will result in an updated global policy file. Global policies are categorized based on if they are relevant to EEs or non-relevant to EEs. All EE relevant policies are compiled and signed by the Policy Generator (PG). Non-EE relevant policies are then appended and the entire policy file is signed again. This structure allows global limits (signed by the central PG) to be securely communicated to EEs while allowing individual RAs to assign customized values. This is described in [Step 18.3: Generate Global Policy file](#).

The local policy file is constructed by combining the complete, signed EE relevant section of the global policy file and RA specific custom policy values and/or local policies. Any of these changes in the technical policies, which directly affect the EEs operating under the jurisdiction of a particular RA, may also result in an updated local policy file. Changes to customized global policies or locally defined policies may also result in an updated local policy file. The local policy file is then signed by the RA. The EEs being operated by that RA should then download that RA's specific, updated, local policy file whenever the EE next communicates with the SCMS. This is described in [Step 18.2: Generate Local Policies for EEs](#).

There are also global certificate chain files, each version of which contains a copy of all SCMS component certificates. When any of these certificate chains change due to additions, revocations, and other revisions, the PG generates a new version of this file and distributes it to other SCMS components. In addition, each RA will create a local certificate chain file that contains (at a minimum) all of the PCA certificate chains that are used to issue pseudonym certificates for the EEs under that RA's authority. These are described in [Step 18.5: Generate Global and Local Chain File](#).

Assumptions

- The SCMS manager develops and documents global policies
- Technical global policies may include acceptable ranges within which technical local policy options may be set

Design

There are three types of policies:

- Global policies
 - Are mandatory policies that are defined by the SCMS Manager and their values set by the SCMS Manager
 - The values can be a single value, a list or a range
 - Global policies are further categorized as EE relevant and non-EE (or component) relevant
 - Are signed by the PG
- Custom policies
 - Are global policies where a specific RA has modified the values
 - Only global policies that are list or range types may have custom values
 - The custom values can be single values, a list or a range
 - All custom values must be within the limits defined by the global values
 - Are signed by an RA
- Local policies
 - Are operator-specific and not defined by the SCMS manager
 - Local policies shall not override or be substituted for policies/limits defined by the SCMS manager
 - Are signed by an RA

There are two types of certificate chain files

- Global certificate chain file
 - Contains IEEE 1609.2 certificates of all SCMS components
 - Contains elector endorsements
 - Contains root CA endorsements
- Local certificate file
 - Contains all IEEE 1609.2 certificates that are required by a specific EE(s) to validate certificates issued to the EE
 - Contains elector endorsements
 - Contains root CA endorsements
 - Optionally, contains other SCMS component IEEE 1609.2 certificates that may be useful to the EE (to validate messages from other EEs)
- There is a global PG which is operated by the SCMS manager
- PG's certificate is signed by the top-level certificate (top-level ICA, if available, and root CA otherwise)
- PG signs the technical global policy files using its complete security chain

- The technical global policy files are mandatory sets of policies applicable to SCMS components and EEs
- A repository includes technical global policies for all the different SCMS components and EEs
- PG creates a technical global policy file containing global technical policies that are applicable to RAs and EEs and provides this file to all RAs
- The respective RA conveys local policies, which are pertinent to EEs, and to the EEs through local policy files constructed by each RA
- For the PoC, the technical global policy files can be transferred manually
- PG creates a Global Certificate Chain File (GCCF) containing all certificate chains of the overall SCMS and provides this file to all RAs

Step 18.1: Policy Configuration Options

Configuration Options

Configuration options are available for global and local policy parameters.

List of Global Configuration Options

Identifier	Affected Component	Configuration Option	Description	Required for PoC	Option Value for PoC	Comment
scms_version	(all)	SCMS Version	Version number of the SCMS	Y	1	
global_cert_chain_file_id	RA, PCA, EE	The Global Certificate Chain File (GCCF) version	This identifier is used to determine if the EE's version of the LCCF is up-to-date. The identifier for the LCCF is mirrored from the GCCF identifier by the RA and included in the LCCF file name. If the GCCF and related LCCF identifier in the LCCF file name indicate that a newer LCCF version is available, the RA will download the updated LCCF to the EE.	Y	2 bytes	Additional information on the GCCF File can be found in: Global Certificate Chain File .
overdue_CRL_tolerance	EE	Maximum time to maintain trust past next CRL date	How long an EE can continue to operate without a CRL update past a next CRL date before deciding that messages are not trustworthy and rejecting all of them (and turning on an appropriate driver warning indicator).	Y	2 weeks	This is not expected to be implemented in the EEs for PoC, but should be included in the global file for PoC. There must be some point at which this transition occurs, or CRLs are greatly weakened; and it makes sense for the tolerance to change as CRL distribution technology improves.
i_period	RA, EE	The length of the certificate's i-period in minutes	Currently the i-value is defined as one week (or 10080 minutes) but this might change with more connectivity.	Y	1 week	Global parameter

Identifier	Affected Component	Configuration Option	Description	Required for PoC	Option Value for PoC	Comment
min_certs_per_i_period	RA, EE	Minimum certificates per i-period	The minimum number of certificates an EE receives per i-value (currently i-value = week). This number is also the j-value. Currently that is 20 per week and this might change over time.	Y	20	No maximum number capabilities. Note that CRL plan means that this can be set no higher than 255.
cert_validity_model	RA, EE	Certificate validity model	Pseudonym certificates are either "concurrently" or "non-concurrently" valid.	Y	concurrent	This setting means that the 20 certs per week are all concurrently valid during that week, also affects CRL.
max_available_cert_supply	RA, PCA, EE	Maximum time with which to provision OBEs with pseudonym certificates	How many years' worth of pseudo certs should be provided during the initial provisioning, and then maintained by top-off. For PoC, it is currently 3 years.	Y	3 years	Affects ability to make major changes in the overall system; also affects size of CRL.
max_cert_request_age	RA	Maximum Individual Certificate Request Age	Controls maximum amount of time an Individual Certificate Request can stay in the aggregator waiting to be shuffled.	Y	2 days	NOTE: this is only for certificate requests (not for top-offs) - minimum number or timing minimum, whichever comes first. In deployment, this will be an infinite number. Use of this option in deployment, if allowed, will require permission from the SCMS Manager.
shuffle_threshold	RA	Shuffle Threshold	Specifies the minimum number of Individual Certificate Requests to accumulate before shuffling and sending to PCA.	Y	1000	This is being considered as the minimum number for full privacy mode. Global sets acceptable option value limits. Local sets option value within Global limits.
hash_of_request_size	RA, PCA	The length in bytes of the "hash of request"	The length in bytes of the "hash of request" that PCA and RA use to identify individual requests.	Y	32 bytes	Full hash for PoC

Identifier	Affected Component	Configuration Option	Description	Required for PoC	Option Value for PoC	Comment
max_gpf_gccf_retrieval_interval	RA, PCA, LA	Maximum time interval between requesting GPF and GCCF updates	SCMS Components need to know when the contents of the Global Policy File (GPF) or Global Certificate Chain File (GCCF) change.	Y	1 day	Under current procedures, SCMS Components need to request the GPF and GCCF.
rse_application_cert_validity	RA, PCA, RSE	Validity period of RSE application certificates		Y	1 week + rse_application_cert_overlap	The value of this parameter shall be the total validity period, including the overlap interval. The initial PoC value should be 169 hours (1 week + 1 hour of overlap)
rse_application_cert_overlap	RA, PCA, RSE	Overlap period of RSE application certificates		Y	1 hour	

Table 51 List of Global Configuration Options

List of local configuration options

Identifier	Affected Component	Configuration Option	Description	Required for PoC	Option Value for PoC	Comment
shuffle_threshold	RA	Shuffle Threshold	Controls how many Individual Certificate Requests to accumulate before shuffling and sending to PCA.	Y	1000	This is being considered as the minimum number for full privacy mode; Global sets acceptable option value limits; Local sets option value within Global limits.
certs_per_i_period	RA, LA, EE	The actual number of certificates per i-value. certs_per_i_value must be equal or larger than min_certs_per_i_value.	This is the actual number of certificates an EE receives. For POC, this is an RA setting (i.e., the setting is per RA, or possibly sub-RA, not necessarily per EE).	Y	20/40	Current plan is to use 20 as the main PoC value, but to test that 40 would also work. Note that CRL plan means that this can be set no higher than 255. All affected components and/or EEs do not necessarily need to be notified separately from the results of cert update requests.

Identifier	Affected Component	Configuration Option	Description	Required for PoC	Option Value for PoC	Comment
address_la1	RA	Addresses of LA ₁	Used to communicate with Linkage Authority 1.	Y		Local configuration to be approved by SCMS Manager?
address_la2	RA	Addresses of LA ₂	Used to communicate with Linkage Authority 2.	Y		Local configuration to be approved by SCMS Manager?
address_pca	RA	Address of PCA	Used to communicate with the Pseudonym Certificate Authority.	Y		Local configuration to be approved by SCMS Manager?
tls_cert_ra	RA	TLS Certificate for RA	X.509 certificate used for transport layer security.	Y		Local configuration
shared_key_update_interval	PCA, LA	Shared symmetric key between LA and PCA	Maximum time between changes to pre-linkage value encryption/decryption key.	Y		Local configuration to be approved by SCMS Manager?
tls_cert_pca	PCA	TLS Certificate for PCA	X.509 certificate used for transport layer security.	Y		Local configuration
tls_cert_la	LA	TLS Certificate for LA	X.509 certificate used for transport layer security.	Y		Local configuration

Table 52 List of Local Configuration Options

Time Limited Configuration Options

It is valuable to define a time validity for options. This capability is very useful when the value of a configuration option changes. For instance, if `i_period` changes from one week to one day on January 1st, 2030, it is necessary to inform all EEs ahead of time about the change.

Format

This is done by including a time validity for each configuration option. Each configuration option entry can take the following time validity options:

1. N/A: there is no timely limitation for this configuration parameter
2. Sequence of configuration option value and time validity - There is a sequence of the following per configuration option entry
 - a. The configuration option value
 - b. Start time: The start-time when the configuration option value starts being valid. This is 'N/A' if the start-time was in the past. The current/first entry is always 'N/A.'
 - c. End time: The end-time until the configuration option value ends being valid. This is 'N/A' if there is no defined end-time. The last entry is always 'N/A' (open ended).

Example

The following example provides two time dependent options for the parameter `la_identifier_size`:

```
la_identifier_size, {[2, N/A, 12/31/2015], [4, 1/1/2016, N/A]}
```

Here the text in `[]` is one option, and there are two options. The first option indicates a byte size of 2 bytes for `la_identifier_size`, valid until 12/31/2015 without any start date. The second option indicates a byte size of 4 bytes for `la_identifier_size`, valid from 1/1/2016 without any end date.

PoC

PoC will test the format and delivery of this extended policy configuration, but each identifier entry will have a single open-ended time span.

Requirements

1. *Uniqueness of global policy file*: Each global policy file shall be unique in the sense that it supersedes a previous global policy file, and there is exactly one valid technical global policy file
2. *Completeness of configuration option entry*: Each configuration option entry shall be complete in the sense that it provides a configuration option value for any time in the future. This implies that the first time entry and the last time entry are always open ended ('N/A').
3. *Uniqueness of configuration option entry*: Each configuration option entry shall be unique and unambiguous, and at no point in time shall there be two valid entries
4. *Minimum options*: Each configuration option entry shall be minimal, and two subsequent time periods shall not use the same option value.

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-630	MANUAL PROCESS	Global policy file options	The global policy files shall include global configuration options from the list of configuration options listed in 18.1 - Policy Configuration Options	These values must be consistent throughout the SCMS in order to maintain nationwide interoperability	For PoC, these configuration options may be implemented manually.	
SCMS-631	MANUAL PROCESS	Local policy	Local policies shall include local configuration options from the list of configuration options listed in 18.1 - Policy Configuration Options	shuffle threshold and certs per i period must be consistent throughout the SCMS in order to maintain nationwide interoperability; remaining local policies may be unique for particular components and might be considered part of component configuration options, subject to SCMS Manager approval in most cases	For PoC, these configuration options will be implemented manually.	
SCMS-1226	EE REQUIREMENT	EE Timely Limited Configuration Options	EE shall support the use of timely limited configuration options.	It must be possible to define a time at which configuration option values change.		On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1227	EE REQUIREMENT	EE Timely Limited Configuration Options: POC	For POC, EE shall support the parsing of a timely limited configuration option policy file.	For POC, this feature will not be tested; however, the final policy file format will be used.	EE does not need to parse, process, and handle more than one choice though. If there is more than one choice, EE will only	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					consider the first choice and assume that this first choice is always valid.	
SCMS-1349	SCMS POC OUT OF SCOPE	RA Timely Limited Configuration Options	RA shall support the use of timely limited configuration options.	It must be possible to define a time at which configuration option values change.		RA
SCMS-1350	SCMS POC OUT OF SCOPE	RA Timely Limited Configuration Options: POC	For POC, RA shall support the parsing of a timely limited configuration option policy file. RA does not need to parse, process and handle more than one choice though. If there is more than one choice, RA will only consider the first choice and assume that this first choice is always valid.	For POC, this feature will not be tested, however, the final policy file format will be used.		RA

Table 53 Use Case 18.1 - Requirements

[6 issues](#)

Step 18.2: Generate Local Policies for EEs

Background and Strategic Fit

It is the responsibility of the authorized managers of EE operations to configure EEs properly. The RA, therefore, needs to provide its own appropriate, RA-specific local policy file to the EEs under its jurisdiction. Local EE policies are defined by OEMs, or other authorized managers of EE operations, for particular EE devices or EE device groups. The local EE policies must be consistent with relevant global policies. The RA needs to construct its own local policy file, within any restrictions imposed by global policies, and include all fields in the global policy file that are relevant to the EEs within that RA's jurisdiction.

Design

The Local Policy File (LPF) has one section of interest: Custom Policy. This section is a local representation of the Global Policy File with custom changes requested by the RA that issues the file. The RA has the option to remove any GPF values that are not relevant for any of the EE's that it services. The RA may also modify some global default values and replace them with local settings. The data elements for the Custom Policy section of the LPF are identical to the data elements for the GPF (listed here: [Step 18.1: Policy Configuration Options](#)). The Policy Generator (PG) must validate and sign the custom policy.

In creating the Custom Policy section of the LPF, it is assumed that the RA will start with the latest version of the Global Policy File (GPF) and make adjustments or delete specific data elements based on the needs of the EEs that it services. If the RA chooses to make no changes to the GPF, it must copy the content of the GPF into the Custom Policy section of the LPF. This allows the EEs to download a single policy file (the complete LPF) which contains all relevant policies.

Once the Custom Policy is created, the RA shall send a copy of the data structure to the PG to be validated and signed. Since the Custom Policy shares the same structure as the GPF, the RA's host ID is added to the Custom Policy to identify clearly which RA created the content. If the PG approves the Custom Policy, it will sign the complete structure (including the RA Host ID) and send it back to the RA. Note that if the RA Host ID changes, it will need to request a new Custom Policy signature to match the new Host ID.

Specific details on which GPF parameters may be modified or eliminated when translating the GPF into the Custom Policy section of the LPF must be defined by the SCMS Manager and implemented by the Policy Generator in validating signature requests.

Access & Download

To download the LPF, the EE will retrieve it from an URL defined in [RA - Services View](#).

The EE will download the files via a HTTP get request, analogous with the mechanism used to download the pseudonym certificate batch files.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1513	MANUAL PROCESS	RA to submit local policy file to PG for signature	RA shall submit the local policy file (LPF) to the PG for approval and signing prior to delivering the file to any EEs.	The LPF needs to be approved by the PG to ensure that it conforms to the limits set by the SCMS manager.	This is not in scope for the POC. The PG will check all values in the LPF to make sure that all required properties are included and that they do not exceed the limits defined by the SCMS Manager. If the PG approves of the LPF contents, it will sign the file and return it to the RA for distribution to EEs. The RA and PG will both add their signatures to the "signatures" list in the LPF data structure. EEs will check that both signatures are present and valid before applying the LPF values.	PG, RA
SCMS-1583	EE REQUIREMENT	EE parses LPF	EE shall parse the local policy file (LPF) and react to changed parameters accordingly.	EE must be able to understand the LPF. For each parameter,	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				EE will either updates its configuration, or ignore that parameter (e.g., for new parameters).		Equipment (RSE)
SCMS-1687	MANUAL PROCESS	RA constructs its local policy file	The RA shall construct its own local policy file (LPF), within any restrictions imposed by global policies, and include all fields in the global policy file (GPF) that are relevant to the EEs within that RA's jurisdiction.	It is the responsibility of the authorized managers of EE operations to configure EEs properly. The RA therefore needs to provide its own appropriate, RA-specific local policy file to the EEs under its jurisdiction.		RA
SCMS-1726	CLOSED	File name format LPF and LCCF	<p>RA shall name LPF and LCCF using the following scheme:</p> <pre>local_policy_<lpfglobalversion>_<lpflocalversion>.abc local_certificate_chains_<lccfglobalversion>_<lccflocalversion>.abc</pre> <p>where: <*globalversion> is the version id of the file. Both <*globalversion> and <*localversion> is 4 hex digit counter starting at 0000. abc is the file extension identifying the encoding format. The only file extension support for POC is oer that is indicated that file is encoded using OER. For each file, the counter value shall be unique to that file. The value shall be incremented each time the file's content changes.</p>	to have a defined naming scheme for files to be downloaded by EEs.	abc could e.g. be zip or tar. Version number is required to maintain re-freshness of LPF and LCCF The local policy file is expected to be updated at intervals; the unique identifier supports version control File naming format needs to be re-evaluated for full deployment.	RA

Table 54 Use Case 18.2 - Requirements

[4 issues](#)

Not Doing

The current Design foresees two sections in the LPF, whereas the second section is not used in the current version of the SCMS but might be utilized in future versions:

1. Custom Policy - This section is a local representation of the Global Policy File with custom changes requested by the RA that issues the file. The RA has the option to remove any GPF values that are not relevant to any of the EE's that is services. The RA may also modify some global default values and replace them with local settings. The data elements for the Custom Policy section of the LPF are identical to the data elements for the GPF (listed here: [Step 18.1: Policy Configuration Options](#)). The Policy Generator (PG) must validate and sign the custom policy.
2. Local Policy - This section contains local parameters that are not included in the Global Policy File but helps manage the EEs under RA's jurisdiction through additional configuration parameters. This section is signed by the RA only and added to the LPF after the Custom Policy was added.

Step 18.3: Generate Global Policies for EEs

Goals

The goal is to provide global policies that are valid for all EEs.

Background and Strategic Fit

The Policy Generator (PG) prepares a Global Policy File (GPF) that includes all global policies that are relevant to the EEs. The PG makes the GPF available to all SCMS components. The RA decides which of the global policies in the GPF are relevant for the EEs under that RA's jurisdiction, determines specific values within option ranges allowed in the GPF, and creates an RA-specific Local Policy File (LPF) containing this information. The RA sends its LPF to the PG for approval and signature. The RA updates its LPF whenever there is a change in the GPF that affects the information in its LPF, and subsequently makes its current LPF available to all EEs within its jurisdiction.

Assumptions

- The PG will generate a Global Policy File (GPF), which includes global policies relevant for EEs, as listed in [Step 18.1: Policy Configuration Options](#)
- The PG will make the GPF available to all RAs
- The RA will combine policy fields in the GPF that are relevant to the EEs under its jurisdiction with its particular local policy fields relevant to those EEs
- The RA will send its combined local policy file to PG for assessment of compliance with all relevant global policies
- If approved, the PG will sign the RA-specific integrated policy file (local policy file - LPF) and send it back to the appropriate RA
- The RA will make the RA-specific integrated policy file (local policy file - LPF) available to all EEs within its jurisdiction
- The RA will convey changes to the global policies that affect EEs to all EEs within its jurisdiction through an updated LPF

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-629	SCMS POC OUT OF SCOPE	SCMS Version	The global policy shall be capable of changing the SCMS version (see global policy parameters in 18.1 - Policy Configuration Options)	Major changes in the SCMS over time may be require; this SCMS version designation would indicate such a major change in the system	Out-of-scope for PoC as it is not intended to change version during PoC deployment	PG
SCMS-633	MANUAL PROCESS	Global Policy File Distribution	RA shall have mechanisms to receive the signed Global Policy File from the Policy Generator (PG).	The SCMS Manager develops and documents global policies, prepares appropriate global policy files for EEs and signs them within its Policy Generator function; RAs need to have these files to convey them to the EEs	Other authorized EE managers (such as OEMs) may also need to have mechanisms to receive signed global policy files from the PG in order to provide these files to EEs using for out-of-band communication. This might be a manual process for PoC.	RA
SCMS-634	CLOSED	File name format GPF and GCCF	<p>PG shall name GPF and GCCF using the following scheme:</p> <pre>global_policy_<gpfglobalversion>.abc global_certificate_chains_<gccfglobalversion>.abc</pre> <p>where: <*globalversion> is the version id of the GPF or GCCF. <*globalversion> is 4 hex digit counter starting at 0000. abc is the file extension identifying the encoding format. The only file extension support for POC is oer that is indicated that file is encoded using OER.</p>	The global policy file is expected to be updated at intervals; the unique identifier supports version control	File naming format needs to be re-evaluated for full deployment.	PG

Key	Status	Summary	Description	Justification	Notes	Component/s
			For each file, the counter value shall be unique to that file. The value shall be incremented each time the file's content changes.			
SCMS-635	CLOSED	Generation time	The global policy shall have a generation time	In addition to the identifier, the generation time helps to establish and confirm the precedence order of the global policy file	A generation time confirmation would help with version control mechanisms	PG
SCMS-636	CLOSED	Activation time	The global policy shall have an activation time	The activation time determines at what point in time any changes in the global policy file should be implemented	This helps to provide an orderly implementation of changes to global policies. Having multiple global policy files concurrently valid should be avoided. The SCMS Manager should use global policy ID as a sequential versioning device, with only the most recent release being valid, whether its activation is before or after previously valid versions.	PG
SCMS-637	CLOSED	Signed Global Policy	The global policy file shall be signed by the PG.	The SCMS Manager has the responsibility to set global policies, so the PG function within the SCMS Manager needs to sign the global policy files to ensure authenticity	This file preparation and signing may be a manual process in PoC, since the SCMS Manager function is not being implemented	PG

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-638	MANUAL PROCESS	Duplicate Entries	The RA shall ensure that the field entries in the local policy files that it provides to the EEs within its jurisdiction (e.g., OEM proprietary) are within the ranges and restrictions for those data fields in the current Global Policy File. If there is a duplicate field entry in both local and global policies, the global policy field entry, if more restrictive, shall take precedence.	Local policies need to be set within the allowable global policy range	For OBE, will not be tested in POC; likely to be implemented as a manual process in PoC on RA side.	RA
SCMS-640	SCMS POC OUT OF SCOPE	Field sizes for SCMS protocols and SCMS datatypes	The global policy shall be capable of changing the field sizes for SCMS protocols and SCMS data types (see global policy parameters in 18.1 - Policy Configuration Options)	Technological evolution may require longer field sizes for global policy parameters listed in 18.1 in the future; capability to change these allows for evolutionary change within the system	Out-of-scope for PoC	PG
SCMS-641	SCMS POC OUT OF SCOPE	Identifier sizes for SCMS protocols and SCMS datatypes	The global policy shall be capable of changing the identifier sizes (e.g. LA identifier) for SCMS protocols and SCMS data types.	Expansion of the number of components in the SCMS may require longer identifier sizes in the future for global policy parameters; capability to change these allows for evolutionary change within the system	Out-of-scope for PoC	PG
SCMS-642	MANUAL PROCESS	Overdue CRL tolerance	The global policy shall be capable of specifying overdue_CRL_tolerance as a time period.	Overdue CRL tolerance is expected to vary as the numbers of deployed devices increases; this	Automated process is out-of-scope for PoC; likely to be implemented as a manual process in PoC	PG

Key	Status	Summary	Description	Justification	Notes	Component/s
				parameter therefore needs to be adjustable		

Table 55 Use Case 18.3 - Requirements

[10 issues](#)

Design

Whenever there is a change in global policies that affect EEs, the RA constructs an updated version of its own LPF, gets its LPF approved (and signed) by the PG, and then makes the LPF available to the EEs within that RA's jurisdiction, i.e., whenever the EE submits a new certificate request, or otherwise contacts the RA, as appropriate. In the cases where the EE software and hardware can still support the global changes in the system, the EE will implement the changes upon receipt of the LPF containing those changes. If the policy changes are too significant for the EE to continue being functional, the EE may need to be updated or else possibly operate in a legacy mode. This could likely be managed by the relevant RA within the restrictions of global policies, but is out-of-scope for PoC.

Step 18.4: Generate Global and Local Certificate Chain File

Goals

The intended use of the Global Certificate Chain File (GCCF) and Local Certificate Chain File (LCCF) is to facilitate the distribution of certificates among SCMS components and EEs. Collecting certificate chains into these files will significantly reduce the need for collaborative distribution of certificates. These files will be the primary mechanism to inform components and EEs about new certificates in the system including replacements for components that have been revoked or whose certificates have expired or retired.

Structure

The GCCF shall contain a copy of all SCMS component certificates. It will also contain the root certificate endorsement signed by electors and any elector endorsements for newly added electors. Specifically, it will contain endorsements for all electors' certificates that have been added since the launch of the SCMS and are still valid.

Each RA will create an LCCF that contains, at a minimum, all of the PCA certificate chains that are used to issue pseudonym certificates for its EEs (this is to support P2P certificate distribution) and the SCMS certificates of all components that the EE must interact with or trust (RA, MA, CRLG, Root CA and elector endorsements). Optionally, an RA may choose to provide other PCA certificate chains in the LCCF. Any EE connecting to its associated RA shall get the current LCCF if the RA has a later version than the EE. For the POC, all content in GCCF will be contained in the required section of LCCF and these files will be created manually. The GCCF and LCCF are not signed as each certificate within the file has a signature. The recipient of a GCCF or LCCF must validate all signatures up to a trusted CA prior to trusting certificates in these files.

Example: Let us say for a particular EE, RA uses PCA1 and PCA2 for generating its pseudonym certificates. RA must provide full certificate chains for PCA1 and PCA2 in the LCCF. The RA may choose to provide certificate chains for other PCAs as well.

Using this LCCF, EEs will be able to:

- Validate certificates generated by their PCA
- Respond to a certificate request in P2P certificate distribution protocol
- Validate certificates signed by any other PCA that the RA included in the LCCF

In order to validate certificates signed by PCAs that were not included in the LCCF, the EE must request the PCA certificate chains from other EEs via collaborative distribution. The EE must validate all PCA certificate chains obtained via collaborative distribution.

Access & Download

To download the LCCF, the EE will retrieve it from an URL defined in [RA - Services View](#).

The EE will download the files via a HTTP get request, analogous with the mechanism used to download the pseudonym certificate batch files.

Format

The following diagram shows the relationship between GCCF and LCCF. Note that GCCF and LCCF do not contain initial elector or root CA certificates. However, they contain subsequent ballots endorsing elector and root CA certificates, as well as those new certificates themselves.

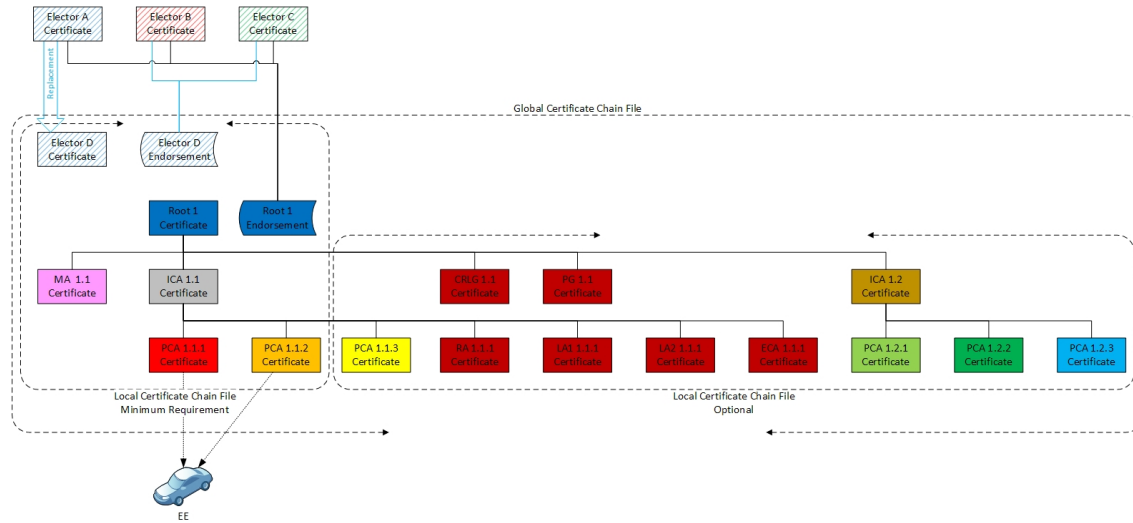
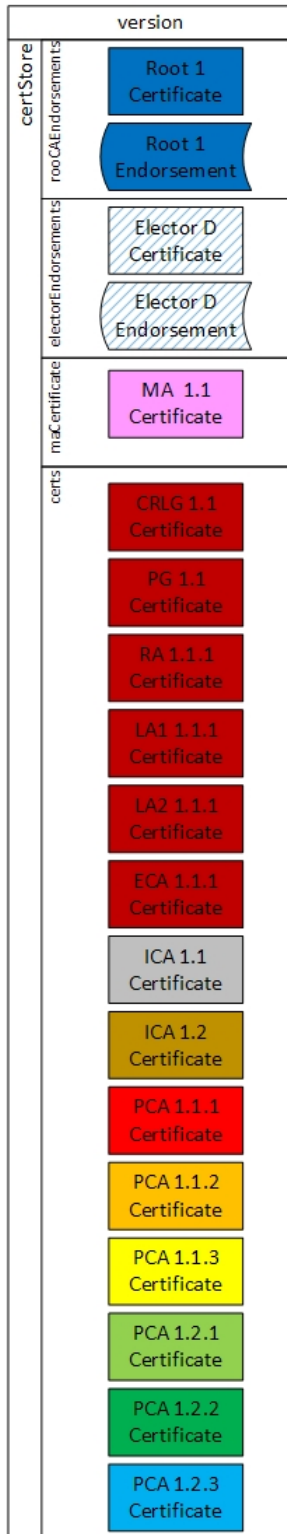


Figure 58 Relationship GCCF-LCCF

The following diagram shows the structure of GCCF and LCCF.

