Microsoft

Common Criteria Evaluation

Microsoft Intune

Security Target

Document Information	
Version Number	1.4
Updated On	December 30, 2024

Version History

Version	Date	Summary of changes
1.0	November 8, 2024	Release for Check Out
1.1	December 6, 2024	Address ECR Comments
1.2	December 17, 2024	Update Annex C
1.3	December 20, 2024	Update Section 6.2.4 – Cryptographic Operations
1.4	December 30, 2024	Update Section 6.2.4 Table 19

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

TABLE OF CONTENTS

LIST OF TABLES

1 Security Target Introduction

This section presents the following information required for a Common Criteria (CC) evaluation:

- Identifies the Security Target (ST) and the Target of Evaluation (TOE)
- Specifies the ST conventions
- Describes the organization of the ST

1.1 ST Reference

ST Title: Microsoft Intune Security Target

ST Version: version 1.4, December 30, 2024

1.2 TOE Reference

Target of Evaluation: Microsoft Intune (2411), https://intune.microsoft.com
Microsoft Company Portal App (for Android), v5.0.6375.0 (7015796)

1.3 TOE Overview

This ST defines the Microsoft Intune TOE for the purposes of CC evaluation.

Microsoft Intune is a cloud-based service that focuses on mobile device management (MDM) and mobile application management (MAM). Administrators can control how their organization's mobile devices are used and also configure specific policies to control applications installed on the devices. There is only one Intune service (version) that is continuously offered with rolling feature enhancements which is located at the URL described in section 1.2. Intune is part of Microsoft's Enterprise Mobility + Security (EMS) suite.

The Microsoft Intune Company Portal is a mobile device management agent.

1.3.1 TOE Type

Microsoft Intune is a mobile device management system that includes MDM Server and Mobile Application Store (MAS) functionality. The Microsoft Company Portal app provides the MDM agent for Android devices.

1.3.2 TOE Usage

The TOE is used and/or managed by the following roles:

a) **Intune Tenant Administrators.** Enterprise IT administrators that manage users and configure policies for enrolled mobile devices. These administrators use a web browser to access the Intune management interface: Microsoft Intune Admin Center.

b) **Mobile Device Users.** Enterprise users whose mobile device is managed by Microsoft Intune. Usage is dependent on the type of device being managed:

   i) **Android Devices.** The Microsoft Intune Company Portal App is installed to provide the MDM agent to Android devices.

   ii) **iOS Devices.** The Microsoft Intune Company Portal App is installed to facilitate convenient device enrollment however all MDM agent functionality is provided by the native iOS agent.

c) **Microsoft Service Operators.** Microsoft personnel who administer the TOE platform which includes the Intune cloud service and underlying Azure infrastructure.

The table below identifies each user and administrator role within the Intune MDM system and correlates the applicable TOE element in which they interact.

Role	Privilege	MDM Server	MDM Agent	Azure Platform
Mobile Device Users	User		X	
Intune Tenant Administrators	Admin	X		
Microsoft Service Operators	Admin			X

1.3.3 TOE Use Case

Microsoft Intune supports the needs and use cases of many different types of organizations. This Security Target meets the Enterprise Device High Security use case as defined in Appendix G of the MDM PP. Deviations from the use case template selections are identified in Annex A of this ST.

1.3.4 TOE Security Services

This section summarizes the security services provided by the TOE:

- **Security Audit:** Microsoft Intune has the ability to generate, review, protect, and restrict access to audit and event logs as required by the MDM PP and MDM Agent PP-Module. Audit information generated by the system includes the date and time of the event, the user identity that caused the event to be generated, and other event specific data. Authorized administrators can review audit logs and have the ability to search and sort audit records securely via the Microsoft Intune Admin Center console or via the Graph API. In the context of this evaluation, the protection profile requirements cover generating audit events, which events should be audited, and providing secure storage for audit event entries.
- **Cryptographic Support:** Microsoft Intune provides cryptographic functions that support encryption/decryption, cryptographic signatures, cryptographic hashing, cryptographic key agreement, and random number generation. The TOE additionally provides support for X.509 certificates including validation functions. Certificates are issued during device enrollment and are used for authentication and protection of both user and system data while in transit.

- **Identification and Authentication** Each Microsoft Intune administrator must be identified and authenticated based on administrator-defined policy prior to performing any TSF-mediated functions. An interactive user invokes a trusted path to protect their identity and credentials. Microsoft Intune maintains databases of accounts including their identities, authentication information, group associations, and administrative privileges. Microsoft Intune provides the ability to use, store, and protect X.509 certificates that are used for mobile devices. Communications between the Mobile device and Intune are facilitated with authenticated TLS sessions.
- **Security Management:** Microsoft Intune includes several functions to manage security policies on registered devices. Microsoft Intune MDM policy functions include the ability to define the minimum password length, the number of failed logon attempts, the duration of lockout, and password age. MDM Policy management is available to Intune administrators that have sufficient permissions or are members of an applicable role-based group as described in FMT_SMR.1. Successfully enrolled devices are issued an X.509 certificate that is used for identification and authentication.
- **Protection of the TOE Security Functions**: Microsoft Intune provides several features to ensure the protection of TOE security functions. Intune protects against unauthorized data disclosure and modification by requiring authenticated TLS sessions between registered devices and Intune. All Intune components employ self-testing features on start-up that ensure the integrity of executable code and any cryptographic functions.
- **Trusted Path/Channels for Communications**: Microsoft Intune uses TLS and HTTPS to provide a trusted path for communications between Intune and remote administrators as well as registered devices. Trusted channels provided by Intune include the Microsoft Intune Admin Center for Administrator use via HTTPS, and a X.509 authenticated TLS channel for device enrollment and continual policy updates.

1.3.5 Non-TOE Components

The TOE operates with the following components in the environment:

a) **Mobile Devices.** The TOE supports Android and iOS based mobile devices and was tested with:

i. iOS 15 executing on the following hardware:

- iPhone 12

ii. Android 13 executing on the following hardware:

- Samsung Galaxy S21 Ultra 5G

iii. Android 11 executing on the following hardware:

- Google Pixel 4a 5G

Note: The Intune service also supports the management of devices listed at https://learn.microsoft.com/en-us/mem/intune/fundamentals/supported-devices-browsers however these devices were not tested as part of this evaluation.

b) **Audit Server.** The TOE can send audit events to any endpoint that is able to use Microsoft Graph API via HTTPS.

1.4 TOE Architecture Description

The TOE is a mobile device management system. Consistent with the MDM PP TOE Boundary[1], Microsoft Intune is a product natively hosted on the Microsoft Azure cloud architecture which provides the network environment. Each customer controls a single, logically isolated tenant that is assigned a unique domain and Tenant ID in the Intune service. Figure 1 below depicts the TOE architecture.

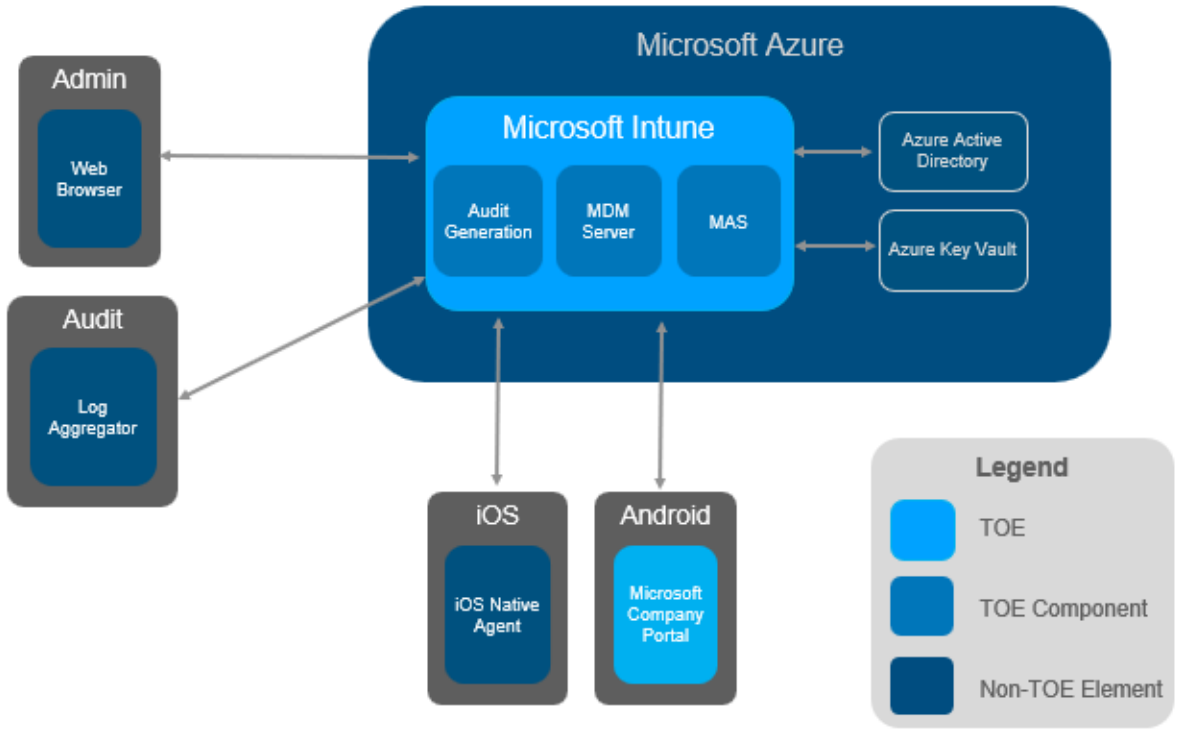

Figure 1: Simple TOE Architecture

[1] Per section 1.3.1 of the MDM PP "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 TOE is comprised of the following elements:

a) **Microsoft Intune.** Includes the MDM server, Mobile Application Store (MAS), and Audit service components that together deliver the Intune TOE. The TOE platform is the Microsoft Windows Server OS provided by Azure. All communication with Intune is secured via TLS/HTTPS.

b) **Microsoft Company Portal (Android App).** The application which provides the MDM agent for Android and enforces Intune policies on the mobile device. The agent platform is the Android mobile device OS.

The TOE environment also includes the following non-TOE elements:

a) **Microsoft Azure.** Microsoft's cloud platform that provides a trusted environment for Intune and Active Directory as described in section 1.4 in addition to hosting Azure Key Vault, the platform provided service for securely storing certificates and cryptographic keys.

b) **Azure Active Directory.** Provides authentication services for administrators and tenant users.

c) **Admin.** Administration of Microsoft Intune is provided by the Microsoft Intune Admin Center via web browser.

d) Audit. Microsoft Intune can provide audit events to an external log aggregation service / audit server over HTTPS via the Graph API.

e) **iOS Native Agent.** The Intune TOE supports Apple iOS devices via the native iOS MDM agent. **Note:** Since the iOS agents are evaluated as part of the Apple iOS evaluations, Intune was tested to ensure it can manage those devices, but the agent's behavior on those devices was not otherwise tested.

1.4.1 Evaluated Configuration Guidance

The TOE includes the following documents for providing guidance on achieving the evaluated configuration:

a) Microsoft Intune Operational and Administrative Guidance, Version 1.2

Within this ST, when specifically referring to a type of TSF, the TSF type will be explicitly stated. Otherwise, the term TSF refers to the total of all TSFs within the TOE.

1.4.2 Security Environment and TOE Boundary

1.4.2.1 Logical Boundaries

Conceptually the Microsoft Intune TOE can be thought of as a collection of the following security services which the ST describes with increasing detail:

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TOE Security Functions

- Trusted Path and Channels

1.4.2.2 Physical Boundaries

Microsoft Intune is the service available at https://intune.microsoft.com

Microsoft Intune is built on Azure Virtual Machines with the following underlying platform components[2]:

a) **Operating System.** Microsoft Windows Server comprising of the following versions:

i. **Microsoft Windows Server 2019 Datacenter**
(CCN-CC-12/2020[3], Certification Date: 2020-06-05)

Microsoft Intune Company Portal is the application installed on mobile devices to facilitate enrollment and command execution. The TOE was tested on the following mobile device platforms:

a) Android 13 (*VID11342[4]*) with Microsoft Intune Company Portal version 5.0.6375.0 (7015796)

b) Android 11 (*VID11124[5]*) with Microsoft Intune Company Portal version 5.0.6375.0 (7015796)

Note: On Apple iOS-based devices, the Microsoft Intune Company Portal App utilizes the native iOS MDM agent to handle all MDM functionality including enrollment, retire, device wipe, and other policy enforcement commands issued by Intune. Intune Company Portal App version 5.2409.0 (53.2409829.001) on iOS 15 was tested only to verify Intune can successfully manage those devices.

The administrator and user must follow the instructions in the *Microsoft Intune Operational and Administrative Guidance, v1.0* to configure and remain in the evaluated configuration.

1.5 Product Description

Microsoft Intune is a cloud-based service that focuses on mobile device management (MDM) and mobile application management (MAM). Microsoft Intune is used to control organization devices and how they are used. Microsoft Intune allows an administrator to configure specific policies that govern access to corporate data, application usage, and enforcement of security features and functions on the device whether it is a personal BYOD device, or corporate issued device. Microsoft Intune is part of Microsoft's Enterprise Mobility + Security (EMS) suite.

1.6 Conventions, Terminology, Acronyms

This section specifies the formatting information used in the security target.

1.6.1 Conventions

Where applicable the following conventions are used to identify operations:

[2] Per MDM PP ALC_CMS.1.1E Application Note: In cases where the MDM software is Software as a Service, running in a cloud environment where they have little to no control of the operating system and underlying hardware, the evaluated configuration is considered a snapshot of the MDM software with the OS and/or VM versions used at the time of testing.
[3] CCRA certificate identifier for *Microsoft Windows 10 version 1909 and Microsoft Windows Server version 1909*, Certification Date: 2020-06-05, https://www.commoncriteriaportal.org/nfs/ccpfiles/files/epfiles/2019-47-ST-lite.pdf
[4] NIAP validation identifier for *Samsung Electronics Co., Ltd. Samsung Galaxy Devices on Android 13 – Spring*, Certification Date: 04/26/2023
[5] NIAP validation identifier for *Google Pixel Phones on Android 11.0*, Certification Date: 02/08/2021

- **Iteration**: Iterated requirements (components and elements) are identified with letter following the base component identifier. For example, iterations of FMT_MOF.1 are identified in a manner similar to FMT_MOF.1(Audit) (for the component) and FCS_COP.1.1(Audit) (for the elements).
- **Assignment**: Assignments are identified in brackets and bold (e.g., **[assigned value]**).
- **Selection**: Selections are identified in brackets, bold, and italics (e.g., ***[selected value]***).
    - Assignments within selections are identified using the previous conventions, except that the assigned value would also be italicized and extra brackets would occur (e.g., ***[selected value [assigned value]]***).
- **Refinement**: Refinements are identified using bold text (e.g., **added text**) for additions and strike-through text (e.g., ~~deleted text~~) for deletions.

1.6.2 Terminology

The following terminology is used in the ST:

Term	Definition
CC	Common Criteria
EAL	Evaluation Assurance Level
EMS	Microsoft Enterprise Mobility + Security
Graph API	Gateway for programmatic access to data in Enterprise Mobility + Security
MAM	Mobile Application Management
MDM	Mobile Device Management
PP	Protection Profile
ST	Security Target
Tenant	An instance of Azure Active Directory (Azure AD)
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSS	TOE Summary Specification

1.6.3 Acronyms

The acronyms used in this security target are specified in Section 1.6.2.

1.7 ST Overview and Organization

This ST contains the following additional sections:

- CC Conformance Claims (Section 2): Formal conformance claims which are examined during the evaluation.
- Security Problem Definition (Section 3): Describes the threats, organizational security policies and assumptions that pertain to the TOE.
- Security Objectives (Section 4): Identifies the security objectives that are satisfied by the TOE and the TOE operational environment.
- Security Requirements (Section 5): Presents the security functional and assurance requirements met by the TOE.

- TOE Summary Specification (TSS) (Section 6): Describes the security functions provided by the TOE to satisfy the security requirements and objectives.
- Protection Profile Conformance Claim (Section 7): Presents the rationale concerning compliance of the ST with the ***Protection Profile for Mobile Device Management*** and ***PP-Module for MDM Agents.***
- Rationale (Section 8): Presents the rationale for the security objectives, requirements, and TOE Summary Specification as to their consistency, completeness and suitability.
- Annex A: Presents the security management function requirements for conformance to the G.2 enterprise device high security use case.

2 CC Conformance Claims

This ST and the Microsoft Intune TOE are consistent with the following specifications:

a) Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, Version 3.1, Revision 5, April 2017, extended (Part 2 extended)
b) Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements Version 3.1, Revision 5 April 2017, (Part 3 conformant)
c) NIAP PP-Configuration for Mobile Device Management (MDM) and MDM Agents, v1.0
    a. Base PP: Protection Profile for Mobile Device Management, Version 4.0, April 25, 2019 (PP_MDM_V4.0)
    b. PP-Module: NIAP PP-Module for MDM Agents, version 1.0, April 25, 2019 (MDM Agent PP-Mod) (MOD_MDM_AGENT_V1.0)

d) NIAP Technical Decisions per Table 1

Table 1 NIAP Technical Decisions

TD #	Name	Source	Applicability Rationale
NIAP Protection Profile for Mobile Device Management v4.0 (PP_MDM_V4.0)			
TD0438	TST and TUD on the MDM Agent	PP_MDM_V4.0	Applicable
TD0461	Security Audit for Distributed TOEs	PP_MDM_V4.0	Applicable
TD0462	MDM Distributed TOE: Registration Channel Updates	PP_MDM_V4.0	Not Applicable – FCO_CPC_EXT.1 not claimed.
TD0479	FMT_SMF.1(1) Reliance on MDF Evals	PP_MDM_V4.0	Applicable
TD0552	SFR Rationale and Implicitly Satisfied SFRs	PP_MDM_V4.0	Applicable
TD0594	Distributed TOE tests in FCO_CPC_EXT.1.3	PP_MDM_V4.0	Not Applicable – FCO_CPC_EXT.1 not claimed.

