SAMSUNG

Administrator Guide

Samsung Android 13 on Galaxy Devices

October 16, 2023

Version: 9.0.2

Copyright Notice

About this document

This document describes the enterprise guidance for the deployment of Samsung devices in accordance with the Common Criteria-validated configuration. The document is intended for mobile device administrators deploying Samsung devices.

Document Identification

Document ID	Samsung MDF Admin Guidance v9.0.2
Document Title	Samsung Android 13 on Galaxy Devices Administrator Guide

Contents

1 Introduction

1.1 Scope of Document

This document is intended as a guide for administrators deploying Samsung devices in the enterprise. The guidance provided here focuses on how to configure devices to be in an approved configuration based on the Protection Profile for Mobile Device Fundamentals v3.3 and the PP-Module for Virtual Private Network (VPN) Clients v2.4 for the Samsung devices specified here.

The document is evolutionary. It will cover all devices evaluated with a common major version of Android.

1.1.1 End-User Guidance

This guidance document is focused on the central management of Samsung mobile devices. Guidance related to user functions on a device, such as managing Bluetooth connections or setting authentication credentials are outside the scope of this documentation. End-user guidance can be found both on the device (most functions are guided through the user interface with descriptions and help) or from the Samsung support website. Links to online guidance can be found in section 1.5 References.

1.2 Overview of Document

Samsung mobile devices are designed to maintain a secure mobile environment. To successfully deploy and maintain such an environment requires coordination with multiple parties including:

- Enterprise/Mobile Device Management (EDM/MDM) software
- Carriers
- Mobile Device Administrators
- Users

This document is designed for the Mobile Device Administrators, to provide guidance in how to configure and deploy Samsung mobile devices within an enterprise environment. This includes information about API controls that can be used within the EDM/MDM software to achieve this configuration.

1.3 Terminology & Glossary

Evaluated Device	Processor
ADB	Android Debug Tool
ADT	Android Development Tools
API	Application Programming Interface
BYOD	Bring Your Own Device

Evaluated Device	Processor
CA	Certificate Authority
COPE	Corporately-Owned, Personally Enabled
EDM MDM	Enterprise Device Management Mobile Device Management NOTE: EDM will be used for consistency
FBE	File-Based Encryption
FOTA	Firmware Over-the-Air
KPE	Knox Platform for Enterprise
MDF MDFPP	Mobile Device Fundamentals Mobile Device Fundamentals Protection Profile
SDK	Software Development Kit
SoC	System on a Chip
TLS	Transport Layer Security
VPN	Virtual Private Network

Table 1 - Acronyms

1.4 Evaluated Devices

The Common Criteria evaluation was performed on a set of devices covering a range of processors. These devices were chosen based on the commonality of their hardware across several different devices that are also claimed through equivalency. All device models are evaluated with Samsung Android 13.

The evaluation was performed on the following devices (note that the evaluation period is listed in parenthesis for each device):

Device Name	Chipset Vendor	SoC	Arch	Kernel	Build Number
Galaxy Z Fold5 5G	Qualcomm	Snapdragon 8 Gen 2 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy Z Flip5 5G	Qualcomm	Snapdragon 8 Gen 2 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy Tab S9 Ultra	Qualcomm	Snapdragon 8 Gen 2 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy Tab S9+	Qualcomm	Snapdragon 8 Gen 2 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy Tab S9	Qualcomm	Snapdragon 8 Gen 2 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy A52 5G	Qualcomm	Snapdragon 750G (SM7225)	ARMv8	4.19	TP1A.220624.014
Galaxy A42 5G	Qualcomm	Snapdragon 750G (SM7225)	ARMv8	4.19	TP1A.220624.014
Galaxy A71 5G	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	ARMv8	4.14	TP1A.220624.014
Galaxy A51 5G	Qualcomm	Snapdragon 765G (SM7250)	ARMv8	4.14	TP1A.220624.014
Galaxy Tab Active3	Samsung	Exynos 9810	ARMv8	5.1	SP1A.210812.016
Galaxy S23 FE	Qualcomm	Snapdragon 8 Gen 1 Mobile Platform	ARMv8	5.15	TP1A.220624.014
Galaxy S23 FE	Samsung	Exynos 2200	ARMv8	5.4	SP1A.210812.016

Table 2 - Evaluated Devices

1.4.1 Device Equivalency Claims

Many Samsung devices share common capabilities in different form factors, and Samsung provides common capabilities, including support for the configurations necessary for the evaluation on these devices. The following table shows the devices for which equivalence is being claimed from a device that is explicitly evaluated.

Evaluated Device	SoC	Equivalent Devices	Differences
Galaxy Z Flip5 5G	Snapdragon 8 Gen 2	Galaxy Z Fold5 5G	Z Fold5 > Z Flip5 in terms of display size
		Galaxy Tab S9 Ultra	Tab S9 devices are tablets (no voice calling) with S Pen
		Galaxy Tab S9+	Tab S9 Ultra > Tab S9+ > Tab S9 in terms of display size
		Galaxy Tab S9	All Tab S9 has under screen image fingerprint sensor
Galaxy A52 5G	Snapdragon 750	Galaxy A42 5G	A52 5G > A42 5G screen resolution & RAM
Galaxy A71 5G	Snapdragon 765	Galaxy A51 5G	A71 5G > A51 5G in terms of display size
Galaxy S23 FE	Exynos 2200	N/A	
Galaxy S23 FE	Snapdragon 8 Gen 1 (SM8475)	N/A	
Galaxy Tab Active3	Exynos 9810	N/A	T577 & T575 tablets have 5G T570 tablets only have Wi-Fi

Table 3 - Device Equivalence

The differences between the evaluated devices and the equivalent ones do not relate to security claims in the evaluated configuration. The Wi-Fi chipsets are the same for each series of common devices.

1.4.2 Device Details

The model numbers and evaluated versions of the mobile devices being claimed are as follows:

Device Name	Chipset Vendor	Chipset Name	Base Model Number	Carrier Models
Galaxy Z Fold5 5G	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	SM-F946	W, B, N, U1, U
Galaxy Z Flip5 5G	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	SM-F731	W, B, C, N, U1, U
Galaxy A51 5G	Qualcomm	Snapdragon 765G (SM7250)	SM-A516	V, SC54A*, SCG07*,
Galaxy A71 5G	Qualcomm	Snapdragon 765G (SM7250)	SM-A716	U1, U, V
Galaxy A52 5G	Qualcomm	Snapdragon 750G (SM7225)	SM-A526	W, B, U1, U, SC-53B
Galaxy Tab S9 Ultra	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	SM-X916 SM-X900	B, N None
Galaxy Tab S9+	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	SM-X818 SM-X816 SM-X810	U B,N,E None
Galaxy Tab S9	Qualcomm	Snapdragon 8 Gen 2 (SM8550)	SM-X716 SM-X710	B, N None
Galaxy Tab Active3	Samsung	Exynos 9810	SM-T577 SM-T575 SM-T570	U N,None None
Galaxy A42 5G	Qualcomm	Snapdragon 750G (SM7225)	SM-A426 SM-S426	B, N, U1, U D, L
Galaxy S23 FE	Samsung	Exynos 2200 Snapdragon 8 Gen 1 (SM8450)	SM-S711	B W, U1, U

Table 4 – Device Details

The Carrier Models column specifies the specific versions of the devices that have the validated configuration. These additional letters/numbers denote carrier specific models (such as U = US Carrier unified build). Only models with the suffixes listed in the table can be placed into the validated configuration. The carrier models marked by * are explicit model numbers for those carriers and do not follow the standard specified for other models.

