CSPN Security Target
KNOX Container Technology

Document Status

Application date	Not applicable
Current version	1.0-en

	Developers	**Sponsor**	**Evaluator**
Organisation(s)	Samsung	Samsung	Trusted Labs

Distribution

Name or role	**Organisation**
Developers	Samsung
Evaluator	Trusted Labs
Certifiers	ANSSI

Revision History

Date	**Version**	**Comment**
2013-12-24	0.1-en	English initial version
2013-12-24	0.3-en	Integration of Samsung Initial comments
2015-01-23	0.4-en	Change of TOE – Knox 2.0 version
2015-02-10	0.5-en	New version following Samsung comments
2015-02-24	0.6-en	
2015-02-27	0.7-en	
2015-03-12	0.8-en	
2015-05-05	0.9-en	Updated following test environment discussion. Updated product information.
2015-05-06	1.0-en	Updated following review. Up-issued to 1.0

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

Legal Information

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

Though every care has been taken to ensure the accuracy of this document, Samsung Electronics Co, Ltd. cannot accept responsibility for any errors or omissions or for any loss occurred to any person, whether legal or natural, from acting, or refraining from action, as a result of the information contained herein. Information in this document is subject to change at any time without obligation to notify any person of such changes.

Samsung Electronics Co, Ltd. may have patents or patent pending applications, trademarks copyrights or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give the recipient or reader any license to these patents, trademarks copyrights or other intellectual property rights.

No part of this document may be communicated, distributed, reproduced or transmitted in any form or by any means, electronic or mechanical or otherwise, for any purpose, without the prior written permission of Samsung Electronics Co, Ltd.

The document is subject to revision without further notice.

All brand names and product names mentioned in this document are trademarks or registered trademarks of their respective owners.

Table of Contents

Table of Figures

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

1 Introduction

This document is the security target for the CSPN evaluation [RD1] of the Knox Container Technology.

1.1 PRODUCT IDENTIFICATION

Category	Identification
Product commercial name	Samsung KNOX
Evaluated version number	KNOX 2.3 Container 2.3.0 on SM-G900F Container 2.3.1 on SM-N910F SE Android 2.2.1
Product evaluated on	SM-N910F Samsung Galaxy Note 4 with Android 5.0.1 SM-G900F Samsung Galaxy S5 with Android 5.0.0
Product category	Embedded software

1.2 REFERENCES

Code	Reference	Name
[RD1]	N°915/SGDN/DCSSI/SDR du 25 avril 2008	Certification de sécurité de premier niveau des technologies de l'information
[RD2]	July 2014	Samsung KNOX Workspace, User Guide – KNOX 2.0

1.3 ACRONYMS

AES	Advanced Encryption Standard
CSPN	Certification Sécuritaire de Premier Niveau
EMM	Enterprise Mobility Management
FIPS	Federal Information Processing Standard
IPSec	Internet Protocol Security
IT	Information Technology
MAC	Mandatory Access Control
MCM	Mobile Container Management
MDM	Mobile Device Management/Manager
ODE	On-Device Encryption
OS	Operating System

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

PBKDF	Password-Based Key Derivation Function
PKM	Periodic Kernel Measurement
RCP	Remote Content Provider
RKP	Real-time Kernel Protection
ROM	Read-Only Memory
SSL	Secure Sockets Layer
SSO	Single Sign On
TEE	Trusted Execution Environment
TIMA	TrustZone-based Integrity Measurement Architecture
TOE	Target Of Evaluation
VPN	Virtual Private Network

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

2 Product Description

2.1 General Product Description

Samsung KNOXv2.x is the next-generation of the secured Android platform introduced by Samsung in 2013 as Samsung KNOX. Targeted primarily at mid and high-tier devices, it leverages hardware security capabilities to offer multiple levels of protection for the operating system and applications.

