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 Assurance Package for Flaw Remediation, Version 1.0 conformant.
- 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 does not conform to any assurance packages.

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:

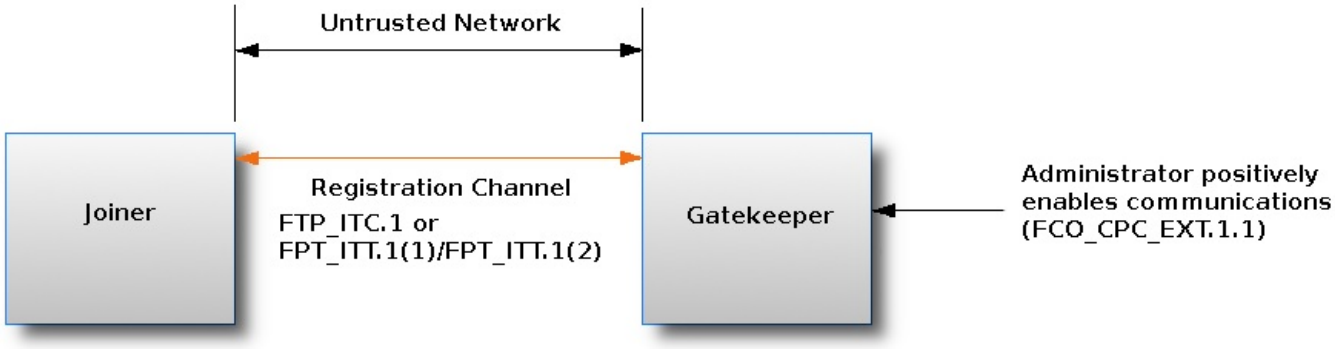
Additional Information

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.

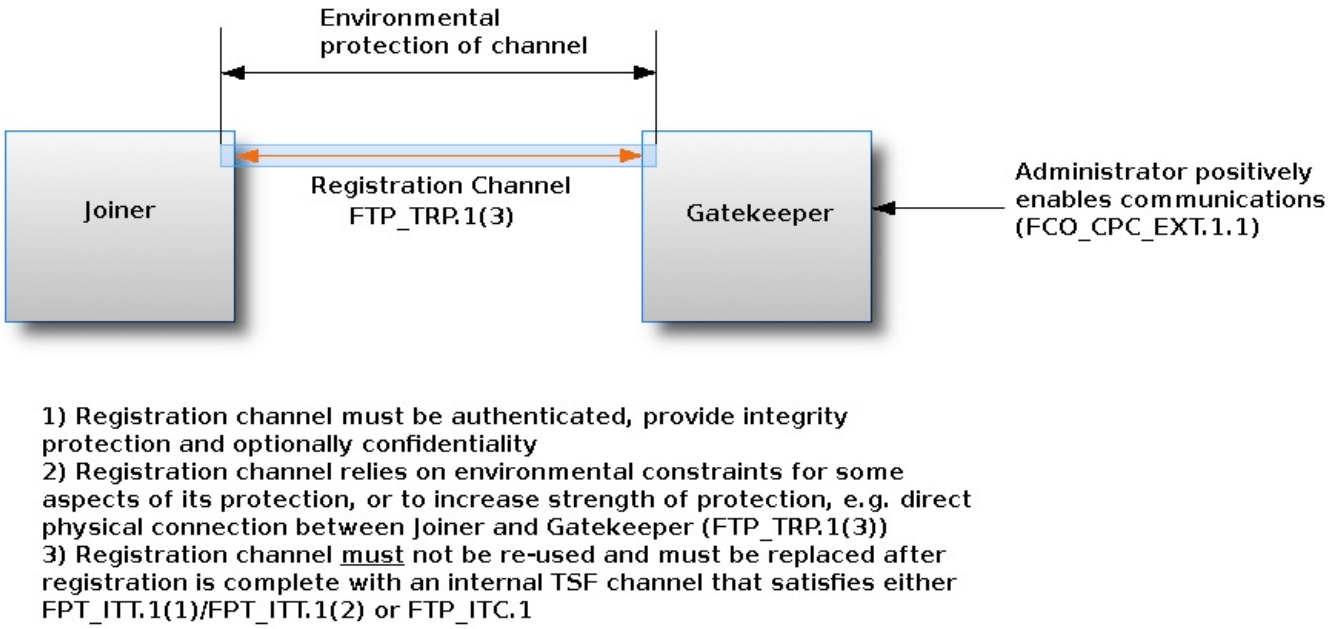


Figure 3: Distributed TOE registration using channel satisfying FTP_TRP.1/TRUSTPATH_JOIN

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.

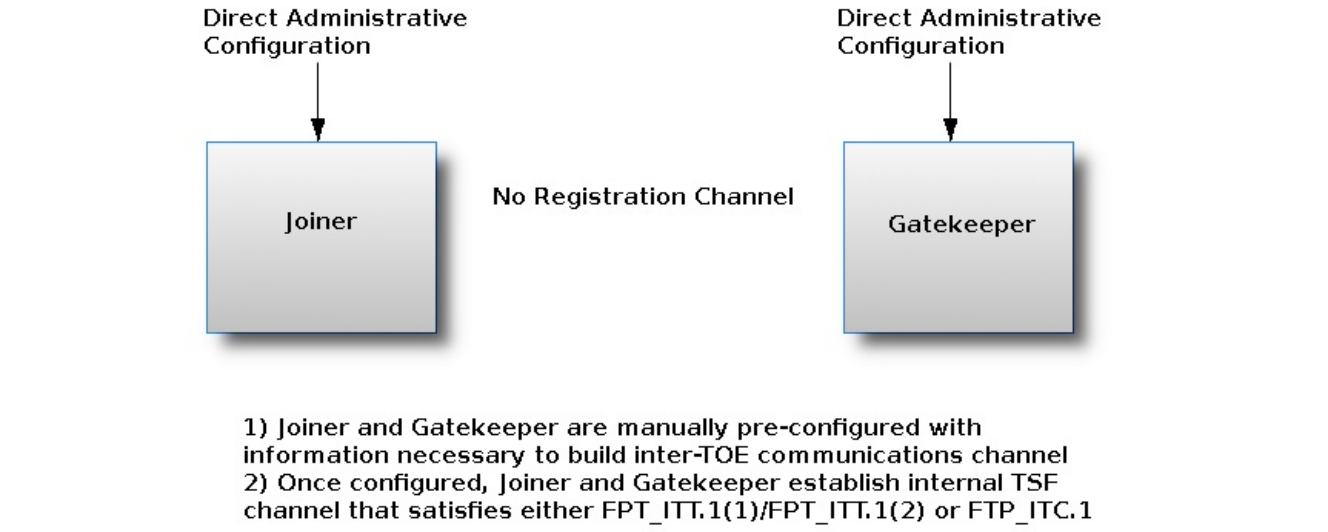


Figure 4: Distributed TOE registration without a registration channel

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;

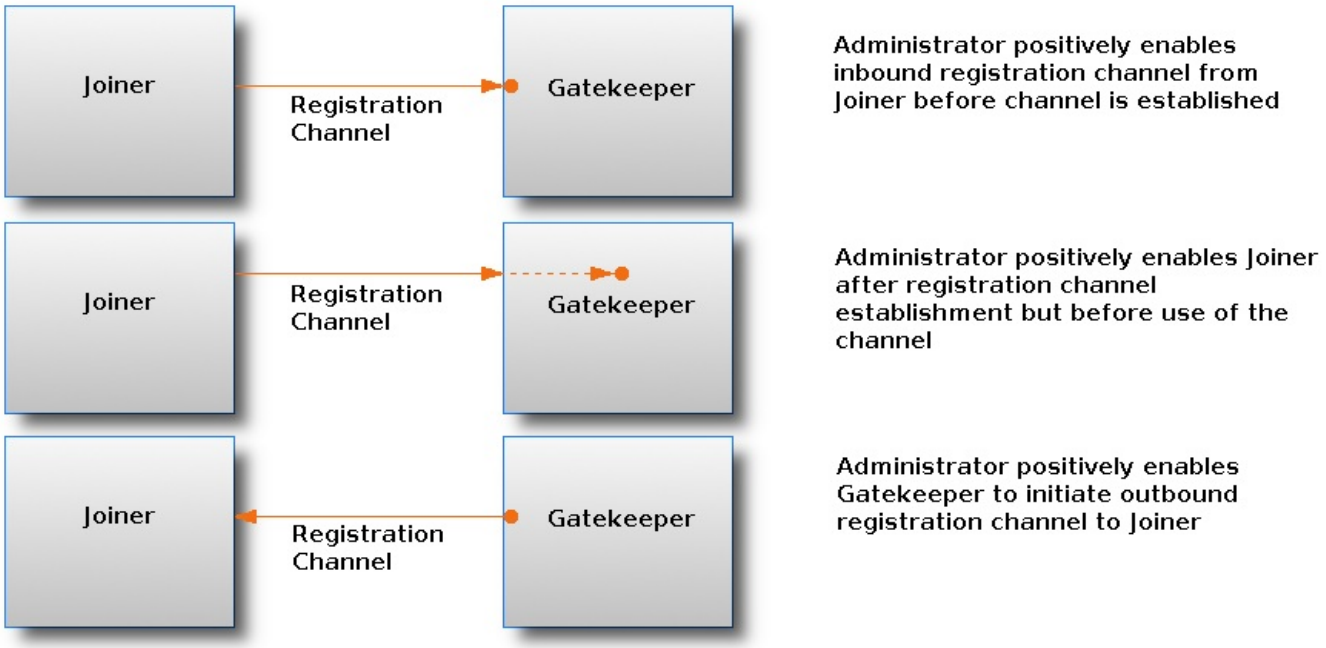


Figure 5: Joiner enablement options for Distributed TOEs

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 2). 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 2 , Table 3 , and Table 4 .

