Protection Profile for Mobile Device Management

Version: 4.1-Draft
2024-11-15
National Information Assurance Partnership

Revision History

Version	Date	Comment
1.0	2013-10-21	Initial Release
1.1	2014-02-07	Typographical changes and clarifications to front-matter
2.0	2014-12-31	Separation of MDM agent SFRsUpdated cryptography, protocol, X.509 requirements. Updated management functions to match MDFPPv2.0. Included SSH as a remote administration protocol. Removed IPsec as protocol to communicate to MDM agent. Added X509 enrollment objective requirement. Added Optional Mobile Application Store requirements.
3.0	2016-11-21	Updates to align with Technical Decisions Added requirements to support BYOD use case Removed IPsec and SSH requirements, which are now contained in EPs
4.0	2018-09-24	Updates to align with Technical Decisions Removed platform dependency Removed TLS SFRs and use the TLS Functional Package Allowed for a distributed TOE
4.1	2024-11-15	Updates to align with Technical Decisions Updates to align with CC:2022

Contents

1 Introduction

1.1 Compliant Targets of Evaluation

The Mobile Device Management (MDM) system consists of two primary components: the MDM server software and the MDM agent. Optionally, the MDM system may consist of a separate Mobile Application Store (MAS) server.

1.1.1 TOE Boundary

The MDM system operational environment consists of the mobile device on which the MDM agent resides, the platform on which the MDM server runs, and an untrusted wireless network over which they communicate, as pictured below.

Figure 1: MDM System Operating Environment

The **MDM server** is software (an application, service, etc.) on a general-purpose platform, a network device, or cloud architecture executing in a trusted network environment. The MDM server provides administration of the mobile device policies and reporting on mobile device behavior. The MDM server is responsible for managing device enrollment, configuring and sending policies to the MDM agents, collecting reports on device status, and sending commands to the agents. The MDM server may be standalone or distributed, where a distributed TOE is one that requires multiple distinct components to operate as a logical whole in order to fulfill the requirements of this PP (a more extensive description of distributed MDMs is given in section 3).

The **MDM agent** establishes a secure connection back to the MDM server controlled by an enterprise administrator and configures the mobile device per the administrator's policies. The MDM agent is addressed in the PP-Module for MDM Agents. If the MDM agent is installed on a mobile device as an application developed by the MDM developer, the PP-Module extends this PP and is included in the TOE. In this case, the TOE security functionality specified in this PP must be addressed by the MDM agent in addition to the MDM Server. Otherwise, the MDM agent is provided by the mobile device vendor and is out of scope of this PP; however, MDMs are required to indicate the mobile platforms supported by the MDM server and must be tested against the native MDM agent of those platforms.

The **Mobile Application Store (MAS)** hosts applications for the enterprise, authenticates agents, and securely transmits applications to enrolled mobile devices. The MAS functionality can be included as part of the MDM server Software or can be logically distinct. If the MAS functionality is on a physically separate server, then the TOE is distributed with the MDM server and MAS server being separate components.

1.2 Terms

The following sections list Common Criteria and technology terms used in this document.

1.2.1 Common Criteria Terms

Assurance	Grounds for confidence that a TOE meets the SFRs [CC].
Base Protection Profile (Base-PP)	Protection Profile used as a basis to build a PP-Configuration.
Collaborative Protection Profile (cPP)	A Protection Profile developed by international technical communities and approved by multiple schemes.
Common Criteria (CC)	Common Criteria for Information Technology Security Evaluation (International Standard ISO/IEC 15408).
Common Criteria	Within the context of the Common Criteria Evaluation and Validation Scheme (CCEVS), an IT security evaluation facility accredited by the National Voluntary Laboratory

Testing Laboratory	Accreditation Program (NVLAP) and approved by the NIAP Validation Body to conduct Common Criteria-based evaluations.
Common Evaluation Methodology (CEM)	Common Evaluation Methodology for Information Technology Security Evaluation.
Distributed TOE	A TOE composed of multiple components operating as a logical whole.
Extended Package (EP)	A deprecated document form for collecting SFRs that implement a particular protocol, technology, or functionality. See Functional Packages.
Functional Package (FP)	A document that collects SFRs for a particular protocol, technology, or functionality.
Operational Environment (OE)	Hardware and software that are outside the TOE boundary that support the TOE functionality and security policy.
Protection Profile (PP)	An implementation-independent set of security requirements for a category of products.
Protection Profile Configuration (PP-Configuration)	A comprehensive set of security requirements for a product type that consists of at least one Base-PP and at least one PP-Module.
Protection Profile Module (PP-Module)	An implementation-independent statement of security needs for a TOE type complementary to one or more Base-PPs.
Security Assurance Requirement (SAR)	A requirement to assure the security of the TOE.
Security Functional Requirement (SFR)	A requirement for security enforcement by the TOE.
Security Target (ST)	A set of implementation-dependent security requirements for a specific product.
Target of Evaluation (TOE)	The product under evaluation.
TOE Security Functionality (TSF)	The security functionality of the product under evaluation.
TOE Summary Specification (TSS)	A description of how a TOE satisfies the SFRs in an ST.

1.2.2 Technical Terms

API Application Programming Interface	A specification of routines, data structures, object classes, and variables that allows an application to make use of services provided by another software component, such as a library. APIs are often provided for a set of libraries included with the platform.
Administrator	The person who is responsible for management activities, including setting the policy that is applied by the enterprise on the mobile device.
Critical Security Parameter	Security-related information whose disclosure or modification can compromise the security of a cryptographic module or authentication system.
Data	Program or application or data files that are stored or transmitted by a server or MD.
Data	A key used to encrypt data-at-rest.

Encryption Key	
Developer Modes	States in which additional services are available to a user in order to provide enhanced system access for debugging of software.
Enrolled State	The state in which a mobile device is managed by a policy from an MDM.
Enrollment over Secure Transport	Cryptographic protocol that describes an X.509 certificate management protocol targeting public key infrastructure (PKI) clients that need to acquire client certificates and associated certificate authority (CA) certificates.
Enterprise Applications	Applications that are provided and managed by the enterprise as opposed to a public application store.
Enterprise Data	Any data residing in enterprise servers or temporarily stored on mobile devices to which the mobile device user is allowed access according to the security policy defined by the enterprise and implemented by the administrator.
Key Encryption Key	A key that is used to encrypt other keys, such as DEKs or storage repositories that contain keys.
Locked State	Mobile device state where the device is powered on but most functionality is unavailable for use without authentication.
Mobile Device	A device which is composed of a hardware platform and its system software. The device typically provides wireless connectivity and may include software for functions like secure messaging, email, web, VPN connection, and VoIP (Voice over IP), for access to the protected enterprise network, enterprise data and applications, and for communicating to other MDs.
Mobile Device Management	Products that allow enterprises to apply security policies to MDs. This system consists of two primary components: the MDM server and the MDM agent.
Mobile Device User	The person who uses and is held responsible for an MD.
Operating System	Software which runs at the highest privilege level and can directly control hardware resources. Modern mobile devices typically have at least two primary operating systems: one which runs on the cellular baseband processor and one which runs on the application processor. The platform of the application processor handles most user interaction and provides the execution environment for apps. The platform of the cellular baseband processor handles communications with the cellular network and may control other peripherals. The term OS, without context, may be assumed to refer to the platform of the application processor.
Powered-Off State	Mobile device shutdown state.
Protected Data	All non-TSF data on the mobile device, including user or enterprise data. Protected data is encrypted while the mobile device is in the powered-off state. This includes keys in software-based storage. May overlap with sensitive data.
Root Encryption Key	A key tied to a particular device that is used to encrypt all other keys for that device.
Sensitive Data	Data that is encrypted by the mobile device. May include all user or enterprise data or may be data for specific applications such as emails, messaging, documents, calendar items, or contacts. May be protected while the mobile device is in the locked state. Must include at minimum some keys in software-based key storage.
Trust Anchor Database	A list of trusted root Certificate Authority certificates.
Unenrolled State	Mobile device state when it is not managed by an MDM.
Unlocked State	Mobile device state where it is powered on and its functionality is available for use.