GCCF Structure



LCCF Structure

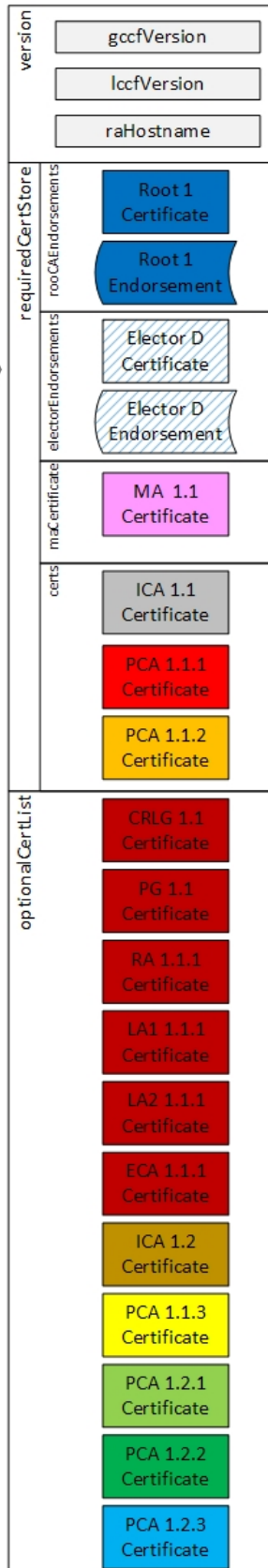


Figure 59 GCCF/LCCF Structure

Global Certificate Chain File (GCCF) Generation:

PG creates the GCCF and makes it available to all RAs whenever there is an update. It shall have the version number for updating purposes. Note that the version numbers are for management purposes only and do not serve any security purpose. The version number is the indicator that the content of the file has changed and is not an indicator of the validity of the content of the file. For the POC, the creation of GCCF is a manual process.

The GCCF structure shall contain the following elements:

Element	Notes										
version	This is a 16 bit unsigned integer that represents a unique identifier for this GCCF. It is generated by the PG when the GCCF is published (note that this value is not signed by the PG, it is for informational purposes only).										
certStore	This is a structure that holds the following values: <table border="1"> <thead> <tr> <th>Element</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>rootCAEndorsements</td> <td>One or more root certificate with signatures from at least 'n' valid electors where n >= the value of quorum defined in the GPF</td> </tr> <tr> <td>electorEndorsements</td> <td>List of electors that have been added since the launch of this instance of the SCMS (initial electors are not listed in the GCCF) with signatures from at least 'n' valid electors (not including the one endorsed) where n >= the value of quorum defined in the GPF</td> </tr> <tr> <td>maCertificate</td> <td>MA certificate</td> </tr> <tr> <td>certs</td> <td>List of certificates - Note that it is the responsibility of the generator of this file (the PG in the case of GCCF) to ensure that the list contains a complete chain with all signers required to validate any certificate on the chain all the way up to the root CA</td> </tr> </tbody> </table>	Element	Notes	rootCAEndorsements	One or more root certificate with signatures from at least 'n' valid electors where n >= the value of quorum defined in the GPF	electorEndorsements	List of electors that have been added since the launch of this instance of the SCMS (initial electors are not listed in the GCCF) with signatures from at least 'n' valid electors (not including the one endorsed) where n >= the value of quorum defined in the GPF	maCertificate	MA certificate	certs	List of certificates - Note that it is the responsibility of the generator of this file (the PG in the case of GCCF) to ensure that the list contains a complete chain with all signers required to validate any certificate on the chain all the way up to the root CA
Element	Notes										
rootCAEndorsements	One or more root certificate with signatures from at least 'n' valid electors where n >= the value of quorum defined in the GPF										
electorEndorsements	List of electors that have been added since the launch of this instance of the SCMS (initial electors are not listed in the GCCF) with signatures from at least 'n' valid electors (not including the one endorsed) where n >= the value of quorum defined in the GPF										
maCertificate	MA certificate										
certs	List of certificates - Note that it is the responsibility of the generator of this file (the PG in the case of GCCF) to ensure that the list contains a complete chain with all signers required to validate any certificate on the chain all the way up to the root CA										

Table 56 GCCF Structure Elements

Note that for the PoC, the GCCF will contain all certificates for all SCMS components.

Creation of Local Certificate Chain File (LCCF)

The RA creates the LCCF and makes it available to all EEs whenever there is an update. For the POC, the creation of LCCF is a manual process. It is up to OEMs or other authorized RA operators to decide whether they want to use the complete GCCF as their LCCF, or create only a specific, proprietary LCCF using limited, pertinent information from the GCCF.

The LCCF structure shall contain the following elements:

Element	Notes								
version	This is a structure that holds the following values: <table border="1"> <thead> <tr> <th>Element</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>gccfVersion</td> <td>This is the version number of the GCCF that was used to generate this LCCF</td> </tr> <tr> <td>lccfVersion</td> <td>This 16-bit, unsigned integer is a unique ID for this version of the LCCF that was derived from the specific GCCF on which it is based. The RA that issued this LCCF assigns this value.</td> </tr> <tr> <td>raHostname</td> <td>The fully qualified domain name (FQDN) of the RA that generated this file</td> </tr> </tbody> </table>	Element	Notes	gccfVersion	This is the version number of the GCCF that was used to generate this LCCF	lccfVersion	This 16-bit, unsigned integer is a unique ID for this version of the LCCF that was derived from the specific GCCF on which it is based. The RA that issued this LCCF assigns this value.	raHostname	The fully qualified domain name (FQDN) of the RA that generated this file
Element	Notes								
gccfVersion	This is the version number of the GCCF that was used to generate this LCCF								
lccfVersion	This 16-bit, unsigned integer is a unique ID for this version of the LCCF that was derived from the specific GCCF on which it is based. The RA that issued this LCCF assigns this value.								
raHostname	The fully qualified domain name (FQDN) of the RA that generated this file								
requiredCertStore	This is a structure that holds the following values: <table border="1"> <thead> <tr> <th>Element</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>rootCAEndorsements</td> <td>The content of this field MUST be identical to the root CA endorsement list contained in the GCCF on which this file is based</td> </tr> <tr> <td>electorEndorsements</td> <td>The content of this field MUST be identical to the elector endorsement list contained in the GCCF on which this file is based</td> </tr> <tr> <td>maCertificate</td> <td>MA certificate</td> </tr> </tbody> </table>	Element	Notes	rootCAEndorsements	The content of this field MUST be identical to the root CA endorsement list contained in the GCCF on which this file is based	electorEndorsements	The content of this field MUST be identical to the elector endorsement list contained in the GCCF on which this file is based	maCertificate	MA certificate
Element	Notes								
rootCAEndorsements	The content of this field MUST be identical to the root CA endorsement list contained in the GCCF on which this file is based								
electorEndorsements	The content of this field MUST be identical to the elector endorsement list contained in the GCCF on which this file is based								
maCertificate	MA certificate								

Element	Notes		
	<table border="1"> <tr> <td>certs</td> <td>List of certificates - This must include the full certificate chain for the root itself and for all ECAs and PCAs that it services. There may be other required content based on current SCMS policy.</td> </tr> </table>	certs	List of certificates - This must include the full certificate chain for the root itself and for all ECAs and PCAs that it services. There may be other required content based on current SCMS policy.
certs	List of certificates - This must include the full certificate chain for the root itself and for all ECAs and PCAs that it services. There may be other required content based on current SCMS policy.		
optionalCertList	This is a list of certificates. This list may include any additional certificates that the generating RA chooses to include. It should not duplicate any certificates already contained in the requiredCertStore.		

Table 57 LCCF Structure Elements

Note that for PoC, the requiredCertStore will contain the full certificate chains for all PCAs and the optionalCertList will be empty.

Use Cases Affected

1. [Use Case 1: SCMS Component Setup](#)
2. [Use Case 2: OBE Bootstrapping \(Manual\)](#) and [Use Case 12: RSE Bootstrapping \(Manual\)](#)
 - a. During bootstrap the device gets all the necessary certificates, ECA, RA, MA and LCCF
3. [Step 3.3: Initial Download of Pseudonym Certificates](#), [Step 3.5: Top-off Pseudonym Certificates](#), [Step 13.3: Download RSE Application Certificate](#), [Step 19.3: Initial Download of OBE Identification Certificates](#), and [Step 19.5: Top-off OBE Identification Certificates](#)
 - a. RA provides the updated LCCF
4. [Use Case 11: Backend Management](#)

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-634	CLOSED	File name format GPF and GCCF	<p>PG shall name GPF and GCCF using the following scheme:</p> <pre>global_policy_<gpfglobalversion>.abc global_certificate_chains_<gccfglobalversion>.abc</pre> <p>where: <*globalversion> is the version id of the GPF or GCCF. <*globalversion> is 4 hex digit counter starting at 0000. abc is the file extension identifying the encoding format. The only file extension support for POC is oer that is indicated that file is encoded using OER. For each file, the counter value shall be unique to that file. The value shall be incremented each time the file's content changes.</p>	The global policy file is expected to be updated at intervals; the unique identifier supports version control	File naming format needs to be re-evaluated for full deployment.	PG
SCMS-711	CLOSED	Global Certificate Chain List File	The Policy Generator shall create the Global Certificate Chain File (GCCF) with all valid certificate chains and a unique identifier encoded in the file name.	EEs need to know all valid certificate chains in order to validate messages from other EEs and communicate with the SCMS	For PoC, this file may be implemented manually (and is expected to be very small for PoC).	PG
SCMS-1062	MANUAL PROCESSES	Revoke Component : PG Update Global Certificate Chain File	The Policy Generator shall update the GCCF and remove all impacted certificates as soon as it receives the notification that any back-end component has been revoked.	Having an updated certificate chain file makes verification processes at EEs more efficient.	When a back-end component is revoked, it may impact the validity of other certificates on the GCCF. Specifically, when any	PG

Key	Status	Summary	Description	Justification	Notes	Component/s
					CA is revoked, all certificates that were issued by (i.e. signed by) that CA will become invalid and therefore must be removed from the GCCF. This is particularly important if a Root CA is revoked, but it applies equally to other CA revocations.	
SCMS-1352	CLOSED	Local Certificate Chain File (LCCF) generation	RA shall use appropriate information from the Global Certificate Chain File to create a Local Certificate Chain File for EEs within its jurisdiction that contains at least all the PCA certificate chains that are used to issue pseudonym certificates for those EEs (this is to support P2P certificate distribution) and the GCCF/LCCF version ID per SCMS-1354.	to support P2P certificate distribution.	Based upon the current GCCF, each RA creates its own LCCF that contains, as a minimum, all the PCA certificate chains that are used to issue pseudonym certificates for the EEs within its	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					jurisdiction and the GCCF/LCCF version ID per SCMS-1354. Optionally, RA could choose to provide additional PCA certificate chains in the LCCF.	
SCMS-1355	CLOSED	GCCF and LCCF generation in POC	RA shall put all content of GCCF in the required section of LCCF.	as there is only one single PCA in PoC	"PoC only" requirement . For POC these files will be created manually. GCCF and LCCF are not signed.	RA
SCMS-1413	MANUAL PROCESSES	Revoke Elector: PG Update Global Certificate Chain File	The Policy Generator shall update the GCCF as soon as it receives the "Revoke Elector" message and remove the "Add Elector" message of the revoked Elector.	Having an updated certificate chain file makes verification processes at EEs more efficient.		PG
SCMS-1626	CLOSED	Global Certificate Chain File Distribution	RA shall retrieve the Global Certificate Chain File (GCCF) from the Policy Generator (PG) at a regular interval - at least as frequently as specified in max_gpf_gccf_retrieval_interval.	The SCMS Manager develops policy about how often RA should retrieve GCCF and indicates this requirement in max_gpf_gccf_retrieval_interval in the GPF. For POC RA retrieves GCCF daily.	OEM may also need to have mechanisms to retrieve GCCF from policy generator. For PoC this interval	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					is daily. This might be a manual process for PoC.	
SCMS-1632	EE REQUIREMENT	EE parse LCCF	EE shall parse the local certificate chain file (LCCF) and adjust its store of trusted certificate chains accordingly.	The EE needs to be able to understand the certificate chains included in the LCCF and to maintain its own list of trusted certificate chains based upon the input from the LCCF.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCMS-1726	CLOSED	File name format LPF and LCCF	<p>RA shall name LPF and LCCF using the following scheme:</p> <pre> local_policy_<lpfglobalversion>_<lpflocalversion>.abc local_certificate_chains_<lccfglobalversion>_<lccflocalversion>.abc </pre> <p>where: <*globalversion> is the version id of the file. Both <*globalversion> and <*localversion> is 4 hex digit counter starting at 0000. abc is the file extension identifying the encoding format. The only file extension support for POC is oer that is indicated that file is encoded using OER. For each file, the counter value shall be unique to that file. The value shall be incremented each time the file's content changes.</p>	to have a defined naming scheme for files to be downloaded by EEs.	abc could e.g. be zip or tar. Version number is required to maintain refreshness of LPF and LCCF The local policy file is expected to be updated at intervals; the unique identifier supports version control File naming format needs to be re-evaluated for full deployment.	RA

Table 58 Use Case 18.4 - Requirements

[9 issues](#)

Use Case 19: OBE Identification Certificate Provisioning

Goals

The goal is the initial request of OBE identification certificates and then subsequent top-up.

Background and Strategic Fit

The OBE identification certificate provisioning is the process by which a bootstrapped OBE receives an identification certificate. As there are no location privacy or tracking concerns for identification certificates (but anonymity concerns), the RA is not required to shuffle the requests (unlike the case of pseudonym certificates). Butterfly keys are still used to allow easy top-up. Revocation is enabled by adding individual identification certificates to the CRL, but OBE Identification certificates do not use linkage values. Each OBE only receives one identification certificate per time period, except for a minimal overlap period to account for critical events.

This use case involves the following SCMS components:

- Pseudonym Certificate Authority (PCA)
- Registration Authority (RA)

The validity duration of identification certificate is dependent on the connectivity of the OBE. Smaller validity durations will potentially reduce the number of CRL entries but require more connectivity of the OBE to RA.

Assumptions

- The OBE is assumed to have a valid enrollment certificate that empowers it to request OBE identification certificates; specifically related to its SSID and SSP combination in the enrollment certificate. Some applications may require additional enrollment certificates to be added to the OBE, such as first responder vehicles. The addition of another enrollment certificate would occur in a secure environment.
- The OBE is assumed to have Root CA, RA and PCA certificates
- The OBE is assumed to have relevant address(s) to communicate with the RA
- The identification certificate that is issued has a validity period consistent with an associated application

Design

The following flow chart documents the general flow of steps an OBE needs to carry out in the given order to obtain identification certificates. It is not a 100% accurate description of the process. Please refer to the use case's steps and their requirements for a complete description of the process.

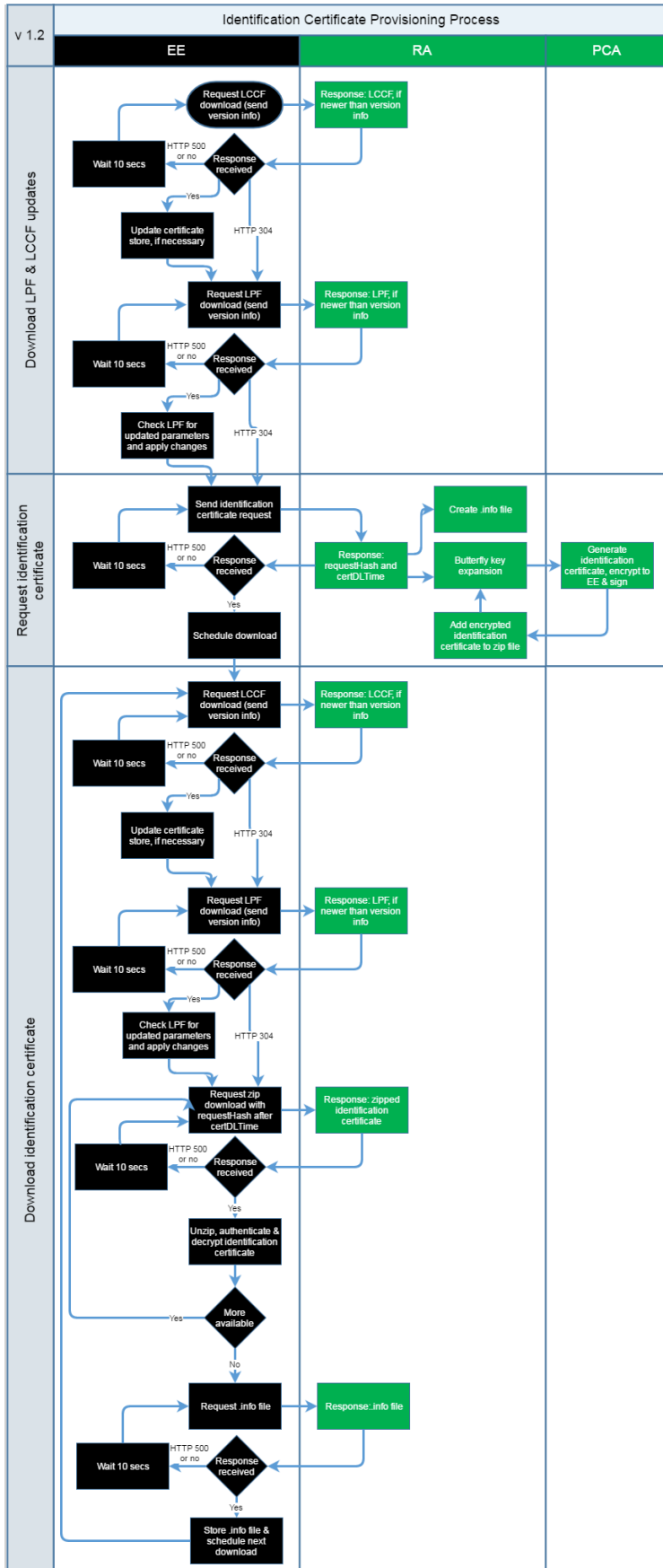


Figure 60 Identification Certificate Provisioning Process

OBE Identification certificates use [Butterfly Keys](#) for the certificate signature key (mandatory), butterfly keys for the certificate encryption key (optional), and butterfly keys for a response encryption key (mandatory). The use-case works as follows:

- Initial Request
 - EE creates a random, signature, butterfly key public seed (elliptic curve point) and a random, expansion, function parameter. EE signs those with its enrollment certificate.
 - Optional: EE creates a random, encryption, butterfly key public seed (elliptic curve point) and a random, expansion, function parameter. The resulting encryption keys are optionally used as encryption key in a certificate.
 - EE creates a response, encryption, butterfly key public seed (elliptic curve point) and a random, expansion, function parameter. The resulting encryption keys are used by PCA to encrypt the issued certificate to EE.
 - EE provides the signed (with enrollment certificate) signature, butterfly key public seed and expansion function parameter and a response, encryption, butterfly key public seed and expansion function parameter to RA. EE optionally provides the encryption butterfly key public seed and expansion function parameter. All parameters are signed with EE's enrollment certificate, and encrypted to RA.
 - RA verifies all received parameters
 - RA creates butterfly keys based on the policy (either policy linked to EE and/or PSID; e.g., one certificate per month for month, one hour of overlap between certificates). RA creates Butterfly keys for the certificate signature key, for the response encryption key, and optionally for the certificate encryption key.
 - RA creates a revocation identifier (RIF) for EE
 - RA does not shuffle nor wait on purpose before forwarding to PCA
 - RA forwards to PCA the certificate signature butterfly key (B_i), RIF, response encryption key (H_i), and optionally the certificate encryption key (E_i)
 - PCA issues the certificate using B_i and RIF and, if available, E_i. PCA then encrypts the issued certificate with H_i and signs the encrypted certificate
 - RA collects PCA's responses, bundles them in file(s), and stores it in a folder
 - EE can now download the file(s)
- Top-up
 - RA regularly checks and will initiate a generation of certificates as needed and defined in the policy
 - RA will look-up RIF and calculate the proper Butterfly key(s) and send butterfly keys B_i, H_i, and RIF to PCA. If available, RA will also include E_i.
 - PCA issues certificates, encrypts to EE, and sign the encrypted certificate
 - RA collects PCA's responses, bundles them in file(s), and stores the file(s) in a folder
 - EE can now download the file

At a high level, two steps are relevant towards an OBE:

1. [Step 19.1: Request for OBE Identification Certificates](#)
2. [Step 19.3: Initial Download of OBE Identification Certificates](#)
3. [Step 19.5: Top-off OBE Identification Certificates](#)

Step 19.1: Request for OBE Identification Certificates

Goals

The goal of this use case is to define the messages and actions that allow a device to request new identification certificates from the RA.

Background and Strategic Fit

The OBE decides to request an identification certificate from its preconfigured RA.

Having determined which RA to submit the request to, the OBE creates a request, signs it with the enrollment certificate, encrypts the signed request for the RA and sends it to the RA. The RA checks to make sure that the request is correct and authorized.

Assumptions

In order to facilitate the certificate request process, the following prerequisites should be met:

- The OBE has successfully completed [Use Case 2: Bootstrapping](#)

Process Steps

1. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#) by using the API documented in [RA - Download local policy file](#) and [RA - Download Local Certificate Chain File](#)
 - a. If there is an updated LCCF, the OBE applies all changes to its trust-store (necessary for PCA Certificate Validations)
 - b. If there is an updated LPF, the OBE applies those changes
2. The OBE creates the request, signs it with the enrollment certificate, encrypts the signed request for the RA and sends it to the RA using the API documented [RA - Request Identification Certificate Provisioning](#).
3. The RA ensures that the request is correct and authorized before it starts with [Step 19.2: OBE Identification Certificate Generation](#)

Error Handling

1. The OBE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in errors
2. The OBE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The OBE will need to execute the certification/bootstrap process again to exit a revoked state.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SCMS-341	EE REQUIREMENT	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>Iana Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7251</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7251</td> </tr> </tbody> </table>				Iana Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251
			Iana Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7251																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7251																									
SCMS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Road-side Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campllc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campllc.org/browse/SCMS-405).	RA
SCMS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available,	Note that LPF might have the same content as the global policy file (GPF).	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				and that configuration shall be current.		
SCMS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SCMS-520	EE REQUIREMENT	Request only initial set	OBE shall make a certificate provisioning request only for the initial set of pseudonym and application certificates or when the certificate parameters change	Because top-up certificates are generated automatically by the RA.	This is out of scope as it defines OBE behavior.	On-board Equipment (OBE)
SCMS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SCMS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				limit the number of retries.		Equipment (RSE)
SCM S-529	CLOSED	Store enrollment certificate and butterfly parameters	RA shall store enrollment certificate and butterfly parameters for each OBE for its lifetime.	so that OBE can be revoked properly. Arbitrary number based on historical trends for vehicle ownership. For example, collector vehicles that are kept on the road for longer than typical vehicles.	PoC will only store 3 years	RA
SCM S-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				using the latest policy for new downloaded certificates . Policy definition details are available at Use Case 18: Provide and Enforce Technical Policies.		Equipment (RSE)
SCMS-754	EE REQUIREMENT	Sign certificate request	The EE shall sign certificate requests with its enrollment certificate.	So that RA can verify that the certificate request was not been modified in transit and to verify that the certificate request is originating from a valid EE	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				<p>. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information, as appropriate for the EEs under its jurisdiction.</p> <p>. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.</p>		
SCM S-776	EE REQUIREMENT	Encrypt certificate request	The EE shall encrypt the request using the RA certificate.	So that the request is shared confidentialia	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				lly between the EE and RA.		Equipment (RSE)
SCM S-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information , it is important to have an error handling whenever the EE is not able to get the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.		
SCMS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant information.		RA
SCMS-987	TESTS FAILED	Error code: raWrongParameters	RA shall log "Error code: raWrongParameters", if a device sends request with wrong parameters.	To enable server side diagnostics and to avoid giving potential attackers relevant information.		RA
SCMS-988	TESTS FAILED	Error code: raRetries	RA shall log "Error code: raRetries", if the EE retries within the time specified in SCMS-522 .	To enable server side diagnostic		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				s and to avoid giving potential attackers relevant information . Retry not allowed within 2 seconds.		
SCMS-990	TESTS FAILED	Error code: raMoreThanAllowedTries	RA shall return status code HTTP 500, if the EE violates SCMS-523 , and log "Error code: raMoreThanAllowedTries".	To avoid DoS attacks		RA
SCMS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid giving a potential attacker sensitive information .	RA response to EE shall follow SCMS-1397	RA
SCMS-1076	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code, if EE is not able to verify the digital signature of the local policy file.	As the local policy file contains security relevant configuration, it is essential	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				to verify if a recently downloaded version of that file is coming from a trustworthy source.		
SCMS-1082	CLOSED	Error code: raInvalidSignature	The RA shall log "Error code: raInvalidSignature", if the EE does not sign the certificate request with its enrollment certificate or if the signature is invalid.	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unsigned request might be an indication for misbehavior.	RA
SCMS-1083	CLOSED	Error code: raRequestNotEncrypted	The RA shall log "Error code: raRequestNotEncrypted", if the EE does not encrypt the certificate request using the RA's 1609 certificate.	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unencrypted certificate request might be an indication for misbehavior.	RA
SCMS-1084	CLOSED	Error code: raInvalidCredentials	The RA shall log "Error code: raInvalidCredentials", if the EE has invalid credentials (blacklisted, expired, unauthorized)	To enable server side diagnostics and to avoid giving potential attackers relevant information	A request with invalid credentials might be an indication for misbehavior.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-1085	TESTS FAILED	Error code: raUnauthorizedRequest	RA shall log "Error code: raUnauthorizedRequest", if an EE makes an unauthorized request (invalid permissions)	To enable server side diagnostics and to avoid giving potential attackers relevant information	An unauthorized request might be an indication for misbehavior.	RA
SCMS-1086	TESTS FAILED	Error code: raMalformedRequest	RA shall log "Error code: raMalformedRequest", if an EE makes a malformed request not captured in https://jira.campllc.org/browse/SCMS-1082 https://jira.campllc.org/browse/SCMS-1083 https://jira.campllc.org/browse/SCMS-1084 SCMS-1085 .	To enable server side diagnostics and to avoid giving potential attackers relevant information	A malformed request might be an indication for misbehavior.	RA
SCMS-1087	CLOSED	Error code: raMismatch	The RA shall log "Error code: raMismatch", if this RA does not service the requesting EE.	To enable server side diagnostics and to avoid giving potential attackers relevant information	A request from an EE that is not serviced by the requested RA might be an indication for misbehavior.	RA
SCMS-1088	CLOSED	Error code: raInvalidTimeReceived	The RA shall return status code HTTP 500, if the EE has send an invalid system time, and log "Error code: raInvalidTimeReceived".	To avoid EEs using the invalid certificates		RA
SCMS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
			revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.			Equipment (RSE)
SCMS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SCMS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	RA
SCMS-1210	EE REQUIREMENT	EE Secure Key Storing	EE shall store the following keys in tamper-resistant (or equivalent) storage: <ul style="list-style-type: none"> Private enrollment key Butterfly key parameters (seed + expansion function parameter) All private keys (e.g., of OBE application certificates and private keys calculated from the Butterfly key parameters) 	To avoid extraction of private keys via software-based attacks.	This is out of scope since it defines EE's behavior. It is highly recommended to protect the content encryption key by a TPM-like mechanism that offers secure boot and that protects the keys against software-based attacks. Additional details are listed in Hardware, Software and OS Security	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process		On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				and reduce communication cost.		
SCMS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Road-side Equipment (RSE)
SCMS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EEs need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				EEs to validate the certificate chain.		
SCMS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to malicious actors	<ul style="list-style-type: none"> Standard TCP (https://jira.campllc.org/browse/SCMS-1090) and TLS (https://jira.campllc.org/browse/SCMS-977) errors shall be reported to EEs All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	CRL Store, RA
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SCMS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				validating certificates and answering P2P certificate requests.		
SCMS-1512	EE REQUIREMENT	Generating Butterfly Key seeds and expansion function	The EE shall generate butterfly key seeds and expansion function.	Protect privacy of data during transfer by not extracting the keys.	For OBE pseudonym certificates, OBE will generate Butterfly key parameters for the certificate signature keys and the response encryption key. For OBE identification certificates, OBE will generate Butterfly key parameters for the certificate signature keys, and optionally for certificate encryption keys and response encryption keys.	On-board Equipment (OBE)
SCMS-1625	TESTS FAILED	RA-EE Certificate Request Ack Message	<p>RA-EE Certificate Request Ack Message shall contain the following information:</p> <p>Case: Certificate Provisioning Request Accept</p> <ul style="list-style-type: none"> Version Low order 8-bytes of the SHA-256 hash of the encoded "ToBeSigned" certificate request from the device Time at which the first certificate batches will be available for download (represented by IEEE 1609.2 Time32) URL of the certificate repository (common for all devices serviced by a specific RA) <p>Case: Certificate Provisioning Request Reject</p> <ul style="list-style-type: none"> HTTP 500 error code 	As the EE needs to know, when and where it can go to download certificates		RA
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				to the SCMS. This will reduce the resources required and improve throughput.		
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2612	REVIEW	Store butterfly parameters	RA shall store butterfly parameters for each OBE for the estimated functional lifetime of the OBE.	So that the certificate pre-generation and revocation can function properly. Arbitrary number based on historical trends for vehicle ownership. For example, collector vehicles that are		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				kept on the road for longer than typical vehicles.		