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 - Optional Requirements

As indicated in the introduction to this PP, the baseline requirements (those that must be performed by the TOE) are contained in the body of this PP. This appendix contains three other types of optional requirements:

The first type, defined in Appendix A.1 Strictly Optional Requirements, are strictly optional requirements. If the TOE meets any of these requirements the vendor is encouraged to claim the associated SFRs in the ST, but doing so is not required in order to conform to this PP.

The second type, defined in Appendix A.2 Objective Requirements, are objective requirements. These describe security functionality that is not yet widely available in commercial technology. Objective requirements are not currently mandated by this PP, but will be mandated in the future. Adoption by vendors is encouraged, but claiming these SFRs is not required in order to conform to this PP.

The third type, defined in Appendix A.3 Implementation-dependent Requirements, are Implementation-dependent requirements. If the TOE implements the product features associated with the listed SFRs, either the SFRs must be claimed or the product features must be disabled in the evaluated configuration.

A.1 Strictly Optional Requirements

A.1.1 Auditable Events for Strictly Optional Requirements

Table 2: Auditable Events for Strictly Optional Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FAU_SAR.1	No events specified	N/A
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	No additional information
FTA_TAB.1	Change in banner setting	No additional information

A.1.2 Class: Security Audit (FAU)

FAU_SAR.1 Audit Review

FAU_SAR.1.1

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** provide [*authorized administrators*] with the capability to read [*all audit data*] from the audit data.

FAU_SAR.1.2

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** provide the audit data in a manner suitable for the **authorized administrators** to interpret the information.

Application Note: The intent of this requirement is to ensure that the administrator can view and interpret the audit data and to prevent unauthorized users from accessing the logs.

Evaluation Activities

FAU_SAR.1
TSS
If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
The evaluator shall check the operational guidance and ensure that it describes how the administrator accesses the audit data and describes the format of audit data.

Tests
The evaluator shall attempt to view the audit data as the authorized administrator and verify that the action succeeds. The evaluator shall ensure the audit data generated during testing match the format specified in the administrative guide.

FAU_SEL.1 Security Audit Event Selection

FAU_SEL.1.1

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]to select the set of events to be audited from the set of all auditable events based on the following attributes:

1. [*event type*]
2. [*success of auditable security events*
3. *failure of auditable security events*
4. *[**assignment**: other attributes]*]

Application Note: The intent of this requirement is to identify all criteria that can be selected to trigger an audit event. The ST author must select whether the TSF or the platform maintains the audit data. For the ST author, the assignment is used to list any additional criteria or "none."

Evaluation Activities

FAU_SEL.1
TSS
If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
The evaluator shall review the administrative guidance to ensure that the guidance itemizes all event types, as well as describes all attributes that are to be selectable in accordance with the requirement, to include those attributes listed in the assignment. The administrative guidance shall also contain instructions on how to set the pre-selection as well as explain the syntax (if present) for multi-value pre-selection. The administrative guidance shall also identify those audit data that is always recorded, regardless of the selection criteria currently being enforced.

Tests
The evaluator shall also perform the following tests:
- *Test FAU_SEL.1:1: For each attribute listed in the requirement, the evaluator shall devise a test to show that selecting the attribute causes only audit events with that attribute (or those that are always recorded, as identified in the administrative guidance) to be recorded.*
- *Test FAU_SEL.1:2: [conditional] If the TSF supports specification of more complex audit pre-selection criteria (e.g., multiple attributes, logical expressions using attributes) then the evaluator shall devise tests showing that this capability is correctly implemented. The evaluator shall also, in the test plan, provide a short narrative justifying the set of tests as representative and sufficient to exercise the capability.*

A.1.3 Class: TOE Access (FTA)

FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1

Before establishing a user session, the [*TSF*] shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** display an [*administrator-specified advisory notice and consent warning*] message **regarding use of the TOE.**

Application Note: This requirement is to ensure that an advisory notice or consent banner is presented to the user on start-up or unlock of the TSF.

Evaluation Activities

FTA_TAB.1
TSS
If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If "implement functionality" is selected, the TSS shall describe when the banner is displayed.

Guidance
The evaluator follows the operational guidance to configure a notice and consent warning message.

Tests
The evaluator shall also perform the following test: The evaluator shall start up or unlock the TSF. The evaluator shall verify that the notice and consent warning message is displayed in each instance described in the TSS.

A.2 Objective Requirements

A.2.1 Auditable Events for Objective Requirements

Table 3: Auditable Events for Objective Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FAU_CRP_EXT.1	No events specified	N/A
FCO_CPC_EXT.1	Enabling or disabling communications between a pair of components.	Identities of the endpoint's pairs enabled or disabled.
FIA_UAU.4	Attempt to reuse enrollment data	Enrollment data
FIA_X509_EXT.3	Generation of Certificate Request Message	Content of Certificate Request Message
	Success or failure of verification	Issuer and Subject name of added certificate or reason for failure
FIA_X509_EXT.4	Update of EST Trust Anchor Database	Subject name of added Root CA
	Generation of Certificate Enrollment Request	• Issuer and Subject name of EST Server • Method of authentication • Issuer and Subject name of certificate used to authenticate • Content of Certificate Request Message
	Success or failure of enrollment	Issuer and Subject name of added certificate or reason for failure
FMT_SAE_EXT.1	Enrollment attempted after expiration of authentication data	Identity of user
FTP_TRP.1/TRUSTPATH_JOIN	Initiation and termination of the trusted channel	Trusted channel protocol

A.2.2 Class: Security Audit (FAU)

FAU_CRP_EXT.1 Support for Compliance Reporting of Mobile Device Configuration

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].

Application Note: The intent of this requirement is that the MDM server be able to provide compliance information about enrolled mobile devices for use by other enterprise security infrastructure systems. There are active standards efforts underway by the Internet Engineering Task Force (IETF) Security Automation and Continuous Monitoring (SACM) Working Group and others to define protocols and standards to assess and report on endpoint device posture. We expect that this requirement will evolve in future versions of this Protection Profile as standards efforts mature.

Evaluation Activities

FAU_CRP_EXT.1
TSS
TBD

Guidance
The evaluator shall check to ensure that the operational guidance contains instructions on how to access the MDM server's compliance reporting interface.

Tests
- *Test FAU_CRP_EXT.1:1: Using the operational guidance, the evaluator shall demonstrate the ability to access the compliance reporting interface from an authorized entity and successfully obtain information about enrolled devices.*
- *Test FAU_CRP_EXT.1:2: The evaluator shall attempt to access the compliance reporting interface from an unauthorized entity and demonstrate that the attempt is denied.*

A.2.3 Class: Communication (FCO)

FCO_CPC_EXT.1 Component Registration Channel Definition

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.

Application Note: This SFR is only applicable if the TOE is distributed and therefore has multiple components that need to communicate via an internal TSF channel. When creating the TSF from the initial pair of components, either of these components may be identified as the TSF for the purposes of satisfying the meaning of "TSF" in this SFR.

The intention of this requirement is to ensure that there is a registration process that includes a positive enablement step by an administrator before components joining a distributed TOE can communicate with the other components of the TOE and before the new component can act as part of the TSF. The registration process may itself involve communication with the joining component: many implementations use a bespoke process for this, and the security requirements for the "registration communication" are then defined in FCO_CPC_EXT.1.2 . Use of this "registration communication" channel is not deemed inconsistent with the requirement of FCO_CPC_EXT.1.1 (i.e., the registration channel can be used before the enablement step, but only in order to complete the registration

process).

The channel selection (for the registration channel) in FCO_CPC_EXT.1.2 is essentially a choice between the use of a normal secure channel that is equivalent to a channel used to communicate with external IT entities (FTP_ITC.1) or existing TOE components (FPT_ITT.1/INTER_XFER and FPT_ITT.1/INTER_XFER_AGENT), or else a separate type of channel that is specific to registration (FTP_TRP.1/TRUSTPATH_ENROLL or FTP_TRP.1/TRUSTPATH_JOIN). If the TOE does not require a communications channel for registration (e.g., because the registration is achieved entirely by configuration actions by an administrator at each of the components) then the main selection in FCO_CPC_EXT.1.2 is completed with the "No channel" option.

If the ST author selects the FTP_ITC.1 or FPT_ITT.1/INTER_XFER and FPT_ITT.1/INTER_XFER_AGENT channel type in the main selection in FCO_CPC_EXT.1.2 then the TSS identifies the relevant SFR iteration that specifies the channel used. If the ST author selects the FTP_TRP.1/TRUSTPATH_ENROLL or FTP_TRP.1/TRUSTPATH_JOIN channel type, then the TSS (possibly with support from the operational guidance) describes details of the channel and the mechanisms that it uses (and describes how the registration process ensures that the channel can only be used by the intended joiner and gatekeeper). Note that the FTP_TRP.1/TRUSTPATH_ENROLL or FTP_TRP.1/TRUSTPATH_JOIN channel type may require support from security measures in the operational environment (see the definition of FTP_TRP.1/TRUSTPATH_ENROLL or FTP_TRP.1/TRUSTPATH_JOIN for details).

If the ST author selects the FTP_ITC.1 or FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT channel type in the main selection in FCO_CPC_EXT.1.2 then the ST identifies the registration channel as a separate iteration of FTP_ITC.1 or FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT and gives the iteration identifier (e.g., "FPT_ITT.1/Join") in an ST Application Note for FCO_CPC_EXT.1.

Note that the channel that is set up and used for registration may be adopted as a continuing internal communication channel (i.e., between different TOE components) provided that the channel meets the requirements of FTP_ITC.1 or FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT. Otherwise the registration channel is closed after use and a separate channel is used for the internal communications.

