GovShield Version 1.60.05

Security Target

ST Version: 1.0
February 6, 2026

General Dynamics Mission Systems
9500 Innovation Drive
Manassas, VA 20110

Prepared By:

Booz | Allen | Hamilton
delivering results that endure

Cyber Assurance Testing Laboratory
1100 West Street
Laurel, MD 20707

Table of Contents

Table of Figures

Table of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation.

1.1.1 ST Identification

ST Title: GovShield Version 1.60.05 Security Target
ST Version: 1.0
ST Publication Date: February 6, 2026
ST Author: Booz Allen Hamilton

1.1.2 Document Organization

Chapter 1 of this document provides identifying information for the ST and TOE as well as a brief description of the TOE and its associated TOE type.

Chapter 2 describes the TOE in terms of its physical boundary, logical boundary, exclusions, and dependent Operational Environment components.

Chapter 3 describes the conformance claims made by this ST.

Chapter 4 describes the threats, assumptions, objectives, and organizational security policies that apply to the TOE.

Chapter 5 defines extended Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs).

Chapter 6 describes the SFRs that are to be implemented by the TSF.

Chapter 7 describes the SARs that will be used to evaluate the TOE.

Chapter 8 provides the TOE Summary Specification, which describes how the SFRs that are defined for the TOE are implemented by the TSF.

1.1.3 Terminology

This section defines the terminology used throughout this ST. The terminology used throughout this ST is defined in Table 1 and 2. These tables are to be used by the reader as a quick reference guide for terminology definitions.

Table 1: Customer-Specific Terminology

Term	Definition
System Administrator	The class of TOE Administrators that have complete access to a GovShield environment, including the underlying Windows Server 2022 platform.

Table 2: CC-Specific Terminology

Term	Definition
Administrator	The claimed Protection Profile defines an Administrator as the person who is responsible for management activities, including setting the policy that is applied by the enterprise on the mobile device. This TOE defines separate user roles.
Authorized Administrator	Synonymous with Administrator.
MD User	User with a mobile device (MD).
Trusted Channel	An encrypted connection between the TOE and a system in the Operational Environment.
Trusted Path	An encrypted connection between the TOE and the application an Authorized Administrator uses to manage it (web browser, terminal client, etc.).
User	In a CC context, any individual who has the ability to manage TOE functions or data.

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 3. This table is to be used by the reader as a quick reference guide for acronym definitions.

Table 3: Acronym Definitions

Acronym	Definition
CA	Certificate Authority
CC	Common Criteria
CMP	Certificate Management Protocol
CPU	Central Processing Unit
CSP	Critical Security Parameter
FQDN	Fully Qualified Domain Name
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure over a bidirectional TLS encrypted tunnel
IP	Internet Protocol
IT	Information Technology
MAS	Mobile Application Store
MD	Mobile Device
MDM	Mobile Device Management
NFC	Near-Field Communication
NIAP	National Information Assurance Partnership
OS	Operating System
OSP	Organizational Security Policy
PP	Protection Profile
QR code	Quick Response code
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function

UI	User Interface
URL	Uniform Resource Locator
USB	Universal Serial Bus
VPN	Virtual Private Network
WLAN	Wireless Local Area Network

1.1.5 Reference

[1] Protection Profile for Mobile Device Management, version 4.0 [MDMPP]
[2] Protection Profile Module for Mobile Device Management Agents, version 1.0 [AGENTMOD]
[3] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-001
[4] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-002
[5] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-003
[6] Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-004

1.2 TOE Reference

The TOE is the GovShield Version 1.60.05 comprising of the GovShield Server and one or more GovShield Clients installed on Android Mobile Devices. The minimum configuration for this evaluation is one GovShield Server, and one GovShield Client installed on an Android Mobile Device. Including additional GovShield Clients installed on multiple Android Mobile Devices as part of an operational configuration will not affect the validity of the functional claims made within this document and the Common Criteria certification.

1.3 TOE Overview

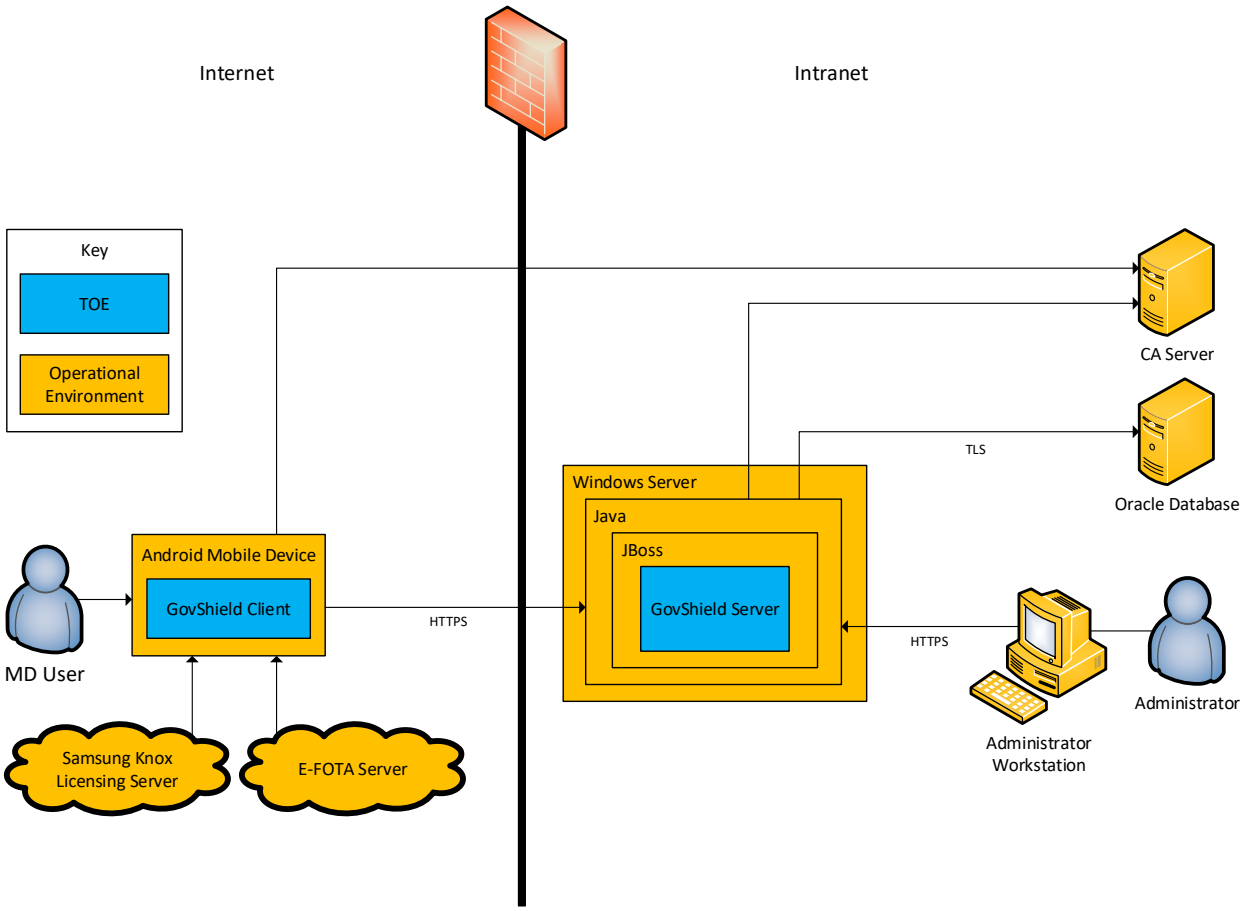
The TOE is a Mobile Device Management product and is comprised of an MDM Server component (GovShield Server) and one or more agent components called GovShield Clients. The GovShield Server component provides a centralized enterprise level management capability for a collection of Android Mobile Devices running GovShield Clients. The GovShield Server is also a Mobile Application Store (MAS) Server that allows managed devices to download apps from a trusted repository that resides within the organization managing the device. The management functionality includes management of Administrators and MD users, mobile device enrollment, mobile device status, mobile device policy management, and application management. Administrators access the GovShield Server through a browser-based web GUI on an Administrative Workstation in order to manage users, policies, and the Android Mobile Devices.

The GovShield Server runs on a Microsoft Windows Server 2022 operating system and authentication to the GovShield Server is provided locally with user data stored in the remote Oracle Database. Additionally, the GovShield Server stores its configuration and audit data within the remote Oracle Database. All communications with the Oracle Database are over TLS. The GovShield Server also

connects to a Certificate Authority (CA) Server during device enrollment so that the TOE can provide each Android Mobile Device with a unique X.509v3 certificate generated by the CA Server.

In the evaluated configuration, the GovShield Client runs on Android Mobile Devices with Android 15 operating system. The communication channel between the GovShield Client and the GovShield Server is protected by HTTPS. The GovShield Clients provide status about the Android Mobile Devices to the GovShield Server, and policy information from the GovShield Server is received to be enforced on the Android Mobile Devices. The Samsung Knox Licensing Server is used to activate the Knox platform on enrolled Android Mobile Devices. The Samsung Knox Enterprise Firmware-over-the-air (E-FOTA) Server allows the TOE Administrators to push firmware updates to one or more enrolled Android Mobile Devices. Figure 1 depicts the network configuration of the TOE.

Figure 1: TOE Boundary



1.4 TOE Type

The TOE is a Mobile Device Management product consisting of the GovShield Server and one or more GovShield Clients which run on mobile devices. The [MDMPP] states:

"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."

The MDM Server TOE type is justified because the TOE software provides centralized enterprise level management capabilities for MDM Agents (GovShield Clients) running on mobile devices, including enrollment, policy management and device status and the MDM Server (GovShield Server) runs on Microsoft Windows Server 2022, which is a general-purpose platform.

The [MDMPP] also states:

"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 Module for MDM Agents. If the MDM Agent is installed on a mobile device as an application developed by the MDM developer, the 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."

This statement is re-iterated in the [AGENTMOD]. The MDM Agent TOE type is justified because the MDM Agent software (GovShield Client) is installed on a mobile device as an application developed by General Dynamics Mission Systems (General Dynamics) and establishes a secure connection back to the MDM Server (GovShield Server) protected by HTTPS.

2 TOE Description

This section provides a description of the TOE in its evaluated configuration. This includes the physical and logical boundaries of the TOE.

2.1 Evaluated Components of the TOE

The following table describes the TOE components in the evaluated configuration:

Table 4: Evaluated Components of the TOE

Component	Definition
GovShield Server (Server)	This satisfies the MDM Server Component of the TOE as it provides an enterprise-level management capability for a collection of mobile devices, including the administration of mobile device policies, reporting on device behavior, and sending commands to the GovShield Client(s). This MDM Server Component also provides a Mobile Application Store (MAS) Server that allows managed devices to download apps from a trusted repository.
GovShield Client (Client)	This satisfies the MDM Agent Component of the TOE as it is a General Dynamics-developed application installed on mobile devices running the Samsung Android 15 operating system and uses the Android platform to establish a secure connection back to the GovShield Server for the GovShield Client to provide status and receive policy information for the device.

2.2 Components and Applications in the Operational Environment

The following table lists components and applications in the TOE's operational environment that must be present for the TOE to be operating in its evaluated configuration:

Table 5: Components of the Operational Environment

Component	Definition
Android Mobile Device	The MDM Agent Component of the TOE (GovShield Client) is an application that is installed on mobile devices running the Android 15 operating system; so that the TOE can provide management functionality to the device.
Certification Authority (CA) Server	The GovShield Server connects to the CA Server during device enrollment so that the TOE can provide each Android Mobile Device with a unique X.509v3 certificate generated by the CA Server.
E-FOTA Server	Samsung Knox Enterprise Firmware-over-the-air (E-FOTA) Server allows the TOE Administrators to push firmware updates to one or more enrolled Android Mobile Devices.
Java Runtime Environment	Java Runtime Environment is installed on the Windows Server 2022 and provides resources to Java based programs. JBoss EAP and subsequently the TOE operate on this runtime environment. In the evaluated configuration, OpenJDK 11 is used.
JBoss EAP	JBoss Enterprise Application Platform (EAP) is a Java based application server on which the TOE is installed. In the evaluated configuration, JBoss EAP 7.4 is used.
Oracle Database	The RDBMS database used to store the TOE's audit, configuration settings, and device data. In the evaluated configuration, Oracle Database 19 is used.

Samsung Knox Licensing Server	The TOE communicates with the Samsung Knox Licensing Server to verify the Knox licensing key provides to the GovShield Client on an Android Mobile Device. Once the key is verified by the Samsung Knox Licensing Server, it will activate the Android Mobile Device's Knox platform which provides the TOE access to enterprise functions of the Android Mobile Device.
Windows Server 2022	This is the OS that the GovShield Server is installed on.
Workstation	Any general-purpose computer that is used by an Administrator to manage the TOE. For the TOE to be accessed remotely, the workstation is required to have a browser to access the TOE's GUI based interface.

2.3 Excluded from the TOE

The following optional products, components, and/or applications can be integrated with the TOE but are not included in the evaluated configuration. They provide no added security related functionality for the evaluated product. They are separated into three categories: not installed, installed but requires a separate license, and installed but not part of the TSF.

2.3.1 Not Installed

There are no optional components that are omitted from the installation process.

2.3.2 Installed but Requires a Separate License

There are no excluded components that are installed and require a separate license.

2.3.3 Installed But Not Part of the TSF

This section contains functionality or components that are part of the purchased product but are not part of the TSF relevant functionality that is being evaluated as the TOE:

There are no discrete individual components that are excluded from the TSF. Note however that the logical boundary of the TOE only includes the functionality that satisfies the Security Functional Requirements in the claimed Protection Profiles. If the product provides functionality that is not used to satisfy any of these requirements, it is considered to be security-non-interfering functionality.

2.4 Physical Boundary

2.4.1 Hardware

The TOE is a software product. All hardware that is present is part of the TOE's Operational Environment.

2.4.2 Software

The GovShield Server runs on an environment containing Windows Server 2022, OpenJDK 11, and JBoss EAP 7.4. The software version of the GovShield Server is 1.60.05.

The GovShield Client runs on the Android 15 operating systems in its evaluated configuration. The software version of the GovShield Client is 1.60.05.

Refer to the GovShield Installation Guidance v1.0 for information on delivery of the TOE software.

2.5 Logical Boundary

The TOE is comprised of several security features. Each of these security features consists of several security functionalities, as identified below. This ST includes both the GovShield Server and the GovShield Client functionality.

1. Security Audit
2. Communication
3. Cryptographic Support
4. Identification and Authentication
5. Security Management
6. Protection of the TSF
7. TOE Access
8. Trusted Path/Channels

2.5.1 Security Audit

The GovShield Server component of the TOE creates audit records for auditable events related to administrative actions, configuration of the GovShield Server itself, and server-initiated management activities that affect one or more managed Android Mobile Devices. The GovShield Server's MAS Server functionality also generates audit records when it experiences a failure to push or update an application on a managed mobile device. The audit records are stored in a remote Oracle Database and transferred over a TLS encrypted trusted channel. Audit records can be viewed in the web GUI.

The GovShield Server provides Authorized Administrators with the ability to view information about enrolled mobile devices and to review alerts when various events occur. Alerts are generated based on information provided by the GovShield Client during a reachability event. The GovShield Client also generates audit records for the activities it performs as a result of its interactions with the GovShield Server.

2.5.2 Communication

The GovShield Client's Android Mobile Devices are registered with the GovShield Server for enrollment. This requires an Administrator to enable communications between these TOE components by managing a whitelisted set of Android Mobile Devices that are allowed to enroll with the GovShield Server. The enrollment process occurs over an HTTPS/TLS trusted channel that is handled by each TOE components' underlying platform. An Administrator can disable the communications between a GovShield Client and the GovShield Server by unenrolling or wiping the enrolled Android Mobile Device.

2.5.3 Cryptographic Support

Cryptographic services for the GovShield Server and the GovShield Client are provided by their respective underlying platforms. The cryptographic services include encryption/decryption services, credential/key storage, key establishment, key destruction, hashing services, signature services, and hashed message authentication. All cryptographic services use the respective TOE components' platform provided DRBG functionality to support their cryptographic operations.

The GovShield Server invokes the Java Runtime Environment Platform and its BSafe cryptographic library for cryptographic services to establish TLS and HTTPS/TLS trusted channels and paths to ensure secure communications of data in transit. The MAS Server is integrated with the GovShield Server, so it invokes the same cryptography services. The GovShield Server also invokes its platform to digitally sign policies sent to the GovShield Clients.

The GovShield Client invokes its underlying Android Mobile Device platform's BoringSSL cryptographic module for cryptographic services to also establish trusted communications. The GovShield Client also invokes its underlying platform to verify the digital signatures of all policies received from the GovShield Server.

2.5.4 Identification and Authentication

The GovShield Client registers with the GovShield Server so that the Android Mobile Device can be enrolled into management by the GovShield Server. This is accomplished by MD user or Administrator installing the GovShield Client software using a QR code, and then entering their authentication credentials through the GovShield Client interface. This will initiate the enrollment connection to the GovShield Server. The GovShield Server will authenticate the user as well as verify that the Android Mobile Device is allowed to enroll based upon being included within a whitelisted set of Android Mobile Devices. The GovShield Server will then send the GovShield Client its initial payload to include a unique X.509v3 certificate for trusted communications and the public key to verify the signature of policies. The GovShield Client then downloads its assigned policy from the GovShield Server to complete the enrollment process.

Administrators (through the web GUI and GovShield Client) and MD users (through the GovShield Client) cannot access the GovShield Server without being authenticated. Administrators and MD users can view the configurable consent warning banner prior to authentication via their respective interfaces.

The GovShield Server relies on the underlying platform to provide X.509v3 certificate services for signing policies that are sent to GovShield Client. The GovShield Server also relies on its platform to provide its X.509v3 certificate as part of TLS and HTTPS/TLS session establishment in all cases where the GovShield Server needs to provide an X.509v3 certificate. The GovShield Client relies on the underlying platform's cryptographic modules to provide X.509v3 certificate services for signature verification for signed policies sent from the GovShield Server, and in support of HTTPS/TLS connections from the GovShield Client to the GovShield Server.

2.5.5 Security Management

Authorized Administrators manage the TOE through the GovShield Server's web GUI which provides the ability to manage users, policies, and the Android Mobile Devices. An Administrator or MD user initiates the installation of the TOE's GovShield Client on the Android Mobile Device; which will communicate with the GovShield Server to enroll in management. Once enrolled, the TOE will prevent user-directed unenrollment from management.