Figure 1: Samsung KNOXv2 overview

Samsung KNOX Workspace offers a multi-faceted security solution rooted in the tamper-resistant device hardware, through the Linux kernel and the Android operating system.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

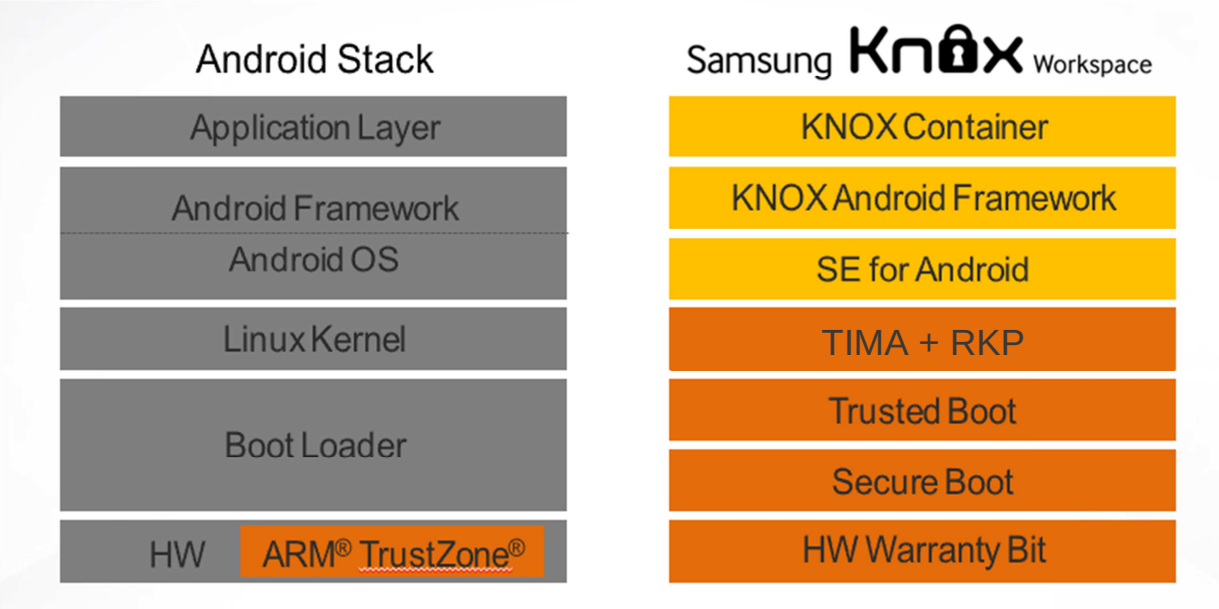

Figure 2: Android and KNOX stacks

The KNOX platform includes a number core features that address enterprise security needs:

- Hardware Warranty Bit to prevent access to Samsung KNOX Container and prevents the Trustzone Key Store from releasing keys if an unauthorized kernel is detected. The Warranty Bit cannot be reset, even by reloading Samsung original binaries.
- Secure Boot and Trusted Boot to ensure authenticity and integrity of the boot chain
- TrustZone-based Integrity Measurement Architecture (TIMA) providing integrity monitoring of the Kernel
- Trustzone Real-time Kernel Protection
- TrustZone-based Remote Attestation functionality, allowing on-demand attestation of device integrity.
- SE for Android integration to enforce Mandatory Access Control (MAC) policies to isolate applications and data in the platform.
- TrustZone-based Key Store to provide protection using a device unique key in hardware for encryption keys.
- TrustZone-based On-Device Encryption - system integrity verified at boot time before data decryption occurs
- The Samsung KNOX Container providing an isolated Android environment to secure enterprise applications and data from applications outside the container
- Comprehensive Mobile Device Management support to allow administrators to manage devices and apply security policies in a flexible manner to meet their mobility requirements.
- A multi-vendor Generic VPN framework with a variety of third-party SSL and IPSec VPN available
- An open smart card framework that enables third party vendors to integrate smart card solutions which applications can access within the KNOX environment via standardized interfaces.

Samsung also provides an ecosystem of enterprise services that build on top of these core device security components, such as Single-Sign On, Active Directory support and enterprise application stores.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

Samsung KNOX is available with licenses targeting a range of different users, offering different levels of device management and features. Samsung KNOX Workspace offers the full feature set and flexible device management.

Samsung MyKnox, Express and Premium versions use the same Container technology as KNOX Workspace with a simplified feature range available to the customer.

The current security target focuses on Samsung KNOX Container technology when used with features available in the Workspace version of the product.

The subsections that follow describe Samsung KNOX application and platform security services.

