Device and Credentials Recoverability

Introduction

This document describes expectations on what device partners should implement to achieve a device Root of Trust (RoT) that is the trusted anchor to enable secure recovery and provisioning, and re-provisioning for Digital Rights Management (DRM) systems.

Furthermore, this document aims to facilitate a common understanding about renewal and provisioning including the necessary functional requirements, the use cases and the threat models applicable to those use cases. The aim is to guide technology providers in the design of the necessary software and hardware capabilities, and for streaming service providers to deploy reliable solutions with a good user experience.

Audience

This document is targeted for security and engineering teams in the device partner organizations responsible for implementing the requirements with respect to device and credentials recoverability.

Criteria for success

- The definition is easy to understand.
- The definition provides the device partner with actionable expectations.
- The definition builds a practical intuition for the main use cases.
- The definition is DRM scheme-agnostic
- The definition is hardware and software implementation agnostic.

User Roles

The following user role definitions are used in this document.

User Role	Description
Consumer	The role assigned to a person, or persons, receiving content through a protected terminating device. E.g. A typical user with a device provisioned for the service within the Service Provider's delivery network.
Security Provider	The role assigned to an entity that maintains the end-to-end security of the device and the backend services.

	E.g. A group of one or more parties such as Original Equipment Manufacturer (OEM), System On-Chip (SOC) and DRM service providers.
Service Provider	The role assigned to an entity that operates and manages the systems to deliver protected content. E.g. A full-stack telecommunications provider or a cloud-based Over The Top (OTT) delivery service.

Definitions

The following general definitions are used in this document.

Term	Description
Application RoT (ARoT)	An application specific service that is trusted through verification anchored in the PRoT. E.g. DRM TCB (DRM TA, TEE OS etc.)
Backend Services	Trusted (remote) services provided by the Service or the Security Provider necessary for the use cases defined below. E.g. Secure Update Service, Verification Service or Provisioning Service.
Boot Read Only Memory (ROM)	The first (immutable) code to execute in the device after release from reset. The Boot ROM must be on-chip. Typically implemented as Mask ROM, locked OTP Fuse or On-chip Flash memory.
Digital Rights Management (DRM) TCB	The TCB that is critical for the security of a DRM subsystem.
Immutable PRoT	The part of the PRoT that is inherently trusted and never changes on a production device. E.g. Boot ROM, OTP Fuses.
Isolated Location	Contain security sensitive data such as device keys, DRM secrets etc. These are typically non-volatile and apply only to very limited areas of on-chip storage. E.g. OTP Fuse or Flash memory.

One-Time Programmable (OTP) Fuses	A special type of non-volatile memory (NVM) that is programmable once (to a logical 1), after which the memory cannot be reset or re-programmed (to a logical 0).
Platform Root-of-Trust (PRoT)	The set of components that serve as the trust anchor for security implementations in the device. The PRoT consists of Immutable and Updateable elements.
PRoT Services	A set of generic security related functions that are bound in some way to the PRoT. E.g. Secure storage, Cryptography Service, Attestation Service.
Trusted Computing Base (TCB)	The set of hardware and/or software components that are critical for the security of a system.
Trusted Facility	A facility under the control of the Security Provider that is implicitly trusted. E.g. Manufacturing facility.
Updateable PRoT	The part of the PRoT that is changeable in the device and trusted through verification anchored in the Immutable PRoT. The Updateable PRoT provides implementations of PRoT Services.

Use Cases

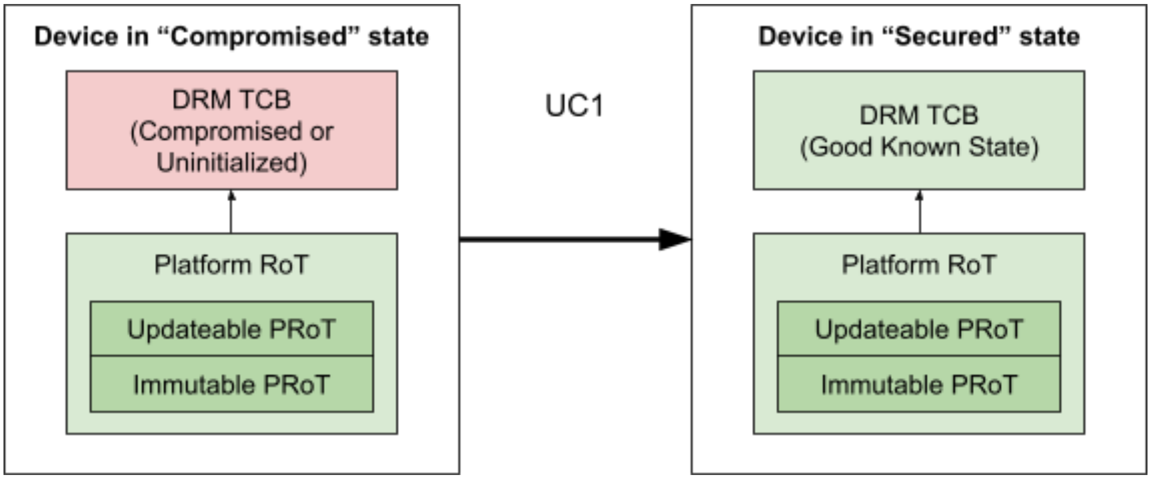
Let us assume that the device in question is in a security compromised state where a vulnerability has been found in the trusted computing base (TCB) associated with the DRM solutions and the device needs to be recovered to a good known state.