The GovShield Server can be used to transmit specific commands to an Android Mobile Device such as forcibly locking the device or initiating a wipe operation. The GovShield Server can also define policies that specify the configuration settings for an Android Mobile Device. These configuration settings can include functionality such as configuration of the password policy and what settings are applied to Wi-Fi

connections. The GovShield Server transmits commands and policies to the GovShield Client for enforcement on the Android Mobile Device. The GovShield Server invokes its underlying platform to sign all policy data and the GovShield Agent invokes its underlying platform to validate the signed policies when they are received.

2.5.6 Protection of the TSF

The communications between the GovShield Server and GovShield Client is protected using HTTPS/TLS which is provided by the underlying platforms of the TOE components.

The GovShield Server relies on its platform to perform the self-test functionality. This includes the platform performing self-tests to verify the integrity and operation of the operational environment components (operating system, hardware and cryptographic module), and the integrity of stored TSF executable code against known SHA-256 checksums.

The Administrator invokes the GovShield Server's underlying platform to update the GovShield Server software, and the platform will verify the GovShield Server's software digital signature as part of the installation process. The Administrator distributes updates to the GovShield Client software through the GovShield Server's MAS Server functionality, and the GovShield Client will invoke the Android Mobile Device to verify the software update's digital signature before installing it. The TOE components' software contains third-party libraries. The TOE components use only documented APIs from their underlying platforms.

2.5.7 TOE Access

The GovShield Server displays a configurable consent warning banner on the web GUI's login page. The GovShield Client displays a configurable consent warning banner on the GovShield Client's login page.

2.5.8 Trusted Path/Channels

The trusted communication channels between the GovShield Server and the GovShield Client, and the GovShield Client and the Oracle Database, make use of HTTPS/TLS and TLS respectively. The trusted communication channels are provided by the TOE components' underlying platforms.

The GovShield Server platform uses HTTPS/TLS to provide a trusted path between itself and remote Administrators through the web GUI and mobile device users during the enrollment of a GovShield Client on an Android Mobile Device.

3 Conformance Claims

3.1 CC Version

This ST is compliant with Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 5, April 2017.

3.2 CC Part 2 Conformance Claims

This ST and Target of Evaluation (TOE) are Part 2 (extended) to include all applicable NIAP and International interpretations through February 6, 2026.

3.3 CC Part 3 Conformance Claims

This ST and Target of Evaluation (TOE) are Part 3 (conformant) to include all applicable NIAP and International interpretations through February 6, 2026.

3.4 PP Claims

This ST claims exact conformance to the following Protection Profiles:

- Protection Profile for Mobile Device Management, version 4.0 [MDMPP]
- Protection Profile Module for Mobile Device Management Agents, version 1.0 [AGENTMOD]

The TOE claims exact compliance to the Protection Profile for Mobile Device Management, version 4.0 and Protection Profile Module for Mobile Device Management Agents, version 1.0, which are conformant with CC Part 3.1 Release 5.

The TOE claims the following Selection-Based SFRs that are defined in the appendices of the claimed PP:

[MDMPP]:

- FAU_GEN.1(2)
- FMT_MOF.1(3)
- FMT_SMF.1(3)
- FMT_SMR.1(2)
- FPT_ITT.1(2)

The TOE claims the following Optional SFRs that are defined in the appendices of the claimed PP:

[MDMPP]:

- FAU_SAR.1
- FTA_TAB.1

The TOE claims the following Objective SFRs that are defined in the appendices of the claimed PP:

[MDMPP]:

- FCO_CPC_EXT.1

This does not violate the notion of exact conformance because the PPs specifically indicate these as allowable selections, options, and objectives, and provides both the ST author and evaluation laboratory with instructions on how these claims are to be documented and evaluated.

3.5 Package Claims

This ST does not claim conformance to any Functional Packages.

3.6 Package Name Conformant or Package Name Augmented

This ST does not claim conformance to any Functional Packages.

3.7 PP-Configuration Conformance Claims

The PP-Configuration for Mobile Device Management (MDM) and MDM Agents, Version 1.0, January 27, 2020. [CFG_MDM-MDM_AGENT_V1.0] includes the following components:

- Base-PP: Protection Profile for Mobile Device Management, Version 4.0 (PP_MDM_V4.0)
- PP-Module: PP-Module for MDM Agents, Version 1.0 (MOD_MDM_AGENT_V1.0)

Conformance Statement:

This PP-Configuration, and its components specified are conformant to Parts 2 (extended) and 3 (conformant) of Common Criteria Version 3.1, Release 5.

CC Conformance Claims:

This ST and Target of Evaluation (TOE) claims exact compliance to the PP Configuration [CFG_MDM-MDM_AGENT_V1.0] which is conformant with CC Part 3.1 Release 5.

SAR Statement:

The SARs specified for this PP-Configuration are taken from, and identical to, those specified in the MDM Base-PP (PP_MDM_V4.0).

3.8 Conformance Claim Rationale

The [MDMPP] states the following:

“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”.

The [AGENTMOD] states the following:

“The MDM system consists of two primary components: the MDM Server software and the MDM Agent. This PP-Module specifically addresses the MDM Agent. The MDM Agent establishes a secure connection back to the MDM Server, from which it receives policies to enforce on the mobile device. Optionally, the

MDM Agent interacts with the Mobile Application Store (MAS) Server to download and install enterprise-hosted applications.

A compliant MDM Agent is installed on a mobile device as an application (supplied by the developer of the MDM Server software) or is part of the mobile device's OS. This PP-Module builds on either the MDF PP or the MDM PP. A TOE that claims conformance to this PP-Module must also claim conformance to one of those PPs as its Base-PP. A compliant TOE is obligated to implement the functionality required in the Base-PP along with the additional functionality defined in this PP-Module in order to mitigate the threats that are defined by this PP-Module.

This PP-Module shall build on the MDF PP if the TOE is a native part of a mobile operating system. The TOE for this PP-Module combined with the MDF PP is the mobile device itself plus the MDM Agent. If the MDM Agent is part of the mobile device's OS, the MDM Agent may present multiple interfaces for configuring the mobile device, such as a local interface and a remote interface. Agents conforming to this PP-Module must at least offer an interface with a trusted channel that serves as one piece of an MDM system. Conformant MDM Agents may also offer other interfaces, and the configuration aspects of these additional interfaces are in scope of this PP-Module.

This PP-Module shall build on the MDM Server PP if the TOE is a third-party application that is provided with an MDM Server and installed on a mobile device by the user after acquiring the mobile device. The distributed TOE for this PP-Module combined with the MDM Server PP is the entire MDM environment, which includes both the MDM Server and the MDM Agent. Even though the mobile device itself is not part of the TOE, it is expected to be evaluated against the MDF PP so that its baseline security capabilities can be assumed to be present."

The MDM Server component (GovShield Server) of the TOE is designed to provide centralized management capabilities of the MDM Agent component (GovShield Client) of the TOE which reside on mobile devices. The GovShield Client communicates with the GovShield Server over a secure channel. Therefore, the conformance claim is appropriate.

3.9 Technical Decisions

The following is the list of NIAP Technical Decisions that are applicable to the ST/TOE and a summary of their impact:

TD #	Title	References	Changes			Analysis to this evaluation	
			SFR	AA	Notes	NA	Reason
TD0438	TST and TUD on the MDM Agent	[MDMPP] FPT_TST_EXT.1 and FPT_TUD_EXT.1.1	X		X		Clarifying distributed TOE components Footnote 15
TD0461	Security Audit for Distributed TOEs	[MDMPP] Section 6.2.2, Bullet 2			X		Clarify audit transfer on distributed TOE
TD0462	MDM Distributed TOE: Registration Channel Updates	[MDMPP] Section 3.1 and FCO_CPC_EXT.1	X		X		Update registration channel selection for distributed TOE Footnote 3, 4

TD0479	FMT_SMF.1(1) Reliance on MDF Evals	[MDMPP] FMT_SMF.1(1)		X	X		TOE claims more functions than evaluation
TD0497	SFR Rationale, Consistency of SPD, and Implicitly Satisfied SFRs	[AGENTMOD] Section 5, Section 6, and Appendix H			X		Clarifies SFR rationale, consistency of the security problem definition, and implicitly satisfied SFRs
TD0552	SFR Rationale and Implicitly Satisfied SFRs	[MDMPP] Section 6 and Appendix I			X		Clarifies SFR rationale and implicitly satisfied SFRs
TD0594	Distributed TOE tests in FCO_CPC_EXT.1.3	[MDMPP] FCO_CPC_EXT.1.3		X			Clarifies tests for a claimed SFR
TD0600	Conformance claim sections updated to allow for MOD_VPNC_V2.3	[AGENTMOD] Section 2 and [MDMPP] Section 2			X	X	PP-Module for VPN Clients, Version 2.3 was not included as a conformance claim for this evaluation
TD0616	MDM PP Use Case Mappings	[MDMPP] Appendix G			X		Fixes references
TD0629	Audit Events for Startup and Shutdown	[MDMPP] FAU_GEN.1.1(1)	X		X		Updates start up and shut down audit events to a selection Footnote 1
TD0641	Alternative revocation checking for MDM	[MDMPP] FIA_X509_EXT.1(1), FIA_X509_EXT.1(2), and FIA_X509_EXT.2	X	X	X		Updates to revocation checking and validation of ECC certificates Footnote 7
TD0650	Conformance claim sections updated to allow for MOD_VPNC_V2.3 and 2.4	[AGENTMOD] Section 2 and [MDMPP] Section 2			X	X	PP-Module for VPN Clients, Versions 2.3 and 2.4 were not included as conformance claims for this evaluation
TD0660	Mislabeled SFRs in MDM Agent Auditable Events Table	[AGENTMOD] FAU_GEN.1(2)	X				Updates mislabeled SFRs in its auditable events table Footnote 2, 3
TD0673	MDM-Agent PP-Module updated to allow for new PP and PP-Module Versions	[AGENTMOD] Section 1.1, Section 2			X	X	Added Base-PPs and PP-Modules were not selected for this evaluation.
TD0754	MDM Policy Authenticity	[MDMPP] FMT_POL_EXT.1, FIA_CLI_EXT.1, FIA_TOK_EXT.1, FIA_X509_EXT.5	X	X	X		Updates SFRs, application notes, and assurance activities for the referenced SFRs. Footnote 8, 9, 10
TD0755	MDM-Agent Policy Authenticity	[AGENTMOD] FMT_POL_EXT.2, FCS_STG_EXT.4, FMT_SMF_EXT.4, MOD MDM AGENT V1.0-SD	X	X	X		Updates SFRs, application notes, and assurance activities for the referenced SFRs. Footnote 11, 12, 14

TD0784	Terminology Change in MDMPP: Extended to Functional Package	[MDMPP] Common Criteria Terms, Conformance Claims, FIA_X509_EXT.2.1, FTP_ITC.1.1(1), FTP_TRP.1.1(1), FTP_TRP.1.3(1), FPT_ITT.1.1(1)	X		X		Updates terms, conformance claims, SFRs and application notes for the referenced SFRs. Note: All changes were selections not applicable to the TOE.
TD0844	Addition of Assurance Package for Flaw Remediation V1.0 Conformance Claim	[MDMPP] Conformance Claims			X		Added optional conformance claims that were not selected for this evaluation
TD0887	Management of x509 certificates for cloud.	[MDMPP] FMT_SMF.1.1(2)	X		X		Updates SFR and application note. Footnote 13
TD0895	Third Party Libraries in FPT_LIB_EXT.1.1	[MDMPP] FPT_LIB_EXT.1.1		X	X		Updated TSS and application note.
TD0914	Addition of PKG_TLS_V2.0 to Conformance Claims	[MDMPP] Section 2			X	X	TLS 2.0 Functional Package is optional. Not claiming TLS
TD0922	Clarification to Application Note for FCS_RBG_EXT.1.2	[MDMPP] FCS_RBG_EXT.1.2			X		Application note update
TD0935	Clarification when CTR_DRBG is Selected for FCS_RBG_EXT.1.2 in PP_MDM_V4.0	[MDMPP] FCS_RBG_EXT.1.2	X				Updated a selection for SFR. This addition option was not selected.
TD0951	Adding FIPS 186-5 in PP_MDM_V4.0	[MDMPP] FCS_CKM.1.1 FCS_COP.1(3)	X	X	X		Updated option for 186-5 or 186-4 and Testing wording Footnote 5, 6

Note that Technical Decisions were not considered to be applicable if any of the following conditions were true:

- The Technical Decision does not apply to the [MDMPP] or [AGENTMOD].
- The Technical Decision applies to an SFR that was not claimed by the TOE.
- The Technical Decision was superseded by a more recent Technical Decision.
- The Technical Decision is issued as guidance for future versions of the [MDMPP] or [AGENTMOD].

4 Security Problem Definition

4.1 Threats

Note: Unless otherwise stated Threats, Organizational Security Policies (OSPs), Assumptions and Security Objectives apply to both the Agent and Server.

This section identifies the threats against the TOE. These threats have been taken from the [MDMPP] and [AGENTMOD].

Table 6: TOE Threats

Threat	Threat Definition
T.MALICIOUS_APPS	[MDMPP] Malicious or flawed application threats exist because apps loaded onto a mobile device may include malicious or exploitable code. An administrator of the MDM or mobile device user may inadvertently import malicious code, or an attacker may insert malicious code into the TOE, resulting in the compromise of TOE or TOE data.
T.BACKUP	[AGENTMOD] An attacker may try to target backups of data or credentials and exfiltrate data. Since the backup is stored on either a personal computer or end user's backup repository, it's not likely the enterprise would detect compromise.
T.NETWORK_ATTACK	[MDMPP] An attacker may masquerade as an MDM Server and attempt to compromise the integrity of the mobile device by sending malicious management commands.
T.NETWORK_EAVESDROP	[MDMPP] An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may monitor and gain access to data exchanged between the MDM Server and other endpoints.
T.PHYSICAL_ACCESS	[MDMPP] The mobile device may be lost or stolen, and an unauthorized individual may attempt to access user data. Although these attacks are primarily directed against the mobile device platform, the TOE configures features, which address these threats.

4.2 Organizational Security Policies

This section identifies the organizational security policies which are expected to be implemented by an organization that deploys the TOE. These policies have been taken from the [MDMPP] and [AGENTMOD].

Table 7: TOE Organizational Security Policies

Policy	Policy Definition
P.ACCOUNTABILITY	Personnel operating the TOE shall be accountable for their actions within the TOE.
P.ADMIN	The configuration of the mobile device security functions must adhere to the Enterprise security policy.
P.DEVICE_ENROLL	A mobile device must be enrolled for a specific user by the administrator of the MDM prior to being used in the Enterprise network by the user.
P.NOTIFY	The mobile user must immediately notify the administrator if a mobile device is lost or stolen so that the administrator may apply remediation actions via the MDM system.

4.3 Assumptions

The specific conditions listed in this section are assumed to exist in the TOE's Operational Environment. These assumptions have been taken from the [MDMPP] and [AGENTMOD].

Table 8: TOE Assumptions

Assumption	Assumption Definition
A.COMPONENTS_RUNNING (applies to distributed TOEs only)	[MDMPP] For distributed TOEs it is assumed that the availability of all TOE components is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. It is also assumed that in addition to the availability of all components it is also checked as appropriate that the audit functionality is running properly on all TOE components.
A.CONNECTIVITY	The TOE relies on network connectivity to carry out its management activities. The TOE will robustly handle instances when connectivity is unavailable or unreliable.
A.MDM_SERVER_PLATFORM	[MDMPP] The MDM Server relies upon a trustworthy platform and local network from which it provides administrative capabilities. The MDM Server relies on this platform to provide a range of security-related services including reliable timestamps, user and group account management, logon and logout services via a local or network directory service, remote access control, and audit log management services to include offloading of audit logs to other servers. The platform is expected to be configured specifically to provide MDM services, employing features such as a host-based firewall, which limits its network role to providing MDM functionality.
A.MOBILE_DEVICE_PLATFORM	[AGENTMOD] The MDM Agent relies upon mobile platform and hardware evaluated against the MDF PP and assured to provide policy enforcement as well as cryptographic services and data protection. The mobile platform provides trusted updates and software integrity verification of the MDM Agent.
A.PROPER_ADMIN	One or more competent, trusted personnel who are not careless, willfully negligent, or hostile, are assigned and authorized as the TOE Administrators, and do so using and abiding by guidance documentation.
A.PROPER_USER	Mobile device users are not willfully negligent or hostile, and use the device within compliance of a reasonable Enterprise security policy.

4.4 Security Objectives

This section identifies the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its environment in meeting the security needs.

4.4.1 TOE Security Objectives

This section identifies the security objectives of the TOE. These objectives have been taken directly from the [MDMPP] and [AGENTMOD].

Table 9: TOE Objectives

Objective	Objective Definition
O.ACCOUNTABILITY	The TOE must provide logging facilities, which record management actions undertaken by its administrators.
O.APPLY_POLICY	The TOE must facilitate configuration and enforcement of enterprise security policies on mobile devices via interaction with the mobile OS and the MDM Server. This will include the initial enrollment of the device into management, through its entire lifecycle, including policy updates and its possible unenrollment from management services.
O.DATA_PROTECTION_TRANSIT	Data exchanged between the MDM Server and the MDM Agent must be protected from being monitored, accessed, or altered.

O.INTEGRITY	[MDMPP] The TOE will provide the capability to perform self tests to ensure the integrity of critical functionality, software, firmware, and data has been maintained. The TOE will also provide a means to verify the integrity of downloaded updates.
O.MANAGEMENT	[MDMPP] The TOE provides access controls around its management functionality.
O.QUALITY	[MDMPP] To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs.
O.STORAGE	[AGENTMOD] To address the issue of loss of confidentiality of user data in the event of loss of a mobile device (T.PHYSICAL), conformant TOEs will use platform provide key storage. The TOE is expected to protect its persistent secrets and private keys.