1.3 Use Cases

This PP defines four use cases:

[USE CASE 1] Enterprise-owned device for general-purpose enterprise use

An enterprise-owned device for general-purpose business use is commonly called Corporate-Owned, Personally-Enabled (COPE). This use case entails a significant degree of enterprise control over configuration and software inventory. Enterprise administrators use an MDM product to establish policies on the mobile devices prior to user issuance. Users may use internet connectivity to browse the web, access corporate mail, or run enterprise applications, but this connectivity may be under significant control of the enterprise. The user may also be expected to store data and use applications for personal, non-enterprise use. The enterprise administrator uses the MDM product to deploy security policies and query mobile device status. The MDM may issue commands for remediation actions.

[USE CASE 2] Enterprise-owned device for specialized, high-security use

An enterprise-owned device with intentionally limited network connectivity, tightly controlled configuration, and limited software inventory is appropriate for specialized, high-security use cases. As in the previous use case, the MDM product is used to establish such policies on mobile devices prior to issuance to users. The device may not be permitted connectivity to any external peripherals. It may only be able to communicate via its Wi-Fi or cellular radios with the enterprise-run network, which may not even permit connectivity to the internet. Use of the device may require compliance with usage policies that are more restrictive than those in any general-purpose use case, yet may mitigate risks to highly sensitive information. Based on the operational environment and the acceptable risk level of the enterprise, those security functional requirements outlined in Section 5 of this PP along with the selections in the Use Case 2 template defined in Appendix G are sufficient for the high-security use case.

[USE CASE 3] Personally-owned device for personal and enterprise use

A personally-owned device which is used for both personal activities and enterprise data is commonly called Bring Your Own Device (BYOD). The device may be provisioned for access to enterprise resources after significant personal usage has occurred. Unlike in the enterprise-owned cases, the enterprise is limited in what security policies it can enforce because the user purchased the device primarily for personal use and is unlikely to accept policies that limit the functionality of the device. However, because the enterprise allows the user full (or nearly full) access to the enterprise network, the enterprise will require certain security policies, for example a password or screen lock policy and health reporting, such as the integrity of the mobile device system software, before allowing access. The administrator of the MDM can establish remediation actions, such as wipe of the enterprise data, for non-compliant devices. These controls could potentially be enforced by a separation mechanism built-in to the device itself to distinguish between enterprise and personal activities, or by a third-party application that provides access to enterprise resources and leverages security capabilities provided by the mobile device. Based on the operational environment and the acceptable risk level of the enterprise, those security functional requirements outlined in Section 5 of this PP along with the selections in the Use Case 3 template defined in Appendix G are sufficient for the secure implementation of this BYOD use case.

[USE CASE 4] Personally-owned device for personal and limited enterprise use

A personally-owned device may also be given access to limited enterprise services such as enterprise email. The enterprise may not need to enforce any security policies on this device because the user does not have full access to the enterprise or enterprise data. However, the enterprise may want secure email and web browsing with assurance that the services being provided to those clients by the mobile device are not compromised. Based on the operational environment and the acceptable risk level of the enterprise, those security functional requirements outlined in Section 5 of this PP are sufficient for the secure implementation of this BYOD use case.

2 Conformance Claims

Conformance Statement

An ST must claim exact conformance to this PP.

The evaluation methods used for evaluating the TOE are a combination of the workunits defined in [CEM] as well as the Evaluation Activities for ensuring that individual SFRs and SARs have a sufficient level of supporting evidence in the Security Target and guidance documentation and have been sufficiently tested by the laboratory as part of completing ATE_IND.1. Any functional packages this PP claims similarly contain their own Evaluation Activities that are used in this same manner.

CC Conformance Claims

This PP is conformant to Part 2 (extended) and Part 3 (conformant) of Common Criteria CC:2022, Revision 1.

PP Claim

This PP does not claim conformance to any Protection Profile.

The following PPs and PP-Modules are allowed to be specified in a PP-Configuration with this PP:

- Protection Profile for Mobile Device Fundamentals, Version 3.3
- Protection Profile for Application Software, Version 1.4
- PP-Module for MDM Agents, Version 1.0
- PP-Module for VPN Client, Version 2.5

Package Claim

- This PP is Functional Package for TLS, Version 1.1 conformant.
- This PP is Functional Package for SSH, Version 1.0 conformant.
- This PP is Functional Package for X.509, Version 1.0 conformant.
- This PP is Assurance Package for Flaw Remediation, Version 1.0 conformant.

The functional packages to which the PP conforms may include SFRs that are not mandatory to claim for the sake of conformance. An ST that claims one or more of these functional packages may include any non-mandatory SFRs that are appropriate to claim based on the capabilities of the TSF and on any triggers for their inclusion based inherently on the SFR selections made.

Evaluation Methods

This PP incorporates evaluation activies from the following Evaluation Methods documents:

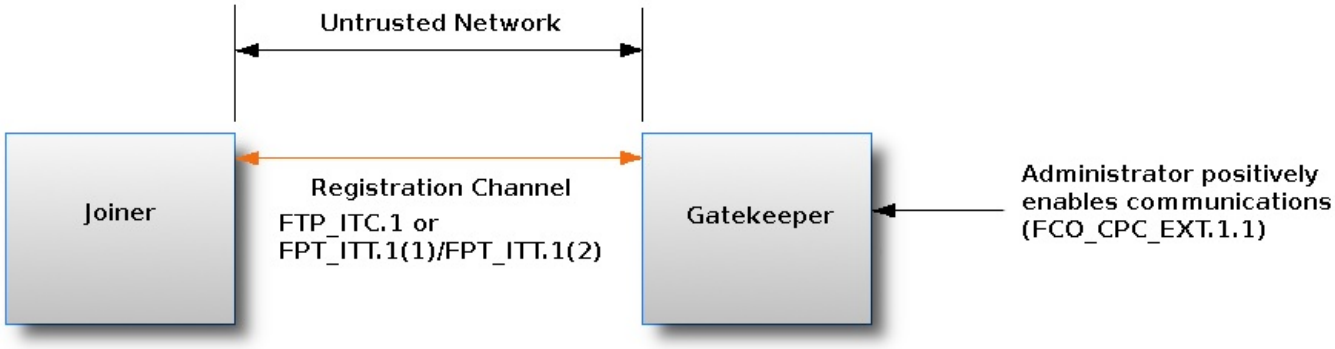
Additional Information

3 Introduction to Distributed TOEs

This PP includes support for distributed MDM TOEs. MDMs can sometimes be composed of multiple components operating as a logical whole. Frequently we see this architecture when dealing with products hosted in the cloud and offered as Software as a Service. There are a number of different architectures; but fundamentally, they are variations of the following model where the SFRs of this PP can only be fulfilled if the components are deployed and operate together. To be considered a distributed TOE, a minimum of two interconnected components are required.