The threat model and the functional security requirements are defined further based on the use case definitions below.

Note:
The concepts and terminology necessary to consume the threat model and functional security requirements are provided next in the Concepts and Terminology section.

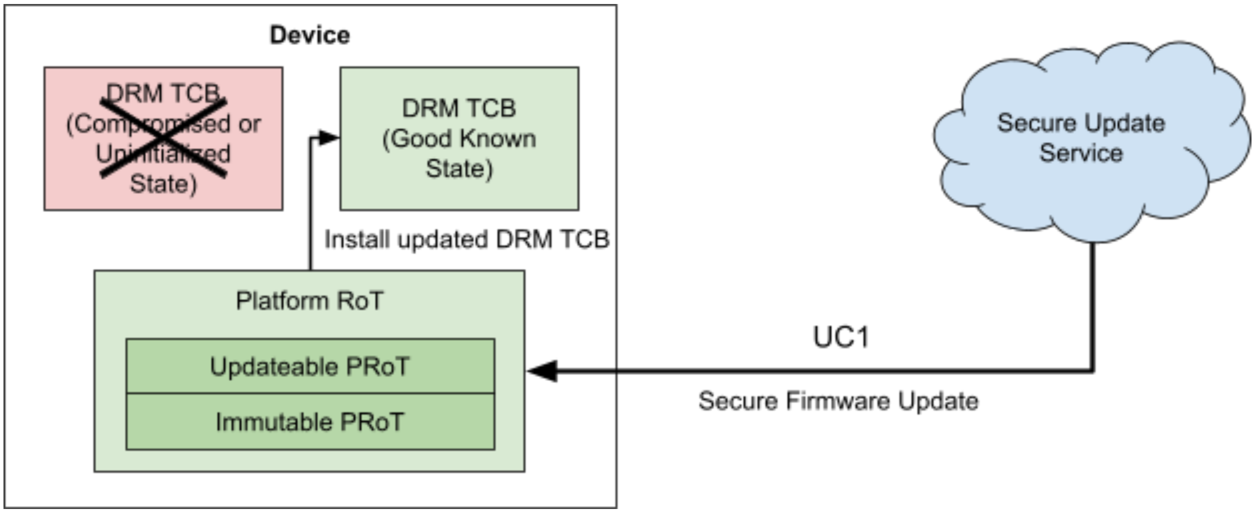
UC1: TCB Initialization and Recovery

The goal of UC1 is to perform TCB initialization and recovery. To illustrate this, we use the DRM TCB as an example of being uninitialized or compromised in the following diagram.


Device in "Compromised" state
DRM TCB (Compromised or Uninitialized)
Platform RoT
Updateable PRoT
Immutable PRoT
UC1
Device in "Secured" state
DRM TCB (Good Known State)
Platform RoT
Updateable PRoT
Immutable PRoT