4.4.2 Security Objectives for the Operational Environment

The TOE's operational environment must satisfy the following objectives:

Table 10: Operational Environment Objectives

Objective	Objective Definition
OE.COMPONENTS_RUNNING	[MDMPP] For distributed TOEs the administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.
OE.PROPER_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
OE.PROPER_USER	Users of the mobile device are trained to securely use the mobile device and apply all guidance in a trusted manner.
OE.IT_ENTERPRISE	The Enterprise IT infrastructure provides security for a network that is available to the TOE and mobile devices that prevents unauthorized access.
OE.MOBILE_DEVICE_PLATFORM	[AGENTMOD] The MDM Agent relies upon the trustworthy mobile platform and hardware to provide policy enforcement as well as cryptographic services and data protection. The mobile platform provides trusted updates and software integrity verification of the MDM Agent.
OE.TIMESTAMP	[MDMPP] Reliable timestamp is provided by the operational environment for the TOE.
OE.WIRELESS_NETWORK	A wireless network will be available to the mobile devices.

4.5 Security Problem Definition Rationale

The assumptions, threats, OSPs, and objectives that are defined in this ST represent the assumptions, threats, OSPs, and objectives that are specified in the Protection Profiles to which the TOE claims conformance.

5 Extended Components Definition

5.1 Extended Security Functional Requirements

The extended Security Functional Requirements that are claimed in this ST are taken directly from the PPs to which the ST and TOE claim conformance. These extended components are formally defined in the PPs in which their usage is required.

5.2 Extended Security Assurance Requirements

There are no extended Security Assurance Requirements in this ST.

6 Security Functional Requirements

6.1 Conventions

The CC permits four functional component operations—assignment, refinement, selection, and iteration—to be performed on functional requirements. This ST will highlight the operations in the following manner:

- **Assignment:** allows the specification of an identified parameter. Indicated with *italicized text*.
- **Refinement:** allows the addition of details. Indicated with **bold text**.
- **Selection:** allows the specification of one or more elements from a list. Indicated with underlined text.
- **Iteration operation:** are identified with a number inside parentheses (e.g., "(1)").

When multiple operations are combined, such as an assignment that is provided as an option within a selection or refinement, a combination of the text formatting is used.

Text that is formatted in a claimed PP, such as if the PP's instantiation of the SFR has a refinement (bolded font), or a completed assignment (inside brackets), the formatting is not preserved when reproduced in this ST. Only the assignments and selections made by the ST author are within [brackets]. This is so that the reader can easily identify the operations that are performed by the ST author.

6.2 Security Functional Requirements Summary

The following tables list the SFRs claimed by the TOE per platform. SFRs that originate from the Mobile Device Management Protection Profile are denoted by a [MDMPP]; SFRs that originated from the Mobile Device Management Agents PP-Module are denoted by [AGENTMOD].

Table 11: Security Functional Requirements for the TOE

Class Name	Component Identification	Component Name
Security Audit	[MDMPP] FAU_ALT_EXT.1	Server Alerts
	[AGENTMOD] FAU_ALT_EXT.2	Agent Alerts
	[MDMPP] FAU_GEN.1(1)	Audit Data Generation
	[MDMPP] FAU_GEN.1(2)	Audit Generation (MAS Server)
	[AGENTMOD] FAU_GEN.1(2)	Audit Data Generation
	[MDMPP] FAU_NET_EXT.1	Network Reachability Review
	[MDMPP] FAU_SAR.1	Audit Review
	[AGENTMOD] FAU_SEL.1(2)	Security Audit Event Selection
	[MDMPP] FAU_STG_EXT.1	External Trail Storage
Communication	[MDMPP] FCO_CPC_EXT.1	Component Registration Channel Definition
Cryptographic Support	[MDMPP] FCS_CKM.1	Cryptographic Key Generation
	[MDMPP] FCS_CKM.2	Cryptographic Key Establishment
	[MDMPP] FCS_CKM_EXT.4	Cryptographic Key Destruction
	[MDMPP] FCS_COP.1(1)	Cryptographic Operation (Confidentiality Algorithms)
	[MDMPP] FCS_COP.1(2)	Cryptographic Operation (Hashing Algorithms)

Class Name	Component Identification	Component Name
	[MDMPP] FCS_COP.1(3)	Cryptographic Operation (Signature Algorithms)
	[MDMPP] FCS_COP.1(4)	Cryptographic Operation (Keyed-Hash Message Authentication)
	[MDMPP] FCS_RBG_EXT.1	Extended: Random Bit Generation
	[MDMPP] FCS_STG_EXT.1	Cryptographic Key Storage
	[AGENTMOD] FCS_STG_EXT.1(2)	Cryptographic Key Storage
Identification and Authentication	[MDMPP] FIA_ENR_EXT.1	Enrollment of Mobile Device into Management
	[AGENTMOD] FIA_ENR_EXT.2	Agent Enrollment of Mobile Device into Management
	[MDMPP] FIA_UAU.1	Timing of Authentication
	[MDMPP] FIA_X509_EXT.1(1)	X.509 Certificate Validation
	[MDMPP] FIA_X509_EXT.2	X.509 Certificate Authentication
	[MDMPP] FIA_CLI_EXT.1	Client Authorization
Security Management	[MDMPP] FMT_MOF.1(1)	Management of Functions Behavior
	[MDMPP] FMT_MOF.1(2)	Management of Functions Behavior (Enrollment)
	[MDMPP] FMT_MOF.1(3)	Management of Functions in (MAS Server Downloads)
	[MDMPP] FMT_POL_EXT.1	Trusted Policy Update
	[AGENTMOD] FMT_POL_EXT.2	Agent Trusted Policy Update
	[MDMPP] FMT_SMF.1(1)	Specification of Management Functions (Server configuration of Agent)
	[MDMPP] FMT_SMF.1(2)	Specification of Management Functions (Server Configuration of Server)
	[MDMPP] FMT_SMF.1(3)	Specification of Management Functions (MAS Server)
	[AGENTMOD] FMT_SMF_EXT.4	Specification of Management Functions
	[MDMPP] FMT_SMR.1(1)	Security Management Roles
	[MDMPP] FMT_SMR.1(2)	Security Management Roles (MAS Server)
	[AGENTMOD] FMT_UNR_EXT.1	User Unenrollment Prevention
Protection of the TSF	[MDMPP] FPT_API_EXT.1	Use of Supported Services and APIs
	[MDMPP] FPT_ITT.1(2)	Internal TOE TSF Data Transfer (MDM Agent)
	[MDMPP] FPT_LIB_EXT.1	Use of Third Party Libraries
	[MDMPP] FPT_TST_EXT.1	Functionality Testing
	[MDMPP] FPT_TUD_EXT.1	Trusted Update
TOE Access	[MDMPP] FTA_TAB.1	Default TOE Access Banners
Trusted Path/Channels	[MDMPP] FTP_ITC_EXT.1	Trusted Channel
	[MDMPP] FTP_ITC.1(1)	Inter-TSF Trusted Channel (Authorized IT Entities)
	[MDMPP] FTP_TRP.1(1)	Trusted Path (for Remote Administration)
	[MDMPP] FTP_TRP.1(2)	Trusted Path (for Enrollment)

6.3 Security Functional Requirements

6.3.1 Class FAU: Security Audit

6.3.1.1 *[MDMPP] FAU_ALT_EXT.1 Server Alerts*

FAU_ALT_EXT.1.1

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

a. Change in enrollment status
b. Failure to apply policies to a mobile device
c. [[*last time a device performed a policy reachability event with the MDM Server, failure to install an application from the MAS Server, failure to update an application from the MAS Server*]].

6.3.1.2 *[AGENTMOD] FAU_ALT_EXT.2 Agent Alerts*

FAU_ALT_EXT.2.1

The MDM Agent shall provide an alert via the trusted channel to the MDM Server in the event of any of the following audit events:

- successful application of policies to a mobile device,
- [generating] periodic reachability events,
- [
  - change in enrollment state,
  - failure to install an application from the MAS Server,
  - failure to update an application from the MAS Server,

  ].

FAU_ALT_EXT.2.2

The MDM Agent shall queue alerts if the trusted channel is not available.

6.3.1.3 *[MDMPP] FAU_GEN.1(1) Audit Data Generation*

FAU_GEN.1.1(1)[1]

The TSF shall [invoke platform-provided functionality, implement functionality] to generate an audit record of the following auditable events:

a. All administrative actions
b. [Commands issued to the MDM Agent]
c. Specifically defined auditable events listed in **Table 13**
d. [start up and shut down of the MDM system].

FAU_GEN.1.2(1)

The TSF shall record within each TSF audit record at least the following information:

[1] TD0629

- date and time of the event
- type of event
- subject identity
- (if relevant) the outcome (success or failure) of the event
- additional information in **Table 13**
- [*no other audit relevant information*].

Table 12: Server Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_ALT_EXT.1 (man)	Type of alert.	Identity of Mobile Device that sent alert.
FAU_GEN.1(1) (man)	None.	None.
FAU_GEN.1(2) (sel)	None.	None.
FAU_NET_EXT.1 (man)	None.	None.
FAU_SAR.1 (opt)	None.	None.
FAU_STG_EXT.1 (man)	None.	None.
FCO_CPC_EXT.1 (obj)	Enabling or Disabling communications between a pair of components.	Identities of the endpoints pairs enabled or disabled.
FCS_CKM_EXT.4 (man)	None.	None.
FCS_CKM.1 (man)	[None]	No additional information.
FCS_CKM.2 (man)	None.	None.
FCS_COP.1(1) (man)	None.	None.
FCS_COP.1(2) (man)	None.	None.
FCS_COP.1(3) (man)	None.	None.
FCS_COP.1(4) (man)	None.	None.
FCS_RBG_EXT.1 (man)	Failure of the randomization process.	No additional information.
FCS_STG_EXT.1 (man)	None.	None.
FIA_ENR_EXT.1 (man)	Failure of MD user authentication.	Presented username.

FIA_UAU.1 (man)	None.	None.
FIA_X509_EXT.1(1) (man)	Failure to validate X.509 certificate.	Reason for failure.
FIA_X509_EXT.2 (man)	Failure to establish connection to determine revocation status.	No additional information.
FIA_CLI_EXT.1 (man)	None.	None.
FMT_MOF.1(1) (man)	Issuance of command to perform function. Change of policy settings.	Command sent and identity of MDM Agent recipient(s). Policy changed and value or full policy.
FMT_MOF.1(2) (man)	Enrollment by a user.	Identity of user.
FMT_MOF.1(3) (sel)	None.	None.
FMT_POL_EXT.1 (man)	None.	None.
FMT_SMF.1(1) (man)	None.	None.
FMT_SMF.1(2) (man)	Success or failure of function.	No additional information.
FMT_SMF.1(3) (sel)	None.	None.
FMT_SMR.1(1) (man)	None.	None.
FMT_SMR.1(2) (sel)	None.	None.
FPT_API_EXT.1 (man)	None.	None.
FPT_ITT.1(2) (sel)	Initiation and termination of the trusted channel.	Trusted channel protocol. Identity of initiator and recipient.
FPT_LIB_EXT.1 (man)	None.	None.
FPT_TST_EXT.1 (man)	Initiation of self-test. Failure of self-test. Detected integrity violation.	Algorithm that caused failure. The TSF code file that caused the integrity violation.
FPT_TUD_EXT.1 (man)	Success or failure of signature verification.	No additional information.
FTA_TAB.1 (opt)	Change in banner setting.	No additional information.
FTP_ITC.1(1) (man)	Initiation and termination of the trusted channel.	Trusted channel protocol. Non-TOE endpoint of connection.
FTP_ITC_EXT.1 (man)	None.	None.
FTP_TRP.1(1) (man)	Initiation and termination of the trusted channel.	Trusted channel protocol. Identity of administrator.

FTP_TRP.1(2) (man)	Initiation and termination of the trusted channel.	Trusted channel protocol.

6.3.1.4 *[MDMPP] FAU_GEN.1(2) Audit Generation (MAS Server)*

FAU_GEN.1.1(2)

The MAS Server shall be able to generate an audit record 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.

FAU_GEN.1.2(2)

The [MAS Server] shall record within each TSF audit record at least the following information:

- date and time of the event
- type of event
- mobile device identity
- [*none*]

6.3.1.5 *[AGENTMOD] FAU_GEN.1(2) Audit Data Generation*

FAU_GEN.1.1(2)

The MDM Agent shall [invoke platform-provided functionality, implement functionality] to generate an MDM Agent audit record of the following auditable events:

a. Startup and shutdown of the MDM Agent;
b. All auditable events for not specified level of audit; and
c. MDM policy updated, any modification commanded by the MDM Server, specifically defined auditable events listed in **Table 14**, and [no other events].

FAU_GEN.1.2(2)

The [TSF, TOE platform] shall record within each MDM Agent audit record at least the following information:

a. Date and time of the event, type of event, subject identity, (if relevant) the outcome (success or failure) of the event, and additional information in **Table 14**; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP-Module/ST, [*none*].

Table 13: Agent Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_ALT_EXT.2	Success/failure of sending alert.	No additional information.
FAU_GEN.1(2)[2]	None.	N/A

[2] TD0660

FAU_SEL.1(2)[3]	All modifications to the audit configuration that occur while the audit collection functions are operating.	No additional information.
FCS_STG_EXT.1(2)	None.	N/A
FIA_ENR_EXT.2	Enrollment in management.	Reference identifier of MDM Server.
FMT_POL_EXT.2	Failure of policy validation.	Reason for failure of validation.
FMT_SMF_EXT.4	Outcome (Success/failure) of function.	No additional information.
FMT_UNR_EXT.1.1	[None]	No additional information.

6.3.1.6 *[MDMPP] FAU_NET_EXT.1 Network Reachability Review*

FAU_NET_EXT.1.1

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

6.3.1.7 *[MDMPP] FAU_SAR.1 Audit Review*

FAU_SAR.1.1

The TSF shall [implement functionality] to provide Authorized Administrators with the capability to read all audit data from the audit records.

FAU_SAR.1.2

The TSF shall [implement functionality] to provide the audit records in a manner suitable for the Authorized Administrators to interpret the information.

6.3.1.8 *[AGENTMOD] FAU_SEL.1(2) Security Audit Event Selection*

FAU_SEL.1.1(2)

The TSF shall [implement functionality] to select the set of events to be audited from the set of all auditable events based on the following attributes:

a. event type
b. success of auditable security events, failure of auditable security events, [*none*].

6.3.1.9 *[MDMPP] FAU_STG_EXT.1 External Trail Storage*

FAU_STG_EXT.1.1

[3] TD0660

The TSF shall be able to use a trusted channel per FTP_ITC.1(1) to transmit audit data to an external IT entity and [no other method].

6.3.2 Class FCO: Communication

6.3.2.1 *[MDMPP] FCO_CPC_EXT.1 Component Registration Channel Definition*

FCO_CPC_EXT.1.1

The TSF shall [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[4]

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

- A channel that meets the secure registration channel requirements in [FTP_TRP.1(2)],

] for at least TSF data.

FCO_CPC_EXT.1.3

The TSF shall [implement functionality] to enable an administrator to disable communications between any pair of TOE components.

6.3.3 Class FCS: Cryptographic Support

6.3.3.1 *[MDMPP] FCS_CKM.1 Cryptographic Key Generation*

FCS_CKM.1.1[5]

The TSF shall [invoke platform-provided functionality] to generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm [

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meets FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.1,
- ECC schemes using "NIST curves" P-384 and [no other curves] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2

].

6.3.3.2 *[MDMPP] FCS_CKM.2 Cryptographic Key Establishment*

FCS_CKM.2.1

[4] TD0462
[5] TD0951

The TSF shall [invoke platform-provided functionality] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1:RSA Cryptography Specifications Version 2.1",
- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography",

].

6.3.3.3 *[MDMPP] FCS_CKM_EXT.4 Cryptographic Key Destruction*

FCS_CKM_EXT.4.1

The TSF shall destroy plaintext keying material and critical security parameters by [

- invoking platform-provided functionality with the following rules:
  - For volatile memory, the destruction shall be executed by [
    - a single direct overwrite consisting of [zeroes] ,
    - destruction of reference to the key directly followed by a request for garbage collection

    ]
  - For non-volatile memory that consists of the invocation of an interface provided by the underlying platform that [
    - logically addresses the storage location of the key and performs a [single] direct overwrite consisting of [zeroes, a new value of a key] ,

    ]

].

FCS_CKM_EXT.4.2

The TSF shall destroy all plaintext keying material and critical security parameters (CSPs) when no longer needed.

6.3.3.4 *[MDMPP] FCS_COP.1(1) Cryptographic Operation (Confidentiality Algorithms)*

FCS_COP.1.1(1)

The TSF shall [invoke platform-provided functionality] to perform encryption/decryption in accordance with a specified cryptographic algorithm: [

- AES-GCM (as defined in NIST SP 800-38D),

] and cryptographic key sizes [128-bit, 256-bit].

6.3.3.5 *[MDMPP] FCS_COP.1(2) Cryptographic Operation (Hashing Algorithms)*

FCS_COP.1.1(2)

The TSF shall [invoke platform-provided functionality] to perform cryptographic hashing in accordance with a specified cryptographic algorithm [SHA-256, SHA-384] and message digest sizes [256, 384] bits that meet the following: FIPS Pub 180-4.

6.3.3.6 *[MDMPP] FCS_COP.1(3) Cryptographic Operation (Signature Algorithms)*

FCS_COP.1.1(3)[6]

The TSF shall [invoke platform-provided functionality] to perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 4

].

6.3.3.7 *[MDMPP] FCS_COP.1(4) Cryptographic Operation (Keyed-Hash Message Authentication)*

FCS_COP.1.1(4)

The TSF shall [invoke platform-provided functionality] to perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC -[SHA-256, SHA-384], key sizes [*256 bits, 384 bits*], and message digest sizes [256, 384] bits that meet the following: FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-4, "Secure Hash Standard."

6.3.3.8 *[MDMPP] FCS_RBG_EXT.1 Extended: Random Bit Generation*

FCS_RBG_EXT.1.1

The TSF shall [invoke platform-provided functionality] to perform all deterministic random bit generation services in accordance with NIST Special Publication 800-90A using [CTR_DRBG (AES)].

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by an entropy source that accumulates entropy from [a platform-based RBG] with a minimum of [256 bits] of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and hashes that it will generate.

6.3.3.9 *[MDMPP] FCS_STG_EXT.1 Cryptographic Key Storage*

FCS_STG_EXT.1.1

The TSF shall utilize [platform-provided key storage] for all persistent secrets and private keys.