TD0600	Conformance claim sections updated to allow for MOD_VPNC_V2.3	PP_MDM_V4.0	Applicable
TD0616	MDM PP Use Case Mappings	PP_MDM_V4.0	Applicable
TD0629	Audit Events for Startup and Shutdown	PP_MDM_V4.0	Applicable
TD0641	Alternative revocation checking for MDM	PP_MDM_V4.0	Applicable
TD0650	Conformance claim sections updated to allow for MOD_VPNC_V2.3 and 2.4	PP_MDM_V4.0	Applicable
TD0754	MDM Policy Authenticity	PP_MDM_V4.0	Applicable
TD0784	Terminology Change in MDMPP: Extended to Functional Package	PP_MDM_V4.0	Applicable
TD0844	Addition of Assurance Package for Flaw Remediation V1.0 Conformance Claim	PP_MDM_V4.0	Not Applicable to this Security Target
TD0887	Management of x509 certificates for cloud	PP_MDM_V4.0	Applicable
TD0895	Third Party Libraries in FPT_LIB_EXT.1.1	PP_MDM_V4.0	Applicable
NIAP PP-Module for MDM Agents v1.0 (MOD_MDM_AGENT_V1.0)			
TD0491	Update to FMT_SMF_EXT.4 Test 2	MOD_MDM_AGENT_V1.0	Superseded by TD0755
TD0497	SFR Rationale, Consistency of SPD, and Implicitly Satisfied SFRs	MOD_MDM_AGENT_V1.0	Applicable
TD0600	Conformance claim sections updated to allow for MOD_VPNC_V2.3	MOD_MDM_AGENT_V1.0	Applicable
TD0650	Conformance claim sections updated to allow for MOD_VPNC_V2.3 and 2.4	MOD_MDM_AGENT_V1.0	Applicable
TD0660	Mislabeled SFRs in MDM Agent Auditable Events Table	MOD_MDM_AGENT_V1.0	Applicable
TD0673	MDM-Agent PP-Module updated to allow for new PP and PP-Module Versions	MOD_MDM_AGENT_V1.0	Applicable
TD0755	MDM-Agent Policy Authenticity	MOD_MDM_AGENT_V1.0	Applicable

3 Security Problem Definition

The security problem definition consists of the threats to security, organizational security policies, and usage assumptions as they relate to Windows. The assumptions, threats, and policies are copied from the Protection Profile for Mobile Device Management, Version 4.0, April 25, 2019 (PP_MDM_V4.0) and the PP-Module for MDM Agents, Version 1.0, April 25, 2019 (MDM Agent PP-Mod).

3.1 Threats to Security

Table 2 presents known or presumed threats to protected resources that are addressed by Microsoft Intune based on conformance to the Protection Profile for Mobile Device Management.

Table 2 Threats (PP_MDM_V4.0)

Threat Identifier	Description
T.MALICIOUS_APPS	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.NETWORK_ATTACK	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	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	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.

Table *3* presents known or presumed threats to protected resources that are addressed by Microsoft Intune based on conformance to the PP-Module for MDM Agents.

Table 3 Threats (MDM Agent PP-Mod)

Threat Identifier	Description
T.BACKUP	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.

3.2 Organizational Security Policies

An organizational security policy is a set of rules or procedures imposed by an organization upon its operations to protect its sensitive data and IT assets. Table 4 and **Table *5*** describe organizational security policies which are necessary for conformance to the protection profile.

Table 4 Organizational Security Policies (PP_MDM_V4.0)

Security Policy Identifier	Description
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.

Table 5 Organizational Security Policies (MDM Agent PP-Mod)

Security Policy Identifier	Description
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.

3.3 Secure Usage Assumptions

Table 6 and **Table *7*** describe the core security aspects of the environment in which Microsoft Intune is intended to be used. It includes information about the physical, personnel, procedural, and connectivity aspects of the environment.

The following specific conditions are assumed to exist in an environment where the TOE is employed in order to conform to the protection profile:

Table 6 Assumptions (PP_MDM_V4.0)

Assumption	Description
A.COMPONENTS_ RUNNING (applies to distributed TOEs only)	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	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.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.

Assumption	Description
A.PROPER_USER	Mobile device users are not willfully negligent or hostile and use the device within compliance of a reasonable Enterprise security policy.

Table 7 Assumptions (MDM Agent PP-Mod)

Assumption	Description
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.MOBILE_DEVICE_ PLATFORM	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 Security Objectives

This section defines the security objectives for Microsoft Intune and its supporting environment. Security objectives, categorized as either TOE security objectives or objectives by the supporting environment, reflect the stated intent to counter identified threats, comply with any organizational security policies identified, or address identified assumptions. All of the identified threats, organizational policies, and assumptions are addressed under one of the categories below.

4.1 TOE Security Objectives

Table 8 describes the security objectives for Microsoft Intune which are needed to comply with the PP_MDM_V4.0.

Table 8 Security Objectives for the TOE (PP_MDM_V4.0)

Security Objective	Source
O.ACCOUNTABILITY	The TOE must provide logging facilities which record management actions undertaken by its administrators. Addressed by: FAU_ALT_EXT.1, FAU_GEN.1(1), FAU_GEN.1(2) (SEL-BASED), FAU_NET_EXT.1, FAU_SAR.1 (OPTIONAL), FAU_SEL.1 (OPTIONAL), FAU_STG_EXT.1, FAU_STG_EXT.2 (SEL-BASED)
O.APPLY_POLICY	The TOE must facilitate configuration and enforcement of enterprise security policies on mobile devices via interaction with

Security Objective	Source
	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. Addressed by: FIA_ENR_EXT.1, FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3) (SEL-BASED), FMT_SAE_EXT.1 (OBJECTIVE), FMT_SMF.1(1), FMT_SMF.1(2), FMT_SMF.1(3) (SEL-BASED).
O.DATA_PROTECTION_TRANSIT	Data exchanged between the MDM Server and the MDM Agent must be protected from being monitored, accessed, or altered. Addressed by: FAU_STG_EXT.1, FCS_CKM_EXT.4, FCS_CKM.1, FCS_CKM.2, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_HTTPS_EXT.1 (SEL-BASED), FCS_IV_EXT.1 (SEL-BASED), FCS_RBG_EXT.1, FCS_STG_EXT.1, FCS_STG_EXT.2 (SEL-BASED), FIA_X509_EXT.1(1), FIA_X509_EXT.1(2) (SEL-BASED), FIA_X509_EXT.2, FIA_X509_EXT.3 (OBJECTIVE), FIA_X509_EXT.4 (OBJECTIVE), FIA_CLI_EXT.1, FPT_ITT.1(1) (SEL-BASED), FPT_ITT.1(2) (SEL-BASED), FTP_ITC_EXT.1, FTP_ITC.1(1), FTP_ITC.1(2) (SEL-BASED), FTP_TRP.1(1), FTP_TRP.1(2), FTP_TRP.1(3) (OBJECTIVE).
O.INTEGRITY	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. Addressed by: FIA_X509_EXT.2, FCO_CPC_EXT.1 (OBJECTIVE), FPT_TST_EXT.1, FPT_TUD_EXT.1
O.MANAGEMENT	The TOE provides access controls around its management functionality. Addressed by: FAU_CRP_EXT.1 (OBJECTIVE), FIA_ENR_EXT.1, FIA_UAU.1, FIA_UAU_EXT.4(1) (OBJECTIVE), FIA_UAU_EXT.4(2) (OBJECTIVE), FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3) (SEL-BASED), FMT_POL_EXT.1, FMT_SMF.1(1), FMT_SMF.1(2), FMT_SMF.1(3) (SEL-BASED), FMT_SMR.1(1), FMT_SMR.1(2) (SEL-BASED), FTA_TAB.1 (OPTIONAL).
O.QUALITY	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. Addressed by: FPT_API_EXT.1, FPT_LIB_EXT.1

Table *9* describes the security objectives for Microsoft Intune which are needed to comply with the MDM Agent PP-Module.

Table 9 Security Objectives for the TOE (MDM Agent PP-Mod)

Security Objective	Source
O.ACCOUNTABILITY	The TOE must provide logging facilities, which record management actions undertaken by its administrators. Addressed by: FAU_ALT_EXT.2, FAU_GEN.1(2), FAU_SEL.1(2)
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. Addressed by: FAU_STG_EXT.3 (objective), FIA_ENR_EXT.2, FMT_POL_EXT.2, FMT_SMF_EXT.4, FMT_UNR_EXT.1
O.DATA_PROTECTION_TRANSIT	Data exchanged between the MDM Server and the MDM Agent must be protected from being monitored, accessed, or altered. Addressed by: FCS_DTLSS_EXT.1 (from TLS Package), FCS_DTLSC_EXT.1 (from TLS Package), FCS_TLSC_EXT.1 (from TLS Package), FCS_TLSC_EXT.2 (from TLS Package), FCS_TLSS_EXT.1 (from TLS Package), FCS_TLSS_EXT.2 (from TLS Package), FPT_ITT.1(2) (if MDM is Base-PP), FPT_NET_EXT.1 (objective), FTP_ITC_EXT.1(2) (if MDF is Base-PP), FTP_TRP.1(2) (if MDF is Base-PP)
O.STORAGE	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. Addressed by: FCS_STG_EXT.1(2) (if MDM is Base-PP), FCS_STG_EXT.4 (if MDF is Base-PP)

4.2 Security Objectives for the Operational Environment

The TOE is assumed to be complete and self-contained and, as such, is not dependent upon any other products to perform properly. However, certain objectives with respect to the general operating

environment must be met. Table 10 and **Table 11** describe the security objectives for the operational environment as specified in the protection profile.

Table 10 Security Objectives for the Operational Environment (PP_MDM_V4.0)

Environment Objective	Description
OE.COMPONENTS_RUNNING	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.TIMESTAMP	Reliable timestamp is provided by the operational environment for the TOE.
OE.WIRELESS_NETWORK	A wireless network will be available to the mobile devices.

Table 11 Security Objectives for the Operational Environment (MDM Agent PP-Mod)

Environment Objective	Description
OE.DATA_PROPER_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
OE.DATA_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	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.WIRELESS_NETWORK	A wireless network will be available to the mobile devices.

4.3 Security Objectives Rationale

Rationale on how the assumptions, threats, and organization security policies map to the security objectives and specific SFRs is described in Section 8 and further in the MDM PP and MDM Agent PP-Module.

5 Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) for the TOE. The requirements in this section have been drawn from the Protection Profile for Mobile Device Management, Version 4.0, April 25, 2019 (PP_MDM_V4.0) and the PP-Module for MDM Agents, Version 1.0, April 25, 2019 (MDM Agent PP-Mod).

Conventions:

Where requirements are drawn from the protection profile, the requirements are copied verbatim, except for some changes to required identifiers to match the iteration convention of this document, from that protection profile and only operations performed in this security target are identified.

The extended requirements, extended component definitions and extended requirement conventions in this security target are drawn from the protection profile; the security target reuses the conventions from the protection profile which include the use of the word "Extended" and the "_EXT" identifier to denote extended functional requirements. The security target assumes that the protection profile correctly defines the extended components and so they are not reproduced in the security target.

Where applicable the following conventions are used to identify operations:

- **Iteration**: Iterated requirements (components and elements) are identified with letter following the base component identifier. For example, iterations of FMT_MOF.1 are identified in a manner similar to FMT_MOF.1(Audit) (for the component) and FCS_COP.1.1(Audit) (for the elements).
- **Assignment**: Assignments are identified in brackets and bold (e.g., **[assigned value]**).
- **Selection**: Selections are identified in brackets, bold, and italics (e.g., ***[selected value]***).
    - Assignments within selections are identified using the previous conventions, except that the assigned value would also be italicized and extra brackets would occur (e.g., ***[selected value [assigned value]]***).
- **Refinement**: Refinements are identified using bold text (e.g., **added text**) for additions and strike-through text (e.g., ~~deleted text~~) for deletions.

5.1 TOE Security Functional Requirements

This section specifies the SFRs for the TOE.

Table 12 TOE Security Functional Requirements for PP_MDM_V4.0

Requirement Class	Requirement Component
Security Audit (FAU)	FAU_ALT_EXT.1 Server Alerts
	FAU_GEN.1(1) Audit Data Generation
	FAU_GEN.1(2) Audit Generation (MAS Server)
	FAU_NET_EXT.1 Network Reachability Review
	FAU_STG_EXT.1 External Trail Storage
	FAU_STG_EXT.2 Audit Event Storage
	FCS_CKM.1 Cryptographic Key Generation

Requirement Class	Requirement Component
Cryptographic Support (FCS)	FCS_CKM.2 Cryptographic Key Establishment
	FCS_CKM_EXT.4 Cryptographic Key Destruction
	FCS_COP.1(1) Cryptographic Operation (Confidentiality Algorithms)
	FCS_COP.1(2) Cryptographic Operation (Hashing Algorithms)
	FCS_COP.1(3) Cryptographic Operation (Signature Algorithms)
	FCS_COP.1(4) Cryptographic Operation (Keyed-Hash Message Authentication)
	FCS_RBG_EXT.1 Extended: Random Bit Generation
	FCS_RBG_EXT.1/ANDROID Extended: Random Bit Generation (Android)
	FCS_STG_EXT.1 Cryptographic Key Storage
Identification & Authentication (FIA)	FIA_ENR_EXT.1 Enrollment of Mobile Device into Management
	FIA_UAU.1 Timing of Authentication
	FIA_X509_EXT.1(1) X.509 Certificate Validation
	FIA_X509_EXT.2 X.509 Certificate Authentication
	FIA_CLI_EXT.1 Client Authorization
Security Management (FMT)	FMT_MOF.1(1) Management of Functions Behavior
	FMT_MOF.1(2) Management of Functions Behavior (Enrollment)
	FMT_MOF.1(3) Management of Functions in (MAS Server Downloads)
	FMT_POL_EXT.1 Trusted Policy Update
	FMT_SMF.1(1) Specification of Management Functions (Server configuration of Agent)
	FMT_SMF.1(2) Specification of Management Functions (Server Configuration of Server)
	FMT_SMF.1(3) Specification of Management Functions (MAS Server)
	FMT_SMR.1(1) Security Management Roles
	FMT_SMR.1(2) Security Management Roles (MAS Server)
Protection of the TSF (FPT)	FPT_API_EXT.1 Use of Supported Services and APIs
	FPT_ITT.1(2) Internal TOE TSF Data Transfer (MDM Agent)
	FPT_LIB_EXT.1 Use of Third Party Libraries
	FPT_TST_EXT.1 Functionality Testing
	FPT_TUD_EXT.1 Trusted Update
Trusted Path/Channels (FTP)	FTP_ITC_EXT.1 Trusted Channel
	FTP_ITC.1(1) Inter-TSF Trusted Channel (Authorized IT Entities)
	FTP_ITC.1(2) Inter-TSF Trusted Channel (MDM Agent)
	FTP_TRP.1(1) Trusted Path (for Remote Administration)
	FTP_TRP.1(2) Trusted Path (for Enrollment)

Table 13 TOE Security Functional Requirements for MDM Agent PP-Mod

Requirement Class	Requirement Component
Security Audit (FAU)	FAU_ALT_EXT.2 Agent Alerts
	FAU_GEN.1(2) Audit Data Generation
	FAU_SEL.1(2) Security Audit Event Selection
Cryptographic Support (FCS)	FCS_STG_EXT.1(2) Cryptographic Key Storage

Identification & Authentication (FIA)	FIA_ENR_EXT.2 Agent Enrollment of Mobile Device into Management
Security Management (FMT)	FMT_POL_EXT.2 Agent Trusted Policy Update
	FMT_SMF_EXT.4 Specification of Management Functions
	FMT_UNR_EXT.1 User Unenrollment Prevention

5.1.1 Security Audit (FAU)

5.1.1.1 Security Audit For MDM PP

5.1.1.1.1 Server Alerts (FAU_ALT_EXT.1)

FAU_ALT_EXT.1.1 The TSF shall alert the administrators in the event of any of the following:

- Change in enrollment status
- Failure to apply policies to a mobile device
- [***No other events***]

5.1.1.1.2 Audit Data Generation (FAU_GEN.1(1))

FAU_GEN.1.1(1) **Refinement:** The TSF shall [***invoke platform-provided functionality, implement functionality***] to generate an audit record of the following auditable events:

- All administrative actions
- [***Commands issued to the MDM Agent***]
- Specifically defined auditable events listed in ~~Table 2~~ **Table 14**
- [***no other events***].

Application Note: This SFR has been modified by TD0629.

Application Note: This SFR has been modified by TD629

FAU_GEN.1.2(1) The TSF shall record within each TSF audit record at least the following information:

- 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 2~~ **Table 14**
- [**no other audit relevant information**].

Table 14 Audit Events (PP_MDM_V4.0)

Requirement	Auditable Events	Additional Audit Record Contents	TOE Component[6]
FAU_ALT_EXT.1 (man)	*Type of alert.*	*Identity of Mobile Device that sent alert.*	MDM Server

[6] The MDM system relies on the underlying Azure platform. Each component responsible for **generating** the audit messages identified in the table is listed in the TOE Component column.