Specific requirements for Preparative Procedures relating to FCO_CPC_EXT.1 are defined in the Evaluation Activities.

Evaluation Activities

FCO_CPC_EXT.1
TSS
If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the audit record protection functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
The evaluator shall examine the guidance documentation to confirm that it contains instructions for enabling and disabling communications with any individual component of a distributed TOE. The evaluator shall confirm that the method of disabling is such that all other components can be prevented from communicating with the component that is being removed from the TOE (preventing the remaining components from either attempting to initiate communications to the disabled component, or from responding to communications from the disabled component).

Tests
- *Test FCO_CPC_EXT.1:1: The evaluator shall confirm that an IT entity that is not currently a member of the distributed TOE cannot communicate with any component of the TOE until the non-member entity is enabled by an administrator for each of the non-equivalent TOE components that it is required to communicate with (non-equivalent TOE components are as defined in the minimum configuration for the distributed TOE)*
- *Test FCO_CPC_EXT.1:2: The evaluator shall confirm that after enablement, an IT entity can communicate only with the components that it has been enabled for. This includes testing that the enabled communication is successful for the enabled component pair, and that communication remains unsuccessful with any other component for which communication is possible but has not been explicitly enabled.*

Some TOEs may set up the registration channel before the enablement step is carried out, but in such a case the channel must not allow communications until after the enablement step has been completed.

- *Test FCO_CPC_EXT.1:3: The evaluator shall separately disable each TOE component in turn and ensure that the other TOE components cannot then communicate with the disabled component, whether by attempting to initiate communications with the disabled component or by responding to communication attempts from the disabled component. In situations where one component acts as the 'Gatekeeper' for all other components, the test would involve disabling the components in turn on the Gatekeeper and ensuring that the TOE no longer communicates with disabled components.*

A.2.4 Class: Identification and Authentication (FIA)

FIA_UAU.4 Sinlgle-Use Authenfztication Mechanisms

FIA_UAU.4.1

The TSF shall prevent reuse of authentication data related to[**assignment**: *identified authentication mechanisms*].

Application Note: This requirement references the authentication mechanisms used to authenticate the user for enrollment in FIA_ENR_EXT.1.1 . If a username and password is used to authenticate the user for enrollment, the password must not be reused. Thus if the user has two devices enrolled in management or needs to re-enroll the same device (i.e., after a device wipe), the password must be different for each enrollment. Additionally, if two different users are enrolling the password must be different for each user.

Evaluation Activities

FIA_UAU.4
TSS
The evaluator shall verify that the TSS contains a description of the process of enrollment for each MDM agent or platform listed as supported in the ST. This description shall include the method of user authentication (username or password, token, etc.) and how reuse of the authentication data is prevented.

Guidance
The evaluator shall ensure that the administrative guidance describes the methods of restricting user enrollment and that it instructs the administrator on how to configure the restrictions.

Tests

- *Test FIA_UAU.4:1: The evaluator shall enroll a device providing correct credentials. The evaluator shall attempt to enroll a second device using the same credentials used to enroll the first device. The evaluator shall verify that the second device could not enroll.*

FIA_X509_EXT.3 X.509 Enrollment

FIA_X509_EXT.3.1

The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and[**selection**: *device-specific information*, *Common Name*, *Organization*, *Organizational Unit*, *Country*].

Application Note: The public key is the public key portion of the public-private key pair generated by the TOE as specified in FCS_CKM.1.1 .

As Enrollment over Secure Transport (EST) is a new standard that has not yet been widely adopted, this requirement is included as an interim objective requirement in order to allow developers to distinguish those products which have to have the ability to generate Certificate Request Messages but do not yet implement EST.

FIA_X509_EXT.3.2

The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

Evaluation Activities

FIA_X509_EXT.3
TSS
If the ST author selects "device-specific information," the evaluator shall verify that the TSS contains a description of the device-specific fields used in certificate requests.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
The evaluator shall check to ensure that the operational guidance contains instructions on requesting certificates from a CA, including generation of a Certificate Request Message. If the ST author selects "Common Name," "Organization," "Organizational Unit," or "Country," the evaluator shall ensure that this guidance includes instructions for establishing these fields before creating the certificate request message.

Tests
- *Test FIA_X509_EXT.3:1: The evaluator shall use the operational guidance to cause the TOE to generate a certificate request message. The evaluator shall capture the generated message and ensure that it conforms to the format specified. The evaluator shall confirm that the certificate request provides the public key and other required information, including any necessary user-input information.*
- *Test FIA_X509_EXT.3:2: The evaluator shall demonstrate that validating a certificate response message without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the certificate response message, and demonstrate that the function succeeds. The evaluator shall then delete one of the certificates, and show that the function fails.*

FIA_X509_EXT.4 Alternate X.509 Enrollment

FIA_X509_EXT.4.1

The TSF shall use the Enrollment over Secure Transport (EST) protocol as specified in RFC 7030 to request certificate enrollment using the simple enrollment method described in RFC 7030 Section 4.2.

FIA_X509_EXT.4.2

The TSF shall be capable of authenticating EST requests using an existing certificate and corresponding private key as specified by RFC 7030 Section 3.3.2.

FIA_X509_EXT.4.3

The TSF shall be capable of authenticating EST requests using HTTP Basic Authentication with a username and password as specified by RFC 7030 Section 3.2.3.

FIA_X509_EXT.4.4

The TSF shall perform authentication of the EST server using an Explicit Trust Anchor following the rules described in RFC 7030, section 3.6.1.

Application Note: EST also uses HTTPS as specified in FCS_HTTPS_EXT.1 to establish a secure connection to an EST server, and thus, the ST author must also include FCS_HTTPS_EXT.1 in the main body of the ST. The separate Trust Anchor Database dedicated to EST operations is described as Explicit Trust Anchors in RFC 7030.

FIA_X509_EXT.4.5

The TSF shall be capable of requesting server-provided private keys as specified in RFC 7030 Section 4.4.

FIA_X509_EXT.4.6

The TSF shall be capable of updating its EST-specific Trust Anchor Database using the "Root CA Key Update" process described in RFC 7030 Section 4.1.3.

FIA_X509_EXT.4.7

The TSF shall generate a Certificate Request Message for EST as specified in RFC 2986 and be able to provide the following information in the request: public key and[**selection**:
- *device-specific information*
- *Common Name, Organization, Organizational Unit, and Country*

].

FIA_X509_EXT.4.8

The TSF shall validate the chain of certificates from the Root CA certificate in the Trust Anchor Database to the EST Server CA certificate upon receiving a CA Certificates Response.

Application Note: The public key referenced in FIA_X509_EXT.4.7 is the public key portion of the public-private key pair generated by the TOE as specified in FCS_CKM.1.

Evaluation Activities

FIA_X509_EXT.4
TSS
TBD

Guidance
The evaluator shall check to ensure that the operational guidance contains instructions on requesting certificates from an EST server, including generating a Certificate Request Message.

Tests
The evaluator shall also perform the following tests. Other tests are performed in conjunction with the TLS evaluation activities.

- *Test FIA_X509_EXT.4:1: The evaluator shall use the operational guidance to cause the TOE to request certificate enrollment from an EST server using the simple enrollment method described in RFC 7030 Section 4.2, authenticating the certificate request to the server using an existing certificate and private key as described by RFC 7030 Section 3.3.2. The evaluator shall confirm that the resulting certificate is successfully obtained and installed in the TOE key store.*
- *Test FIA_X509_EXT.4:2: The evaluator shall use the operational guidance to cause the TOE to request certificate enrollment from an EST server using the simple enrollment method described in RFC 7030 Section 4.2, authenticating the certificate request to the server using a username and password as described by RFC 7030 Section 3.2.3. The evaluator shall confirm that the resulting certificate is successfully obtained and installed in the TOE key store.*
- *Test FIA_X509_EXT.4:3: The evaluator shall modify the EST server to return a certificate containing a different public key than the key included in the TOEs certificate request. The evaluator shall use the operational guidance to cause the TOE to request certificate enrollment from an EST server. The evaluator shall confirm that the TOE does not accept the resulting certificate since the public key in the issued certificate does not match the public key in the certificate request.*
- *Test FIA_X509_EXT.4:4: The evaluator shall configure the EST server or use a man-in-the-middle tool to present a server certificate to the TOE that is present in the TOE general Trust Anchor Database but not its EST-specific Trust Anchor Database. The evaluator shall cause the TOE to request certificate enrollment from the EST server. The evaluator shall verify that the request is not successful.*
- *Test FIA_X509_EXT.4:5: The evaluator shall configure the EST server or use a man-in-the-middle tool to present an invalid certificate. The evaluator shall cause the TOE to request certificate enrollment from the EST server. The evaluator shall verify that the request is not successful. The evaluator shall configure the EST server or use a man-in-the-middle tool to present a certificate that does not have the CMC RA purpose and verify that requests to the EST server fail. The tester shall repeat the test using a valid certificate and a certificate that contains the CMC RA purpose and verify that the certificate enrollment requests succeed.*
- *Test FIA_X509_EXT.4:6: The evaluator shall use a packet sniffing tool between the TOE and an EST server. The evaluator shall turn on the sniffing tool and cause the TOE to request certificate enrollment from an EST server. The evaluator shall verify that the EST protocol interaction occurs over a Transport Layer Security (TLS) protected connection. The evaluator is not expected to decrypt the connection but rather observe that the packets conform to the TLS protocol format.*
- *Test FIA_X509_EXT.4:7: The evaluator shall use the operational guidance to cause the TOE to request a server-provided private key and certificate from an EST server. The evaluator shall confirm that the resulting private key and certificate are successfully obtained and installed in the TOE key store.*
- *Test FIA_X509_EXT.4:8: The evaluator shall modify the EST server to, in response to a server-provided private key and certificate request, return a private key that does not correspond with the public key in the returned certificate. The evaluator shall use the operational guidance to cause the TOE to request a server-provided private key and certificate. The evaluator shall confirm that the TOE does not accept the resulting private key and certificate since the private key and public key do not correspond.*
- *Test FIA_X509_EXT.4:9: The evaluator shall configure the EST server to provide a "Root CA Key Update" as described in RFC 7030 Section 4.1.3. The evaluator shall cause the TOE to*

request CA certificates from the EST server and shall confirm that the EST-specific Trust Anchor Database is updated with the new trust anchor.