[6] TD0951

6.3.3.10 *[AGENTMOD] FCS_STG_EXT.1(2) Cryptographic Key Storage*

FCS_STG_EXT.1.1(2)

The MDM Agent shall use the platform-provided key storage for all persistent secret and private keys.

6.3.4 Class FIA: Identification and Authentication

6.3.4.1 *[MDMPP] FIA_ENR_EXT.1 Enrollment of Mobile Device into Management*

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 [*[Android Device ID]*] and [no other features].

6.3.4.2 *[AGENTMOD] FIA_ENR_EXT.2 Agent Enrollment of Mobile Device into Management*

FIA_ENR_EXT.2.1

The MDM Agent shall record the reference identifier of the MDM Server during the enrollment process.

6.3.4.3 *[MDMPP] FIA_UAU.1 Timing of Authentication*

FIA_UAU.1.1

The TSF shall [implement functionality] to allow [*acceptance of the warning banner*] on behalf of the user to be performed before the user is authenticated with the Server.

FIA_UAU.1.2

The TSF shall [implement functionality] that requires each user to be successfully authenticated with the Server before allowing any other TSF-mediated actions on behalf of that user.

6.3.4.4 *[MDMPP] FIA_X509_EXT.1(1) X.509 Certificate Validation*

FIA_X509_EXT.1.1(1)[7]

The TSF shall [invoke platform-provided 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.

[7] TD0641

- 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 [OCSP as specified in RFC 6960, CRL as specified in RFC 5280 Section 6.3].
- 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.

FIA_X509_EXT.1.2(1)

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

6.3.4.5 *[MDMPP] FIA_X509_EXT.2 X.509 Certificate Authentication*

FIA_X509_EXT.2.1

The TSF shall [

- invoke platform-provided functionality to use X.509v3 certificates as defined by RFC 5280 to support authentication for [HTTPS, TLS], and [
    - policy signing

],

]

FIA_X509_EXT.2.2

When the [TOE platform] cannot establish a connection to determine the validity of a certificate, the TSF shall [invoke platform-provided functionality] to [not accept the certificate].

6.3.4.6 *[MDMPP] FIA_CLI_EXT.1 Client Authorization*

FIA_CLI_EXT.1.1[8]

The TSF shall require a unique [certificate] for each client device.

6.3.5 Class FMT: Security Management

6.3.5.1 *[MDMPP] FMT_MOF.1(1) Management of Functions Behavior*

FMT_MOF.1.1(1)

[8] TD0754

The TSF shall restrict the ability to perform the functions

- listed in FMT_SMF.1(1)
- enable, disable, and modify policies listed in FMT_SMF.1(1)
- listed in FMT_SMF.1(2)
- [enable, disable and modify policies listed in FMT_SMF.1(3)]

to authorized administrators.

6.3.5.2 *[MDMPP] FMT_MOF.1(2) Management of Functions Behavior (Enrollment)*

FMT_MOF.1.1(2)

The MDM Server shall restrict the ability to initiate the enrollment process to authorized administrators and MD users.

6.3.5.3 *[MDMPP] FMT_MOF.1(3) Management of Functions in (MAS Server Downloads)*

FMT_MOF.1.1(3)

The MAS Server shall restrict the ability to download applications, allowing only enrolled mobile devices that are compliant with MDM policies and assigned to a user in the application access group to perform this function.

6.3.5.4 *[MDMPP] FMT_POL_EXT.1 Trusted Policy Update*

FMT_POL_EXT.1.1

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

FMT_POL_EXT.1.2[9]

The TSF shall sign policies and policy updates using a private key associated with [an X509 certificate] trusted by the agent for policy verification.

FMT_POL_EXT.1.3[10]

For each unique policy managed by the TSF, the TSF shall validate that the policy is appropriate for an agent using [client authentication via an X509 certificate representing the agent].

6.3.5.5 *[AGENTMOD] FMT_POL_EXT.2 Agent Trusted Policy Update*

FMT_POL_EXT.2.1[11]

The MDM Agent shall only accept policies and policy updates that are digitally signed by a private key that has been authorized for policy updates by the MDM Server.

FMT_POL_EXT.2.2[12]

[9] TD0754
[10] TD0754
[11] TD0755
[12] TD0755

The MDM Agent shall not install policies if the signature check fails.

6.3.5.6 *[MDMPP] FMT_SMF.1(1) Specification of Management Functions (Server configuration of Agent)*

FMT_SMF.1.1(1)

The MDM Server shall be capable of communicating the following commands to the MDM Agent:

1. transition to the locked state (MDF Function 6)
2. full wipe of protected data (MDF Function 7)
3. unenroll from management
4. install policies
5. query connectivity status
6. query the current version of the MD firmware/software
7. query the current version of the hardware model of the device
8. query the current version of installed mobile applications
9. import X.509v3 certificates into the Trust Anchor Database (MDF Function 11)
10. install applications (MDF Function 16)
11. update system software (MDF Function 15)
12. remove applications (MDF Function 14)

and the following commands to the MDM Agent: [

- 13. remove Enterprise applications (MDF Function 17),
- 14. wipe Enterprise data (MDF Function 28),
- 16. alert the user,

] and the following MD configuration policies:

25. password policy:
    a. minimum password length
    b. minimum password complexity
    c. maximum password lifetime (MDF Function 1)
26. session locking policy:
    a. screen-lock enabled/disabled
    b. screen lock timeout
    c. number of authentication failures (MDF Function 2)
27. wireless networks (SSIDs) to which the MD may connect (MDF Function 2)
28. security policy for each wireless network:
    a. [
        - specify the CA(s) from which the MD will accept WLAN authentication server certificate(s),

       ]
    b. ability to specify security type
    c. ability to specify authentication protocol
    d. specify the client credentials to be used for authentication
    e. [*no other WLAN management functions*] (WLAN Client Function 1)

29. application installation policy by [
    - specifying authorized application repository(s),
    - denying application installation

    ], (MDF Function 8)

30. enable/disable policy for [*camera, microphone*] across device, [
    - no other method

    ], (MDF Function 5)

and the following MD configuration policies: [

- 32. enable/disable policy for [*cellular, NFC*], (MDF Function 4),
- 33. enable/disable policy for data signaling over [*USB, removable storage card (SD card)*], (MDF Function 24),
- 34. enable/disable policy for [*Wi-Fi tethering*], (MDF Function 25),
- 35. enable/disable policy for developer modes, (MDF Function 26),
- 36. enable policy for data-at-rest protection, (MDF Function 20),
- 37. enable policy for removable media's data-at-rest protection, (MDF Function 21),
- 38. enable/disable policy for local authentication bypass, (MDF Function 27),
- 47. the unlock banner policy, (MDF Function 36),
- 49. enable/disable [
  - USB mass storage mode,

  ] (MDF Function 39),
- 51. enable/disable [
  - Hotspot functionality authenticated by [passcode],

  ] (MDF Function 41),
- 55. enable/disable policy for use of Biometric Authentication Factor (MDF Function 23),
- 58. enable/disable automatic updates of system software,
- 60. [no other policies].

]

6.3.5.7 *[MDMPP] FMT_SMF.1(2) Specification of Management Functions (Server Configuration of Server)*

FMT_SMF.1.1(2) [13]

The TSF shall be capable of performing the following management functions:

b. configure the [
   - devices specified by [[*Android Device ID*]],

   ] and [no other features] allowed for enrollment

c. [
   0. choose X.509v3 certificates for MDM Server use
   2. configure the TOE unlock banner,
   3. configure periodicity of the following commands to the agent: [*Polling Interval, App Polling Interval, Audit Log Publishing Rate*],
   6. configure the interaction between TOE components,
   ]

[13] TD0887

6.3.5.8 *[MDMPP] FMT_SMF.1(3) Specification of Management Functions (MAS Server)*

FMT_SMF.1.1(3)

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

a. Configure application access groups
b. Download applications
c. [no other functions]

6.3.5.9 *[AGENTMOD] FMT_SMF_EXT.4 Specification of Management Functions*

FMT_SMF_EXT.4.1[14]

The MDM Agent shall be capable of interacting with the platform to perform the following functions:

- [Import the certificates to be used for authentication of MDM Agent communications],
- [administrator-provided device management functions in MDM PP]
- [no additional functions].

FMT_SMF_EXT.4.2

The MDM Agent shall be capable of performing the following functions:

- Enroll in management
- Configure whether users can unenroll from management
- [configure periodicity of reachability events].

6.3.5.10 *[MDMPP] FMT_SMR.1(1) Security Management Roles*

FMT_SMR.1.1(1)

The TSF shall maintain the roles administrator, MD user, and [no additional roles].

FMT_SMR.1.2(1)

The TSF shall be able to associate users with roles.

6.3.5.11 *[MDMPP] FMT_SMR.1(2) Security Management Roles (MAS Server)*

FMT_SMR.1.1(2)

The TSF shall additionally maintain the roles enrolled mobile devices, application access groups, and [MD *user, administrator*].

FMT_SMR.1.2(2)

The MAS Server shall be able to associate users with roles.

[14] TD0755

6.3.5.12 *[AGENTMOD] FMT_UNR_EXT.1 User Unenrollment Prevention*

FMT_UNR_EXT.1.1

The MDM Agent shall provide a mechanism to enforce the following behavior upon an attempt to unenroll the mobile device from management: [prevent the unenrollment from occurring].

6.3.6 Class FPT: Protection of the TSF

6.3.6.1 *[MDMPP] FPT_API_EXT.1 Use of Supported Services and APIs*

FPT_API_EXT.1.1

The TSF shall use only documented platform API's.

6.3.6.2 *[MDMPP] FPT_ITT.1(2) Internal TOE TSF Data Transfer (MDM Agent)*

FPT_ITT.1.1(2)

The TSF shall [

- invoke platform-provided functionality to use [
  - HTTPS

  ],

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

6.3.6.3 *[MDMPP] FPT_LIB_EXT.1 Use of Third Party Libraries*

FPT_LIB_EXT.1.1

The MDM software shall be packaged with only [*a list of third-party libraries in Appendix A*].

6.3.6.4 *[MDMPP] FPT_TST_EXT.1 Functionality 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 [invoke platform-provided 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 [TOE platform]-provided cryptographic services.

6.3.6.5 *[MDMPP] FPT_TUD_EXT.1 Trusted Update*

FPT_TUD_EXT.1.1[15]

[15] TD0438

The TSF shall provide Authorized Administrators the ability to query the current version of the software.

FPT_TUD_EXT.1.2

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

6.3.7 Class FTA: TOE Access

6.3.7.1 *[MDMPP] FTA_TAB.1* *Default TOE Access Banners*

FTA_TAB.1.1

Before establishing a user session, the TSF shall [implement functionality] to display an Administrator-specified advisory notice and consent warning message regarding use of the TOE.

6.3.8 Class FTP: Trusted Path/Channels

6.3.8.1 *[MDMPP] FTP_ITC_EXT.1* *Trusted Channel*

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,

] that is logically distinct from other communication channels, as specified in [FPT_ITT.1(2)].

6.3.8.2 *[MDMPP] FTP_ITC.1(1)* *Inter-TSF Trusted Channel (Authorized IT Entities)*

FTP_ITC.1.1(1)

The TSF shall [

- invoke platform-provided functionality to use [
  - mutually authenticated TLS,

  ],

] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [*[Oracle Database]*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2(1)

The TSF shall [invoke platform-provided functionality] to permit the MDM Server or other authorized IT entities to initiate communication via the trusted channel.

FTP_ITC.1.3(1)

The TSF shall [invoke platform-provided functionality] to initiate communication via the trusted channel for [*audit data, authentication, and configuration data*].

6.3.8.3 *[MDMPP] FTP_TRP.1(1) Trusted Path (for Remote Administration)*

FTP_TRP.1.1(1)

The TSF shall [

- invoke platform-provided functionality to use [
  - HTTPS,

  ],

] to provide a trusted communication path between itself as a [server] and remote administrators that is logically distinct from other communication paths and provides assured identification of its endpoints and protection of the communicated data from modification, disclosure.

FTP_TRP.1.2(1)

The TSF shall [invoke platform-provided functionality] to permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3(1)

The TSF shall [invoke platform-provided functionality] to require the use of the trusted path for all remote administration actions.

6.3.8.4 *[MDMPP] FTP_TRP.1(2) Trusted Path (for Enrollment)*

FTP_TRP.1.1(2)

The TSF shall [

- invoke platform-provided functionality to use [
  - HTTPS

  ],

] to provide a trusted communication path between itself (as a server) and MD users that is logically distinct from other communication paths and provides assured identification of its endpoints and protection of the communicated data from disclosure and detection of modification of the communicated data from modification, disclosure.

FTP_TRP.1.2(2)

The TSF shall [invoke platform-provided functionality] to permit MD users to initiate communication via the trusted path.

FTP_TRP.1.3(2)

The TSF shall [invoke platform-provided functionality] to require the use of the trusted path for all MD user actions.

6.4 Statement of Security Functional Requirements Consistency

The Security Functional Requirements included in the ST represent all required SFRs specified in the PPs against which exact conformance is claimed and a subset of the selection-based, optional, and objective SFRs. All hierarchical relationships, dependencies, and unfulfilled dependency rationales in the ST are considered to be identical to those that are defined in the claimed PPs.

7 Security Assurance Requirements

This section identifies the Security Assurance Requirements (SARs) that are claimed for the TOE. The SARs which are claimed are in exact conformance with the [MDMPP] and [AGENTMOD].

7.1 Class ASE: Security Target evaluation

7.1.1 ST introduction (ASE_INT.1)

7.1.1.1 *Developer action elements:*

ASE_INT.1.1D

The developer shall provide an ST introduction.

7.1.1.2 *Content and presentation elements:*

ASE_INT.1.1C

The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C

The ST reference shall uniquely identify the ST.

ASE_INT.1.3C

The TOE reference shall uniquely identify the TOE.

ASE_INT.1.4C

The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C

The TOE overview shall identify the TOE type.

ASE_INT.1.6C

The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C

The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C

The TOE description shall describe the logical scope of the TOE.

7.1.1.3 *Evaluator action elements:*

ASE_INT.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_INT.1.2E

The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

7.1.2 Conformance claims (ASE_CCL.1)

7.1.2.1 *Developer action elements:*

ASE_CCL.1.1D

The developer shall provide a conformance claim.

ASE_CCL.1.2D

The developer shall provide a conformance claim rationale

7.1.2.2 *Content and presentation elements:*

ASE_CCL.1.1C

The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C

The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C

The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C

The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C

The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C

The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C

The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C

The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C

The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C

The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

7.1.2.3 *Evaluator action elements:*

ASE_CCL.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence

7.1.3 Security objectives for the operational environment (ASE_OBJ.1)

7.1.3.1 *Developer action elements:*

ASE_OBJ.1.1D

The developer shall provide a statement of security objectives.

7.1.3.2 *Content and presentation elements:*

ASE_OBJ.1.1C

The statement of security objectives shall describe the security objectives for the operational environment.

7.1.3.3 *Evaluator action elements:*

ASE_OBJ.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.4 Extended components definition (ASE_ECD.1)

7.1.4.1 *Developer action elements:*

ASE_ECD.1.1D

The developer shall provide a statement of security requirements.

ASE_ECD.1.2D

The developer shall provide an extended components definition.

7.1.4.2 *Content and presentation elements:*

ASE_ECD.1.1C

The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C

The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C

The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C

The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C

The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

7.1.4.3 *Evaluator action elements:*

ASE_ECD.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_ECD.1.2E

The evaluator shall confirm that no extended component can be clearly expressed using existing components.

7.1.5 Stated security requirements (ASE_REQ.1)

7.1.5.1 *Developer action elements:*

ASE_REQ.1.1D

The developer shall provide a statement of security requirements.

ASE_REQ.1.2D

The developer shall provide a security requirements rationale.

7.1.5.2 *Content and presentation elements:*

ASE_REQ.1.1C

The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.1.2C

All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.1.3C

The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.1.4C

All operations shall be performed correctly.

ASE_REQ.1.5C

Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.1.6C

The statement of security requirements shall be internally consistent.

7.1.5.3 *Evaluator action elements:*

ASE_REQ.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.6 TOE summary specification (ASE_TSS.1)

7.1.6.1 *Developer action elements:*

ASE_TSS.1.1D

The developer shall provide a TOE summary specification.

7.1.6.2 *Content and presentation elements:*

ASE_TSS.1.1C

The TOE summary specification shall describe how the TOE meets each SFR.

7.1.6.3 *Evaluator action elements:*

ASE_TSS.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_TSS.1.2E

The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

7.2 Class ADV: Development

7.2.1 Basic Functional Specification (ADV_FSP.1)

7.2.1.1 *Developer action elements:*

ADV_FSP.1.1D

The developer shall provide a functional specification.

ADV_FSP.1.2D

The developer shall provide a tracing from the functional specification to the SFRs.

7.2.1.2 *Content and presentation elements:*

ADV_FSP.1.1C

The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2C

The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3C

The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4C

The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

7.2.1.3 *Evaluator action elements:*

ADV_ FSP.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_ FSP.1.2E

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

7.3 Class AGD: Guidance Documentation

7.3.1 Operational User Guidance (AGD_OPE.1)

7.3.1.1 *Developer action elements:*

AGD_OPE.1.1D

The developer shall provide operational user guidance.

7.3.1.2 *Content and presentation elements:*

AGD_OPE.1.1C

The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C

The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C

The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C

The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C

The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6C

The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C

The operational user guidance shall be clear and reasonable.

7.3.1.3 *Evaluator action elements:*

AGD_OPE.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.2 Preparative Procedures (AGD_PRE.1)

7.3.2.1 *Developer action elements:*

AGD_PRE.1.1D

The developer shall provide the TOE, including its preparative procedures.

7.3.2.2 *Content and presentation elements:*

AGD_ PRE.1.1C

The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_ PRE.1.2C

The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

7.3.2.3 *Evaluator action elements:*

AGD_ PRE.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_ PRE.1.2E

The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

7.4 Class ALC: Life Cycle Support

7.4.1 Labeling of the TOE (ALC_CMC.1)

7.4.1.1 *Developer action elements:*

ALC_CMC.1.1D

The developer shall provide the TOE and a reference for the TOE.

7.4.1.2 *Content and presentation elements:*

ALC_CMC.1.1C

The TOE shall be labeled with its unique reference.

7.4.1.3 *Evaluator action elements:*

ALC_CMC.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.4.2 TOE CM Coverage (ALC_CMS.1)

7.4.2.1 *Developer action elements:*

ALC_CMS.1.1D

The developer shall provide a configuration list for the TOE.