Requirement	Auditable Events	Additional Audit Record Contents	TOE Component[6]
FAU_GEN.1(1) (man)	*None.*		~
FAU_GEN.1(2) (sel)	*None.*		~
FAU_NET_EXT.1 (man)	*None.*		~
FAU_STG_EXT.1 (man)	*None.*		~
FAU_STG_EXT.2 (sel)	*None.*		~
FCS_CKM_EXT.4 (man)	*None.*		~
FCS_CKM.1 (man)	*[**None**]*	*No additional information*	~
FCS_CKM.2 (man)	*None.*		~
FCS_COP.1(1) (man)	*None.*		~
FCS_COP.1(2) (man)	*None.*		~
FCS_COP.1(3) (man)	*None.*		~
FCS_COP.1(4) (man)	*None.*		~
FCS_RBG_EXT.1 (man)	*Failure of the randomization process.*	*No additional information.*	MDM Platform
FCS_RBG_EXT.1 /ANDROID (man)	*Failure of the randomization process.*	*No additional information.*	MDM Platform
FCS_STG_EXT.1 (man)	*None.*		~
FIA_ENR_EXT.1 (man)	*Failure of MD user authentication.*	*Presented username.*	MDM Server
FIA_UAU.1 (man)	*None.*		~
FIA_X509_EXT.1(1) (man)	*Failure to validate X.509 certificate*	*Reason for failure.*	MDM Platform
FIA_X509_EXT.2 (man)	*Failure to establish connection to determine revocation status.*	*No additional information.*	MDM Platform
FIA_CLI_EXT.1 (man)	*None.*		~
FMT_MOF.1(1) (man)	*Issuance of command to perform function.*	*Command sent and identity of MDM Agent recipient(s).*	MDM Server
	Change of policy settings.	*Policy changed and value or full policy.*	MDM Agent MDM Server
FMT_MOF.1(2) (man)	*Enrollment by a user.*	*Identity of user.*	MDM Server
FMT_MOF.1(3) (sel)	*None.*		~
FMT_POL_EXT.1 (man)	*None.*		~
FMT_SMF.1(1) (man)	*None.*		~
FMT_SMF.1(2) (man)	*Success or failure of function.*	*No additional information.*	MDM Server
FMT_SMF.1(3) (sel)	*None.*		~
FMT_SMR.1(1) (man)	*None.*		~
FMT_SMR.1(2) (sel)	*None.*		~
FPT_API_EXT.1 (man)	*None.*		~

Requirement	Auditable Events	Additional Audit Record Contents	TOE Component[6]
FPT_ITT.1(2) (sel)	*Initiation and termination of the trusted channel.*	*Trusted channel protocol. Identity of initiator and recipient.*	MDM Platform
FPT_LIB_EXT.1 (man)	*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.*	MDM Platform
FPT_TUD_EXT.1 (man)	*Success or failure of signature verification.*	*No additional information.*	MDM Platform
FTP_ITC.1(1) (man)	*Initiation and termination of the trusted channel.*	*Trusted channel protocol. Non-TOE endpoint of connection.*	MDM Platform
FTP_ITC.1(2) (sel)	*Initiation and termination of the trusted channel.*	*Trusted channel protocol. Non-TOE endpoint of connection.*	MDM Platform
FTP_ITC_EXT.1 (man)	*None.*		~
FTP_TRP.1(1) (man)	*Initiation and termination of the trusted channel.*	*Trusted channel protocol. Identity of administrator.*	MDM Platform
FTP_TRP.1(2) (man)	*Initiation and termination of the trusted channel.*	*Trusted channel protocol.*	MDM Platform

5.1.1.1.3 Audit Generation (MAS Server) (FAU_GEN.1(2))

This selection-based component depends upon selection in FMT_MOF.1.1(1)

FAU_GEN.1.1(2) **Refinement:** 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) **Refinement:** 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
- [**no other audit relevant information**]

5.1.1.1.4 Network Reachability Review (FAU_NET_EXT.1)

FAU_NET_EXT.1.1 The TSF shall provide authorized administrators with the capability to read the network connectivity status of an enrolled agent.

5.1.1.1.5 External Trail Storage (FAU_STG_EXT.1)

FAU_STG_EXT.1.1 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 [***store audit data locally***].

5.1.1.1.6 Audit Event Storage (FAU_STG_EXT.2)

This selection-based component depends upon selection in FAU_STG_EXT.1.1

FAU_STG_EXT.2.1 The TSF shall [***invoke platform-provided functionality***] to protect the stored audit records in the audit trail from unauthorized modification.

5.1.1.2 Security Audit For MDM Agent PP-Module

5.1.1.2.1 Agent Alerts (FAU_ALT_EXT.2)

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.

5.1.1.2.2 Audit Data Generation (FAU_GEN.1(2))

FAU_GEN.1.1(2) **Refinement:** The MDM Agent shall [***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 1~~* **Table 15** *and* [***no other events***]*.*

Table 15 Audit Events (MDM Agent PP-Mod)

Requirement	Auditable Events	Additional Audit Record Contents	TOE Component
FAU_ALT_EXT.2	Success/failure of sending alert.	No additional information.	MDM Agent MDM Server
FAU_GEN.1(2)	None.	N/A	~
FAU_SEL.1(2)	All modifications to the audit configuration that occur while the audit collection	No additional information.	MDM Agent MDM Server

Requirement	Auditable Events	Additional Audit Record Contents	TOE Component
	functions are operating.		
FCS_STG_EXT.4/ FCS_STG_EXT.1(2)	None.		~
			~
FIA_ENR_EXT.2	Enrollment in management.	Reference identifier of MDM Server.	MDM Agent
FMT_POL_EXT.2	Failure of policy validation.	Reason for failure of validation.	MDM Agent
FMT_SMF_EXT.4	Outcome (Success/failure) of function.	No additional information.	MDM Agent
FMT_UNR_EXT.1.1	[***Attempt to unenroll***]	No additional information.	MDM Agent

Application Note: This SFR has been modified by TD0660.

FAU_GEN.1.2(2) **Refinement:** The [***TSF***] 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, **[no other audit relevant information].**

5.1.1.2.3 Security Audit Event Selection (FAU_SEL.1(2))

FAU_SEL.1.1(2) **Refinement:** 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,* **[no other attributes]***].*

5.1.2 Cryptographic Support (FCS)

5.1.2.1 Cryptographic Support for MDM PP

5.1.2.1.1 Cryptographic Key Generation (FCS_CKM.1)

FCS_CKM.1.1 **Refinement:** 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,***
- ***ECC schemes using "NIST curves" P-384 and [P-256, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 ,***

]

5.1.2.1.2 Cryptographic Key Establishment (FCS_CKM.2)

FCS_CKM.2.1 **Refinement:** 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",***

].

5.1.2.1.3 Cryptographic Key Destruction (FCS_CKM_EXT.4)

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],***

    ]
  - ***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]***

    ]

].

FCS_CKM_EXT.4.2 The TSF shall destroy all plaintext keying material and critical security parameters (CSPs) when no longer needed.

5.1.2.1.4 Cryptographic Operation (Confidentiality Algorithms) (FCS_COP.1(1))

FCS_COP.1.1(1) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to perform encryption/decryption in accordance with a specified cryptographic algorithm: [

- ***AES-CBC (as defined in FIPS PUB 197, and NIST SP 800-38A) mode,***

- ***AES-GCM (as defined in NIST SP 800-38D),***
- ***AES Key Wrap (KW) (as defined in NIST SP 800-38F),***
- ***AES-CCM (as defined in NIST SP 800-38C)***

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

5.1.2.1.5 Cryptographic Operation (Hashing Algorithms) (FCS_COP.1(2))

FCS_COP.1.1(2) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to perform cryptographic hashing in accordance with a specified cryptographic algorithm [***SHA-256, SHA-384, SHA-512***] and message digest sizes [***256, 384, 512***] bits that meet the following: FIPS Pub 180-4.

5.1.2.1.6 Cryptographic Operation (Signature Algorithms) (FCS_COP.1(3))

FCS_COP.1.1(3) **Refinement:** 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, "Digital Signature Standard (DSS)", Section 4,***
- ***ECDSA schemes using "NIST curves" P-384 and [P-256, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5***

]

5.1.2.1.7 Cryptographic Operation (Keyed-Hash Message Authentication) (FCS_COP.1(4))

FCS_COP.1.1(4) **Refinement:** 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, SHA-512],*** key sizes [**128 and 256 bits**], and message digest sizes [***256, 384, 512***] bits that meet the following: FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-4, "Secure Hash Standard."

5.1.2.1.8 Random Bit Generation (FCS_RBG_EXT.1)

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 software-based noise source, 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.

5.1.2.1.9 Random Bit Generation (Android) (FCS_RBG_EXT.1/ANDROID)

FCS_RBG_EXT.1.1 /ANDROID The TSF shall [***invoke platform-provided functionality***] to perform all deterministic random bit generation services in accordance with NIST Special

	Publication 800-90A using [***Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES)***].
FCS_RBG_EXT.1.2 /ANDROID	The deterministic RBG shall be seeded by an entropy source that accumulates entropy from [***a hardware-based noise source]*** 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.

5.1.2.1.10 Cryptographic Key Storage (FCS_STG_EXT.1)

FCS_STG_EXT.1.1	The TSF shall utilize [***platform-provided key storage***] for all persistent secrets and private keys.

5.1.2.2 Cryptographic Support for MDM Agent PP-Module

5.1.2.2.1 Cryptographic Key Storage (FCS_STG_EXT.1(2))

FCS_STG_EXT.1.1(2)	**Refinement:** The MDM Agent shall use the [*platform-provided key storage*] for all persistent secret and private keys.

5.1.3 Identification and Authentication (FIA)

5.1.3.1 Identification and Authentication for MDM PP

5.1.3.1.1 Enrollment of Mobile Device into Management (FIA_ENR_EXT.1)

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 [***IMEI, [serial number]***] and [***a number of devices, [OS type, OS version]***].

5.1.3.1.2 Timing of Authentication (FIA_UAU.1)

FIA_UAU.1.1	**Refinement:** The TSF shall ***[invoke platform-provided functionality***] to allow [**request password reset**] on behalf of the user to be performed before the user is authenticated with the Server.
FIA_UAU.1.2	**Refinement:** The TSF shall [***invoke platform-provided functionality***] that requires each user to be successfully authenticated with the Server before allowing any other TSF-mediated actions on behalf of that user.

5.1.3.1.3 X.509 Certification Validation (FIA_X509_EXT.1(1))

FIA_X509_EXT.1.1(1)	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. • 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, a CRL as specified in RFC 5759 Section 5, an OCSP TLS Status Request Extension (i.e., OCSP stapling) as specified in RFC 6066, OCSP TLS Multi-Certificate Status Request Extension (i.e., OCSP Multi-Stapling) as specified in RFC 6961*]**.
- 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.

Note: This SFR has been modified by TD0641.

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.

5.1.3.1.4 X.509 Certificate Authentication (FIA_X509_EXT.2)

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 [***
    - ***code signing for system software updates,***
    - ***code signing for integrity verification,***

].

]

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***].

5.1.3.1.5 Client Authorization (FIA_CLI_EXT.1)

FIA_CLI_EXT.1.1 The TSF shall require a unique [***certificate***] for each client device.

Application Note: This SFR has been modified by TD0754.

5.1.3.2 Identification and Authentication for MDM Agent PP-Module

5.1.3.2.1 Agent Enrollment of Mobile Device into Management (FIA_ENR_EXT.2)

FIA_ENR_EXT.2.1 The MDM Agent shall record the reference identifier of the MDM Server during the enrollment process.

5.1.4 Security Management (FMT)

5.1.4.1 Security Management for MDM PP

5.1.4.1.1 Management of Functions Behavior (FMT_MOF.1(1))

FMT_MOF.1.1(1) **Refinement:** 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].

5.1.4.1.2 Management of Functions Behavior (Enrollment) (FMT_MOF.1(2))

FMT_MOF.1.1(2) **Refinement:** The **MDM Server** shall restrict the ability to *[initiate the enrollment process*] to [*authorized administrators and MD users*].

5.1.4.1.3 Management of Functions in (MAS Server Downloads) (FMT_MOF.1(3))

This selection-based component depends upon selection in FMT_MOF.1.1(1).

FMT_MOF.1.1(3) **Refinement:** 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.

5.1.4.1.4 Trusted Policy Update (FMT_POL_EXT.1)

FMT_POL_EXT.1.1 The TSF shall provide digitally signed policies and policy updates to the MDM Agent.

FMT_POL_EXT.1.2 The TSF shall sign policies and policy updates using a private key associated with [***an X509 certificate***] trusted by the agent for policy verification.

FMT_POL_EXT.1.3 For each unique policy managed by the TSF, the TSF shall validate that the policy is appropriate for an agent using [***client authentication via an X509 certificate representing the agent***].

This SFR has been modified by TD0754.

Application Note:

5.1.4.1.5 Specification of Management Functions (Server Configuration of Agent) (FMT_SMF.1(1))

FMT_SMF.1.1(1) **Refinement:** 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:
[

15. remove imported X.509v3 certificates and [

- ***no other X.509v3 certificates***

] in the Trust Anchor Database (MDF Function 12),
16. alert the user,

] and the following MD configuration policies:

25. password policy:
    minimum password length
    minimum password complexity
    maximum password lifetime (MDF Function 1)
26. session locking policy:
    screen-lock enabled/disabled
    screen lock timeout
    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 additional WLAN management functions**] (WLAN Client Function 1)
29. application installation policy by [

- ***specifying a set of allowed applications and versions (an application whitelist),***

], (MDF Function 8)

30. enable/disable policy for [**access to camera (Android 13 on Samsung Galaxy device only)**] across device, [
    - ***no other method***

], (MDF Function 5)

and the following MD configuration policies:

[

31. enable/disable policy for the VPN protection across MD, [
- ***no other method***

] (MDF Function 3),
36. enable policy for data-at-rest protection, (MDF Function 20),
54. enable/disable policy for the Always-On VPN protection across device (MDF Function 45),

5.1.4.1.6 Specification of Management Functions (Server Configuration of Server) (FMT_SMF.1(2))

FMT_SMF.1.1(2)	**Refinement:** The TSF shall be capable of performing the following management functions: a. b. configure the [- ***a number of devices***] and [*[**OS Type**]]* allowed for enrollment c. [***2. configure the TOE unlock banner*** ***8. [configure device enrollment type]****]*
Application Note:	This SFR has been modified by TD0887. Line 'a.' is intentionally blank per the TD resolution and does not represent a requirement, selection, or assignment.

5.1.4.1.7 Specification of Management Functions (MAS Server) (FMT_SMF.1(3))

This selection-based component depends upon selection in FMT_MOF.1.1(1)

FMT_SMF.1.1(3)	**Refinement:** The MAS Server shall be capable of performing the following management functions: a. Configure application access groups b. Download applications c. ***[no other functions]***

5.1.4.1.8 Security Management Roles (FMT_SMR.1(1))

FMT_SMR.1.1(1)	**Refinement:** 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.

5.1.4.1.9 Security Management Roles (MAS Server) (FMT_SMR.1(2))

This selection-based component depends upon selection in FMT_MOF.1.1(1)

FMT_SMR.1.1(2)	**Refinement:** The TSF shall additionally maintain the roles enrolled mobile devices, application access groups, and [**no additional identified roles**].
FMT_SMR.1.2(2)	**Refinement:** The MAS Server shall be able to associate users with roles.

5.1.4.2 Security Management for MDM Agent PP-Module

5.1.4.2.1 Agent Trusted Policy Update (FMT_POL_EXT.2)

FMT_POL_EXT.2.1	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	The MDM Agent shall not install policies if the signature check fails.

Application Note: This SFR has been modified by TD0755.

5.1.4.2.2 Specification of Management Functions (FMT_SMF_EXT.4)

FMT_SMF_EXT.4.1	The MDM Agent shall be capable of interacting with the platform to perform the following functions: • [***import the server public key***], • [***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 • [***no other functions***].

Application Note: This SFR has been modified by TD0755.

5.1.4.2.3 User Unenrollment Prevention (FMT_UNR_EXT.1)

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: [***apply remediation actions***].

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Protection of the TSF for MDM PP

5.1.5.1.1 Use of Supported Services and API's (FPT_API_EXT.1)

FPT_API_EXT.1.1	The TSF shall use only documented platform API's.

5.1.5.1.2 Internal TOE TSF Data Transfer (MDM Agent) (FPT_ITT.1(2))

This selection-based component depends upon selection in FTP_ITC_EXT.1.1

FPT_ITT.1.1(2) **Refinement:** 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.

5.1.5.1.3 Use of Third Party Libraries (FPT_LIB_EXT.1)

FPT_LIB_EXT.1.1 The MDM software shall be packaged with only [**MDM Agent third-party libraries listed in Annex C, MDM Service third-party libraries listed in non-public Library Document**]***.***

Application Note: This SFR is impacted by TD0895.

5.1.5.1.4 Functionality Testing (FPT_TST_EXT.1)

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.

5.1.5.1.5 Trusted Update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1 The TSF shall provide Authorized Administrators the ability to query the current version of the ~~MDM Server~~ software.

FPT_TUD_EXT.1.2 The TSF shall [***invoke platform-provided 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.

Application Note: This SFR has been modified by TD0438.

5.1.6 Trusted Path / Channels (FTP)

5.1.6.1 Trusted Path / Channels for MDM PP

5.1.6.1.1 Trusted Channel (FTP_ITC_EXT.1)

FTP_ITC_EXT.1.1 The TSF shall provide a communication channel between itself and [

- ***an MDM Agent that is internal to the TOE,***
- ***an MDM Agent that is external to the TOE,***

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

5.1.6.1.2 Inter-TSF Trusted Channel (Authorized IT Entities) (FTP_ITC.1(1))

FTP_ITC.1.1(1) **Refinement:** The TSF shall [

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

],

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

FTP_ITC.1.2(1) **Refinement:** 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) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to initiate communication via the trusted channel for [**audit server, authentication server**].

5.1.6.1.3 Inter-TSF Trusted Channel (MDM Agent) (FTP_ITC.1(2))

This selection-based component depends upon selection in FTP_ITC_EXT.1.1

FTP_ITC.1.1(2) **Refinement:** The TSF shall **[**

invoke platform-provided functionality to use [
HTTPS
],

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

FTP_ITC.1.2(2) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to permit the TSF and MDM Agent to initiate communication via the trusted channel.

FTP_ITC.1.3(2) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to initiate communication via the trusted channel for all communication between the TSF and the MDM Agent

5.1.6.1.4 Trusted Path (For Remote Administration) (FTP_TRP.1(1))

FTP_TRP.1.1(1) **Refinement:** 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) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3(1) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to require the use of the trusted path for [all remote administration actions].

5.1.6.1.5 Trusted Path (For Enrollment) (FTP_TRP.1(2))

FTP_TRP.1.1(2) **Refinement:** The TSF shall [

invoke platform-provided functionality to use [

- ***TLS***

],

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

FTP_TRP.1.3(2) **Refinement:** The TSF shall [***invoke platform-provided functionality***] to require the use of the trusted path for [all MD user actions].

5.2 TOE Security Assurance Requirements

5.2.1 CC Part 3 Assurance Requirements

The following table is the collection of CC Part 3 assurance requirements from the Protection Profile for Mobile Device Management and PP-Module for MDM Agents.

Table 16 TOE Security Assurance Requirements

Assurance Class	Assurance Components
Security Target (ASE)	ST Introduction (ASE_INT.1)
	Conformance Claims (ASE_CCL.1)
	Security Objectives for the Operational Environment (ASE_OBJ.1)
	Extended Components Definition (ASE_ECD.1)
	Stated Security Requirements (ASE_REQ.1)
	TOE Summary Specification (ASE_TSS.1)
Development (ADV)	Basic Functional Specification (ADV_FSP.1)
Guidance Documents (AGD)	Operational User Guidance (AGD_OPE.1)
	Preparative Procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labeling of the TOE (ALC_CMC.1)
	TOE CM Coverage (ALC_CMS.1)
Tests (ATE)	Independent Testing – Sample (ATE_IND.1)
Vulnerability Assessment (AVA)	Vulnerability Survey (AVA_VAN.1)

5.2.2 Assurance Activities

Full definitions and descriptions of all security assurance requirements and activities are listed in the MDM PP and MDM Agent PP-Module. Refer to these documents for further reference.