- *Test FIA_X509_EXT.4:10: The evaluator shall configure the EST server to provide a "Root CA Key Update" as described in RFC 7030 Section 4.1.3, but shall modify part of the NewWithOld certificate's generated signature. The evaluator shall cause the TOE to request CA certificates from the EST server and shall confirm that the EST-specific Trust Anchor Database is not updated with the new trust anchor since the signature did not verify.*
- *Test FIA_X509_EXT.4:11: The evaluator shall use the operational guidance to cause the TOE to generate a certificate request message. The evaluator shall capture the generated message and ensure that it conforms with the format specified by RFC 2986. The evaluator shall confirm that the certificate request provides the public key and other required information, including any necessary user-input information.*

A.2.5 Class: Security Management (FMT)

FMT_SAE_EXT.1 Security Attribute Expiration

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.

Application Note: This requirement references the user authenticator used for device enrollment in management in FIA_ENR_EXT.1.1 . The user authenticator must only be valid for a configurable time limit. If the authenticator is expired, even if entered correctly, enrollment must not occur.

The length of the time the authenticator is valid for is configured per function c.5 in FMT_SMF.1/SERVER_CONF_SERVER. If FMT_SAE_EXT.1 is included in the ST, then function g must be selected in FMT_SMF.1/SERVER_CONF_SERVER.

Evaluation Activities

FMT_SAE_EXT.1
TSS
The evaluator shall verify that the TSS contains a description of the process of enrollment for each MDM agent or platform listed as supported in the ST. This description shall be the method of user authentication (username or password, token, etc.).

Guidance
The evaluator shall check to ensure that the operational guidance contains instructions to configure the expiration time for each method of user authentication listed in the TSS.

Tests

- *Test FMT_SAE_EXT.1:1: The evaluator shall configure the MDM server according to the administrative guidance to set an expiration time for the enrollment authentication data. For each method of user authentication listed in the TSS, the evaluator shall attempt to enroll using authentication data that has expired. The evaluator shall verify that enrollment was unsuccessful.*

A.2.6 Class: Trusted Path/Channels (FTP)

FTP_TRP.1/TRUSTPATH_JOIN Trusted Path (for Joining)

FTP_TRP.1.1/TRUSTPATH_JOIN

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** provide a communication path between itself and **a joining component** that is logically distinct from other communication paths and provides assured identification of[**selection**: ***the TSF endpoint***, ***both joining component and TSF endpoint***]and protection of the communicated data from [*modification*] **and**[**selection**: *disclosure*, ***none***].

FTP_TRP.1.2/TRUSTPATH_JOIN

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** permit[**selection**: *the TSF*, ***the joining***

component]to initiate communication via the trusted path.

FTP_TRP.1.3/TRUSTPATH_JOIN

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]**to** require the use of the trusted path for [*joining components to the TSF under environmental constraints identified in[**assignment**: reference to operational guidance]]* .

Application Note: This SFR implements one of the types of channel identified in the main selection for FCO_CPC_EXT.1.2 . The "joining component" in FTP_TRP.1/TRUSTPATH_JOIN is the IT entity that is attempting to join the distributed TOE by using the registration process.

The effect of this SFR is to require the ability for components to communicate in a secure manner while the distributed TSF is being created (or when adding components to an existing distributed TSF). When creating the TSF from the initial pair of components, either of these components may be identified as the TSF for the purposes of satisfying the meaning of "TSF" in this SFR.

The selection at the end of FTP_TRP.1/TRUSTPATH_JOIN recognises that in some cases confidentiality (i.e., protection of the data from disclosure) may not be provided by the channel. The ST author distinguishes in the TSS whether in this case the TOE relies on the environment to provide confidentiality (as part of the constraints referenced in FTP_TRP.1.3/TRUSTPATH_JOIN) or whether the registration data exchanged does not require confidentiality (in which case this assertion must be justified). If "none" is selected, then this word may be omitted in the ST to improve readability.

The assignment in FTP_TRP.1.3/TRUSTPATH_JOIN ensures that the ST highlights any specific details needed to protect the registration environment. Note that when the ST uses FTP_TRP.1/TRUSTPATH_JOIN for the registration channel then this channel cannot be reused as the normal inter-component communication channel (the latter channel must meet FTP_ITC.1 or FPT_ITT.1/INTER_XFER / FPT_ITT.1/INTER_XFER_AGENT). Specific requirements for Preparative Procedures relating to FTP_TRP.1/TRUSTPATH_JOIN are defined in the Evaluation Activities.

Evaluation Activities

FTP_TRP.1/TRUSTPATH_JOIN

TSS

The evaluator shall examine the TSS to determine that the methods of joining TOE components are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of joining are consistent with those specified in the requirement, and are included in the requirements in the ST.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance

The evaluator shall confirm that the operational guidance contains instructions for joining TOE components for each supported method.

Tests

The evaluator shall also perform the following tests:

- *Test FTP_TRP.1/TRUSTPATH_JOIN:1: The evaluator shall ensure that the communications path for joining components to the TSF is tested for each distinct (nonequivalent) component type, setting up the connections as described in the guidance documentation and ensuring that communication is successful. In particular the evaluator shall confirm that requirements on environment protection for the registration process are consistent with observations made on the test configuration (for example, a requirement to isolate the components from the Internet during registration might be inconsistent with the need for a component to contact a license server). If no requirements on the registration environment are identified as necessary to protect confidentiality, then the evaluator shall confirm that the key used for registration can be configured (following the instructions in the guidance documentation) to be at least the same length as the key used for the internal TSF channel that is being enabled. The evaluator shall confirm that the key used for the channel is unique to the pair of components (this is done by identifying the relevant key during the registration test: it is not necessary to examine the key value).*
- *Test FTP_TRP.1/TRUSTPATH_JOIN:2: The evaluator shall follow the guidance*

documentation to ensure that in fact the communication channel can be enabled by an administrator for all the TOE components identified in the guidance documentation as capable of initiation.

- *Test FTP_TRP.1/TRUSTPATH_JOIN:3: The evaluator shall ensure that if the guidance documentation states that the channel data is encrypted then the data observed on the channel is not plaintext.*
- *Test FTP_TRP.1/TRUSTPATH_JOIN:4: The evaluator shall ensure that, for each different pair of nonequivalent component types that can use the registration channel, the connection is physically interrupted during a joining attempt. The evaluator shall ensure that when physical connectivity is restored, communications are appropriately protected.*

A.3 Implementation-dependent Requirements

This PP does not define any Implementation-dependent requirements.

Appendix B - Selection-based Requirements

As indicated in the introduction to this PP, the baseline requirements (those that must be performed by the TOE or its underlying platform) are contained in the body of this PP. There are additional requirements based on selections in the body of the PP: if certain selections are made, then additional requirements below must be included.

B.1 Auditable Events for Selection-based Requirements

Table 4: Auditable Events for Selection-based Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1/MAS_SERVER	No events specified	N/A
FAU_STG.2	No events specified	N/A
FCS_HTTPS_EXT.1	Failure of the certificate validity check	• Issuer Name and Subject Name of certificate • User's authorization decision
FCS_IV_EXT.1	No events specified	N/A
FCS_RBG.2	No events specified	N/A
FCS_RBG.3	No events specified	N/A
FCS_RBG.4	No events specified	N/A
FCS_RBG.5	No events specified	N/A
FCS_STG_EXT.2	No events specified	N/A
FIA_TOK_EXT.1	No events specified	N/A
FIA_X509_EXT.1/CERTVAL_SEL	Failure to validate X.509 certificate	Reason for failure
FMT_MOF.1/MANAGEMENT_MAS	No events specified	N/A
FMT_SMF.1/MAS	No events specified	N/A
FMT_SMR.1/SECMAN_ROLES_MAS	No events specified	N/A
FPT_ITT.1/INTER_XFER	Initiation and termination of the trusted channel	• Trusted channel protocol • Identity of initiator and recipient
FPT_ITT.1/INTER_XFER_AGENT	Initiation and termination of the trusted channel	• Trusted channel protocol • Identity of initiator and recipient
FTP_ITC.1/INTER_TSF_XFER_AGENT	Initiation and termination of the trusted channel	• Trusted channel protocol • Non-TOE endpoint of connection

B.2 Class: Security Audit (FAU)

FAU_GEN.1/MAS_SERVER Audit Data Generation (MAS Server)

> ***The inclusion of this selection-based component depends upon selection in FMT_MOF.1.1/FUNCBE.***

FAU_GEN.1.1/MAS_SERVER

The **MAS Server** shall be able to generate audit data of the following auditable events: [

a. *Failure to push a new application on a managed mobile device*
b. *Failure to update an existing application on a managed mobile device.*

]

Application Note: The MDM agent is required to report to the MAS Server on successful receipt of an application or update on a managed mobile device, and failures can be inferred from the absence of such alerts.