7.4.2.2 *Content and presentation elements:*

ALC_CMS.1.1C

The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2C

The configuration list shall uniquely identify the configuration items.

7.4.2.3 *Evaluator action elements:*

ALC_CMS.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.5 Class ATE: Tests

7.5.1 Independent Testing - Conformance (ATE_IND.1)

7.5.1.1 *Developer action elements:*

ATE_IND.1.1D

The developer shall provide the TOE for testing.

7.5.1.2 *Content and presentation elements:*

ATE_IND.1.1C

The TOE shall be suitable for testing.

7.5.1.3 *Evaluator action elements:*

ATE_IND.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2E

The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

7.6 Class AVA: Vulnerability Assessment

7.6.1 Vulnerability Survey (AVA_VAN.1)

7.6.1.1 *Developer action elements:*

AVA_VAN.1.1D

The developer shall provide the TOE for testing.

7.6.1.2 *Content and presentation elements:*

AVA_VAN.1.1C

The TOE shall be suitable for testing.

7.6.1.3 *Evaluator action elements:*

AVA_VAN.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2E

The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3E

The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

8 TOE Summary Specification

The following sections identify the security functions of the TOE and describe how the TSF meets each claimed SFR. They include Security Audit, Communication, Cryptographic Support, Identification and Authentication, Security Management, Protection of the TSF, TOE Access, and Trusted Path/Channels. The following table defines which distributed TOE component(s) perform the capabilities described by the SFR.

Table 14: SFR and TOE Component Mapping

Requirement	Server	Agent
[MDMPP] FAU_ALT_EXT.1	X	
[AGENTMOD] FAU_ALT_EXT.2		X
[MDMPP] FAU_GEN.1(1)	X	X
[MDMPP] FAU_GEN.1(2)	X	X
[AGENTMOD] FAU_GEN.1(2)		X
[MDMPP] FAU_NET_EXT.1	X	
[MDMPP] FAU_SAR.1	X	
[AGENTMOD] FAU_SEL.1(2)		X
[MDMPP] FAU_STG_EXT.1	X	X
[MDMPP] FCO_CPC_EXT.1	X	X
[MDMPP] FCS_CKM.1	X	X
[MDMPP] FCS_CKM.2	X	X
[MDMPP] FCS_CKM_EXT.4	X	X
[MDMPP] FCS_COP.1(1)	X	X
[MDMPP] FCS_COP.1(2)	X	X
[MDMPP] FCS_COP.1(3)	X	X
[MDMPP] FCS_COP.1(4)	X	X
[MDMPP] FCS_RBG_EXT.1	X	X
[MDMPP] FCS_STG_EXT.1	X	X
[AGENTMOD] FCS_STG_EXT.1(2)		X
[MDMPP] FIA_ENR_EXT.1	X	X
[AGENTMOD] FIA_ENR_EXT.2		X
[MDMPP] FIA_UAU.1	X	X
[MDMPP] FIA_X509_EXT.1(1)	X	X
[MDMPP] FIA_X509_EXT.2	X	X
[MDMPP] FIA_CLI_EXT.1	X	X
[MDMPP] FMT_MOF.1(1)	X	
[MDMPP] FMT_MOF.1(2)	X	
[MDMPP] FMT_MOF.1(3)	X	
[MDMPP] FMT_POL_EXT.1	X	
[AGENTMOD] FMT_POL_EXT.2		X
[MDMPP] FMT_SMF.1(1)	X	
[MDMPP] FMT_SMF.1(2)	X	
[MDMPP] FMT_SMF.1(3)	X	
[AGENTMOD] FMT_SMF_EXT.4		X
[MDMPP] FMT_SMR.1(1)	X	
[MDMPP] FMT_SMR.1(2)	X	

[AGENTMOD] FMT_UNR_EXT.1		X
[MDMPP] FPT_API_EXT.1	X	X
[MDMPP] FPT_ITT.1(2)	X	X
[MDMPP] FPT_LIB_EXT.1	X	X
[MDMPP] FPT_TST_EXT.1	X	[16]
[MDMPP] FPT_TUD_EXT.1	X	X
[MDMPP] FTA_TAB.1	X	X
[MDMPP] FTP_ITC_EXT.1	X	X
[MDMPP] FTP_ITC.1(1)	X	
[MDMPP] FTP_TRP.1(1)	X	
[MDMPP] FTP_TRP.1(2)	X	X

Note: SFRs that originate from the Mobile Device Management Protection Profile are denoted by a [MDMPP], and SFRs that originated from the Mobile Device Management Agent PP-Module are denoted by [AGENTMOD].

The minimum configuration for this evaluation is one Server and one Client installed on an Android device. Including additional Clients installed on multiple Android devices as part of an operational configuration will not affect the validity of the functional claims made within the TSS. All TSS descriptions regarding the role, operation, and management of a Client would be consistent with every additional Client and Android device added to the minimum evaluated configuration. Therefore, all TSS descriptions regarding the Client can be read with the understanding that the descriptions would apply to one or more of these TOE components and the method in which the additional TOE components met the SFRs would be the same as their minimum configuration equivalent.

Note: The TSS evaluation activities that apply to only the Server component are denoted by [SERVER] and to only the Client component are denoted by [AGENT]. If the TSS evaluation activity applies to both components, it is not denoted.

8.1 Security Audit

8.1.1 **[MDMPP] FAU_ALT_EXT.1**

[SERVER] The GovShield Server component of the TOE provides Authorized Administrators with the ability to view information about enrolled mobile devices and to review alerts when various events occur. Alerts are generated based on information provided by the GovShield Client during a reachability event as described under FAU_ALT_EXT.2. By navigating to the 'Device Alerts' dashboard of the web GUI, Authorized Administrators can review the following types of alerts generated based upon a reachability event from a GovShield Client:

- successful application of policies to a mobile device – success or failure of applying a policy
- periodic reachability events – the GovShield Client checking in with the GovShield Server
- change in enrollment state – initial enrollment as well as unenrollment due to wiping action received from the GovShield Server

[16] TD0438

- failure to install an application from the MAS Server – failure to install an application downloaded from the GovShield Server based upon the GovShield Client's 'APK Management' configuration
- failure to update an application from the MAS Server – failure to install an application update downloaded from the GovShield Server based upon the GovShield Client's 'APK Management' configuration

8.1.2 [AGENTMOD] FAU_ALT_EXT.2

[AGENT] The GovShield Client component of the TOE provides the ability to alert the GovShield Server in the event that certain behavior on the underlying mobile device is observed. The alert for a device becoming enrolled/unenrolled from management is sent by the GovShield Client as part of the enrollment/unenrollment process which requires communication with the GovShield Server. All other alerts are based upon policies being assigned to the mobile device. Each GovShield Client has a policy assigned, and within the policy the 'Polling Interval (mins)', 'App Polling Interval (mins)', and 'Audit Log Publishing Rate (hours)' variables have been defined for when the GovShield Client needs to perform a reachability event with the GovShield Server. These variables can be set to a minimum of 1 minute and a maximum of 1440 minutes. The 'App Polling Interval (mins)' variable is specific to initiating communication regarding the management of applications from the TOE's MAS Server functionality. The 'Audit Log Publishing Rate (hours)' variable determines when the GovShield Client will initiate connection to send its audit records to the GovShield Server. The 'Polling Interval (mins)' variable covers initiating communication for all other TOE functionality related to a GovShield Client's policy.

The GovShield Client will initiate a connection to the GovShield Server when the timing associated with either of these variables is met. Thus, when this communication is successful, this generates a reachability event that will be recorded by the GovShield Server. These reachability events result in the following activities, respective to the purpose of the reachability event:

- the GovShield Server receives information regarding the status of the GovShield Client and its underlying mobile device, and
- the GovShield Client is informed of any actions it needs to take for managing its assigned policy and the management of applications installed on the mobile device.

Depending on the actions that the GovShield Client will need to perform, there will be additional communication sessions between these TOE components to complete some actions. If additional information is needed from the GovShield Server, the GovShield Client will establish a new session with the GovShield Server to download the additional information (i.e., latest assigned policy or enterprise application). This communication is also recorded as a reachability event by the GovShield Server.

When the action is a new policy being assigned, the GovShield Client will process the downloaded policy per the process described under FMT_POL_EXT.2. When the action is an update to the list of the managed enterprise applications by the TOE, the GovShield Client will attempt to install/update the current version of the application(s) downloaded based upon the latest GovShield Client's 'APK Management' configuration. Once an action is complete (successfully or unsuccessfully), the GovShield Client will initiate a connection to the GovShield Server to provide an alert on the outcome of these actions. This communication is also recorded as a reachability event by the GovShield Server. If this connection cannot be established, the GovShield Client will queue the alert in its internal non-volatile

storage to be sent during the next relevant reachability event. The maximum amount of storage for queued alerts is equal to the storage space available on the Android Mobile Device. The only reason an alert would not be generated is because there is no remaining storage space available on the Android Mobile Device.

Since all alerts are generated based upon the initial communications between the TOE components for an enrollment action or action resulting from a polling interval, the GovShield Client will not continuously be creating alerts during a disconnected state with the GovShield Server. Thus, the GovShield Client will generate and queue alerts related to its current actions but no new actions will be initiated until the communication is re-established. This prevents storage exhaustion for queued alerts.

8.1.3 **[MDMPP] FAU_GEN.1(1)**

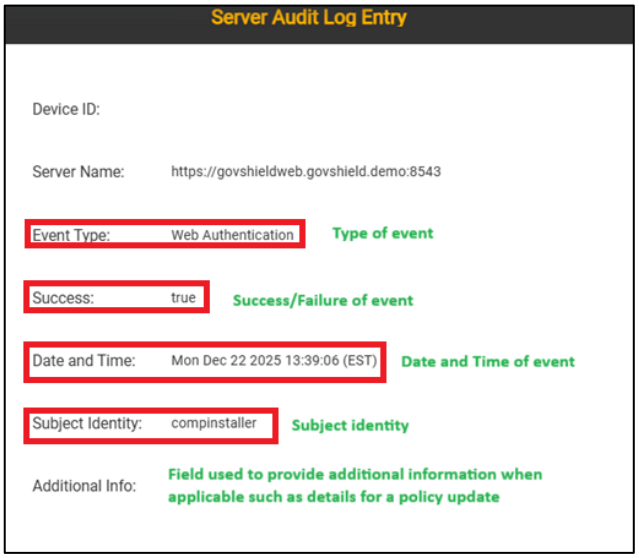
The GovShield Server and GovShield Client components of the TOE generate auditable events for their own behavior. These TOE components also rely on their underlying platforms to generate audit events. All TOE components rely on their underlying platforms to generate audit logs for their startup and shutdown. The GovShield Server generates audit logs for all administrative actions and all commands that are sent to managed devices from the GovShield Server. Audit records are generated by the GovShield Server for MDM Agent alerts (FAU_ALT_EXT.2). The GovShield Client will also generate audit records defined under [AGENTMOD] FAU_GEN.1(2). The TOE components and their underlying platforms also generate audit records for the specific auditable events listed in Table 16 below.

Table 15: Auditable Events by Enforcing Component

Requirement	Auditable Event(s)	Component Generating Record
FAU_ALT_EXT.1	Type of alert.	GovShield Server
FCO_CPC_EXT.1	Enabling or Disabling communications between a pair of components.	GovShield Server
FCS_RBG_EXT.1	Failure of the randomization process.	GovShield Server Platform GovShield Client Platform Note: The auditing for this SFR is invoked by the platforms' cryptographic modules.
FIA_ENR_EXT.1	Failure of MD user authentication.	GovShield Server
FIA_X509_EXT.1(1)	Failure to validate X.509 certificate.	GovShield Server Platform GovShield Client Platform Note: Platform auditing for this SFR for: (1) TLS/HTTPS is invoked by the platforms' mechanisms which implement TLS and HTTPS communication, and (2) verifying signed policies is invoked by the GovShield Client platform's signature services.
FIA_X509_EXT.2	Failure to establish connection to determine revocation status.	GovShield Server Platform GovShield Client Platform Note: Platform auditing for this SFR for: (1) TLS/HTTPS is invoked by the platforms' mechanisms which implement TLS and HTTPS communication, and (2) verifying

Requirement	Auditable Event(s)	Component Generating Record
		signed policies is invoked by the GovShield Client platform's signature services.
FMT_MOF.1(1)	Issuance of command to perform function. Change of policy settings.	GovShield Server
FMT_MOF.1(2)	Enrollment by a user.	GovShield Server
FMT_SMF.1(2)	Success or failure of function.	GovShield Server
FPT_ITT.1(2)	Initiation and termination of the trusted channel.	GovShield Server Platform Note: Platform auditing for this SFR for TLS/HTTPS is invoked by the platforms' mechanisms which implement TLS/HTTPS communication.
FPT_TST_EXT.1	Initiation of self-test. Failure of self-test. Detected integrity violation.	GovShield Server Platform Note: Platform auditing for this SFR is invoked by the platform's own self-test mechanisms when the TOE's software is started or restarted
FPT_TUD_EXT.1	Success or failure of signature verification.	GovShield Server Platform GovShield Client Platform Note: Platform auditing for this SFR is invoked by the platforms during the installation process to check the digital signature of the software update.
FTA_TAB.1	Change in banner setting.	GovShield Server
FTP_ITC.1(1)	Initiation and termination of the trusted channel.	GovShield Server Platform Note: Platform auditing for this SFR for TLS is invoked by the platforms' mechanisms which implement TLS communication.
FTP_TRP.1(1)	Initiation and termination of the trusted channel.	GovShield Server Platform Note: Platform auditing for this SFR for TLS/HTTPS is invoked by the platforms' mechanisms which implement TLS/HTTPS communication.
FTP_TRP.1(2)	Initiation and termination of the trusted channel.	GovShield Server Platform Note: Platform auditing for this SFR for TLS/HTTPS is invoked by the platforms' mechanisms which implement TLS/HTTPS communication.

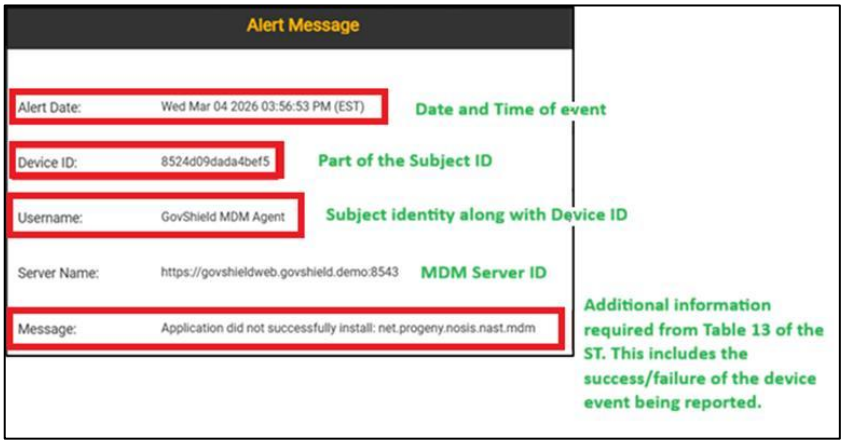
Each MDM Server record uses a templated box that identifies the Device ID (if applicable), Server Name, Event Type, Success/Failure, Date and Time, Subject Identity, and Additional Information.

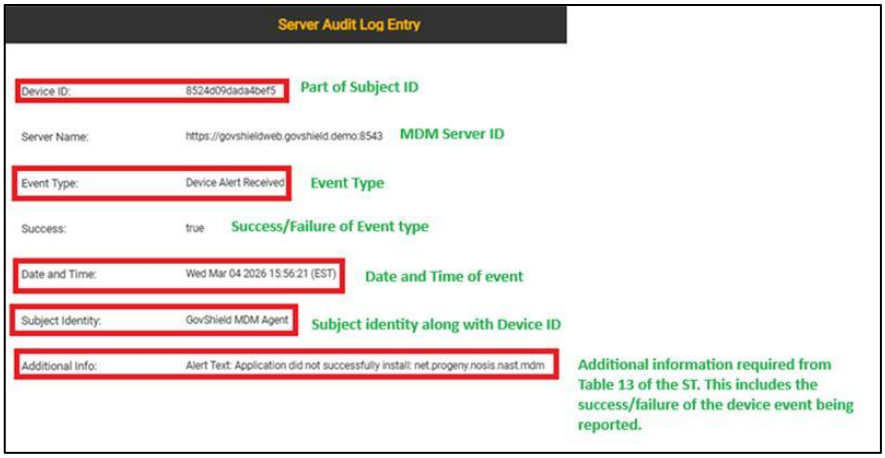


8.1.4 [MDMPP] FAU_GEN.1(2)

Based upon the GovShield Client's associated 'APK Management' configuration, the GovShield Client connects to the GovShield Server, which includes the MAS Server functionality of the TOE, to attempt to download and install the current version of the applications associated with the GovShield Client's 'APK Management' configuration. The GovShield Client will inform the GovShield Server of failed attempts to download and install a new application or an application update to the Android Mobile Device. The GovShield Server will then create an audit record of the event.

Each MAS Server record uses a templated box that identifies the Device ID (if applicable), Server Name, Event Type, Success/Failure, Date and Time, Subject Identity, and additional information. The audit record requires an alert message (first figure) from the device to be sent and received by the MDM server (second figure) for all failed application updates.