Table 59 Use Case 19.1 - Requirements

[49 issues](#)

Design

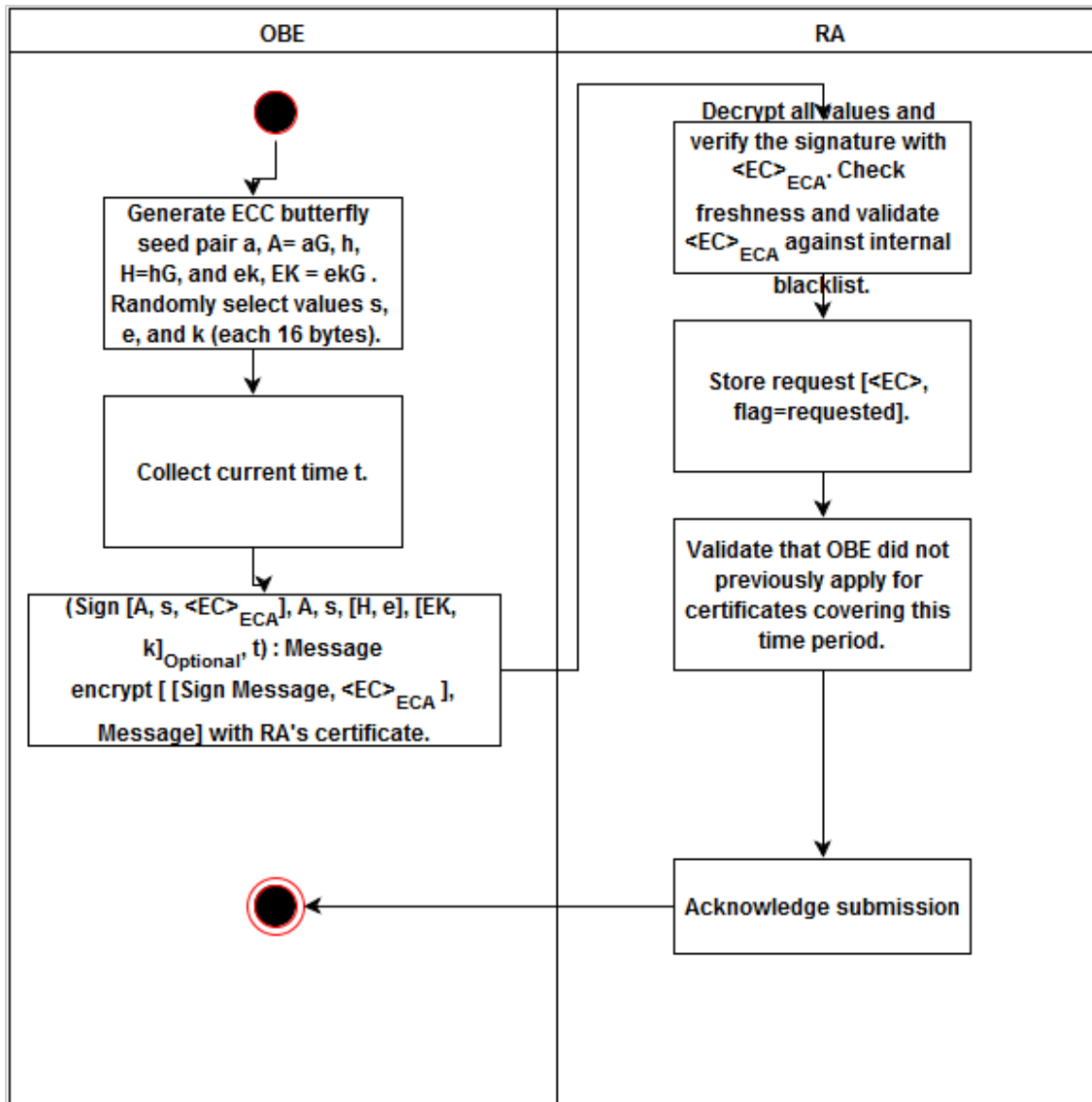


Figure 61 OBE-RA Communication

EE Request

The EE initiates the Certificate Provisioning Request message in order to provide the RA with critical information (key parameters, current time, etc.) necessary for the OBE identification certificate generation. New devices may experience some delay between the initial request and the time the first certificate is available for download to accommodate provisioning processes such as certificate generation and certificate encryption. The RA will store information from the initial Certificate Provisioning Request message and use for ongoing certificate pre-generation until:

- The device provides new parameters in a subsequent Certificate Provisioning Request
- The device is blacklisted at the RA due to misbehavior or malfunction

The Certificate Provisioning Request message shall be sent once for each unique request. No subsequent Certificate Provisioning Request is necessary to acquire new certificates.

Security / Privacy

The Certificate Provisioning Request message shall use signing and encryption to ensure:

- The request has not been modified in transit
- The RA can verify the message came from the device
- The request is shared confidentially between the device and RA

The EE shall sign the request with the Enrollment Certificate. The EE shall also encrypt the request using the RA certificate and encapsulate in a 1609.2 frame of type encrypted.

Message Contents

The EE shall use the ASN.1 defined for creating the Request Certificate message, details can be found at [RA - Identification Certificate Provisioning Request](#) . In order for a request to be validated by the RA, the EE shall include the following information in the Certificate Provisioning Request message:

- Version
- EE enrollment certificate
- Butterfly public seed / expansion function (see [Butterfly key](#) for details) parameters for:
 - Certificate signing key (signed with enrollment certificate)
 - Response encryption key (to encrypt the created certificate towards EE)
 - Optionally certificate encryption key
- Current device time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)
- Requested certificate start time: 32-bit denoting number of seconds since the Epoch (as defined in 1609.2)

RA Response

The RA response to the Certificate Provisioning Request message may be *accept* (indicated by a Request Acknowledgement) or *reject* (indicated by a HTTP 500). Specific error codes should be hidden from EEs to avoid providing useful information to malicious actors. RA shall log the specific error for future investigation.

RA - EE Request Acknowledgement

The Request Acknowledge message is initiated by the RA in response to a Certificate Provisioning Request message successfully received from the EE. If the EE request is received and processed without triggering an error (invalid signature, blacklisted, etc.) the RA processes the certificate request and begins certificate pre-generation. The Request Acknowledge message provides the EE with the URL and the time where and at which the first certificates batches will be available for download.

Security / Privacy

The Request Acknowledge message shall use signing and encryption to ensure:

- The request has not been modified in transit
- The device can verify the message came from the RA
- The request is shared confidentially between the device and RA

The RA shall sign and encrypt the Request Acknowledge message using the RA certificate and encapsulate in a 1609.2 frame of type encrypted.

Message Contents

The RA shall use the ASN.1 defined for creating the Request Acknowledge message, which can be found at [RA - Identification Certificate Provisioning Request](#) and shall include the following information:

- Case: Certificate Provisioning Request *Accept*
 - Version
 - Low order 8-bytes of the SHA-256 hash of the encoded "ToBeSigned" certificate request from the device. Returns 0 if RA cannot calculate hash of the original request.
 - Time at which the first certificate batches will be available for download (represented by IEEE 1609.2 Time32)
 - URL of the certificate repository (common for all devices serviced by an specific RA)
- Case: Certificate Provisioning Request *Reject*
 - HTTP-500 Error Code

EE Response

If the RA provides a positive acknowledgement (*accept*) to a Certificate Provisioning Request, the EE moves forward with the certificate batch download process using the provided URL and time both given in the acknowledge message.

If the EE does not receive an acknowledgement from the RA in response to the request within the defined time, the EE should retry. Several conditions may necessitate the EE sending the request more than once. This may be due to:

- Request lost in transit (no TCP ack)
- RA offline, unavailable or the RA network address has changed (EE must query DNS for latest RA network information)
- The EE possesses an invalid RA certificate and cannot establish secure communications
- The EE received HTTP-500 Error Code

The EE should not attempt to transmit the Request Certificate message without having completed the prerequisites.

ASN.1 Specification

- [ee-ra.asn](#)
- [scms-protocol.asn](#)
- [scms-base-types.asn](#)
- [scms-error.asn](#)
- [scms-policy.asn](#)
- [scms-common-errors.asn](#)
- [1609dot2-schema.asn](#)
- [1609dot2-base-types.asn](#)

Step 19.3: Initial Download of OBE Identification Certificates

Goals

The goal is to provide a reliable, secure, and timely method for certified devices to download OBE identification certificates.

Background and Strategic Fit

The purpose of this use-case is to provide a defined method that a certified OBE can use to download OBE identification certificates. The download will include:

1. File(s) X_i.zip that each include one file X_i with a certificate
2. A .info file that includes the time when new certificates will be available
3. A local certificate chain file containing all PCA certificate chains required to validate the identification certificates, but not the policy file

Assumptions

- The OBE has successfully completed [Step 19.1: Request for OBE Identification Certificates](#)
- The RA retrieved the issued certificates from the PCA, zipped, and stored them in a folder for OBE to download

Process Steps

1. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#), as done before in [Step 19.1: Request for OBE Identification Certificates](#)
 - a. If there is an updated LCCF, the EE applies all changes to its trust-store (necessary for the PCA Certificate Validations)
 - b. If there is an updated LPF, the EE applies those changes
2. The OBE downloads the new OBE identification certificates using the API documented in [RA - Download Identification Certificate](#)
3. The OBE downloads .info file using the API documented in [RA - Download .info File](#)

Error Handling

1. The OBE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in critical errors
2. The OBE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The device will need to execute the certification/bootstrap process again to exit a revoked state.

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SC MS-341	EE REQUIREMENT	EE TLS Cipher Suite	The EE shall support at least the following TLS cipher suites for all communications to SCMS components:	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
			<table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>				IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515
			IANA Value				Description	Reference																			
			0xC0,0x23				TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																			
			0xC0,0x24				TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																			
			0xC0,0x2B				TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																			
			0xC0,0x2C				TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																			
			0xC0,0xAC				TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																			
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SC MS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	RA
SC MS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).	RA
SC MS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration	Note that LPF might have the same content as the global policy file (GPF).	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				available, and that configuration shall be current.		
SC MS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SC MS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption for the download process. (https://jira.campplc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campplc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SC MS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SC MS-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				location of EEs.		
SC MS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SC MS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-537	CLOSED	RA-to-EE encryption	The RA-to-EE communication shall be encrypted.	To avoid that an adversary is able to read EE's enrollment certificate (protect location privacy) or, in case of pseudonym certificates, that an adversary is able to read PCA-encrypted	For pseudonym certificates, this counters a somewhat exotic attack: if an attacker eavesdrops all individually encrypted pseudonym certificates (encrypted by PCA to EE), and then later extracts the Butterfly keys (e.g., after the car arrived on the junk yard), the attacker is able to track the target vehicle in a retrofit manner assuming that attacker has access to a large database of tracking data. For other certificates, this is just an add-on security layer.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				pseudonym certificates.		
SC MS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	
SC MS-543	CLOSED	Individual certificate downloads	RA shall support individual certificate batch, or certificate file, downloads by EEs.	The design allows download of individual certificate batches, or files, to avoid that an EE needs to download all certificates each time. This also allows easier resume of a download.		RA
SC MS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SC MS-547	CLOSED	Available certificate batches	The number of certificate batches, or certificate files, available for download shall be configurable (e.g. 3 years) as defined by the configuration option max_available_cert_supply in the global policy.	This might change during the lifetime of the SCMS. It might even vary for different EEs.		RA
SC MS-548	CLOSED	X.info file	RA shall provide an .info file for download by EE.	The .info file provides	In order for the EE to determine the earliest time which new certificate batches will be	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				information when new pseudonym certificates, or identification certificates, can be downloaded.	available for download, the RA shall maintain a file in each device specific repository. This file will contain a timestamp at which the RA is predicted to update certificate batches in the device repository. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004). The file shall be named according to the following format: X.info Where X is the lower 8-bytes of the SHA-256 hash of device request in hexadecimal	
SC MS-549	CLOSED	Keep Certificates	The RA shall allow the EE to download certificates that have previously been downloaded, so long as the devices credentials are still valid and the certificates are not expired.	to recover from a loss of certificates at the device level (e.g., disk corruption).		RA
SC MS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				available at Use Case 18: Provide and Enforce Technical Policies.		
SC MS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information , as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				the download request to RA and the response will include a newer LCCF if available.		
SC MS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to get the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-964	CLOSED	Error code: raCertFileUnavailable	RA shall return status code HTTP 500 to EE, if certificate batch is not available and log "Error code: raCertFileUnavailable.	to enable EE side error handling.		RA
SC MS-965	EE REQUIREMENT	Error code: eeCertFileDownloadFailed	If OBE is not able to download pseudonym or identification certificate files (e.g., because there is none or it is corrupted), OBE shall implement OEM defined error handling and store the error code in OBE's error log file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-969	EE REQUIREMENT	Error code: eeCertificateFileDecryptionFailed	EE shall log this error code, if EE is not able to decrypt an encrypted certificate.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-971	EE REQUIREMENT	Error code: eeCertificateVerificationFailed	EE shall log this error code, if EE is not able to verify a certificate.	This is to verify the issued certificate.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-973	EE REQUIREMENT	Error code: eeCertContentFalse	EE shall log this error code, if EE is not able to parse a certificate, or if the certificate has wrong content.	To enable EE side diagnostics	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid giving potential attackers relevant information	This is not in ASN.1 but http 404	RA
SC MS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SC MS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				giving potential attackers relevant information.		
SC MS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-981	CLOSED	Error code: raNoPcaCertificateChainFileAvailable	RA shall return status code HTTP 500, if Local Certificate Chain File is not available and log "Error code: raNoPcaCertificateChainFileAvailable".	To enable client side error handling.		RA
SC MS-982	CLOSED	X.info file update period	RA shall update the .info file at least on a weekly basis.	The .info file is updated regularly to provide timely updates to EE		RA
SC MS-984	EE REQUIREMENT	Error code: obeInfoFileDownloadFailed	OBE shall log this error code, if it is not able to download the .info file (e.g. because there is none or it is corrupted).	to enable EE side diagnostics.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE)
SC MS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should always be logged for diagnostic purposes and never returned to the EE to avoid	RA response to EE shall follow SCMS-1397	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				giving a potential attacker sensitive information .		
SC MS-1090	CLOSED	Error code: raTcpErrors	RA shall return standard TCP error codes if TCP errors occur and log "Error code: raTcpErrors" and the encountered TCP error.	in order to enable client side error handling.		RA
SC MS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1201	EE REQUIREMENT	EE certificate download via HTTPS over TCP/IP	EE shall use HTTPS (TLS) over TCP/IP to download files from the SCMS.	In order to use standard internet technology .	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SC MS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	
SC MS-1214	EE REQUIREMENT	OBE downloads .info file	OBE shall download the .info file each time OBE downloaded pseudonym or identification certificates.	EE requires the information to learn when certificates will be available for download.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SC MS-1215	EE REQUIREMENT	EE contacts RA for certificate download	EE shall try to download certificates any time after the time provided by the time-stamp in the .info file that has been recovered last time EE tried to download, or downloaded, certificates.	To avoid wasting resources by trying to download certificates before they are available.	This is out of scope since it defines EE behavior. The timestamp shall be in the IEEE 1609.2 Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004).	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1228	CLOSED	OBE identification certificate files	<p>RA shall provide each identification certificate to be downloaded by EE as a X_i.zip file in the folder provided in the ack message to the provisioning request.</p> <ul style="list-style-type: none"> • X_i.zip • Where X shall be the lower 8-bytes of the SHA-256 hash of the device request in hexadecimal (case insensitive) • Where i is a file iterator in hexadecimal starting at 0 (case insensitive) • Where the extension is .zip in lowercase 	this convention gives the OBE the ability to locate the file at the RA.	The file iterator i starts at 0 and is then incremented by 1 for each new file. The first issued certificate is stored in X_1.zip, the second certificate is stored in X_2.zip, the 4 billion-th certificate is stored in X_EE6B2800.zip, and so on.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1303	EE REQUIREMENT	Verification of certificate validity	EE shall verify the validity of a received certificate against IEEE 1609.2-v3-D12, clause 5.1 and 5.3.	To verify if the certificate is issued by a trustworthy source and therefore messages signed by this certificate can be trusted.	This is for testing that SCMS issued valid and proper certificates. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EEs need to be able to validate received SCMS certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate the certificate chain.	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request. This is out of scope as it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1377	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA
SC MS-1397	TESTS FAILED	Error reporting to EE	The SCMS Components shall return error code "HTTP 500" to EEs in response to all application level errors at RA.	Specific error codes should be hidden from EEs to prevent useful information from being provided to	<ul style="list-style-type: none"> Standard TCP (https://jira.campplc.org/browse/SCMS-1090) and TLS (https://jira.campplc.org/browse/SCMS-977) errors shall be reported to EEs 	CRL Store, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				malicious actors	<ul style="list-style-type: none"> All errors at the HTTP and higher levels shall be HTTP 500 for RA & ECA 	
SC MS-140-4	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-140-5	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details.	RA
SC MS-142-0	SCMS POC OUT OF SCOPE	Keep track of which authorization (pseudonym, ID, application) certificates are downloaded	The RA shall keep track of how far into the future each device has downloaded its certificates.	This allows revoked devices to be removed from the CRL (or deprioritized for inclusion on the CRL) once they exhaust the set of certificates that they have downloaded.	This is an optimization for CRL handling and therefore out of scope for PoC implementation.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1639	EE REQUIREMENT	Download certificate batches	OBE shall not attempt to download certificate batches for i-value periods more than max_available_cert_supply in the future	To reduce resource usage by not attempting to download certificate batches that do not exist.	<ul style="list-style-type: none"> This is out of scope as it defines EE behavior. This is the OBE counterpart of https://jira.campllc.org/browse/SCMS-547 	
SC MS-2251	CLOSED	One OBE identification certificate file per zip file	<p>RA shall zip exactly one identification certificate file per certificate download file. The content of the certificate file is the binary representation of the encrypted identification certificate.</p> <ul style="list-style-type: none"> X_i X shall be the lower 8-bytes of the SHA-256 hash of the device request in hexadecimal (case insensitive) i is a file iterator in hexadecimal starting at 0 (case insensitive) Where there is no extension 	There is only one OBE identification certificate allowed at any given time (except for overlap) and therefore there should only be one	The file iterator i starts at 0 and is then incremented by 1 for each new file. The first issued certificate is stored in X ₁ , the second certificate is stored in X ₂ , the 4 billion-th certificate is stored in X _{EE6B2800} , and so on.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificate per zip file.		
SC MS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 60 Use Case 19.3 - Requirements

[62 issues](#)

Step 19.5: Top-off OBE Identification Certificates

Goals

The goal is to provide a reliable, secure, and timely method for certified devices to download credentials.

Background and Strategic Fit

The purpose of this use case is to provide a defined method that a certified OBE can use to download subsequent batches of credentials. The step at hand is to top-up OBE identification certificates. It is similar to [Step 19.3: Initial Download of OBE Identification Certificates](#). Differences are documented in this section. Also, see [Step 19.4: Schedule generation of subsequent batch of OBE identification certificates](#) for full details of the process to schedule certificate pre-generation.

Assumptions

- The OBE has successfully completed [Step 19.1: Request for OBE Identification Certificates](#)
- The OBE has successfully completed [Step 19.3: Initial Download of OBE Identification Certificates](#)
- The RA retrieved the issued certificates from PCA, zipped, and stored them in a folder for OBE to download

Process Steps

1. The OBE checks that, and if necessary waits until, the current time matches or is after the timestamp given in the .info file
2. The OBE downloads the [Local Policy File \(LPF\)](#) and the [Local Certificate Chain File \(LCCF\)](#), as done before in [Step 19.3: Initial Download of OBE Identification Certificates](#)
 - a. If there is an updated LCCF, OBE applies all changes to its trust-store (necessary for PCA Certificate Validations)
 - b. If there is an updated LPF, OBE applies those changes
3. The OBE downloads the new OBE identification certificates
4. The OBE downloads .info file using the API documented in [RA - Download .info File](#)

Error Handling

- The EE will abandon further interactions with the RA after a certain number of failed communication attempts resulted in critical errors
- The EE will not attempt to execute the certificate provisioning process if it finds itself on the latest CRL (assumes that a willful violator has not compromised the device). The device will need to execute the certification/bootstrap process again to exit a revoked state.
- The EE may terminate the certificate batch download process if sufficient storage is not available for subsequent batches

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SCMS-341	EE REQUIREMENT	EE TLS Cipher Suite	<p>The EE shall support at least the following TLS cipher suites for all communications to SCMS components:</p> <table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC751</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC751</td> </tr> </tbody> </table>	IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC751	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC751	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
IANA Value	Description	Reference																									
0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																									
0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																									
0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																									
0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																									
0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC751																									
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC751																									
SCMS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Road-side Equipment (RSE)																					
SCMS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is	RA																					

Key	Status	Summary	Description	Justification	Notes	Component/s
					<p>CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.</p> <p>specified in RFC 6066, Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will return to the EE.</p> <p>For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.</p>	
SCMS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	<p>Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campplc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campplc.org/browse/SCMS-405).</p>	RA
SCMS-509	CLOSED	Stop pre-generating pseudonym and OBE identification certificates for revoked device	RA shall stop pre-generating pseudonym and OBE identification certificates for a device that has been revoked by the MA, i.e., for a device that appears on RA's internal blacklist.	so that computing resources are not wasted by generating certificates for revoked devices		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCMS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available, and that configuration shall be current.	Note that LPF might have the same content as the global policy file (GPF).	RA
SCMS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SCMS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption (https://jira.camplic.org/browse/SCMS-537 and RA-EE authentication (https://jira.camplic.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA
SCMS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCM S-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the location of EEs.		RA
SCM S-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SCM S-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious RA. RA authenticates via its	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance		
SCMS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key). OCSP stapling is specified in RFC 6066 , Section 8.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-544	CLOSED	Download resume	RA shall support byte-wise resume of certificate batch, certificate file, or policy file, downloads, even if EE switches the IP address.	To improve reliability of the download protocol.		RA
SCMS-576	CLOSED	Update .info file	The RA shall update .info files for all EEs even if no new certificate batches are created.	The EE uses the .info file to determine when the earliest the next download is allowed to happen.	Timestamp in .info file is dynamically calculated based on system load. PoC scope will be to update .info file for non-revoked EEs only.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SCM S-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce Technical Policies .	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SCM S-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				with the new chain information, as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.		
SCMS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				not able to get the latest version of that file.		
SCM S-954	EE REQUIREMENT	Error code: eePolicyVerificationFailed	EE shall log this error code in EE's error log file, if EE is not able to verify the digital signature of the local policy file.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-956	EE REQUIREMENT	Error code: eePolicyFileParsingFailed	EE shall log this error code in EE's error log file, if EE is not able to parse the successfully downloaded local policy file (e.g., because it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to read the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-958	EE REQUIREMENT	Error code: eeConnectionFailed	EE shall log this error code, if it cannot connect to RA because there is a connection timeout.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
						Equipment (RSE)
SCMS-964	CLOSED	Error code: raCertFileUnavailable	RA shall return status code HTTP 500 to EE, if certificate batch is not available and log "Error code: raCertFileUnavailable.	to enable EE side error handling.		RA
SCMS-976	CLOSED	Error code: raInvalidURL	RA shall log "Error code: raInvalidURL", if EE requests invalid URL.	To enable server side diagnostics and to avoid giving potential attackers relevant information	This is not in ASN.1 but http 404	RA
SCMS-977	CLOSED	TLS error codes	RA shall return standard TLS error codes if TLS errors occur.	In order to enable client side error handling.		RA
SCMS-978	CLOSED	Error code: raAuthenticationFailed	RA shall log "Error code: raAuthenticationFailed", if EE-to-RA authentication fails.	To enable server side diagnostics and to avoid giving potential attackers relevant information.		RA
SCMS-979	EE REQUIREMENT	Error code: eeAuthenticationFailed	EE shall log "Error code: eeAuthenticationFailed", if RA-to-EE authentication fails.	To enable EE side diagnostics.	This is out of scope since it defines EE's behavior. This is part of TLS handshake. OEM defines EE error handling.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1065	CLOSED	Error code: raBlacklisted	RA shall log "Error code: raBlacklisted" if the requesting EE has been blacklisted.	Error's produced by an EE should	RA response to EE shall follow SCMS-1397	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				always be logged for diagnostic purposes and never returned to the EE to avoid giving a potential attacker sensitive information.		
SCM S-1164	EE REQUIREMENT	OBE next download timing	OBE shall use the stored .info file to schedule the next download attempt.	The .info file contains the timestamp when the next batch of certificates (pseudonym or identification) will be available for download. This timestamp is the earliest the OBE is allowed to connect to the RA for the next download. The timestamp shall be in the IEEE 1609.2	This is out of scope since it defines EE's behavior. <ul style="list-style-type: none"> If no pseudonym certificates are available on the OBE for the current i_period (week), the OBE is allowed to make a download attempt at any time. If no pseudonym certificates are available on the OBE for the next i_period (week), the OBE is allowed to make a download attempt at any time. If no identification certificate is available on the OBE for the current or next time period, the OBE is allowed to make a download attempt at any time. 	On-board Equipment (OBE)

Key	Status	Summary	Description	Justification	Notes	Component/s
				Time32 format (the number of (TAI) seconds since 00:00:00 UTC, January 1, 2004).		
SCM S-1171	EE REQUIREMENT	EE revoked	<p>EEs that are revoked shall not attempt to download LCCF, LPF, pseudonym certificates, identification certificates or file misbehavior reports. Exceptions to this are:</p> <ul style="list-style-type: none"> • EE is unable to determine its revocation status • EE has no pseudonym or identification certificates available in local storage • EE is attempting to perform a re-enrollment operation 	To avoid unnecessary load at the RA.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCM S-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SCM S-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					(e.g., linkage information, certificate digest, etc.) that allows the RA to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	
SCMS-1214	EE REQUIREMENT	OBE downloads .info file	OBE shall download the .info file each time OBE downloaded pseudonym or identification certificates.	EE requires the information to learn when certificates will be available for download.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE)
SCMS-1263	EE REQUIREMENT	EE download resume	EE shall try byte-wise resume of CRL downloads from the CRL store, certificate batches, certificate files, or policy files from RA in case a previous download failed.	This will improve reliability of the download process and reduce communication cost.		On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1270	EE REQUIREMENT	Network connection	EEs shall use TCP/IP to communicate with the SCMS.	SCMS components (server) are only reachable by standard TCP/IP networking methods.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1282	EE REQUIREMENT	Error code: eeDecompressionError	EE shall log "Error code: eeDecompressionError", if it is not able to decompress the received certificate.	To allow error reaction and	Out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				investigation.		Equipment (RSE)
SCMS-1289	EE REQUIREMENT	OBE identification certificate duplicate downloads	The OBE shall not download OBE identification certificates that are already verified and stored in OBE.	During top-up downloads, the EE shall only download OBE identification certificates that are not currently verified and stored on the device. This is to prevent repeated downloads of the same content.		On-board Equipment (OBE)
SCMS-1291	EE REQUIREMENT	Expired Certificate Files	The OBE shall only download OBE identification certificate files for the current and future time periods.	Only download certificates that are not expired yet.		On-board Equipment (OBE)
SCMS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Road-side Equipment (RSE)
SCMS-1356	EE REQUIREMENT	EE uses internal certificate store	The EE shall use its internal certificate store to validate received SCMS certificates and respond to P2P certificate requests.	EES need to be able to validate received SCMS	EE does not need to store all certificate chains, the LCCF provides the minimum set and EEs can learn additional chains via P2P certificate request.	On-board Equipment (OBE), Road-side

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificates based on their certificate chain up to the SCMS root CA. EEs need to respond to P2P certificate requests to enable receiving EEs to validate the certificate chain.	This is out of scope as it defines EE's behavior.	Equipment (RSE)
SCMS-1377	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA
SCMS-1404	EE REQUIREMENT	EE send data via HTTP post over TCP/IP	EE shall use HTTP post to send data towards the RA	To allow the SCMS endpoint to serve everything based on HTTP protocol	RA - Services View will document the actual HTTP post details. This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1405	CLOSED	RA accept authenticated HTTP post requests	RA shall accept HTTP post requests only from authenticated EEs.	To allow the SCMS endpoint to serve everything based on	RA - Services View will document the actual HTTP post details.	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				HTTP protocol		
SCMS-1420	SCMS POC OUT OF SCOPE	Keep track of which authorization (pseudonym, ID, application) certificates are downloaded	The RA shall keep track of how far into the future each device has downloaded its certificates.	This allows revoked devices to be removed from the CRL (or deprioritized for inclusion on the CRL) once they exhaust the set of certificates that they have downloaded.	This is an optimization for CRL handling and therefore out of scope for PoC implementation.	RA
SCMS-1421	EE REQUIREMENT	LCCF validation in EE	The EE shall verify the LCCF and then update the internal certificate store each time it receives a new LCCF.	To have the latest certificate chain update available for validating certificates and answering P2P certificate requests.	This is out of scope as it defines EEs behavior	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-1639	EE REQUIREMENT	Download certificate batches	OBE shall not attempt to download certificate batches for i-value periods more than max_available_cert_supply in the future	To reduce resource usage by not attempting to download	<ul style="list-style-type: none"> This is out of scope as it defines EE behavior. This is the OBE counterpart of https://jira.camp11c.org/browse/SCMS-547 	

Key	Status	Summary	Description	Justification	Notes	Component/s
				certificate batches that do not exist.		
SCMS-2463	EE REQUIREMENT	EE transactions per TLS session	EE shall perform as many SCMS transactions as possible using a single TLS session.	To minimize the number of separate TLS sessions to the SCMS. This will reduce the resources required and improve throughput.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SCMS-2610	EE REQUIREMENT	Use FQDN found in certificate	EEs shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)

Table 61 Use Case 19.5 - Requirements

[50 issues](#)

Design Notes

- See [Step 19.3: Initial Download of OBE Identification Certificates](#) for full details of the download process
- From the SCMS point of view, the basic process for "top-up" certificate downloads is the same as that used for initial provisioning as detailed in [Step 19.3: Initial Download of OBE Identification Certificates](#). However, this is an incremental download and not a full download of all available certificate files. The number of files downloaded shall be factored in system sizing requirements.
- From the OBE's point of view, the process is slightly different from the process for initial provisioning
- The RA will record the last time an OBE established a connection. This last connection time will be used to stop pre-generating pseudonym certificates if there is no activity for a period of time.
- The RA will automatically resume pre-generating pseudonym certificates when an OBE reestablishes a connection. The new certificates will be available for download at the time specified in the .info file.

Use Case 20: EE Re-Enrollment

EE re-enrollment will be integrated with the to-be-awarded "SCMS PoC extension" project as SCMS PoC release 3.0. Until then these are preliminary concepts.

Goals

All End Entities (EEs, including OBEs and RSEs) receive an [Enrollment Certificate](#) as part of a secure initial provisioning process (see [Use Case 2: OBE Bootstrapping \(Manual\)](#) and [Use Case 12: RSE Bootstrapping \(Manual\)](#) for details). This certificate is used to authenticate the EE to an RA for all secured transactions with the SCMS. When this certificate approaches its expiration, the EE must be [re-established](#) to receive a new certificate. There are also cases where infrastructure components (such as an ICA or Root CA) may be revoked without directly impacting the EEs that have certificates that are chained back to the revoked component. The re-enrollment use cases describe secure procedures for maintaining the integrity and security of EE enrollment certificates in these situations.

Assumptions

The EE has a non-revoked, non-expired enrollment certificate and the EE has not been placed on the RA's blacklist.

Step 20.1: EE Enrollment Certificate Rollover

Goals

Define a procedure to securely re-enroll a non-revoked EE when its enrollment certificate is about to expire.

During the bootstrap process, an EE is issued an enrollment certificate by an Enrollment CA (ECA) via a DCM in a secure environment, which is used to authenticate communication between an EE and the RA. When an enrollment certificate approaches its expiration date, it must be rolled over to a new certificate so that the EE can continue to authenticate with the RA. This process does not take place in a secure environment, and no trusted DCM is available. Instead, the existing enrollment certificate is used to facilitate secure communication with the RA performing a similar task to the DCM.

Some EEs may not have reliable network access, so the request to re-enroll and the retrieval of the new enrollment certificate are separated into two individual transactions. This separation also allows the RA to choose the timing for when it will forward the request to the Enrollment CA. This time delay may be

needed to ensure that the RA has access to an ECA with an expiration time that will allow for the validity period of the new enrollment certificate. When an EE requests re-enrollment, the RA will return a time estimate for when the new certificate will be ready for download. This procedure is similar to the process of requesting and downloading pseudonym certificates.

An EE may request re-enrollment at any time, if it has a currently valid enrollment certificate and the EE has not been added to the RA's blacklist. The RA for the EE's current enrollment certificate will accept only one request for re-enrollment. The new enrollment certificate will have a validity period that begins when the current enrollment certificate expires (there is no overlap in the validity period for enrollment certificates).