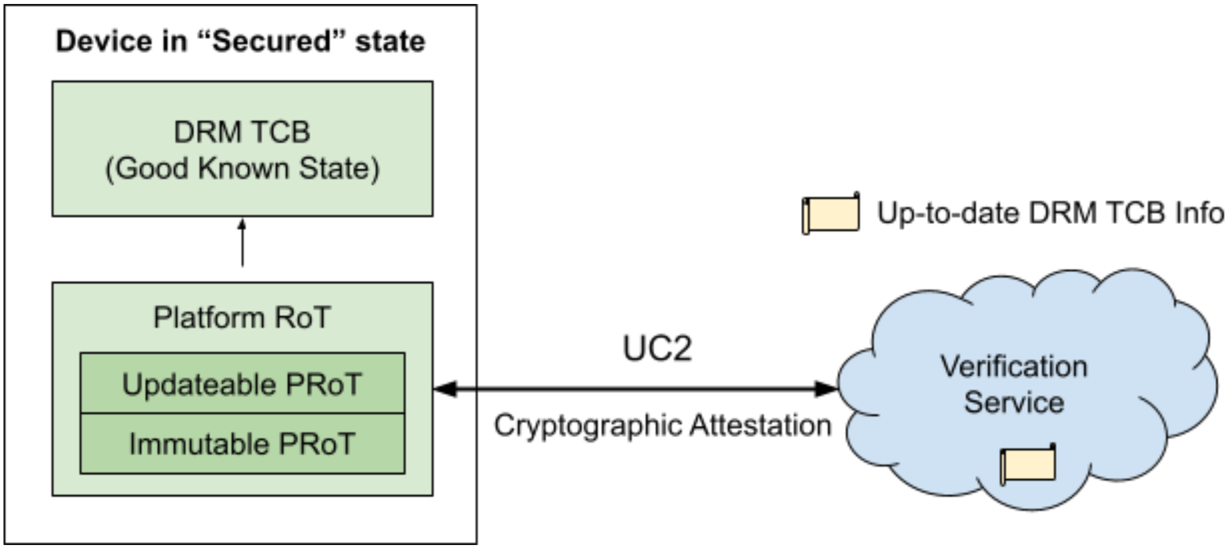
TCB Initialization and Recovery is typically accomplished by issuing an over-the-air secure firmware update to the device, containing the latest DRM TCB components. This may be the first released version or a later version which has fixes for identified vulnerabilities.

This action is performed by the Security Provider.



UC2: Proof of Security State of DRM TCB

The goal of UC2 is to provide cryptographic proof that the device has been recovered to the “Secured” state. Though that is the goal, it is also important to state the corollary here, which is that the failure of a device to provide a cryptographic proof would mean that it is not in the “Secured” state.



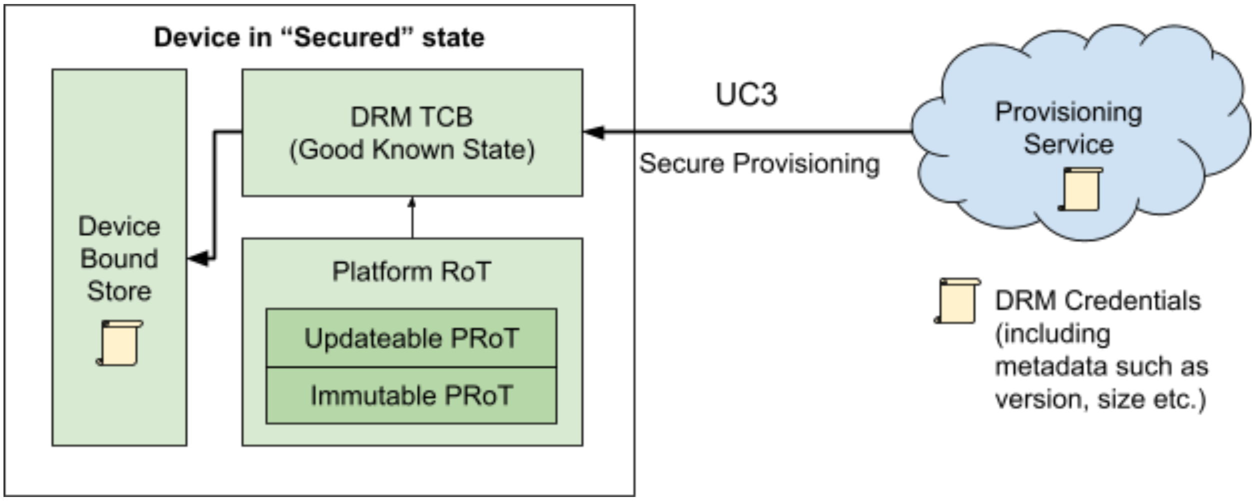
The verification of device recovery is expected to be performed by a remote verification service, which contains up-to-date DRM TCB information. The device itself is not expected to know if it is in a compromised state or not.

This action may be performed by the Service Provider(s) or the Security Provider.

UC3: DRM Credentials Provisioning

Once verified, the device can be re-provisioned with DRM credentials, which is the goal for UC3. If the device is recovered from a DRM TCB compromise (and not the DRM credentials themselves), it is expected that the same credentials are re-provisioned in the device. On the contrary, if the DRM credentials were compromised, it is expected that new credentials are provisioned in the device.