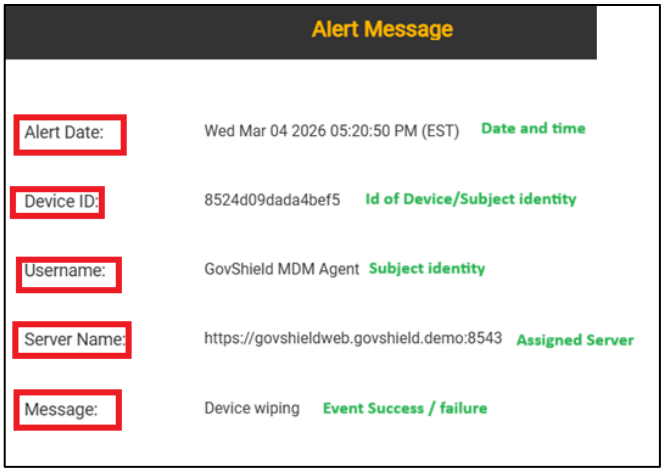
8.1.5 **[AGENTMOD] FAU_GEN.1(2)**

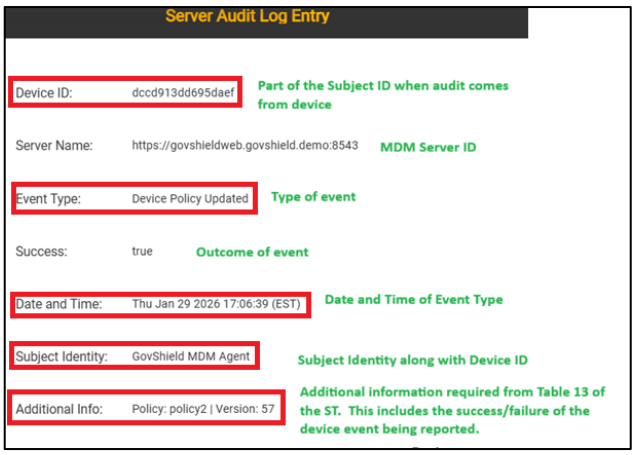
[AGENT] The GovShield Client component of the TOE and its underlying platform generate audit events for activities on the Android Mobile Device. The underlying platform generates audit logs for the GovShield Client's startup and shutdown. The GovShield Client also generates audit data for the specific auditable events listed in Table 17 below.

Table 16: Agent Auditable Events by Enforcing Component

Requirement	Auditable Event(s)	Component Generating Record
FAU_ALT_EXT.2	Success/failure of sending alert.	GovShield Client
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	GovShield Client
FIA_ENR_EXT.2	Enrollment in management.	GovShield Client
FMT_POL_EXT.2	Failure of policy validation.	GovShield Client Note: The GovShield Client's platform performs the X.509 validation check but it is the GovShield Client which will produce the audit record if the check fails.
FMT_SMF_EXT.4	Outcome (Success/failure) of function.	GovShield Client GovShield Client Platform Note: The GovShield Client Platform is invoked by a request for the mobile device's software to be updated through the E-FOTA Server.

Each Device audit record uses a templated box that has a Date and Timestamp, identifies the Device ID as part of subject identity, Username as part of subject identity, Server Name to which device is assigned, Message which is the Event Type and Success/Failure. The audit record information is the same but is displayed slightly differently depending on whether it is viewed on the Alerts page or the Server Audit Log Entry page. Examples of both formats are depicted below.





8.1.6 **[MDMPP] FAU_NET_EXT.1**

[SERVER] Authorized Administrators have two methods for checking the network connectivity status of an enrolled Android Mobile Device on the web GUI. Under the 'Device Management' dashboard, each enrolled Android Mobile Device has a date and time defined under the "Last Contacted" column. This identifies the overall last time the GovShield Client connected to the GovShield Server for a reachability event, regardless of which reachability event action initiated the alert. Meanwhile, under the 'Device Alerts' dashboard of the web GUI, every reachability event from a GovShield Client is listed as an alert for the Android Mobile Device. Each alert has a date and time defined under the Alert Date column. Therefore, an Administrator can observe all of the reachability events for an Android Mobile Device, with an Android Mobile Device's latest reachability event alert identifying the last time GovShield Client connected to the Server. The GovShield Client initiates all reachability events as described under FAU_ALT_EXT.2.

8.1.7 **[MDMPP] FAU_SAR.1**

[SERVER] For audited events that are related to the GovShield Server component of the TOE, the 'Server Audit Log' dashboard on the web GUI provides Administrators with the ability to review these audit records. For audited events that are related to the GovShield Client component of the TOE and sent to the GovShield Server for storage in the Oracle Database, the 'Audit Log' and 'Device Alerts' dashboards on

the web GUI provide Administrators with the ability to review these audit records. These dashboards provide a graphical view of the log data in a human-readable format. The audit data can be searched and sorted using these dashboards.

Table 16, Section 8.1.4, and Table 17 identify the auditable events that are logged by the GovShield Server, the GovShield Client, and when applicable their underlying platforms. Only the auditable events that are generated by a TOE component are applicable to this requirement and will be reviewable via these web GUI dashboards. Audit records generated by an underlying TOE platform can be reviewed using the mechanisms made available by the underlying platform.

8.1.8 **[AGENTMOD] FAU_SEL.1(2)**

[AGENT] There is no specific configuration to turn on and off auditing on the TOE, thus the GovShield Client and its underlying platform will always perform auditing. However, an Authorized Administrator creates policies on the GovShield Server and will assign them to one or more Android Mobile Devices. These policies include requirements for the GovShield Client to generate audit records for the functionality configured in the policies. Once the GovShield Client receives and applies a policy requiring auditing, the GovShield Client will always generate the necessary audit records. For this reason, the selection of auditable events is configurable through policy.

The TSF does not support specification of more complex audit pre-selection criteria, such as multiple attributes or logical expressions using attributes.

8.1.9 **[MDMPP] FAU_STG_EXT.1**

[SERVER] Audit data managed by the GovShield Server will be transmitted from the GovShield Server to a remote Oracle Database over a TLS v1.2 encrypted trusted channel as described under FTP_ITC.1(1). The audit data that is transferred includes audit records generated by the GovShield Server as well as audit records that are received from the GovShield Client. All TOE audit data is sent to the Oracle Database in real time. This does not include audit data that is generated by these TOE components' underlying platforms, as this audit data is not managed by the TOE's software boundary. It is therefore the responsibility of the Operational Environment to securely transfer this audit data to a remote location for permanent storage.

Note that the GovShield Server's functionality relies on an established connection to the remote Oracle Database. If the connection to the remote Oracle Database is broken, all functionality (including auditing and audit processing) of the GovShield Server stops until this connection is re-established.

[AGENT] Audit records generated by the GovShield Clients are transmitted to the GovShield Server over the HTTPS internal TOE trusted channel as described under FTP_ITC_EXT.1. As described above, once these audit records reach the GovShield Server, the audit records will be sent to the remote Oracle Database over a TLS v1.2 encrypted trusted channel as described under FTP_ITC.1(1).

The GovShield Client's audit data is stored in a local database which is part of the GovShield Client software. All of the GovShield Client's audit data is stored together regardless of which portion of the policy generated the audit data, and the maximum number of audit events which can be stored in the local database is based upon the Android Mobile Device's available disk space. Once a connection for transmitting audit data is established between the GovShield Client and the GovShield Server, all audit data will be transmitted and upon successful transmission the audit data will be removed from the

GovShield Client's local database. If a connection cannot be established to the GovShield Server when the audit data is expected to be transmitted, the GovShield Client will continue to queue the audit data within its local database. Additionally, the GovShield Client will the continuously attempt to connect to the GovShield Server on a hardcoded 1-minute time internal to send all of its queued audit data.

The GovShield Client will establish a connection to the GovShield Server to send the audit data based upon either the GovShield Client's policy for transmitting audit data or the MD user manually pushing the audit data. The GovShield Client's policy for transmitting audit data has an administratively configurable time period set through the 'Audit Log Publishing Rate (hours)' variable that can be configured by Authorized Administrators on the GovShield Server's web GUI. The MD user manually pushing the audit data is accomplished by the MD user configuring the interval for transmitting the audit data through the GovShield Client's interface.

8.2 Communication

8.2.1 [MDMPP] FCO_CPC_EXT.1

[SERVER] The GovShield Clients that can join the TOE by enrolling with the GovShield Server are limited based upon the Android Mobile Devices' Android Device ID. The configuration of this enrollment restrictions is the enablement step by the Authorized Administrator through the GovShield Server's web GUI. The entire enrollment process from the GovShield Server's perspective is described under FIA_ENR_EXT.1 and FIA_CLI_EXT.1. The GovShield Server relies on its underlying platform to provide the secure registration channel to the GovShield Clients that attempt to enroll and join the TOE. This secure registration channel is described under FTP_TRP.1(2) and is used for all communications between the GovShield Server and the GovShield Clients during the enrollment process. After a GovShield Client has enrolled and joined the TOE, an Authorized Administrator can disable the communication between a GovShield Client and the GovShield Server by unenrolling or wiping the enrolled Android Mobile Device. This will result in the removal of the GovShield Client and the unenrollment of the Android Mobile Device which will prevent communication between the Android Mobile Device and the GovShield Server.

[AGENT] The entire enrollment process from the GovShield Client's perspective is described under FIA_ENR_EXT.1, FIA_ENR_EXT.2, and FIA_CLI_EXT.1. The GovShield Client relies on its underlying platform to provide the secure registration channel to the GovShield Server when enrolling into management and joining the TOE. This secure registration channel is described under FTP_TRP.1(2) and is used for all communications between the GovShield Clients and the GovShield Server during the enrollment process.

8.3 Cryptographic Support

[SERVER] Cryptographic services for the GovShield Server are provided by the underlying Java Runtime Environment Platform. The Java Runtime Environment Platform uses the BSafe cryptographic library to perform all cryptographic services.

[AGENT] Cryptographic services for the GovShield Client is provided by the underlying Android mobile device platform. The GovShield Client uses the Android platform's BoringSSL cryptographic module to perform all claimed cryptographic services.

8.3.1 [MDMPP] FCS_CKM.1

[SERVER] The GovShield Server invokes the platform provided functionality for asymmetric key generation in support of HTTPS and TLS communications. The underlying platform provides functionality to support RSA schemes using cryptographic key sizes of 2048-bit or greater that meet FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.1 and ECC schemes using “NIST curves” P-384 that meet FIPS PUB 186-5, “Digital Signature Standard (DSS)”, Appendix B.4.

[AGENT] The GovShield Client’s software invokes the underlying platform provided functionality in support of HTTPS communications. The platform supports ECC schemes using “NIST curves” P-384 that meet FIPS_PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2

8.3.2 [MDMPP] FCS_CKM.2

[SERVER] The GovShield Server invokes its underlying platform to perform key establishment for HTTPS and TLS communications using the following two key establishment schemes:

- RSA key establishment conforming to “RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017”, and
- Elliptic curve-based key establishment conforming to NIST Special Publication 800-56A.

[AGENT] The GovShield Client invokes its underlying mobile device platform to perform key establishment for HTTPS communications using the following key establishment scheme:

- Elliptic curve-based key establishment conforming to NIST Special Publication 800-56A.

8.3.3 [MDMPP] FCS_CKM_EXT.4

[SERVER] The GovShield Server invokes the underlying platform’s FIPS cryptographic module to destroy the keys and cryptographic security parameter data when no longer needed. The invoking of the destruction of keys stored in volatile memory occurs as a result of the GovShield Server making a cryptographic function call which requires a key and/or cryptographic security parameter data to be generated by the platform. The platform will destroy the reference to the key followed by a request for garbage collection when the generated cryptographic data is no longer needed. This occurs without requiring a separate function call to the platform by the GovShield Server. The invoking of the destruction of keys stored in non-volatile memory occurs as a result of the Authorized Administrator replacing the key within the Java KeyStore. The platform will destroy the key with a single direct overwrite with a new value of a key.

[AGENT] The GovShield Client’s software invokes the underlying mobile device platform to perform key destruction. The invoking of key destruction occurs as a result of the GovShield Client making a cryptographic function call which requires a key and/or cryptographic security parameter data to be generated by its platform. The platform will therefore perform key destruction when the generated cryptographic data is no longer needed, without requiring a separate function call for key destruction from the GovShield Client. Key data maintained by the GovShield Client platform in volatile memory are erased by a one-pass overwrite with zeroes. Key data maintained by the GovShield Client’s platform in non-volatile memory is stored in flash memory and is erased by a one-pass overwrite with zeroes.

8.3.4 [MDMPP] FCS_COP.1(1)

[SERVER] The GovShield Server invokes its underlying platform to perform AES encryption/decryption services for HTTPS and TLS communications. Data in transit is protected using GCM mode and either 128-bit or 256-bit keys, which is conformant to NIST SP 800-38D.

[AGENT] The GovShield Client invokes its underlying mobile device platform to perform AES encryption/decryption for HTTPS communications. The platform uses GCM mode and 128-bit keys, which is conformant to NIST SP 800-38D.

8.3.5 [MDMPP] FCS_COP.1(2)

[SERVER] The GovShield Server invokes its underlying platform to provide SHA-256 and SHA-384 cryptographic hashing services, conformant to FIPS PUB 180-4, with digest sizes of 256 and 384 bits, respectively. Cryptographic hashing services are used for the following purposes:

- SHA-256 and SHA-384 are used by HMAC in message authentication in support of TLS and HTTPS communication.
- SHA-256 in support of ECDSA with P-256 curves used for digital signature services in support of HTTPS communication.
- SHA-256 is used for the digital signing of policies (ECDSA with P-256 curve).

[AGENT] The GovShield Client invokes the underlying mobile device platform to provide SHA-256 cryptographic hashing services, conformant to FIPS PUB 180-4, with digest sizes of 256 bits. Cryptographic hashing services are used for the following purposes:

- SHA-256 is used by HMAC in message authentication in support of HTTPS communication.
- SHA-256 in support of ECDSA with P-256 curves used for digital signature services in support of HTTPS communication.
- SHA-256 is used for the digital signing of policies (ECDSA with P-256 curve).

8.3.6 [MDMPP] FCS_COP.1(3)

[SERVER] The GovShield Server invokes its underlying platform to provide all digital signature services in accordance with FIPS PUB 186-5. RSA with 2048-bit keys is used for policy signature generation, and digital signature services in support of HTTPS and TLS communication.

[AGENT] The GovShield Client invokes the underlying mobile device platform to provide digital signature services in accordance with FIPS PUB 186-5. RSA with 2048-bit keys is used for policy signature verification functionality, and digital signature services in support of HTTPS communication.

8.3.7 [MDMPP] FCS_COP.1(4)

[SERVER] The GovShield Server invokes its platform to provide keyed-hash message authentication services conformant to FIPS PUBs 180-4 and 198-1. HMAC-SHA-256 and HMAC-SHA-384 are used to perform keyed-hash message authentication, with respective digest sizes of 256 and 384 bits in support of trusted communication.

[AGENT] The GovShield Client invokes the underlying mobile device platform to provide keyed-hash message authentication services conformant to FIPS PUBs 180-4 and 198-1. HMAC-SHA-256 is used to

perform keyed-hash message authentication, with respective digest sizes of 256 bits in support of trusted communication.

8.3.8 [MDMPP] FCS_RBG_EXT.1

Note: The GovShield Server and GovShield Client software do not directly invoke their respective platforms' deterministic random bit generator. Instead, the TOE's software indirectly invokes their platforms' deterministic random bit generator by directly invoking platform components, which in turn directly invoke the deterministic random bit generator.

[SERVER] The GovShield Server invokes the Java Runtime Environment Platform to provide random bit generation services. The underlying platform's cryptographic module provides an AES counter DRBG (CTR_DRBG) that conforms to NIST SP 800-90A. In order to provide sufficient randomness to the keys generated by this DRBG (as specified in NIST SP 800-57), the DRBG is seeded with at least 256 bits of entropy which is gathered from a platform based RBG.

[AGENT] The GovShield Client software invokes the underlying mobile device platform to provide random bit generation services. The Android platform provides an AES counter DRBG (CTR_DRBG) that conforms to NIST SP 800-90A. In order to provide sufficient randomness to the keys generated by this DRBG (as specified in NIST SP 800-57), the DRBG is seeded with at least 256 bits of entropy from the platform's hardware-based noise source.

8.3.9 [MDMPP] FCS_STG_EXT.1

[SERVER] The TOE's Java Runtime Environment Platform is responsible for storing keys and relies on the BSafe cryptographic library to invoke the storage of persistent secrets and private keys which are produced through their operation. All persistent private keys are stored in the encrypted Java KeyStore/TrustStore and all user credentials are stored in authentication repositories. The X.509v3 certificate, for policy signing, is generated by the CA Server and loaded into the Java KeyStore after a request is made by the Authorized Administrator through the TOE. The X.509v3 certificates used for HTTPS and TLS are created by the CA Server, and loaded into the encrypted Java KeyStore/TrustStore by an Authorized Administrator through the underlying platform's interface.

The following table contains the list of keys and CSPs for the GovShield Server platform:

Table 17: Keys and CSPs for GovShield Server Operation

Key	Purpose	Origin	Storage
GovShield Server Private Key	Signs the handshake parameters to prove identity. Used to create the digital signature of the policy data	CA server	Java KeyStore
X.509 GovShield Server Certificate Chain/Server Public Key for TLS communications	Proves identity to the client (via certificate)	CA server	Java KeyStore
CA Root Certificate for GovShield Server	Invoking the generation of GovShield Client certificate during enrollment	CA server	Java TrustStore

	Validation of GovShield Client certificate chain which must end in a trusted root CA certificate		
CA Root certificate for DB Server Public Key	Validation of Database Server certificate chain which must end in a trusted root CA certificate	CA server Imported on device as part of installation steps	Volatile Memory Android TrustStore
GovShield Client Certificate Chain/Client Public Key	Used to verify the client's digital signature during the TLS handshake.	CA Server Forwarded to Agent via Server as part of enrollment	Volatile memory
Database Server Certificate Chain/Server Public Key	Encrypts the secret (Pre-Master Secret) for the server	CA Server	Volatile memory
Ephemeral Keys	Temporary cryptographic keys generated as part of a single TLS connection	Java Platform - BSafe Crypto	Volatile memory
Session Keys	Temporary, symmetric keys used to encrypt and decrypt the data sent during a single communication session.	Java Platform - BSafe Crypto	Volatile memory
RBG CSPs	Random Bit Generation	Java Platform - BSafe Crypto	Volatile memory
X.509 GovShield Server Certificate Chain/Server Public Key for digital signature	Server Public Key used in signing policy	CA server	Java KeyStore
Hashing CSP	Create a unique "fingerprint" of policy	Java Platform - BSafe Crypto	Volatile memory
Encryption CSP	Uses Private Key to encrypt that hash	Java Platform - BSafe Crypto	Volatile memory

[AGENT] The FCS_STG_EXT.1(2) section describes key storage for the GovShield Client.