3.1 Registration of Distributed TOE Components

When dealing with a distributed TOE, a number of separate components need to be brought together in the operational environment in order to create the TOE. This requires that trusted communications channels are set up between certain pairs of components (it is assumed that all components need to communicate with at least one other component, but not that all components need to communicate with all other components). The underlying model for creation of the TOE is to have a "registration process" in which components "join" the TOE. The registration process starts with two components, one of which (the "joiner") is about to join an existing TOE by registering with the other (the "gatekeeper"). The two components will use one or more specified authentication and communication channel options so that the components authenticate each other and protect any sensitive data that is transmitted during the registration process (e.g., a key might be sent by a gatekeeper to the joiner as a result of the registration). The following figures illustrate the three supported registration models. Figure 2 illustrates a distributed TOE registration approach which uses an instance of FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT or FTP_ITC.1/INTER_XFER_IT to protect the registration exchange.



1) Registration may be performed over any untrusted network
2) Registration performed over IPsec, TLS, SSH, or HTTPS channel
3) Choose FPT_ITT.1(1) if certificate revocation checking is not performed
4) Choose FPT_ITT.1(2) if registration is between the TSF and an MDM agent that is included in the TOE
5) Choose FTP_ITC.1 if certificate revocation checking is performed
6) Registration channel may be re-used for internal TSF communications

Figure 2: Distributed TOE registration using channel satisfying FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT or FTP_ITC.1/INTER_XFER_IT

The second approach (Figure 3) uses an alternative registration channel and supports use cases where the channel relies on environmental security constraints to provide the necessary protection of the registration exchange.

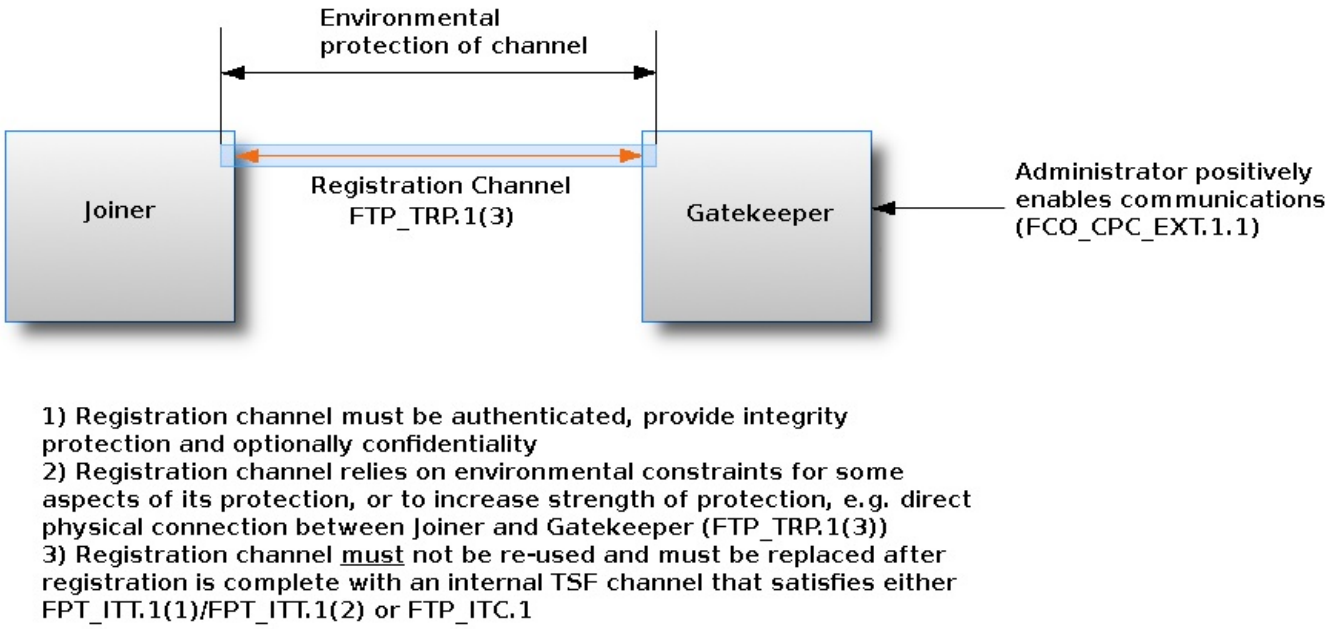


Figure 3: Distributed TOE registration using channel satisfying FTP_TRP.1/TRUSTPATH_JOIN

The final approach (Figure 4) supports use cases where registration is performed manually through direct configuration of both the Joiner and Gatekeeper devices. Once configured, the two components establish an internal TSF channel that satisfies FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT or FTP_ITC.1/INTER_XFER_IT.

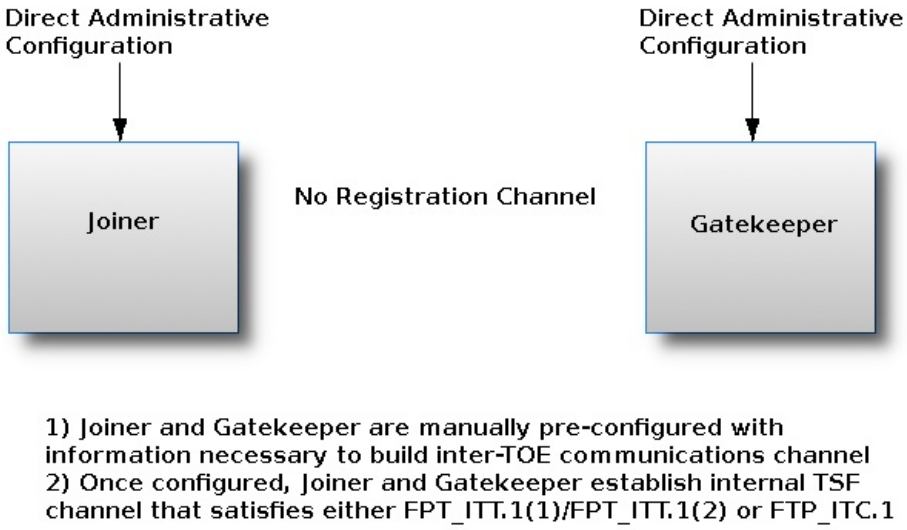


Figure 4: Distributed TOE registration without a registration channel

In each case, during the registration process, the administrator must positively enable the joining components before it can act as part of the TSF. Figure 5 illustrates the approaches that this enablement step may take;