FAU_GEN.1.2/MAS_SERVER

The[**selection**: ***MAS Server***, ***MAS Server platform***]shall record within the audit data at least the following information:

- date and time of the auditable event
- type of event
- mobile device identity
- [**assignment**: *other audit relevant information*]

Application Note: All audits must contain at least the information mentioned in FAU_GEN.1.2/MAS_SERVER , but may contain more information which can be assigned. The ST author must identify in the TSS which information of the audit data is performed by the TSF and that which is performed by the TOE platform.

This requirement is claimed if "enable, disable, and modify policies listed in FMT_SMF.1/MAS" is selected in FMT_MOF.1.1/FUNCBE.

Evaluation Activities

FAU_GEN.1/MAS_SERVER
TSS
The evaluator shall check the TSS and ensure that it provides a format for audit data. Each audit data format type must be covered, along with a brief description of each field.

Guidance
The evaluator shall check the administrative guide and ensure that it provides a format for audit data. Each audit data format type must be covered, along with a brief description of each field.

The evaluator shall check to make sure that the description of the fields contains the information required in FAU_STG.2.1 .

Tests
The evaluator shall verify that when an application or update push fails, that the audit data generated match the format specified in the guidance and that the fields in each audit data have proper entries.

When verifying the test results from FMT_MOF.1.1/FUNCBE , the evaluator shall ensure the audit data generated during testing match the format specified in the administrative guide, and that the fields in each audit data have the proper entries.

Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to ensure that the administrative guidance provided is correct verifies that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit data are generated as expected.

FAU_STG.2 Audit Event Storage

The inclusion of this selection-based component depends upon selection in FAU_STG.1.1.

FAU_STG.2.1

The TSF shall[**selection**: ***invoke platform-provided functionality***, ***implement functionality***]to protect the stored audit data in the audit trail from unauthorized **modification** .

Application Note: If "store audit data locally" is selected in FAU_STG.1.1 , this SFR must be included in the ST.

The purpose of this requirement is to ensure that audit data are stored securely. The ST author is responsible for selecting whether audit data are maintained when audit storage or failure occurs. The ST author must choose a means by which audit data are saved and select the events during which the data will be saved. The TSF may rely on the underlying operating system for this

functionality, and the first selection should be made appropriately.

FAU_STG.2.2

The TSF shall be able to[**selection, choose one of**: *prevent*, *detect*]unauthorized modifications to the stored audit data in the audit trail.

Evaluation Activities

FAU_STG.2
TSS
If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the audit data protection functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If "implement functionality" is selected, the evaluator shall ensure that the TSS describes how the audit data are protected from unauthorized modification or deletion. The evaluator shall ensure that the TOE uses audit trail specific protection mechanisms.

Guidance
TBD

Tests
The evaluator shall perform the following tests:
- *Test FAU_STG.2:1: The evaluator shall access the audit trail as an unauthorized user and attempt to modify and delete the audit data. The evaluator shall verify that these attempts fail.*
- *Test FAU_STG.2:2: The evaluator shall access the audit trail as an authorized user and attempt to modify and delete the audit data. The evaluator shall verify that these attempts succeed. The evaluator shall verify that only the records intended for modification and deletion are modified and deleted.*

B.3 Class: Cryptographic Support (FCS)

FCS_HTTPS_EXT.1 HTTPS Protocol

The inclusion of this selection-based component depends upon selection in FPT_ITT.1.1/INTER_XFER, FPT_ITT.1.1/INTER_XFER_AGENT, FTP_ITC.1.1/INTER_TSF_XFER_AGENT, FTP_ITC.1.1/INTER_XFER_IT, FTP_TRP.1.1/TRUSTPATH_ENROLL, FTP_TRP.1.1/TRUSTPATH_REM_ADMIN.

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.

Application Note: The TLS Functional Package must be included in the ST, with the following selections made:
- FCS_TLS_EXT.1:
  - TLS must be selected
  - Either client or server is selected as appropriate
- FCS_TLSC_EXT.1.1 or FCS_TLSS_EXT.1.1 (as appropriate):
  - The cipher suites selected must correspond with the algorithms and hash functions allowed in FCS_COP.1.

Validity is determined by the certificate path, the expiration date, and the revocation status in accordance with RFC 5280.

The requirement is claimed if the TSF selects HTTPS in any iteration of FPT_ITT.1, FTP_ITC.1, or FTP_TRP.1.

Evaluation Activities

FCS_HTTPS_EXT.1

TSS
TBD

Guidance
TBD

Tests
- *Test FCS_HTTPS_EXT.1:1: The evaluator shall attempt to establish an HTTPS connection with a web server, observe the traffic with a packet analyzer, and verify that the connection succeeds and that the traffic is identified as TLS or HTTPS.*

FCS_IV_EXT.1 Initialization Vector Generation

The inclusion of this selection-based component depends upon selection in FCS_STG_EXT.1.1.

FCS_IV_EXT.1.1

The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to generate IVs in accordance withTable 5 .

Application Note: This requirement must be included in the ST if the selection in FCS_STG_EXT.1 indicates that the TSF is protecting private keys and persistent secrets with encryption rather than the platform-provided key storage.

Table 5 lists the requirements for composition of IVs according to the corresponding NIST Special Publications for each cipher mode. The composition of IVs generated for encryption according to a cryptographic protocol is addressed by the protocol. Thus, this requirement addresses only IVs generated for key storage encryption.

Table 5: References and IV Requirements for NIST-approved Cipher Modes

Cipher Mode	Reference	IV Requirement
Electronic Codebook (ECB)	SP800-38A	No IV
Counter (CTR)	SP800-38A	"Initial Counter" shall be non-repeating. No counter value shall be repeated across multiple messages with the same secret key.
Cipher Block Chaining (CBC)	SP800-38A	IVs shall be unpredictable. Repeating IVs leak information about whether the first one or more blocks are shared between two messages, so IVs should be non-repeating in such situations.
Output Feedback (OFB)	SP800-38A	IVs shall be non-repeating and shall not be generated by invoking the cipher on another IV.
Cipher Feedback (CFB)	SP800-38A	IVs should be non-repeating as repeating IVs leak information about the first plaintext block and about common shared prefixes in messages.
XOR Encrypt XOR (XEX) Tweakable Block Cipher with Ciphertext Stealing (XTS)	SP800-38E	No IV. Tweak values shall be non-negative integers, assigned consecutively, and starting at an arbitrary non-negative integer.
Cipher-based Message Authentication Code (CMAC)	SP800-38B	No IV
Key Wrap and Key	SP800-38F	No IV

Wrap with Padding		
Counter with CBC-Message Authentication Code (CCM)	SP800-38C	No IV. Nonces shall be non-repeating.
Galois Counter Mode (GCM)	SP800-38D	IV shall be non-repeating. The number of invocations of GCM shall not exceed 2^32 for a given secret key unless an implementation only uses 96-bit IVs (default length).

Evaluation Activities

FCS_IV_EXT.1
TSS
If "invoke platform-provided functionality" is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the IV generation is invoked for each mode selected in the MDM server's ST (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If "implement functionality" is selected:

The evaluator shall examine the TSS to ensure that it details the encryption of user credentials, persistent secrets, and private keys and the generation of the IVs used for that encryption.

Guidance
TBD

Tests
The evaluator shall ensure that the generation of IVs for each key encrypted by the same KEK meets Table 5 .

FCS_RBG.2 Random Bit Generation (External Seeding)

The inclusion of this selection-based component depends upon selection in FCS_RBG.1.2.

FCS_RBG.2.1

The TSF shall be able to accept a minimum input of[**assignment**: *minimum input length greater than zero*]from a TSF interface for the purpose of seeding.

Application Note: This requirement is claimed when a DRBG is seeded with entropy from one or more noise sources that is outside the TOE boundary. Typically the entropy produced by an environmental noise source is conditioned such that the input length has full entropy and is therefore usable as the seed. However, if this is not the case, it should be noted what the minimum entropy rate of the noise source is so that the TSF can collect a sufficiently large sample of noise data to be conditioned into a seed value.

This requirement is claimed if "TSF interface for seeding" is selected in FCS_RBG.1.2 .

Evaluation Activities

FCS_RBG.2
TSS
There are no additional TSS evaluation activities for this component.

Guidance
There are no additional Guidance evaluation activities for this component.

Tests
There are no test activities for this component.

FCS_RBG.3 Random Bit Generation (Internal Seeding - Single Source)

FCS_RBG.3.1

The TSF shall be able to seed the RBG using a[**selection, choose one of**: *TSF software-based noise source*, *TSF hardware-based noise source[**assignment**: name of noise source]*]with a minimum of[**assignment**: *number of bits*]bits of min-entropy.

Application Note: This requirement is claimed when a DRBG is seeded with entropy from a single noise source that is within the TOE boundary. Min-entropy should be expressed as a ratio of entropy bits to sampled bits so that the total amount of data needed to ensure full entropy is known, as well as the conditioning function by which that data is reduced in size to the seed.

This requirement is claimed if "TSF noise source..." is selected in FCS_RBG.1.2 .

Evaluation Activities

FCS_RBG.3
TSS
There are no additional TSS evaluation activities for this component.

Guidance
There are no additional Guidance evaluation activities for this component.

Tests
There are no test activities for this component.

FCS_RBG.4 Random Bit Generation (Internal Seeding - Multiple Sources)

FCS_RBG.4.1

The TSF shall be able to seed the RBG using[**selection**: *[**assignment**: number], TSF software-based noise sources*, *[**assignment**: number], TSF hardware-based noise sources*].

Application Note: This requirement is claimed when a DRBG is seeded with entropy from multiple noise sources that are within the TOE boundary. FCS_RBG.5 defines the mechanism by which these sources are combined to ensure sufficient minimum entropy.

This requirement is claimed if "multiple TSF noise sources..." is selected in FCS_RBG.1.2 .

Evaluation Activities

FCS_RBG.4
TSS
There are no additional TSS evaluation activities for this component.