8.3.10 **[AGENTMOD] FCS_STG_EXT.1(2)**

[AGENT] All the GovShield Client's persistent keys are stored in the encrypted Android Keystore/TrustStore of the device. The GovShield Client relies on its platform's BoringSSL to invoke the storage of persistent secrets and private keys which are produced through their operation. This cryptographic module is invoked by the platform APIs available to the GovShield Client when requesting an encryption function. (see Section 8.6.1 for the list of APIs)

The following table contains the list of keys and CSPs for the GovShield Client platform:

Table 18: Keys and CSPs for GovShield Client Operation

Key	Purpose	Origin	Storage
GovShield Client Private Key	Signs the handshake to prove the client's identity.	CA server	Android Keystore

GovShield Client Certificate Chain/Client Public Key	Sent to the server to verify the client's identity for mutual authentication	CA server	Volatile memory Android Keystore
X.509 GovShield Server Certificate Chain/Server Public Key	Received as part of TLS communication	GovShield Server	Volatile Memory
CA Root Certificate for GovShield Server	Validation of GovShield Server certificate chain which must end in a trusted root certificate	CA server Imported on device as part of installation steps	Volatile Memory Android TrustStore
Ephemeral Keys	Temporary cryptographic keys generated as part of a single TLS connection	Java Platform - BSafe Crypto	Volatile memory
Session Keys	temporary, symmetric keys used to encrypt and decrypt the data sent during a single communication session.	Java Platform - BSafe Crypto	Volatile memory
RBG CSPs	Random Bit Generation	Android Platform - BoringSSL	Android Platform - BoringSSL
X.509 GovShield Server Certificate Chain/Server Public Key for digital signature	Public Key used to validate digital signature of Policy	Forwarded to GovShield Agent via GovShield Server as part of enrollment	Volatile memory
Hashing CSP	Hashes the received policy to verify against the digital signature	Java Platform - BSafe Crypto	Volatile memory
Decryption CSP	Uses the Servers public key to decrypt the signature leaving hash for comparison	Java Platform - BSafe Crypto	Volatile memory

8.4 Identification and Authentication

8.4.1 **[MDMPP] FIA_ENR_EXT.1**

The Android Mobile Device must be in its factory reset state in order to be enrolled into management. Once the Android Mobile Device is booted, a MD user or an Administrator will scan a QR code created for the enrollment of devices. This is accomplished by tapping on the Android Mobile Device's screen five (5) times to launch the device's QR code scanner and then scanning the QR code. Within the QR code is an embedded URL for the GovShield Server. The Android Mobile Device will connect to the GovShield Server to download and install the GovShield Client software. The MD user or Administrator will then enter their username and password credentials for authentication through the GovShield Client interface which will be used to establish the enrollment connection that is described under FTP_TRP.1(2). Once the HTTPS/TLS connection between the GovShield Client and the GovShield Server is established, this enrollment connection is used until enrollment is complete or the enrollment process is terminated.

The GovShield Client will then send the user credentials to the GovShield Server for authentication and the GovShield Server will validate them against the user table in the Oracle Database. If authentication fails, the enrollment process is stopped and the GovShield Server informs the user by relaying the failed authentication attempt to the GovShield Client. If authentication is successful, the GovShield Server will request the GovShield Client to provide the Android Mobile Device's Android Device ID; which the GovShield Server will confirm against the Administrator configurable whitelisted set of Android Device IDs. If the Android Mobile Device's Android Device ID is not found in the whitelist, the enrollment process is stopped and the GovShield Server informs the user by relaying the failed enrollment attempt to the GovShield Client. If the Android Mobile Device's Android Device ID is found in the whitelist, the remainder of the enrollment process can proceed.

The GovShield Server then requests an X.509v3 certificate using Certificate Management Protocol (CMP) from the CA Server. The enrolling Android Mobile Device will use this unique X.509v3 certificate for post-enrollment HTTPS/TLS communications with the GovShield. The GovShield Server then creates an initial payload for that Android Mobile Device which includes its unique X.509v3 certificate and the public key to verify the signature of policies received by the GovShield Client. The GovShield Server sends the initial payload to the GovShield Client for configuration and storage of the payload contents. The GovShield Client then downloads its assigned policy from the GovShield Server and will request its platform to verify the signed policy per the process described under FMT_POL_EXT.2. Included within the policy received by the GovShield Client is the Knox Key. Before any other part of the policy is applied, the GovShield Client provides the Knox Key (contains the Samsung Knox Licensing Server's FQDN) to the Android Mobile Device and the device connects to the Samsung Knox Licensing Server to register the Android Mobile Device's Knox license using the Knox Key. Once the Knox license is validated, the Android Mobile Device's Knox Platform is activated and the GovShield Client applies the remaining portions of the policy to the Android Mobile Device. Once complete, the GovShield Client informs the GovShield Server that the policy was implemented, and the enrollment process is complete.

8.4.2 **[AGENTMOD] FIA_ENR_EXT.2**

[AGENT] During the enrollment process, an MD user or Administrator will scan the QR code which requires tapping on the screen five (5) times to scan the QR code. Within the QR code is an embedded URL for the GovShield Server. The GovShield Client will record the URL of the GovShield Server during a successful enrollment of the Android Mobile Device. The URL will then be used as the GovShield Server's reference identifier for subsequent communications between the GovShield Client and GovShield Server.

8.4.3 **[MDMPP] FIA_UAU.1**

The GovShield Server has a configurable consent warning banner which is displayed and requires acceptance prior to authentication taking place for the web GUI. The GovShield Client has a consent warning banner which is displayed and requires acceptance prior to authentication taking place to the GovShield Server. All other means of user interaction with the TOE requires the Administrator to be authenticated to the web GUI, or the MD user or Administrator to be authenticated via the GovShield Client. The GovShield Server validates all username and password credentials it receives via the web GUI and from GovShield Clients against the user table within the Oracle Database. If the credentials match those stored in the Oracle Database, authentication is successful, and access is granted by the originating

TOE component. If the credentials do not match those stored in the Oracle Database, authentication is not successful, and access is denied by the originating TOE component.

8.4.4 [MDMPP] FIA_X509_EXT.1(1) and [MDMPP] FIA_X509_EXT.2

[SERVER] The GovShield Server relies on the underlying platform to provide X.509v3 certificate services for signing policies that are sent to GovShield Client. The GovShield Server's underlying platform signs policies with the certificate that the platform generates specifically for policy signing based upon an Authorized Administrator's operation through the web GUI. The GovShield Server also relies on its platform to provide its X.509v3 certificate as part of TLS and HTTPS/TLS session establishment in all cases where the GovShield Server is the server component in the session (e.g., connections to GovShield Clients) as well as upon the server component's request where the GovShield Server is the client component and mutual authentication has been configured. The GovShield Server's platform will validate all certificates it receives as part of TLS and HTTPS/TLS session establishment. The HTTPS/TLS connection between the GovShield Clients and the GovShield Server requires that the X.509 certificate be loaded in the Java Keystore as this is the location required by the GovShield Server's platform.

The GovShield Server's platform performs the following checks in order to determine if a certificate is valid:

- Certificate validation and certificate path validation conforms to RFC 5280.
- The certificate path must terminate with a trusted CA certificate.
- All certificates must have the basicConstraints extension present, the CA flag set to TRUE for all CA certificates, and any path constraints are met.
- All CA certificates must have the caSigning purpose in the key usage field.
- The GovShield Server's platform uses the OCSP as specified in RFC 6960 to verify revocation status. Revocation checking occurs each time a certificate is presented for a validation check. If the GovShield Server's platform cannot establish a connection to determine the validity of a certificate, then the certificate is not accepted.
- The extendedKeyUsage field must be valid based on the following rules:
    - Certificates used for trusted updates and executable code integrity verification must 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 must 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 must 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 must 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 must 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.

Additionally, the GovShield Server platform's certificate validation service will ensure that all certificate paths terminate with a CA certificate and that all CA certificates include the basicConstraints extension with the CA flag set to TRUE.

[AGENT] The GovShield Client relies on the underlying platform's cryptographic modules to provide X.509v3 certificate services for signature verification for signed policies sent from the GovShield Server, and in support of HTTPS/TLS connections from the GovShield Client to the GovShield Server. The GovShield Client is issued an X.509v3 certificate per the process described under FIA_CLI_EXT.1, and receives from GovShield Server the public key used for verification of the signed policies during the enrollment process.

The GovShield Client platform will present an X.509v3 certificate as part of the HTTPS/TLS session establishment in all cases where the GovShield Client is the client component and mutual authentication has been configured (e.g., connections to the GovShield Server). Upon request for an X.509v3 certificate, the GovShield Client's underlying platform will present the unique X.509v3 certificate that the Android Mobile Device was issued as part of FIA_CLI_EXT.1. Additionally, the GovShield Client requests its underlying platform to perform X.509v3 certificate services for the verification of signed policies received from the GovShield Server before the policies are applied by GovShield Server. The underlying platform is able to determine which certificate to use for the validity check based upon the presented certificate's data.

The GovShield Client platform performs the following checks in order to determine if a certificate is valid:

- Certificate validation and certificate path validation conforms to RFC 5280.
- The certificate path must terminate with a trusted CA certificate.
- All certificates must have the basicConstraints extension present, the CA flag set to TRUE for all CA certificates, and any path constraints are met.
- All CA certificates must have the caSigning purpose in the key usage field.
- The GovShield Client's platform uses CRL as specified in RFC 5280 Section 6.3 to verify revocation status. Revocation checking occurs each time a certificate is presented for a validation check. If the GovShield Client's platform cannot establish a connection to determine the validity of a certificate, then the certificate is not accepted.
- The extendedKeyUsage field must be valid based on the following rules:
    - Certificates used for trusted updates and executable code integrity verification must 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 must 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 must 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 must 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 must 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.

Additionally, the GovShield Client platform certificate validation services will ensure that all certificate paths terminate with a CA certificate and that all CA certificates include the basicConstraints extension with the CA flag set to TRUE.

8.4.5 [MDMPP] FIA_CLI_EXT.1

[SERVER] The GovShield Server requires each Android Mobile Device to have a unique X.509v3 certificate which is used by the GovShield Server platform to perform the client-side authentication of the Android Mobile Device as part of the GovShield Client to GovShield Server communications that are described under FPT_ITT.1(2). The GovShield Server requests a unique X.509v3 certificate using CMP from the CA Server and includes the X.509v3 certificate within the initial payload sent to the Android Mobile Device during enrollment.

[AGENT] Each Android Mobile Device receives its unique X.509v3 certificate within the initial payload sent to it during enrollment by the GovShield Server. Once the Android Mobile Device receives its unique X.509v3 certificate and the enrollment process is complete, all subsequent communications between a GovShield Client and the GovShield Server occurs over the connection described under FPT_ITT.1(2). The GovShield Client presents its unique X.509v3 certificate as part of the establishment of the post-enrollment connections to the GovShield Server.

8.5 Security Management

8.5.1 [MDMPP] FMT_MOF.1(1)

[SERVER] The GovShield Server provides the capability to manage its own functionality as well as the behavior of the Android Mobile Devices that are under management. Only Authorized Administrators can manage the TOE through the GovShield Server's web GUI. The web GUI provides the ability for Administrators to configure the X.509v3 certificate for policy signing, the Android Mobile Devices that are allowed to enroll by defining an Android Device ID whitelist, the unlock banner for the web GUI, the time period for the periodic checks between a GovShield Client and the GovShield Server, and the interaction between TOE components (as described under FMT_SMF.1(2)). The web GUI also provides Administrators the ability to perform functionality related to its MAS Server capabilities (as described in FMT_SMF.1(3) below) which includes the ability to configure application access groups and upload applications for GovShield Clients to download. For the configuration of the Android Mobile Devices under management, the full list of functions that the TSF can perform and a reference to where in the web GUI this behavior can be found is described under FMT_SMF.1(1).

8.5.2 [MDMPP] FMT_MOF.1(2)

[SERVER] Initiating the enrollment process can be performed by either a MD user or an Administrator with physical custody of the mobile device. The enrollment process is unchanged regardless of the role that initiates enrollment. During enrollment, the MD user or Administrator must enter their username and password though the GovShield Client interface. The GovShield Client then sends these credentials to the GovShield Server for validation against the user table within the Oracle Database. If the MD user or Administrator passes authentication, they are authorized to perform the enrollment process and the remainder of the enrollment process described under FIA_ENR_EXT.1. If authentication fails, the enrollment process will not proceed.

8.5.3 [MDMPP] FMT_MOF.1(3) and [MDMPP] FMT_SMF.1(3)

[SERVER] The MAS Server component of the GovShield Server provides the ability to configure application access using the web GUI. An 'application access group' is the set of Android Mobile Devices

which have been assigned to a policy for application management. The MAS Server is defined in the web GUI under the 'APK Management' dashboard. Applications can be added to the MAS Server as an individual file that is uploaded to the GovShield Server and stored in the Oracle Database. During the application's initial upload or after the upload has been completed, the application is assigned to one or more policies. The Android Mobile Devices that are also assigned to one of those same policies will then have the ability to download the application. Note that each Android Mobile Device can only be assigned one policy at a time. Upon each GovShield Client's next reachability event related to application management, for each application assigned to the GovShield Client's policy, the GovShield Client will download and install the application or the latest application update made available through the MAS Server. This is as long as the Android Mobile Device is properly enrolled and compliant with MDM policies assigned to the device.

8.5.4 [MDMPP] FMT_POL_EXT.1

[SERVER] The GovShield Server provides policies that can be assigned to one or more Android Mobile Devices. The GovShield Server invokes its platform to digitally sign the policies with a trusted X.509v3 certificate using RSA 2048 with SHA-256. Policies are then transmitted to the assigned Android Mobile Devices, and each policy's signature will be validated by the GovShield Client's platform before the GovShield Client applies the policy to its Android Mobile Device. This process requires the GovShield Server to provide the GovShield Client the public key used to verify the signature of policies as part of an initial payload during enrollment of an Android Mobile Device.

8.5.5 [AGENTMOD] FMT_POL_EXT.2

[AGENT] Policies are sent to a device's GovShield Client by the GovShield Server when they are assigned to that Android Mobile Device and the GovShield Client performs a reachability event to determine an updated or new policy has been assigned. Every policy is signed with a trusted X.509v3 certificate using RSA 2048 with SHA-256 by the GovShield Server's platform. The GovShield Client requests its underlying platform to verify each policy before the GovShield Client will apply the policy to the Android Mobile Device. Verification of a policy's signature is performed as described under FIA_X509_EXT.1(1) and FIA_X509_EXT.2. If a policy is received that is not signed, signed by an incorrect certificate, or the certificate is deemed invalid, the policy will not be applied on the Android Mobile Device.

8.5.6 [MDMPP] FMT_SMF.1(1)

[SERVER] The GovShield Server component of the TOE has the ability to issue commands and configuration policies to enrolled Android Mobile Devices. The following table lists the management functions that can be performed by the GovShield Server as defined by the MDM PP as well as whether this behavior is enforced by the GovShield Client or by the underlying mobile device platform.

Table 19: Management Functions

Command	Claimed in VID11593[17]	Implemented By
1. transition to the locked state	Yes	GovShield Client
2. full wipe of protected data	Yes	GovShield Client

[17] TD0479

3. unenroll from management	Yes	GovShield Client
4. install policies	Yes	GovShield Client
5. query connectivity status	No	GovShield Client
6. query the current version of the MD firmware/software	No	GovShield Client
7. query the current version of the hardware model of the device	No	GovShield Client
8. query the current version of installed mobile applications	No	GovShield Client
9. import X.509v3 certificates into the Trust Anchor Database	Yes	GovShield Client
10. install applications	Yes	GovShield Client
11. update system software	Yes	Platform
12. remove applications	Yes	GovShield Client
13. remove Enterprise applications	Yes	GovShield Client
14. wipe Enterprise data	Yes	GovShield Client
16. alert the user	No	GovShield Client
25. password policy	Yes	GovShield Client
26. session locking policy	Yes	GovShield Client
27. wireless networks (SSIDs) to which the MD may connect	Yes	GovShield Client
28. security policy for each wireless network	Yes	GovShield Client
29. application installation policy	Yes	GovShield Client
30. enable/disable policy for camera and/or microphone across device	Yes	GovShield Client
32. enable/disable policy for cellular and/or NFC	Yes	GovShield Client
33. enable/disable policy for data signaling over USB and/or removable storage card (SD card)	No	GovShield Client
34. enable/disable policy for Wi-Fi tethering	No	GovShield Client
35. enable/disable policy for developer modes	Yes	GovShield Client
36. enable policy for data-at-rest protection	Yes	GovShield Client
37. enable policy for removable media's data-at-rest protection	Yes	GovShield Client
38. enable/disable policy for local authentication bypass	Yes	GovShield Client
47. the unlock banner policy	Yes	GovShield Client
49. enable/disable USB mass storage mode	Yes	GovShield Client
51. enable/disable Hotspot functionality authenticated by passcode	No	GovShield Client
55. enable/disable policy for use of Biometric Authentication Factor	Yes	GovShield Client
58. enable/disable automatic updates of system software	No	GovShield Client

8.5.7 **[MDMPP] FMT_SMF.1(2)**

[SERVER] The GovShield Server provides the ability to manage its own behavior. Listed below are the internal management functions that are provided along with information about how those functions are performed:

- **Configuration of the devices specified by Android Device ID allowed for enrollment:** Administrators define a whitelist of allowed Android Mobile Devices by their Android Device ID within the 'Device Management' dashboard of the web GUI.
- **Configuration of X.509v3 certificates for GovShield Server use**: GovShield Server utilizes the X.509v3 certificate which is imported into the Java Keystore for its HTTPS/TLS internal channel, trusted channel and trusted path communications. Administrators will generate the X.509v3 certificate used for policy signing within the 'System Configuration' dashboard of the web GUI.
- **Configuration of TOE unlock banner**: Administrators can define the Consent Banner (i.e., TOE unlock banner) within the 'System Configuration' dashboard of the web GUI.
- **Configuration of the periodicity of agent communications with the Polling Interval, App Polling Interval, and Audit Log Publishing Rate**: Administrators define these MDM Configurations as part of each policy created within the 'Policies' dashboard. The 'App Polling

Interval (mins)' variable is specific to initiating communication regarding the management of applications from the TOE's MAS Server functionality. The 'Audit Log Publishing Rate (hours)' variable determines when the GovShield Client will initiate connection to send its audit records to the GovShield Server. The 'Polling Interval (mins)' variable covers initiating communication for all other TOE functionality related to a GovShield Client's policy.