Assumptions

- The EE possesses a valid enrollment certificate that has not been blacklisted by the RA.
- The EE has not previously requested re-enrollment using the currently valid enrollment certificate.
- An ECA is available to sign re-enrollment requests.
- The ECA's certificate will be valid for the entire duration of any re-enrollment request that it signs.
- For any EE, only one enrollment certificate may be issued for a particular PSID/SSP combination at a time (see [Certificate Types](#) for details).
- An EE should only be allowed to initiate one re-enrollment request for a particular PSID/SSP combination.
- The new enrollment certificate will have the same PSID/SSP, and will have a validity period starting at the expiry date of the old enrollment certificate (there is no overlap in the validity period for enrollment certificates).
- EEs have the ability to generate a new verification key pair for the new enrollment certificate (no key injection).
- Some EEs have limited network connectivity, therefore the steps of initiating a re-enrollment request, downloading the new enrollment certificate, and validating the new enrollment certificate shall be completed as an asynchronous process.
- An EE may request re-enrollment at any time
- An EE will only possess one valid enrollment certificate at a time, and may only make a single re-enrollment request using its currently valid enrollment certificate.
- The RA can store at least two enrollment certificates for each EE: The current enrollment certificate and the new enrollment certificate.
 - The existence, or lack thereof, of a stored new enrollment certificate provides a mechanism to track the current stage of re-enrollment.

Design

Due to the fact that some EEs may have limited network connectivity, the re-enrollment process takes place in two phases:

1. The EE contacts the RA to initiate a re-enrollment request. If the RA accepts the request, it will inform the EE of a time when it may come back to download the new enrollment certificate.
2. The EE returns to the RA to download a new enrollment certificate

This approach is meant to match the process used to request and download pseudonym certificates (see [Use Case 3: OBE Pseudonym Certificates Provisioning](#)). In practice, a re-enrollment request can be sent and the new enrollment certificate retrieved at the same time the EE is requesting, downloading, or topping off its pseudonym certificates. Note that, as described in [Step 3.1: Request for Pseudonym](#)

[Certificates](#), an EE must update its LPF and LCCF files any time it connects to the RA. If multiple transactions are performed during the same session, then this step only needs to be performed once.

The following sections outline these steps in detail.

EE Initiates the Re-enrollment Request

If an EE possesses a valid enrollment certificate and has not yet requested re-enrollment, then it may perform the following during its next transaction with the RA:

1. Create a new verification key pair and use it to construct an enrollment certificate request with the same properties (same PSID/SSP) used in the original enrollment certificate.
 - a. The only changes allowed in the new CSR is the validity period for the certificate, with the start time of the new certificate being set to the expiry time of the existing certificate. See [Use Case 2: OBE Bootstrapping \(Manual\)](#) for details on formatting the CSR.
 - b. The enrollment certificate request is signed using the new verification key.
2. Construct a new signed message containing the new enrollment certificate request and sign that message with the current enrollment certificate private key. This is a re-enrollment request.
3. Send the re-enrollment request to the RA, using the current enrollment certificate to authenticate to the RA. The RA will validate the request (see below) and reply to the EE with a time indicating when the EE can return to download the new certificate and a hash of the request which must be used to retrieve the new certificate. This mirrors the process used to schedule pseudonym certificate downloads ([Use Case 3: OBE Pseudonym Certificates Provisioning](#)). Note that after reconstructing the new enrollment private key, the EE shall delete the ephemeral key pair that was used in the request.

RA Processes EE's Request and ECA's Response

Upon receiving a re-enrollment request from the EE, the RA performs the following steps:

1. Perform the following checks on the re-enrollment request:
 - a. Validate that the EE's current enrollment certificate has not been blacklisted.
 - b. Ensure that the RA database does not already contain a new enrollment certificate or scheduled re-enrollment request for the EE.
 - c. Validate the "outer" signature on the re-enrollment request message using the public key in the currently valid enrollment certificate.
 - i. Note: The ECA will validate the "inner" signature on the enrollment certificate request (the payload of the message) using the verification public key in the message. There is no need for the RA to check this signature.
 - d. Verify that the requested start time in the re-enrollment request matches the expiration date of the currently active enrollment certificate.
 - e. Verify that the re-enrollment request has the same PSID / SSP attributes as the current enrollment certificate.
2. Store the re-enrollment request in the database. The presence of a re-enrollment request in the database signifies that the EE has a re-enrollment request in progress.
3. Respond to the EE with a requestHash and eCertDLTime to schedule the download of the new enrollment certificate.
 - a. The eCertDLTime may be any time that is less than three (3) years prior to the expiration date of the current enrollment certificate. This ensures that the currently active ECA used by the RA for re-enrollment will have a valid life span sufficient to generate a new enrollment certificate with a full life span. See [PoC Certificate Expiration Timelines](#) for details on the relationship of these certificate validity periods.

4. Schedule a time to activate the re-enrollment request shortly before the eCertDLTime that was calculated in step 3.
 - a. The amount of time allotted for this procedure is implementation dependent. It is recommended that the RA design account for the work load of the RA and the accompanying ECA to ensure that the new enrollment certificate is available when the EE returns to download.
5. Sign the re-enrollment request using the RA private key and forward the signed request to the ECA.
6. Upon receiving the EE's new enrollment certificate from the ECA, store it in the database (replacing or removing the pending re-enrollment request and storing the new enrollment certificate); or, if an error is returned, store the error message in place of the new certificate. Create a relation between the previous enrollment certificate and the new enrollment certificate for revocation and pseudonym certificate download purposes.
7. Once the current enrollment certificate has expired, the RA shall delete it from the database. After this happens, the RA will have only one enrollment certificate for the EE which makes it possible for the EE to request the next enrollment certificate.

ECA Processes New Enrollment Request

Upon receiving an enrollment certificate request from the RA, the ECA performs the following steps:

1. Validate the RA signature.
2. Verify the signature that was created by the EE using the validation private key on the validation public key.
 - a. This step proves that the entity that generated the request was in possession of the validation private key.
3. Validate the validity period of the certificate request.
 - a. Note: The ECA may be issued under a new Root CA and ICA than the EE's current enrollment certificate or the RA certificate. This is OK as long as the ECA can validate the RA signature and the validity periods of the new enrollment certificate are within the ECA's validity period.
4. Generate a new enrollment certificate and sends it back to the RA for delivery to the EE; or, return an error to the RA.

Diagrams

The figure below shows the relationship of the RA and ECA in the re-enrollment process. The next figure is a process diagram that outlines the overall re-enrollment procedure.

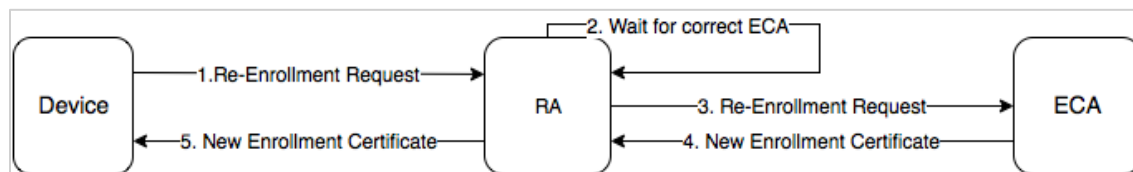


Figure 62 Role Of The RA And ECA In Re-enrollment

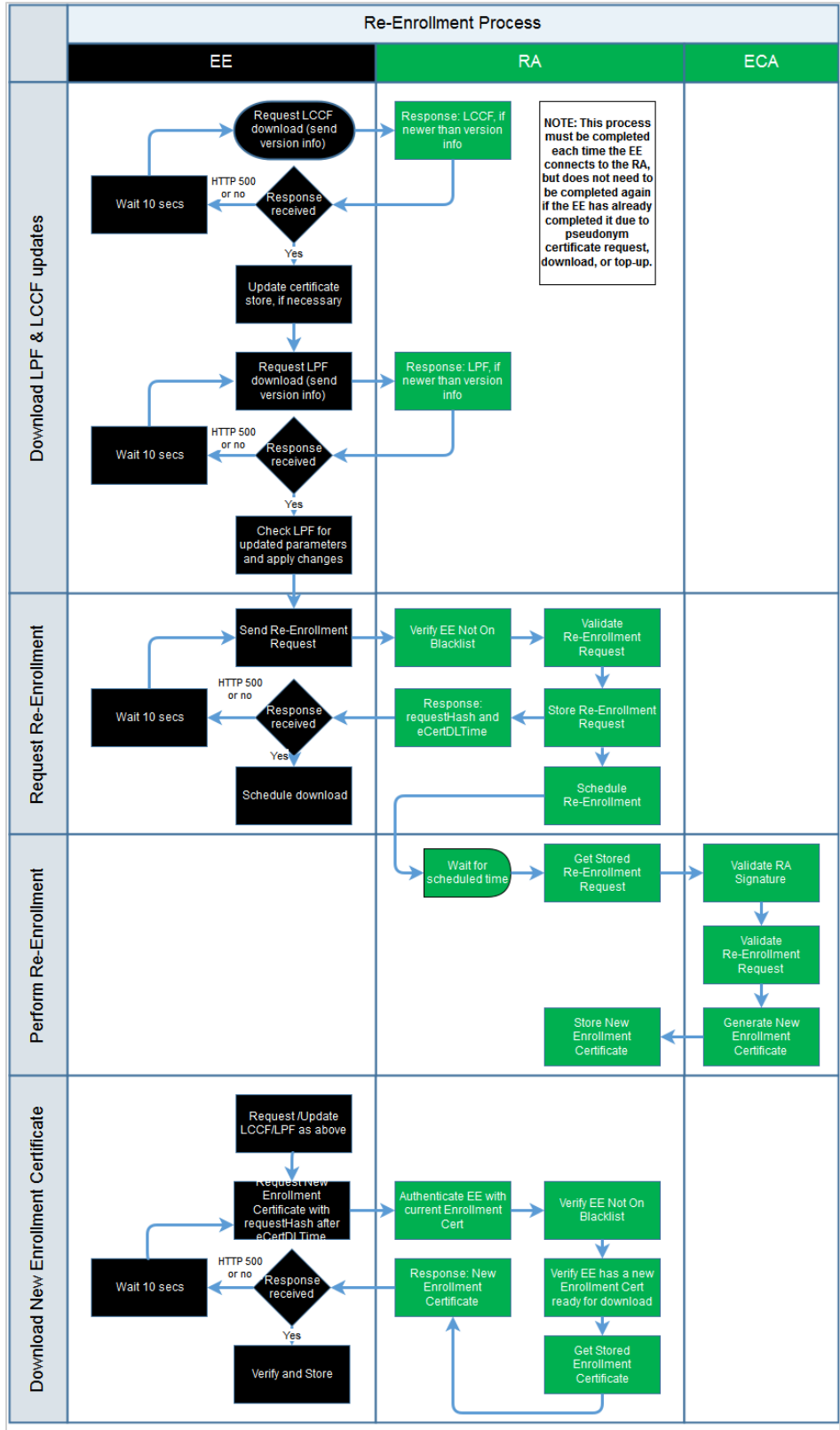


Figure 63 Re-enrollment Process Diagram

Requirements

Key	Status	Summary	Description	Justification	Notes	Component/s																					
SC MS-341	EE REQUIREMENT	EE TLS Cipher Suite	<p>The EE shall support at least the following TLS cipher suites for all communications to SCMS components:</p> <table border="1"> <thead> <tr> <th>IANA Value</th> <th>Description</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0xC0,0x23</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x24</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2B</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0x2C</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384</td> <td>RFC5289</td> </tr> <tr> <td>0xC0,0xAC</td> <td>TLS_ECDHE_ECDSA_WITH_AES_128_CCM</td> <td>RFC7515</td> </tr> <tr> <td>0xC0,0xAD</td> <td>TLS_ECDHE_ECDSA_WITH_AES_256_CCM</td> <td>RFC7515</td> </tr> </tbody> </table>	IANA Value	Description	Reference	0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289	0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289	0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289	0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289	0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515	0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515	This is the requirement for the SSL transport tunnel.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
IANA Value	Description	Reference																									
0xC0,0x23	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	RFC5289																									
0xC0,0x24	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	RFC5289																									
0xC0,0x2B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	RFC5289																									
0xC0,0x2C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	RFC5289																									
0xC0,0xAC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	RFC7515																									
0xC0,0xAD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	RFC7515																									
SC MS-411	EE REQUIREMENT	EE Authentication to RA for Request	The EE shall authenticate its requests with its enrollment certificate and signed timestamp to avoid replay attacks on the RA.	Messages from EEs to an RA must be secure against replay attacks. The signed time stamp from the EE enables the RA to validate the freshness of EE requests.	This is out of scope since it defines EE's behavior. In the case of re-enrollment (Use Case 22), the EE must use the current, active enrollment certificate to authenticate to the RA.	On-board Equipment (OBE), Roadside Equipment (RSE)																					
SC MS-459	CLOSED	OCSP: Stapled for RA to OBE	The RA shall respond to an OBE request for an OCSP stapled certificate.	Most OBEs do not have access to CRL updates or a reliable network connection to an OCSP server, so the RA must provide an OCSP stapled response so that the OBE can validate the RA's TLS certificate.	OCSP stapling provides improved performance compared to CRLs. OCSP stapling is specified in RFC 6066 , Section 8. The RA will be able to respond to the OBE's request for an OCSP stapled certificate. The RA itself will rely on an OCSP service to sign its certificate validation request, which it will	RA																					

Key	Status	Summary	Description	Justification	Notes	Component/s
					return to the EE. For the PoC, the RA will refer to an X.509 CRL to validate certificates of SCMS back-end components (MA, LA, and PCA). OCSP will not be used for back-end component certificate validation.	
SC MS-507	TESTS PASSED	Maintain an Internal Blacklist	RA shall maintain an Internal Blacklist and keep it updated based on the communications with the MA.	So that revoked EEs are not able to authenticate with the RA anymore	Every logical RA has its own internal blacklist that is not shared with anyone else. To prevent compromised components to speak with the RA, the RA needs to validate against the SCMS component CRL (compare https://jira.campllc.org/browse/SCMS-859 SCMS-504) and the X.509 CRL (https://jira.campllc.org/browse/SCMS-405).	RA
SC MS-512	CLOSED	Policy file	RA shall always provide a local policy file (LPF) available for download by EE.	There is always a global configuration available, and that configuration shall be current.	Note that LPF might have the same content as the global policy file (GPF).	RA
SC MS-513	CLOSED	RA downloads via TCP/IP	RA shall provide downloads over TCP/IP.	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc.	RA
SC MS-514	CLOSED	RA download via HTTPS	RA shall provide downloads over HTTPS (TLS).	To utilize standard internet protocols for the download process.	Downloads could be e.g., policy file, Global Certificate Chain File, certificates, .info file etc. TLS will provide encryption (https://jira.campllc.org/browse/SCMS-537 and RA-EE authentication (https://jira.campllc.org/browse/SCMS-539 . IEEE 1609.2 certificates within a TLS session will be used for EE-RA authentication (SCMS-538).	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-515	CLOSED	RA requires EE authentication	The RA shall require EE authentication for authenticated transactions.	To ensure that only a proper EE can send requests, download certificates or files.	It is not cost effective to provide OBEs with TLS certificates currently. Instead, the OBE will use TLS to authenticate the other endpoint (as a server) and will use its SCMS certificate to identify itself. EE authenticates via its IEEE 1609.2 enrollment certificate. The details of the authentication process are defined EE-RA Communications - General Guidance	RA
SC MS-517	CLOSED	Tunneling through LOP	RA shall provide downloads only via a LOP interface, which removes all location information from the incoming request.	to anonymize the location of EEs.		RA
SC MS-521	CLOSED	Acknowledge request	RA shall acknowledge the receipt of EE's request with a TCP ACK within a specified amount of time, currently set to be 1 sec.	So that EEs know that RA received their request.		RA
SC MS-522	EE REQUIREMENT	Retry request	EE shall retry, if it does not receive a response from RA (file download, TCP ACK, RA accept request ACK, HTTP 500, or HTTP 304) within a specified amount of time, currently set to be 10 sec from the time of request.	To ensure that the request is received by the RA.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-523	EE REQUIREMENT	Number of retries	EE shall limit the number of retries to a maximum of 10 in a 60 minute period	To reduce resource usage, EEs shall limit the number of retries.	This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-529	CLOSED	Store enrollment certificate and butterfly parameters	RA shall store enrollment certificate and butterfly parameters for each OBE for its lifetime.	so that OBE can be revoked properly. Arbitrary number based on historical trends for vehicle ownership. For example, collector vehicles that are kept	PoC will only store 3 years	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
				on the road for longer than typical vehicles.		
SC MS-537	CLOSED	RA-to-EE encryption	The RA-to-EE communication shall be encrypted.	To avoid that an adversary is able to read EE's enrollment certificate (protect location privacy) or, in case of pseudonym certificates, that an adversary is able to read PCA-encrypted pseudonym certificates.	For pseudonym certificates, this counters a somewhat exotic attack: if an attacker eavesdrops all individually encrypted pseudonym certificates (encrypted by PCA to EE), and then later extracts the Butterfly keys (e.g., after the car arrived on the junk yard), the attacker is able to track the target vehicle in a retrofit manner assuming that attacker has access to a large database of tracking data. For other certificates, this is just an add-on security layer.	RA
SC MS-539	EE REQUIREMENT	RA authentication to EE	The EE shall require RA Authentication before any communication starts.	EE checks whether it talks to proper RA before communication starts and to avoid sending its enrollment certificate to a malicious RA. RA authenticates via its TLS X.509 certificate. The details of the authentication process are defined in EE-RA Communications - General Guidance	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-541	EE REQUIREMENT	OCSP stapling - EE	The EE shall use the TLS Certificate Status Request extension (OCSP stapling) to verify RA revocation status.	To avoid connecting to a revoked and potentially rogue RA.	This is out of scope since it specifies EE's behavior. If EE does not support this feature, the following might happen: An adversary that extracted the RA's private key and that successfully spoofed DNS is able to learn EE's enrollment certificate (but not EE's private key).	On-board Equipment (OBE), Roadside Equipment (RSE)

Key	Status	Summary	Description	Justification	Notes	Component/s
					OCSP stapling is specified in RFC 6066 , Section 8.	
SC MS-709	EE REQUIREMENT	Check for and Download Policy Updates	EE shall check for and download policy updates upon establishing communications with the RA	It is necessary to ensure that the EE is always using the latest policy for new downloaded certificates. Policy definition details are available at Use Case 18: Provide and Enforce Technical Policies .	If no policy file is available on the EE, the EE is allowed to make a download attempt at any time. This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-754	EE REQUIREMENT	Sign certificate request	The EE shall sign certificate requests with its enrollment certificate.	So that RA can verify that the certificate request was not been modified in transit and to verify that the certificate request is originating from a valid EE	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-768	CLOSED	RA - Local Certificate Chain File	RA shall provide a Local Certificate Chain File to EEs for download.	To enable EEs to verify certificates without further CA certificate downloads. If the file name of the Global Certificate Chain File indicates a new version, the RA will update its Local Certificate Chain File with the new chain information, as appropriate for the EEs under its jurisdiction. EEs send their current LCCF's version number in the download request to RA and the response will include a newer LCCF if available.		RA

Key	Status	Summary	Description	Justification	Notes	Component/s
SC MS-776	EE REQUIREMENT	Encrypt certificate request	The EE shall encrypt the request using the RA certificate.	So that the request is shared confidentially between the EE and RA.	This is out of scope since it defines EE behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-952	EE REQUIREMENT	Error code: eePolicyFileDownloadFailed	EE shall log the error code in EE's error log file, if EE is not able to download the local policy file (e.g., because there is none or it is corrupted).	As the policy file is essential for the system to work correctly and contains security relevant information, it is important to have an error handling whenever the EE is not able to get the latest version of that file.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1189	EE REQUIREMENT	Trust Chain Broken - EE	The EE shall not attempt to request or download pseudonym certificate batches, OBE identification certificate files, RSE application certificate, or a new enrollment certificate, if any component in the trust chain of EE's enrollment certificate is revoked. In this case, EE also shall not attempt to download a local policy file or local certificate chain file from RA.	To reduce resources, since RA will reject request.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS-1203	CLOSED	Check time stamp	RA shall check the signed (by EE) time-stamp and allow a tolerance of 5 seconds.	To counter replay or delay attacks.		RA
SC MS-1204	CLOSED	Check blacklist	RA shall reject EE request and respond with HTTP 500, if EE is listed on its blacklist.	To reject request, and not provide any useful information to EE.	If EE is listed, RA will reject the connection. Otherwise, RA will proceed with the authentication process. The Internal Blacklist Manager (IBLM) of the Misbehavior Authority (MA) updates the RAs on which devices to exclude from granting certificates. Therefore, it sends out revocation information (e.g., linkage information, certificate digest, etc.) that allows the RA	RA

Key	Status	Summary	Description	Justification	Notes	Component/s
					to identify the enrollment certificate of the corresponding device and put it on the internal blacklist. The RA does not send out enrollment certificates to the IBLM.	
SC MS-1353	EE REQUIREMENT	EE request LCCF from RA	The EE shall check for an updated Local Certificate Chain File (LCCF) upon establishing communications with the RA	To be able to verify SCMS certificates based on their certificate chain.	All the certificate chains will contain certificates up to the root CA including elector endorsement for the root CA certificate. This is out of scope since it defines EE behavior	On-board Equipment (OBE), RA, Roadside Equipment (RSE)
SC MS-1377	CLOSED	RA check whitelisted ECA	RA shall validate that the enrollment certificate used by the EE for authentication is issued by a whitelisted ECA.	To ensure that only a proper EE can send requests, download certificates or files.	Whitelist defined in SCMS-1371	RA
SC MS-1419	CLOSED	ECA issues implicit certificates	ECA shall issue implicit OBE and RSE enrollment certificates	To save storage space and over-the-air bytes		ECA
SC MS-1600	CLOSED	Enrollment certificate lifetime	ECA shall issue Enrollment Certificates with an expiration date on or before 00:00:00 UTC January 1, 2025.	To avoid any need to update enrollment certificates during the CV-Pilot project.	Maximum life span 1,084 sixtyHours. This is for CV-Pilot only.	ECA
SC MS-1906	EE REQUIREMENT	Enrollment certificate corresponds to the private key	The enrollment key-pair generator (OBE, RSE, or DCM) shall check that the enrollment certificate corresponds to the private key	This is necessary because otherwise the device won't be able to use the enrollment certificate for requesting pseudonym/identification/application certificates.	If re-enrolling, no DCM is available and this check must be done by the EE.	DCM, On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-1907	EE REQUIREMENT	Enrollment certificate verification	The enrollment key-pair generator (OBE, RSE, or DCM) shall check that the enrollment certificate correctly verifies, including building a chain back to the root CA.	This is necessary because otherwise the device won't be able to use the enrollment certificate for requesting pseudonym/identification		DCM, On-board Equipment (OBE), Roadside

Key	Status	Summary	Description	Justification	Notes	Component/s
				n/application certificates.		Equipment (RSE)
SC MS-1910	EE REQUIREMENT	Verification key pair generation algorithm	EE shall generate the verification key pair using an algorithm approved for use within the SCMS.	Because only those algorithms will be supported by the SCMS.	See Approved Cryptographic Algorithms This is out of scope as it defines EE behavior.	On-board Equipment (OBE), Roadside Equipment (RSE)
SC MS-2475	REVIEW	Re-Enrollment Validate Current Enrollment Cert	RA shall validate the signature of the current enrollment certificate in order to initiate a re-enrollment request.	The re-enrollment process requires a currently valid enrollment certificate.	When an EE is initially enrolled we require a secure connection to the DCM. For re-enrollment, this is replaced by the authenticated, current enrollment certificate.	RA
SC MS-2476	REVIEW	Store new enrollment cert for download	The RA shall store the new enrollment certificate until it is fetched by the EE.	The new enrollment certificate is not necessarily generated at the time of request by the EE.		RA
SC MS-2477	REVIEW	Keep new enrollment cert	The RA shall allow the EE to re-download its new enrollment certificate provided the device's credentials are still valid and not expired.	To allow for recovery after data loss.		RA
SC MS-2478	REVIEW	PSID in re-enrollment request must match current enrollment PSID	RA shall verify that the PSID in the re-enrollment request matches the PSID in the current enrollment certificate.	New enrollment certificate must have identical permissions.		RA
SC MS-2479	REVIEW	Re-Enrollment certificate lifetime	RA shall ensure that the new enrollment certificate has the same validity period as the current enrollment certificate, with the start time of the new enrollment certificate equal to the expiry time of the current enrollment certificate.	The certificate validation chain for the new enrollment certificate must be valid for its entire lifetime.		RA
SC MS-2480	REVIEW	Store all re-enrollment requests / certificates	The RA shall keep a record of all pending enrollment requests/certificates. Once an enrollment certificate expires, there is no need to store it.	To track the status of enrollment and to detect duplicate re-enrollment requests.		RA
SC MS-	REVIEW	Schedule generation of new	RA shall schedule a time to forward the re-enrollment request to the ECA at a time that is no sooner than	ECA at time of request may not have validity	The 2 year overlap is based on the current design of a 6 year	ECA, RA

Key	Status	Summary	Description	Justification	Notes	Component/s
248 1		enrollment certificate	two years after the start date of the current enrollment certificate.	period that covers new enrollment cert. Must wait for new ECA to come online.	enrollment cert and an 11 years ECA cert with 2 year overlap (for ECA certificates).	
SC MS- 248 2	REVIEW	Delete pending re-enrollment requests upon blacklisting	RA shall delete any pending re-enrollment request or stored new enrollment certificate, if the corresponding EE becomes blacklisted.	A blacklisted EE cannot authenticate with the RA and should not be able to obtain a new enrollment certificate.		RA
SC MS- 248 3	EE REQUIREMENT	Secure Key Generation	EE shall generate the private key for use in a new enrollment certificate in a hardware secure module.	The maintain confidentiality of private keys.		On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS- 261 0	EE REQUIREMENT	Use FQDN found in certificate	EES shall use the FQDN specified in the "id" field of the SCMS component certificate to contact the component.	The IP address of SCMS components are not guaranteed to be static and may change at any time.	This is out of scope since it defines EE's behavior.	On-board Equipment (OBE), Road-side Equipment (RSE)
SC MS- 261 1	REVIEW	Store enrollment certificate	RA shall store the following for each OBE: <ul style="list-style-type: none"> All non-expired enrollment certificates The most recent expired enrollment certificate 	So that OBEs can be revoked or re-enrolled properly.		RA

Table 62 Use Case 20.1 - Requirements

Software Design – MA and RA Services Views

MA - Services View

Please refer to the [Common - Services View](#) section for an introduction of the Services View.

General Notes

All Misbehavior Authority Services use the same scheme (**https**) and port **8894**. That is, all the requests to MA will have URLs that look like:

https://<SERVER>:8894/<PATH>

Where <SERVER> is the IP or host name, and PATH is the name of the service.

For all the services, the HTTP Content-Type is set to application/octet-stream.

No information is returned in case of error, just an HTTP code of 500.

Services Summary for EE-MA Communications

Service Name	<PATH>
MA - Download CRL	/download-crl/

MA - Download CRL

EEs use this service to download the CRL File.

PORT	8894
PATH	/download-crl/
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	Empty
HTTP Response Body	ASN.1 Serialized CRL File as defined by the CompositeCrl ASN.1 definition

Preconditions

1. CRL has been generated by the CRL Generator and stored in the CRL Store

Postconditions

1. Returned file contains ASN.1 Serialized Composite CRL File as defined by the CompositeCrl ASN.1 definition

Quality of Protection

- MA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- This download service **IS NOT** protected via download authentication message as other download service. This service does not authenticate the downloading entity.

RA - Services View

Please refer to the [Common - Services View](#) section for an introduction of the Services View.

General Notes

All Registration Authority Services use the same scheme (**https**) and port **8892**. That is, all the requests to RA will have URLs that look like:

https://<SERVER>:8892/<PATH>

Where <SERVER> is the IP or host name and PATH is the name of the service.

For all the services, the HTTP Content-Type is set to application/octet-stream.

No information is returned in case of error, just an HTTP code of 500, in a production environment. QA environment will give back errors as outlined in [Overview of Used Error Codes](#).

Services Summary for EE-RA Communications

Service Name	<PATH>
Request Application Certificate Provisioning	/provision-application-certificate
Download .info file	/download/info
Download local policy file	/download/policy/local
Download Pseudonym Certificate Batch	/download/batch
Retrieve Registration Authority Certificate	/retrieve-ra-certificate
Request Identification Certificate Provisioning	/provision-identity-certificate
Download Identification Certificate	/download/identity-certificate
Request Pseudonym Certificate Batch Provisioning	/provision-pseudonym-certificate-batch
Download Application Certificate	/download/application-certificate
Download Local Certificate Chain File	/download/local-certificate-chain
Submit Misbehavior Report	/process-misbehavior-report

Table 63 Services Summary For EE-RA Communications

RA - Request Pseudonym Certificate Batch Provisioning

OBEs use this service to request the initial batch of Pseudonym Certificates. After the initial batch is requested, subsequent batches are automatically provisioned.

PORT	8892
PATH	/provision-pseudonym-certificate-batch
HTTP Method	POST
HTTP Request Body	ASN.1 serialized SecuredPseudonymCertProvisioningRequest
HTTP Response Body	SignedPseudonymCertProvisioningAck with a requestHash property containing the lower 8 bytes of the request hash. This value will identify this device for the download of the requested certificate. The reply property contains a PseudonymCertProvisioningAck with a certDLTime property containing the expected time for download of the requested batch, and a certDLURL property containing the URL where the batch can be downloaded.

Preconditions

1. Policy referenced in the request message is previously known
2. PLV Cache has at least one PLV chain
3. Device is not revoked

Postconditions

None.

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. This service will be used once for each initial provisioning request for each new OBE. There may also be a very small addition of OBEs re-requesting provisioning in order to update their

parameters. However, this should be a low enough volume to have no significant impact on these calculations.

Quality Metric	1 Year	3 Years	5 Years	10 Years
Throughput	(17,000,000 new vehicles) / 31,557,600 seconds/year = .5 batch requests / second	(17,000,000 new vehicles) / 31,557,600 seconds/year = .5 batch requests / second	(17,000,000 new vehicles) / 31,557,600 seconds/year = .5 batch requests / second	(17,000,000 new vehicles) / 31,557,600 seconds/year = .5 batch requests / second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.
- Incoming message is encrypted (within the ASN.1 message structure) with the RA Component certificate public key

RA - Download .info File

OBEs use this service to determine the earliest date on which a new batch of pseudonym certificate will become available. There will be a .info file for each device containing this information.

PORT	8892
PATH	/download/info
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	HTTP Header 'Download-Req' containing a Base64 encoded ASN.1 serialized SecuredAuthenticatedDownloadRequest , containing a SignedAuthenticatedDownloadRequest , containing a ScopedAuthenticatedDownloadRequest , containing an AuthenticatedDownloadRequest with a <i>filename</i> property matching the regular expression <code>[0-9A-F]{16}\.info</code> . That is, the name part of the file name is the 16 hexadecimal digits Request Hash obtained during initial provisioning request of this device.
HTTP Response Body	File containing a time stamp of when the next batch is <i>estimated</i> to be available. Due to varying system load, an exact download time cannot be provided. If the device receives error 5065 then it should reschedule the download for a later time.

Preconditions

1. Device had previously requested a certificate batch
2. Time stamp in the request *AuthenticatedDownloadRequest* is not more than 5 seconds apart from the server's time. (Controlled by the *ra.client-signature-ttl-sec* configuration setting.)