2.1.1 APPLICATION SECURITY SERVICES

Samsung KNOX Workspace Container

Samsung KNOX Workspace is a defense-grade dual persona container product designed to separate, isolate, encrypt, and protect enterprise data from attackers. This work/play environment ensures work data and personal data are separated and that only the work container is managed by the enterprise. Personal information such as photos and messages are not managed or controlled by the IT department. Once activated, the KNOX Workspace product is tightly integrated into the KNOX platform.

Workspace provides this separate environment within the mobile device, complete with its own home screen, launcher, applications, and widgets.

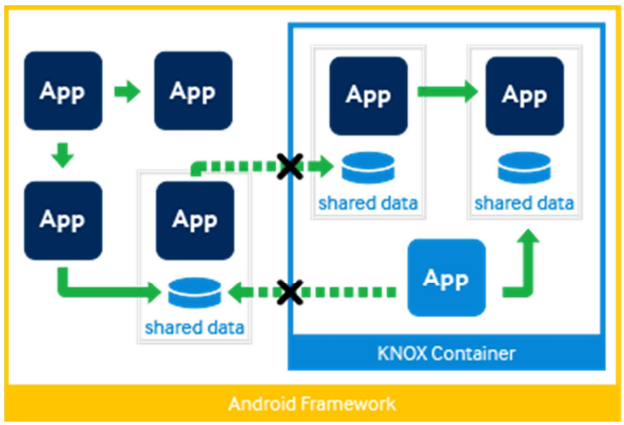

Figure 3: KNOX Container Data Isolation

Applications and data inside the container are isolated from applications outside the container, that is, applications outside the container cannot use Android inter-process communication or data-sharing methods with applications inside the container.

For example, photos taken with the camera inside the container are not viewable in the Gallery outside the container. The same restriction applies to copying and pasting. Note that the contacts and calendar apps represent an exception, since container contacts and the calendar can be made visible inside the KNOX container and in the personal work space.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

When allowed by IT policy, some application data such as contacts and calendar data can be shared across the Workspace boundary. The end user can choose whether to share contacts and calendar notes between Workspace and personal space; however, IT policy ultimately controls this option.

All data stored by applications inside the container are encrypted in a dedicated container file system using a strong encryption algorithm (AES-256). A password is required to gain access to applications inside the container.

On Device Encryption

The On-device Data Encryption (ODE) feature allows users and enterprise IT administrators to encrypt data on the entire device, as well as any configured Samsung KNOX Container. The ODE feature on Samsung devices uses a FIPS 140-2 compliant Advanced Encryption Standard (AES) cipher algorithm with a 256-bit key (AES-256) and offers the levels of security required by government and regulated industries such as healthcare and finance. The key utilized for this encryption is derived from a user-created passphrase using well-known key-derivation algorithms such as Password-Based Key Derivation Function 2 (PBKDF2).

TrustZone protection for ODE can be optionally enabled by the administrator. If this is enabled, the ODE mechanism encryption key derivation requires both the correct user passcode to be entered before the operating system is booted, but also the integrity of the system to be verified before cryptographic keys are released from TrustZone. This means ODE mechanism is afforded the same hardware based protection as the KNOX container encrypted file system.

Mobile Device Management

Samsung KNOX builds upon Samsung's industry leading Mobile Device Management (MDM) capabilities by providing additional policies for security, enterprise integration, and enterprise applications such as asset tracking, remote control, and so on.

The enterprise can manage the container like any other IT asset using an MDM solution; this container management process is called Mobile Container Management (MCM). Samsung KNOX supports many of the leading MDM solutions on the market. MCM is affected by setting policies in the same fashion as traditional MDM policies. Samsung KNOX Workspace includes a rich set of policies for authentication, data security, VPN, e-mail, application blacklisting, whitelisting, and so on.