Guidance
There are no additional Guidance evaluation activities for this component.

Tests
There are no test activities for this component.

FCS_RBG.5 Random Bit Generation (Combining Noise Sources)

FCS_RBG.5.1

The TSF shall[**assignment**: *combining operation*][**selection**: *output from TSF noise sources*, *input from TSF interfaces for seeding*]to create the entropy input into the derivation function as defined in[**assignment**: *list of standards*], resulting in a minimum of[**assignment**: *number of bits*]bits of min-entropy.

Application Note: This requirement is claimed if "multiple TSF noise sources..." is selected in FCS_RBG.1.2.

Evaluation Activities

FCS_RBG.5
TSS
There are no additional TSS evaluation activities for this component.

Guidance
There are no additional Guidance evaluation activities for this component.

Tests
There are no test activities for this component.

FCS_STG_EXT.2 Encrypted Cryptographic Key Storage

The inclusion of this selection-based component depends upon selection in FCS_STG_EXT.1.1.

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*].

Application Note: This requirement states that keys used by the TSF shall not be kept in plaintext. The intent of this requirement is to ensure that the private keys, credentials, and persistent secrets cannot be accessed in the TOE in an unencrypted state, allowing an attacker to access keys without having to exhaust the AES key space.

This requirement must be included in the ST if the selection in FCS_STG_EXT.1 indicates that the TSF is protecting private keys and persistent secrets with encryption rather than the platform-provided key storage.

Evaluation Activities

FCS_STG_EXT.2
TSS
The evaluator shall examine the TSS to ensure it describes in detail how user credentials, persistent secret and private keys are stored and encrypted. The evaluator shall review the TSS to determine that it makes a case that key material is not written unencrypted to persistent memory and that it identifies the mode of encryption.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key encryption functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
TBD

Tests
TBD

B.4 Class: Identification and Authentication (FIA)

FIA_TOK_EXT.1 Client Tokens

The inclusion of this selection-based component depends upon selection in FIA_CLI_EXT.1.1.

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.

Application Note: This SFR is included in the ST if "token as defined in FIA_TOK_EXT.1" is selected in FIA_CLI_EXT.1.

Evaluation Activities

FIA_TOK_EXT.1
TSS
The evaluator shall review the TSS and verify that the TSF uses either unique identifiers from the client device or a server-specific mechanism to generate a unique token that will be used for verifying the identity of the client device. If the server generates the token using cryptographic functions, it must use algorithms in FCS_COP.1(ANY) (specific algorithms as needed by the vendor).

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If "implement functionality" is selected, the evaluator shall examine the TSS to verify that it describes the methods to generate the token.

Guidance
None.

Tests
For each MDM agent or platform listed as supported in the ST:
- *Test FIA_TOK_EXT.1:1: The evaluator shall use appropriate combinations of specialized operational environment and development tools (debuggers, simulators, etc.) for the TOE and instrumented TOE builds as needed to perform this test.*
- *Test FIA_TOK_EXT.1:2: The evaluator shall concurrently enroll 10 devices and ensure that the token for each is unique, per the methods described in the TSS.*

FIA_X509_EXT.1/CERTVAL_SEL X.509 Certificate Validation

The inclusion of this selection-based component depends upon selection in FPT_ITT.1.1/INTER_XFER.

FIA_X509_EXT.1.1/CERTVAL_SEL

The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension, that the CA flag is set to TRUE for all CA certificates, and that any path constraints are met.
- The TSF shall validate that any CA certificate includes caSigning purpose in the key usage field.
- The TSF shall validate the revocation status of the certificate using[**selection**: *OCSP as specified in RFC 6960, CRL as specified in RFC 5280 Section 6.3, a CRL as specified in RFC 5759 Section 5, an OCSP TLS Status Request Extension (i.e., OCSP stapling) as specified in RFC 6066, OCSP TLS Multi-Certificate Status Request Extension (i.e., OCSP Multi-Stapling) as specified in RFC 6961, no revocation method*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
  - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
  - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the EKU field.
  - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the EKU field.
  - Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in

the EKU field.

Application Note: FIA_X509_EXT.1.1/CERTVAL_SEL should be chosen if the TOE is distributed and the protocols selected in FPT_ITT.1/INTER_XFER use X.509 certificates for peer authentication. In this case, the use of revocation list checking is optional as there are additional requirements surrounding the enabling and disabling of the ITT channel as defined in FCO_CPC_EXT.1. If revocation checking is not supported, the ST author should select "no revocation method." However, if certificate revocation checking is supported, the ST author selects whether this is performed using OCSP or CRLs.

This SFR lists the rules for validating certificates. The ST author selects whether revocation status is verified using OCSP or CRLs. FIA_X509_EXT.2 requires that certificates are used for trusted channels; this use requires that the extendedKeyUsage rules are verified. Certificates may optionally be used for code signing and policy signing and, if implemented, must be validated to contain the corresponding extendedKeyUsage.

OCSP stapling and OCSP multi-stapling only support TLS server certificate validation. If other certificate types are validated, either OCSP or CRL should be claimed. If OCSP is not supported the EKU provision for checking the OCSP Signing purpose is met by default.

Regardless of the selection of *implement functionality* or *invoke platform-provided functionality* , the validation is expected to end in a trusted root CA certificate in a root store managed by the platform.

FIA_X509_EXT.1.2/CERTVAL_SEL

The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to treat a certificate as a CA certificate only if the basicConstraints extension is present and the CA flag is set to TRUE.

Application Note: This requirement applies to certificates that are used and processed by the TOE or platform and restricts the certificates that may be added as trusted CA certificates.

Evaluation Activities

FIA_X509_EXT.1.1/CERTVAL_SEL
TSS
If "invoke platform-provided functionality" is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

The TSS must describe when revocation checking is performed. It is expected that revocation checking is performed when a certificate is used in an authentication step. It is not sufficient to verify the status of an X.509 certificate only when it is loaded onto the device. If "implement functionality" is selected:

The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place. The evaluator ensures the TSS also provides a description of the certificate path validation algorithm.

The TSS must describe when revocation checking is performed. It is expected that revocation checking is performed when a certificate is used in an authentication step. It is not sufficient to verify the status of an X.509 certificate only when it is loaded onto the device.

Guidance
TBD

Tests

The tests described must be performed in conjunction with the other certificate services evaluation activities, including each of the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator shall create a chain of at least three certificates: the node certificate to be tested, an Intermediate CA, and the self-signed Root CA.

- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:1: The evaluator shall demonstrate that validating a certificate without a valid certification path results in the function failing, for each of the following reasons, in turn:*
  - *by establishing a certificate path in which one of the issuing certificates is not a CA certificate,*

- *by omitting the basicConstraints field in one of the issuing certificates,*
- *by setting the basicConstraints field in an issuing certificate to have CA=False,*
- *by omitting the CA signing bit of the key usage field in an issuing certificate, and*
- *by setting the path length field of a valid CA field to a value strictly less than the certificate path.*

The evaluator shall then establish a valid certificate path consisting of valid CA certificates, and demonstrate that the function succeeds. The evaluator shall then remove trust in one of the CA certificates, and show that the function fails.

- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:3: The evaluator shall test that the TOE can properly handle revoked certificates--conditional on whether CRL, OCSP or OCSP stapling is selected; if multiple methods are selected, then a test shall be performed for each method. The evaluator shall test revocation of the node certificate and revocation of the intermediate CA certificate (i.e., the intermediate CA certificate should be revoked by the root CA). If OCSP stapling per RFC 6066 is the only supported revocation method, testing revocation of the intermediate CA certificate is omitted. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails.*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:4: If OCSP option is selected, the evaluator shall send the TOE an OCSP response signed by a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall cause a CA to sign a CRL with a certificate that has a key usage extension but does not have the cRLsign key usage bit set, and verify that validation of the CRL fails.*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:5: The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:6: The evaluator shall modify any byte in the last byte of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)*
- *Test FIA_X509_EXT.1.1/CERTVAL_SEL:8:*
  Test 8a: (Conditional on support for EC certificates as indicated in FCS_COP.1/SIGN_ALG). The evaluator shall establish a valid, trusted certificate chain consisting of an EC leaf certificate, an EC Intermediate CA certificate not designated as a trust anchor, and an EC certificate designated as a trusted anchor, where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.
  Test 8b: (Conditional on support for EC certificates as indicated in FCS_COP.1/SIGN_ALG). The evaluator shall replace the intermediate certificate in the certificate chain for Test 8a with a modified certificate, where the modified intermediate CA has a public key information field where the EC parameters uses an explicit format version of the Elliptic Curve parameters in the public key information field of the intermediate CA certificate from Test 8a, and the modified Intermediate CA certificate is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.

FIA_X509_EXT.1.2/CERTVAL_SEL

TSS

If "invoke platform-provided functionality" is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance

TBD

Tests

. The evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA.

- *Test FIA_X509_EXT.1.2/CERTVAL_SEL:1: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate does not contain the basicConstraints extension. The validation of the certificate path fails.*
- *Test FIA_X509_EXT.1.2/CERTVAL_SEL:2: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate has the cA flag in the basicConstraints extension not set. The validation of the certificate path fails.*
- *Test FIA_X509_EXT.1.2/CERTVAL_SEL:3: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate has the cA flag in the basicConstraints extension set to TRUE. The validation of the certificate path succeeds.*

B.5 Class: Security Management (FMT)

FMT_MOF.1/MANAGEMENT_MAS Management of Functions in (MAS Server Downloads)

The inclusion of this selection-based component depends upon selection in FMT_MOF.1.1/FUNCBE.

FMT_MOF.1.1/MANAGEMENT_MAS

The **MAS Server** shall restrict the ability to [*enable, modify the behavior of*] the functions [*downloading applications*] to [*enrolled mobile devices that are compliant with MDM policies and assigned to a user in the application access group*].