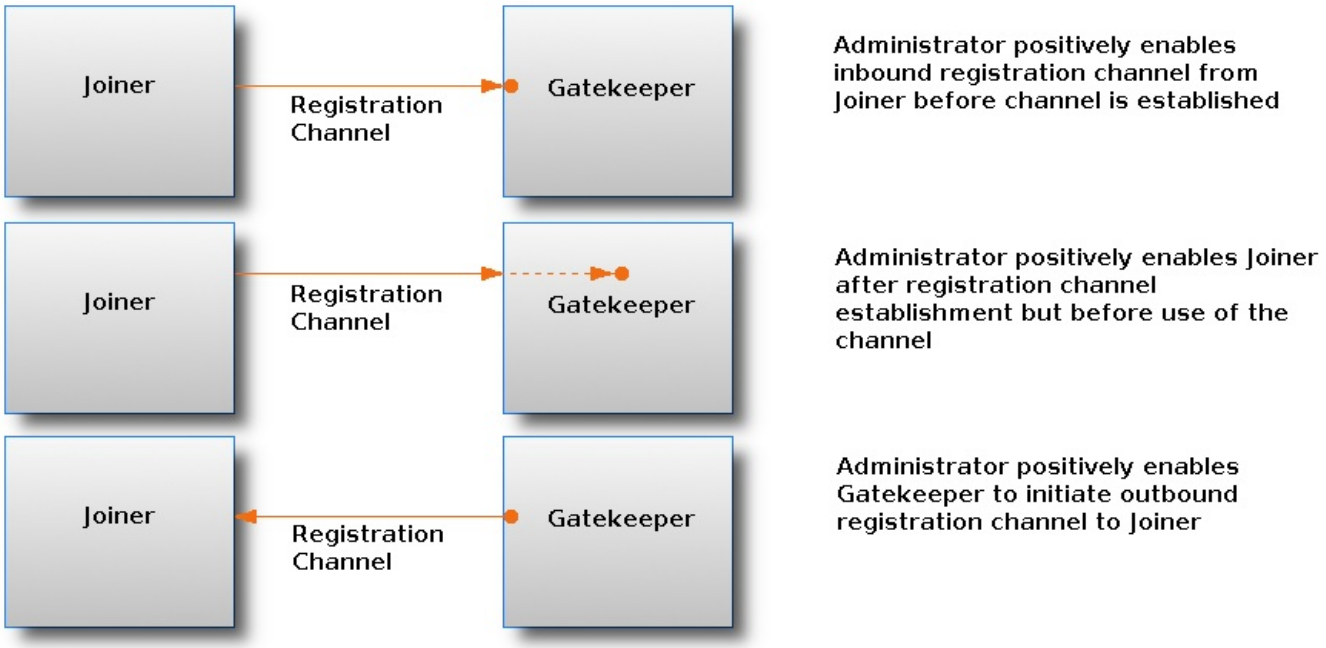


Figure 5: Joiner enablement options for Distributed TOEs

Note that in the case where no registration channel is required, that is the joiner and gatekeeper are directly configured (Figure 4), enablement is implied as part of this direct configuration process.

After registration, the components will communicate between themselves using a normal SSH, TLS, DTLS, IPsec, or HTTPS channel (which is specified in an ST as an instance of FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT or FTP_ITC.1/INTER_XFER_IT in terms of Section and Table t-audit-optional). This channel for inter-component communications is specified at the top level with the new (extended) SFR FCO_CPC_EXT.1 and is in addition to the other communication channels required for communication with entities outside the TOE (which are specified in an ST as instances of FTP_ITC.1/INTER_XFER_IT and FTP_TRP.1/TRUSTPATH_REM_ADMIN.

3.2 Allocation of Requirements in Distributed TOEs

For a distributed TOE, the security functional requirements in this PP need to be met by the TOE as a whole, but not all SFRs will necessarily be implemented by all components. The following categories are defined in order to specify when each SFR must be implemented by a component:

- **All Components ("All")** - All components that comprise the distributed TOE must independently satisfy the requirement.
- **At least one Component ("One")** - This requirement must be fulfilled by at least one component within the distributed TOE.
- **Feature Dependent ("Feature Dependent")** - These requirements will only be fulfilled where the feature is implemented by the distributed TOE component (note that the requirement to meet the PP as a whole requires that at least one component implements these requirements if they are specified in Section).

Table 1 specifies how each of the SFRs in this PP must be met, using the categories above.

Table 1: Security Functional Requirements for Distributed TOEs

Requirement	Description	Distributed TOE SFR Allocation
FAU_ALT_EXT.1	Server Alerts	One
FAU_CRP_EXT.1	Support for Compliance Reporting of Mobile Device Configuration	One
FAU_GEN.1/AUDITGEN	Audit Data Generation	All
FAU_GEN.1/MAS_SERVER	Audit Data Generation	Feature Dependent
FAU_NET_EXT.1	Network Reachability Review	One
FAU_SAR.1	Audit Review	Feature Dependent
FAU_SEL.1	Security Audit Event Selection	One
FAU_STG.1	External Trail Storage	All

FAU_STG.2	Audit Event Storage	Feature Dependent
FCO_CPC_EXT.1	Communication Partner Control	All
FCS_CKM.1	Cryptographic Key Generation	Feature Dependent
FCS_CKM.2	Cryptographic Key Establishment	All
FCS_CKM.6	Cryptographic Key Destruction	All
FCS_COP.1.1/CONF_ALG	Cryptographic Operation (Confidentiality Algorithms)	All
FCS_COP.1.1/HASH_ALG	Cryptographic Operation (Hashing Algorithms)	All
FCS_COP.1.1/KEY_HASH	Cryptographic Operation (Keyed-Hash Message Authentication)	All
FCS_COP.1.1/SIGN_ALG	Cryptographic Operation (Signature Algorithms)	All
FCS_HTTPS_EXT.1	HTTPS Protocol	Feature Dependent
FCS_IV_EXT.1	Initialization Vector Generation	Feature Dependent
FCS_RBG.1	Random Bit Generation (RBG)	All
FCS_RBG.2	Random Bit Generation (External Seeding)	Feature Dependent
FCS_RBG.3	Random Bit Generation (Internal Seeding - Single Source)	Feature Dependent
FCS_RBG.4	Random Bit Generation (Internal Seeding - Multiple Sources)	Feature Dependent
FCS_RBG.5	Random Bit Generation (Combining Noise Sources)	Feature Dependent
FCS_STG_EXT.1	Cryptographic Key Storage	All
FCS_STG_EXT.2	Encrypted Cryptographic Key Storage	Feature Dependent
FIA_CLI_EXT.1	Client Authorization	One
FIA_ENR_EXT.1	Enrollment of Mobile Device into Management	One
FIA_TOK_EXT.1	Client Tokens	One
FIA_UAU.1	Timing of Authentication	One
FIA_UAU.4	Single-Use Authentication Mechanisms	One
FIA_X509_EXT.1/CERTVAL_MAN	X.509 Certification Validation	Feature Dependent
FIA_X509_EXT.1/CERTVAL_SEL	X.509 Certification Validation	Feature Dependent
FIA_X509_EXT.2	X.509 Certificate Authentication	Feature Dependent
FIA_X509_EXT.3	X.509 Enrollment	Feature Dependent
FIA_X509_EXT.4	Alternate X.509 Enrollment	Feature Dependent
FMT_MOF.1/FUNCBE	Management of functions behaviour	Feature Dependent
FMT_MOF.1/MANAGEMENT_ENROLL	Management of functions behaviour (Enrollment)	Feature Dependent
FMT_MOF.1/MANAGEMENT_MAS	Management of Functions in (MAS Server Downloads)	Feature Dependent
FMT_POL_EXT.1	Trusted Policy Update	One
FMT_SAE_EXT.1	Security Attribute Expiration	One
FMT_SMF.1/MAS	Specification of Management Functions (MAS Server)	Feature Dependent
FMT_SMF.1/SERVER_CONF_AGENT	Specification of Management Functions	One