The following table shows the Security software versions for each device.

Device Name	MDF v3.3 Release	BT v1.0 Release	WLAN v1.0 Release	VPN PP-MOD v2.4 Release	Knox Release
All devices	1	1	2	1.1	3.9

Table 5 - Security Software Versions

The version number is broken into two parts showing the Protection Profile or Extended Package version as well as the software version that is certified. For example, the Galaxy S23 FE would show "MDF v3.3 Release 1".

1.5 References

The following websites provide up to date information about Samsung device certifications.

Site	Information	URL
Samsung Knox Portal	Common Criteria documentation, Application Version List, Tools	https://docs.samsungknox.com/admin/knox-platform-for-enterprise/kbas/common-criteria-mode.htm
Samsung Knox SDK	Samsung Knox developer guides including EDM APIs	https://docs.samsungknox.com/dev/knox-sdk/index.htm
Galaxy S Device Support	Manuals & User Guides for Galaxy S devices	https://www.samsung.com/us/support/mobile/phones/galaxy-s
Galaxy Note Device Support	Manuals & User Guides for Galaxy Note devices	https://www.samsung.com/us/support/mobile/phones/galaxy-note
Galaxy Tablet Device Support	Manuals & User Guides for Galaxy Tab devices	https://www.samsung.com/us/support/mobile/tablets/galaxy-tabs
NIAP	Product Compliant List for Samsung Electronics	https://www.niap-ccevs.org/Product/PCL.cfm?par303=Samsung%20Electronics%20Co%2E%2C%20Ltd%2E
	Approved Protection Profiles	https://www.niap-ccevs.org/Profile/PP.cfm
NIST CMVP	Validated Cryptographic Modules	https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program/Validated-Modules/Search
NIST CAVP	Validated Cryptographic Algorithms	https://csrc.nist.gov/projects/cryptographic-algorithm-validation-program
NIST SP 800-63B	NIST SP 800-63B Digital Identity Guidelines	https://pages.nist.gov/800-63-3/sp800-63b.html

Table 6 – Reference Websites

2 Mobile Device Deployment

2.1 Device Overview

The mobile device is a combination of hardware running an Android operating system with modifications made to increase the level of security provided to end users and enterprises. The mobile device is intended for use as part of an enterprise messaging solution providing mobile staff with enterprise connectivity. With a focus on enterprise security, the mobile device also provides support for IKEv2 VPN tunnels using certificates, providing flexibility based on the environment.

The mobile device combines with an EDM solution to enable the enterprise to watch, control and administer all deployed mobile devices, across multiple service providers as well as facilitate secure communications through a VPN tunnel. This partnership provides a secure mobile environment that can be managed and controlled by the environment and reduce the risks that can be introduced when enabling mobility in the enterprise, whether through a Bring-Your-Own-Device (BYOD) or a Corporate-Owned deployment.

The Samsung Software Development Kit (SDK) builds on top of the existing Android security model by expanding the current set of security configuration of options to over 600 configurable policies and including additional security functionality such as application blacklisting. The ability to set these policies is based on the capabilities of the EDM.

2.2 Evaluated Device Capabilities

The product provides a significant amount of security capabilities with the core capabilities being included within the common criteria evaluation including:

Security Feature	Description
Device data protection. The mobile device provides security functionality to protect data at rest.	**File-Based Encryption (FBE).** The mobile device automatically encrypts data on the internal flash media of the device using AES 256.
	Removable storage encryption. The mobile device can encrypt all files placed onto, or already residing on, removable storage attached to the device (not all devices support removable media).
	Sensitive data protection. The mobile device has the ability to securely store incoming data that is considered sensitive such that it can't be decrypted without the user logging in.
Application Management. The device provides a number of security functions to manage device software.	**Application resource restrictions.** All applications are run within a controlled environment that limits applications to only accessing only authorized data and resources.
Access Control. The device can implement access control that reduces mobile user permissions and assists	**Device lock.** The mobile device can be configured to lock automatically after a defined period of inactivity (1 to 60 minutes) limiting access to device functions accept those that are explicitly authorized such as emergency calls.
	Local wipe. The mobile device has the ability to wipe encryption keys/data on a device after a defined number of authentication attempts are surpassed.

Security Feature	Description
in reducing unauthorized access.	**Credential complexity.** The mobile device can enforce enterprise password policies forcing users to use a defined level of complexity in device passwords.
	Biometrics Use. The mobile device can provide biometric authentication for access to the device complementary to password policies, restricting access based on failed attempts.
	Privileged access. The mobile device can be configured to restrict mobile user's access to privileged functions such as device configurations.
	Hotspot Control. The mobile device can be configured to act as a hotspot for sharing Internet access to other devices.
	Wireless network settings. The wireless network configuration of the mobile device can be specified, providing requirements or pre-loaded networks.
Enterprise device management. Enterprise administrators can control and audit mobile endpoint configurations and wipe device if needed.	**Remote wipe.** An enterprise administrator can send a message to the mobile device to wipe all local storage and the SD card.
	Security policy. The mobile device and VPN can be configured by an EDM solution that supports the Samsung Enterprise SDK.
	Auditing. The mobile device can monitor and generate records related to security-relevant events within the device.
Secure Channel. Enterprise devices can securely connect to the enterprise network.	**VPN.** The mobile device provides a secure communications channel to the VPN Gateway.

Table 7 – Mobile Device Security Features

2.3 Deployment Architecture

The first step in deploying Samsung devices is to decide on both an EDM solution and an appropriate architecture. These selections are beyond the scope of this guidance. There are many approaches to how the management infrastructure can be configured, from on premise servers to cloud to hybrid approaches combining the two. The specifics of the architecture should be discussed with the EDM solution vendor.

Ideally, the deployed EDM solution should be evaluated to the requirements of the Protection Profile for Mobile Device Management (MDMPP).

2.3.1 Deployment Environment

The enterprise environment must provide all of the services required to operate and manage devices. The basic components of this model include:

Component	Description
Enterprise/Mobile Device Management Solution	The EDM Solution secures monitors, manages and supports mobile devices deployed across the organization. Controlling and protecting the data and configuration settings for all mobile devices in the network reduces security risks. As part of the EDM solution, an app (usually called an Agent) is installed onto the mobile device. This Agent implements the policies from the EDM and can communicate back to the server, sending status information and logs for review.

Component	Description
Secure Tunnel Termination	A secure VPN tunnel should be initialized between the managed Android devices and the Enterprise Environment to prevent unauthorized access to enterprise resources. The connection should be based on certificates deployed on the Android user devices. Ideally, mutual authentication is deployed, meaning that both the Android user devices authenticate themselves with a certificate but also the gateway to the enterprise environment. Mutual authentication serves to prevent Android user devices to login into an unauthorized enterprise network and on the other hand prevents the unauthorized login of untrusted devices into the enterprise environment. For services that do not require a VPN, TLS should always be used to encrypt access to the site. Similar to the VPN, mutual authentication between the client and server is recommended. Note that EDM access to the between the device and server does not need to be through a VPN but is expected to have its own secure channel for communications.
Directory Services	The directory services should be set up to store, organize and provide access to information in a directory.
Business Applications	Business applications allow enterprise users to fulfill or access certain business tasks pertinent to requirements. This may include management tools, accounting utilities and contact management software/solutions.
Certificate Services	Certificate services must be implemented to manage all certificate needs throughout the enterprise environment. This includes issuing new Android device user certificates that are needed to facilitate secure communications through a VPN or TLS connection. It is possible that the certificate services could be provided by a third party instead of a stand-alone internal service for the organization.