This action is performed by the Security Provider.



Additional Goals Realized

In addition to the above (UC1, 2 and 3), the following goals are enabled with the above use cases.

- **Determining DRM TCB compromise**: Determining that the current DRM TCB in a device is compromised would entail:
    - Performing UC2 by a remote verification service which has up-to-date DRM TCB information on the device.
- **Recovery of DRM credentials**: If the DRM credentials are compromised while the DRM TCB is preserved in a good known state, recovery of the DRM credentials would entail:
    - Performing UC2 to verify the device is in a secured state by a remote verification service.
    - Performing UC3 to provision new DRM credentials from a provisioning service.
- **In-field provisioning of DRM credentials**: A device deployed to the field without any provisioned DRM credentials may follow the same steps as **Recovery of DRM credentials**, to provision the DRM credentials in the device from a provisioning service.

Concepts and terminology

Device Security Architecture

The device security architecture described here is based on the PSA Security Model, part of the Platform Security Architecture (PSA) programme[1,2]. The Security Model defines the top-level concepts considered necessary for a secure compute-centric connected platform. In the context of CDSA, the platforms are compute centric, perhaps assisted by some selected dedicated hardware.

The Security Model[3] (SM) identifies a Secure Processing Environment (SPE) and a Non-secure Processing Environment (NSPE). The SPE must be isolated from the NSPE by hardware means.

- The SPE hosts the Platform Root of Trust, the Platform Root of Trust Services and any Application Root of Trust Service(s).
- The NSPE hosts all the non-secure system software and application-specific software components that provide the required device functionality. Typically, the system software may comprise an operating system together with any middleware, standard stacks and libraries, chip-specific device drivers, etc.

As specified in the Definitions section, the Platform Root of Trust (PRoT) consists of an Immutable PRoT and an Updateable PRoT. The Immutable PRoT typically means hardware, which from the security perspective, is inherently trusted and is the trust anchor. As an example, the Boot ROM is part of the Immutable PRoT. However, its immutability means that any fault in all likelihood cannot be rectified without some level of chip redesign. For this reason, the PSA SM advocates keeping the immutable part as small as possible, essentially to facilitate a high level of design verification and any penetration testing. The rest of the PRoT functionality is provided, therefore, in the Updateable PRoT. The balance is a matter for the chip designer.

In the context of this document, the use cases listed concern the update and/or the recoverability of ARoTs that implement the DRM security sensitive services. In the case where a TEE is deployed, the update and/or the recoverability must consider the DRM Trusted Application (TA) and the Trusted Execution Environment Operating System (TEE OS).

Isolation

Isolation ensures that less trusted software cannot compromise more trusted software. More generally, this means that software in one component cannot compromise the code, run-time state, and secrets of any other component.

[1] The Platform Security Architecture programme was established by Arm to address the lack of security awareness, initially, for IoT devices. All PSA documentation mentioned here is agnostic of any Arm architecture and implementation.

[2] https://developer.arm.com/architectures/architecture-security-features/platform-security

[3] https://www.psacertified.org/app/uploads/2020/10/DEN0079_PSA_SM_BETA-0.pdf

Isolation of the SPE from the NSPE has already been mentioned. However, isolation should apply also to Isolated Locations, dedicated security related IP, as well as the more software centric isolation provided through memory management at the Operating System and or the Hypervisor levels.

Platform Security Parameters

A device platform requires at least the immutable parameters, or equivalent, listed below.

Parameter	Definition
Implementation ID	Non-secret data that uniquely identifies an implementation. Typically, this will allow identification of the end consumers device type through the device manufacturer, the model and version number, and any other data necessary to uniquely identify the Immutable PRoT.
Instance ID	A public value that identifies the specific end consumer device instance and hence the Initial Attestation Key associated with that device.
Hardware Unique Key	A secret key unique to each device instance that is used to derive other device unique secrets and to bind data to that device instance.
Boot Validation Key	Used for authentication during the secure boot process.
Initial Attestation Key	The private part of an asymmetric key-pair accessible only to the Attestation service.