	(Server configuration of Agent)	
FMT_SMF.1/SERVER_CONF_SERVER	Specification of Management Functions (Server configuration of Server)	Feature Dependent
FMT_SMR.1/SECMAN_ROLES	Security Management Roles	One
FMT_SMR.1/SECMAN_ROLES_MAS	Security Management Roles	Feature Dependent
FPT_API_EXT.1	Use of Supported Services and APIs	All
FPT_FLS.1	Failure with Preservation of Secure State	All
FPT_ITT.1/INTER_XFER	Internal TOE TSF Data Transfer	Feature Dependent
FPT_ITT.1/INTER_XFER_AGENT	Internal TOE TSF Data Transfer (MDM Agent)	Feature Dependent
FPT_LIB_EXT.1	Use of Third-Party Libraries	All
FPT_TST.1	TSF Self-Testing	All
FPT_TST_EXT.1	Functionality Testing	All (except for agent components)
FPT_TUD_EXT.1	Trusted Update	All
FTA_TAB.1	Default TOE Access Banners	One
FTP_ITC.1/INTER_TSF_XFER_AGENT	Inter-TSF Trusted Channel (MDM Agent)	One
FTP_ITC.1/INTER_XFER_IT	Inter-TSF Trusted Channel (Authorized IT Entities)	One
FTP_ITC_EXT.1	Trusted Channel	One
FTP_TRP.1/TRUSTPATH_ENROLL	Trusted Path for Enrollment	Feature Dependent
FTP_TRP.1/TRUSTPATH_JOIN	Trusted Path for Joining	Feature Dependent
FTP_TRP.1/TRUSTPATH_REM_ADMIN	Trusted Path for Remote Administration	Feature Dependent

Only those SFRs included in the ST are required to be audited. The ST for a distributed TOE must include a mapping of SFRs to each of the components of the TOE. (Note that this deliverable is examined as part of the ASE_TSS.1 and AVA_VAN.1 Evaluation Activities.) The ST for a distributed TOE may also introduce a "minimum configuration" and identify components that may have instances added to an operational configuration without affecting the validity of the CC certification. Appendix E describes Evaluation Activities relating to these equivalency aspects of a distributed TOE (and hence what is expected in the ST).

3.3 Security Audit for Distributed TOEs

For distributed TOEs, the handling of audit information might be more complicated than for TOEs consisting only of one component. There are a few basic requirements to be fulfilled:

- Every component must be able to generate audit information.
- Every component must be able to buffer audit information and forward it to another TOE component or an external audit server. Optionally, each component may store audit information locally.
- For the overall TOE it must be possible to send out audit information to an external audit server.

In general, every component must be able to generate its own audit information. It would be possible that every component also stores its own audit information locally as well as every component could be able to send out audit data to an external audit server. It would also be sufficient that every component would be able to generate its own audit data and buffer it locally before the information is sent out to one or more other TOE components for local storage or transmission to an external audit server. For the transfer of audit records between TOE components the secure connection via FTP_ITC.1/INTER_XFER_IT or FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT must be used.

Such a solution would still be suitable to fulfill the requirement that all audit-related SFRs have to be fulfilled by all TOE components, although formally not every component would support local storage or transfer to an external audit server itself.

Regarding the establishment of inter-TOE communication, error conditions as well as successful connection and tear-down events should be captured by both ends of the connection.

All TOE components shall be able to generate its own audit data according to FAU_GEN.1 for all SFRs that it implements. For distributed TOEs, a mapping shall be provided to show which auditable events according to FAU_GEN.1 are covered by which components (also giving a justification that the records generated by each component cover all the SFRs that it implements). The overall TOE has to provide audit information about all events defined for FAU_GEN.1. As a result, at least one TOE component has to be assigned to every auditable event defined for FAU_GEN.1. The part of the mapping related to Table t-audit-mandatory shall be consistent with the mapping of SFRs to TOE components for ASE_TSS.1 in the sense that all components defined as generating audit information for a particular SFR should also contribute to that SFR in the mapping for ASE_TSS.1. This applies not only to audit events defined for mandatory SFRs but also to all audit events for optional, selection-based, and objective SFRs as defined in Table t-audit-optional , Table t-audit-objective , and Table t-audit-sel-based .

If one or more of the optional audit components FAU_STG.1 or FAU_STG.2 are selected in the ST derived from this PP, then the SFR mapping for ASE_TSS.1 must include a specific identification of the TOE components to which they apply.

4 Security Problem Definition

4.1 Threats

T.MALICIOUS_APPS

Malicious or flawed application threats exist because apps loaded onto amobile device may include malicious or exploitable code. An administrator of the MDM or mobile device usermay inadvertently import malicious code, or an attacker may insert malicious code into the TOE, resulting in thecompromise of TOE or TOE data.

Appendix A - Implementation-dependent Requirements

Implementation-dependent Requirements Appendix defines requirements that must be claimed in the ST if the TOE implements particular product features. For this technology type, the following product features require the claiming of additional SFRs:

Appendix B - Extended Component Definitions

This appendix contains the definitions for all extended requirements specified in the PP.