Table 8 – Enterprise Deployment Component Services

Figure 1 shows an example of a high-level design of an enterprise-based environment.

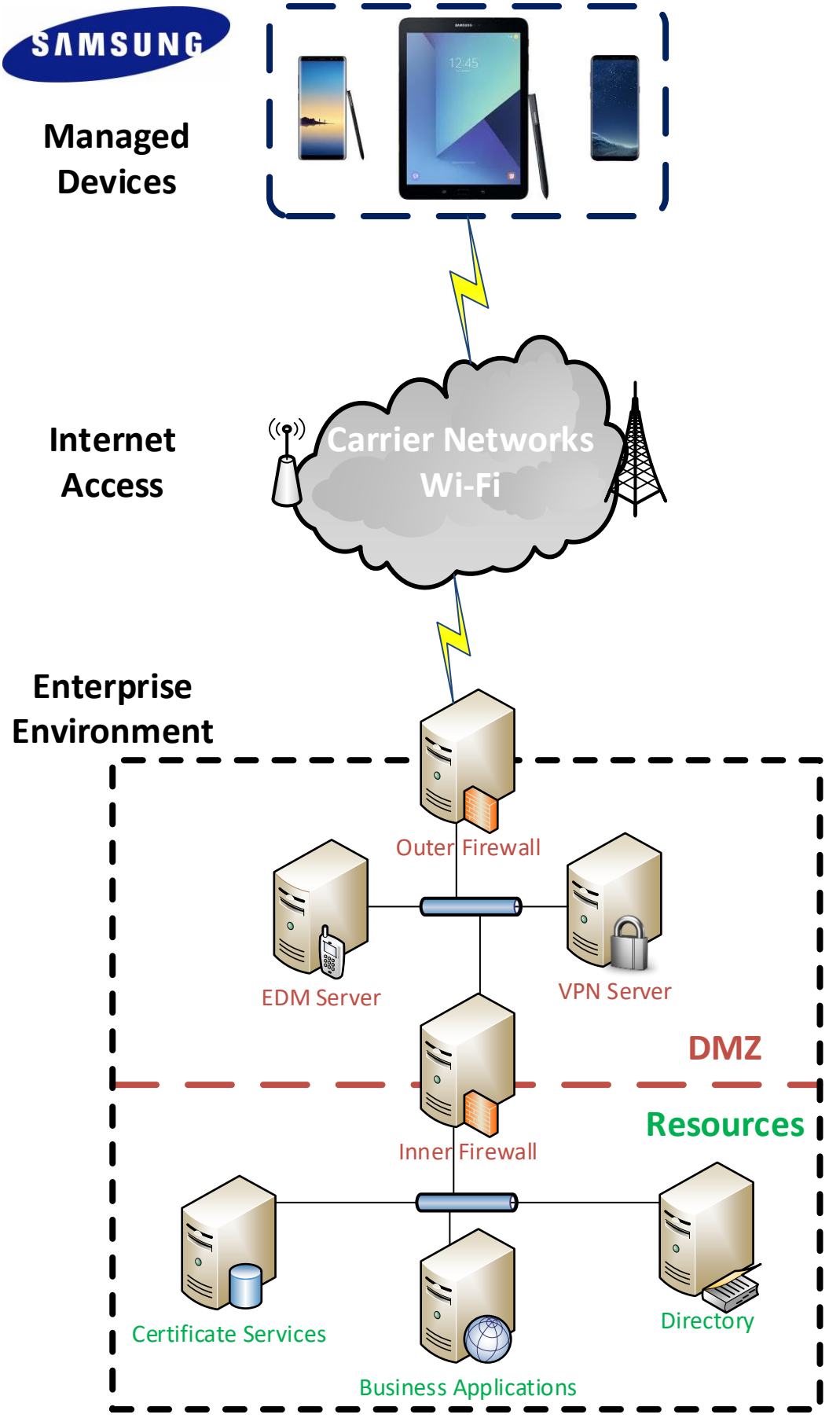


Figure 1 - Example Enterprise Architecture

2.3.2 EDM Solution Selection

To manage the mobile devices, an EDM must be deployed. This EDM should support the Samsung Knox APIs to enable the capabilities documented in this guide. The more complete the EDM vendor support, the more capabilities can be controlled on the device.

To enable capabilities such as remote wipe of a device, the EDM must be placed within the Enterprise environment such that it can communicate over the internet. This communication can be enabled with or without a VPN, though it is normally recommended to have EDM traffic outside the VPN to ensure access is most widely available to the device. If operating outside the VPN, it is recommended to ensure the MDM solution is configured to use another security protocol (e.g., TLS) when communicating with the device.

When selecting an EDM solution, care should be taken to ensure the ability to configure the Common Criteria configuration. The Common Criteria Configuration section provides the specific information about the Knox APIs that are necessary to support this configuration and can be used to check the compatibility of the EDM solution with the needs of the Enterprise.

2.4 Provisioning of Samsung Devices

As noted above, the secure deployment of enterprise devices is reliant on many components beyond the mobile device itself. It is expected that within the Enterprise environment the EDM solution and other required services are securely installed and configured according to the security requirements of the organization.

Once the EDM is installed and available, it is possible to begin provisioning end user devices. The provisioning process will prepare the devices for a policy configuration to be deployed, enabling the device to be placed into a Common Criteria configuration.

The mobile device must be enrolled with the EDM server to enable administration via the EDM. Enrollment is accomplished by installing the EDM Agent application onto the device. There many methods and configurations for doing this depending on the deployment scenario. The EDM documentation for deployment should be followed.

Once a device has been enrolled to the EDM, other optional configurations may be set, depending on the organization security policy. These are not required to place the device into a Common Criteria configuration, but are best practices for mobile devices.

NOTE: Configurations that are included as part of the controls for the Common Criteria configuration are not included here.

The following list provides some of the most common additional configuration items that may be done on a mobile device:

- Install applications required for enterprise productivity
- Provision client certificates by either:
    - Using the EDM server;
    - Using the Android Development Tools (ADT) to manually push certificates to each device via USB

        - Using the Android Debug Tool (ADB) required USB debugging to be enabled on the device for provisioning of the certificates (it can be disabled once this operation is complete)
    - Placing the certificates on a microSD card and import using the device user interface
    - The certificates commonly deployed are:
        - Enterprise CA certificate (used to validate the server certificates presented by the VPN endpoint and reverse proxy)
        - Wi-Fi client certificate (for authentication to an EAP-TLS Wi-Fi AP)
        - VPN client certificate (for authentication to the enterprise VPN endpoint)
        - SSL client certificate (for authentication to the reverse proxy for intranet services)
- Configure the VPN client to connect to the enterprise VPN endpoint
    - Enable 'Always-On' VPN
- Configure the email client to connect to the enterprise server

2.4.1 Work Profile Configurations

Through the Knox Platform, Samsung provides multiple device configurations to support different levels of application segmentation. The configurations available are dependent on how the EDM is installed on the device. A Samsung device can be placed into an evaluated configuration both with and without a work profile being configured on the device.

A Knox work profile provides a complete separate environment on a device that is fully managed by the EDM. A work profile can be used to enable some level of personal use of the device while allowing the enterprise to manage the separation of data (what can or cannot be shared, the visibility of notifications from the work profile outside, etc.), or to provide additional security for certain apps and data on the device. In this configuration, access to the work profile requires two separate authentications, one to the device and a second to the work profile.