Postconditions

1. RA returned a file containing a single IEEE 1609 format Time32 time stamp that the device will use to schedule a subsequent "top-up" download

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

$$f(\text{year}) = \left(\sum_{i=1}^{\text{year}-1} 17\text{million} \right) + \frac{17\text{million}}{2}$$

Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. The number of vehicles that potentially invoke this service in a week is a function of the number of years in service:

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
Throughput	Assuming half of all new cars plus all existing vehicles need to check for notifications in the same week.	8,500,000 new vehicles / 604,800 seconds/week = 14 requests / second	(8,500,000 new vehicles + 34,000,000 existing vehicles) / 604,800 seconds/week = 70 requests / second	(8,500,000 new vehicles + 68,000,000 existing vehicles) / 604,800 seconds/week = 126 requests / second	(8,500,000 new vehicles + 153,000,000 existing vehicles) / 604,800 seconds/week = 267 requests / second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDHE-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.

RA - Download Local Policy File

EEs use this service to download a local policy file.

PORT	8892
PATH	/download/policy/local
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	Optionally, the request may include the standard HTTP Header 'If-None-Match' containing the filename of the local policy file that the EE currently possesses, excluding any path . For example: If-None-Match: "local_policy_0001_0001.oer" This is used to prevent the same policy file from being downloaded by the device multiple times.
HTTP Response Body	File containing the local policy represented by an OER encoded ASN.1 serialized SignedLocalPolicyFile . The file name returned is of the form: local_policy_<X>_<Y>.<Z> Where: <ul style="list-style-type: none"> • X is the global version number • Y is the local policy version • Z is one of the permitted encoding formats (oer) from the file name in the request message OR An HTTP code of 304 (Not Modified), if the provided file name in the 'If-None-Match' header matches the current version available on the RA server.

Preconditions

None

Postconditions

1. Returned file contains policy that the device will use

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. The number of vehicles that potentially invoke this service in a week is a function of the number of years in service:

$$f(\text{year}) = \left(\sum_{i=1}^{\text{year}-1} 17\text{million} \right) + \frac{17\text{million}}{2}$$

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
Throughput	Assuming half of all new cars plus all existing vehicles need to download local policy in the same week.	8,500,000 new vehicles / 604,800 seconds/week = 14 requests / second	(8,500,000 new vehicles + 34,000,000 existing vehicles) / 604,800 seconds/week = 70 requests / second	(8,500,000 new vehicles + 68,000,000 existing vehicles) / 604,800 seconds/week = 126 requests / second	(8,500,000 new vehicles + 153,000,000 existing vehicles) / 604,800 seconds/week = 267 requests / second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256

RA - Download Pseudonym Certificate Batch

OBEs use this service to download a batch of Pseudonym Certificates for a specific time period.

PORT	8892
PATH	/download/batch
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	<p>HTTP Header 'Download-Req' containing a Base64 encoded ASN.1 serialized SecuredAuthenticatedDownloadRequest, containing a SignedAuthenticatedDownloadRequest, containing a ScopedAuthenticatedDownloadRequest, containing an AuthenticatedDownloadRequest with a <i>filename</i> property of the form [0-9A-F]{16}_{[0-9A-F]{1,8}.zip, where the first group of 16 hexadecimal digits is the device's request hash obtained from the initial provision pseudonym certificate batch request, and the second group of up to 8 hexadecimal digits is the i-value. Example: AB09281C9867DE53_F.zip corresponds to i value 15, for device with request hash AB09281C9867DE53.</p> <p>Range (optional) as defined in RFC 2616: To support partial downloads for resuming interrupted transfers. Examples:</p> <ol style="list-style-type: none"> 1. From byte offset 500 to 700: Range : bytes=500-700 2. Starting from byte offset 1000 to the end: Range : bytes=1000-
HTTP Response Body	If no Range header is present, the entire zip file corresponding to the requested batch. If a Range header is present, the specified bytes of the referenced file.

Preconditions

1. The requested batch has already been generated
2. The requesting device has not been previously revoked

Postconditions

1. The zip file corresponding to the batch specification in the request URL is returned.
2. The content of the zip file is organized as a flat directory containing n files (where $0 \leq n \leq j_max - 1$) with the naming format:
 - a. X_Y (NOTE: no file extension)
 - b. Where X is the i-value representing the SCMS I period in which the certificate is valid in hexadecimal
 - c. Where Y is a sequence of "j" values from $j = 0$ to $j = j_max-1$ in hexadecimal
 - d. Example zip file contents for period $i=55$, $j = 20$:
 - i. 37_0
 - ii. 37_1
 - iii. ...
 - iv. 37_12
 - v. 37_13
 - e. The contents of each individual file within the .zip is a binary OER encoding of the appropriate [SignedEncryptedCertificateResponse](#).

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. The number of files downloaded in a year is a function of the number of years in service:

$$f(year) = \left(\sum_{i=1}^{year-1} (17million \times 52weeks) \right) + (17million \times 156weeks)$$

which assumes 17 million vehicles are added each year.

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
Throughput	Assuming 17 million new vehicles downloading the initial three years' worth of certificates (156 files) plus old cars downloading one year's worth of certificates (52 files).	With no previous year vehicles: 17m x 52 weeks * 3 years = 2,652 million files Divided by the number of seconds in a year:	With and two years' worth of old vehicles (34 million): 17m (new) + 34m (old) Old cars only download one year's worth of certificates (52 files) while new cars download	With four years' worth of old vehicles (68 million): 17m (new) + 68m (old) Old cars only download one year's worth of certificates (52 files) while new cars download	At the end of the first 10 years, there will be a total of 170 million cars in the system out of which, 153 million will be old vehicles: 17m (new) + 153m (old) The new cars will be downloading three years' worth of certificates (156 weeks), while the rest of the vehicles will be topping up only (52 weeks). Since

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
		2,652 million files / 31,557,600 seconds = 85 files per second	three years' worth of certificates (156 files) so: (17m * 156 files) + (34m * 52 files) = 2,652 million files + 1,768 million files = 4,420 million files Divided by the number of seconds in a year: 4,420 million files /31,557,600 seconds = ~141 files per second	three years' worth of certificates (156 files) so: (17m * 156 files) + (68m * 52 files) = 2,652 million files + 3,536 million files = 6,188 million files Divided by the number of seconds in a year: 6,188 million files /31,557,600 seconds = ~197 files per second	each file contains one week's worth of certificates, we can express this in number of files: (17m x 156 files) + (153m x 52 files) = (2,652mf + 7,956mf) = 10,608 million files Divided by the number of seconds in a year: 15,028 million files / 31,557,600 seconds = ~337 files per second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.

RA - Retrieve Registration Authority Certificate

EEs use this service to refresh its locally cached RA certificate.

PORT	8892
PATH	/retrieve-ra-certificate
HTTP Method	POST
HTTP Request Body	ASN.1 serialized <i>SecuredRACertRequest</i> PDU Message.
HTTP Response Body	Serialized IEEE 1609.2 certificate

Preconditions

None

Postconditions

1. The 1609 certificate is returned to the device.

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. The number of vehicles likely to download the RA certificate is a function of the number of years in service:

$$f(\text{year}) = \left(\sum_{i=1}^{\text{year}-1} 17\text{million} \right) + \frac{17\text{million}}{2}$$

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
Throughput	Assuming half of all new cars plus all existing vehicles need to refresh their copy of the RA certificate in one week.	8,500,000 new vehicles / 604,800 seconds/week = 14 requests / second	(8,500,000 new vehicles + 34,000,000 existing vehicles) / 604,800 seconds/week = 70 requests / second	(8,500,000 new vehicles + 68,000,000 existing vehicles) / 604,800 seconds/week = 126 requests / second	(8,500,000 new vehicles + 153,000,000 existing vehicles) / 604,800 seconds/week = 267 requests / second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256

RA - Request Identification Certificate Provisioning

OBEs use this service to request the new identification certificates. After the initial batch is requested, subsequent batches are automatically provisioned.

PORT	8892
PATH	/provision-identity-certificate
HTTP Method	POST
HTTP Request Body	ASN.1 serialized SecuredIdCertProvisioningRequest
HTTP Response Body	ASN.1 serialized SignedIdCertProvisioningAck with a requestHash property containing the lower 8 bytes of the request hash. This value will identify this device from this point on, and it is to be used in subsequent download calls. The reply property contains a PseudonymCertProvisioningAck with a certDLTime property containing the expected time for download of the requested certificate and a certDLURL property containing the URL where the certificate can be downloaded.

Preconditions

1. Policy referenced in the request message is previously known
2. EE is not revoked

Postconditions

None.

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.

RA - Download Identification Certificate

OBEs use this service to download a previously requested Identification Certificate.

PORT	8892
PATH	/download/identity-certificate
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	HTTP Header 'Download-Req' containing a Base64 encoded ASN.1 serialized SecuredAuthenticatedDownloadRequest containing a SignedAuthenticatedDownloadRequest containing a ScopedAuthenticatedDownloadRequest containing an AuthenticatedDownloadRequest with a <i>filename</i> property of the form [0-9A-F]{16}_i.zip, where the group of 16 hexadecimal digits is the device's request hash obtained from the provision identification certificate request , and i is a file iterator in hexadecimal starting at 0 (both are case insensitive). Example: AB09281C9867DE53_F.zip corresponds to i = 15, for a device with request hash AB09281C9867DE53. There shall be exactly one identification certificate per file. Range (optional) as defined in RFC 2616: To support partial downloads for resuming interrupted transfers. Examples: 1. From byte offset 500 to 700: Range : bytes=500-700 2. Starting from byte offset 1000 to the end: Range : bytes=1000-
HTTP Response Body	If no Range header is present, the entire tar file corresponding to the requested batch. If a Range header is present, the specified bytes of the referenced file.

Preconditions

1. The requested certificate has already been generated
2. The requesting device has not been previously revoked

Postconditions

1. The zip file corresponding to the certificate specified in the request URL is returned.
2. The content of the tar file is organized as a flat directory containing 1 file named as in:
 - o X_i
 - o X shall be the lower 8-bytes of the SHA-256 hash (16 hexadecimal digits) of the device request in hexadecimal (case insensitive)
 - o Where there is no extension

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

For PoC the volume for this interface is still TBD but is not expected to have significant impact on system throughput requirements.

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.

RA - Request Application Certificate Provisioning

RSEs use this service to request new application certificates. After the initial certificate is requested, subsequent certificates are **NOT** automatically provisioned.

PORT	8892
PATH	/provision-application-certificate
HTTP Method	POST
HTTP Request Body	ASN.1 serialized SecuredAppCertProvisioningRequest
HTTP Response Body	ASN.1 serialized SignedAppCertProvisioningAck with a requestHash property containing the lower 8 bytes of the request hash. This value will identify this device for the download of the requested certificate. The reply property contains a PseudonymCertProvisioningAck with a certDLTime property containing the expected time for download of the requested certificate and a certDLURL property containing the URL where the certificate can be downloaded.

Preconditions

1. Policy referenced in the request message is previously known
2. EE is not revoked

Postconditions

None.

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

For PoC the volume for this interface is 50,000 RSEs. This is not expected to have significant impact on system throughput requirements.

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.
- Incoming message is encrypted (within the ASN.1 message structure) with the RA Component certificate public key.

RA - Download Application Certificate

RSEs use this service to download a previously requested Application Certificate.

PORT	8892
PATH	/download/application-certificate
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	HTTP Header 'Download-Req' containing a Base64 encoded ASN.1 serialized SecuredAuthenticatedDownloadRequest , containing a SignedAuthenticatedDownloadRequest , containing a ScopedAuthenticatedDownloadRequest , containing an AuthenticatedDownloadRequest with a filename property of the form [0-9A-F]{16}.zip, where the group of 16 hexadecimal digits is the

PORT	8892
	device's request hash obtained from the initial provision application certificate request . There shall be exactly one application certificate per file. Range (optional) as defined in RFC 2616: To support partial downloads for resuming interrupted transfers. Examples: 1. From byte offset 500 to 700: Range : bytes=500-700 2. Starting from byte offset 1000 to the end: Range : bytes=1000-
HTTP Response Body	If no Range header is present, the entire zip file corresponding to the requested batch. If a Range header is present, the specified bytes of the referenced file.

Preconditions

1. The requested certificate has already been generated
2. The requesting device has not been previously revoked

Postconditions

1. The file corresponding to the certificate specified in the request URL is returned
2. The content of the file is exactly one application certificate file per certificate download file. The content of the certificate file is the binary representation of the application certificate named as in:
 - o X
 - o X shall be the lower 8-bytes of the SHA-256 hash of the device request in hexadecimal (case insensitive 16 hexadecimal digits)
 - o Where there is no extension

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

For PoC, the volume for this interface is still TBD but is not expected to have significant impact on system throughput requirements.

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.

RA - Download Local Certificate Chain File

EEs use this service to download a local certificate chain file.

PORT	8892
PATH	/download/local-certificate-chain
HTTP Method	GET
HTTP Request Body	Empty
HTTP Request Headers	Optionally, the request may include the standard HTTP Header 'If-None-Match' containing the file name of the local certificate chains file that the EE currently possesses, excluding any path . For example: If-None-Match: "local_certificate_chains_01_03.oer" This is used to prevent the same file from being downloaded by the device multiple times.

PORT	8892
HTTP Response Body	<p>File containing the local certificate chains represented by an OER encoded ASN.1 serialized ScopedLocalCertificateChainFile. The file name returned is of the form: local_certificate_chains_<X>_<Y>.<Z></p> <p>Where:</p> <ul style="list-style-type: none"> • X is the global certificate chain version, i.e., the <i>cert_chain_file_id</i> parameter found in the Global Policy File • Y is the local certificate chain version • Z is one of the permitted encoding formats (oer) from the file name in the request message <p>OR</p> <p>An HTTP code of 304 (Not Modified), if the provided file name in the 'If-None-Match' header matches the current version available on the RA server.</p>

Preconditions
None

Postconditions
1. Returned file contains SCMS certificates chains that the device will use

Error Handling
See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service
Estimated values are per logical unit, meaning multiple individual nodes can contribute to achieve the desired level of service. The number of vehicles that potentially invoke this service in a week is a function of the number of years in service:

$$f(\text{year}) = \left(\sum_{i=1}^{\text{year}-1} 17\text{million} \right) + \frac{17\text{million}}{2}$$

Quality Metric	Rationale	1 Year	3 Years	5 Years	10 Years
Throughput	Assuming half of all new cars plus all existing vehicles need to download local certificate chain file in the same week.	8,500,000 new vehicles / 604,800 seconds/week = 14 requests / second	(8,500,000 new vehicles + 34,000,000 existing vehicles) / 604,800 seconds/week = 70 requests / second	(8,500,000 new vehicles + 68,000,000 existing vehicles) / 604,800 seconds/week = 126 requests / second	(8,500,000 new vehicles + 153,000,000 existing vehicles) / 604,800 seconds/week = 267 requests / second

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256

RA - Submit Misbehavior Report
EEs use this service to submit a Misbehavior Report (MBR) that RA will forward to the Misbehavior Authority.

PORT	8892
PATH	/process-misbehavior-report
HTTP Method	POST

PORT	8892
HTTP Request Body	ASN.1 serialized <i>SecuredMisbehaviorReport</i>
HTTP Response Body	Empty

Preconditions

1. EE is not revoked

Postconditions

None.

Error Handling

See "RA-EE Errors" in [Overview of Used Error Codes](#)

Quality of Service

For PoC, the volume for this interface will be estimated by the currently underway V2V-CR Project (GMBD), but is not expected to have significant impact on system throughput requirements.

Quality of Protection

- RA protects access with HTTPS (TLS V1.2)
- Supports at a minimum OpenSSL cipher suite ECDHE-ECDSA-AES128-SHA256
- Uses certificate-based client authentication of data signed by the device enrollment certificate, validated at the application layer. This is a supplement to the one-way TLS authentication, to provide two-way authentication with a TLS/1609.2 hybrid scheme.
- Incoming message is encrypted (within the ASN.1 message structure) with the MA Component certificate public key

Cryptography Background

CB1: Cryptographic Services

Standard Services: Confidentiality, Integrity, Authenticity

The standard cryptographic services are confidentiality, integrity, and authenticity. They are provided by the cryptographic mechanisms of encryption and authentication. Two fictional people, Alice and Bob, are used in the following descriptions to help simplify the explanations.

Confidentiality means that when Alice sends a message to Bob, she knows that no one can learn anything (except its length) about the message in transit. Confidentiality is provided by *encryption*.

Integrity means that when Alice sends a message to Bob, she knows that if the message is altered in transit, Bob will be able to detect that the message has been modified; this provides a deterrent to an attacker who may want to modify the message.

Authenticity means that when Alice sends a message to Bob, she knows that Bob can be certain that the message actually came from her.

Authenticity and *integrity* are typically provided together (authenticity is of little use without integrity) by *authentication*.

Cryptographic mechanisms allow Alice and Bob to leverage a small amount of secure information into a large amount of secure data. This small amount of information is a key. For confidentiality, Alice uses a key to encrypt the data and Bob uses a related key to decrypt the data. For authentication, Alice uses a key to apply an authentication code to the data, and Bob uses a related key to check that the code is

valid. Although a great deal of attention is paid to particular encryption algorithms (such as the algorithm by Rivest, Shamir, and Adleman (RSA), the advanced encryption standard (AES), and so on), real-life key management is a much more difficult problem than choosing a cryptographic algorithm, and many more weaknesses are caused in systems by poor key management than by a poor choice of cryptographic algorithm.

Privacy

A main goal of the SCMS is to protect the privacy of drivers. This means that it should provide the following services:

- **Anonymity:** A message should contain no information that explicitly identifies the driver, the passengers, or the vehicle.
- **Unlinkability:** It should be difficult for an eavesdropper to track a driver or vehicle by recording its BSM transmissions.

Unlinkability is not a binary property of the system. For example, an eavesdropper who is able to record all messages sent by a vehicle will be able to track that vehicle by constructing the path indicated by that vehicle's BSMs. However, it is a design goal that the V2V communications system does not increase the risk that an individual may be tracked.

For purposes of this report, the requirement is that if a vehicle's messages contain data that is unique to the vehicle, the data should change frequently such that it is extremely difficult for an eavesdropper to track that vehicle. This in turn means that:

- Any application identifiers should change frequently. This is supported in the TemporaryID field in the BSM.
- Any network identifiers, such as source Media Access Control (MAC) addresses, should change frequently. This is permitted by IEEE Standard 802.11 and actively supported by service primitives in IEEE Standard 1609.4.
- Any cryptographic information unique to the vehicle should change frequently. As discussed below, messages in the system are authenticated by signing them with digital certificates, which are issued by a certificate authority (CA). To meet the requirement, a device must either have multiple digital certificates, or share its certificate with other vehicles. Previous research has concluded that shared certificates are not viable (cf. e.g., Jason J. Haas, Yih-Chun Hu, and Kenneth P. Laberteaux. *The Impact of Key Assignment on VANET Privacy. Security and Communication Networks*. 3(2):233-249, John Wiley & Sons, Ltd., March 2010). Only the case where each device has multiple certificates is considered in this system.
- All identifier changes should be synchronized: if one identifier does not change between observations, the attacker can link even if all other identifiers change.
- The vehicle's privacy should be preserved even if the attacker has inside information from one of the SCMS components.

However, in addition to supporting privacy, the system design has to support identification of misbehaving devices in order to remove them from the system. These two goals are fundamentally in contradiction. This SCMS design allows identification of devices for misbehavior detection purposes only through a series of defined interactions between SCMS components. No individual SCMS component can identify a device, and the information revealed to any SCMS component can be controlled.

The Vehicle Infrastructure Integration Consortium (VIIC) provides a full discussion of the policy requirements arising from this high-level requirement for privacy-by-design.

CB2: Types of Cryptographic Algorithms

There are two different types of keyed cryptographic algorithms, which use very different types of key management. This section discusses those keyed algorithms and also two other important cryptographic primitives, hash functions and random number.

Symmetric Algorithms

In a symmetric algorithm, the sender and receiver use the same key (or keys that are related to each other in some trivial-to-derive way). Alice uses k_1 to encrypt; Bob uses k_1 to decrypt. Alice uses k_2 to authenticate; Bob uses k_2 to validate. Symmetric algorithms have two significant properties:

- They are fast (which translates into implementations being low cost). For example, AES, a symmetric encryption algorithm, can encrypt 81 MB per second on a 2 GHz processor, or generate authentication codes on 1,000,000 messages per second with a size of 100 bytes per message.
- They require *secure, private key exchange*. Before Alice and Bob can use a key k to communicate, they must securely agree on k in such a way that no other party (except perhaps a trusted center) knows k . This means that Alice and Bob must have some kind of pre-existing relationship to use symmetric cryptography.
 - NOTE: In a vehicular setting, vehicles are often encountering each other for the first time and do not have a pre-existing relationship. This is one of the main reasons why symmetric key cryptography is not being considered for use in authenticating V2V safety messages.

Public Key Algorithms

In an *asymmetric or public key algorithm*, the encryption and decryption, or authentication and validation, keys come as a pair, *Pub and Priv*, with the property that they are related but that it is very expensive (in terms of computer time) for someone who only knows *Pub* to work out *Priv*. *Pub* is called the public key. *Priv* is called the private key. The private key is not widely shared and usually known only to the key owner; the public key can be distributed widely. They are used this way:

- For confidentiality: Alice uses Bob's (note, not Alice's) public key to encrypt the message. Only Bob knows his own private key, so only Bob can decrypt the message.
- For authentication: Alice uses her own private key to generate the authentication code – for a public key algorithm, this is called *signing*. Bob uses Alice's (note, not Bob's) public key to validate the authentication code – for a public key algorithm, the authentication code is called a *signature* and the validation is called *verification*. If the signature verifies with Alice's public key, then the signature was generated with Alice's private key and the message was not modified. Because only Alice knows her own private key, that means that Alice generated the signature and so that the message came from Alice. For performance reasons, an actual implementation would perform the signature operation on a checksum of the message only.

Public-key algorithms have two significant properties:

- They are *relatively slow* compared to symmetric algorithms (which translates into implementations being more expensive in terms of processing compared to symmetric algorithms). For example, ECDSA-256, the public key algorithm that is used in the CAMP VSC3 design, can generate about 1500 signatures per second on a 2 GHz processor and can verify only about 300 signatures per second.
- They require *authenticated key exchange*, but the key exchange can be *public*. If Alice has some assurance that a public key belongs to Bob, she can use that key to verify Bob's signed messages or encrypt messages to him even if many other people know the public key as well. Alice knows that a public key belongs to Bob usually because the CA attests to it by signing Bob's public key. Bob's public key is signed by the CA and is referred to as Bob's certificate. So long as Alice and Bob trust the CA and have access to the CA's public key, they can trust that keys signed by the CA are genuine. This makes public key cryptography ideal for settings where two parties encounter each other briefly and need to trust each other's communications, even if they do not have access to an

online key service. This is the relevant setting for V2V communications, which is why CAMP VSC3 and IEEE 1609.2 use public key cryptography.

Hash Functions

There is a third useful type of cryptographic algorithm, known as the hash function. A hash function produces a cryptographically strong, fixed-length checksum of a message. The output from the hash function, often called a hash or a digest, is cryptographically strong in the following senses:

- The output looks random: small changes to the input message result in significant changes to the hash
- It is computationally infeasible to find a message that hashes to a particular hash value. (Hash functions are non-invertible, or have pre-image resistance.)
- It is computationally infeasible to find two messages that hash to the same value. (Hash functions have collision resistance.)
- Hashes are fast, comparable to or faster than symmetric algorithms. In the CAMP VSC3 SCMS, hashes are used for a number of purposes:
 - A truncated hash of a certificate can be used as an identifier in messages signed by that certificate, so that the sender does not have to send the full certificate with every message
 - Messages are hashed before signing them: the (private-key) signature operation is actually applied to the hash of the message but not to the message itself. This has both security and efficiency benefits and is standard practice in cryptographic systems outside the CAMP VSC3 system.
 - Hashes are used to generate linkage values as described in [SCP2: Linkage Values](#)

Random Number Generators

Random number generators are used to generate keys and other random data within a system that uses cryptography. Since the security of a system depends on private and secret keys being unobtainable through unauthorized exposure, it is important that the random number generators used to generate them are good. In this context, “good” means a number of things:

- An attacker must not be able to determine the next output from the random number generator, no matter how much previous output the attacker has seen. This means that the output must be statistically random and contain no bias. If the random number generator is used to generate an integer modulo some modulus n , all numbers between 0 and $n-1$ must be equally probable with no bias towards particular values.
- If the random number generator uses an internal state, an attacker must not be able to guess the internal state of the random number generator and use this to predict output. This means that:
 - The internal state must be large enough to be infeasible to guess by brute force
 - The initialization process that initialized the internal state must be infeasible to reproduce
- If the random number generator uses some hardware-produced randomness source, the output from this source must be infeasible to reproduce

As well as secret and private keys, random number generators are used for other purposes within the SCMS:

- When signing with Elliptic Curve Digital Signature Algorithm (ECDSA), a fresh entirely random number must be generated for each signature with the same key. Repeated random numbers, random numbers with a bias, or random numbers with a known relationship to each other will reveal the private key.

- Random numbers are used by the PCA when creating implicit certificates or when expanding a butterfly signing key (see [SCP1: Butterfly Keys](#)). If these random numbers are not good, it can result in the Registration Authority (RA) being able to track a device, or even the PCA's private key being revealed.
- Random numbers are used to generate Linkage Seeds (LSs) for linkage chains (see [CB3: Public Key Infrastructure](#), [SCP2: Linkage Values](#)). If these random numbers are not good, it can result in a device being trackable by a Linkage Authority (LA) or the PCA.

All SCMS components, as well as EEs, must be equipped with industrial quality random number generators, e.g., one of the [Approved Random Number Generators](#).

CB3: Public Key Infrastructure

In a symmetric key system, each sender and receiver pair needs to share a secret key, thus resulting in a significant amount of shared keys. The great advantage of public key cryptography is that it makes it feasible for parties to communicate securely with each other, even if they have never encountered each other before and do not have access to an online service.

Alice sends a signed message to Bob. Bob can trust this message without having previously seen Alice's certificate if both of these statements are true:

- Alice signed the message, and the signature verifies using Alice's public key from her certificate
- Alice's certificate is signed by the private key, which corresponds to the public key from a CA certificate. Bob already knows the CA certificate and is able to verify Alice's certificate using the CA certificate's public key.

This may be extended. Bob does not need to know the CA certificate that issued Alice's certificate. This CA certificate, call it Certificate (CA1), could have been issued by another CA, call it CA2. If Bob knows Certificate (CA2), and receives both Certificate (Alice) and Certificate (CA1) in the signed message, he can still trust Alice's message by verifying that Alice signed the message, that her certificate was issued by CA1 and that CA1's certificate was issued by CA2. This can obviously be extended any number of times until the certificate chain reaches a root certificate. A root certificate is a certificate that was signed by its own private key. The root key is the key to trusting the entire PKI. The root public key has to be distributed securely so that recipients do not receive the wrong key and so trust the wrong certificates. The root private key also must be protected very carefully – anyone who had access to the private key would in principle be able to set up an entire CA hierarchy made of compromised CAs, which would be trusted by everyone who knew the public key. For this reason in real-world PKI deployments, the root key is used as infrequently as possible and is kept and used on a machine that cannot be accessed from an external network.

The CAMP SCMS design features a CA hierarchy, with:

- A root CA that issues certificates for other CAs but not for vehicles or other end-entities
- Optionally, intermediate CAs (ICAs), which obtain their certificates from other CAs above them and also issue certificates for other CAs rather than end-entities. The advantage of using intermediate CAs is that if an intermediate CA is compromised, it is less catastrophic than if the root CA is compromised, so this gives the system more flexibility to introduce new CAs without running the risks incurred by using the root CA key. It is possible to use intermediate CAs in a cascade, so an intermediate CA is either validated by the root CA or the intermediate CA above it.
- Enrollment authorities that issue enrollment certificates (long-term certificate signing requests) for the end-entities. These enrollment certificates are used only to communicate with the SCMS, not with other vehicles or end-entities. Note: the lifetime of the certificate is currently assumed to be the lifetime of a car (e.g., 30 years). However, this still needs discussion as it influences the size of the internal blacklist and is hence a cost issue. Note: the certificate lifetime and the lifetime of the actual CA do not have to be equal.

- Pseudonym CAs that issue certificates for the applications on the cars

The CAMP SCMS also distinguishes between the CA, which actually signs the certificate and the RA, which approves certificate requests. This results in a system diagram that appears complicated at first glance. In fact, this aspect of the CAMP approach is fully in line with standard PKIs used elsewhere in government and industry. This complexity of the abstract architecture allows for flexibility and robustness in introducing new CAs, retiring old ones, and allowing different organizations to take responsibility for authorizing activities that they properly have jurisdiction over. The initial deployment may not require all the boxes on the diagram to be filled immediately; however, it is important for the initial system to support migration to the full CAMP SCMS architecture, even if this migration happens slowly.

Certificates

A certificate links its holder's public key to a statement about the holder, such as an identity or a list of permissions. The statement is trusted because it is attested to by a CA. A receiver checks that the statement is true about a particular signed message by first using the public key of the CA to verify the certificate and subsequently the sender's public key to verify the signature on the message. If the receiver trusts the CA, and the signature on the message verifies, then the receiver knows that the public key owner signed the message and therefore the statement (identity, permissions, etc.) can be trusted as true about the message sender.

The standard way of creating and trusting a certificate is:

- The certificate contains the public key
- The CA signs the certificate
- The receiver verifies the CA signature on the certificate and the public key holder's signature on the message

This requires two verifications on the receiver's side and further requires (with recommended cryptographic algorithm choices) 64 bytes on each certificate to contain the CA signature.

Implicit Certificates

Implicit certificates are a different way of creating and trusting a certificate. With implicit certificates, the certificate requester and the CA cooperate to derive a final public key from the seed public key that the requester submits with the request. Instead of including a signature in the certificate, the CA includes a reconstruction value. A message recipient can combine the reconstruction value with the CA's public key and the rest of the contents of the certificate to recover the certificate holder's public key. This public key is only correct if the reconstruction value was created by the CA. Therefore, the CA's approval of the holder's public key is implicit, which means the public key only works if the CA was involved in creating it. This is different to an explicit approval as in standard certificates, where the public key's validity is explicitly confirmed by the CA signature.