B.1 Extended Components Table

All extended components specified in the PP are listed in this table:

Table 2: Extended Component Definitions

Functional Class	Functional Components
Class: Communication (FCO)	FCO_CPC_EXT Component Registration Channel Definition
Class: Cryptographic Support (FCS)	FCS_HTTPS_EXT HTTPS Protocol FCS_IV_EXT Initialization Vector Generation FCS_STG_EXT Encrypted Cryptographic Key Storage
Class: Identification and Authentication (FIA)	FIA_CLI_EXT Client Authorization FIA_ENR_EXT Enrollment of Mobile Device into Management FIA_TOK_EXT Client Tokens
Class: Protection of the TSF (FPT)	FPT_API_EXT Use of Supported Services and APIs FPT_LIB_EXT Use of Third-Party Libraries FPT_TST_EXT Functionality Testing FPT_TUD_EXT Trusted Update
Class: Security Audit (FAU)	FAU_ALT_EXT Server Alerts FAU_CRP_EXT Support for Compliance Reporting of Mobile Device Configuration FAU_NET_EXT Network Reachability Review
Class: Security Management (FMT)	FMT_POL_EXT Trusted Policy Update FMT_SAE_EXT Security Attribute Expiration
Class: Trusted Path/Channels (FTP)	FTP_ITC_EXT Trusted Channel

B.2 Extended Component Definitions

B.2.1 Class: Communication (FCO)

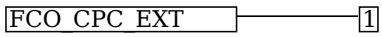
This PP defines the following extended components as part of the FCO class originally defined by CC Part 2:

B.2.1.1 FCO_CPC_EXT Component Registration Channel Definition

Family Behavior

This family describes the registration process, including the capability for the administrator to enable or disable communications between a distributed TOE and other components of the TOE.

Component Leveling



FCO_CPC_EXT.1, Component Registration Channel Definition, defines requirements for the registration process for distributed TOEs.

Management: FCO_CPC_EXT.1

There are no management activities foreseen.

Audit: FCO_CPC_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Enabling or disabling communications between a pair of components.Identities of the endpoint's pairs enabled or disabled.

FCO_CPC_EXT.1 Component Registration Channel Definition

Hierarchical to: No other components.

Dependencies to: FPT_ITT.1 TSF Data TransferFTP_TRP.1 Trusted Path

FCO_CPC_EXT.1.1

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to require an Administrator to enable communications between any pair of TOE components before such communication can take place.

FCO_CPC_EXT.1.2

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to implement a registration process in which components establish and use a communications channel that uses[**selection**:

- *A channel that meets the secure channel requirements in[**selection**: FTP_ITC.1, FPT_ITT.1/INTER_XFER, FPT_ITT.1/INTER_XFER_AGENT]*
- *A channel that meets the secure registration channel requirements in[**selection**: FTP_TRP.1/TRUSTPATH_ENROLL, FTP_TRP.1/TRUSTPATH_JOIN]*
- *No channel*

]for at least TSF data.

FCO_CPC_EXT.1.3

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to enable an administrator to disable communications between any pair of TOE components.

B.2.2 Class: Cryptographic Support (FCS)

This PP defines the following extended components as part of the FCS class originally defined by CC Part 2:

B.2.2.1 FCS_HTTPS_EXT HTTPS Protocol

Family Behavior

This family defines requirements for protecting HTTP communications between the TOE and an external IT entity.

Component Leveling

FCS_HTTPS_EXT.1, HTTPS Protocol, defines requirements for the implementation of the HTTPS protocol.

Management: FCS_HTTPS_EXT.1

There are no management activities foreseen.

Audit: FCS_HTTPS_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Failure of the certificate validity check.Issuer Name and Subject Name of certificate.User's authorization decisionNo additional information

FCS_HTTPS_EXT.1 HTTPS Protocol

Hierarchical to:	No other components.
Dependencies to:	FCS_TLS_EXT.1 TLS Protocol[FCS_TLSC_EXT.1 TLS Client Protocol or FCS_TLSS_EXT.1 TLS Server Protocol

FCS_HTTPS_EXT.1.1

The TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2

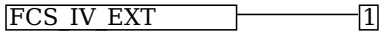
The TSF shall implement HTTPS using TLS in accordance with the Package for Transport Layer Security.

B.2.2.2 FCS_IV_EXT Initialization Vector Generation

Family Behavior

This family defines requirements for generating IVs in accordance with NIST-approved cipher modes.

Component Leveling



FCS_IV_EXT.1, Initialization Vector Generation, defines requirements for generating IVs.

Management: FCS_IV_EXT.1

There are no management activities foreseen.

Audit: FCS_IV_EXT.1

There are no auditable events foreseen.

FCS_IV_EXT.1 Initialization Vector Generation

Hierarchical to: No other components.

Dependencies to: No dependencies.

FCS_IV_EXT.1.1

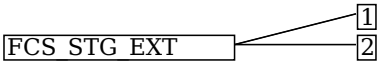
The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to generate IVs in accordance withTable iv .

B.2.2.3 FCS_STG_EXT Encrypted Cryptographic Key Storage

Family Behavior

This family defines requirements for ensuring the protection of keys and secrets.

Component Leveling



FCS_STG_EXT.1, Cryptographic Key Storage, defines requirements for the security of persistent secrets and private keys.

FCS_STG_EXT.2, Encrypted Cryptographic Key Storage, defines requirements for preventing access to private keys and persistent secrets.

Management: FCS_STG_EXT.1

The following actions could be considered for the management functions in FMT. Import keys or secrets into the secure key storage (MDF Function 9)

Audit: FCS_STG_EXT.1

There are no auditable events foreseen.

FCS_STG_EXT.1 Cryptographic Key Storage

Hierarchical to: No other components.

Dependencies to: No dependencies.

FCS_STG_EXT.1.1

The TSF shall use[**selection**: *platform-provided key storage, encryption as specified in FCS_STG_EXT.2*]for all persistent secrets and private keys.

Management: FCS_STG_EXT.2

There are no management activities foreseen.

Audit: FCS_STG_EXT.2

There are no auditable events foreseen.

FCS_STG_EXT.2 Encrypted Cryptographic Key Storage

Hierarchical to: No other components.

Dependencies to: No dependencies.

FCS_STG_EXT.2.1

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to encrypt all keys using AES in the[**selection**: *Key Wrap (KW) mode, Key Wrap with Padding (KWP) mode, GCM, CCM, CBC mode*].

B.2.3 Class: Identification and Authentication (FIA)

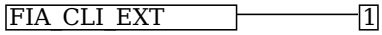
This PP defines the following extended components as part of the FIA class originally defined by CC Part 2:

B.2.3.1 FIA_CLI_EXT Client Authorization

Family Behavior

This family defines requirements for unique certificate or token use.

Component Leveling



FIA_CLI_EXT.1, Client Authorization, defines requirements for a unique certificate or token for each client device.

Management: FIA_CLI_EXT.1

There are no management activities foreseen.

Audit: FIA_CLI_EXT.1

There are no auditable events foreseen.

FIA_CLI_EXT.1 Client Authorization

Hierarchical to: No other components.

Dependencies to: No dependencies.

FIA_CLI_EXT.1.1

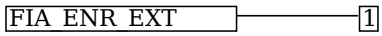
The TSF shall require a unique[**selection**: *certificate*, *token as defined in FIA_TOK_EXT.1*]for each client device.

B.2.3.2 FIA_ENR_EXT Enrollment of Mobile Device into Management

Family Behavior

This family defines requirements for authenticating remote users and limiting user enrollment.

Component Leveling



FIA_ENR_EXT.1, Enrollment of Mobile Device into Management, defines requirements for authenticating and limiting user actions.

Management: FIA_ENR_EXT.1

The following actions could be considered for the management functions in FMT. Configure the specific device models.Configure the specific time period.