2.5 Lock Screen

2.5.1 Biometric Authentication

Biometric authentication using fingerprints is a ubiquitous feature available across all devices, albeit with varying sensor positions. These positions primarily include the side button type and the under-display type. In the side button position, the fingerprint sensor integrates with the power button, while the sensor sits below the display panel in the under-display type. Although the precise physical location of the sensor may vary, user interactions with the system remain consistent.

A device can accommodate up to 20 distinct fingerprints, but a software mechanism verifies the image's quality and information during each trial. This verification process has a strong reliance on the sensor's area, other hardware features, and software algorithms. To enroll a fingerprint, the user must first establish a password for the device. Once set, users can navigate to the device's settings to manage fingerprints, add new fingerprints, or delete existing ones. However, to perform these actions, users must enter their password.

When enrolling a fingerprint, the device provides on-screen guidance to capture quality data, asking users to

move their finger to cover a wider area. To acquire optimal fingerprint samples, the device prompts users to ensure they maintain proper finger placement on the sensor, providing multiple user-friendly messages until quality data acquisition is complete.

It is important to note that biometric authentication cannot be used immediately following a power event, such as a power-on or a reboot. Users must first authenticate using their password before gaining access to biometric authentication. Once authenticated, users can utilize their fingerprints for up to 72 hours without requiring a device reboot.

3 Common Criteria Configuration

This section of the guide will list the configuration settings that are reviewed as part of the Common Criteria evaluation. Some of these settings are required for the device to be placed into a validated configuration while others are optional and can be used at the discretion of the organization and the attendant security policies.

3.1 Approved Cryptography

Part of the Common Criteria-evaluated configuration is the availability of approved cryptographic engines for use by the system and applications. Samsung has chosen to utilize NIST-validated cryptographic algorithms within the cryptographic modules on its devices for the Common Criteria configuration. These algorithms are made available for use by applications installed on the device through the normal Android Framework APIs.

Samsung provides the following cryptographic modules with NIST-validated algorithms on all the evaluated devices:

- Samsung BoringSSL Cryptographic Module
- Samsung Kernel Cryptographic Module
- Samsung SCrypto Cryptographic Module

In addition, the following cryptographic modules with NIST-validated algorithms are available, depending on the CPU:

- Samsung Flash Memory Protector (on devices with Samsung Exynos processors)
- QTI Inline Crypto Engine (on devices with Qualcomm Snapdragon processors)

All modules always run in a FIPS-validated mode. BoringSSL, for compatibility reasons, provides access to non-FIPS algorithms. Developers should not utilize non-FIPS algorithms in a validated configuration (but these are necessary to ensure functionality with many commercial services). Samsung integrates the cryptographic modules directly into Android so they can be accessed by any app using the native Android APIs. The APIs providing access to FIPS-validated algorithms are detailed in the section 6 Developer References. Wi-Fi connections can utilize both FIPS and non-FIPS algorithms for compatibility reasons. To ensure the use of FIPS-validated algorithms in Wi-Fi connections the Wi-Fi Access Point should be configured to specify the proper ciphersuites.

Note: It is possible that some applications will implement their own cryptography instead of relying on the modules provided with the device. It is the responsibility of those vendors to validate their own cryptography. Samsung recommends that developers utilize the cryptographic functions provided with the device using the native Android APIs.

3.2 Enabling CC Mode

The Samsung devices listed in this document support a Common Criteria (CC) Mode. This CC Mode provides feedback on whether or not the device meets the minimum required configuration according to the MDF requirements.

While there are two methods for enabling CC Mode on a device, only the EDM-managed method will be explained here.

NOTE: The CC Mode app is for testing and not intended as a deployment tool.

3.2.1 CC Mode Status

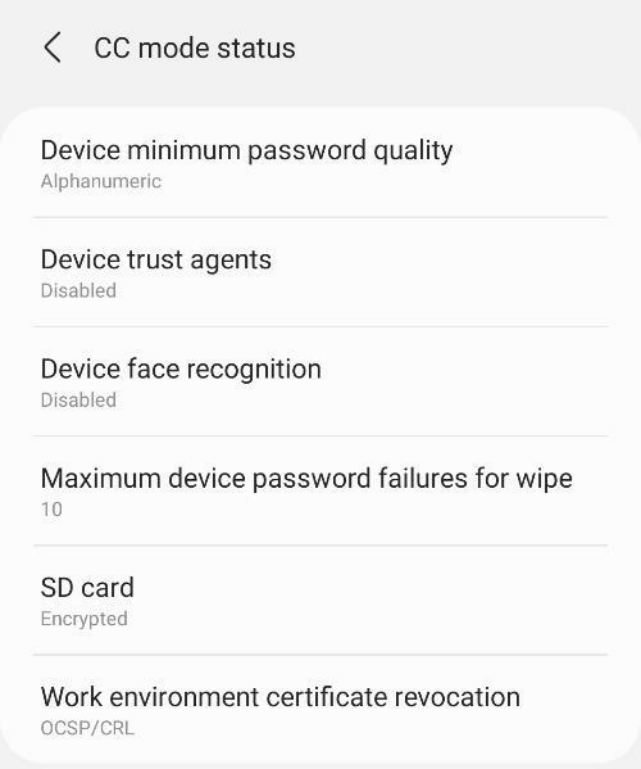
CC Mode has two possible states:

Status	Description
Enabled	CC Mode has been turned on
Disabled	CC Mode has been turned on but an integrity check or self-test has failed (such as a FIPS 140-2 self-test)

Table 9 – CC Mode Status

The status of the CC Mode check is entered into the audit log through a series of entries about each of the conditions necessary for CC Mode.

The CC Mode status can be seen by a user in ***Settings/About phone/Software Security Version*** under ***CC mode status***. This will only appear when CC Mode has been enabled on the device. Tapping on the status will show the current settings related to meeting the evaluated configuration, the same settings recorded in the audit log.


CC mode status
Device minimum password quality
Alphanumeric
Device trust agents
Disabled
Device face recognition
Disabled
Maximum device password failures for wipe
10
SD card
Encrypted
Work environment certificate revocation
OCSP/CRL

The following settings are listed:

Setting	Possible Values	Description
Device minimum password quality	Unspecified Something Numeric Numeric Complex Alphabetic Alphanumeric Complex Biometric	The specified level of the password requirements (recommended minimum is alphanumeric)
Device trust agents	Enabled/Disabled	Status of Smart Lock function
Device face recognition	Enabled/Disabled	Status of face biometric
Maximum device password failures for wipe	30 or less	The number of failed authentication attempts before the device will be wiped
SDCard Encryption	Encrypted/Decrypted	Status of whether SD Card encryption is set
Work environment certificate revocation	None/CRL/OCSP	None = no revocation checking CRL = CRL checking enabled OCSP/CRL = OCSP as primary and CRL as secondary is enabled Work environment – this means that the setting is applied to the work profile

Table 10 - CC mode status Page

Note: It is unlikely a user will see the Disabled state as the failures necessary to meet this condition are such that the device is unlikely to boot.

3.3 Device Common Criteria Settings

This section will lay out all the settings which have been reviewed as part of the MDF-validated configuration. While there are mandatory settings, there are also many optional settings that can be configured to the needs of the enterprise.

There are two possible configurations a device can be placed in.

Configuration	Description
Managed	This is a company-owned, fully managed device
WP-C (Work Profile – Company Owned)	This is a company-owned device with work profile providing for separation between personal and work apps

Table 11 – Device Configurations

The WP-C configuration supports work profiles, while the managed configuration does not. Several of the settings may be applied to either the device, the work profile or both, depending on the scope of the setting (global to the device, or what is called user, which can be the device or a work profile). Each setting is marked under each configuration to show where the setting can be applied.