6 TOE Summary Specification (TSS)

This chapter describes the Microsoft Intune security functions that satisfy the security functional requirements of the Protection Profile for Mobile Device Management and PP-Module for MDM Agents. The TOE also includes additional relevant security functions which are also described in the following sections, as well as a mapping to the security functional requirements satisfied by the TOE.

This section presents the TOE Security Functions (TSFs) and a mapping of security functions to Security Functional Requirements (SFRs). The TOE performs the following security functions:

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TSF
- Trusted Path and Channels

6.1 Security Audit

6.1.1 Server Alerts (FAU_ALT_EXT.1)

Microsoft Intune provides alerting mechanisms to an administrator via the Microsoft Intune Admin Center dashboard. A notifications tab within the dashboard aggregates any alerts that are active or in progress and may require the attention of an administrator. Alerts shown on the dashboard include changes in the enrollment status of a device, and failures to apply policies to a device. There is no limit on the number of notifications that can be queued in the notifications tab.

More information on alerts and dashboards in Microsoft Intune Admin Center can be found here: https://learn.microsoft.com/en-us/mem/intune/fundamentals/tutorial-walkthrough-endpoint-manager.

6.1.2 Agent Alerts (FAU_ALT_EXT.2)

Intune managed devices are configured to check in with the Intune server at a standard refresh frequency. The check in cycle occurs every eight hours approximately but can shorter or longer depending on the device and its connectivity. During this period, any alerts or notifications are queued on the device until the next check in at which point they are delivered to Intune.

More information on device check-in intervals can be found here https://docs.microsoft.com/en-us/mem/intune/configuration/device-profile-troubleshoot#how-long-does-it-take-for-devices-to-get-a-policy-profile-or-app-after-they-are-assigned

An administrator may also force a sync manually via the Microsoft Intune Admin Center console. This action will notify the managed device to check in immediately with Intune and send any queued alerts or notifications. The maximum amount of storage that queued messages may consume is commensurate with the available space on the local onboard storage or installed SD card. Alternatively, if the user explicitly denies permissions on the device for the Company Portal app to write data logs to the devices' storage, logs may still be sent via email by the user.

All device check-ins are performed over an authenticated TLS channel between the managed device and Intune service.

Each time a device checks in with the Intune server, it queries for new or updated policies that are applicable to it. If there is a new or updated policy assigned to the device, it will immediately be received by the device in addition to any pending actions on the device. A device may not complete a scheduled check in, or otherwise fail to check in with Intune if there is a loss of network connectivity or the device is powered off. If a device fails to check in at the regular check-in interval, it will reattempt a check in at the next scheduled check in.

More information on obtaining policy updates for a device can be found here https://docs.microsoft.com/en-us/mem/intune/remote-actions/device-sync

If a managed device experiences a failure to install an application assigned to it, Intune is notified and an administrator can access troubleshooting details within the Microsoft Intune Admin Center console as described here https://docs.microsoft.com/en-us/troubleshoot/mem/intune/troubleshoot-app-install#app-troubleshooting-details

More information on managing Company Portal App permissions can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/end-user-company-portal-messages#allow-company-portal-to-access-photos-media-and-files-on-your-device

6.1.3 Audit Data Generation (FAU_GEN.1(1)) & Audit Generation (MAS Server) (FAU_GEN.1(2))

Microsoft Intune automatically generates the audit records required by the protection profile by default. A complete list of the audit records generated by the TOE are listed in **Table 14** and **Table 15** and includes the following events:

a) Administrative actions and changes (MDM System)

b) Commands issued to the MDM agent or device (MDM System)

c) All auditable events listed in **Table 14** and **Table 15** (MDM System, MDM Platform)

Each audit record includes the following fields:

- **a) Date.** Includes the day, month, year, and timestamp of the activity
- **b) Initiated By (Actor).** Includes information on who caused the activity.
- **c) Application Name.** Name of the application involved in the activity.
- **d) Activity.** The friendly name of the activity being performed.
- **e) Target.** The subject of the event, including properties that were changed.
- **f) Category.** Identifies the category type of the event.

The status of each event is also logged to indicate a success or failure.

The MDM system (Intune) generates audit records when a failure to download a new or updated application to a managed device occurs. All administrative actions and changes, in addition to commands issued to a managed device are logged and shown in the management dashboards of Intune to monitor execution status.

The TOE also leverages audit logs generated by the underlying MDM platform (Windows) which pertain to the system, service events, cryptographic, and security related events.

More information on audit logs in Microsoft Intune can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/monitor-audit-logs

6.1.4 Audit Data Generation (FAU_GEN.1(2))

The Microsoft Intune Admin Center dashboard contains an Audit Log for administrators to view. Granular information related to an event is collected and displayed in the activity details of the audit log which contains the following sections:

a) **Activity.** Contains the date, time, event name and category, and correlation ID for tracking.

b) **Activity Status.** Contains the activity and operation type, and status or outcome of the event.

c) **Initiated By (Actor).** Describes the execution method of the event and the identity of who caused the event.

d) **Scope Tag(s).** Provides classification of audit events and can be used to provide administrators with certain permissions to view objects.

e) **Target(s).** Contains the type and objectID of the target.

f) **Modified Properties.** Indicates the properties of the device or policy that have been changed.

Audit logs in the Microsoft Intune Admin Center can be filtered by Category, Activity, and Date Range. More information can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/monitor-audit-logs

In addition to the audit log found in the Intune Tenant Administration section of the Microsoft Intune Admin Center, Intune also generates and stores a security event log which contains security relevant audit records including administrator authentication events, and changes to permission levels for administrators. Data fields contained in these log events include Date, Time, User, Event ID, Source, Outcome, and Category.

The Microsoft Graph API may also be used to retrieve audit events. For details on using the Graph API, see https://docs.microsoft.com/en-us/graph/api/resources/intune-auditing-auditevent?view=graph-rest-1.0

6.1.5 Security Audit Event Selection (FAU_SEL.1(2))

Audit events are not modifiable and are always recorded by default. Administrators cannot disable logging once it has been enabled. An administrator may filter audit log events by category, activity, and date range within the Intune Admin Center.

More information on reviewing audit logs in Microsoft Intune Admin Center can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/monitor-audit-logs

6.1.6 Network Reachability Review (FAU_NET_EXT.1)

Intune provides the ability for administrators to determine the connectivity status of enrolled devices in the Intune Admin Center. Managed devices are configured to check in with Intune periodically. Intune

will notify the device to perform the check in approximately every eight hours. If a device does not check in to get policy or profile updates after the first notification from Intune, Intune will make another three attempts. If a device is turned off, it may not receive these notifications from Intune to Check in, however once the device comes back online, a check in will be conducted at the regular scheduled interval.

Devices that are recently enrolled will temporarily check-in more frequently for compliance and configuration checks. Android device check-ins will occur every three minutes for fifteen minutes, then every fifteen minutes for two hours, and then every eight hours. iOS devices will check in every fifteen minutes for one hour, and then every eight hours.

An administrator may also force a check in via the 'Sync' feature in the Microsoft Intune Admin Center. Or a device user may trigger a check for policy or profile updates by tapping the sync button in the Company Portal settings page.

6.1.7 External Trail Storage (FAU_STG_EXT.1) & Audit Event Storage (FAU_STG_EXT.2)

Microsoft Intune relies on the underlying Windows platform to store audit data that is generated and consumed by Intune. The platform ensures the confidentiality and integrity of stored audit data by providing an Intune Tenant specific database. The availability of audit data review is provided locally through the Microsoft Intune Admin Center console by authorized Intune tenant administrators, or via the platform by authorized Microsoft Service Operators with specific roles and appropriate permissions. Audit logs cannot be modified or deleted. Additionally, Intune can securely send audit events to any endpoint that is capable of using the Microsoft Graph API via HTTPS.

For details on using the Graph API, see https://docs.microsoft.com/en-us/graph/api/resources/intune-auditing-auditevent?view=graph-rest-1.0

6.2 Cryptographic Support

6.2.1 Cryptographic Key Generation (FCS_CKM.1)

The TOE invokes platform provided functionality to leverage RSA and ECC schemes in accordance with the FIPS PUB 186-4 standard to generate asymmetric cryptographic keys as part of key establishment. During the device enrollment process the TOE will generate an RSA or ECDSA keypair to generate a certificate for the device, which is then used for future TLS sessions.

6.2.2 Cryptographic Key Establishment (FCS_CKM.2)

The TOE invokes platform provided functionality support the generation of asymmetric keys during the key establishment process for TLS and HTTPS sessions. Intune supports both RSA (2048-bits or greater key size) and ECC (curves P-256, P-384, and P-521) key establishment schemes. The underlying Windows environment uses the Cryptography API: Next Generation (CNG) API exclusively for its own encryption needs. CNG natively implements Suite B algorithms including AES (128 and 256), SHA-2 (256, 384, and 512), ECDH, and ECDSA over the NIST-standard prime curves P-256, P-384, and P-521. Enrolled devices use TLS for trusted channel communications with the Intune service during scheduled check ins.

6.2.3 Cryptographic Key Destruction (FCS_CKM_EXT.4)

The TOE relies on platform provided storage to store and manage RSA certificate private keys on the clients. Within the Intune service, certificates are stored for long-term use in the Azure Key Vault. All private keys and secrets in Key Vault are stored in an encrypted state. The TOE also invokes platform provided functionality in the underlying Windows operating system to leverage the CNG key isolation service that hosts secret and private keys in a protected process for tamper mitigation and unauthorized access to sensitive key materials.

The underlying Windows platform will destroy keys and key material by overwriting all storage areas for plaintext keying material and critical security parameters, including any storage such as memory buffers for the key. When keys are deleted, the Azure Key Vault service automatically places them in a "Soft Deleted" state where they are retained for an interval of up to 90 days. At the same time, Key Vault will schedule the deletion of the key data for execution after the retention interval. The soft-delete retention period cannot be disabled by users or administrators. Once the retention interval has elapsed, the Key Vault service automatically performs a purge of the soft-deleted keys which are then permanently and irrecoverably deleted. All key destruction functions were tested as part of the Windows Server 2019 Common Criteria evaluation. As such, all descriptions of these functions have been derived from the evidence submitted and approved in that evaluation and remain unchanged.

When keys, secrets, and CSP's are no longer needed by the TOE or TOE platform for a network session, due to either a normal end to the session or abnormal termination, or after protecting sensitive data using DPAPI, they are deleted by Key Vault as described above and in section 5.1.2.1.3. All keys and CSP's used by the Android agent are deleted by Android KeyStore when they are no longer needed due to the removal of the MDM profile when the device has been removed from MDM management.

For volatile memory, the overwrite is performed with a single direct overwrite consisting of zeros using the *RtlSecureZeroMemory* function immediately after that data is no longer needed or following the termination of a normal session. For non-volatile memory, the overwrite is also performed with a single direct overwrite consisting of zeros. Android agents rely on platform provided storage (Android KeyStore) for storage and management of private keys, device certificates, and the destruction of keys and CSP's.

The following table describes the keys and secrets used by the TOE platform for networking and data protection.

Table 17 Cryptographic Keys and CSPs

Key / CSP	Description
Symmetric encryption/decryption keys	Keys used for AES (FIPS 197) encryption/decryption for TLS.
HMAC keys	Keys used for HMAC-SHA256, HMAC-SHA384, and HMAC-SHA512 (FIPS 198-1).
Asymmetric ECDSA Public Keys	Keys used for the verification of ECDSA digital signatures using the P-256, P-384, and P-521 curves (FIPS 186-4) for TLS.
Asymmetric ECDSA Private Keys	Keys used for the calculation of ECDSA digital signatures using the P-256, P-384, and P-521 curves (FIPS 186-4) for TLS.

Key / CSP	Description
Asymmetric RSA Public Keys	Keys used for the verification of RSA digital signatures (FIPS 186-4) for TLS and signed product updates.
Asymmetric RSA Private Keys	Keys used for the calculation of RSA digital signatures (FIPS 186-4) for TLS as well as TPM-based health attestations. The key size can be 2048 or 3072 bits.
Asymmetric DSA Private Keys	Keys used for the calculation of DSA digital signatures (FIPS 186-4) for TLS. The key size can be 2048 or 3072 bits.
Asymmetric DSA Public Keys	Keys used for the verification of DSA digital signatures (FIPS 186-4) for TLS. The key size can be 2048 or 3072 bits.
DH Private and Public values	Private and public values using ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144 Diffie-Hellman key establishment for TLS.
ECDH Private and Public values	Private and public values using the P-256, P-384, and P-521 curves in EC Diffie-Hellman key establishment for TLS.
DPAPI master secret	512-bit random value used by DPAPI
DPAPI master AES key	256-bit encryption key that protects the DPAPI master secret
DPAPI AES Key	256-bit encryption key used by DPAPI
DRBG seed	seed for the main DRBG, zeroized during reseeding

6.2.4 Cryptographic Operations - Confidentiality Algorithms (FCS_COP.1(1)), Hashing Algorithms (FCS_COP.1(2)), Signature Algorithms (FCS_COP.1(3)), and (Keyed-Hash Message Authentication) (FCS_COP.1(4))

The TOE performs all encryption, decryption, signing, and hashing operations by leveraging the Windows platform provided cryptographic libraries and use independent modules known as Cryptographic Service Providers (CSP's). For MDM agents on Android devices, platform provided libraries are also leveraged to provide cryptographic functionality.

The TOE platform natively implements the Suite B algorithms, including algorithms for AES (128 and 256 key sizes), the SHA-2 family (SHA-256, SHA-384 and SHA-512) of hashing algorithms, elliptic curve Diffie Hellman (ECDH), and elliptical curve DSA (ECDSA) over the NIST-standard prime curves P-256, P-384, and P-521. Transport Layer Security (TLS), makes use of elliptic curve Diffie-Hellman (ECDH) included in Suite B as well as hashing functions. The tables below describe the platform provided cryptographic algorithm standards leveraged by the TOE and include references to relevant NIAP validation identifiers and certificate numbers that pertain to those platforms.

Table 18 Cryptographic Algorithm Standards and Evaluation Methods for Windows

Cryptographic Operation	Standard	Requirement	Evaluation Method
Encryption/Decryption	FIPS 197 AES NIST SP 800-38A CBC mode	FCS_COP.1(*)	CCN-CC-12/2020[7]

[7] CCRA certificate identifier for *Microsoft Windows 10 version 1909 and Microsoft Windows Server version 1909*, Certification Date: 2020-06-05, https://www.commoncriteriaportal.org/nfs/ccpfiles/files/epfiles/2019-47-ST-lite.pdf

	NIST SP 800-38C CCM mode		
	NIST SP 800-38E XTS mode		
	NIST SP 800-38F KW mode		
	NIST SP 800-38D GCM mode		
Digital signature (key generation)	FIPS 186-4 RSA	FCS_CKM.1	
Digital signature (generation)	FIPS 186-4 RSA	FCS_COP.1(*)	
Digital signature (verification)	FIPS 186-4 RSA	FCS_COP.1(*)	
Digital signature (generation and verification)	FIPS 186-4 DSA	Added as a prerequisite of NIST CAVP KAS # C1897, # C2047	
Digital signature (key generation)	FIPS 186-4 ECDSA	FCS_CKM.1	
Digital signature (signature generation and verification)	FIPS 186-4 ECDSA	FCS_CKM.1	
Hashing	FIPS 180-4 SHA-256, SHA-384, SHA-512	FCS_COP.1 (*)	
Keyed-Hash Message Authentication Code	FIPS 198-2 HMAC	FCS_COP.1(*)	
Random number generation	NIST SP 800-90 CTR_DRBG	FCS_RBG_EXT.1	
Key agreement	NIST SP 800-56A ECDH	FCS_CKM.2	
Key establishment	NIST SP 800-56B RSA	FCS_CKM.2	

Table 19 Cryptographic Algorithm Standards and Evaluation Methods for Android

Cryptographic Operation	Standard	Requirement	Samsung Galaxy / Android 13	Google Pixel / Android 11
Key Generation	FIPS 186-4 ECC FIPS 186-4 RSA	FCS_CKM.1	VID11342[8]	VID11124[9]
Key Establishment	NIST SP 800-56A ECDH NIST SP 800-56B RSA	FCS_CKM.2		
Encryption/Decryption	NIST SP 800-38A NIST SP 800-38D	FCS_COP.1(1)		

[8] NIAP validation identifier for *Samsung Electronics Co., Ltd. Samsung Galaxy Devices on Android 13 – Spring,* Certification Date: 04/26/2023
[9] NIAP validation identifier for *Google Pixel Phones on Android 11.0,* Certification Date: 02/08/2021

Cryptographic Operation	Standard	Requirement	Samsung Galaxy / Android 13	Google Pixel / Android 11
	NIST SP 800-38F (AES 128, 256-bit)			
Hashing	FIPS Pub 180-4 (SHA 256, 384, 512)	FCS_COP.1(2)		
Signature Generation	FIPS Pub 186-4 ECDSA (P-256, P-384, P-521) FIPS Pub 186-4 RSA	FCS_COP.1(3)		
Keyed-Hash	FIPS Pub 198-1 FIPS Pub 180-4 (HMAC-SHA 256, 384, 512) (128, 256-bits)	FCS_COP.1(4)		
DRBG (CTR)	NIST SP 800-90A	FCS_RBG_EXT.1		

6.2.5 Extended: Random Bit Generation (FCS_RBG_EXT.1), Random Bit Generation (Android) (FCS_RBG_EXT.1/ANDROID)

The TOE MDM server leverages the underlying Windows operating environment of the TOE platform to implement Deterministic Random Bit Generation (DRBG) in accordance with NIST Special Publication 800-90. Intune leverages the output of a cascade of two SP800-90 AES-256 counter mode based DRBG's (in Kernel mode), and four cascaded SP800-90 AES-256 DRBG's (in User mode) to generate random bits. The underlying Windows entropy pool seeds the RBG which can provide 128 or 256 bits to programmatic callers.

The TOE platform utilizes different entropy sources (deterministic and nondeterministic) which produce entropy data that is used for random number generation to support Intune. In particular, this entropy data together with other data (such as the nonce) seed the DRBG algorithm. The entropy pool is populated using the following values:

- An initial entropy value from a seed file provided to the Windows OS Loader at boot time (512 bits of entropy).
- A calculated value based on the high-resolution CPU cycle counter which fires after every 1024 interrupts (a continuous source providing 16384 bits of entropy).
- Random values gathered periodically from a Trusted Platform Module (TPM), (320 bits of entropy on boot, 384 bits thereafter on demand based on an OS timer).
- Random values gathered periodically by calling the RDRAND CPU instruction, (256 bits of entropy).