Application Note: This requirement is claimed if "enable, disable, and modify policies listed in FMT_SMF.1/MAS" is selected in FMT_MOF.1.1/FUNCBE.

Evaluation Activities

FMT_MOF.1/MANAGEMENT_MAS
TSS
The evaluator shall examine the TSS to determine that all methods of initiating an application download or update push are specified.

Guidance
The evaluator shall confirm that the operational guidance contains how to initiate an application download or update push.

Tests
The evaluator shall ensure that the MAS Server verifies that the mobile device is enrolled in the MDM server and is in a compliant state. The evaluator shall verify that an application cannot be downloaded from the MAS Server prior to enrolling the device with the MDM. The evaluator shall partially enroll the mobile device, so the device is connected to the MDM server, but is not compliant and verify that applications cannot be downloaded.

FMT_SMF.1/MAS Specification of Management Functions (MAS Server)

The inclusion of this selection-based component depends upon selection in FMT_MOF.1.1/FUNCBE.

FMT_SMF.1.1/MAS

The **MAS Server** shall be capable of performing the following management functions: [

- *Configure application access groups*
- *Download applications*
- *[**selection**: [**assignment**: other MAS management functions], no other functions]*

]

Application Note: This requirement captures all the configuration functionality in the MAS Server to configure the underlying MAS Server. The ST author can add more commands and configuration policies by completing the assignment statement.

The MAS Server must be able to create groups to configure which applications a user can access based on which group they are in. If the MAS Server uses the groups defined by the MDM, then it must communicate with the MDM server (if separate server) to determine which applications the user can access.

This requirement is claimed if "enable, disable, and modify policies listed in FMT_SMF.1/MAS" is selected in FMT_MOF.1.1/FUNCBE.

Evaluation Activities

FMT_SMF.1/MAS
TSS
The evaluator shall examine the TSS to ensure that it describes each management function listed.

The evaluator shall examine the TSS to determine if the MAS Server creates its own groups or relies on the groups specified by the MDM server.

Guidance
The evaluator shall confirm that the operational guidance contains how to create and define user groups and how to specify which applications are accessible by which group.

The evaluator shall verify the operational guidance includes detailed instructions of what options are available and how to configure each management functional capability listed.

Tests
The evaluator shall ensure that the MAS client can only access the applications specified for the group they are enrolled in. The evaluator shall create a user group, making sure that the MAS client user is excluded from the group. Verify that an application accessible to that group cannot be accessed. The evaluator shall include the MAS client user in the group and assure that the application can be accessed.

FMT_SMR.1/SECMAN_ROLES_MAS Security Roles (MAS Server)

The inclusion of this selection-based component depends upon selection in FMT_MOF.1.1/FUNCBE.

FMT_SMR.1.1/SECMAN_ROLES_MAS
The TSF shall **additionally** maintain the roles of [*enrolled mobile devices, application access groups, and[**assignment**: additional authorized identified roles]*].

FMT_SMR.1.2/SECMAN_ROLES_MAS
The **MAS Server** shall be able to associate users with roles.

Application Note: It is envisioned that the MAS Server will be configured and maintained by different user roles. The assignment is used by the ST author to list the roles that are supported. At a minimum, one administrative role must be supported. If no additional roles are supported, then "no additional roles" is stated. The MD user role is used for enrollment of mobile devices to the MAS according to FIA_ENR_EXT.1.

This requirement is claimed if "enable, disable, and modify policies listed in FMT_SMF.1/MAS" is selected in FMT_MOF.1.1/FUNCBE.

Evaluation Activities

FMT_SMR.1/SECMAN_ROLES_MAS
TSS
The evaluator shall examine the TSS to verify that it describes the administrator role and the powers granted to and limitations of the role.

Guidance
The evaluator shall review the operational guidance to ensure that it contains instructions for administering the TOE and which interfaces are supported.

Tests
In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this PP be tested; for instance, if the TOE can be administered through a local hardware interface or HTTPS then both methods of administration must be exercised during the evaluation team's test activities.

B.6 Class: Protection of the TSF (FPT)

FPT_ITT.1/INTER_XFER Internal TOE TSF Data Transfer

The inclusion of this selection-based component depends upon selection in FTP_ITC_EXT.1.1.

FPT_ITT.1.1/INTER_XFER

The TSF shall[**selection**:

- ***invoke platform-provided functionality to use[selection: IPsec, mutually authenticated TLS, mutually authenticated DTLS, HTTPS, SSH]***
- ***implement functionality using[selection: IPsec as defined in the PP-Module for VPN Client, mutually authenticated TLS as defined in the Package for Transport Layer Security, mutually authenticated DTLS as defined in the Package for Transport Layer Security, HTTPS in accordance with FCS_HTTPS_EXT.1, SSH as defined in the Functional Package for Secure Shell]***

]**to** protect **all** data from [*disclosure and modification*] when it is **transferred** between separate parts of the TOE.

Application Note: This requirement ensures all communications between components of a distributed TOE are protected through the use of an encrypted communications channel. The data passed in this trusted communication channel are encrypted as defined in the protocol chosen in the second selection.

The trusted channel uses secure protocols that preserve the confidentiality and integrity of MDM communications. The ST author chooses the mechanism or mechanisms supported by the TOE. To support mutual authentication FIA_X509_EXT.1/CERTVAL_SEL should be included in the ST. This channel may also be used as the registration channel for the registration process, as described in section 3.1 and FCO_CPC_EXT.1.2 .

If "IPsec as defined in the PP-Module for VPN Client" is selected, the TSF must claim conformance to a PP-Configuration that includes the VPN Client PP-Module.

If the ST author selects "SSH as defined in the Functional Package for Secure Shell," the TSF must be validated against the FP for Secure Shell with the MDM PP. It should be noted that due to constraints imposed by this PP that sha1 cannot be used.

If the ST author selects "mutually authenticated TLS as defined in the Package for Transport Layer Security" or "mutually authenticated DTLS as defined in the Package for Transport Layer Security," the TSF must be validated against requirements from the Package for Transport Layer Security, with the following selections made:

- FCS_TLS_EXT.1:
  - either TLS or DTLS is selected depending on the selection made in
  - either client or server is selected as appropriate
- FCS_TLSC_EXT.1.1 or FCS_TLSS_EXT.1.1 (as appropriate):
  - The cipher suites selected must correspond with the algorithms and hash functions allowed in FCS_COP.1.
  - mutual authentication must be selected

Protocol, RBG, Certificate validation, algorithm, and similar services may be met with platform-provided services.

This requirement is claimed if "FPT_ITT.1/INTER_XFER" is selected in FTP_ITC_EXT.1.

If HTTPS is chosen, FCS_HTTPS_EXT.1 must be included in the ST.

Evaluation Activities

FPT_ITT.1/INTER_XFER
TSS
The evaluator shall examine the TSS to determine that the methods and protocols used to protect distributed TOE components are described. The evaluator shall also confirm that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance

The evaluator shall confirm that the operational guidance contains instructions for establishing the communication paths for each supported method.

Tests

- *Test FPT_ITT.1/INTER_XFER:1: The evaluator shall ensure that communications using each specified (in the operational guidance) communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.*
- *Test FPT_ITT.1/INTER_XFER:2: The evaluator shall ensure, for each method of communication, the channel data is not sent in plaintext.*

FPT_ITT.1/INTER_XFER_AGENT Internal TOE TSF Data Transfer (MDM Agent)

The inclusion of this selection-based component depends upon selection in FTP_ITC_EXT.1.1.

FPT_ITT.1.1/INTER_XFER_AGENT

The TSF shall[**selection**:

- ***invoke platform-provided functionality to use[selection: mutually authenticated TLS, mutually authenticated DTLS, HTTPS]***
- ***implement functionality using[selection: mutually authenticated TLS as defined in the Package for Transport Layer Security, mutually authenticated DTLS as defined in the Package for Transport Layer Security, HTTPS in accordance with FCS_HTTPS_EXT.1]***

]**to** protect all data from [*disclosure and modification*] when it is **transferred** between **the TSF and MDM agent.**

Application Note: This requirement ensures all communications between the TSF and MDM agent are protected through the use of an encrypted communications channel. The data passed in this trusted communication channel are encrypted as defined in the protocol chosen in the second selection.

The trusted channel uses secure protocols that preserve the confidentiality and integrity of MDM communications. The ST author chooses the mechanism or mechanisms supported by the TOE. To support mutual authentication FIA_X509_EXT.1/CERTVAL_MAN should be included in the ST. This channel may also be used as the registration channel for the registration process, as described in section 3.1 and FCO_CPC_EXT.1.2 .

If the ST author selects "mutually authenticated TLS as defined in the Package for Transport Layer Security" or "mutually authenticated DTLS as defined in the Package for Transport Layer Security," the TSF must be validated against requirements from the Package for Transport Layer Security, with the following selections made:

- FCS_TLS_EXT.1:
  - either TLS or DTLS is selected depending on the selection made in FPT_ITT.1.1(2)
  - either client or server is selected as appropriate
- FCS_TLSC_EXT.1.1 or FCS_TLSS_EXT.1.1 (as appropriate):
  - The cipher suites selected must correspond with the algorithms and hash functions allowed in FCS_COP.1.
  - mutual authentication must be selected

Protocol, RBG, Certificate validation, algorithm, and similar services may be met with platform-provided services.

If HTTPS is chosen, you must include FCS_HTTPS_EXT.1 in the ST. This requirement is claimed if "FPT_ITT.1/INTER_XFER_AGENT" is selected in FTP_ITC_EXT.1.

If HTTPS is chosen, FCS_HTTPS_EXT.1 must be included in the ST.