Virtual Private Network

KNOX provides a rich set of VPN features to address a wide-range of enterprise mobile device deployment scenarios. At the core of KNOX VPN framework is the Generic VPN Service that enables VPN vendors to provide a wide range of features and configurability.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

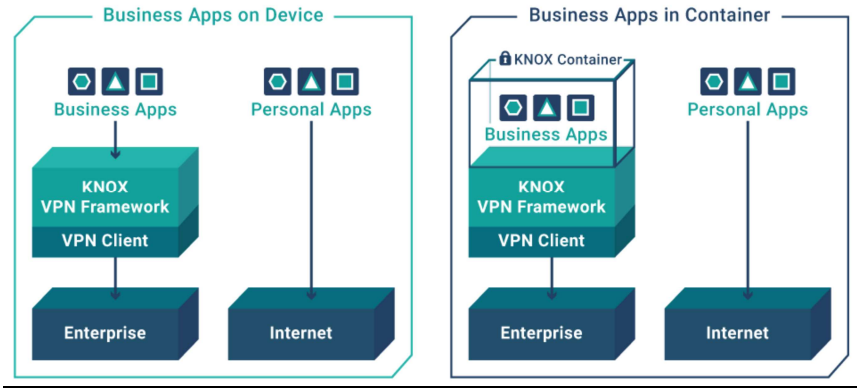

Figure 4: KNOX Per-App VPN

VPN features of KNOX include:

- Administrator-configured System VPN
- Administrator-configured Per-App VPN
- Administrator-configured Workspace VPN
- Multiple concurrent VPN connections
- IPsec and SSL VPN support
- Administrator-configured FIPS and non-FIPS VPN mode
- Common Access Card (CAC)-based authentication
- Always on VPN connections with auto-reconnect
- VPN tunnel chaining

2.1.2 PLATFORM SECURITY SERVICES

Secure Boot and Trusted Boot

The security of the device boot process is a core part of Samsung's layered security approach to the KNOX platform.

The startup process for Android begins with the primary bootloader, which is loaded from ROM.

The boot process is sequential in nature with each bootloader completing its task and executing the next bootloader in the sequence, finally loading the Android bootloader which loads the Android operating system.

Figure 5: Android Boot Sequence

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

Secure Boot is a security mechanism that prevents unauthorized bootloaders and operating systems from loading during the startup process. Secure Boot is implemented by each bootloader cryptographically verifying the next bootloader in the sequence using a certificate chain that has its root-of-trust resident in the hardware. The boot process is terminated if verification fails at any step.

Secure Boot is effective in preventing unauthorized bootloaders (and sometimes the kernel when it is also applied to the kernel binary). However, Secure Boot is unable to distinguish between different versions of authorized binaries. Secure Boot is not effective in preventing non-Samsung kernels from running on these devices. This exposes an attack surface that poses a potential threat to enterprise applications and data.

Samsung KNOX implements Trusted Boot (in addition to Secure Boot) to address this limitation of Secure Boot.

With Trusted Boot, measurements of the bootloaders are recorded in secure memory during the boot process. At runtime, TrustZone applications use these measurements to make security-critical decisions, such as verifying the release of security keys, container activation, and so on.

Additionally, if the bootloader is unable to verify the Android kernel, the Hardware Warranty Bit, a one-time programmable memory area (colloquially called a fuse) is written to indicate suspected tampering. Even if the boot code is restored to its original factory state, this evidence of tampering remains.

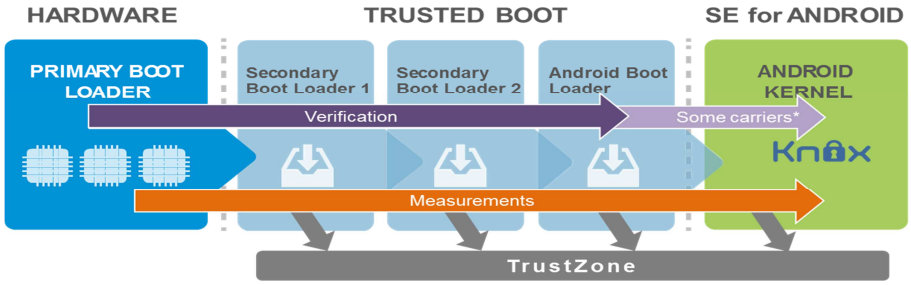

Figure 6: Samsung Trusted Boot

However, the boot process is not halted, and the bootloader continues to boot the Android operating system. This process ensures that normal operation of the device is not affected.

Security Enhancements for Android

Samsung KNOX introduced Security Enhancements for Android (SE for Android) in 2012 to enforce Mandatory Access Control (MAC) policies. These enhancements protect applications and data by strictly defining what each process is allowed to do, and which data it can access.