The settings used for the configuration are made up of settings from both the Google Android Enterprise APIs and the Samsung Knox Platform for Enterprise APIs. All Samsung Knox APIs specified are part of the Knox Platform for Enterprise (KPE) set of APIs and require a Knox Platform for Enterprise license to be used.

The settings have also been marked as mandatory or objective (or in the case of CC Mode, Always).

All the settings are included in the attached spreadsheet on the Device Settings worksheet.

Settings Table.xlsx

3.3.1 Common Criteria Minimal Configuration

To configure the device into the minimal evaluated configuration, all settings marked as Always and Mandatory must be set. Once these have been set, the device configuration can be verified by reviewing the audit records from the device boot.

The optional configuration settings can be used to meet the deployment needs of the organization. These settings have been covered in the evaluation, but the specific settings of those items does not affect the evaluated configuration.

The following settings must be configured via the EDM after CC Mode has been enabled:

1. Set Password Quality
2. Enable the Maximum Password Failure Wipe Policy
3. Disable Smart Lock
4. Enable SD Card Encryption
5. Disable Debugging Features (Developer options)

NOTE: SD Card encryption is only needed on devices that support inserting removable media.

6. Enable Revocation Checking

NOTE: The administrator can choose either CRL or OCSP revocation checking.

If biometrics are enabled, the following setting must be configured:

1. Disable Face Lock

To ensure overall control of the Common Criteria configuration, CC Mode cannot be disabled by an end user except by performing a factory reset. It is possible to change the CC Mode status through the EDM.

3.3.1.1 Application Allow/Block Listing Settings

Allow/Block listing is done using the full name of the application (such as com.android.testingapp).

The application removal process will automatically clear data associated with the application stored in the application directories. Data created or stored outside the application directories (such as photos by a camera application or documents created by a word processor) will not be removed when the application is uninstalled.

The method for configuring these lists is highly dependent on the EDM solution chosen. Please refer to the EDM specific guidance on exactly how to set these policies.

Note: The Application Allow/Block lists will not have any impact on apps that are part of the system image. Built-in apps can instead be Disabled.

3.4 VPN Client Settings

The device also includes an evaluated VPN client. There are two ways to configure the built-in VPN client, depending on the needs of the organization, via the Standard APIs or via the Knox Generic VPN APIs.

The Standard APIs provide a basic set of functionality for a VPN client that is configured for the entire device (all traffic would pass through this VPN profile).

The Knox Generic VPN APIs provide a highly flexible method for configuring VPNs that can include the ability to control access to applications or groups of applications to specific tunnels. The Knox VPN framework can be used to control tunnels both inside and outside the work profile, depending on where the VPN client is installed (inside or outside the work profile). The Knox VPN framework can be used with the built-in Samsung VPN client or with third-party VPN client vendors, depending on the needs of the organization.

The settings for configuring a VPN client profile can be found in the Settings Table spreadsheet (see Device Common Criteria Settings) on the VPN Settings worksheet. The specific settings can be used for profiles that are compliant with the Common Criteria configuration.

3.4.1 VPN Profile Settings (All)

3.4.1.1 Valid Certificate Types for IKEv2

While the menu selection for the type of tunnel states IPsec IKEv2 RSA it is possible to utilize both RSA and ECDSA certificates for the tunnel. As long as the certificates are valid (not expired, properly formatted, etc.) they can be used for the VPN configuration.

3.4.2 VPN Profile Settings (Standard APIs)

3.4.2.1 Server Certificate for the Gateway

It is possible to specify a Server Certificate for the Gateway in the configuration of a VPN tunnel. This certificate will override any certificate provided by the Gateway during the negotiation of the tunnel.

This certificate may be loaded through the UI or EDM. See the device User Guidance for more information about loading certificates manually.

3.4.3 Knox VPN Profile Settings (Knox Generic APIs)

Configuring the VPN via Knox Generic VPN APIs has the benefit of allowing per-app routing to the VPN client. For example, all container packages can be forced to go through one tunnel, while personal applications are routed through another, or not at all.

The Knox VPN framework can be used with the built-in Samsung VPN client or with third-party VPN client vendors, depending on the needs of the organization.

To use the Knox VPN framework, the following is needed:

Setting	Value	Description
VPN Installer(s)	APKs from vendor	Installation package(s) from the VPN client vendor for installation on the device. Generally (though not always) this would include 2 files.
VPN profile(s)	JSON files	The VPN profile(s) to be deployed on the device
"vpn" folder	JSON files and vendor.ini	The full set of configurations (including Knox configuration) needed for deployment of the VPN profile

Table 12 – Knox VPN Framework Components

The VPN client vendor would provide the files above though the JSON configuration would have to be edited by the Administrator. More information about the JSON configuration can be found here: https://docs.samsungknox.com/dev/knox-sdk/VPN-json.htm.

3.4.3.1 Samsung VPN Client Configuration for Knox VPN Profile

Using Knox Generic APIs requires installation of the Samsung Proxy APK on the device, which translates configuration received through these APIs onto the underlying Samsung VPN client. The use of other Proxy APKs could be used to support non-Samsung VPN clients (that is not covered here).

Note: Using the Samsung VPN APK will configure the Knox VPN Profile to point to the evaluated VPN client.

Provided the profile configuration string has been created as per the next section, the API flow for creating and starting a VPN connection will be createVpnProfile() -> addPackagesToVpn() -> activateVpnProfile() API.

The API flow for removing a VPN profile will be activateVpnProfile() (De-activate it) -> removeVpnProfile() API.

Note: When adding packages to a VPN profile, use User0 for the whole device and User10 or User100 (depending on the device) for the work profile.

3.4.3.2 JSON Configuration String

This is an example JSON file for the Knox VPN Client Profile.

```
{
  "KNOX_VPN_PARAMETERS": {
    "profile_attribute": {
      "profileName": "ss1",
      "host": "",
      "isUserAuthEnabled": true,
      "vpn_type": "ipsec",
      "vpn_route_type": 1
    },
    "knox": {
      "connectionType": "keepon",
      "chaining_enabled": "-1",
      "uidpid_search_enabled": "0"
    },
    "vendor": {
      "basic": {
       "autoretry": "1",
        "username": "sampleu",
        "password": "samplepw",
        "authentication_type": "type",
        "host": "111.111.111.111"
      },
      "ipsec_xauth_psk": {
        "identifier": "test@sta.com",
        "pre_shared_key": "example",
        "dns_search_domains": [],
        "dns_servers": [
          "8.8.8.8"
        ],
        "frwd_routes": [
          "10.0.0.0\/8"
        ]
      },
      "ipsec_xauth_rsa": {
        "user_cert_alias": "",
        "ca_cert_alias": "",
        "server_cert_alias": "",
        "dns_search_domains": [],
        "dns_servers": [
          "8.8.8.8"
        ],
        "frwd_routes": [
          "10.0.0.0\/8"
        ]
      },
      "ipsec_ike2_psk": {
        "identifier": "test@sta.com",
        "pre_shared_key": "example",
        "dns_search_domains": [],
        "dns_servers": [
          "8.8.8.8"
        ],
        "frwd_routes": [
          "10.0.0.0\/8"
        ]
      },
      "ipsec_ike2_rsa": {
        "user_cert_alias": "",
        "ca_cert_alias": "",
        "server_cert_alias": "",
        "dns_search_domains": [],
        "dns_servers": [
          "8.8.8.8"
        ],
        "frwd_routes": [
          "10.0.0.0\/8"
        ],
        "ocsp_url": ""
      }
    }
  }
}
```


Example Xauth-PSK JSON

The JSON has 3 sections:

1. The profile_attribute section

   This section of the JSON message contains profile details.