Each entropy source is independent of the other sources and does not depend on time. Entropy data is gathered from all sources in raw formats prior to undergoing health tests before it is used as an input for

the DRBG. The main source of entropy is the system CPU cycle counter which continuously tracks hardware interrupts within the TOE platform.

The TOE platform defends against tampering of the random number generation (RNG) / pseudorandom number generation (PRNG) sources by encapsulating its use in Kernel Security Device Driver. The interface for the underlying Windows random number generator is BCryptGenRandom

The CNG provider for random number generation is the AES_CTR_DRBG, if Windows requires the use of a salt it uses the Windows RBG.

For Android devices, the MDM Agent leverages the underlying platform of the Android operating environment to implement Deterministic Random Bit Generation (DRBG) using Hash, HMAC, and CTR based DRBGs in accordance with NIST SP800-90A. The DRBG is seeded by an entropy source that accumulates entropy from a hardware-based noise source with a minimum of 256 bits of entropy.

6.2.6 Cryptographic Key Storage (FCS_STG_EXT.1 & FCS_STG_EXT.1(2))

CNG includes a user-mode key isolation service designed specifically to host secret and private keys in a protected process to mitigate tampering or access to sensitive key materials for user-mode processes. The TOE does not store keys in plaintext at any time. Ephemeral keys and secrets are deleted once they are no longer needed for a network session, or at normal end of session termination.
Windows overwrites each intermediate storage area for plaintext key/critical cryptographic security parameter.

Device MDM agents store their private keys associated with certificates in platform provided key stores.

In Android devices, the MDM agent calls the Android KeyStore API on the device to facilitate the secure storage of all cryptographic keys and persistent secrets in the Android KeyStore. Only Android platform API's are used.

Table 20 Android MDM Agent Keys and Secrets

Key/Persistent Secret	Purpose	Storage
TLS Keys	Keys used to protect communications with the Intune server.	Stored on the device in persistent storage.
UID	Unique Device ID.	Stored on the device in persistent storage.
Device Certificate	The certificate issued to a device upon initial registration with the Intune server, and used to authenticate the device when performing check-ins.	Stored on the device in persistent storage.

In iOS devices, the MDM agent calls the iOS Keychain Services API on the device to facilitate the secure storage of all cryptographic keys and persistent secrets in the iOS Keychain on non-volatile storage. Only iOS platform API's are used.

6.3 Identification and Authentication

6.3.1 Enrollment of Mobile Device into Management (FIA_ENR_EXT.1) and Agent Enrollment of Mobile Device into Management (FIA_ENR_EXT.2)

Enrollment of a device in Intune can be completed by an administrator or the user in possession of the device. iOS devices are enrolled using the native iOS agent via the Microsoft Intune Company Portal App and Android devices are enrolled through Android Device Administrator via the Intune Company Portal App.

An administrator must first create an account consisting of a username and password in Intune via an existing Active Directory environment or within the Users page of the Intune Admin Center and ensure an Intune license has been assigned to the user. Once user information has been set up in Intune, the user can download the Microsoft Intune Company Portal App from the Apple App store or Google Play store and log in with the appropriate credentials to begin the device registration and enrollment process.

An Intune administrator can create and manage multiple enrollment restrictions to define what devices may enroll into Intune management. Enrollment restrictions can be specified by OS type, OS version, and a configurable number of devices per user. For Android and iOS devices, an IMEI (International Mobile Equipment Identity) number can be used as a unique identifier to create a whitelist that restricts enrollment to specific devices. Intune also supports the use of a serial number to restrict enrollment of iOS devices only.

During the successful enrollment of a device via the Company Portal app, an X509 certificate issued by Intune is installed on the device. All future communication between the device and Intune service leverages this certificate to conduct authenticated TLS sessions initiated by the device.

Additional information on enrolling devices in Intune can be found here https://docs.microsoft.com/en-us/mem/intune/enrollment/

When devices register with Intune during the enrollment process, the device agent records the unique URL of the TOE for future communications with Intune. This is facilitated through the use of the Company Portal app that users log into during enrollment.

When successfully enrolled, the device can then receive commands and policy updates from Intune during regular check in intervals which are configured by the TOE and initiated by the device.

By default, users cannot unenroll or otherwise remove their device from the Company Portal app until it has been removed from Intune management. Once an enrolled device has been removed from Intune and is no longer managed by the organization, the user will lose all access to organization data, apps, and configurations.

More information on removing an iOS device from Company Portal can be found here https://docs.microsoft.com/en-us/mem/intune/user-help/unenroll-your-device-from-intune-ios

6.3.2 Timing of Authentication (FIA_UAU.1)

The TOE requires authentication prior to accessing the Microsoft Intune Admin Center console, and any security configuration or functionality. Administrators cannot perform any actions on the TOE, including

modification of TSF functionality, until they have been authenticated successfully. Prior to authentication, the only actions an administrator can take are those associated with authenticating to the TOE which include logging in, creating a new generic Microsoft account, and selecting additional sign-in options such as use of a security key, or selection of a different organization to log in to.

Administrators navigate to the Intune Admin Center console over HTTPS. Intune then determines if a security token associated with a tenant is present for the administrator. If no token is present, Intune redirects to Azure Active Directory where the administrator provides a username and password. Upon successful authentication via Azure Active Directory, a token that correlates only to the tenant in which the administrator is a member of is granted to the administrator before redirection back to the Intune Admin Center console where access is then granted to the specific tenant. This method prevents any inter-tenant communication or data access.

Users connecting to Intune must provide a username and password for authentication using the Company Portal app prior to any enrollment of devices or access to company data. Prior to logging in, users cannot perform any actions via the Company Portal app other than logging in, or requesting a password reset by selecting this option at the login page.

6.3.3 X.509 Certificate Validation (FIA_X509_EXT.1(1))

The TOE leverages the platform to validate all certificates used within Intune, its components, and those issued to MDM agents. The underlying Windows OS stores all related Intune certificates, including trusted root certificates, in a certificate store. The certificate database provided by the Windows operating environment of the TOE platform facilitates the validation process and conducts lookups of the certificate to check its revocation status via OCSP per RFC 6960 or CRL per RFC 5759 Section 5, including the specific validation listed in section 5.1.3.1.3, and all applicable usage constraints such as Server Authentication for networking sessions, and Code Signing when installing TOE and TOE platform related updates. Revocation checking is performed during all updates to the underlying Windows platform, code signing, policy signing, and during communication with Windows platform services such as Azure Active Directory and Graph API. Revocation checking is also performed by the Android agent during device enrollment, and upon all subsequent device check-ins.

OCSP stapling and OCSP multi-stapling are also supported per RFC 6066 and RFC 6961 respectively and do not require any external components. Each Windows component that uses X.509 certificates within the MDM system is responsible for performing certificate validation and will select a certificate based on criteria such as entity name for the communication partner, any extended key usage constraints, and cryptographic algorithms associated with the certificate. The TOE platform will also use the same kinds of information along with a certification path and certificate trust lists as part of deciding whether to accept the certificate.

Should the validation fail or otherwise be unable to check the status of a certificate, the TSF will not accept the certificate and will not establish a trusted network channel. In the event that a certificate of a device has been revoked, a connection to Intune may still occur if there is an MDM '*wipe*' or '*retire'* command pending. In this case, the connection will be allowed only to securely wipe or retire the device from Intune.

Certificate properties that are verified include the entity name, extended key usage constraints, and cryptographic algorithms associated with the certificate. The certification path is also used to determine

whether to accept the certificate by checking that the Root certificate's subject key identifier (SKI) matches the intermediate certificate's authority key identifier (AKI) and the intermediate certificate's subject key identifier (SKI) matches the leaf certificate's authority key identifier (AKI). During the validation process, the TOE platform will also verify the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates. The basicConstraints extension contains the 'SubjectType' and 'PathLengthConstraint' parameters where the 'SubjectType' is set to 'EndEntity' for certificates issued by the TOE and set to 'CA' for intermediate and root certificates. The 'PathLengthConstraint' is set to '1' for root, '0' for intermediate, and none for leaf.

6.3.4 Certificate Authentication (FIA_X509_EXT.2)

The TOE invokes platform-provided functionality to leverage X.509 certificates in support of authentication for all HTTPS and TLS connections. Certificate authentication is performed by the TOE platform when a connection is made between the Intune service and a managed device during a device check-in. The TOE also relies on the Windows platform to perform certificate authentication for code signing when software updates are installed on the system, and code signing for integrity verification of mobile applications.

X.509 authentication of updates to the underlying Windows operating system, mobile applications, and integrity verification is mandatory functionality implemented by the TOE platform and cannot be bypassed. If the TOE platform fails to validate a certificate via the rules defined in FIA_X509_EXT.1(1) and described in section 6.3.3, or if a connection to validate the status of the certificate via OCSP or CRL cannot be established, the TOE will not accept the certificate.

6.3.5 Client Authorization (FIA_CLI_EXT.1)

Each successfully enrolled mobile device is issued a unique certificate by the TOE which authenticates the device and only allows access to the tenant in which the device was enrolled. This certificate is stored in the platform keystore and is used for authentication during scheduled check-ins over a TLS session.

6.4 Security Management

6.4.1 Management of Functions Behavior (FMT_MOF.1(1))

Microsoft Intune restricts the ability to manage TSF functionality to authorized administrators only. An authorized administrator that successfully authenticates to the TOE is presented with the ability to perform the actions specified by the SFRs in the claimed protection profiles, including the creation and application of policies that Intune enforces on enrolled devices. An administrator has no ability to modify TSF data or device policies prior to authentication. More information on the Microsoft Intune Role-based Access Control (RBAC) model can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/role-based-access-control

6.4.2 Management of Functions Behavior (Enrollment) (FMT_MOF.1(2))

Users that self enroll a device must also be authorized and successfully authenticate to the TOE prior to device registration and enrollment. Administrators may restrict the ability to enroll a device to specific users and devices via enrollment restriction policies. More information on device enrollment restrictions can be found here

https://docs.microsoft.com/en-us/mem/intune/enrollment/enrollment-restrictions-set

6.4.3 Management of Functions in (MAS Server Downloads) (FMT_MOF.1(3))

Microsoft Intune provides functionality for administrators to control specific applications that are made available for download by users via the Company Portal App or platform app stores if allowed by policy. Administrators can granularly control the availability of specific applications based on enrolled device groups and users via application assignment policies. Microsoft Intune will automatically update most app types as updates become available.

For more information on app types in Intune, see https://docs.microsoft.com/en-us/mem/intune/apps/apps-add#app-types-in-microsoft-intune

Additional information on updating apps can be found here https://docs.microsoft.com/en-us/mem/intune/apps/apps-add#installing-updating-or-removing-required-apps

6.4.4 Trusted Policy Update (FMT_POL_EXT.1)

Intune provides new and updated policy data to MDM agents via TLS. This approach ensures that all policy and configuration data transmitted between the MDM Server and MDM agent is protected by leveraging the X509 certificate provisioned on the device during enrollment. Additionally, policy and configuration data is signed by Intune via the Intune cert service which generates an RSA signature using a private key associated with an X509 certificate and verified by the agent via its own certificate. Intune administrators target the appropriate application of Intune policies to specific users, devices, and device groups within Intune. Intune will only send policies and policy updates to agents that are the subject of such targeting by an Intune administrator. All enrolled devices will perform a check-in with Intune every 8 hours (this interval is not administrator configurable). During check-in, devices will authenticate to Intune using their X509 certificates and validate the new policies and policy updates assigned by the Intune administrator. Devices that are not enrolled in Intune management and don't have an Intune provisioned certificate installed, are unable to perform a check-in and receive policy updates.

6.4.5 Agent Trusted Policy Update (FMT_POL_EXT.2)

MDM agents receive signed policy data through a protected and authenticated TLS connection with the MDM server. Each time a managed device performs a check-in with the Intune server, the payload of the device policy is signed by the Intune cert service. If verification of the RSA signature by the agent is successful, the policy is then applied to the device. If the signature is not present or otherwise cannot be verified, the policy is not applied to the device.

Note: on iOS devices, if the policy signature is not present or cannot be verified, the iOS MDM agent will check the connection of the payload origin to determine if the origin is trusted and its certificate can be validated. If the certificate of the origin is successfully validated, the policy will be applied to the device, otherwise the policy is not applied.

6.4.6 Specification of Management Functions (Server Configuration of Agent) (FMT_SMF.1(1))

Microsoft Intune provides administrators the ability to send commands to enrolled devices and configure all features and policies required by the claimed protection profiles. These commands are sent

to the agents on enrolled devices which then execute the commands to enforce the policies assigned by Intune. The table below identifies the required (R) and additionally selected (S) functions that Intune can implement on Android and iOS devices. Device-level support for each function is identified by "Y" for yes and "N" for no. The tested devices include the Samsung Galaxy S21 Ultra 5G (Knox), Google Pixel 4a 5G, and Apple iPhone 12.

Configuration of these features and policies can be achieved using Intune Compliance policies and Configuration profiles. More information on Intune Compliance policies can be found here https://docs.microsoft.com/en-us/mem/intune/protect/device-compliance-get-started

and more information on Intune Configuration profiles can be found here https://docs.microsoft.com/en-us/mem/intune/configuration/

Note: **Table 21** below includes iOS only as a demonstrative illustration of the capabilities of Intune with respect to managing Apple devices running iOS software. iOS-based devices were not considered to be in scope or were otherwise tested as part of this evaluation.

Table 21 Management Functions Provided by the TOE

#	Management Function	Required / Selected	Android 13 (Samsung Galaxy)	Android 11 (Google Pixel)	iOS 15 (Apple iPhone)
1.	transition to the locked state (MDF Function 6)	R	Y	Y	Y
2.	full wipe of protected data (MDF Function 7)	R	Y	Y	Y
3.	unenroll from management	R	Y	Y	Y
4.	install policies	R	Y	Y	Y
5.	query connectivity status	R	Y	Y	Y
6.	query the current version of the MD firmware/ software	R	Y	Y	Y
7.	query the current version of the hardware model of the device	R	Y	Y	Y
8.	query the current version of installed mobile applications	R	Y	Y	Y
9.	import X.509v3 certificates into the Trust Anchor Database (MDF Function 11)	R	Y	Y	Y
10.	install applications (MDF Function 16)	R	Y	Y	Y
11.	update system software (MDF Function 15)	R	Y	Y	Y
12.	remove applications (MDF Function 14)	R	Y	Y	Y
15.	remove imported X.509v3 certificates and [• ***no other X.509v3 certificates***] in the Trust Anchor Database (MDF Function 12)	S	Y	N	Y
16.	alert the user	S	Y	Y	Y
25.	password policy:	R	Y	Y	Y

#	Management Function	Required / Selected	Android 13 (Samsung Galaxy)	Android 11 (Google Pixel)	iOS 15 (Apple iPhone)
	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)	R	Y	Y	Y
27.	wireless networks (SSIDs) to which the MD may connect (MDF Function 2)	R	Y	Y	Y
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 additional WLAN management functions**] (WLAN Client Function 1)	R	Y	Y	Y
29.	application installation policy by [○ ***specifying a set of allowed applications and versions (an application whitelist),***], (MDF Function 8)	R	Y	N[10]	N
30.	enable/disable policy for [**access to camera (Android 13 on Samsung Galaxy device only)**] across device, [○ ***no other method***], (MDF Function 5)	R	Y	N[11]	N

[10] This function is not available in Android Device Administrator management. Samsung Galaxy Knox devices running Android 13 preserve this functionality.
[11] This function has been deprecated by Google in Android Device Administrator management starting in Android 10. Samsung Galaxy Knox devices running Android 13 preserve this functionality.

#	Management Function	Required / Selected	Android 13 (Samsung Galaxy)	Android 11 (Google Pixel)	iOS 15 (Apple iPhone)
31.	enable/disable policy for the VPN protection across MD, [○ ***no other method***] (MDF Function 3)	S	Y	Y	Y
36.	enable policy for data-at-rest protection, (MDF Function 20)	S	Y	Y	Y
54.	enable/disable policy for the Always-On VPN protection across device (MDF Function 45),	S	N	N	Y

6.4.7 Specification of Management Functions (Server Configuration of Server) (FMT_SMF.1(2))

Intune extends its capabilities to manage registered devices, and also provides the ability to configure and manage its own security functions. Available functions include restricting the number of devices that a user can enroll, and restricting the OS type of devices that can enroll. Additionally, Intune supports the configuration of an unlock banner at the time of administrator login, and the ability to modify the enrollment type of a device to 'Personal' or 'Corporate' thereby limiting or extending access to additional information collected by Intune.

More information on configuring the TOE unlock banner can be found here:
https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/customize-branding

More information on what information can be accessed by Intune can be found here:
https://docs.microsoft.com/en-us/mem/intune/user-help/what-info-can-your-company-see-when-you-enroll-your-device-in-intune

The Intune operational environment assigns a single, unique Tenant ID and domain to support the logical isolation of customer data, and enforces the correlation of an administrator to a specific tenant.

6.4.8 Specification of Management Functions (MAS Server) (FMT_SMF.1(3))

The TOE provides MAS server functionality to enrolled devices. Intune supports the configuration of application groups that are accessible to assigned users and groups, as well as the ability to download and host applications for deployment to enrolled devices either forced by an administrator or in an application store format which users can manually download approved apps from.

6.4.9 Specification of Management Functions (FMT_SMF_EXT.4)

Enrollment of devices in Intune requires a tenant administrator to define an existing user or create a new user account with password. Users must also have an assigned Microsoft Intune or Microsoft Enterprise Mobility + Security E5 license. The user must also install the Microsoft Company Portal App which facilitates the enrollment of the device when a user initiates the registration process. During

enrollment, the MDM agent interacts with the platform to facilitate the import of the MDM server public key to authenticate and secure future communications with the MDM system.

When successfully enrolled, the device can then receive commands and policy updates from Intune during regular check in intervals which are configured by the TOE and initiated by the device.

By default, users cannot unenroll or otherwise remove their device from the Company Portal app until it has been removed from Intune management. Once an enrolled device has been removed from Intune and is no longer managed by the organization, the user will lose all access to organization data, apps, and configurations.