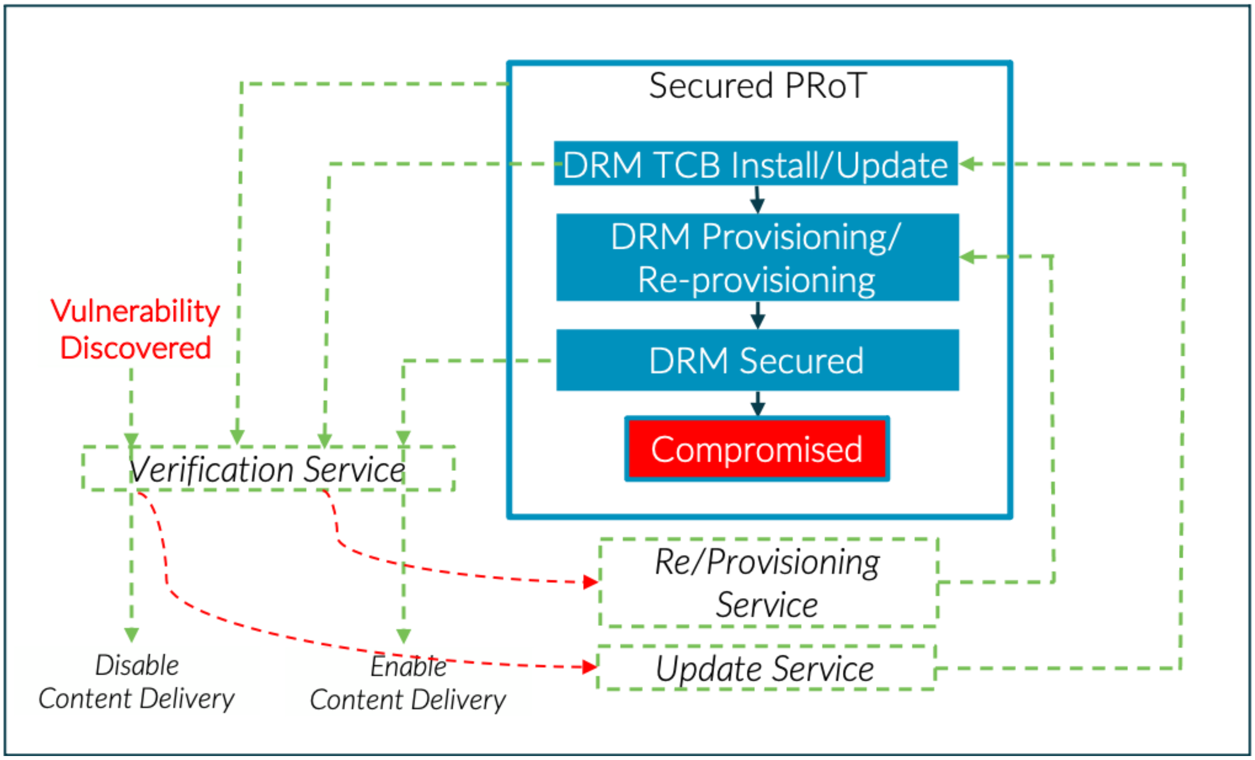
DRM TCB Security Lifecycle

A lifecycle defines the states of an object through its lifetime. Each state in the security lifecycle defines the security properties in that state. The states supported will be implementation specific.

As a guide, the PSA Security Model identifies the typical states of the PRoT, and the essential properties. In the context of this document the assumption is that the PRoT is in the “Secured” state, as defined in the PSA Security Model. Only when the PRoT is in this “Secured” state should the DRM TCB and its credentials be considered to be secure and any attestation claims valid. Thus the lifecycle state should be reflected in any attestation claim.

The diagram below shows a typical lifecycle of the DRM TCB and its provisioned credentials. This is on the assumption that the PRoT is in the “Secured” state, which means that any attestation claims signed by the PSA RoT Initial Attestation Service in any of the states shown can be used by a relying party. This document is concerned with the steps necessary to return the device to the “DRM Secured” state.

The states actually supported will be implementation specific. Note also that there may be more than one DRM installation, each with its own lifecycle and state.



- **DRM TCB Install/Update**: the device either has no DRM installed or the existing version is being updated. The DRM system is not considered to be secured. Anti-rollback checks can be part of the attestation claims.
- **DRM Provisioning/Re-provisioning**: The DRM credentials are either being provisioned (for a new installation) or re-provisioned (in the event of recovery from a vulnerability). Credentials should only be provisioned once the DRM TCB has been verified as correctly installed. The DRM system is not considered to be secured.
- **DRM Secured**: the DRM is fully operational and secured. This is the primary security state for most of the life of the DRM. Only in this state should the DRM system be used for the management of DRM protected content.
- **Compromised**: the DRM system is determined to have been compromised. The compromise may be detected on the device (a reporting mechanism, not shown, is required) or may be off-device. The DRM eco-system is then responsible for determining the consequences and the corrective action.