- profileName: A string value to name a profile configuration.
- host: A string value representing the VPN server in IP format or domain name format.
- vpn_type: A string value representing the VPN connection type. For example, "ipsec" or "ssl" or "$random-string"
- vpn_route_type: An integer value, 1 or 0, to determine whether or not the message defines a Knox profile; 1 confirms

2. The knox section

   This section covers the Knox settings that are enabled for the given profile. The options configured in the Knox section of the JSON message, except for chaining, are applied to all apps added to the profile.

- uidpid_search_enabled: To enable (1), to disable (0)

- chaining_enabled: To determine if encryption is chained (1) or non-chained (0). The default value is -1;
- ConnectionType: A string value to find the connection type:
    - keepon: starting or stopping a VPN connection does not depend on starting/stopping apps. This is the default value.
    - ondemand: starting or stopping a VPN connection depends on starting/stopping apps.
- proxy-server: A string value to represent the static proxy server in either IP address or domain name format.
- proxy-port: A string value to represent the static proxy server port number.
- pac-url: A string value to represent the PAC url in either IP address or domain name format.
- proxy-auth: An integer value to represent the authentication needed for the proxy server referenced in the PAC URL.
    - 0: Default value, no authentication needed
    - 1: Basic authentication needed
    - 3: NTLMV2 authentication needed
- proxy-username: A string value indicating that the admin provided credentials for the proxy-server authentication for both static, and PAC, proxy support.
- proxy-password: A string value indicating that the admin provided credentials for the proxy-server authentication for both static, and PAC, proxy support.

3. The vendor section

   This section of the JSON message contains vendor-specific details related to VPN clients

3.4.3.3 Server Certificate for the Gateway

It is possible to specify a Server Certificate for the Gateway in the configuration of a VPN tunnel, by providing the server_cert_alias string corresponding to a certificate previously installed into the keystore. This certificate will override any certificate provided by the Gateway during the negotiation of the tunnel.

This certificate may be loaded through the UI or EDM. See the device User Guidance for more information about loading certificates manually.

3.4.4 VPN Gateway Configuration Control

There are many configuration options for a VPN tunnel that only be configured from the gateway. The VPN client will utilize these settings from the gateway configuration to construct the secure tunnel. The following is a list of the settings that must be configured through the gateway:

- Encryption settings – while the VPN client will use FIPS validated encryption, the gateway will specify which algorithms should be used.
- IKE Protocols & Authentication – the gateway specifies which IKE protocols authentication techniques are required for establishing the connection.
- IPsec Session Key cryptoperiod – the gateway specifies the session key cryptoperiod and can be used to configure periods under 1 hour in duration.

3.4.5 Third-Party VPN Clients (Device)

While Samsung devices come with a Common Criteria-certified VPN client, Enterprise customers may also use a VPN client from a third party vendor. Android provides the public class android.net.VpnService for third party vendors to build VPN clients that can be installed within Android.

These clients may contain additional capabilities beyond those provided by the built-in Android or Samsung clients. VPN client software built using this interface may provide their own management interface outside of that provided by Samsung.

3.5 Additional Common Criteria Features

3.5.1 Sensitive Data Protection

Depending on the model, there are two different ways to protect sensitive data. The S23 and its equivalent devices use the NIAPSEC library to encrypt and decrypt files, for all the other devices Samsung has added capabilities for Sensitive Data Protection using the Knox Platform API's. This feature is designed to allow applications that run in the background and receive information to protect that information upon receipt. This feature is provided as part of the device, but its use is dependent on applications having been written to the APIs providing the capability. It is expected that this list will grow over time, but is currently limited to the Samsung Email application contained within the work profile.

The API for Sensitive Data Protection exists for different Knox Platform configurations, but unless an application has been written to the API, it will not take advantage of the Sensitive Data Protection function.

3.5.2 Background Network Communications

Samsung Android devices are usually configured by default to send anonymous usage data (including location, device ID etc.) to Google and Samsung servers. This can be disabled through device settings and will need to be enforced through procedural controls.

Samsung Android devices do not need to be associated with a Google account to operate as required within the enterprise. For example, it is still possible to receive push notifications through Google Cloud Messaging. Knox EDM APIs can be used to prevent users from signing in to these services (see EDM guidance).

3.5.3 Knox Separated Apps

Knox Separated Apps is focused solely on providing a highly segmented application group with no ability to share anything across the group boundary. Instead of a full environment (as in a work profile), Knox Separated Apps provides the administrator with the ability to allow or block apps from being placed into the application group, and only these apps will be available. Knox Separated Apps does not require any separate authentication, providing seamless access to the isolated apps to the end user.

4 End User Procedures

While the administrator can configure the device, the end user of the device will interact with the resulting configuration. Specific instructions about procedures for an end user can be found in the support links in section 1.5 References. There the user can specifically select their device and have tailored usage instructions.

4.1 User Authentication

When allowed, a user will be able to enroll fingerprint biometrics for use at the lock screen as an alternative to entering a password. Detailed instructions for configuring these methods can be found under the "Secure" or "Security" section of the guide for the specific device. However as explained in Section 2.5.1, the device provides on-screen help for enrolling a fingerprint.

4.1.1 Setting Passwords

Passwords and biometrics are available (depending on the configuration) for use to prevent unauthorized access to the device. A user must always have a primary method of authentication set for authentication and should not be shared with anyone. Recommendations for setting strong passwords can be found in NIST SP 800-63B, section 5.1.1, Memorized Secrets.

4.1.2 Two-step Verification

When the work profile is configured for Two-step verification (also called multi-factor or hybrid authentication), the user must provide both a biometric and password to login successfully. The user will see a new option in the Screen Lock Type that will allow the user to configure both components of the authentication credentials.

When the Two-step verification is selected, the user will be prompted to choose the first lock type, which will be a Password. Once the password has been entered, the user will be prompted to enter a biometric from those available for use. If the biometric has not yet been registered, the user will be prompted to re-enter the password before continuing to register the biometric.

The process for entering the password or registering a biometric in the same manner as when used individually (specified in 4.1 User Authentication). The Two-step verification process provides a wizard to register both components at once.

4.2 Wi-Fi Connectivity

While the administrator may pre-configure some Wi-Fi networks via the EDM, the user has local control over the Wi-Fi connectivity of the device, including the ability to enable/disable Wi-Fi and to connect/reconnect to networks. Detailed instructions for connecting to Wi-Fi networks can be found under the "Connections" section of the guide for the specific device.

Wi-Fi connections can sometimes be dropped (such as when moving out of range). Generally, the device will automatically reconnect to the network once in range, but when this does not happen, following the steps used to establish a new connection by selecting the available network would start the reconnection. This process will not require re-entry of any configuration information but will start the connection using the configuration already stored.

4.3 Bluetooth Connectivity

When connecting your device to various other Bluetooth devices it is important to be sure they are properly paired. To Pair a device, use the following steps:

1. Open your phone or tablet's Settings app
2. Tap Connected devices › Connection preferences › Bluetooth. Make sure Bluetooth is turned on.
3. Tap Pair new device.
4. Tap the name of the Bluetooth device you want to pair with your phone or tablet.
5. Follow any on-screen steps.

Detailed instructions including pictures for pairing Bluetooth devices can be found under the “Connections” section of the user guide for the specific device or in the Interactive Guide under “Connections -> Connect to Bluetooth Devices”.