More information on removing an Android device from Company Portal can be found here https://docs.microsoft.com/en-us/mem/intune/user-help/unenroll-your-device-from-intune-android

More information on removing an iOS device from Company Portal can be found here https://docs.microsoft.com/en-us/mem/intune/user-help/unenroll-your-device-from-intune-ios

6.4.10 Security Management Roles (FMT_SMR.1(1))

All Intune users must be assigned in Intune license before they can interact with Intune services. Mobile device users will not be able to enroll devices unless they are assigned an Intune License.

Intune provides multiple built-in roles to facilitate a granular role-based access control model within the Intune Tenant. These roles are a subset of the 'Administrator' role (referred to as 'Tenant Admin' within Intune) as required by the SFR and provide scoped permissions based on the role function. Any user assigned one or more of these built-in roles is considered an administrator within the TOE and its environment.

To create, edit, or assign administrator roles, an administrator of an Intune Tenant must be assigned one of the two following roles in Azure AD:

- **Global Administrator.** Provides access to all administrative features in Azure Active Directory, as well as services that use Azure AD identities, including Intune.
- **Intune Administrator.** Provides access to all administrative features within Intune, including access to all Intune audit data, management of users and devices, and the ability to create and manage groups.
  Note: In the Microsoft Graph API, this role is identified as "Intune Service Administrator".

Intune built-in administrator roles cannot be deleted. The name, description, type, or permissions assigned to the built-in administrator role also cannot be deleted or modified. The following built-in administrator roles are available for assigning administrative permissions within the Intune Tenant:

a) **Application Manager.** Manages mobile and managed applications, can read device information and can view device configuration profiles.
b) **Help Desk Operator.** Performs remote tasks on users and devices; can assign applications or policies to users or devices.
c) **Intune Role Administrator.** Manages custom Intune roles and adds assignments for built-in Intune roles. It's the only Intune role that can assign permissions to Administrators.

d) **Policy and Profile Manager.** Manages compliance policy, configuration profiles, Apple enrollment, corporate device identifiers, and security baselines.
e) **Read Only Operator.** Views user, device, enrollment, configuration, and application information. Can't make changes to Intune.

If a built-in administrator role does not satisfy the requirements of an administrative function or organization, custom roles can be created with custom permission sets.

More information on the role-based access control implementation in Intune can be found here https://docs.microsoft.com/en-us/mem/intune/fundamentals/role-based-access-control

The Microsoft Service Operator role is only available to specific Microsoft personnel who administer the Intune cloud infrastructure and is not available to customers.

6.4.11 Security Management Roles (MAS Server) (FMT_SMR.1(2))

The TOE also leverages user and device groups in order to assign specific applications. These groups can be used to apply device policies, or exclude specific devices or users from downloading a specified app. An Intune administrator must be assigned one of the following administrator roles, as described in section 6.4.10, to manage applications and application assignments to the user and device groups:

- Global Administrator
- Intune Administrator
- Application Manager

6.4.12 User Unenrollment Prevention (FMT_UNR_EXT.1)

Intune administrators can modify the actions taken when a user attempts to unenroll a device. By default, when a user removes the management profile and uninstalls the Company Portal App, all corporate data stored on the device or granted access to via Intune policies is wiped from the device. An administrator may also forcibly unenroll the device via the Microsoft Intune Admin Center. If this action is taken, the command will be received by the device when it connects to a network and checks in with Intune. This action will also wipe the device of corporate data.

6.5 Protection of the TSF

6.5.1 Use of Supported Services and API's (FPT_API_EXT.1)

The TOE leverages The Cryptography API: Next Generation (CNG) for its cryptographic needs. The underlying Windows environment exclusively uses CNG and provides public API's for external developers.

More information on CNG can be found here:
https://docs.microsoft.com/en-us/windows/win32/seccng/cng-portal

The TOE also employs the use of the Microsoft Graph API. This RESTful web API enables access to Microsoft Cloud service resources including access to Intune resources and audit logs.

More information on Graph API can be found here:
https://docs.microsoft.com/en-us/graph/use-the-api

MDM agents for Android devices only leverage the API's provided by the Android platform. The following Android platform APIs are used:

- **com.google.android.gms.security.ProviderInstaller**
  - ProviderInstaller.installIfNeeded
- **android.security.KeyChain**
  - Keychain.getCertificateChain
- **android.security.KeyChainAliasCallback**
  - KeyChainAliasCallback
- **android.security.keystore**
  - KeyGenParameterSpec
  - KeyInfo
  - KeyProperties
  - KeyProtection
  - StrongBoxUnavailableException

More information on Android platform API's can be found here:
https://developer.android.com/reference

MDM agents for iOS devices leverage the iOS platform API's in support of cryptographic services.

6.5.2 Internal TOE TSF Data Transfer (MDM Agent) (FPT_ITT.1(2))

The TOE leverages the underlying Windows platform to implement an encrypted HTTPS channel between the MDM agent and the Intune server to protect all data from modification and disclosure while in transit.

6.5.3 Use of Third Party Libraries (FPT_LIB_EXT.1)

Microsoft Intune is provided as a service, and includes or relies on third-party software and libraries listed in a non-public Library Document. The MDM Agent includes or relies on the third-party software and libraries listed in Annex C.

6.5.4 Functionality Testing (FPT_TST_EXT.1)

The TOE performs self-tests and integrity checking on its components at start up. The platform on which the TOE operates will verify the integrity of any executable code and ensures drivers, software updates, and applications are digitally signed. Public-key cryptography is used to implement code signing and verify digital signatures by checking that hashes match their expected values.

6.5.5 Trusted Update (FPT_TUD_EXT.1)

The TOE verifies the integrity of program code using a combination of Secure Boot and Code integrity capabilities in Windows. Any system or application updates are code signed and verified at startup. Intune is provided as a service by Microsoft, therefore any updates to the service are provided by Microsoft and securely executed using the methods mentioned above.

6.6 Trusted Path/Channels

6.6.1 Trusted Channel (FTP_ITC_EXT.1)

The TOE leverages the platform to implement HTTPS between itself and enrolled devices to secure all communications and provide trusted channels to MDM agents internal to the TOE (Android devices) and external to the TOE (Apple iOS agent) and are logically distinct from other communication channels as specified in FTP_ITC.1(2). The TOE issues device certificates upon enrollment which are used to secure these communication channels and ensure confidentiality and integrity of data in transit.

6.6.2 Inter-TSF Trusted Channel (Authorized IT Entities) (FTP_ITC.1(1)) and Inter-TSF Trusted Channel (MDM Agent) (FTP_ITC.1(2))

The TOE leverages the platform to provide several trusted channels to protect data in transit to ensure confidentiality and integrity. HTTPS is used to provide trusted communications between the TOE and authorized IT entities including the use of Graph API for external audit log access.

HTTPS is also used for secure communications between the TOE and services provided by the TOE platform such as Azure Active Directory. Device agents also leverage HTTPS when initiating communications with Intune during scheduled check-ins and policy updates.

The underlying Windows platform is responsible for leveraging Schannel to implement these trusted channels.

More information on Schannel can be found here:
https://docs.microsoft.com/en-us/windows-server/security/tls/tls-ssl-schannel-ssp-overview

The following table summarizes the TLS RFCs implemented by the TOE platform.

Table 22 TLS RFCs Implemented by Windows

RFC #	Name	How Used
3268	Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)	Specifies additional ciphersuites implemented by Windows.
4346	The Transport Layer Security (TLS) Protocol Version 1.1	Specifies requirements for TLS 1.1.
4366	Transport Layer Security (TLS) Extensions	Obsoletes RFC 3546 Requirements for TLS 1.1 extensions implemented by Windows.
4492	Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)	Specifies additional ciphersuites implemented by Windows.
4681	TLS User Mapping Extension	Extends TLS to include a User Principal Name during the TLS handshake.
5246	The Transport Layer Security (TLS) Protocol Version 1.2	Obsoletes RFCs 3268, 4346, and 4366. Specifies requirements for TLS 1.2.
5289	TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)	Specifies additional ciphersuites implemented by Windows.

6.6.3 Trusted Path (For Remote Administration) (FTP_TRP.1(1))

The TOE leverages the platform to provide a trusted communication path for remote administration. HTTPS is used to secure communications between Microsoft Intune Admin Center and the remote administrator when connected via modern web browser. This method allows for secure access to the security functions of the TOE.

6.6.4 Trusted Path (For Enrollment) (FTP_TRP.1(2))

The TOE leverages the platform to provide a TLS channel for secure communications between enrolled devices and Intune management. A device will initiate this connection by logging into the Microsoft Intune Company Portal app on the device to be enrolled. During enrollment, the device receives a certificate from Intune which is then used for authentication of all future TLS sessions.

7 Protection Profile Conformance Claim

This Security Target is in compliance with the Protection Profile for Mobile Device Management, Version 4.0, April 25, 2019 (PP_MDM_V4.0) and the PP-Module for MDM Agents, Version 1.0, April 25, 2019 (MDM Agent PP-Mod).

The requirements in the protection profile are assumed to represent a complete set of requirements that serve to address any interdependencies. All security functional requirements and objectives in this security target have been reproduced from the protection profile and PP-Module so that all dependencies between SFRs are satisfied by the inclusion of the relevant components.

8 Rationale

8.1 Rationale for SFR Mapping to Security Objectives

This section provides a mapping of SFR's to the security objectives of the TOE. The rationale for this mapping is per the MDM PP and MDM Agent PP-Module and is further described in the TOE Summary Specification section of this ST.

Table 23 Mapping Between SFRs and Security Objectives (PP_MDM_V4.0)

Objective	SFR	Rationale
O.ACCOUNTABILITY – The TOE must provide logging facilities which record management actions undertaken by its administrators.	FAU_ALT_EXT.1	The PP includes FAU_ALT_EXT.1 to define the ability of the TSF to generate alerts when certain actions occur.
	FAU_GEN.1(1)	The PP includes FAU_GEN.1(1) to require the TSF to generate audit records of security-relevant events, including management actions.
	FAU_NET_EXT.1	The PP includes FAU_NET_EXT.1 to require the TSF to record the connectivity status of enrolled agents.

Objective	SFR	Rationale
	FAU_STG_EXT.1	The PP includes FAU_STG_EXT.1 for the TSF to securely transmit its audit data to an external entity.
	FAU_GEN.1(2) (selection- based)	The PP includes FAU_GEN.1(2) to require the TSF to generate records of MAS Server functionality if the TSF supports this capability.
	FAU_STG_EXT.2 (selection-based)	The PP includes FAU_STG_EXT.2 to require the TSF to protect stored audit records from unauthorized modification if these records are stored locally.
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.	FIA_ENR_EXT.1	The PP includes FIA_ENR_EXT.1 for the TSF to perform the initial enrollment of devices into management, including applying restrictions on what devices can be enrolled.
	FMT_MOF.1(1)	The PP includes FMT_MOF.1(1) to define the supported TSF management functions, including those used to enable, disable, and apply policies to enrolled devices.
	FMT_MOF.1(2)	The PP includes FMT_MOF.1(2) for the TSF to restrict the enrollment process to authorized administrators and mobile device users.
	FMT_SMF.1(1)	The PP includes FMT_SMF.1(1) to specify that the TSF is capable of sending configuration information and enterprise security policies to an MDM Agent.
	FMT_MOF.1(3) (selection-based)	The PP includes FMT_MOF.1(3) to enforce restrictions on access to the MAS Server from enrolled devices based on applied policies.
	FMT_SMF.1(3) (selection-based)	The PP includes FMT_SMF.1(3) to specify that the TSF is capable of configuring the MAS Server to enforce restrictions on enrolled devices attempting to access it.
	FMT_SMR.1(2) (selection-based)	The PP includes FMT_SMR.1(2) to define roles on the MAS Server, if this capability is supported, that are used to determine the extent to which enrolled devices can access data on the MAS Server.
O.DATA_PROTECTION_TRANSIT - Data exchanged between the MDM Server and the MDM Agent must be protected from	FAU_STG_EXT.1	The PP includes FAU_STG_EXT.1 which requires the TSF to use a trusted channel for the external transmission of audit data.

Objective	SFR	Rationale
being monitored, accessed, or altered.	FCS_CKM_EXT.4	The PP includes FCS_CKM_EXT.4 to ensure that the TSF destroys plaintext keying material and critical security parameters when no longer needed in support of securing data in transit.
	FCS_CKM.1	The PP includes FCS_CKM.1(1) to define whether the TSF or the platform generates asymmetric keys that are used in support of securing data in transit.
	FCS_CKM.2	The PP includes FCS_CKM.2 to define whether the TSF or the platform performs key establishment in support of securing data in transit.
	FCS_COP.1(1)	The PP includes FCS_COP.1(1) to define the symmetric AES encryption algorithms used in support of securing data in transit.
	FCS_COP.1(2)	The PP includes FCS_COP.1(2) to define the hash algorithms used in support of securing data in transit.
	FCS_COP.1(3)	The PP includes FCS_COP.1(3) to define the digital signature algorithms used in support of securing data in transit.
	FCS_COP.1(4)	The PP includes FCS_COP.1(4) to define the HMAC algorithms used in support of securing data in transit.
	FCS_RBG_EXT.1 FCS_RBG_EXT.1/ ANDROID	The PP includes FCS_RBG_EXT.1 and FCS_RBG_EXT.1/ANDROID to define whether random bit generation services are implemented by the TSF or the platform. The TOE may rely on the use of a random bit generator to create keys that are subsequently used in support of securing data in transit.
	FCS_STG_EXT.1	The PP includes FCS_STG_EXT.1 to define whether the TSF or the Operational Environment protects key data that may be used in support of securing data in transit.
	FIA_ENR_EXT.1	The PP includes FIA_ENR_EXT.1 which requires the TSF to use a trusted channel for the agent enrollment process.
	FIA_X509_EXT.1(1)	The PP includes FIA_X509_EXT.1(1) to define validation rules for X.509 certificates that may be used in support of securing data in transit.

Objective	SFR	Rationale
	FIA_X509_EXT.2	The PP includes FIA_X509_EXT.2 to define the TOE functions that support the use of X.509 certificates. This includes protection of data in transit.
	FIA_CLI_EXT.1	The PP includes FIA_CLI_EXT.1 to require the TSF to enforce uniqueness for client certificates that are used in support of securing data in transit.
	FTP_ITC_EXT.1	The PP includes FTP_ITC_EXT.1 to define the trusted channels used by the TOE where security of data in transit is enforced.
	FTP_ITC.1(1)	The PP includes FTP_ITC.1(1) to define a trusted communication channel between itself and trusted external servers.
	FTP_TRP.1(1)	The PP includes FTP_TRP.1(1) to define requirements for securing data in transit for administrative communications.
	FTP_TRP.1(2)	The PP includes FTP_TRP.1(2) to define requirements for securing data in transit for agent enrollment.
	FTP_ITC.1(2) (selection-based)	The PP includes FTP_ITC.1(1) to define a trusted communication channel between itself and an MDM agent if the MDM agent is not part of the TOE.
O.INTEGRITY - 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.	FCS_COP.1(2)	The PP includes FCS_COP.1(2) to require the TSF to include a mechanism to cryptographically assert and verity the integrity of data using a hash algorithm.
	FCS_COP.1(3)	The PP includes FCS_COP.1(3) to require the TSF to include a mechanism to cryptographically assert and verify the integrity of data using a digital signature.
	FIA_X509_EXT.2	The PP includes FIA_X509_EXT.2 to define the TOE functions that support the use of X.509 certificates. This includes code signing for system software updates, integrity verification, and policy signing.
	FMT_POL_EXT.1	The PP includes FMT_POL_EXT.1 to ensure the integrity of the policies and policy updates to the MDM Agent are digitally signed to protect their integrity.
	FPT_TST_EXT.1	The PP includes FPT_TST_EXT.1 to require The TSF to run a suite of self tests to ensure the correct operation of the TSF

Objective	SFR	Rationale
		and the integrity of stored TSF executable code.
	FPT_TUD_EXT.1	The PP includes FPT_TUD_EXT.1 to define requirements for trusted update of TSF executable code, including that the integrity of this update data can be verified.
O.MANAGEMENT - The TOE provides access controls around its management functionality.	FIA_UAU.1	The PP includes FIA_UAU.1 to require the TSF to enforce access control on its management interface by requiring user authentication.
	FMT_MOF.1(1)	The PP includes FMT_MOF.1(1) for the TSF to restrict the functions to enable, disable, modify, and monitor functions and policies to authorized administrators.
	FMT_MOF.1(2)	The PP includes FMT_MOF.1(2) to restrict the enrollment process to authorized administrators and mobile device users.
	FMT_SMF.1(1)	The PP includes FMT_SMF.1(1) to define the security-relevant management functions that the MDM server is capable of communicating to the MDM Agent.
	FMT_SMF.1(2)	The PP includes FMT_SMF.1(2) to define the security-relevant management functions that the MDM server has for its own configuration.
	FMT_SMR.1(1)	The PP includes FMT_SMR.1(1) to define the security management roles that are used as the basis for access control enforcement.
	FMT_SMF.1(3) (selection-based)	The PP includes FMT_SMF.1(3) to define the MAS Server management functionality if this capability is supported.
	FMT_SMR.1(2) (selection-based)	The PP includes FMT_SMR.1(2) to define the management roles that apply to the MAS Server if this capability is supported.
O.QUALITY - 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	FPT_API_EXT.1	The PP includes FPT_API_EXT.1 to enforce quality of implementation by ensuring that the MDM software uses only documented platform APIs to supports its security functionality.

Objective	SFR	Rationale
services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs.		
	FPT_LIB_EXT.1	The PP includes FPT_LIB_EXT.1 to enforce quality of implementation by ensuring that the MDM software does not include any unnecessary or unexpected third-party libraries which could present a privacy threat or vulnerability.

Application Note: This table has been modified by TD0552.