Secure Boot and Secure Update

All devices must support a secure boot flow to ensure only authorized software can be executed on the device. Secure boot, sometimes called verified boot, uses cryptography to verify the code and associated metadata for the next stage of boot. Secure boot starts from the Boot ROM in the Immutable PRoT. Execution of the next stage proceeds only if any validation checks on the verified metadata pass, for example, firmware hash, version comparison etc. The secure boot flow must apply to all the firmware and software in the SPE including the DRM TCB. It is recommended that the PRoT anchored secure boot flow also authenticates the first stage of the NSPE code. Secure boot considerations are covered in the Arm-agnostic Boot-Platform Security Guide document[4].

Update of firmware is crucial for fixing security vulnerabilities and enhancing the features of devices that are already deployed. It is essential that the update mechanism cannot be used to compromise the device with unauthorized software.

Anti-rollback is used to reject earlier versions of the firmware, software or data that may contain known errors or vulnerabilities. Secure boot must only allow components that have the same or newer (typically higher) version number than the reference version number for that component to be executed. Anti-rollback is closely tied to recovery and specific requirements are captured in the functional requirements section.

Attestation

The mechanisms above can be used to construct a secure device around a Root of Trust. A user or client of software deployed on that device requires a way to assess that the device is actually secure. In that, they are termed a 'relying party'. The method used is to acquire an Attestation generated by the device and examine that Attestation to assess trustworthiness. Attestation is normally acquired by making a call to a secure service.

Attestation is the generation of a statement containing a set of claims that describe the construction and the state of the device. The claim set should include the security state of a device. This may include:

- the identity of the hardware implementation
- measurements taken at boot time for the device firmware
- software versions
- run-time measurements identifying the device workload
- hardware configuration (eg. any OTP fuse settings)
- the status of any debug ports

Claims may also include statements about the product lifecycle phase, for example, development, deployment, returns, and end-of-life.

Once evidence in the form of claims has been created, that set of claims must be verified and proven to be valid to the satisfaction of the security policy required by the relying party. In this way the trustworthiness of the device can be established. A PSA RoT must include an attestation service able to cryptographically sign the claim made by the device. This provides cryptographic proof that the claim has been generated by the identified device. Often, a trusted

[4] https://developer.arm.com/architectures/architecture-security-features/platform-security

verifier that has been provisioned with RoT identities and reference values for measurements will be used to verify the claim set.

Use case UC2 requires Attestation in order to assess that the fixed firmware version is part of the security state of the device. Information in the Attestation may also be used as part of the binding state to satisfy UC3.

Threat Model

Common Threats

The list of common threats are provided below.

ID	Asset	Threat Description
COM.T1	Immutable PRoT	An attacker may exploit a vulnerability in the Immutable PRoT to gain access to and/or extract device root secrets.
COM.T2	Updateable PRoT	An attacker is able to load a cryptographically signed but known compromised version of the Updateable PRoT.
COM.T3	Updateable PRoT	An attacker is able to roll-back to a compromised version of the Updateable PRoT if the greatest version number recorded in the device for it can be replaced with a lesser value.
COM.T4	Sensitive Data exchanged over Communication Channel	An attacker is able to use a network sniffer (man-in-the-middle) to access or modify the data exchanged between the device and any remote server.
COM.T5	Sensitive Data exchanged over Communication Channel	An attacker is able to replace the server certificate with a malicious one in the device's trust store and try to establish connection with a malicious server.

UC1 Threats

The list of assets and threats for UC1 are provided below.

ID	Asset	Threat Description
UC1.T1	Immutable PRoT	An attacker can continue to use an immutable PRoT attack to perform ongoing attacks against the DRM TCB.
UC1.T2	DRM TCB	An attacker is able to load a cryptographically signed but known compromised version of the DRM TCB.
UC1.T3	DRM TCB	An attacker is able to roll-back to a compromised version of the DRM TCB if the greatest version number recorded in the device for it can be replaced with a lesser value.

UC2 Threats

The list of assets and threats for UC2 are provided below.

ID	Asset	Threat Description
UC2.T1	Attestation Claims	An attacker may modify or add new claims to be attested if the claims are gathered from untrusted locations in the device.
UC2.T2	Attestation Private Key	An attacker is able to do a BORE (Break-Once-Run-Everywhere) attack to generate attested claims for multiple devices if the same private key (of the key pair) is used across devices.
UC2.T3	Attestation Private Key	An attacker is able to forge attestation claims if the private key is not protected in the device.
UC2.T4	Attestation Certificates	An attacker is able to take control of attestation service, if the device certificate is maliciously signed by a key under attacker's control.
UC2.T5	Attestation Claims	If the known good values of attestation claims are not protected in the attestation verification service, an attacker is able to falsify information and prevent detection of TCB compromise.
UC2.T6	Attestation Claims	An attacker is able to replay the attestation response if there is no nonce used for each attestation request.