- **Configuration of the interaction between TOE components:** Administrators define a whitelist of allowed Android Mobile Devices by their Android Device ID within the 'Device Management' dashboard of the web GUI. Specifying an Android Device ID on the whitelist allows for interaction between the TOE components. Administrators are able to unenroll or wipe an enrolled Android Mobile Device within the 'Device Management' dashboard of the web GUI. When an Android Mobile Device is unenrolled or wiped it will cease all interactions between TOE components.

8.5.8 [AGENTMOD] FMT_SMF_EXT.4

[AGENT] The GovShield Client has the ability to interact with its underlying mobile device platform in order to enforce the device management functions set in the policy assigned to it by the GovShield Server. All commands and configurations that are defined in FMT_SMF.1(1) are processed by the GovShield Client when implementing its assigned policy, and result in the mobile device platform being queried or modified in some way. This also includes the ability to upload a X.509v3 certificate, used for secure internal channel communications with the GovShield Server, into the mobile device's certificate store during the enrollment of the mobile device into management. The only method for the enrollment of the mobile device into management is described under FIA_ENR_EXT.1. The GovShield Server will also configure the mobile device platform to prevent unenrollment from management per the description under FMT_UNR_EXT.1.

The GovShield Client is configured by the GovShield Server to generate periodic reachability events based upon the following variables defined within its policy: 'Polling Interval (mins)', 'App Polling Interval (mins)', and 'Audit Log Publishing Rate (hours)'. The 'App Polling Interval (mins)' variable is specific to initiating communication regarding the management of applications from the TOE's MAS Server functionality. The 'Audit Log Publishing Rate (hours)' variable determines when the GovShield Client will initiate connection to send its audit records to the GovShield Server. The 'Polling Interval (mins)' variable covers initiating communication for all other TOE functionality related to a GovShield Client's policy.

8.5.9 [MDMPP] FMT_SMR.1(1)

[SERVER] The TOE has two user roles defined Administrator (i.e., MDM Administrator in the web GUI) and MD User. When an Administrator creates a new user or modifies a user in the web GUI, the Administrator must assign the user to one of these roles. A user cannot be assigned to both roles simultaneously. The GovShield Server stores all user data, including the role assigned, within the Oracle Database.

Administrators can manage the TOE through the web GUI and every Administrator has access to all functions through this interface. MD Users can only interact with the TOE through the GovShield Client interface. Administrators also have the ability to interact with the TOE through the GovShield Client interface.

8.5.10 [MDMPP] FMT_SMR.1(2)

[SERVER] The MAS Server is logically integrated with the GovShield Server. It is accessed by Administrators using the 'APK Management' dashboard in the web GUI. Since this is not accessed separately from the remainder of the GovShield Server capabilities, the user roles and their ability to interact with the MAS Server functionality is defined in the same manner as for FMT_SMR.1(1) above. The GovShield Server also maintains the roles of enrolled mobile devices and application access groups for MAS Server functionality.

8.5.11 [AGENTMOD] FMT_UNR_EXT.1

[AGENT] In the evaluated configuration, the "Admin Removal" checkbox will be checked (i.e., disabled) as part of the default policy configured for all Android Mobile Devices. The default policy is provided to all GovShield Clients upon enrollment which will prevent the unauthorized removal of the GovShield Client's software from the mobile device. When configured in this manner, the GovShield Client will configure the Android Mobile Device's Samsung Knox functionality to prevent the MD user from removing the 'device owner role' being assigned to the GovShield Client on the device. If a MD user was able to remove the 'device owner role' assignment to the GovShield Client, the MD user would be able to unenroll the mobile device from management.

8.6 Protection of the TSF

8.6.1 [MDMPP] FPT_API_EXT.1

[SERVER] The GovShield Server uses only the supported Windows Server platform APIs listed below in order to function.

- Apache Commons
- Apache Logging
- BSafe Crypto
- FasterXML
- Hibernate
- java.security
- javax.security
- javax.servlet
- JBoss EJB
- Spring Framework
- Spring Security
- Wildfly Elytron

[AGENT] The GovShield Client uses only the supported Android platform APIs listed below in order to function.

- android.security
- BSafe Crypto
- java.security
- javax.crypto

- javax.net.ssl
- Samsung Knox
- SQL Cipher

8.6.2 **[MDMPP] FPT_ITT.1(2)**

[SERVER] The GovShield Server's platform uses HTTPS/TLS (TLS v1.2) to provide a trusted communication between the GovShield Server and an enrolled GovShield Client. These protocols are used to protect the data traversing the channel from disclosure and/or modification. This internal channel is used between the GovShield Server and the GovShield Client for all communications between these TOE components after device enrollment. Since the MAS Server is logically integrated with the GovShield Server, the internal channel to secure its communication with the GovShield Client is the same. The GovShield Server's platform identity is validated through its X.509v3 certificate presented during TLS session establishment. The GovShield Server's platform is invoked by the GovShield Client's platform making a HTTPS/TLS connection request. During TLS session establishment, the GovShield Server's platform will also validate the identity of the presented X.509v3 certificate from the GovShield Client's platform. In the evaluated configuration, the GovShield Server's platform will be configured to allow only the following ciphersuite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289.

[AGENT] The GovShield Client relies on its underlying platform to provide the HTTPS/TLS communication path and to validate the GovShield Server's X.509v3 certificate. The GovShield Client's identity is validated through their X.509v3 certificate presented during TLS session establishment. The GovShield Client's platform always initiates this internal channel based upon being invoked by the GovShield Client initiating a reachability event requiring a connection to the GovShield Server.

8.6.3 **[MDMPP] FPT_LIB_EXT.1**

The TOE is packaged with several third-party libraries in order to function.

[SERVER] The GovShield Server uses only the listed third-party dynamic libraries in Table 21 in order to function.

Table 20: GovShield Server Libraries

Group	Artifact Id	Version
ANTLR	Antlr	2.7.7
Apache Commons Codec	Commons-codec	1.9
Apache Commons Collection	Commons-collections	3.2.1
Apache Commons CSV	Commons-csv	1.3
Apache Commons FileUpload	Commons-fileupload	1.6.0
Apache Commons IO	Commons-io	2.21.0
Apache Commons Logging	Commons-logging	1.0.4
Apache HttpClient	Httpclient	4.5.13
Apache HttpCore	Httpcore	4.3.2
Apache Commons Lang	Commons-lang	2.6
Apache Log4j 1.x Compatibility API	Log4j-api	2.25.4
Apache Log4j API	Log4j-api	2.25.4
Apache Log4j Core	Log4j-core	2.25.4
Apache POI Common	Poi	3.15
Dell BSafe Crypto	crypto-j.jar	7.1
Dell BSafe SSL API	ssl-j.jar	7.3.1

Enterprise JavaBeans™ 3.1 API	Jboss.ejb-api_3.1_spec	1.0.2.Final
ESAPI	Esapi	2.2.1.1
GSON	Gson	2.8.9
Hamcrest All	Hamcrest-all	1.3
Hibernate EntityManager Relocation	Hibernate-entitymanager	4.2.14.Final
Hibernate Validator Engine Relocation Artifact	Hibernate-validator	4.3.1.Final
Ical4j	Ical4j	2.0-alpha1
IText Core	itextpdf	5.5.13.2
Jackson Annotations	Jackson-annotations	2.8.7
Jackson Core	Jackson-core	2.8.7
Jackson Databind	Jackson-databind	v2.21.1
Jackson DataType Hibernate4	Jackson-datatype-hibernate4	2.7.4
Java Authorization Contract For Containers API	Javax.security.jacc-api	1.5
Java Concurrency in Practice Book Annotations	Jcip-annotations	1
Java Servlet API	Javax.servlet-api	3.0.1
Java™ EE Interceptors 1.1 API	Jboss-interceptors-api_1.1_spec	1.0.1.Final
Java™ Message Service (JMS) 1.1 API	Jboss-jms-api_1.1_spec	1.0.1.Final
JBoss Jakarta EJB Api_spec	Jboss-ejb-api_3.2_spec	1.0.0.Final
JBoss Logging 3	Jboss-logging	3.1.4.GA
Joda Time	Joda-time	1.6.2
JSON in Java	json	20180130
JSON Library	Json-lib	2.4
JSON Web Token Support For The JVM	Jjwt	0.9.1
Junit	Junit-dep	4.1
Mockito	Mockito-all	1.10.19
Nimbus JOSE+JWT	Nimbus-jsoe-jwt	10.7
PicketBox	Picketbox	4.0.19.SP2-redhat-1
Project Lombok	Lombok	1.18.46
SLF4J API Module	Slf4j-api	1.7.25
Spring Aop	Spring-aop	4.3.1.RELEASE
Spring Beans	Spring-beans	4.3.1.RELEASE
Spring Context	Spring-context	4.3.1.RELEASE
Spring Core	Spring-core	4.3.1.RELEASE
Spring Expression	Spring-expression	4.3.1.RELEASE
Spring JMS	Spring-jms	5.1.10.RELEASE
Spring Security Core	Spring-security-core	4.1.2.RELEASE
Spring Security Config	Spring-security-config	4.1.2.RELEASE
Spring Security Web	Spring-security-web	4.1.2.RELEASE
Spring Web	Spring-web	4.3.1.RELEASE
Spring Web MVC	Spring-webmvc	4.3.1.RELEASE
Test :: Jetty Servlet Tester	Test-jetty-servlet	7.6.4.v20120524
Xerces2 J	xercesImpl	2.4.0
XML APIs	Xml-apis	1.3.04
Xmlwise	Xmlwise	1.2.11

[AGENT] The GovShield Client uses only the listed third-party dynamic libraries in Table 22 in order to function.

Table 21: GovShield Client Libraries

Group	Artifact Id	Version
androidx.activity	activity-ktx	1.2.0-alpha05
androidx.appcompat	appcompat	1.0.0
androidx.constraintlayout	constraintlayout	1.1.3
androidx.fragment	fragment-ktx	1.3.0-alpha05

androidx.legacy	legacy-support-v4	1.0.0
androidx.lifecycle	lifecycle-extensions	2.0.0-rc02
androidx.navigation	navigation-fragment	2.0.0-rc02
androidx.sqlite	sqlite	2.0.1
com.dlazaro66.qrcodereaderview	qrcodereaderview	2.0.3
com.google.android.material	material	1.0.0.0
commons-io	commons-io	2.21.0
com.hudomju	swipe-to-dismiss-undo	1
com.google.android.gms	play-services-safetynet	18.0.1
com.nimbusds	nimbus-jose-jwt	10.7
httpcore	httpcore	4.4.6
knoxsdk	knoxsdk	3.6
net.zetetic	android-database-sqlcipher	4.4.0
Dell BSafe Crypto	crypto-j.jar	7.1
Dell BSafe SSL API	ssl-j.jar	7.3.1
org.jetbrains.kotlin	kotlin-stdlib	1.3.72

8.6.4 **[MDMPP] FPT_TST_EXT.1**

[SERVER] The GovShield Server relies on its platform to perform the self-test functionality. The underlying Windows Server 2022 platform performs its own self-tests to verify that the platform and its underlying hardware are operating correctly. Additionally, the Java Runtime Environment Platform's BSafe cryptographic module performs its own power-up self-tests upon initial start-up, including cryptographic algorithm known answer tests (KATs) and an integrity verification check.

The Java Runtime Environment Platform is responsible for executing GovShield Server's software integrity check during the start or the restart of the GovShield Server's software. Each time the JBoss EAP is started, whether at boot or restarted manually, the TOE validates all of the .jar files that are contained in the GovShield Server's software file. JBoss performs a SHA-256 checksum on each of the .jar files within the deployed GovShield Server's software file, and compares the calculated checksum to the checksum defined in the manifest listing. The manifest listing is created at build time and provided within the GovShield Server's software file. The manifest can be trusted as the GovShield Server's software file is digitally signed at build time, and is verified by the platform as part of the installation process. If any checksum comparison does not match the GovShield Server software will not be started, and the failure is audited along with identifying which .jar file(s) failed.

These tests are sufficient to validate the correct operation of the TSF because they verify that the platform's cryptographic module which the GovShield Server relies upon is operating correctly, the platform does an integrity check of the GovShield Server's software, and that the Windows Server 2022 platform performs self-tests which confirm that the platform's own functionality as well as its underlying hardware's functionality do not have any anomalies that would cause the TOE's software to be executed in an unpredictable or inconsistent manner.

8.6.5 **[MDMPP] FPT_TUD_EXT.1**

[SERVER] The GovShield Server software updates are acquired by the customer through their General Dynamics sales representatives. The GovShield Server software update provided by General Dynamics is digitally signed during the software build process. The GovShield Server software updates are installed by the Authorized Administrator through the underlying platform directly onto the system; the platform does not have an automatic method of pulling down or installing the updates without Authorized Administrator

(local authorized administrator of the platform) initiation via the platform. The Authorized Administrator accomplishes this by stopping the JBoss EAP service, replacing the GovShield Server software file, and starting the JBoss EAP service. When starting the JBoss EAP service, the platform is invoked which will verify the digital signature of the GovShield Server software update. A successful verification of the digital signature will result in the GovShield Server starting and the installation being complete, and a failed verification of the digital signature will prevent the completion of the installation and the GovShield Server will not start. The before and after version numbers of the GovShield Server software can be checked by clicking on the cog button and "About" button, after authenticating to the web GUI. The GovShield Client's current software version running on an Android Mobile Device can also be queried through the web GUI by an Authorized Administrator. The Authorized Administrator would select the Android Mobile Device on the 'Device Management' dashboard, click the 'Edit' button, and locate the 'GovShield' application in the 'Installed Applications' list.

[AGENT] The GovShield Client software updates are acquired by the customer through their General Dynamics sales representatives. The GovShield Client software update provided by General Dynamics is digitally signed during the software build process. Initiating an update to the GovShield Client software is implemented through the GovShield Server's MAS Server functionality. After authenticating to the web GUI, an Authorized Administrator can upload software updates to the GovShield Client software through the 'APK Management' dashboard. Section 8.5.3 describes the process for managing all applications, including updates to the GovShield Client software, through the GovShield Server's MAS Server functionality and describes the process of uploading the software update, assigning the software update to one or more policies, and then the GovShield Client downloading the software update. Once the update is downloaded onto the Android Mobile Device, the GovShield Client will invoke the platform's application installation process which will verify the software update's digital signature before installing the GovShield Client software update. Only a successful verification of the digital signature will result in the installation of the update to the GovShield Client software, and a failed verification will result in the update process being stopped.

8.7 TOE Access

8.7.1 **[MDMPP] FTA_TAB.1**

[SERVER] The GovShield Server displays a configurable consent warning banner on the web GUI's login page. An Administrator must click the 'ACCEPT' button displayed with the consent warning banner to gain access to the remaining functions of the login page. The consent warning banner can be configured through the web GUI on the 'System Configuration' dashboard by an Authorized Administrator.

[AGENT] The GovShield Client displays a configurable consent warning message within the GovShield Client's login page. An MD user will view the consent warning message banner as a notification before accessing the remaining functions of the GovShield Client's login page. The consent warning message banner can be configured by an Authorized Administrator through the GovShield Server's web GUI on the 'Policies' dashboard under the 'MDM Settings' within the 'App Use Consent Message' field.

8.8 Trusted Path/Channels

8.8.1 **[MDMPP] FTP_ITC_EXT.1**

[SERVER] The TOE has a communication channel between the GovShield Server and the GovShield Client on an Android Mobile Device. The GovShield Server relies on its underlying platform to protect all data from disclosure and modification transferred over this internal communication channel. This communication channel is established once an Android Mobile Device has been fully enrolled into management, is logically distinct from other communication channels, and is specified under FPT_ITT.1(2).

[AGENT] The GovShield Client is internal to the TOE. The GovShield Client relies on its underlying platform to protect all data from disclosure and modification transferred over this internal communication channel.

8.8.2 **[MDMPP] FTP_ITC.1(1)**

[SERVER] The GovShield Server invokes its underlying platform to communicate with third-party systems that reside in the Operational Environment via trusted channels. In the evaluated configuration, the GovShield Server connects with the Oracle Database, which also operates as the audit server, using TLS v1.2 to protect the audit data, authentication data, and TOE configuration data that traverses the channel.

The use of these protocols to establish trusted channels ensures that data in transit will be protected and not subjected to unauthorized modification or disclosure. During TLS session establishment, the GovShield Server's platform will validate the third-party systems' presented X.509v3 certificate to validate their identity. If the third-party system is configured for mutual authentication, the GovShield Server's identity is validated through its X.509v3 certificate presented during TLS session establishment. In the evaluated configuration, the GovShield Server's platform will be configured to allow only the following ciphersuite: TLS_RSA_WITH_AES_256_GCM_ SHA384 as defined in RFC 5288.

The MAS Server is logically integrated with the GovShield Server. As a result of this, all remote communications that the MAS Server requires to operate are identical to those described above.

8.8.3 **[MDMPP] FTP_TRP.1(1)**

[SERVER] The GovShield Server's platform uses HTTPS/TLS (TLS v1.2) to provide a trusted communication between the GovShield Server and Administrators attempting to connect (i.e., invoking the platform) to the web GUI for the purposes of remote administration. These protocols are used to protect the data traversing the channel from disclosure and/or modification. The GovShield Server's identity is validated through its X.509v3 certificate presented during TLS session establishment and the Administrator's identity is validated through their authentication credentials presented to the GovShield Server through the web GUI's login page. In the evaluated configuration, the GovShield Server's platform will be configured to allow only the following ciphersuite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289.

8.8.4 **[MDMPP] FTP_TRP.1(2)**

[SERVER] The GovShield Server's platform uses HTTPS/TLS (TLS v1.2) to provide a trusted communication between the GovShield Server and users. These protocols are used to protect the data

traversing the channel from disclosure and/or modification. This communication path is invoked by users (i.e., an MD user or an Authorized Administrator) for the purposes of enrolling an Android Mobile Device through the GovShield Client. The GovShield Server's identity is validated through its X.509v3 certificate presented during TLS session establishment and the user's identity is validated through their authentication credentials presented to the GovShield Server. In the evaluated configuration, the GovShield Server's platform will be configured to allow only the following ciphersuite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289.

[AGENT] The GovShield Client relies on its underlying platform to provide the HTTPS/TLS communication path and to validate the GovShield Server's X.509v3 certificate. After completing the enrollment of a GovShield Client into the TOE, the initial connection handled by the communication path described above is closed and all future communication between the GovShield Client and the GovShield Server is governed by the internal channel described under the FPT_ITT.1(2) requirement.