Evaluation Activities

FPT_ITT.1/INTER_XFER_AGENT
TSS
The evaluator shall examine the TSS to determine that the methods and protocols used to protect distributed TOE components are described. The evaluator shall also confirm that all

protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance
The evaluator shall confirm that the operational guidance contains instructions for establishing the communication paths for each supported method.

Tests
- *Test FPT_ITT.1/INTER_XFER_AGENT:1: The evaluator shall ensure that communications using each specified (in the operational guidance) communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.*
- *Test FPT_ITT.1/INTER_XFER_AGENT:2: The evaluator shall ensure, for each method of communication, the channel data is not sent in plaintext.*

B.7 Class: Trusted Path/Channels (FTP)

FTP_ITC.1/INTER_TSF_XFER_AGENT Inter-TSF Trusted Channel (MDM agent)

The inclusion of this selection-based component depends upon selection in FTP_ITC_EXT.1.1.

FTP_ITC.1.1/INTER_TSF_XFER_AGENT

The TSF shall[**selection**:
- ***invoke platform-provided functionality to use[selection: mutually authenticated TLS, mutually authenticated DTLS, HTTPS]***
- ***implement functionality using[selection: mutually authenticated TLS as defined in the Package for Transport Layer Security, mutually authenticated DTLS as defined in the Package for Transport Layer Security, HTTPS in accordance with FCS_HTTPS_EXT.1]***

]**to** provide a **trusted** communication channel between itself **(as a server) and the MDM agent** that is logically distinct from other communication channels, provides assured identification of its endpoints, **protects channel data from disclosure, and detects modification of the channel data.**

Application Note: The intent of the mandatory portion of the above requirement is to use the cryptographic protocols identified in the requirement to establish and maintain a trusted channel between the TOE and the MDM agent. If the TOE includes a separate MAS Server, this requirement also addresses the communication between the MAS Server and the MDM agent. Only TLS, DTLS, or HTTPS are used in this trusted channel.

This requirement is to ensure that the transmission of any audit logs, mobile device information data (software version, hardware model, and application versions), and configuration data collected by the MDM agent and sent from the MDM agent to the MDM Server, when commanded, or at configurable intervals, is properly protected. This trusted channel also protects any commands and policies sent by the MDM server to the MDM agent. Either the MDM agent or the MDM server is able to initiate the connection.

For TLS connections between the MDM server and agent, the MDM server is the Server side of the TLS connection, therefore it is appropriate to include the selection-based FCS_TLSS SFRs in the ST, not FCS_TLSC SFRs. With respect to mutual authentication, in cases where the agent is outside of the TOE, it should be verified that the server can support mutual authentication, meaning that the server includes support for client-side certificates for TLS mutual authentication post-enrollment. However, the client side is not evaluated since the agent is not in the TOE.

This trusted channel protects the connection between an enrolled MDM agent and the MDM server. FTP_TRP.1/TRUSTPATH_ENROLL provides a trusted channel to protect the connection between an unenrolled MDM agent and the MDM server during the enrollment operation.

The trusted channel uses TLS, DTLS, or HTTPS as the protocol that preserves the confidentiality and integrity of MDM communications. The ST author chooses the mechanism or mechanisms supported by the TOE.

If the ST author selects "mutually authenticated TLS as defined in the Package for Transport Layer Security" or "mutually authenticated DTLS as defined in the Package for Transport Layer Security," the TSF must be validated against requirements from the Package for Transport Layer Security, with the following selections made:

- FCS_TLS_EXT.1:
  - either TLS or DTLS is selected depending on the selection made in FTP_ITC.1.1/INTER_TSF_XFER_AGENT
  - server must be selected
- FCS_TLSS_EXT.1.1:
  - The cipher suites selected must correspond with the algorithms and hash functions allowed in FCS_COP.1.
  - mutual authentication must be selected

Protocol, RBG, Certificate validation, algorithm, and similar services may be met with platform-provided services.

The requirement implies that not only are communications protected when they are initially established, but also on resumption after an outage. It may be the case that some part of the TOE setup involves manually setting up tunnels to protect other communication, and if after an outage the TOE attempts to re-establish the communication automatically with (the necessary) manual intervention, there may be a window created where an attacker might be able to gain critical information or compromise a connection.

This requirement is claimed if "FTP_ITC.1/INTER_TSF_XFER_AGENT" is selected in FTP_ITC_EXT.1.

If HTTPS is chosen, FCS_HTTPS_EXT.1 must be included in the ST.

FTP_ITC.1.2/INTER_TSF_XFER_AGENT

The TSF shall **[selection: *invoke platform-provided functionality, implement functionality*] to** permit the [*TSF and MDM agent*] to initiate communication via the trusted channel.

FTP_ITC.1.3/INTER_TSF_XFER_AGENT

The TSF shall **[selection: *invoke platform-provided functionality, implement functionality*]to** initiate communication via the trusted channel for [*all communication between the TSF and the MDM agent*]

Evaluation Activities

FTP_ITC.1/INTER_TSF_XFER_AGENT

TSS

The evaluator shall examine the TSS to determine that the methods of agent-server communication are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of remote TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If "invoke platform-provided functionality" is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Guidance

The evaluator shall confirm that the operational guidance contains instructions for configuring the communication channel between the MDM agent and the MDM server for each supported method.

Tests

- *Test FTP_ITC.1/INTER_TSF_XFER_AGENT:1: The evaluator shall ensure that communications using each specified (in the operational guidance) agent-server communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.*
- *Test FTP_ITC.1/INTER_TSF_XFER_AGENT:2: The evaluator shall ensure, for each method of agent-server communication, the channel data is not sent in plaintext.*

- *Test FTP_ITC.1/INTER_TSF_XFER_AGENT:3: The evaluator shall ensure, for each communication channel with the MDM server, that a protocol analyzer identifies the traffic as the protocol under testing.*

Appendix C - Extended Component Definitions

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

C.1 Extended Components Table

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

Table 6: 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

C.2 Extended Component Definitions

C.2.1 Class: Communication (FCO)

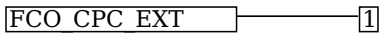
This PP defines the following extended components as part of the FCO class originally defined by CC Part 2:

C.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.

C.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:

C.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

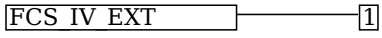
The TSF shall implement HTTPS using TLS in accordance with the Package for Transport Layer Security.

C.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

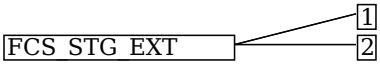
The TSF shall[**selection**: *invoke platform-provided functionality, implement functionality*]to generate IVs in accordance withTable 5 .

C.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*].

C.2.3 Class: Identification and Authentication (FIA)

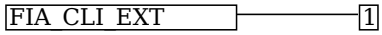
This PP defines the following extended components as part of the FIA class originally defined by CC Part 2:

C.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

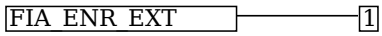
The TSF shall require a unique[**selection**: *certificate*, *token as defined in FIA_TOK_EXT.1*]for each client device.

C.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

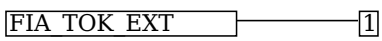
The TSF shall limit the user's enrollment of devices to devices specified by[**selection**: *IMEI*, *[**assignment**: a unique device ID]*]and[**selection**: *specific device models*, *a number of devices*, *specific time period*, *[**assignment**: other features]*, *no other features*].

C.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.

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

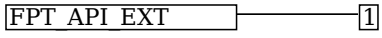
This PP defines the following extended components as part of the FPT class originally defined by CC Part 2:

C.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

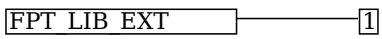
The TSF shall use only documented platform APIs.

C.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

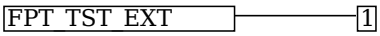
The MDM software shall be packaged with only[**assignment**: *list of third-party libraries*].

C.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

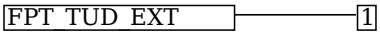
The TSF shall[**selection**: *invoke platform-provided functionality*, *implement functionality*]to provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the[**selection**: *TSF*, *TOE platform*]-provided cryptographic services.

C.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.

C.2.5 Class: Security Audit (FAU)

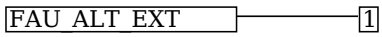
This PP defines the following extended components as part of the FAU class originally defined by CC Part 2:

C.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:

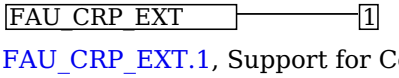
- Change in enrollment status
- Failure to apply policies to a mobile device
- [**selection**: ***[assignment**: other events], no other events*]

C.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:

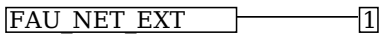
- The current version of the MD firmware or software
- The current version of the hardware model of the device
- The current version of installed mobile applications
- List of MD configuration policies that are in place on the device (as defined in FMT_SMF.1.1/SERVER_CONF_AGENT)
- [**selection**: *[**assignment**: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

C.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.

C.2.6 Class: Security Management (FMT)

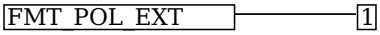
This PP defines the following extended components as part of the FMT class originally defined by CC Part 2:

C.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

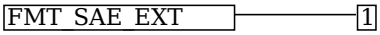
For each unique policy managed by the TSF, the TSF shall validate that the policy is appropriate for an agent using[**selection**: *client authentication via an X509 certificate representing the agent, a token issued to the agent and associated with a policy signing key uniquely associated to the policy*].

C.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.

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

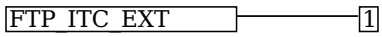
This PP defines the following extended components as part of the FTP class originally defined by CC Part 2:

C.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 D - Acronyms

Table 7: 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 E - Bibliography

Table 8: 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.