Audit: FIA_ENR_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Failure of MD user authentication.Presented username.

FIA_ENR_EXT.1 Enrollment of Mobile Device into Management

Hierarchical to: No other components.

Dependencies to: FIA_UAU.4 Single-Use Authentication MechanismsFMT_SMF.1 Specification of Management Functions

FIA_ENR_EXT.1.1

The TSF shall authenticate the remote users over a trusted channel during the enrollment of a mobile device.

FIA_ENR_EXT.1.2

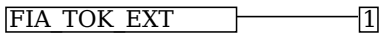
The TSF shall limit the user's enrollment of devices to devices specified by[**selection**: *IMEI*, *[**assignment**: a unique device ID]*]and[**selection**: *specific device models*, *a number of devices*, *specific time period*, *[**assignment**: other features]*, *no other features*].

B.2.3.3 FIA_TOK_EXT Client Tokens

Family Behavior

This family defines requirements for using a unique device to generate unique tokens for client devices.

Component Leveling



FIA_TOK_EXT.1, Client Tokens, defines requirements for generating unique tokens.

Management: FIA_TOK_EXT.1

There are no management activities foreseen.

Audit: FIA_TOK_EXT.1

There are no auditable events foreseen.

FIA_TOK_EXT.1 Client Tokens

Hierarchical to: No other components.

Dependencies to: No dependencies.

FIA_TOK_EXT.1.1

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to use[**selection**: *IMEI, [**assignment**: other unique device ID]*]to generate a unique token for each client device.

B.2.4 Class: Protection of the TSF (FPT)

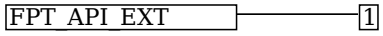
This PP defines the following extended components as part of the FPT class originally defined by CC Part 2:

B.2.4.1 FPT_API_EXT Use of Supported Services and APIs

Family Behavior

This family describes document platform APIs when selecting "invoke platform-provided functionality."

Component Leveling



FPT_API_EXT.1, Use of Supported Services and APIs, defines requirements for API usage.

Management: FPT_API_EXT.1

There are no management activities foreseen.

Audit: FPT_API_EXT.1

There are no auditable events foreseen.

FPT_API_EXT.1 Use of Supported Services and APIs

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_API_EXT.1.1

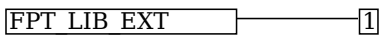
The TSF shall use only documented platform APIs.

B.2.4.2 FPT_LIB_EXT Use of Third-Party Libraries

Family Behavior

This family describes packaging third-party libraries when selecting "implement functionality."

Component Leveling



FPT_LIB_EXT.1, Use of Third-Party Libraries, defines requirements for third-party libraries.

Management: FPT_LIB_EXT.1

There are no management activities foreseen.

Audit: FPT_LIB_EXT.1

There are no auditable events foreseen.

FPT_LIB_EXT.1 Use of Third-Party Libraries

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_LIB_EXT.1.1

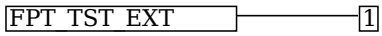
The MDM software shall be packaged with only[**assignment**: *list of third-party libraries*].

B.2.4.3 FPT_TST_EXT Functionality Testing

Family Behavior

This family defines requirements for running self-tests and verifying integrity or executable code.

Component Leveling



FPT_TST_EXT.1, Functionality Testing, defines requirements for the integrity of self-testing.

Management: FPT_TST_EXT.1

There are no management activities foreseen.

Audit: FPT_TST_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Initiation of self-test.Failure of self-test.Detected integrity violation

FPT_TST_EXT.1 Functionality Testing

Hierarchical to: No other components.

Dependencies to: FPT_TST.1 TSF Self-Testing

FPT_TST_EXT.1.1

The TSF shall run a suite of self tests during initial start-up (power on) to demonstrate correct operation of the TSF.

FPT_TST_EXT.1.2

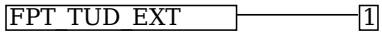
The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the[**selection**: *TSF*, *TOE platform*]-provided cryptographic services.

B.2.4.4 FPT_TUD_EXT Trusted Update

Family Behavior

This family defines requirements for allowing authorized administrators to query software versions, initiate updates, and verify software updates prior to installation.

Component Leveling



FPT_TUD_EXT.1, Trusted Update, defines requirements for authorized administrators to manage software versions and updates.

Management: FPT_TUD_EXT.1

The following actions could be considered for the management functions in FMT. Query the current version of the MD firmware or software.Update system software (MDF Function 15).

Audit: FPT_TUD_EXT.1

The following action should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Success or failure of signature verification

FPT_TUD_EXT.1 Trusted Update

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_TUD_EXT.1.1

The TSF shall provide authorized administrators the ability to query the current version of the software.

FPT_TUD_EXT.1.2

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to provide authorized administrators the ability to initiate updates to TSF software.

FPT_TUD_EXT.1.3

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to provide a means to verify software updates to the TSF using a digital signature mechanism prior to installing those updates.

B.2.5 Class: Security Audit (FAU)

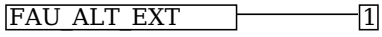
This PP defines the following extended components as part of the FAU class originally defined by CC Part 2:

B.2.5.1 FAU_ALT_EXT Server Alerts

Family Behavior

This family defines requirements for the TSF to alert administrators when a set of specified events occurs.

Component Leveling



FAU_ALT_EXT.1, Server Alerts, defines requirements for alerting the administrator to events.

Management: FAU_ALT_EXT.1

The following actions could be considered for the management functions in FMT. Install policies.

Audit: FAU_ALT_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is include in the PP or ST. Type of alert.Identity of Mobile Device that sent alert.

FAU_ALT_EXT.1 Server Alerts

Hierarchical to: No other components.

Dependencies to: No dependencies.

FAU_ALT_EXT.1.1

The TSF shall alert the administrators in the event of any of the following:

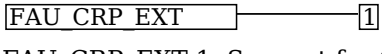
- Change in enrollment status
- Failure to apply policies to a mobile device
- [**selection**: ***[assignment**: other events], no other events*]

B.2.5.2 FAU_CRP_EXT Support for Compliance Reporting of Mobile Device Configuration

Family Behavior

This family defines requirements for the TSF to provide an interface for the MDM server to convey information about mobile devices for other systems.

Component Leveling



FAU_CRP_EXT.1, Support for Compliance Reporting of Mobile Device Configuration, defines requirements for providing information to enrolled mobile devices through a secure channel.

Management: FAU_CRP_EXT.1

The following actions could be considered for the management functions in FMT. Query the current version of the MD firmware or software.Query the current version of the hardware model of the device.Query the current version of installed mobile applications.

Audit: FAU_CRP_EXT.1

There are no auditable events foreseen.

FAU_CRP_EXT.1 Support for Compliance Reporting of Mobile Device Configuration

Hierarchical to: No other components.