4.4 Cellular/Mobile Network Configuration

There may be times when it is necessary to limit the type of Cellular network(s) to which a device should be allowed to connect. The device can be configured to connect to specific combinations of network modes such as 5G, LTE, 3G and 2G. The specific options may be limited by a combination of the SIM and the carrier the phone is connected to at any time (such as when roaming).

To change the network modes used to connect to the cellular network, the user can search for “Mobile Networks” in the user guide. Inside the Mobile Networks settings, the new can select “Network Mode” and choose from the available modes. In many cases the selections will have 2 or more modes with (auto connect) specified; this means the device will connect to any of the listed modes to provide the best cellular connection.

4.5 Certificate Management

While generally certificates would be managed through the EDM, it may be necessary for a user to update the Trust Anchor database locally. A user is not able to change settings managed by the EDM, but is able to add, remove or disable certificates outside the restrictions an EDM may enforce. Detailed instructions for managing certificates locally can be found under the “Credential Storage” section of the user guide for the specific device.

A work profile has an independent Trust Anchor database that is managed separately through the EDM or locally. Managing certificates locally within the work profile will follow the same steps as outside the work profile, but require the user to be authenticated to the work profile to access the settings.

4.6 Using the VPN Client

4.6.1 Always-on Tunnel

When the device has a tunnel configured for Always-on VPN, all traffic will automatically go through this tunnel, and if for some reason a connection for the tunnel cannot be made, no traffic will be allowed to communicate off the device.

4.6.2 "Normal" VPN Tunnels

When VPN tunnels are configured and no tunnel is specified as Always-on, then the user must select the tunnel to be used. The user will select the tunnel from those available at ***Settings/Connections/More connection settings/VPN***.

4.7 Application Permissions

Applications may request access to system services, such as location, to support the functionality of the application. When an application is run for the first time, the user will be prompted to allow (or deny) access to the service for the application. Some services may also have an option for allowing access only when the application is running (preventing access when the application is not active on the screen). Unless a choice is made to allow access one time only, the selection made by the user will be remembered across application restarts.

These permissions can be managed on the device later in the Permission manager available at ***Settings|Privacy|Permission manager***. Here the permissions for each application can be checked and modified as desired.

5 Audit Records

Auditing is enabled and events retrieved through the EDM. A Knox Platform for Enterprise license is required in order to enable the collection of audit records.

Audit records are stored in a compressed format to minimize space and maximize the amount of records that can be stored. When the allocated space is full, the oldest events will be overwritten so the most recent as always maintained (circular logging/buffering). Notifications are sent to the EDM based on the log space becoming full to warn before wrapping occurs.

The minimum amount of allocated space for audit storage is 10MB with a maximum of 50MB, depending on the available free space when activated. There must be at least 200MB of free space when Auditing is enabled (an error is returned to the EDM if not), and no more than 5% of free space will be used, up to the maximum of 50MB. The allocated space is not adjusted after it is initially set.

Within the logging, it is also possible to filter the events that are written to the log.

One important note about the audit capabilities is that they are tied to being enrolled to a management server (EDM). If the device is not enrolled there is no way to enable auditing, and when a device is unenrolled, the audit records are deleted as part of the unenrollment process, so any events created between the last review/upload and the unenrollment will be lost.

5.1 Audit Record Fields

KNOX API Logs:

The KNOX API audit records have eight (8) fields as described in the following table.

Setting	Description
Timestamp	Long value that represents the UTC timestamp
Severity	Integer value representing the severity: 1 (alert), 2 (critical), 3 (error), 4 (warning), 5 (notice)
Group	Integer value representing the group code: 1 (security), 2 (system), 3 (network), 4 (events), 5 (application)
Outcome	Integer value representing the outcome of the event: 1 (success), 0 (failure)
PID	Integer value representing the process ID
USERID	Integer value representing the USERID for which the log was originated ID 0 is for a normal user ID -1 is for system events ID 10-12 or 100-102 is for work profile users (multiple work profiles can be defined, but only one is ever active at one time). IDs 100-102 are for legacy Knox profiles. Separated app profiles will be 10-12.
Component	String representing the facility/Software Component name
Message	Free-form message description of the event (generally a human-readable message)

Table 13 – KNOX API Audit Fields

Security Logs:

A MDM agent acting as Device Owner can control the logging with DevicePolicyManager#setSecurityLoggingEnabled. When security logs are enabled, device owner apps receive periodic callbacks from DeviceAdminReceiver#onSecurityLogsAvailable, at which time new batch of logs can be collected viaDevicePolicyManager#retrieveSecurityLogs. SecurityEvent describes the type and format of security logs being collected.

Audit events from the Security Log are those where the "Keyword" field appears first in the format. For example: <Keyword> (<Date><Timestamp>): <message>

Logcat Logs:

Logcat logs can be read by a command issued via an ADB shell running on the phone. Information about reading Logcat logs can be found here. The command to issue a dump of the logcat logs is:

> adb logcat

Logcat logs cannot be exported from the device outside of using the above ADB command to dump to a file, then retrieving the file from the device (which requires developer settings enabled and administrative permissions).

Logcat logs can also be read by an application (for example an MDM agent) granted permission from an ADB shell:

> adb shell pm grant <application_package_name> android.permission.READ_LOGS

Audit events from the Logcat log are those where the "Keyword" field appears after the timestamp field in the format. For example: Date> <Time> <ID> | <Keyword> <Message>

5.2 Audit Events

The list of audit records that are produced related to the functionality claimed in the MDFPP are listed in the attached spreadsheet. The Event column shows what the audit record that is generated, where the information in the <> may vary (such as the status of the setting being measured, or the value being reported). The Description column describes the audit record and may provide additional information about fields that may be displayed.

The WP-C and Legacy configurations both support work profiles, while the managed configuration does not. Auditing is done at both the device and work profiles. Each audit record is marked under each configuration to show where it can be generated.

AuditEventTable.xlsx

The events categorized with Common Criteria Status are generated when CC Mode is first enabled and on every device boot sequence thereafter. These events will not be generated again if CC Mode is called, but

will only occur during the boot sequence. If the check being made passes, the status will be OK. Otherwise, the message will show corrective actions to be taken.

Most of the management functions for the work profile (such as password management or camera access) generate the same messages as outside the work profile. The messages inside the work profile will be marked with the profile ID (usually 10 or 100 depending on the device).

6 Developer References

6.1 Cryptographic APIs

This section provides information for developers to utilize the evaluated cryptographic APIs while writing their mobile applications. The Reference Link points to more information about the APIs for the specific cryptographic functions.

Cryptographic Function	Evaluated API	Reference Link
AES-CBC 128/256	javax.crypto.Cipher	developer.android.com
AES-GCM 128/256	javax.crypto.Cipher	developer.android.com
SHA-1/256/384/512	java.security.MessageDigest	developer.android.com
HMAC-SHA-1/256/384/512	javax.crypto.Mac	developer.android.com
RSA Key Generation	java.security.KeyPairGenerator java.security.KeyFactory	developer.android.com
ECDSA Key Generation	java.security.KeyPairGenerator	developer.android.com
RSA Signing/Verification	java.security.Signature	developer.android.com
RSA Encryption/Decryption	javax.crypto.Cipher	developer.android.com
ECDSA Signing/Verification	java.security.Signature	developer.android.com
ECDH Key Agreement	java.security.KeyPairGenerator javax.crypto.KeyAgreement	developer.android.com
RBG Random Generation	java.security.SecureRandom	developer.android.com
Certificate Verification	java.security.cert.CertPathValidator	developer.android.com
Key Import, Use, Destruction	javax.crypto.KeyGenerator java.security.KeyPairGenerator java.security.KeyStore android.security.KeyChain	developer.android.com developer.android.com

Table 14 – Cryptographic API Reference

Developers can utilize with the KeyStore or the KeyChain to store their keys/credentials, depending on type of key (symmetric keys can only be stored in the KeyStore). Keys stored in the KeyStore can only be accessed (used or deleted) by the original app or by apps with a common developer with enforcement handled by the KeyStore. Keys stored in the KeyChain can be made globally available (with explicit approval by the user). When a key is imported/created it is assigned authorizations for use which cannot be changed later (i.e. what the key can be used for, how long the key can be available).