UC3 Threats

The list of assets and threats for UC3 are provided below.

ID	Asset	Threat Description
UC3.T1	DRM Credentials	An attacker is able to load a cryptographically signed but known compromised version of the DRM credentials.
UC3.T2	DRM Credentials	An attacker is able to access or modify the DRM specific secrets embedded in DRM credentials if they are not protected.
UC3.T3	DRM Credentials	If the DRM credentials are not device bound, an attacker is able to re-use them in other devices without having access to the secrets.
UC3.T4	Cryptographic	If the keys used to decrypt and integrity verify the DRM

	Keys	credentials are NOT protected in the device, an attacker is able to read or modify the DRM secrets embedded in the DRM credentials.

Functional Requirements

Please refer to IETF RFC 2119 for definitions of terms used in this document.

Assumptions

1. Secure Boot (and Secure Firmware Update) is assumed to be available and enabled in the Platform to prevent unauthenticated code execution. The requirements around secure boot (and secure firmware update) are considered out of scope of this document.
2. If a compromise is detected by the attestation Verification Service (UC2), the Service Provider is expected to take an appropriate action in response, such as:
    a. Identify the affected devices via the attestation Verification Service (UC2), and decide whether to downgrade the service offered to those devices.
    b. Work with the Security Provider to trigger a secure update to recover the TCB in the device.
    c. Use the Verification Service (UC2) to confirm that the affected devices have deployed the corrective actions, which may include the provisioning of new credentials (UC3) and decide on the level of services to the recovered devices.

Common Requirements

ID	Requirement
COM.R1	The Platform RoT SHALL contain an Immutable PRoT and an Updateable PRoT as stated in the definitions sections.
COM.R2	The Immutable PRoT SHALL contain the minimal necessary and sufficient code logic to bring up the platform and perform cryptographic verification of the Updateable PRoT.
COM.R3	The Updateable PRoT SHALL be associated with a trusted monotonically increasing version number.
COM.R4	The Immutable PRoT SHALL record the greatest version number of Updateable PRoT in an Isolated Location in the Platform RoT.
COM.R5	The Immutable PRoT SHALL perform anti-rollback checks (>= version number recorded in Isolated Location) of the Updateable PRoT during device boot.
COM.R6	The Immutable PRoT SHALL be provisioned with Platform Security Parameters in a trusted manufacturing facility.
COM.R7	The Platform Security Parameters SHALL remain immutable in the device.
COM.R8	The Platform RoT SHALL provide a mechanism to recover the Updateable

	PRoT in the device. Examples of scenarios warranting a recovery would be any unintended image corruptions in storage, rollback detection etc.
COM.R9	All network connections between the device and the remote servers (Secure Update Service, Attestation Verification Service, Provisioning Service) SHALL be protected against unauthorized access and unauthorized modifications of the data exchanged, using TLS or a comparable scheme. If using TLS, please refer to RFC 7525 for recommendation on the protocol version and the cipher suites.
COM.R10	For securing the network connection, the identity of the remote servers SHALL be authenticated and the identity of the device SHOULD be authenticated.
COM.R11	The trust anchor corresponding to the remote servers SHALL be protected from unauthorized modification in the device.

UC1 Requirements

ID	Requirement
UC1.R1	The Platform SHALL contain a TCB recovery capability that can be triggered to download an updated DRM TCB from a remote server (e.g. Secure Update Service) in case of a compromise.
UC1.R2	The DRM TCB SHALL be associated with a trusted monotonically increasing version number.
UC1.R3	The Updateable PRoT SHALL record the greatest version number of the DRM TCB update package in an Isolated Location in the Platform RoT.
UC1.R4	The Updateable PRoT SHALL perform anti-rollback checks (>= version number recorded in Isolated Location) of the DRM TCB during download and during device boot to prevent an older compromised version of the DRM TCB to be reinstated in the device.
UC1.R5	The Updateable RoT SHALL provide a mechanism to recover the DRM TCB in the device. Examples of scenarios warranting a recovery would be any unintended image corruptions in storage, rollback detection etc.