Table 24 Mapping Between SFRs and Security Objectives (MDM Agent PP-Mod)

Objective	SFR	Rationale
O.ACCOUNTABILITY - The TOE must provide logging facilities which record management actions undertaken by its administrators.	FAU_ALT_EXT.2	The PP-Module includes FAU_ALT_EXT.2 to support this objective by requiring the TSF to generate alerts back to an MDM Server when security-relevant events occur.
	FAU_GEN.1(2)	The PP-Module includes FAU_GEN.1(2) to support this objective by defining the security-relevant events for which the TSF must generate audit records.
	FAU_SEL.1(2)	The PP-Module includes FAU_SEL.1(2) to support this objective by defining how the set of audited events can be configured.
	FAU_STG_EXT.3 (objective)	The PP-Module includes FAU_STG_EXT.3 to support this objective by optionally requiring the TSF to use platform-provided storage for the security-relevant audit data it generates.
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	FIA_ENR_EXT.2	The PP-Module includes FIA_ENR_EXT.2 to support this objective by requiring the TSF to identify the MDM Server so that the authenticity of policy updates can be determined.
	FMT_POL_EXT.2	The PP-Module includes FMT_POL_EXT.2 to support this objective by requiring the TSF to only accept policy data that can prove its authenticity with a digital certificate.
	FMT_SMF_EXT.4	The PP-Module includes FMT_SMF_EXT.4 to support this objective by defining the

Objective	SFR	Rationale
unenrollment from management services.		management functions the TSF must implement to support its own configuration.
	FMT_UNR_EXT.1	The PP-Module includes FMT_UNR_EXT.1 to support this objective by preventing a user-directed unenrollment operation that would allow for MDM policies to be ignored.
	FPT_NET_EXT.1 (objective)	The PP-Module includes FPT_NET_EXT.1 to support this objective by optionally requiring the TSF to detect when a sustained communications outage with the MDM Server has occurred to indicate that the TSF may be deprived of updated policy data.
O.DATA_PROTECTION_TRANSIT - Data exchanged between the MDM Server and the MDM Agent must be protected from being monitored, accessed, or altered.	FCS_DTLSC_EXT.1 (from TLS Package)	The PP-Module includes FCS_DTLSC_EXT.1 from the TLS package by reference to support this objective because DTLS is one of the protocols the PP-Module allows to protect data in transit.
	FCS_DTLSS_EXT.1 (from TLS Package)	The PP-Module includes FCS_DTLSS_EXT.1 from the TLS package by reference to support this objective because DTLS is one of the protocols the PP-Module allows to protect data in transit.
	FCS_TLS_EXT.1 (from TLS Package)	The PP-Module includes FCS_TLS_EXT.1 from the TLS package by reference to support this objective because it is mandatory to claim when the TLS Package applies so that the TSF's usage of TLS is clearly defined.
	FCS_TLSC_EXT.1 (from TLS Package)	The PP-Module includes FCS_TLSC_EXT.1 from the TLS package by reference to support this objective because TLS is one of the protocols the PP-Module allows to protect data in transit.
	FCS_TLSC_EXT.2 (from TLS Package)	The PP-Module includes FCS_TLSC_EXT.2 from the TLS package by reference to support this objective because it requires TLS to be mutually-authenticated if claimed.
	FCS_TLSS_EXT.1 (from TLS Package)	The PP-Module includes FCS_TLSS_EXT.1 from the TLS package by reference to support this objective because TLS is one

Objective	SFR	Rationale
		of the protocols the PP-Module allows to protect data in transit.
	FCS_TLSS_EXT.2 (from TLS Package)	The PP-Module includes FCS_TLSS_EXT.2 from the TLS package by reference to support this objective because it requires TLS to be mutually-authenticated if claimed.
	FTP_ITC_EXT.1(2) (if MDF is Base-PP)	The PP-Module includes FTP_ITC_EXT.1(2) to support this objective because when the TOE is a mobile device, it must be required to ensure that MDM Server communications are protected.
	FTP_TRP.1(2) (if MDF is Base-PP)	The PP-Module includes FTP_ITC_EXT.1(2) to support this objective because when the TOE is a mobile device, it must be required to ensure that MDM Server enrollment communications are protected.
	FPT_ITT.1(2) (from MDM Base-PP)	This SFR is selection-based in the MDM PP but is required when the TOE includes this PP-Module because it is triggered by the MDM Agent being part of the TOE. This SFR supports the objective by defining the trusted channel the TSF must use to secure connectivity between the MDM Server and MDM Agent components of the distributed TOE.
	FPT_NET_EXT.1 (objective)	The PP-Module includes FPT_NET_EXT.1 to support this objective by optionally requiring the TSF to detect when a sustained communications outage with the MDM Server has occurred as a possible indicator of communications issues.
O.STORAGE - 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.	FCS_STG_EXT.4 (if MDF is Base-PP)	The PP-Module includes FCS_STG_EXT.4 to support this objective because when the TOE is a mobile device, it must be required to ensure that MDM Agent key data is stored securely.
	FCS_STG_EXT.1(2) (if MDM is Base-PP)	The PP-Module includes FCS_STG_EXT.1(2) to support this objective because when the TOE is a MDM Server with MDM Agent capability, it must be required to ensure that MDM Agent key data is stored securely.

Application Note: This table has been modified by TD0497.

8.2 Rationale for Security Objectives

For all content reproduced from the protection profile, the corresponding rationale in that protection profile remains applicable to demonstrate the correspondence between the TOE security functional requirements and TOE security objectives. The tables below identify the mapping between security threats or assumptions and security objectives along with a rationale for the mapping.

Table 25 Security Objectives Rationale (PP_MDM_V4.0)

Threat or Assumption	Security Objective	Rationale
T.MALICIOUS_APPS	O.APPLY_POLICY, O.INTEGRITY	The threat T.MALICIOUS_APPS is countered by O.APPLY_POLICY as this provides the capability to limit the ability to install applications on the MD. The threat T.MALICIOUS_APPS is countered by O.INTEGRITY as this provides the capability to perform self-tests to ensure the integrity of critical functionality, software/firmware and data has been maintained.
T.NETWORK_ATTACK	O.DATA_PROTECTION_TRANSIT	The threat T.NETWORK_ATTACK is countered by O.DATA_PROTECTION_TRANSIT as this provides authentication of the endpoints of a trusted communication path.
T.NETWORK_EAVESDROP	O.DATA_PROTECTION_TRANSIT, O.QUALITY	The threat T.NETWORK_EAVESDROP is countered by O.DATA_PROTECTION_TRANSIT as this provides the capability to communicate using one (or more) standard protocols as a means to maintain the confidentiality of data that are transmitted outside and within the TOE. The threat T.NETWORK_EAVESDROP is countered by O.QUALITY as this provides the capability to invoke platform resources to ensure quality of implementation.

Threat or Assumption	Security Objective	Rationale
T.PHYSICAL_ACCESS	O.APPLY_POLICY	The threat T.PHYSICAL_ACCESS is countered by O.APPLY_POLICY as this provides the capability to configure and apply security policies to ensure the Mobile Device can protect user and enterprise data that it may store or process.
A.COMPONENTS_RUNNING (applies to distributed TOEs only)	OE.COMPONENTS_RUNNING	The operational environment objective OE.COMPONENTS_RUNNING is realized through A.COMPONENTS_RUNNING.
A.CONNECTIVITY	OE.WIRELESS_NETWORK	The operational environment objective OE.WIRELESS_NETWORK is realized through A.CONNECTIVITY.
A.MDM_SERVER_PLATFORM	OE.TIMESTAMP	The operational environment objective OE.TIMESTAMP is realized through A.MDM_SERVER_PLATFORM.
A.PROPER_ADMIN	OE.PROPER_ADMIN	The operational environment objective OE.PROPER_ADMIN is realized through A.PROPER_ADMIN.
A.PROPER_USER	OE.PROPER_USER	The operational environment objective OE.PROPER_USER is realized through A.PROPER_USER.
P.ACCOUNTABILITY	O.ACCOUNTABILITY	The organizational security policy O.ACCOUNTABILITY is realized through P.ACCOUNTABILITY.
P.ADMIN	OE.PROPER_ADMIN	The organizational security policy P.ADMIN is realized through OE.PROPER_ADMIN.
P.DEVICE_ENROLL	OE.IT_ENTERPRISE	The organizational security policy P.DEVICE_ENROLL is realized through OE.IT_ENTERPRISE.
P.NOTIFY	OE.PROPER_USER	The organizational security policy P.NOTIFY is realized through OE.PROPER_USER.

Table 26 Security Objectives Rationale (MDM Agent PP-Mod)

Threat or Assumption	Security Objective	Rationale
T.MALICIOUS_APPS	O.DATA_PROTECTION_TRANSIT, O.APPLY_POLICY	The threat T.MALICIOUS_APPS is countered by O.DATA_PROTECTION_TRANSIT as this provides the capability to protect app loading/updates against malicious insertion from the network. The threat T.MALICIOUS_APPS is countered by O.APPLY_POLICY as this provides policy preventing loading of unapproved apps into the TOE.
T.BACKUP	O.DATA_PROTECTION_TRANSIT, O.APPLY_POLICY	The threat T.BACKUP is countered by O.DATA_PROTECTION_TRANSIT as this provides the capability to communicate using one (or more) standard protocols as a means to maintain the confidentiality of data that are transmitted between the Agent and other entities. The threat T.BACKUP is countered by O.APPLY_POLICY as this provides policy to enforce that backups be stored only in secure, protected locations.
T.NETWORK_ATTACK	O.DATA_PROTECTION_TRANSIT, O.APPLY_POLICY, OE.IT_ENTERPRISE	The threat T.NETWORK_ATTACK is countered by O.DATA_PROTECTION_TRANSIT as this provides the capability to communicate using one (or more) standard protocols as a means to maintain the confidentiality of data that are transmitted between the Agent and other entities. The threat T.NETWORK_ATTACK is countered by O.APPLY_POLICY as this provides a secure configuration of the Agent to protect data that it processes. The threat T.NETWORK_ATTACK is countered by OE.IT_ENTERPRISE by reducing

Threat or Assumption	Security Objective	Rationale
		the network exposure of the mobile device.
T.NETWORK_EAVESDROP	O.DATA_PROTECTION_TRANSIT, O.APPLY_POLICY, OE.IT_ENTERPRISE	The threat T.NETWORK_EAVESDROP is countered by O.DATA_PROTECTION_TRANSIT as this provides the capability to communicate using one (or more) standard protocols as a means to maintain the confidentiality of data that are transmitted between the Agent and other entities. The threat T.NETWORK_EAVESDROP is countered by O.APPLY_POLICY as this provides a secure configuration of the Agent to protect data that it processes. The threat T.NETWORK_EAVESDROP is countered by OE.IT_ENTERPRISE by reducing the network exposure of the mobile device.
T.PHYSICAL_ACCESS	O.ACCOUNTABILITY, O.APPLY_POLICY, O.STORAGE	The threat T.PHYSICAL_ACCESS is countered by O.ACCOUNTABILITY as this provides the capability to log attempts by unauthorized personnel to access data, and to log any access to the data or the device, as well as changes to the device during the time when it is not under the control of an authorized user. The threat T.PHYSICAL_ACCESS is countered by O.APPLY_POLICY as this provides a secure configuration of the Agent to protect data that it processes. The threat T.PHYSICAL_ACCESS is countered by O.STORAGE as this provides the capability to encrypt all user and enterprise data and authentication keys to ensure the confidentiality of data that it stores.

Threat or Assumption	Security Objective	Rationale
A.CONNECTIVITY	OE.WIRELESS_NETWORK	The Operational Environment objective OE.WIRELESS_NETWORK is realized through A.CONNECTIVITY.
A.MOBILE_DEVICE_PLATFORM	OE.MOBILE_DEVICE_PLATFORM	The Operational Environment objective OE.MOBILE_DEVICE_PLATFORM is realized through A.MOBILE_DEVICE_PLATFORM.
A.PROPER_ADMIN	OE.DATA_PROPER_ADMIN	The Operational Environment objective OE.DATA_PROPER_ADMIN is realized through A.PROPER_ADMIN.
A.PROPER_USER	OE.DATA_PROPER_USER	The Operational Environment objective OE.DATA_PROPER_USER is realized through A.PROPER_USER.
P.ACCOUNTABILITY	O.ACCOUNTABILITY	O.ACCOUNTABILITY provides logging of personnel actions in order to provide accountability of all personnel actions within the TOE.
P.ADMIN	O.APPLY_POLICY	The TOE adheres to the Enterprise security policy through the application of O.APPLY_POLICY.
P.DEVICE_ENROLL	O.APPLY_POLICY	The TOE enrolls mobile devices for specific users with policy through the application of O.APPLY_POLICY.
P.NOTIFY	O.APPLY_POLICY	The TOE provides the capability for the administrator to apply remediation actions via the MDM system through policy, which is applied through O.APPLY_POLICY.

8.3 Rationale for Security Functional Requirements Operations

This section provides a rationale that describes how the Security Target reproduced the security functional requirements from the MDM Protection Profile and MDM Agent PP-Module. The table below describes the mapping of security requirements from the protection profile and PP-Module SFR's to the security target SFRs along with rationale for operations. SFR operations left incomplete in the protection

profile and PP-Module have been completed in this Security Target and are identified within each SFR In the TOE Security Functional Requirements section.

Table 27 Rational for SFR Operations

PP or PP-Mod	Requirement	ST Requirement	Operation & Rationale
MDM PP	**FAU_ALT_EXT.1.1**	FAU_ALT_EXT.1	A selection containing an assignment which is allowed by the PP.
MDM PP	**FAU_GEN.1.1(1)**	FAU_GEN.1(1)	Multiple selections and a selection containing an assignment which is allowed by the PP.
MDM PP	**FAU_GEN.1.2(1)**	FAU_GEN.1(1)	An assignment which is allowed by the PP.
MDM PP	**FAU_GEN.1.1(2)**	FAU_GEN.1(2)	No selections or assignments
MDM PP	**FAU_GEN.1.2(2)**	FAU_GEN.1(2)	A selection and assignment which is allowed by the PP.
MDM PP	**FAU_NET_EXT.1.1**	FAU_NET_EXT.1	No selections or assignments.
MDM PP	**FAU_STG_EXT.1.1**	FAU_STG_EXT.1	A selection which is allowed by the PP.
MDM PP	**FAU_STG_EXT.2.1**	FAU_STG_EXT.2	A selection which is allowed by the PP.
PP-Mod	**FAU_ALT_EXT.2.1**	FAU_ALT_EXT.2	Multiple selections and an assignment which is allowed by the PP-Module.
PP-Mod	**FAU_ALT_EXT.2.2**	FAU_ALT_EXT.2	No selections or assignments.
PP-Mod	**FAU_GEN.1.1(2)**	FAU_GEN.1(2)	Multiple selections which are allowed by the PP-Module.
PP-Mod	**FAU_GEN.1.2(2)**	FAU_GEN.1(2)	A selection and an assignment which are allowed by the PP-Module.
PP-Mod	**FAU_SEL.1.1(2)**	FAU_SEL.1(2)	A selection and an assignment which are allowed by the PP-Module.
MDM PP	**FCS_CKM.1.1**	FCS_CKM.1	Multiple selections which are allowed by the PP.
MDM PP	**FCS_CKM.2.1**	FCS_CKM.2	Multiple selections which are allowed by the PP.
MDM PP	**FCS_CKM_EXT.4.1**	FCS_CKM_EXT.4	Multiple selections which are allowed by the PP.
MDM PP	**FCS_CKM_EXT.4.2**	FCS_CKM_EXT.4	No selections or assignments.
MDM PP	**FCS_COP.1.1(1)**	FCS_COP.1(1)	Multiple selections which are allowed by the PP.
MDM PP	**FCS_COP.1.1(2)**	FCS_COP.1(2)	Multiple selections which are allowed by the PP.

PP or PP-Mod	Requirement	ST Requirement	Operation & Rationale
MDM PP	**FCS_COP.1.1(3)**	FCS_COP.1(3)	Multiple selections which are allowed by the PP.
MDM PP	**FCS_COP.1.1(4)**	FCS_COP.1(4)	Multiple selections and an assignment which are allowed by the PP.
MDM PP	**FCS_RBG_EXT.1.1**	FCS_RBG_EXT.1	Multiple selections which are allowed by the PP.
MDM PP	**FCS_RBG_EXT.1.2**	FCS_RBG_EXT.1	Multiple selections which are allowed by the PP.
MDM PP	**FCS_RBG_EXT.1.1 /ANDROID**	FCS_RBG_EXT.1 /ANDROID	Multiple selections which are allowed by the PP.
MDM PP	**FCS_RBG_EXT.1.2 /ANDROID**	FCS_RBG_EXT.1 /ANDROID	Multiple selections which are allowed by the PP.
MDM PP	**FCS_STG_EXT.1.1**	FCS_STG_EXT.1	A selection which is allowed by the PP.
PP-Mod	**FCS_STG_EXT.1.1(2)**	FCS_STG_EXT.1(2)	No selections or assignments.
MDM PP	**FIA_ENR_EXT.1.1**	FIA_ENR_EXT.1	No selections or assignments.
MDM PP	**FIA_ENR_EXT.1.2**	FIA_ENR_EXT.1	Multiple selections and assignments which are allowed by the PP.
MDM PP	**FIA_UAU.1.1**	FIA_UAU.1	A selection and assignment which are allowed by the PP.
MDM PP	**FIA_UAU.1.2**	FIA_UAU.1	A selection which is allowed by the PP.
MDM PP	**FIA_X509_EXT.1.1(1)**	FIA_X509_EXT.1(1)	Multiple selection which are allowed by the PP.
MDM PP	**FIA_X509_EXT.1.2(1)**	FIA_X509_EXT.1(1)	A selection which is allowed by the PP.
MDM PP	**FIA_X509_EXT.2.1**	FIA_X509_EXT.2	Multiple selections which are allowed by the PP.
MDM PP	**FIA_X509_EXT.2.2**	FIA_X509_EXT.2	Multiple selections which are allowed by the PP.
MDM PP	**FIA_CLI_EXT.1.1**	FIA_CLI_EXT.1	A selection which is allowed by the PP.
PP-Mod	**FIA_ENR_EXT.2.1**	FIA_ENR_EXT.2	No selections or assignments
MDM PP	**FMT_MOF.1.1(1)**	FMT_MOF.1(1)	A selection which is allowed by the PP.
MDM PP	**FMT_MOF.1.1(2)**	FMT_MOF.1(2)	No selections or assignments.
MDM PP	**FMT_MOF.1.1(3)**	FMT_MOF.1(3)	No selections or assignments.
MDM PP	**FMT_POL_EXT.1.1**	FMT_POL_EXT.1	No selections or assignments.