Dependencies to: FTP_ITC.1 Inter-TSF Trusted Channel

FAU_CRP_EXT.1.1

The TSF shall provide[**selection**: *an interface that provides responses to queries about the configuration of enrolled devices, an interface that permits the export of data about the configuration of enrolled devices*]to authorized entities over a channel that meets the secure channel requirements in FTP_ITC.1/INTER_XFER_IT. The provided information for each enrolled mobile device includes:

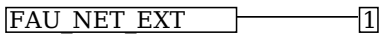
- The current version of the MD firmware or software
- The current version of the hardware model of the device
- The current version of installed mobile applications
- List of MD configuration policies that are in place on the device (as defined in FMT_SMF.1.1/SERVER_CONF_AGENT)
- [**selection**: *[**assignment**: list of other available information about enrolled devices], no other information*].

B.2.5.3 FAU_NET_EXT Network Reachability Review

Family Behavior

This family defines requirements for administrators to see network connectivity status.

Component Leveling



FAU_NET_EXT.1, Network Reachability Review, defines requirements for authorized administrators to read network connectivity status.

Management: FAU_NET_EXT.1

The following actions could be considered for the management functions in FMT. Query connectivity status.

Audit: FAU_NET_EXT.1

There are no auditable events foreseen.

FAU_NET_EXT.1 Network Reachability Review

Hierarchical to: No other components.

Dependencies to: FAU_ALT_EXT.2 Agent Alerts

FAU_NET_EXT.1.1

The TSF shall provide authorized administrators with the capability to read the network connectivity status of an enrolled agent.

B.2.6 Class: Security Management (FMT)

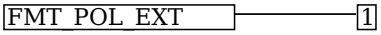
This PP defines the following extended components as part of the FMT class originally defined by CC Part 2:

B.2.6.1 FMT_POL_EXT Trusted Policy Update

Family Behavior

This family describes how to use digitally signed policies and updates by using private keys, and validating the policy is appropriate.

Component Leveling



FMT_POL_EXT.1, Trusted Policy Update, defines requirements for using digitally signed policies and policy updates.

Management: FMT_POL_EXT.1

There are no management activities foreseen.

Audit: FMT_POL_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Attempt to reuse enrollment data.Enrollment data.

FMT_POL_EXT.1 Trusted Policy Update

Hierarchical to: No other components.

Dependencies to: No dependencies.

FMT_POL_EXT.1.1

The TSF shall provide digitally signed policies and policy updates to the MDM agent.

FMT_POL_EXT.1.2

The TSF shall sign policies and policy updates using a private key associated with[**selection**: *an X509 certificate, a public key provisioned to the agent*]trusted by the agent for policy verification.

FMT_POL_EXT.1.3

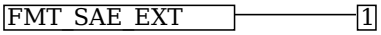
For each unique policy managed by the TSF, the TSF shall validate that the policy is appropriate for an agent using[**selection**: *client authentication via an X509 certificate representing the agent, a token issued to the agent and associated with a policy signing key uniquely associated to the policy*].

B.2.6.2 FMT_SAE_EXT Security Attribute Expiration

Family Behavior

This family defines the requirements for using expiration time for enrollment and denying enrollment if that time has passed.

Component Leveling



FMT_SAE_EXT.1, Security Attribute Expiration, defines requirements for the expiration time for enrollment authentication.

Management: FMT_SAE_EXT.1

The following action could be considered for the management functions in FMT. Configure the length of time the enrollment authenticator is valid.

Audit: FMT_SAE_EXT.1

The following actions should be auditable if FAU_GEN security audit data generation is included in the PP or ST. Enrollment attempted after expiration of authentication data.Identity of user.

FMT_SAE_EXT.1 Security Attribute Expiration

Hierarchical to: No other components.

Dependencies to: FIA_ENR_EXT.1 Enrollment of Mobile Device into ManagementFIA_UAU.4 Single-Use Authentication Mechanisms

FMT_SAE_EXT.1.1

The TSF shall be able to specify a configurable expiration time for enrollment authentication data.

FMT_SAE_EXT.1.2

The TSF shall be able to deny enrollment after the expiration time for the enrollment authentication data has passed.

B.2.7 Class: Trusted Path/Channels (FTP)

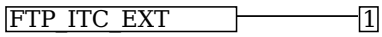
This PP defines the following extended components as part of the FTP class originally defined by CC Part 2:

B.2.7.1 FTP_ITC_EXT Trusted Channel

Family Behavior

The family defines requirements for communication channels between itself and other communication channels.

Component Leveling



FTP_ITC_EXT.1, Trusted Channel, defines requirements for providing logically distinct communication channels.

Management: FTP_ITC_EXT.1

There are no management activities foreseen.

Audit: FTP_ITC_EXT.1

There are no auditable events foreseen.

FTP_ITC_EXT.1 Trusted Channel

Hierarchical to: No other components.

Dependencies to: FPT_ITC.1 Inter-TSF Trusted ChannelFTP_TRP.1 Trusted Path

FTP_ITC_EXT.1.1

The TSF shall provide a communication channel between itself and[**selection**:

- *an MDM agent that is internal to the TOE*
- *an MDM agent that is external to the TOE*
- *other components comprising the distributed TOE*

]that is logically distinct from other communication channels, as specified in[**selection**: *FPT_ITT.1/INTER_XFER*, *FPT_ITT.1/INTER_XFER_AGENT*, *FTP_ITC.1/INTER_TSF_XFER_AGENT*].

Appendix C - Acronyms

Table 3: Acronyms

Acronym	Meaning
Base-PP	Base Protection Profile
CC	Common Criteria
CEM	Common Evaluation Methodology
cPP	Collaborative Protection Profile
EP	Extended Package
FP	Functional Package
OE	Operational Environment
PP	Protection Profile
PP-Configuration	Protection Profile Configuration
PP-Module	Protection Profile Module
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification

Appendix D - Bibliography

Table 4: Bibliography

Identifier	Title
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and general model, CCMB-2022-11-001, CC:2022, Revision 1, November 2022. • Part 2: Security functional requirements, CCMB-2022-11-002, CC:2022, Revision 1, November 2022. • Part 3: Security assurance requirements, CCMB-2022-11-003, CC:2022, Revision 1, November 2022. • Part 4: Framework for the specification of evaluation methods and activities, CCMB-2022-11-004, CC:2022, Revision 1, November 2022. • Part 5: Pre-defined packages of security requirements, CCMB-2022-11-005, CC:2022, Revision 1, November 2022.
[CEM]	Common Methodology for Information Technology Security Evaluation - • Evaluation methodology, CCMB-2022-11-006, CC:2022, Revision 1, November 2022.
[APP PP]	Protection Profile for Application Software, Version 1.4, 2021-10-07
[CSA]	Computer Security Act of 1987, H.R. 145, June 11, 1987.
[MDF PP]	Protection Profile for Mobile Device Fundamentals, Version 3.3, 2022-09-12
[MOD MDMA]	PP-Module for MDM agents, Version 1.0, 2019-04-25
[MOD VPNC]	PP-Module for VPN Client, Version 2.5, 2024-06-24
[OMB]	Reporting Incidents Involving Personally Identifiable Information and Incorporating the Cost for Security in Agency Information Technology Investments, OMB M-06-19, July 12, 2006.