The information flow for implicit certificates is:

- Certificate Creation
 - The certificate requester creates a seed public key
 - The CA calculates a mathematical transformation using the CA private key, the contents of the certificate, and the seed public key, to create:
 - A new public key for the certificate requester, the certified public key
 - A transformation that the certificate requester can use on the seed private key
 - A reconstruction value

- The CA sends the certificate contents, the reconstruction value and the private key transformation back to the certificate requester
- The certificate requester applies the private key transformation to the seed private key to obtain the certified private key
- The certificate requester checks that the certified private key corresponds to the certified public key
- Certificate Use
 - The certificate holder (who was the certificate requester in the previous step) signs a message with the certified private key and attaches the certificate (contents + reconstruction value)
 - The receiver uses the certificate contents, reconstruction value and CA public key to recover the certified public key
 - The receiver verifies the signature on the message with the certified public key

Implicit certificates have the following advantages over standard (explicit) certificates:

- An explicit certificate contains a public key (which is an elliptic curve point) and a signature, while an implicit certificate contains only a reconstruction value (which is an elliptic curve point). An implicit certificate is therefore smaller by the size of the signature, which in this case is 64 bytes. (The private key transformation adds 32 bytes to the certificate response compared to a response for an explicit certificate, but this is less than the signature size and is only included in the certificate response, not in signed messages). It is important to note that more details on this topic can be found in Standards for Efficient Cryptography Group, "SEC 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV)", Working Draft Version 0.97, March 2011, available from <http://www.secg.org>.
- The public key recovery operation and the signature verification can be combined into a single operation that takes approximately the same amount of time as required for a single verification. This is an advantage over explicit certificates, which require two verifications when assuming that the chain of trust ends at the authority issuing the certificate. However, this advantage applies only if a receiver verifies very occasionally. If the receiver verifies multiple messages signed by the same certificate, it is more efficient overall to recover the public key once and cache it. In this case, implicit and explicit certificate verification takes about the same time. A significant population of devices that verify only occasional messages and verify in software is anticipated and for these devices the performance advantages of implicit certificates are very important.

Implicit certificates are covered by patents owned by Certicom Corp. of Mississauga, Ontario, which is currently a wholly-owned subsidiary of BlackBerry Ltd. At the time of this document, there has been an agreement reached between Certicom and the Institute of Electrical and Electronics Engineers (IEEE) concerning the use of the associated patents. OEM lawyers should review [this agreement](#) carefully to determine whether it is acceptable and understand what alternatives might exist.

Detailed Comparison of Explicit and Implicit Certificate Calculations

There are two cases to consider: verifying the certificate chain and message signature and the case where only the message signature is being verified.

Explicit Certificates

Let us first focus on the case of verifying the certificate chain and message signature. In this case, one needs to verify the message signature and the signature on each of the certificates. Verifying requires to perform a "double multiply and add," i.e., calculating $aX + bY$, where X and Y are elliptic curve points and a and b are integers. Let us denote the cost for one double multiply and add by V . The cost for full certificate chain verification is $V * n$, where n is the length of the chain.

Once the full chain is verified, the following information is cached:

[Cert ID, public key, "successfully verified"]

This means that any time a message signed by that certificate is received, only one verification step needs to be performed: a lookup of the certificate establishes that it already has been verified. The cached public key is used to verify the message. The computational cost of this reads V .

Summarizing, the total cost for verifying a certificate chain using explicit certificates reads $V * n$ for the first verify and V for the subsequent ones.

Implicit Certificates

Verifying a message signed with an implicit certificate can be done in two steps: extracting the public key from the certificate and verifying the message. To extract the public key from a certificate, the public key from the issuer's certificate is required. The public key extraction operation is also a double-multiply-and-add. Thus, verifying an implicit certificate chain can be done using $V * (n + 1)$ operations: V for extracting the public key, and $V * n$ for verifying the certificate chain. At the end of the operation,

[Cert ID, public key, "successfully verified"]

is cached. Subsequent messages signed by that certificate can be verified at a cost of V .

Summarizing, the total cost for verifying reads $V * (n + 1)$ for the first verify, and V for the subsequent ones. This is slightly higher than for the explicit certificates case, but it should be observed that the same hardware as for the explicit case can be used. Recall that implicit certificates have an advantage in terms of size (64 byte in the considered case).

Finally, there is a way to improve the computational performance. Consider the case of a signed message with a certificate chain of length 2, i.e.,

[message, end-entity (implicit) certificate, known trusted (explicit) CA cert].

One can combine public key extraction and verification into a single operation, a "triple" multiply and add operation with cost approximately $1.16 * V$. So the first verification comes at a cost of approximately $1.16 * V$ instead of $2 * V$. However, combining operations in this way does not output the public key, so all subsequent operations (e.g., verifying subsequent messages signed with the same certificate) also come at a cost of $1.16 * V$.

Hardware Support

There are two types of double-multiply-and-add that may be supported by hardware:

- Generic double-multiply-and-add, $aX + bY$
- Double-multiply-and-add where one point is the base, $aX + bG$. This second type is easier to accelerate because G is known, so various values can be pre-computed.

Verifying a signature only requires the second type of operation. Implicit certificate key extraction needs the first type. More precisely, it needs a subset of the first type, $aX + Y$. As a consequence, an accelerator for signature verification can only be applied partially for key extraction: it would be used to calculate aX , and Y would have to be added in software.

Adding Y in software would slow things down, but only marginally as a single point add takes less than 1/50 the time for a full multiply. This would add less than one msec to total latency on a 400 MHz processor. However, it is a slowdown compared to explicit certificates.

In conclusion, hardware that supports signature verification may support implicit certificate key extraction with no performance cost (if generic double multiply-and-add is supported), or it may require additional software processing to support implicit certificate key extraction. The software processing is non-zero

time, but given that key extraction happens only when a certificate is first seen, if software processing is needed, its impact is very low.

Conclusions

In the following, certificate chains of reasonable length are assumed. Assuming one verifies signatures only occasionally (verify-on-demand), implicit certificates allow for an improvement in terms of size and computational effort, as there is no need to extract the public key from the implicit certificate. If every message is verified, it makes sense to extract the public key from the implicit certificate. In this case, implicit certificates allow only for improvements in terms of size which comes at the cost of one additional double-multiply-and-add operation at the first verify. As extraction of the public key needs to be performed on the first verify only, the first type of double-multiply-and-add does not necessarily have to be implemented in hardware.

Special Cryptography primitives in SCMS

The CAMP SCMS uses some cryptographic techniques that are not in widespread use in other PKIs. This section provides relevant cryptographic background. In the subsection [Crypto Primitives affecting End-Entity](#), we point out the primitives that also affect EEs.

Notation

To understand the special cryptographic constructions in this section, it is necessary to understand some of the underlying mathematics first. In the Elliptic Curve Cryptography system, the objects of interest are “elliptic curve points” which have the form (x, y) where (x, y) are all the points that are solutions of a particular cubic equation. A point P can be scalar-multiplied by an integer, a (a -times repeated addition of P by itself), to get a new point $Q = aP$. (Upper-case letters are used to indicate points, lower-case to indicate integers). In this coherence, multiplication of a point by an integer is defined so that it follows typical mathematical rules and always generates another point on the curve. The Elliptic Curve Discrete Logarithm Problem is basically the statement:

Given P and $Q = aP$, but not a , it is very difficult to work out the value of a .

In the following, for a , an element of $\{0, 1\}$ and an integer b , a^b denotes a b -bit string of a 's (e.g., 0^{64} is a 64-bit string of 0's); for bit strings c and d , $c \text{ XOR } d$ denotes their exclusive-OR; for bit strings x and y , $x || y$ denotes their concatenation; and for a bit string Z and an integer n , $[Z]_n$ denotes n most significant bits of Z . For a key k and message m , $\text{AES}(k, m)$ means AES encryption of m with k in ECB mode. In addition, unless otherwise noted, l is the order of the elliptic curve, la_id1 and la_id2 are 16-bit identifiers of linkage authorities LA1 and LA2, respectively, i , j , and k are 32-bit strings, and for brevity (i, j) are sometimes denoted by ι (Greek letter *iota*).

Time Periods

1. Cryptographic primitives explained in the sub-pages including [SCP1: Butterfly Keys](#) and [SCP2: Linkage Values](#) generate a sequence of cryptographic values. In other words, both techniques use functions that map from a known sequence (such as 1, 2, 3, ...) to a sequence whose entries are *a priori* unknown and unpredictable. The cryptographic details of the functions do not depend on the exact form of the input sequence, so one natural way they could be defined would be for the input sequence to be a single counter $i = \{0, 1, 2, 3, \dots\}$. In practice, in this document, two different approaches to define the techniques are employed. When defining the techniques for purposes of explaining the core concept, the techniques are written as if they take an input defined by a single counter i . This is the Greek letter *iota*.
2. For purposes of implementation, the input will be defined by two values, i and j . These are related to the pseudonym certificate provisioning model described in [Use Case 3: OBE Pseudonym Certificates Provisioning](#). This use case utilizes a coarse time period with a counter i and a more granular counter j , which is reset to 0 at the start of each i -period.

Note that i and j uniquely define i , an exemplary bijective. The term bijective is a mathematical term describing the characteristics of a function. A bijective function is both injective and surjective and implies a unique one-to-one relationship between the inputs and outputs of the function. When i and j are used for the input sequence, it is assumed that all devices and all SCMS components use the same value of i to denote the same time slot. In other words, i is a globally assigned variable, not a variable that individual OBEs or RAs have the ability to choose at will.

Pseudonym Certificate Validity

The length of the i -period should be the number of minutes in a week, 10080. We need to express it in minutes (as opposed to seconds) because the encoding in 1609.2 lets us use quantities of up to 2^{16} units and there are more than 2^{16} seconds in a week. The lifetime of the certificate is the i -period plus an overlap period. In the old design, the overlap period is one minute, but there are safety concerns with such a small overlap period, so we are extending the overlap period to one hour. This will enable vehicles to postpone certificate change if they are in an alert state that lasts more than a minute. With this extended overlap period, the lifetime of a pseudonym certificate is **10140 minutes**.

The start validity time of a pseudonym certificate is given in seconds since the 1609.2 epoch of 00:00:00 UTC, January 1, 2004.

If leap seconds happen, we may choose to adjust the start validity time of the certificates so it is not always 60×10080 seconds after the start of the previous batch but instead always lines up with the top of the hour. This concern is out of scope for POC and will be addressed later.

Clock Time Corresponding to global $i=0$

For the Safety Pilot, the clock time corresponding to $i=0$ was defined to be 00:00 UTC January 1, 2010. However, a lot has changed since, and in particular, the meanings of i and j have changed significantly in the old design. An important consideration for selecting the new clock time corresponding to $i=0$ is that changing i should cause minimum disruption to safety. According to http://www.forbes.com/2009/01/21/car-accident-times-forbeslife-cx_he_0121driving.html, the fewest deaths by crash happened between 4 and 5 am on Tuesday. With the highest population density on the East Coast, 4:00 am Eastern Standard Time makes most sense as during Daylight Saving Time, it will move to 5:00 am, which is still consistent with the above article. Considering all these, $i=0$ corresponds to: **4:00 am Eastern Standard Time on Tuesday, January 6, 2015** (i.e., in TAI 4023 days, 9 hours plus 3 leap seconds or **347,619,603 TAI seconds** since 1609.2 epoch).

Encryption of Pre-linkage Values by LA for PCA

In the old design, when LA sends pre-linkage values to RA for pseudonym certificate provisioning, it encrypts them for PCA using symmetric encryption. The secret key used for encryption is shared between LA and PCA through an out-of-band means. This has an impact on privacy (though only minor) from a malicious PCA, if RA uses more than one pair of LAs for a given PCA, as PCA can easily tell which pair of LAs were used in any given request sent by RA to PCA. This privacy impact can easily be mitigated if LA were to use public-key encryption for encrypting pre-linkage values for PCA, as a ciphertext generated using public-key encryption, does not need to contain any sender-related (in this case, LA) information. However, the team realized that using public-key encryption will add a significant amount of computational overhead on both LA and PCA, and decided to stick with symmetric encryption, **with a recommendation for RA to keep the number of pairs of LAs per PCA as low as possible, ideally one**.

Misbehavior Investigation: PCA Returns Encrypted Pre-linkage Values to MA

In the old design, during misbehavior investigation, PCA returns pre-linkage values to MA, which MA would then forward to LA. However, there is no need for MA to learn the pre-linkage values; PCA only needs to be able to point to a pre-linkage value that LA can then find information about. The design

change is as follows: PCA returns an encrypted pre-linkage value to MA. The encrypted pre-linkage value matches the encrypted pre-linkage value that LA originally provided to PCA as part of the pseudonym certificate provisioning process.

The new design is described in [Step 8.2: Misbehavior Investigation](#).

Crypto Primitives Affecting End-Entity

All of the changes mentioned below affect end-entities, and therefore they need to be informed to the V2V-SE team.

- [SCP1: Butterfly Keys](#)
- [SCP2: Linkage Values](#)
- [Clock Time corresponding to \$i=0\$](#)
- [Pseudonym Certificate Validity](#)

SCP1: Butterfly Keys

Summary

Butterfly Keys are a novel cryptographic construction that allow a device to request an arbitrary number of certificates, each with different signing keys and each encrypted with a different encryption key, using a request that contains only one verification public key "seed" and one encryption public key "seed" and two "expansion functions." The expansion functions allow a second party to calculate an arbitrarily long sequence of statistically uncorrelated (as far as an outside observer is concerned) public keys such that only the original device knows the corresponding private keys. Without butterfly keys, the device would have to send a unique verification key and a unique encryption key for each certificate. Butterfly keys reduce upload size, allowing requests to be made when there is only spotty connectivity, reduce the work to be done by the requester to calculate the keys, and smoothen peak requests.

A core principle of PKI implementations is that all private keys should be generated on the device that is going to use them. If private keys are generated off the device (and then installed on it), and if the device appears to misbehave, the device owner can claim that the misbehavior was actually carried out by whoever generated the keys. However, in the original CAMP design, a single device had over 100,000 certificates per year. Generating 100,000 distinct certificate requests would be a significant computational burden, and arguably an unnecessary one given that most vehicles are only in operation for about 5% of the time. Additionally, 100,000 distinct certificate requests would take a long time to upload and, if connections from the onboard equipment (device) to the CA are unreliable, there is a risk that certificate requests would not complete successfully within a single communication session.

The CAMP design thus updated its approach to use butterfly keys to address both these concerns. In the butterfly key approach, the certificate requester only needs to generate a single key pair and include the public key in a single certificate request. The difference from the standard approach is that the requester also creates an expansion function that allows a single public key to be expanded into multiple public keys and a single private key to be expanded into multiple private keys, while ensuring that only someone who knows the original private key will know the expanded private keys (i.e., the device). This reduces the computational burden on the device (it only has to generate one key) and also the size of the upload (reduced to less than 1K bytes). The cost is that the download of certificate responses increases in size.

Preliminaries

i and j Values

1. For **pseudonym** certificates, the i value used in any certificate is the *global* i value.

- a. The clock time corresponding to the global $i=0$ shall be as per [Special Cryptographic Primitives in SCMS](#).
 - b. The increment period for the global i value shall be fixed at 1 week, i.e., $7*24$ hours, where *hours* is the field defined under the type Duration in IEEE 1609.2, see <https://github.com/wwhyte-si/1609dot2-asn/blob/master/1609dot2-base-types.asn>.
 - c. The j value shall range from 0 to $j_{\max}-1$. For POC and CV Pilots, j_{\max} for all devices is fixed at 20.
2. For **identification** certificates, the i value used in any certificate is the *local* i value corresponding to the *enrollment* certificate used for requesting that identification certificate.
- a. The clock time corresponding to that local $i=0$ shall be the value of *toBeSigned.validityPeriod.start* field of the *enrollment* certificate.
 - b. The increment period for that local i value shall be the value of *toBeSigned.validityPeriod.duration* field (minus the overlap period, see [PoC Certificate Expiration Timelines](#), [CV Pilot PROD Certificate Expiration Timelines](#)) of the *identification* certificate.
 - c. For POC and CV Pilots, the j value for all devices is fixed at 0.

Description

Butterfly keys make use of ECDLP as follows. There is an agreed “base point” called G (this is standard practice for elliptic curve cryptography). The device generates two ECC key-pairs ($a, A = aG$) (seed for the signing keys) and ($p, P = pG$) (seed for keys used for encrypting the certificates, i.e., encryption keys), and descriptions of two expansion functions f_1 (for signing keys) and f_2 (for encryption keys). The expansion functions map an integer i to another integer in a range from 0 to l , the order of the elliptic curve. Functions f_1 and f_2 are designed to be cryptographically secure, which roughly means that the output looks random so that given two values of $\{f_1(i), i\}$ (or, $\{f_2(i), i\}$), a third party cannot tell whether the two values were generated by the same version of f_1 (or, f_2), or by different versions. The vehicle stores a, p , and descriptions of f_1 and f_2 , and sends A, P , and description of f_1 and f_2 encrypted to the SCMS. In the CAMP design, the expansion functions are defined as:

1. $f_1(k, i) = f_1^{\text{int}}(k, i) \bmod l$, where
 - a. $f_1^{\text{int}}(k, i)$ is the big-endian integer representation of $(\text{AES}(k, x+1) \text{ XOR } (x+1)) \parallel (\text{AES}(k, x+2) \text{ XOR } (x+2)) \parallel (\text{AES}(k, x+3) \text{ XOR } (x+3))$,
 - b. $x+1, x+2$, and $x+3$ are obtained by simply incrementing x by 1 each time, e.g., if $x = 01 \dots 00$, then $x+1 = 01 \dots 01$, $x+2 = 01 \dots 10$, $x+3 = 01 \dots 11$,
 - c. 128-bit input x for AES is derived from time period $i = (i, j)$ as: $(0^{32} \parallel i \parallel j \parallel 0^{32})$.
2. $f_2(k, i)$ is defined in an identical way as $f_1(k, i)$, **except** x is derived as: $(1^{32} \parallel i \parallel j \parallel 0^{32})$.

The “description” of f_1 and f_2 are simply the AES keys ck (for signing keys) and ek (for encryption keys): to generate f_1 and f_2 the device simply generates 2 AES keys ck and ek , and to send the description of f_1 and f_2 the device sends ck and ek .

Now the SCMS has the ability to generate an extremely large number of derived points: it can generate

- $B_i = A + f_1(ck, i) * G$, with $A = aG$ (signing keys)
- $Q_i = P + f_2(ek, i) * G$, with $P = pG$ (encryption keys)

The corresponding private keys will be

- $b_i = a + f_1(ck, i)$ (signing keys)
- $q_i = p + f_2(ek, i)$ (encryption keys)

Since the SCMS doesn't know the original value of a (or, p), it cannot know any of the b_i (or, q_i) values, so it can generate an arbitrary number of public keys for which only the vehicle knows the corresponding private keys. Additionally, because the expansion functions are cryptographically secure, anyone who doesn't know the description of f_1 cannot tell whether two different signing public keys belong to the same series $\{B_i\}$ or to different series. This means that the RA, described in detail later, can safely use f_1 to create an expanded list of signing public keys to send to the CA, and the CA will not be able to tell that the keys belong to the same vehicle.

In the CAMP SCMS, this underlying approach is used as follows. Note: There are a couple of minor technical differences between this description and the CAMP approach – explained after this description, which focuses on the core butterfly key operations and omits optimizations that might obscure the explanation:

- Device generates two 128-bit AES keys ck and ek , for expansion functions of signing keys and encryption keys, respectively, and two “caterpillar” key pairs:
 - $(a, A = aG)$ used for signing, i.e., A to be placed in the certificate
 - $(p, P = pG)$ used for encryption of the certificate

Device sends $\{ck, ek, A, P\}$ to the RA. Note: ck will define the expansion function for the signing keys and ek will define the expansion function for the encryption keys.

- RA uses ck to generate $\{B_i\}$, the series of “cocoon” signing public keys for the certificate requests, and ek to generate $\{Q_i\}$, the series of cocoon encryption public keys for encrypting the certificate response, pairs each B_i with the corresponding Q_i , and sends the pairs $\{B_i, Q_i\}$ to the CA.
- CA does not know which public keys have come from the same device, but RA knows which public keys are in the requests, so CA must further randomize the public keys to hide them from RA. For each request, CA generates a unique random integer c and sets the public key in each certificate to the “butterfly” value $(B_i + cG)$. CA then uses Q_i to encrypt the response, which contains:
 - Certificate containing the public key $(B_i + cG)$
 - CA's contribution to the private key, c
- RA sends the encrypted message to the device along with the corresponding i .
- Device uses ek, p, i to calculate q_i . It uses q_i to decrypt the response and recover the certificate containing the public key $(B_i + cG)$ and CA's contribution to the private key, c . It then uses ck, a, i to calculate b_i . The private key for the certificate is then:
 - Butterfly private key = b_i (calculated above) + c (provided by CA)

Device should at this point check that the recovered private key corresponds to the public key certified by the certificate to ensure that it has been sent the correct certificate.

This means that the device has obtained a set of certificates such that:

- Only the device knows the private keys
- RA does not know the public keys in the certificates
- CA cannot tell from the requests alone which requests have come from the same device

Notes:

1. In the CAMP design, there are a couple of differences. First, implicit certificates are used by a device, so the CA's contribution to the private key is calculated slightly differently; however, the principle is the same, namely that the CA modifies the public key and sends information to the vehicle that allows it to make the corresponding modification to the private key. Moreover, in the [table below](#), butterfly keys process has been summarized for both explicit and implicit certificates. Second, the CA

additionally signs (using its private key) the encrypted implicit certificate to prevent a man-in-the-middle (MITM) attack by the RA. To launch the MITM attack, the RA can simply use a different public key of its choice (for which it knows the corresponding private key) in the request to the CA, so that it can decrypt the encrypted response by the CA, view the underlying certificate, and then while responding to the vehicle encrypt the certificate with the right public key.

2. Since the RA knows the public encryption key J_i , it could in principle create a fake response to the vehicle. This would allow the RA to give a set of known certificates to a target vehicle, allowing the RA to track. However, any fake response will not have been created with the CA private key and so the vehicle can detect this attack and discard the resulting keys.
3. The per-certificate value c generated by the CA is vital in hiding the final certified public key from the RA. If c were a constant, all the certificates would be related to their requests in some known way, and the RA could work out the set of certificates corresponding to a set of certificate requests and track the vehicle. Likewise, if the CA generates c with bad randomness, or with randomness that is known to the RA, then the RA may be able to work out which certificate belongs to which vehicle. (see Random Number Generators under [CB2: Types of Cryptographic Algorithms](#) and [Approved Random Number Generators](#) for further details on “good randomness.”).
4. Test vectors for butterfly expansion function are available at <http://stash.campplc.org/projects/SCMS/repos/crypto-test-vectors/browse/bfkeyexp.txt>

	Device	RA	PCA
Explicit Certs	1) Generate: <ol style="list-style-type: none"> a) AES key ck for expansion function of signing keys b) AES key ek for expansion function of encryption keys c) ECC key pair $(a, A = aG)$ for signing, i.e., “caterpillar” signing key pair d) ECC key pair $(p, P = pG)$ for encryption of the certificate, i.e., “caterpillar” encryption key pair 2) Send (ck, ek, A, P) to RA	3) For each i , compute: <ol style="list-style-type: none"> a) “Cocoon” signing public keys for certificate requests, $B_i = A + f1(ck, i) * G$ b) “Cocoon” encryption public keys for encrypting the certificate response, $Q_i = P + f2(ek, i) * G$ 4) For each i , send (B_i, Q_i) individually to PCA	5) Generate an ECC key pair $(c, C = cG)$ for hiding the signing public key from RA, and compute the “butterfly” public key $(B_i + C)$ 6) Generate an explicit certificate on butterfly public key $(B_i + C)$, encrypt (certificate, c) with Q_i , sign the ciphertext again, and send the signed ciphertext to RA
		7) Collate all the signed ciphertexts along with the corresponding i value for a device and send them to device	
	8) For each i , compute: <ol style="list-style-type: none"> a) Cocoon signing private keys, $b_i = a + f1(ck, i) \pmod{l}$ b) Cocoon decryption keys for decrypting the certificate response, $q_i = p + f2(ek, i) \pmod{l}$ 9) For each i , verify PCA’s signature on the ciphertext: <ol style="list-style-type: none"> a) If verification succeeds, decrypt the ciphertext using q_i to obtain (certificate, c). Compute the “butterfly” private key corresponding to the public key in the certificate: $(b_i + c) \pmod{l}$ b) If verification fails, abort and report to MA. 		
Implicit Certs	1) Generate: <ol style="list-style-type: none"> a) AES key ck for expansion function of signing keys 	3) For each i , compute: <ol style="list-style-type: none"> a) “Cocoon” signing public keys for certificate 	5) Generate an implicit certificate on B_i , let the private and public reconstruction values be r and R , respectively

Device	RA	PCA
b) AES key ek for expansion function of encryption keys c) ECC key pair $(a, A = aG)$ for signing , i.e., "caterpillar" signing key pair d) ECC key pair $(p, P = pG)$ for encryption of the certificate , i.e., "caterpillar" encryption key pair 2) Send (ck, ek, A, P) to RA	requests, $B_i = A + f_1(ck, i) * G$ b) "Cocoon" encryption public keys for encrypting the certificate response, $Q_i = P + f_2(ek, i) * G$ 4) For each i , send (B_i, Q_i) individually to PCA	6) Encrypt (r, R) with Q_i , sign the ciphertext again, and send the signed ciphertext to RA
	7) Collate all the signed ciphertexts along with the corresponding i value for a device and send them to device	
8) For each i , compute: a) Cocoon signing private keys, $b_i = a + f_1(ck, i) \pmod I$ b) Cocoon decryption keys for decrypting the certificate response, $q_i = p + f_2(ek, i) \pmod I$ 9) For each i , verify PCA's signature on the ciphertext: a) If verification succeeds, decrypt the ciphertext using q_i to obtain (r, R) . Reconstruct the "butterfly" private key corresponding to the certificate using b_i and r . b) If verification fails, abort and report to MA.		

Table 64 Butterfly Key

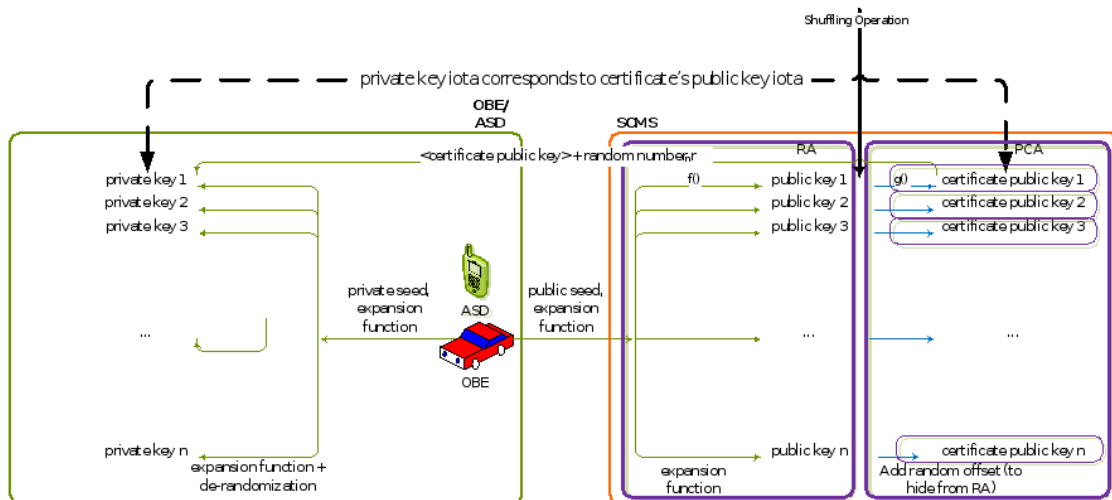


Figure 64 Butterfly Key Mechanism

SCP2: Linkage Values

Summary

To support efficient revocation, end-entity certificates contain a linkage value (LV), which is derived from (cryptographic) linkage seed material. Publication of the seed is sufficient to revoke all certificates belonging to the revoked device, but without the seed an eavesdropper cannot tell which certificates belong to a particular device. Note: The revocation process is designed such that it does not give up backward privacy. For protection against insider attacks by the SCMS, the LV is the combination of two pre-linkage values (PLVs) produced by two independent LAs; this ensures that no single SCMS entity knows all the information belonging to a single device. An extension to the linkage values approach allows for group revocation, so that if all devices of a particular type have a flaw they can be revoked with a single entry on the revocation list, while keeping group membership secret until the relevant group seed is revealed. Group revocation is currently not implemented as no practical real-world use case been identified so far.

LVs and LAs are used to enable the SCMS to help achieve the following preliminary design requirements, which were developed by the research team to ensure appropriate privacy protections and efficiency:

1. There is an efficient way of revoking all the certificates within a device
2. Certificates are not linkable by an eavesdropper unless the owner has been revoked
3. A vehicle is trackable only after its credentials are revoked but not before it was revoked.
4. No single entity within the system is able to determine that two certificates belong to the same device. An exception to this rule is the Misbehavior Authority (MA).
5. No single entity within the SCMS is able to track a vehicle. Once a single LA is introduced, this requirement is not fulfilled any longer. For that reason, two LAs are used and the information which allows for tracking is split between them.

Description

The basic concept of LVs uses the well-known cryptographic construction known as a hash chain. As described above, a hash algorithm is like a cryptographic checksum; if the hash of 'a' is computed as $H(a) = b$, it is very hard for someone who sees only b to derive the input a , but given a and b it is trivial to determine that a hashes to b . Hence, it is desirable to have a series of identifiers in each certificate such that if a secret is revealed, the identifiers can be linked.

First a description of the revocation of individual nodes is provided. For simplicity, a system with a single LA that generates LVs is initially described. This system meets requirements 1), 2), and 3) discussed, above. It does not meet requirement 4), because the LA can link certificates. The following describes the basic process for a single series of certificates. A more detailed description will be provided below. For a complete description of the process see Section 4.2.2.

- LA starts with an initial linkage seed (ILS), $ls(0)$. (This will be different for each vehicle.)
- For each time period $i > 0$, LA sets the LS $ls(i) = H(ls(i-1))$, for some hash function H (SHA-256, a National Institute for Standards and Technology (NIST)-approved standardized hash algorithm that is used throughout IEEE 1609.2 is employed)
- The certificate for each time period i contains the linkage value $lv(i) = AES(ls(i), 0)$
- To revoke a vehicle from time period i onwards, the revocation authority publishes $ls(i)$
- To check revocation at time period $i' > i$, the recipient of a signed message:
 - Hashes $ls(i)$, and then hashes the output of the hash, repeated $(i'-i)$ times to obtain $ls(i')$
 - Calculates $lv(i') = AES(ls(i'), 0)$

- Checks whether the certificate that signed the message contains the LV $lv(i')$. If it does, the certificate is considered revoked and the message is rejected

This achieves requirements 1), 2), and 3) as follows:

- Efficient revocation: Only one value needs be published to revoke all the certificates on a vehicle. The cost of maintaining the revocation data on the receiver side is one hash per revoked vehicle per time period. Hashing is very efficient, so this maintenance is inexpensive in terms of processing.
- Unlinkability against eavesdroppers: To tell if two certificates belong to the same vehicle, an eavesdropper would have to determine the two LSs ls_1, ls_2 that encrypt 0 to the PLVs plv_1, plv_2 in the certificates. Since AES is assumed to be a secure block cipher, this is not possible.
- Retrospective unlinkability: The hash chain can be run forward from the revocation value $ls(i)$, but not backwards to recover previous values of $ls(i)$. (This is a result of the non-invertibility of hash functions.)