6.2 Bluetooth APIs

The device provides access to Bluetooth functions through a standard set of APIs. These can be found at developer.android.com under android.bluetooth and android.bluetooth.le.

6.3 TLS/HTTPS APIs

The device provides access to TLS & HTTPS functions through a standard set of APIs. These can be found at developer.android.com under javax.net.ssl.

6.4 Certificate Pinning

The device provides the ability for applications to utilize certificate pinning to lock the certificates accepted when accessing web services to only those that are specifically expected. This must be done by the app and is not something the user can set on their own. Information about configuring an app to utilize certificate pinning can be found at developer.android.com under Network Security Configuration.

6.5 IPsec VPN APIs

The device provides the ability to configure IPsec VPN tunnels through a standard set of APIs. These can be found at developer.android.com and at the Samsung Knox SDK API reference.

7 Device Delivery and Updates

7.1 Secure Device Delivery

While a Samsung device requires initial configuration before it can be added to the enterprise environment, it is also critical to ensure that the device is received prior to configuration in a secure manner, free from tampering or modification.

It is very important that the devices to be deployed into the enterprise are obtained from reputable carriers to reduce the likelihood that tampering of devices may occur.

Upon receipt, the boxes containing the device should have both a tracking label and two labels placed at either end of the box to indicate whether the box has been opened prior to delivery. If these seals are broken, do not accept the device and return it to your supplier.

The tracking label should look similar to Figure 2 - Tracking Label, while the two tamper labels should appear similar to Figure 3 - Security Seal (Black) or Figure 4 - Security Seal (White).

Figure 2 - Tracking Label

Figure 3 - Security Seal (Black)

Figure 4 - Security Seal (White)

7.1.1 Evaluation Version

There are a number of components to determining the device that is being used and the components on that device (such as the operating system version, the build version, etc.). These are all contained under Settings/About device. The following are version information that can be found:

- ***Model number*** – this is the hardware model
- ***Android version*** – this is the Android OS version
- ***Build number*** – this is the specific binary image version for the device
- ***Security Software Version*** – this shows the Common Criteria evaluations and the version of the software components related to those evaluations on the device

For the Common Criteria evaluation version information see section 1.4.2 Device Details.

7.1.2 Pre-packaged Software Versions

Samsung Android devices come with large amounts of software apps to provide the full breadth of functionality expected by the customer. Some of the apps come from Google, some from Samsung, and others from the cellular carrier. For a list of the apps and their versions contained on a specific device, visit the website where you can download the CC Mode app and select the device you are using. This will provide a complete list of the software installed on the evaluated device.

7.1.2.1 Software Versions on Device

To verify the versions of any software on the device (compared to the list from the website), open ***Settings/Application manager***. Under the heading ***All***, you will see every application on the device (both those that are pre-installed and any you have installed). Selecting an application will display its properties. The version number is shown at the top under the name.

Note: Using adb (USB debugging must be enabled to use adb) it is possible to extract all package version information at once.

7.2 Secure Updates

Once a device has been deployed, it may be desirable to accept updates to the software on the device to take advantage of the latest and greatest features of Samsung Android. Updates are provided for devices as determined by Samsung and the carriers based on many factors.

When updates are made available, they are signed by Samsung with a private key that is unique to the device/carrier combination (i.e. a Galaxy S9 on Verizon will not have an update signed with the same key as a Galaxy S9 on AT&T). The public key is embedded in the bootloader image, and is used to verify the integrity and validity of the update package.

When updates are made available for a specific device (they are generally rolled out in phases across a carrier network), the user will be prompted to download and install the update (see the User Guide for more information about checking for, downloading and installing the update). The update package is checked automatically for integrity and validity by the software on the device. If the check fails, the user is informed that there were errors in the update and the update will not be installed.

7.2.1 Allowed Update Methods

When CC Mode is enabled, only FOTA updates can be installed on the device. Other methods for installing updates (such as Recovery Mode or Samsung KIES) are blocked and cannot be used to update the firmware. This provides insurance against local, physical attacks that could change the software unknowingly.

7.2.2 Blocking Updates

It is possible to block FOTA updates on a device by setting **allowOTAUpgrade()** to be false via the EDM. This can be used either to freeze the software installed or to allow an organization time to test the update before letting it roll out to the user community.

8 Operational Security

8.1 Modes of Operation

The mobile device can be operated in four different modes, depending on the role of the user accessing the device:

- Administrator mode;
- User mode;
- Error mode; and
- Recovery mode

A device is considered to be in Administrator mode before it is delivered to the user. The device is prepared and configured for deployment in the enterprise environment via the Samsung Enterprise SDK. The mobile device administrators are trusted to follow and apply all administrator guidance in a trusted manner. An unprivileged user will not have access to this mode of operation.

If an error or operational failure occurs during the transition from Administrator mode (causing the device to enter the Error mode of operation) to User mode, the administrator should follow the guidance for the EDM for the failure and restore the device to normal operational abilities. If it is not possible to adequately eliminate the error or operational failure, the device is not to be delivered to an end user and should be returned to the supplier.

After the device is configured in accordance with the Common Criteria evaluated settings, the device is ready for deployment to a user. When the user receives the device, only the TouchWiz user interface will be visible and no further changes to the security configuration are possible. Once deployed to a user, the device will be operating in User Mode. Within User Mode, the only security relevant functions accessible for the user are ‘lock screen password protection’, ‘change of password’ and ‘local device wipe’. Typically, an administrator will not access the device in this mode of operation.

The mobile device may also be placed into Recovery mode, bypassing the standard boot process and allowing configuration changes to be made to the installation of Android. However, since this requires the boot loader for the device to be unlocked and is therefore considered out of scope for this environment.

8.2 Wiping Data

The evaluated security configurations provide the ability to both locally or remotely wipe data work profile level or both.

An enterprise initiated remote wipe command (for either the device or just the work profile, depending on the configuration) occurs under the following conditions:

- The enterprise sends a remote wipe command to the device:
    - when the device has been lost or stolen;
    - in response to a reported incident;

- in an effort to resolve current mobile issues; and
- for other procedural reasons such as when an Android device end user leaves the organization.

8.2.1 Wiping the Device

The evaluated security configuration provides for a local and a remote wiping process of Android user devices. This type of wipe works at the storage level and will wipe all data on the device. In a work profile configuration, this will wipe all data including the work profile (as well as everything not in the work profile). This type of wipe is available in all configurations.

The local wipe is manually initiated by the Android device user or after an exceeded number of incorrect login attempts. The remote wipe process is in general remotely initiated by the Enterprise Device Administrator via a remote wipe command.

8.2.2 Wiping the Work Profile

When a Work Profile has been created, it is also possible to wipe only the data stored in the work profile. A wipe of the work profile data will remove the work profile, including apps and data, but it will not remove anything outside the work profile. This process must be initiated remotely by the Enterprise Device Administrator via a remote wipe work profile command.

The only way for a user to wipe the work profile is to unenroll the device from the control of the EDM. When this is done the work profile, all data and apps as well as the EDM Agent will all be removed from the device.