UC2 Requirements

ID	Requirement
UC2.R1	The Updateable PRoT SHALL implement an attestation mechanism to provide cryptographic evidence of the security state of the Platform RoT and the DRM

	TCB to a verifier.
UC2.R2	The set of claims included in the cryptographic evidence for attestation SHALL be aggregated by the Platform RoT from trusted sources in the device such as information in Isolated Locations; information such as firmware measurements (hash digests), version number etc. that are cryptographically verified using a public key anchored in Platform RoT.
UC2.R3	The Updateable PRoT SHALL rely on an asymmetric private key, namely the Initial Attestation Key, anchored (stored or derived) in the Platform RoT for generating the attestation signature.
UC2.R4	The asymmetric public key (tied to a Leaf Certificate) SHALL be chained to a Root Certificate that is known to the Verification Service.
UC2.R5	The chain of trust between the Leaf Certificate and the Root Certificate MAY contain intermediate certificates corresponding to the device model, manufacturer etc.
UC2.R6	The establishment of the certificate chain of trust from the Leaf Certificate to the Root Certificate SHALL be done in a trusted facility.
UC2.R7	The Verification Service SHALL prevent unauthorized modification of known good values of attestation claims for devices.
UC2.R8	The Verification Service SHOULD have the ability to detect device clones (re-use of a compromised private key across more than one device). This may be accomplished by detecting the re-use of the same public key information across multiple devices.

UC3 Requirements

ID	Requirement
UC3.R1	The DRM TCB SHALL implement a provisioning service to ingest DRM credentials from a remote Provisioning Service.
UC3.R2	The DRM TCB SHALL be in a good known state (latest uncompromised version) to perform DRM credentials provisioning.
UC3.R3	The DRM credentials SHALL be identified as valid or compromised based on the group level identifier[5] provided by the underlying DRM subsystem.
UC3.R4	The DRM credentials (including its metadata) SHALL be encrypted and integrity protected using a key that is device or model unique, during storage and

[5] Group-level DRM-based identifiers include system ID for Widevine and model cert hash for PlayReady.

	provisioning from the Provisioning Service.
UC3.R5	If a device unique key is used for encryption and integrity protection of the DRM credentials, it SHALL be derived from the Hardware Unique Key associated with the device as per PSP.R8.
UC3.R6	If a model unique key is for encryption and integrity protection of the DRM credentials, it SHALL be provisioned in the device in a trusted manufacturing facility and bound to the device using a key derived from the Hardware Unique Key.
UC3.R7	The DRM credentials SHALL be accepted in the device, only if the integrity verification is successful and the version number is greater than the one provisioned in the device.
UC3.R8	The DRM credentials (including its metadata) SHALL be encrypted and integrity protected using a key derived from the Hardware Unique Key for storage in the device.

Platform Security Parameters Requirements

The following requirements pertain to the Platform Security Parameters that are used in the use cases UC1, 2 and 3.

ID	Requirement
PSP.R1	The Platform Security Parameters SHALL be immutable in the device.
PSP.R2	The Initial Attestation Key (asymmetric private key) used in UC2 SHALL NOT be accessible outside the Platform RoT.
PSP.R3	The Initial Attestation Key SHALL be solely used for attestation purposes.
PSP.R4	The Initial Attestation Key SHALL be unique to a device or to a group of devices as defined by the term ‘security grouping’ in the Device Identity document.
PSP.R5	The Hardware Unique key SHALL be device unique.
PSP.R6	The Hardware Unique Key SHALL be generated from a Cryptographically Secure Random Number Generator (CSRNG).
PSP.R7	The Hardware Unique key SHALL NOT be used directly, instead new keys SHALL be derived off of it for different purposes.
PSP.R8	An approved Key Derivation Function (KDF) such as IETF RFC 5869 or NIST SP 800-108 SHALL be used for any key derivation purposes.
PSP.R9	The Hardware Unique Key SHALL NOT be accessible outside the Platform

	RoT.

Companies/Contributors

(Listed in Alphabetical Order of Company Name)

Company	Contributors
ARM	Michael Lu, Rob Smart, Simon Frost
Google	Jon Dahlke, Frederic Porter, Prashant Grover, Alex Lee
Microsoft	Sam Wenker, Rahul Dey
Nagra	Laurent Piron
NBC Universal	Alex Olugbile
Netflix, Inc.	Scott Kelly, Srikanth Varadarajan
Sony Pictures Entertainment	Eric Diehl
Warner Media	Mark Nakano

TODO

- Companion doc with implementation specific guidance for UC1 on Arm.
    - Best Practice and more detailed specifications for Arm Based devices.
- Companion doc with implementation specific guidance for Widevine and Playready for UC3.
- Future definitions may be needed in the implementation specific guidelines.
- Need to add a section on Privacy.