However, the system has the problem that the LVs are generated centrally and the entity that generates the LVs knows the complete set of values that belong to a vehicle. To overcome this problem, the CAMP SCMS uses two LAs: LA1 and LA2.

In the description above, there is a single chain of LSs and LVs. In the CAMP SCMS, each of the LAs generates a chain of PLVs. These PLVs are individually encrypted and passed to the PCA; the PCA then XORs them together to obtain the LV that is put in the certificate. Now neither of the LAs knows the XORed linkage values that appear in the final certificate, because neither knows the values produced by the other LA. To revoke, the MA publishes the LSs from both LAs, and the recipient reconstructs both chains of PLVs and carries out the XORing to obtain the LVs for revoked certificates. The following describes the generation process in more detail:

- LA1 starts with a random ILS, $ls_1(0)$
- LA2 starts with a random ILS, $ls_2(0)$
- For each time period $i > 0$:
 - LA1 sets its LS $ls_1(i) = H(ls_1(i-1))$, and LA2 sets its LS $ls_2(i) = H(ls_2(i-1))$
 - LA1 sets its PLV, defined as $plv_1(i) = AES(ls_1(i), 0)$ and LA2 sets its $plv_2(i) = AES(ls_2(i), 0)$
 - The CA sets the LV $lv(i) = plv_1(i) XOR plv_2(i)$ and puts it in the certificate for time period i
- To revoke a vehicle from time period i onward, the revocation authority publishes the linkage seeds $ls_1(i)$ and $ls_2(i)$
- To check revocation at time period $i' > i$, the recipient of a signed message:
 - Iteratively hashes $ls_1(i)$ ($i'-i$) times to obtain $ls_1(i')$; does the same for $ls_2(i)$
 - Calculates PLVs $plv_1(i') = AES(ls_1(i'), 0)$ and $plv_2(i') = AES(ls_2(i'), 0)$
 - Checks whether the certificate that signed the message contains the LV $lv(i') = plv_1(i') XOR plv_2(i')$. If it does, the certificate is considered revoked and the message is rejected.

Four additional refinements in the CAMP SCMS are:

1. Instead of using a single time period counter i , time periods are denoted (i, j) , where i counts up larger time periods (e.g., a day, a week, etc.) and j can be used in one of (at least) two ways: (a) for non-overlapping certificates, it can count up smaller time intervals within the larger time periods (e.g., 5-minute intervals); (b) for overlapping certificates, it can specify the number of certificates that are valid in a given time period i (e.g., fixing the range of j as 1-20 would imply that 20 certificates are valid simultaneously). The LSs $ls_1(i)$ and $ls_2(i)$ are calculated as described above, but the PLVs $plv_1(i, j)$ and $plv_2(i, j)$ are calculated as $AES(ls_1(i), j)$ and $AES(ls_2(i), j)$, respectively. The reason for this is to save time for vehicles that have been offline for some time. If a vehicle has been turned off for 1 year,

without this refinement, at key-on the vehicle will have to carry out $52 * 20$ hashes for each revocation entry to bring its revocation information up to date (assuming that a vehicle is issued 20 simultaneously-valid certificates per week). With this refinement, the vehicle only has to perform one hash per week for each revocation entry. If revocation lists get large, this efficiency gain may be very useful.

2. To reduce the chance of collisions in the PLVs between two LAs, their identities are also employed during the computation of LSs and PLVs: la_id1 and la_id2 are unique 16-bit identity strings associated with LA1 and LA2, respectively. The LSs are calculated as: $ls1(i) = H(la_id1 \parallel ls1(i-1))$, $ls2(i) = H(la_id2 \parallel ls2(i-1))$. The PLVs are calculated as: $plv1(i,j) = AES(ls1(i), la_id1 \parallel j)$, $plv2(i,j) = AES(ls2(i), la_id2 \parallel j)$. This means that even if two LAs produce the same LS for a given time period, they will produce different sets of PLVs (because of the use of the identifier to produce the PLV from the LS), and their LSs will be different in the next time period (because of the use of the identifier to create the next seed from the current seed).
3. To reduce the size of certificate revocation list (CRL), which contains the LSs of the revoked vehicles, the LSs are truncated to 16 bytes.
4. Instead of plain AES, AES is used in the Davies-Meyer mode as a derivation function, which is basically XORing the output of AES with the input. In particular, for a key k and message m , instead of $AES_k(m)$, $(AES_k(m) \text{ XOR } m)$ is returned for every invocation of AES.

The [table below](#) summarizes the linkage value generation and the [figure below](#) visualizes the described scheme. Test vectors for Linkage Values are available at <http://stash.campllc.org/projects/SCMS/repos/crypto-test-vectors/browse/lv.txt>

LA1	LA2	RA	PCA
1. Generate initial linkage seed, $ls1(0)$ (128-bit string chosen at random for every device). 2. Compute linkage seed for i^{th} period through an iterative process defined as: $ls1(i) = [SHA-256(la_id1 \parallel ls1(i-1) \parallel 0^{112})]_{128}$. 3. Compute pre-linkage value, $plv1(i, j) = [(AES(ls1(i), la_id1 \parallel j \parallel 0^{80})) \text{ XOR } (la_id1 \parallel j \parallel 0^{80})]_{72}$. 4. Encrypt $plv1(i, j)$ for PCA, and send it to RA.	1. Generate initial linkage seed, $ls2(0)$ (128-bit string chosen at random for every device). 2. Compute linkage seed for i^{th} period through an iterative process defined as: $ls2(i) = [SHA-256(la_id2 \parallel ls2(i-1) \parallel 0^{112})]_{128}$. 3. Compute pre-linkage value, $plv2(i, j) = [(AES(ls2(i), la_id2 \parallel j \parallel 0^{80})) \text{ XOR } (la_id2 \parallel j \parallel 0^{80})]_{72}$. 4. Encrypt $plv2(i, j)$ for PCA, and send it to RA.	5. Include encrypted $plv1(i, j)$ and $plv2(i, j)$ in the certificate request.	6. Decrypt the packet from RA to obtain $plv1(i, j)$ and $plv2(i, j)$. 7. Compute linkage value, $lv(i, j) = plv1(i, j) \text{ XOR } plv2(i, j)$

Table 65 Linkage Values

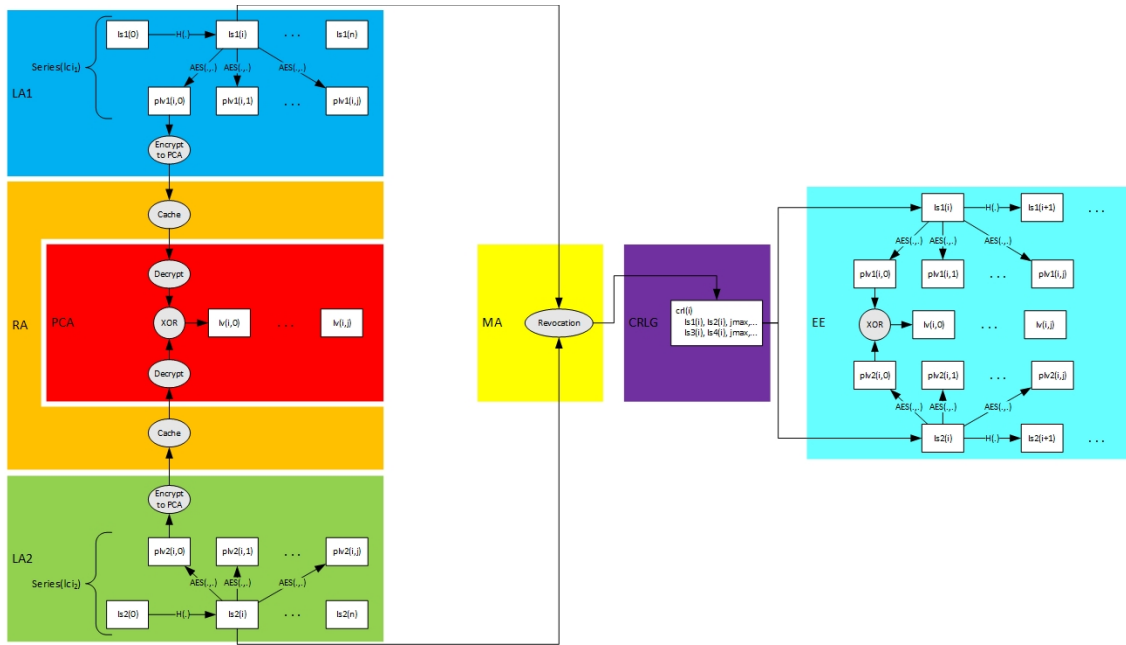


Figure 65 Creation of Individual Linkage Values and Revocation of Individual Device

Crypto Test Vectors

This directory contains test vectors for the following functions as specified [here](#).

Additionally there are test vectors for crypto functions needed for encryption and signing/verification.

All python scripts implement the corresponding functionality in order to depict the mathematical and cryptographic calculations involved.

Linkage Values $lv(i,j)$

- `lv.txt`: test vectors for $i = \{0,1\}$ and j randomly chosen in $[1,20]$
- `lv.py` : Python script that generates the test vectors

Group Linkage Values $glv(i,j,k)$ and Encrypted Indices $ei(j,k)$

- `glv.txt`: test vectors for $i = \{0,1\}$ and j randomly chosen 32-bit value
- `glv.py` : Python script that generates the test vectors

Butterfly Expansion Function

- `bfkeyexp.txt`: test vectors for Butterfly Expansion Function for Certificate and Encryption key pairs
- `bfkeyexp.py` : Python script that generates the test vectors

Key Derivation Function, KDF2 [IEEE-1363a, ANSI X9.63] with SHA-256

- `kdf.txt`: ANSI X9.63 test vectors of KDF2 with SHA-256
- `kdf.py` : Python script that implements KDF2 and tests it against the test vectors included

Message Authentication Code, MAC1 (HMAC)[IEEE-1363a, ANSI X(.71, RFC 2104, 4231] with SHA-256

- `mac1.txt`: RFC 4231 test vectors of HMAC-SHA-256
- `mac1.py` : Python script that implements HMAC-SHA-256 and tests it against the test vectors included

AES-CCM-128 Symmetric Authenticated Encryption [IEEE-1609.2, NIST SP 800-38C]

- `aesccm.txt`: test vectors for AES-CCM-128 Symmetric Authenticate Encryption based on NIST SP 800-38C (and RFC 3610) with parameters defined in IEEE-1609.2
- `aesccm.py` : Python script that generates the test vectors

ECDH Key Agreement [SP800-56A Section 5.7.1.2]

- `ecdh.txt`: test vectors for ECDH Key Agreement Scheme as per SP800-56A Section 5.7.1.2 using NIST test vectors
- `ecdh.py` : Python script that implements ECDH for curve P-256 and tests it against the test vectors included

ECIES Public-Key Encryption [IEEE-1609.2]

- `ecies.txt`: test vectors for ECIES Encryption as per IEEE-1609.2, Used to wrap AES-CCM 128-bit keys

- `ecies.py` : Python script that generates the test vectors

Implicit Certificate Generation and Public/Private Keys Reconstruction [SEC-4]

- `implicit.txt`: test vectors for generating implicit certificates and for reconstructing the corresponding private and public keys as per [SEC-4].
- `implicit.py` : Python script that generates the test vectors

Other files:

- `radix.py`:
- `array.py`: utility scripts for printing the output
- `ecc.py`: Elliptic Curve Cryptosystems core computations

Hash-based Functions

Key Derivation Function, KDF2

- This function is used to expand/derive keys from shared secret and specified input parameters. The derived keys may be used in symmetric-key encryption and/or authentication.
- Based on SHA-256
- Implemented as per IEEE-1363a and ANSI X9.63
- Required in the implementation of [ECIES](#)

Message Authentication Code, MAC1 (HMAC)

- This is a symmetric-key authentication function, i.e., it takes as input an authentication key and the data to be authenticated and outputs an authentication tag that is appended to the data and ensures its integrity and authenticity
- Based on SHA-256
- Implemented as per IEEE-1363a, ANSI X9.71, RFC 2104 and 4231

Required in the implementation of [ECIES](#)

AES-based Functions

AES-CCM Authenticated Encryption

- This is a symmetric-key authenticated encryption function, i.e., it takes as input a symmetric key and a plaintext and outputs a cipher text and an authentication tag. It provides confidentiality, integrity and authenticity of the data.
- Based on AES-128
- Implemented as per IEEE-1609.2 and NIST SP 800-38C
- Used in all data encryption. The symmetric key used is then wrapped with [ECIES](#) and sent along with the encrypted data.

ECC Functions

ECDH Key Agreement

- Elliptic Curve Diffie-Hellman is a public-key primitive where two parties can compute a shared secret by exchanging public keys and employing them and the corresponding private keys in the computation
- Based on ECC over the curve P-256

- Implemented as per NIST SP800-56A Section 5.7.1.2
- Required in the implementation of [ECIES](#)

ECIES Public-key Encryption

- Elliptic Curve Integrated Encryption Scheme is a hybrid encryption primitive composed of public-key key agreement (ECDH), a key derivation function (KDF2) and symmetric-key encryption (XOR) and authentication (MAC1). A Sender employs this scheme to encrypt a message using the public-key of the Recipient.
- The primitives in the brackets above are as per IEEE-1609.2
- Used to wrap (encrypt) AES-CCM-128 encryption keys

Implicit Certificate Generation and Public/Private Keys Reconstruction

- Implicit certificates are employed for pseudonym certificates, enrollment certificates, etc. (see [Certificate Types](#))
They do not contain the subject's public key and are not signed by the issuer, as is the case with explicit certificates, rather they contain a public key reconstruction point that is used to reconstruct the public key of the subject knowing the public key of the issuer. When issued, a private key reconstruction value is sent along from the issuer to the subject and only the subject can reconstruct the actual private key using this value and the private key used in the certificate request. In the case of pseudonym certificates, for example, the subject is the EE and the issuer is the PCA.
- Based on ECC over the curve P-256
- H(CertU) in the script is provided as an illustrative value. See IEEE 1609.2-2016, Sec. 6.4.8, under ENCODING CONSIDERATIONS: "for implicit certificates, the value H(CertU) in SEC 4, section 3, is for purposes of this standard taken to be H (H (canonicalized ToBeSignedCertificate from the subordinate certificate) || H(entirety of issuer Certificate)). See 5.3.2 for further discussion" See the cited section for details.
- **Notes:**
 - CertU is a term in SEC4 standard which only deals with implicit certs and public/private keys reconstruction. It is used in the Python scripts that generate the test vectors with comments that ensure that it's understood to be only for illustrative purpose:
 -
 - tbsdata was arbitrary
 -
 - CertU = Hash (tbsdata || pub key recon point), just to emphasize that however the input to the hash is formulated, it should contain at least the tbs data and the public key reconstruction point for mathematical and cryptographic correctness and assurance of the algorithm
 - In 1609.2-2016, Sec 6.4.5 and 6.4.8, the public key reconstruction point is the verifyKeyIndicator and is already part of ToBeSignedCertificate, which already satisfies the cryptographic requirement in the previous item. In addition to that, there is another input, the issuer cert; as explained in Sec 6.4.8, the hash is calculated as: H (H (canonicalized ToBeSignedCertificate from the subordinate certificate) || H(entirety of issuer Certificate))
 - The hash in 1609.2 is calculated the same way for implicit and explicit certs as in Sec 5.3.1 and Sec. 6.4.8
 - The operations in SEC4 that would require using the 1609.2-2016 hashing algorithm are conveniently listed in 1609.2-2016 Sec 5.3.2:

5.3.2 Implicit Certificates

Implicit certificates were proposed in Brown, Gallant, and Vanstone [B3] and Pintsov and Vanstone [B18] while modifications to protect against attacks were proposed in Brown, Campagna, and Vanstone [B4]. In this standard, implicit certificates are processed as specified in Standards for Efficient Cryptography (SEC) 4 except for the exceptions noted in this subclause.

a) In this standard, an implicit certificate is encoded as an *ImplicitCertificate*, as defined in 6.4.5, and is encoded with the Canonical Octet Encoding Rules (COER). All references to “the certificate *CertU*” in SEC 4 should be taken as referring to the encoded *ImplicitCertificate* except when the implicit certificate is hashed to an integer modulus n . This case is addressed in item b) below.

b) When an implicit certificate is hashed to an integer modulo n , the input is not simply the implicit certificate *CertU* but the information specified below. This affects the following steps in SEC 4:

- 1) Section 3.4, Action 7
- 2) Section 3.5, Action 4
- 3) Section 3.6, Action 2
- 4) Section 3.7, Action 4
- 5) Section 3.8, Action 4

The encoded data input to the hash function is $\text{Hash}(\text{ToBeSignedCertificate from the subordinate certificate as specified in 6.4.8}) \parallel \text{Hash}(\text{Entirety of issuer certificate as specified in 6.4.3})$.

Linkage Values and Butterfly Key Expansion Functions

- These are as specified in [Special Cryptographic Primitives in SCMS](#)

1602.2 and SCMS ASN.1 Objects

[scms-protocol.asn](#):

```
SecuredScmsPDU ::= IEEE1609Dot2Data
```

Figure 66 SCMS-Protocol ASN.1

[1609.2-schema.asn](#):

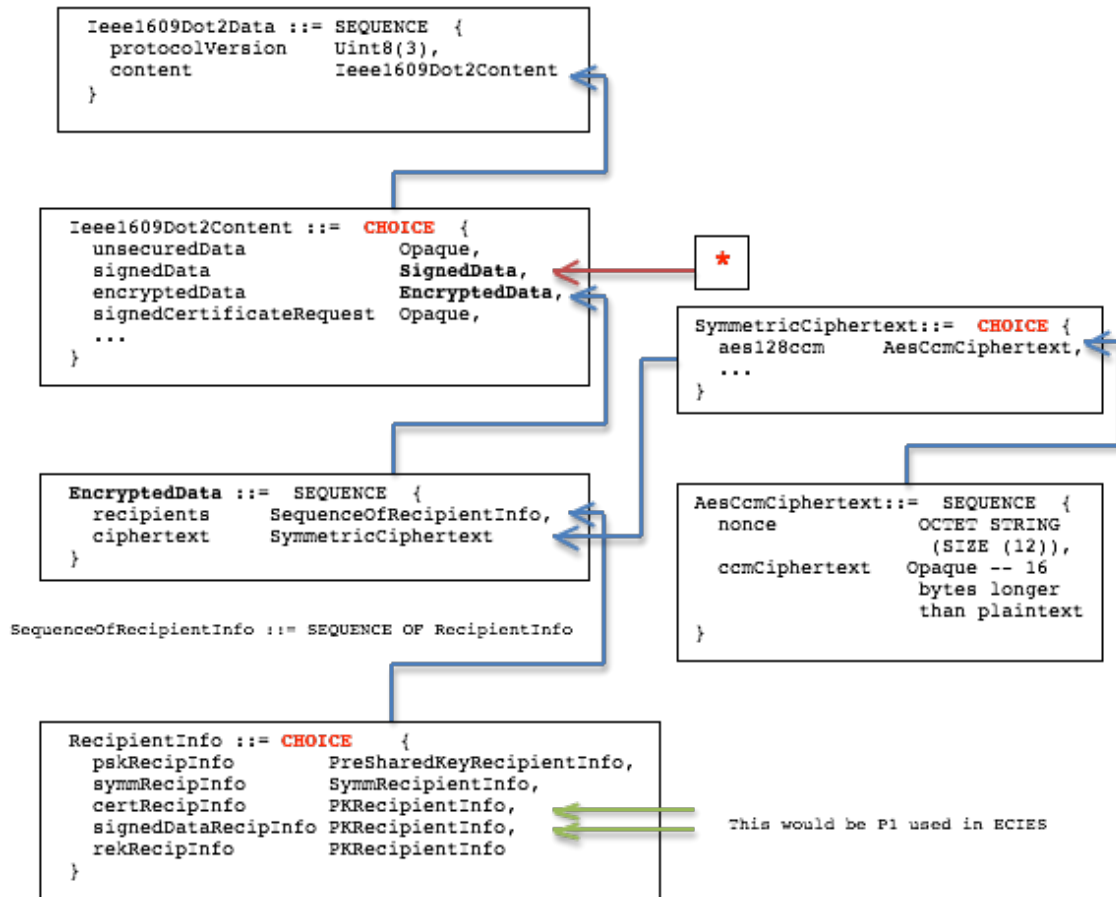


Figure 67 IEEE 1609.2 Schema ASN.1

See [ECIES diagram](#) and [Notes on Encrypted Data](#) below, as well as [PKRecipientInfo](#) and [EciesP256EncryptedKey ASN.1 objects](#) in [1609dot2-schema.asn](#) and [EciesP256EncryptedKey](#) in [1609dot2-base-types.asn](#), to see how the outputs of ECIES are encoded in the RecipientInfo.

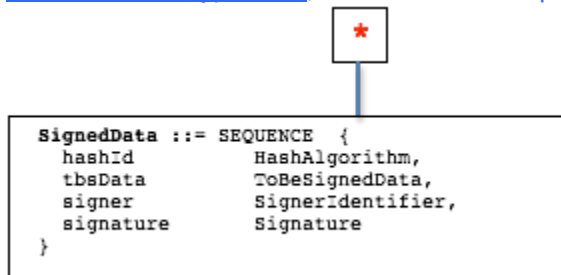


Figure 68 SignedData ASN.1

Example of SignedData from [scms-protocol.asn](#):

```

SignedPseudonymCertProvisioningAck ::= SecuredScmsPDU (WITH COMPONENTS {...,
content (WITH COMPONENTS {...,
  signedData (WITH COMPONENTS {...,
    tbsData (WITH COMPONENTS {...,
      payload (WITH COMPONENTS {...,
        data (WITH COMPONENTS {...,
          content (WITH COMPONENTS {
            unsecuredData (CONTAINING ScopedPseudonymCertProvisioningAck)
          })
        })
      })
    })
  })
  headerInfo (WITH COMPONENTS {...,
    psid (SecurityMgmtPsid),
    generationTime ABSENT,
    expiryTime ABSENT,
    generationLocation ABSENT,
    certLearningRequest ABSENT,
    missingCrlIdentifier ABSENT,
    encryptionKey ABSENT
  })
})
})
})
})

```

```

ScopedPseudonymCertProvisioningAck ::=
ScmsPDU (WITH COMPONENTS {...,
  content (WITH COMPONENTS {
    ee-ra (WITH COMPONENTS {
      raEePseudonymCertProvisioningAck
    })
  })
})

```

Defined in ee-ra.asn:
See EndEntityRaInterfacePDU and
RaEePseudonymCertProvisioningAckMsg

Figure 69 SignedData Example ASN.1

ECIES Encryption as in 1609.2-2016, Sec 5.3.5

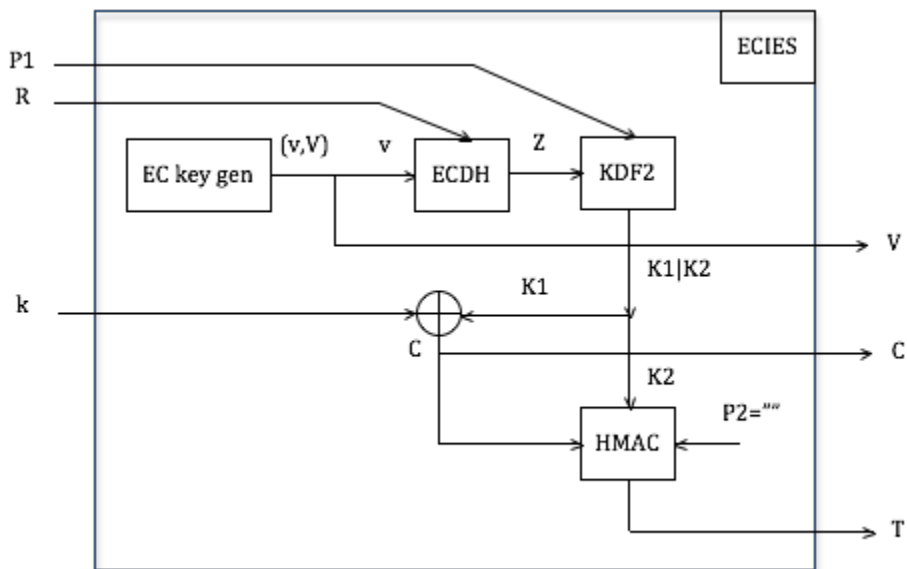


Figure 70 ECIES Encryption

Inputs:

P1: see below

R: recipient's public key

k: AES-CCM symmetric key to be encrypted with ECIES

Outputs:

V: Sender's ephemeral public key

C: Cipher text (encrypted symmetric key k)

T: Tag

Notes:

- KDF is KDF2 [in IEEE 1363a, Section 13.2]:
 $KDF2(Z, P1) = \text{Hash}(Z \parallel \text{Counter} \parallel P1)$,
 where Hash is SHA-256, and the 32-bit counter increases as more output blocks are generated, the output blocks are concatenated to form the KDF output
- MAC2 is HMAC:
 $\text{HMAC}(K2, C) = \text{Hash}(K2 \wedge \text{iPad} \parallel \text{Hash}(K2 \wedge \text{oPad} \parallel C))$,
 where Hash is SHA-256, and iPad and oPad are 256-bit (32-byte) blocks formed by repeating the byte 0x36 and 0x5C, respectively
- in 1609.2-2016
 "Encryption shall use non-DHAES mode. This means that the elliptic curve points shall be converted to octet strings using LSB compressed representation."
 Regarding non-DHAES mode, in IEEE 1363a-2004:
 "The length in bits of the shared secret key K shall be $l + k2$ where $k2$ is the length in bits of the key for the message authentication code (l is the bitlength of the message M to be encrypted). In non-DHAES mode, let $K1$ be the leftmost l bits of K and let $K2$ be the remaining $k2$ bits."
 This means that the KDF output = ENC key | MAC Key, in a (big endian) byte array.
- From 1609.2-2016, Sec 6.3.30:
 On EncryptedData:

This data structure encodes data that has been encrypted to one or more recipients using the recipients' public or symmetric keys as specified in 5.3.4. Sec 5.3.4.1: This section explains how the data is encrypted with a fresh symmetric key generated by the sender, and the symmetric key is then encrypted for the recipient using that recipient's encryption key and refers to Sec 5.3.5 that explains Public Key Encryption using ECIES.

- Sec 6.3.31:
On RecipientInfo:
 - certRecipInfo: The data encryption key was encrypted using the public encryption key in a certificate. This field contains the HashedId8 of the certificate. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the hash of the certificate.
 - signedDataRecipInfo: The data encryption key was encrypted using the public response encryption key from a SignedData. In this case, this field contains the HashedId8 of the 1609Dot2Data containing the SignedData containing the encryption key. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the SHA-256 hash of the 1609Dot2Data containing the response encryption key.
 - rekRecipInfo: The data encryption key was encrypted using a public response encryption key that was not obtained from a SignedData. In this case, this field contains the HashedId8 of the response encryption key. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the hash of the empty string.

1602.2 and SCMS ASN.1 Objects

[scms-protocol.asn:](#)

```
SecuredScmsPDU ::= IEEE1609Dot2Data
```

Figure 71 SCMS-Protocol ASN.1

[1609.2-schema.asn:](#)

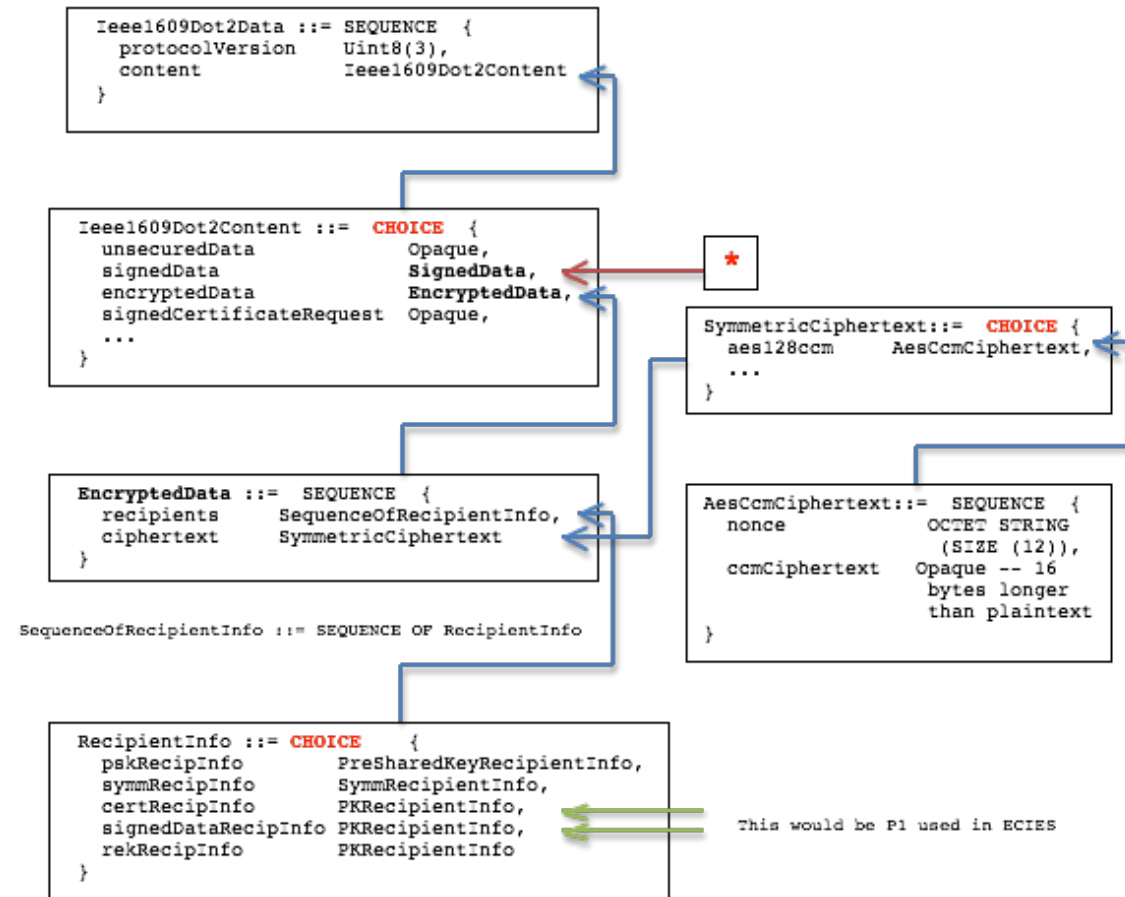


Figure 72 IEEE 1609.2 Schema ASN.1

See [ECIES](#) diagram and Notes on Encrypted Data below, as well as PKRecipientInfo and EciesP256EncryptedKey ASN.1 objects in [1609dot2-schema.asn](#) and EciesP256EncryptedKey in [1609dot2-base-types.asn](#), to see how the outputs of ECIES are encoded in the RecipientInfo.