PP or PP-Mod	Requirement	ST Requirement	Operation & Rationale
MDM PP	**FMT_SMF.1.1(1)**	FMT_SMF.1(1)	Multiple selections and assignments which are allowed by the PP.
MDM PP	**FMT_SMF.1.1(2)**	FMT_SMF.1(2)	Multiple selections and assignments which are allowed by the PP.
MDM PP	**FMT_SMF.1.1(3)**	FMT_SMF.1(3)	A selection which is allowed by the PP.
MDM PP	**FMT_SMR.1.1(1)**	FMT_SMR.1(1)	A selection containing an assignment which is allowed by the PP.
MDM PP	**FMT_SMR.1.2(1)**	FMT_SMR.1(1)	No selections or assignments.
MDM PP	**FMT_SMR.1.1(2)**	FMT_SMR.1(2)	An assignment which is allowed by the PP.
MDM PP	**FMT_SMR.1.2(2)**	FMT_SMR.1(2)	No selections or assignments.
PP-Mod	**FMT_POL_EXT.2.1**	FMT_POL_EXT.2	No selections or assignments.
PP-Mod	**FMT_POL_EXT.2.2**	FMT_POL_EXT.2	No selections or assignments.
PP-Mod	**FMT_SMF_EXT.4.1**	FMT_SMF_EXT.4	Multiple selections which are allowed by the PP Module.
PP-Mod	**FMT_SMF_EXT.4.2**	FMT_SMF_EXT.4	A selection which is allowed by the PP Module.
PP-Mod	**FMT_UNR_EXT.1.1**	FMT_UNR_EXT.1	A selection which is allowed by the PP Module.
MDM PP	**FPT_API_EXT.1.1**	FPT_API_EXT.1	No selections or assignments.
MDM PP	**FPT_ITT.1.1(2)**	FPT_ITT.1(2)	Multiple selections which are allowed by the PP Module.
MDM PP	**FPT_LIB_EXT.1.1**	FPT_LIB_EXT.1	An assignment which is allowed by the PP.
MDM PP	**FPT_TST_EXT.1.1**	FPT_TST_EXT.1	No selections or assignments.
MDM PP	**FPT_TST_EXT.1.2**	FPT_TST_EXT.1	Multiple selections which are allowed by the PP.
MDM PP	**FPT_TUD_EXT.1.1**	FPT_TUD_EXT.1	No selections or assignments. Refined per TD0438.
MDM PP	**FPT_TUD_EXT.1.2**	FPT_TUD_EXT.1	A selection which is allowed by the PP.
MDM PP	**FPT_TUD_EXT.1.3**	FPT_TUD_EXT.1	A selection which is allowed by the PP.
MDM PP	**FTP_ITC_EXT.1.1**	FTP_ITC_EXT.1	Multiple selections which are allowed by the PP.
MDM PP	**FTP_ITC.1.1(1)**	FTP_ITC.1(1)	Multiple selections and an assignment which are allowed by the PP.

PP or PP-Mod	Requirement	ST Requirement	Operation & Rationale
MDM PP	**FTP_ITC.1.2(1)**	FTP_ITC.1(1)	A selection which is allowed by the PP.
MDM PP	**FTP_ITC.1.3(1)**	FTP_ITC.1(1)	A selection and an assignment which are allowed by the PP.
MDM PP	**FTP_ITC.1.1(2)**	FTP_ITC.1(2)	Multiple selections which are allowed by the PP.
MDM PP	**FTP_ITC.1.2(2)**	FTP_ITC.1(2)	A selection which is allowed by the PP.
MDM PP	**FTP_ITC.1.3(2)**	FTP_ITC.1(2)	A selection which is allowed by the PP.
MDM PP	**FTP_TRP.1.1(1)**	FTP_TRP.1(1)	Multiple selections which are allowed by the PP.
MDM PP	**FTP_TRP.1.2(1)**	FTP_TRP.1(1)	A selection which is allowed by the PP.
MDM PP	**FTP_TRP.1.3(1)**	FTP_TRP.1(1)	A selection which is allowed by the PP.
MDM PP	**FTP_TRP.1.1(2)**	FTP_TRP.1(2)	Multiple selections which are allowed by the PP.
MDM PP	**FTP_TRP.1.2(2)**	FTP_TRP.1(2)	A selection which is allowed by the PP.
MDM PP	**FTP_TRP.1.3(2)**	FTP_TRP.1(2)	A selection which is allowed by the PP.

8.4 Rationale for the TOE Summary Specification

This section in conjunction with the TOE Summary Specification (TSS), provides evidence that the security functions are suitable to meet the TOE security requirements. The security functions claimed in this security target work together to provide all the security functionality offered by the TOE and are reproduced as per the requirements of the MDM PP and MDM Agent PP-Module.

The security functions described in the TOE Summary Specification and listed in the table below are all necessary for the required security functionality in the TSF.

Table 28 SFR Mapping to Security Objectives

Requirement	Security Audit	Cryptographic Support	Identification & Authentication	Security Management	Protection of the TSF	Trusted Path / Channel
Protection Profile for Mobile Device Management						
FAU_ALT_EXT.1	X					
FAU_GEN.1(1)	X					
FAU_GEN.1(2)	X					
FAU_NET_EXT.1	X					
FAU_STG_EXT.1	X					
FAU_STG_EXT.2	X					
FCS_CKM.1		X				
FCS_CKM.2		X				
FCS_CKM_EXT.4		X				
FCS_COP.1(1)		X				
FCS_COP.1(2)		X				
FCS_COP.1(3)		X				
FCS_COP.1(4)		X				
FCS_RBG_EXT.1		X				
FCS_RBG_EXT.1/ANDROID		X				
FCS_STG_EXT.1		X				
FIA_ENR_EXT.1			X			
FIA_UAU.1			X			
FIA_X509_EXT.1(1)			X			
FIA_X509_EXT.2			X			
FIA_CLI_EXT.1			X			
FMT_MOF.1(1)				X		
FMT_MOF.1(2)				X		
FMT_MOF.1(3)				X		
FMT_POL_EXT.1				X		
FMT_SMF.1(1)				X		
FMT_SMF.1(2)				X		
FMT_SMF.1(3)				X		
FMT_SMR.1(1)				X		
FMT_SMR.1(2)				X		
FPT_API_EXT.1					X	
FPT_ITT.1(2)					X	
FPT_LIB_EXT.1					X	
FPT_TST_EXT.1					X	

Requirement	Security Audit	Cryptographic Support	Identification & Authentication	Security Management	Protection of the TSF	Trusted Path / Channel
FPT_TUD_EXT.1					X	
FTP_ITC_EXT.1						X
FTP_ITC.1(1)						X
FTP_ITC.1(2)						X
FTP_TRP.1(1)						X
FTP_TRP.1(2)						X
PP-Module for MDM Agent						
FAU_ALT_EXT.2	X					
FAU_GEN.1(2)	X					
FAU_SEL.1(2)	X					
FCS_STG_EXT.1(2)		X				
FIA_ENR_EXT.2			X			
FMT_POL_EXT.2				X		
FMT_SMF_EXT.4				X		
FMT_UNR_EXT.1				X		

8.5 Rationale for SFR Mapping to TOE Components

The table below identifies the mapping between TOE components and SFRs to illustrate the nature of the TOE and functions of the MDM Server (Microsoft Intune), MDM Platform, and MDM Agent (Microsoft Company Portal App).

Table 29 SFR Mapping to TOE Components

Requirement	MDM Server	MDM Platform	MDM Agent
Protection Profile for Mobile Device Management			
FAU_ALT_EXT.1	X		
FAU_GEN.1(1)	X	X	
FAU_GEN.1(2)	X		
FAU_NET_EXT.1	X		
FAU_STG_EXT.1	X		
FAU_STG_EXT.2	X		
FCS_CKM.1		X	
FCS_CKM.2		X	
FCS_CKM_EXT.4		X	
FCS_COP.1(1)		X	

Requirement	MDM Server	MDM Platform	MDM Agent
FCS_COP.1(2)		X	
FCS_COP.1(3)		X	
FCS_COP.1(4)		X	
FCS_RBG_EXT.1		X	
FCS_RBG_EXT.1/ANDROID		X	
FCS_STG_EXT.1		X	
FIA_ENR_EXT.1	X		
FIA_UAU.1	X		
FIA_X509_EXT.1(1)		X	
FIA_X509_EXT.2		X	
FIA_CLI_EXT.1	X		
FMT_MOF.1(1)	X		
FMT_MOF.1(2)	X		
FMT_MOF.1(3)	X		
FMT_POL_EXT.1	X		
FMT_SMF.1(1)	X		
FMT_SMF.1(2)	X		
FMT_SMF.1(3)	X		
FMT_SMR.1(1)	X		
FMT_SMR.1(2)	X		
FPT_API_EXT.1	X		
FPT_ITT.1(2)		X	
FPT_LIB_EXT.1	X		
FPT_TST_EXT.1		X	
FPT_TUD_EXT.1		X	
FTP_ITC_EXT.1	X		
FTP_ITC.1(1)		X	
FTP_ITC.1(2)		X	
FTP_TRP.1(1)		X	
FTP_TRP.1(2)		X	
PP-Module for MDM Agent			
FAU_ALT_EXT.2			X
FAU_GEN.1(2)			X
FAU_SEL.1(2)	X		
FCS_STG_EXT.1(2)			X
FIA_ENR_EXT.2			X
FMT_POL_EXT.2			X
FMT_SMF_EXT.4			X
FMT_UNR_EXT.1			X

Annex A – Enterprise Device High Security Use Case

This template lists additional SFR's, selections, and assignments that best support the Enterprise Device High Security use case as described in Appendix G.2 of the MDM PP. Deviations from this template are identified below and do not preclude the TOE from being used in the scenarios identified by the PP.

Requirement	Action	Status	Selection/Assignment Notes
FMT_SMF.1.1(1) Function 15*	Include in ST.	Selected	*"No other x.509v3 certificates"* selected. **Fully supported by tested Samsung Galaxy Knox device and iOS device only.*
FMT_SMF.1.1(1) Function 16	Include in ST.	Selected	*"Alert the user"* selected.
FMT_SMF.1.1(1) Function 31	Include in ST and select "no other method".	Selected	*"no other method"* selected.
FMT_SMF.1.1(1) Function 32*	Include in ST.	**Not Selected**[12]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 33*	Include in ST. Assign at least USB.	**Not Selected**[13]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 34*	Include in ST. Assign all protocols where the TSF acts as a server.	**Not Selected**[14]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 36	Include in ST.	Selected	*"Enable policy for data-at-rest protection"* selected.

[12] Deviation from Enterprise Device High Security Use Case template.
[13] Deviation from Enterprise Device High Security Use Case template.
[14] Deviation from Enterprise Device High Security Use Case template.

FMT_SMF.1.1(1) Function 37*	Include in ST.	**Not Selected**[15]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 40*	Include in ST.	**Not Selected**[16]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 42*	Include in ST.	**Not Selected**[17]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 47*	Include in ST.	**Not Selected**[18]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 52*	Include in ST.	**Not Selected**[19]	**Not supported by tested Android or iOS device configuration*
FMT_SMF.1.1(1) Function 54*	Include in ST.	Selected	*"Enable/disable policy for the Always-On VPN protection across device"* selected. **Applicable to iOS only*
FMT_SMF.1.1(2) Function c.1	Include in ST.	Exceeded*	*"not accept the certificate"* selected. *This function corresponds to the selection in FIA_X509_EXT.2.2 where the selection has exceeded this requirement.
FMT_SMF.1.1(2) Function c.2	Include in ST.	Selected	*Configure the TOE unlock banner* selected.
FCS_CKM.1.1	Select RSA with key size of 3072 or	Supported	*ECC schemes* selected. **Note:** *RSA with key sizes of 2048-bits or greater* also selected (per MDM PP).

[15] Deviation from Enterprise Device High Security Use Case template.
[16] Deviation from Enterprise Device High Security Use Case template.
[17] Deviation from Enterprise Device High Security Use Case template.
[18] Deviation from Enterprise Device High Security Use Case template.
[19] Deviation from Enterprise Device High Security Use Case template.

	select ECC schemes.		
FCS_CKM.2.1	Select "RSA schemes" or select "ECC schemes".	Supported	*RSA* and *ECC schemes* selected.
FCS_COP.1.1(1)	Select 256 bits	Supported	*256 bits* selected. **Note:** *128 bits* also selected (per MDM PP)
FCS_COP.1.1(2)	Select SHA-384	Supported	*SHA-384* selected. **Note:** *SHA-256* and *SHA-512* also selected (per MDM PP)
FCS_COP.1.1(3)	Select RSA with key size of 3072 or select ECC schemes.	Supported	*"ECDSA schemes"* selected. **Note:** *RSA schemes using key sizes of 2048-bits or greater* also selected (per MDM PP)
FIA_X509_EXT.2.2	Select either "allow the administrator to choose…" or "not accept the certificate".	Supported	*"not accept the certificate"* selected.
FCS_TLSC_EXT.1.1 (from TLS Package)	If included in ST, select "TLS 1.2".	n/a	FCS_TLSC not claimed.
FCS_TLSC_EXT.2.1 (from TLS Package)	If included in ST, select "secp384r1".	n/a	FCS_TLSC not claimed.

Note: This table has been modified by TD0616.

Annex B – Extended Component Definitions

Extended Security Functional Requirements claimed in this ST are provided by the Protection Profiles in which conformance is claimed. For complete definitions of these extended requirements, see their respective PP's as follows:

- Protection Profile for Mobile Device Management Version 4.0, April 25, 2019. (PP_MDM_V4.0)

- PP-Module for MDM Agents Version 1.0, April 25, 2019 (MOD_MDM_AGENT_V1.0)

Annex C – Third-Party Software and Libraries

The Microsoft Intune Company Portal App incorporates material from third parties and includes the following software and libraries:

1. Android v4, v7 and v13 AppCompat Support Library v. 28.0.0 (http://developer.android.com/tools/extras/support-library.html)
2. Android Design Support Library v. 28.0.0 (http://developer.android.com/tools/extras/support-library.html)
3. Platform API v. 27 (http://developer.android.com/sdk/index.html)
4. JSCEP v. 2.5.4 (http://code.google.com/p/jscep)
5. Bouncy Castle v. 1.78 and Bouncy Castle DTLS/TLS API/JSSE Provider v. 1.78 (https://www.bouncycastle.org/)
6. Apache Common Language v. 3.1 (http://commons.apache.org/lang/download_lang.cgi)
7. Sample Code from Android Blog “Some Secure Random Thoughts” 14.08.13 (http://android-developers.blogspot.com/2013/08/some-securerandom-thoughts.html)
8. Dagger (https://github.com/google/dagger)
9. JSR – 330 (http://code.google.com/p/atinject/)
10. Dexmaker v. 1.2 (http://code.google.com/p/dexmaker)
11. AndroidXref-KitKat-textview.java v. 4.4.2 r2 (http://androidxref.com/4.4.2_r2/xref/frameworks/base/core/java/android/widget/TextView.java)
12. AndroidXref-KitKat-edittext.java v. 4.4.2 r2 (http://androidxref.com/4.4.2_r2/xref/frameworks/base/core/java/android/widget/EditText.java)
13. XMLParser.java from the Apache Harmony project (http://harmony.apache.org/)
14. Google-gson v. 2.10.1 (http://code.google.com/p/google-gson/)
15. Android Volley 1.0.15 (https://android.googlesource.com/platform/frameworks/volley)
16. OkHTTP 4.12.0 (http://square.github.io/okhttp/)
17. OkIO 3.9.0 (http://github.com/square/okio)
18. Microsoft-Telemetry-Client-for-Android 2.1.2 (https://github.com/Microsoft/Telemetry-Client-for-Android)
19. Apache Commons Net v. 3.9.0 (http://commons.apache.org/proper/commons-net/)

20. Google libphonenumber v.8.13.16
    (https://github.com/googlei18n/libphonenumber)
21. Google Material Design Icons 3.0.1
    (https://github.com/google/material-design-icons/)
22. Evernote Android-Job 1.2.4
    (https://github.com/evernote/android-job)
23. FasterXml ISO8601Utils
    (https://github.com/FasterXML/jackson-databind)
24. Retrofit 2.9.0
    (https://github.com/square/retrofit)
25. Picasso 2.8
    (https://github.com/square/picasso)
26. RxPreferences 2.0.1
    (https://github.com/f2prateek/rx-preferences)
27. Kotlin 1.9.23
    (https://github.com/JetBrains/kotlin)
28. Nimbus JOSE + JWT 9.37.3
    (https://connect2id.com/products/nimbus-jose-jwt)
29. Editor.java v. 9.0.0_r22
    (https://github.com/aosp-mirror/platform_frameworks_base/blob/master/core/java/android/widget/Editor.java)
30. RxJava v2.2.4
    (https://github.com/reactivex/rxjava)
31. RxPreferences v2.0.1
    (https://github.com/f2prateek/rx-preferences)
32. Zebra MDM Toolkit v2.0.1
    (https://www.zebra.com/us/en/products/software/mobile-computers/mobility-extensions.html)
33. Firebase Android SDK
    (https://github.com/firebase/firebase-android-sdk)
34. AndroidX Support Library
    (https://developer.android.com/jetpack/androidx/)
35. Material Components for Android v1.0.0
    (https://developer.android.com/topic/libraries/support-library)
36. JSON-SMART v. 2.4.7
    (https://github.com/netplex/json-smart-v2)
37. ACCESSORS-SMART v. 2.4.7
    (https://github.com/netplex/json-smart-v2)
38. ASM v. 9.1
    (https://asm.ow2.io/)
39. Moshi 1.15.0
    (https://github.com/square/moshi)
40. httpclient-android v. 4.5.8
    (https://github.com/smarek/httpclient-android)

41. KotlinX Coroutines 1.7.1
    (https://github.com/Kotlin/kotlinx.coroutines)
42. Instacart TrueTime
43. Spotify Mobius
44. SLF4J v2.0.9
    (https://www.slf4j.org/)
45. Apache Commons Lang3 v3.12.0
    (https://commons.apache.org/proper/commons-lang/)
46. Google Tink
    (https://github.com/google/tink)
47. Apache Commons Codec v1.16.0
    (https://commons.apache.org/proper/commons-codec/)
48. Apache Commons IO v2.13.0
    (https://commons.apache.org/proper/commons-io/)
49. JCommander v1.82
    (https://github.com/cbeust/jcommander)
50. SQLite JDBC Driver v3.42.0.0
    (https://github.com/xerial/sqlite-jdbc)
51. OkHTTP Logging Interceptor v4.12.0
    (https://github.com/square/okhttp/tree/master/okhttp-logging-interceptor)
52. Apache HTTP Components v4.5.14
    (https://hc.apache.org/httpcomponents-client-4.5.x/)
53. Google Protocol Buffers v3.24.0|v3.22.3|v3.21.9|v3.17.2|v3.0.1
    (https://github.com/protocolbuffers/protobuf)
54. jsoup v1.16.1
    (https://github.com/jhy/jsoup)
55. Yubikit Android v2.5.0
    (https://github.com/Yubico/yubikit-android)
56. JOSE4j 0.9.6
    (https://bitbucket.org/b_c/jose4j/src/master/)