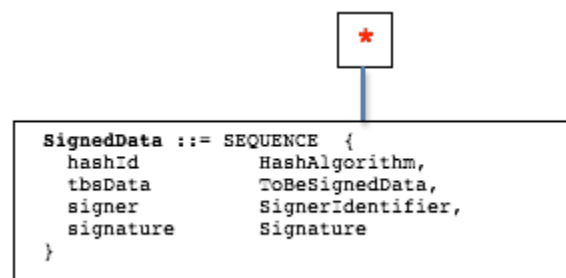


Figure 73 SignedData ASN.1

Example of SignedData from scms-protocol.asn:

```
SignedPseudonymCertProvisioningAck ::= SecuredScmsPDU (WITH COMPONENTS {...,
  content (WITH COMPONENTS {...,
    signedData (WITH COMPONENTS {...,
      tbsData (WITH COMPONENTS {...,
        payload (WITH COMPONENTS {...,
          data (WITH COMPONENTS {...,
            content (WITH COMPONENTS {
              unsecuredData (CONTAINING ScopedPseudonymCertProvisioningAck)
            })
          })
        })
      })
    })
  },
  headerInfo (WITH COMPONENTS {...,
    psid (SecurityMgmtPsid),
    generationTime ABSENT,
    expiryTime ABSENT,
    generationLocation ABSENT,
    certLearningRequest ABSENT,
    missingCrlIdentifier ABSENT,
    encryptionKey ABSENT
  })
})
})
})
})
```

```
ScopedPseudonymCertProvisioningAck ::=
  ScmsPDU (WITH COMPONENTS {...,
    content (WITH COMPONENTS {
      ee-ra (WITH COMPONENTS {
        raEePseudonymCertProvisioningAck
      })
    })
  })
```

Defined in ee-ra.asn:
See EndEntityRaInterfacePDU and
RaEePseudonymCertProvisioningAckMsg

Figure 74 SignedData Example ASN.

ECIES Encryption as in 1609.2-2016, Sec 5.3.5

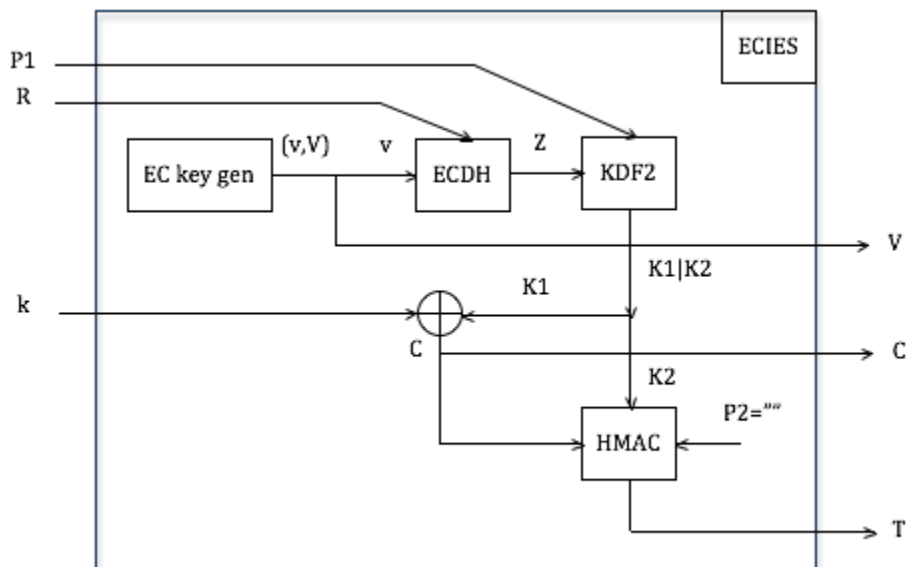


Figure 75 ECIES Encryption

Inputs:

P1: see below

R: recipient's public key

k: AES-CCM symmetric key to be encrypted with ECIES

Outputs:

V: Sender's ephemeral public key

C: Cipher text (encrypted symmetric key k)

T: Tag

Notes:

- KDF is KDF2 [in IEEE 1363a, Section 13.2]:
 $KDF2(Z, P1) = \text{Hash}(Z \parallel \text{Counter} \parallel P1)$,
 where Hash is SHA-256, and the 32-bit counter increases as more output blocks are generated, the output blocks are concatenated to form the KDF output
- MAC2 is HMAC:
 $HMAC(K2, C) = \text{Hash}(K2 \wedge \text{iPad} \parallel \text{Hash}(K2 \wedge \text{oPad} \parallel C))$,
 where Hash is SHA-256, and iPad and oPad are 256-bit (32-byte) blocks formed by repeating the byte 0x36 and 0x5C, respectively
- in 1609.2-2016
 "Encryption shall use non-DHAES mode. This means that the elliptic curve points shall be converted to octet strings using LSB compressed representation."
 Regarding non-DHAES mode, in IEEE 1363a-2004:
 "The length in bits of the shared secret key K shall be $l + k_2$ where k_2 is the length in bits of the key for the message authentication code (l is the bitlength of the message M to be encrypted). In non-DHAES mode, let $K1$ be the leftmost l bits of K and let $K2$ be the remaining k_2 bits."
 This means that the KDF output = ENC key | MAC Key, in a (big endian) byte array.
- From 1609.2-2016, Sec 6.3.30:
 On EncryptedData:

This data structure encodes data that has been encrypted to one or more recipients using the recipients' public or symmetric keys as specified in 5.3.4. Sec 5.3.4.1: This section explains how the data is encrypted with a fresh symmetric key generated by the sender, and the symmetric key is then encrypted for the recipient using that recipient's encryption key and refers to Sec 5.3.5 that explains Public Key Encryption using ECIES.

- Sec 6.3.31:
On RecipientInfo:
 - certRecipInfo: The data encryption key was encrypted using the public encryption key in a certificate. This field contains the HashedId8 of the certificate. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the hash of the certificate.
 - signedDataRecipInfo: The data encryption key was encrypted using the public response encryption key from a SignedData. In this case, this field contains the HashedId8 of the1609Dot2Data containing the SignedData containing the encryption key. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the SHA-256 hash of the leee1609Dot2Data containing the response encryption key.
 - rekRecipInfo: The data encryption key was encrypted using a public response encryption key that was not obtained from a SignedData. In this case, this field contains the HashedId8 of the response encryption key. In this case, the parameter P1 to ECIES as defined in 5.4.5 is the hash of the empty string.

SCMS Policies

SCMS Certificate Policy

The SCMS certificate policy (CP) is a self-contained document identified and managed by Integrity Security Services, a Green Hills Software company. The SCMS V2X Root Certificate Authority CP can be found here: <https://www.ghsiss.com/v2xca>.

SCMS PoC Eligibility Criteria

Eligibility for utilizing the SCMS PoC systems is currently limited to research deployment sites that derive funding from the USDOT. The SCMS POC will only support a select number of connected vehicle applications (mapped by Provider Service Identifier or PSID). Deployers are encouraged to review the list of supported

applications by following this link:

<https://wiki.camplic.org/display/SCP/SCMS+PoC+Supported+V2X+Applications>. Deployment sites that are not funded by the USDOT, research or otherwise, are currently not eligible to enroll in the USDOT SCMS. Deployments that support

commercial activities, state/local maintenance and operations, and production rollout deployments are also not eligible to enroll in the USDOT SCMS. One major reason

for this is that these commercial providers offer certificate services beyond the USDOT SCMS window of 5 years, which is the expected lifespan of the SCMS POC.

Bootstrapping Device Configuration Manager

Policy implementation is delayed until further refinement of the SCMS PoC.

Registration Authority Connectivity

EE devices are required to connect to registration authorities periodically to request certificates and certificate revocation lists (CRLs) and other required files to maintain situational awareness of the SCMS state. Devices connecting to any SCMS component must always request SCMS files upon connection

before requesting any device specific transaction requests. Currently, the EE must request the most recent Local Policy File (LPF) and Local Certificate Chain File (LCCF). The EE must download and process the LPF and LCCF if these files have been updated from the versions stored within the EE.

OBU/ASD SCMS PoC QA System Enrollment

A Vendor requesting enrollment with the SCMS PoC QA system must be a known OBU/ASD vendor that has participated in either the SCMS End Users Group or one of the Certification Operating Councils (COC) DSRC certification plugfests. Vendors must then fill out the SCMS POC QA enrollment form for enrollment with the SCMS PoC QA system, found at https://cvcs.samanage.com/catalog_items/624863-new-scms-poc-enrollment-request-form/service_requests/new.portal?caller=dashboard.

OBU/ASD SCMS PoC Production System Enrollment

Vendors/Pilot Deployments that are seeking enrollment with the SCMS PoC Production System for their OBUs/ASDs, must submit documentation demonstrating how their devices conform to five categories of criteria: Standards Compliance, Comprehensive Installation Planning, Operational Test Readiness Planning, Security Controls and Demonstrated Ability to Interface with SCMS PoC. This documentation must be submitted through the Connected Vehicle Core Systems (CVCS) Samanage website (<https://cvcs.samanage.com/welcome.portal>) using the SCMS POC Production Enrollment Form. Each specific device model and device model software version combination will require a separate enrollment. Additionally, devices that are requesting different PSID and SSP combinations will require separate enrollment. Details of the enrollment process can be found in the SCMS PoC Governmental Implementation Plan. The next five sections provide details on the criteria for the five categories.

Standards Compliance

Vendors/Pilot Deployments requesting enrollment with the SCMS POC Production system must submit itemized results reports from an independent testing facility for conformance to the following industry standards:

1. IEEE 802.11p content as published in the current version of IEEE 802.11
2. IEEE 1609.2, 1609.3 and 1609.4 per the current standards
3. SAE J2945/1 per the current standard
4. SAE J2735 per the current standard

The Omniair led Certification Operating Council (COC) has developed independent tests against portions of these standards, which can be found at https://github.com/certificationoperatingcouncil/COC_TestSpecs, and they also certify independent testing agencies and provide device certifications against these

Comprehensive Installation Plan

Vendors/Pilot Deployments requesting enrollment with the SCMS POC Production system must submit a Comprehensive Installation Plan (CIP) that provides an overview of the proposed installation approach and further identifies the types and number of equipment required to be configured and installed.

Operational Readiness Test Plan

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit an Operational Readiness Test Plan (ORTP) defining the following objectives:

1. Define the test approach;
2. Identify the project items to be tested;
3. Identify the hardware, software, tools, resources and environment to be used to support the testing;
4. Define the types of tests to be performed;
5. Define Resources and Constraints;

6. Define Roles and Responsibilities;
7. Process for change control and communication coordination for key activities.

Security Controls

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must fill out a form on the Connected Vehicle Core Systems (CVCS) Samanage website that requests information about the security controls offered by the OBU/ASD model seeking enrollment. The form can be filled out at <https://cvcs.samanage.com/welcome.portal>. The following tables include the list of questions and controls that are listed in the form.

1. How does the ASD handle bad actors who gain external access via the interface port (or access via a similar nature)?	Check the Device's Protections
The Hardware Security Module (HSM) protects against extraction of the private keys from the operating system or from the CPU.	
Hardware/software solution denies any kind of software modification of any kind of login/access to the operating system.	
The HSM does not allow for reading of private keys, regardless of the state of the application processor.	
The private keys are stored in a one-way storage mechanism with no method of extracting keys.	
The device contains digital signatures such that it verifies the application's authorization before execution to prevent the signing of arbitrary messages.	
The device has a secure boot and programming technique to prevent attacker from accessing the device.	
The device uses and manages cryptographic algorithms, keys, and certificates.	
There is no Application Program Interface (API) to access the HSM.	
Root access to device is protected using two-factor authentication (e.g. password and keys).	

2. How does the ASD handle bad actors who gain internal access to the device and or cryptographic module?	Check the Device's Protections
The HSM prevents an ability to physically remove the storage, cut open the hardware, and/or read-out private key.	
The HSM does not allow connecting directly to internal memory.	
The device has tamper seals to indicate entry of the unit.	
The micro-controller had tamper detection.	
The HSM will "zeroize" if tampering is detected.	
The keys are stored in FLASH and cannot be decrypted after "zeroization."	
The device is FIPS Level 3	
The device is FIPS Level 4	

Devices found to not meet the controls submitted in this form after enrollment has been granted, will have their certificates added to the CRL and denied further access to the SCMS.

Demonstrated Ability to Interface with SCMS PoC

Vendors/Pilot Deployments requesting enrollment with the SCMS POC Production system must demonstrate their ability to interface with the SCMS PoC system. This can be accomplished by:

- Successfully enrolling their specific device model and software with the SCMS PoC QA system
- Successfully requesting Pseudonym and/or Identification Certificates
- Successfully downloading the requested certificates
- Successfully interacting with the Registration Authority

RSU SCMS PoC QA System Enrollment

A Vendor requesting enrollment with the SCMS PoC QA system must be a known RSU vendor that has participated in either the SCMS End Users Group or one of the Certification Operating Councils (COC) DSRC certification plugfests. Vendors must then fill out the SCMS PoC QA enrollment form for enrollment with the SCMS PoC QA system, found at https://cvcs.samanage.com/catalog_items/624863-new-scms-poc-enrollment-request-form/service_requests/new.portal?caller=dashboard.

RSU SCMS PoC Production System Enrollment

Vendors/Pilot Deployments that are seeking enrollment with the SCMS PoC Production System for their RSUs, must submit documentation demonstrating how their devices conform to five categories of criteria: Standards Compliance, Comprehensive Installation Planning, Operational Test Readiness Planning, Security Controls and Demonstrated Ability to Interface with SCMS PoC. This documentation must be submitted through the Connected Vehicle Core Systems (CVCS) Samanage website (<https://cvcs.samanage.com/welcome.portal>) using the SCMS PoC Production Enrollment Form. Each specific device model and device model software version combination will require a separate enrollment. Additionally, devices that are requesting different PSID and SSP combinations will require separate enrollment. Details of the enrollment process can be found in the SCMS PoC Governmental Implementation Plan. The next five sections provide details on the criteria for the five categories.

Standards Compliance

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit itemized results reports from an independent testing facility for conformance to the following industry standards:

1. IEEE 802.11p content as published in the current version of IEEE 802.11
2. IEEE 1609.2, 1609.3 and 1609.4 per the current standards
3. SAE J2735 per the current standard
4. RSU Specification Version 4.1

The Omniair led Certification Operating Council (COC) has developed independent tests against portions of these standards, which can be found at https://github.com/certificationoperatingcouncil/COC_TestSpecs, and they also certify independent testing agencies and provide device certifications against these test suites.

Comprehensive Installation Plan

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit a Comprehensive Installation Plan (CIP) that provides an overview of the proposed installation approach and further identifies the types and number of equipment required to be configured and installed. The following is a template for an CIP.

Operational Readiness Test Plan

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit an Operational Readiness Test Plan (ORTP) defining the following objectives:

1. Define the test approach;
2. Identify the project items to be tested;
3. Identify the hardware, software, tools, resources and environment to be used to support the testing;

4. Define the types of tests to be performed;
5. Define Resources and Constraints;
6. Define Roles and Responsibilities;
7. Process for change control and communication coordination for key activities.

The following is template for an ORTP.

Security Controls

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must fill out a form on the Connected Vehicle Core Systems (CVCS) Samanage website that requests information about the security controls offered by the RSU model seeking enrollment. The form can be filled out at <https://cvcs.samanage.com/welcome.portal>. The following table lists the RSU Specification Version 4.1 security requirements that must be met:

RSU Specification Version 4.1 Security Requirements	
USDOT_RSU-Req_348-v001	USDOT_RSU-Req_351-v002
USDOT_RSU-Req_356-v003	USDOT_RSU-Req_615-v001
USDOT_RSU-Req_616-v001	USDOT_RSU-Req_619-v001
USDOT_RSU-Req_620-v001	USDOT_RSU-Req_621-v001
USDOT_RSU-Req_622-v001	USDOT_RSU-Req_628-v002
USDOT_RSU-Req_629-v001	USDOT_RSU-Req_630-v001
USDOT_RSU-Req_638-v001	USDOT_RSU-Req_640-v001
USDOT_RSU-Req_641-v001	USDOT_RSU-Req_642-v001
USDOT_RSU-Req_643-v001	USDOT_RSU-Req_645-v001

Devices found to not meet the controls submitted in this form after enrollment has been granted, will have their certificates added to the CRL and denied further access to the SCMS.

Demonstrated Ability to Interface with SCMS PoC

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must demonstrate their ability to interface with the SCMS PoC system. This can be accomplished by:

- Successfully enrolling their specific device model and software with the SCMS PoC QA system
- Successfully requesting Application Certificates
- Successfully downloading the requested certificates
- Successfully interacting with the Registration Authority

Back Office System SCMS PoC QA System Enrollment

A Vendor or Pilot Deployment project requesting enrollment with the SCMS PoC QA system for their Back-Office Systems must be a known CV deployment projects that meet the eligibility requirements found in Section 5.1 of this document and have participated in the SCMS End Users Group. Vendors must then fill out the SCMS PoC QA enrollment form for enrollment with the SCMS PoC QA system, found at https://cvcs.samanage.com/catalog_items/624863-new-scms-poc-enrollment-request-form/service_requests/new.portal?caller=dashboard.

Back Office System SCMS PoC Production System Enrollment

Vendors/Pilot Deployments that are seeking enrollment with the SCMS PoC Production System for their Back-Office Systems, must submit documentation demonstrating how their devices conform to five categories of criteria: Standards Compliance, Facility Security, Operational Test Readiness Planning, Security Controls and Demonstrated Ability to Interface with SCMS PoC. This documentation must be submitted through the Connected Vehicle Core Systems (CVCS) Samanage website (<https://cvcs.samanage.com/welcome.portal>) using the SCMS PoC Production Enrollment Form. Each

specific device model and device model software version combination will require a separate enrollment. Additionally, devices that are requesting different PSID and SSP combinations will require separate enrollment. Details of the enrollment process can be found in the SCMS PoC Governmental Implementation Plan. The next five sections provide details on the criteria for the five categories.

Standards Compliance

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit itemized results reports that show DSRC messages created by the Back-Office System are compliant with the current version of the SAE J2735 standard.

Facility Security

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit documentation that shows they are in compliance with the Vendor Facility Security Controls found in Section 6.14 of this document.

Operational Readiness Test Plan

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must submit an Operational Readiness Test Plan (ORTP) defining the following objectives:

1. Define the test approach;
2. Identify the project items to be tested;
3. Identify the hardware, software, tools, resources and environment to be used to support the testing;
4. Define the types of tests to be performed;
5. Define Resources and Constraints;
6. Define Roles and Responsibilities;
7. Process for change control and communication coordination for key activities.

The following is template for an ORTP.

Security Controls

Vendors/Pilot Deployments requesting enrollment with the SCMS POC Production system must submit documentation that shows they are in compliance with Roadside Unit Specification 4.1 Requirements USDOT_RSU-REQ_567-v001 and USDOT_RSU-REQ_586-v001.

Demonstrated Ability to Interface with SCMS PoC

Vendors/Pilot Deployments requesting enrollment with the SCMS PoC Production system must demonstrate their ability to interface with the SCMS PoC system. This can be accomplished by:

- Successfully enrolling their specific device model and software with the SCMS PoC QA system
- Successfully requesting Application Certificates
- Successfully downloading the requested certificates
- Successfully interacting with the Registration Authority
-

End Entity Device and Back Office System Reenrollment

The current QA and Production SCMS PoC systems do not support a dedicated re-enrollment process. If a device needs to be re-enrolled then it must be fully zeroized and must re-accomplish the enrollment and bootstrapping process outlined in the SCMS PoC Governmental Management Concept of Operations (ConOps). Those devices seeking re-enrollment will still be held to the policies outlined in Sections 6.5 through 6.9 of this document.

End Entity Device and Back Office System Certificate Revocation and Blacklisting

The current QA and Production SCMS PoC systems only support a manual certificate revocation process, however, end entities and back office systems can still be reported for certificate revocation and have their current and future certificates added to the appropriate CRL and blacklisted with the SCMS PoC Registration Authority (RA). Devices (OBUs/ASDs, RSUs, Back Office Systems) that are negatively impacting the safety, security or performance of the SCMS PoC QA or Production systems will have their certificates placed on the CRL and will be blacklisted on the RA. A blacklisted device will be ignored by the RA and will not have its requests processed. The following are examples of the types of actions that would result in the revocation of a devices certificates:

- The end entity device or back office system is misbehaving. Misbehavior would include:
 - Sending DSRC messages that adversely impact vehicle safety and/or performance of DSRC communications
 - Interfacing with the SCMS system in a way that adversely impacts performance of the SCMS (e.g. – denial of service (DOS))
- An end entity device or back office system is found to be compromised. A compromised device or system would include:
 - An unauthorized user has gained access to the security or DSRC specific hardware or software resident in an end entity device or back office system
 - An end entity device or back office system was lost and subsequently was found, but the integrity of the device/system cannot be verified
- An end entity device or back office system is found to not be in compliance with the policies found in Sections 6.5 through 6.9 of this document.
- An end entity device or back office system is stolen or lost.
- An end entity device or back office system is sold or transferred to an organization that is not eligible for participation in the USDOT's SCMS PoC project

The process for getting an end entity device or back office system added to the CRL is defined in the SCMS PoC Governmental Management ConOps Section 4.4. If an end entity device and/or back office system has had its certificates revoked, it may be allowed to enroll with the SCMS again if the reason for its revocation has been demonstrated as corrected and it meets the policies outlined in Sections 6.6 through 6.10 of this document.

End Entity Device Software Updates

Software updates to end entity devices must verify that they do not adversely affect the SCMS interface or DSRC communications before they can be deployed to the devices interfacing with the Production SCMS PoC system. Users must demonstrate this through testing with the QA SCMS PoC system before software updates are deployed to end entity devices connected to the Production SCMS PoC system. This does not apply to updates that only change the configuration of end entity devices, assuming that with those new configurations, the devices conform to the policies detailed in Sections 6.6 through 6.10 of this document.

For any change, the vendor must ensure that the changes introduced maintain the certification of the device and software within the device. Failure to maintain proper certification of the device may result in the revocation of the device certificates and removal from SCMS.

Vendor Facility Security Background

The SCMS stakeholders include a number of entities that require secure facilities to ensure continued trust in that entity's V2X communications. These facilities, where devices are bootstrapped or initialized

within SCMS, require increased security. Also, locations that house critical infrastructure components, PKI components, or locations with centralized EE processing require additional security. The trust of the SCMS relies on the security of the keys, keying material, and processing equipment. In addition to physical security controls, logical security controls are required to ensure the security and therefore the success of the SCMS. This policy will define the security requirements for the components and processing centers within the SCMS.

Applicability

This policy is applicable to all SCMS stakeholders' facilities and information systems such as:

- Where secure V2X messages are generated within a data center, server room, or other location
- Vehicle or equipment manufacturing facilities where OBE and RSU bootstrapping activity occurs
- Facilities with Device Configuration Managers (DCM)
- Logical security controls for RSU and OBE

Vehicle SCMS components and RSUs located in the field (at intersections, traffic control signals, or other roadside locations) must be physically protected. However, the protection of this equipment will differ from the traditional server and data center security.

Physical Security Policies

Physical security of the SCMS components must be maintained to protect the integrity of the SCMS program. SCMS PoC participants must ensure the physical security of all components at all times during the operation of the PoC. To accomplish this, operations must:

- Maintain all SCMS equipment within a locked facility at all times where access is limited to only personnel needing access to the SCMS components. For corporate data centers, SCMS components must remain in a physically separated, access-controlled (key locks, logical locks, locked cages, separate room) location from other IT assets (not affiliated with SCMS) and protected at all times from personnel without a need to access the SCMS component
- Physical access to the SCMS components must be logged to ensure a record of each person who accessed or could have accessed the components. Access logs must be collected through badge reader logs (recommended) or a visitor logbook for keyed access
- Critical security components (PKI components, DCM, or other component that accesses or generates multiple device private keying material) must be maintained in the locked area while operating. If operations are not required, the device must be powered off and remain secured while not in use.
- Components with higher availability requirements, such as centralized RSUs, must operate within an environment with redundant power (UPS, generators) to ensure operations for a minimum of 12 hours.

Physical Security of Vehicle and Roadside SCMS components

SCMS components within vehicles and operating roadside also need physical protection. Since these components operate outside of data centers, their security controls must be adjusted to reflect their operating environment.

Vehicle SCMS components (OBE) must be protected from unauthorized tampering. The SCMS module must be installed inside the vehicle in a lockable area of the vehicle (not accessible from under the vehicle). The SCMS module does not need to be separately locked outside of the standard vehicle security system (door locks and alarms).

Roadside SCMS components also need to be protected. Devices at ground level must be protected in locked containers, similar to typical traffic light control panel locked containers. SCMS devices may share this locked space if available. Roadside equipment mounted to traffic poles above ground level should also be locked to prevent tampering. Where available, all roadside devices must be able to generate and deliver tampering alarms back to a central processing location. Municipalities must investigate tampering alarms within 24 hours. If evidence of tampering is discovered, municipalities must notify SCMS PMO within 24 hours of discovery.

Logical Security Policies

To maintain the security of the computing platforms, operations of the SCMS components must include managing the logical security of components. All computing devices within the SCMS must maintain a level of protection to ensure the trust and reliability of the environment. SCMS will facilitate the communication of basic safety messages between vehicles. Threats to the SCMS and its infrastructure components will place lives in danger

- Operating systems and software packages must be patched and updated in a timely manner.
- Anti-virus, network firewalls, intrusion detection, log management and reviews are required to be installed, updated, and regularly reviewed for threats. Organizations are required to investigate any identified threats and report those findings back to the SCMS PMO within two days of discovery.
- While government security requirements are not mandated, SCMS participants are highly encouraged to implement security programs similar to the NIST Risk Management Framework, documented in NIST Special Publication (SP) 800-37 and the security controls identified in NIS SP 800-53, latest revision is NIST 800-53 Rev 4.
- Perform regular vulnerability scanning (at least monthly) and mitigate results in a timely manner
- Independent penetration testing of the environment is highly encouraged at least annually.
- Independent auditing of the computing platforms and information systems is NOT required for PoC.

Additionally, the SCMS must implement security controls to protect the SCMS components, including:

- Access Control – Ensure access to SCMS components is limited to authorized users only, including physical security, logical system, and network access. Accountability for access must be individually based. Shared accounts are not authorized.
- Auditing – System and applications must generate audit events for critical functions and any key event. These event logs must be maintained for a minimum of 90 days. Audit logs must be reviewed on a daily basis and actions taken when anomalies are discovered. Any event involving key misuse, potential key compromise, or message/event compromise/interruption must be reported within 24 hrs. to the SCMS management and US DOT.

Cryptographic Security

Security of the cryptographic components within the SCMS are critical to maintain the trust of the system. SCMS policies surrounding cryptographic modules and key handling are required to ensure trust. Cryptographic modules are utilized within the end entity devices, the PKI components, and any back-end systems supporting the creation of SCMS BS messages. Additionally, support components may also rely on cryptographic modules, including systems for bulk-loading of certificates/key generation and DCMs.

This policy dictates the requirements for all SCMS components where cryptographic keys are created, used, stored, or destroyed.

The National Institute of Standards and Technology (NIST) has developed cryptographic module standards, identified as Federal Information Processing Standard (FIPS) 140-2. SCMS requires all cryptographic modules within SCMS to meet FIPS 140-2 standards. Specifically, SCMS components must meet FIPS 140-2 level 2 for EE devices. Back-end systems, systems with bulk capability, handling multiple key identities, and all PKI infrastructure components must meet FIPS 140-2 Level 3 cryptographic requirements. All cryptographic modules must be formally certified to meet FIPS 140-2 and have an active FIPS 140-2 certificate on file with NIST. Additionally, the cryptographic module must be configured to operate in "FIPS Mode" if turning off FIPS is an option. Additionally, the cryptographic module must be a hardware cryptographic module.

Key Handling:

All SCMS private and secret keys must be protected at all times. Public keys must also be protected until they are bound to an entity with a signed certificate. Public keys contained within a signed certificate may be freely shared and distributed in the clear.

Private keys and secret keys may never be stored or exist in an unencrypted form outside of the boundary of the cryptographic module. Where possible, all private keys must remain within the cryptographic module at all times. Backups of private keys may occur as long as they are protected, encrypted, and controlled. FIPS 140-2 discusses the requirements for key extraction. Private keys should not be backed up with a few exceptions. Infrastructure component keys, where loss would result in the inability to operate the SCMS, may be securely archived for backup purposes. These backups must be stored in a secure area such as another cryptographic module or a vault.

Bulk Loading of Certificates

Vendors may require the ability to bulk-load certificates and keys to expedite manufacturing of devices or vehicles. Ensuring the secure loading of these devices is critical to SCMS security. Each manufacturer must develop procedures for ensuring the security of the devices throughout the bootstrapping process and bulk load. Below are a set of criteria and goals to protect the environment. Each manufacturer must address these areas within their procedures.

- Manufacturers must ensure that all key generation, loading, and processing occurs on an approved FIPS 140-2 device.
- Keys can be generated off device and loaded onto the end entity if the following occurs
 - All keys are generated in an approved FIPS 140-2 device meeting all requirements in this policy and any other policy for SCMS.
 - Keys are exported using key encryption. Split-key is not authorized in automated or bulk situations.
 - Keys are loaded into the end entity device as encrypted, secured data blocks.
 - At no time are private keys exposed outside of the FIPS 140-2 boundary in unencrypted form
 - At no time are FIPS 140-2 key processing and storage requirements bypassed or suspended.
 - If key generation is centralized, private keys are NOT retained once generated and exported if they are loaded into an end entity device.

- Key escrow is not allowed for private, signature keys.
- Pseudonym certificates are not bulk loaded through manufacturing. These certificates must be requested by the end entity device and loaded through normal communications. Manufacturers may implement connectivity to obtain these certificates but are not authorized to request or load pseudonym certificates on behalf of the device.
- Physical security of the end entity devices, DCM, and any component supporting the bulk loading must be protected in accordance with the physical security policies as identified in section 5.13.3 of this document.
- Networks utilized for bulk loading must be isolated from other communications networks while loading certificates and keys. Connectivity to SCMS for certificate generation, certificate retrieval, or other transaction must occur on a separate network from any network processing bulk key generation activities.
- Physical access to the end entity devices or bulk-loading equipment must be strictly controlled and only accessible to personnel with a strong need to access those components. Access during bulk loading and key generation should be further restricted to only personnel required to manage those processes.
- Any SCMS component (e.g. ECA) operated at the manufacturing site for the purpose of bulk loading or generating certificates for bulk devices must be protected in accordance with the SCMS Certificate Policies and Certification Practice Statement.

Currently, SCMS is not implementing bulk loading of devices. As more details of required processes emerges, this policy will be updated to reflect the needs of the SCMS.

Addition of SCMS Application

Any user that wants to add an application that does not have an existing PSID registered with IEEE as of June 30, 2017, will be required to utilize one of the CAMP reserved SCMS PSIDs. CAMP has reserved 16 PSIDs for future use. These PSIDs will be provided on a first come, first served basis. To request the use of one of these PSIDs, a user must follow the process identified in Section 4.7 of the SCMS Governmental Management ConOps. Users will need to formally request the use of a reserved PSID, then submit an application for that PSID to the IEEE 1609 working group for approval. If approved by IEEE, then the user will receive the PSID and also be responsible for defining the Service Specific Permissions (SSP) for that PSID.