Samsung's innovative collaborations with the authors of SELinux resulted in the gold standard for Android security. In version 4.4 of AOSP, Google introduced a subset of the SE for Android enhancements Samsung pioneered (i.e., the SELinux portion). Samsung continues to lead Google, and all others, in continuing to implement new extensions of SE for Android.

Our improvements allow us to protect areas of the Android framework to which access was previously unrestricted. Our policy protects software created by Samsung, AOSP, and other third-party partners. The increased enforcement granularity from our AOSP enhancements, and Samsung's industry-leading granular access policies that define over 200 unique security domains, are designed together to enforce the tightest restrictions with the lowest rates of over- or under-privileging.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

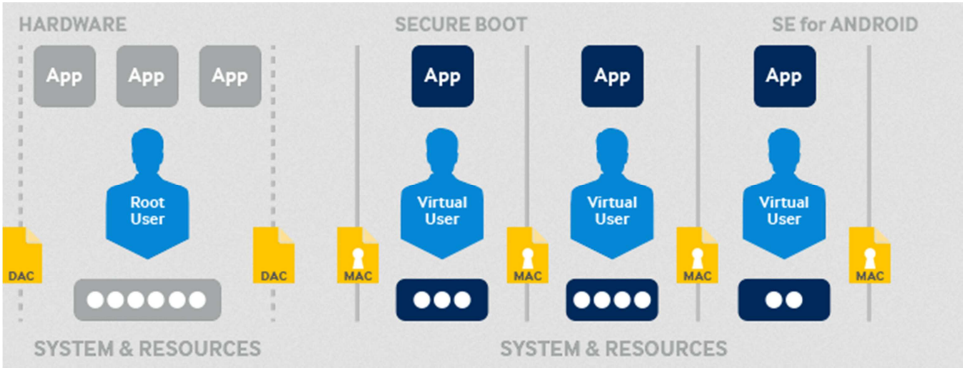

Figure 7: SE for Android MAC

This architecture prevents a compromise in one domain from propagating to other domains or the underlying mobile operating system (OS). This additional security reduces threats of tampering and bypassing of application security mechanisms. It also minimizes the amount of damage that can be caused by malicious or flawed applications, as applications are provided the minimum amount of permission required for their task.

TrustZone-based Integrity Measurement Architecture (TIMA)

The system protection offered by SE for Android relies on the assumption of OS kernel integrity. If the kernel itself is compromised (by a perhaps as yet unknown future vulnerability), SE for Android security mechanisms could potentially be disabled and rendered ineffective.

Samsung's TrustZone-based Integrity Measurement Architecture (TIMA) was developed to close this vulnerability. TIMA leverages hardware features, specifically TrustZone, to ensure that it cannot be pre-empted or disabled by malicious software.

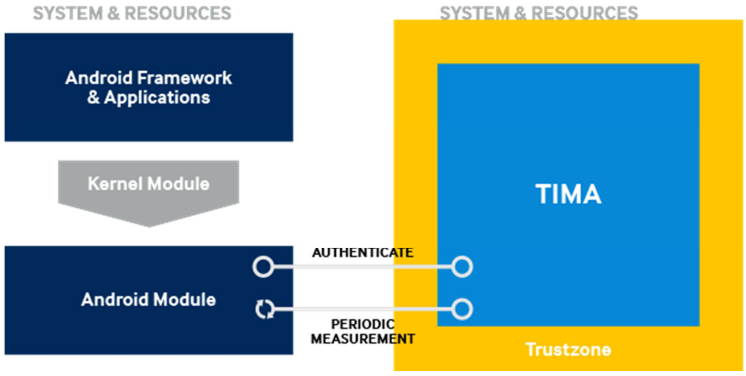

Figure 8: TrustZone-based Integrity Measurement Architecture

TIMA Periodic Kernel Measurement (PKM) performs continuous periodic monitoring of the kernel to detect if legitimate kernel code and data have been modified by malicious software. In addition, TIMA also monitors key SE for Android data structures in OS kernel memory to prevent malicious attacks from corrupting them and potentially disabling SE for Android.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

RKP uses special techniques to take full control over the normal world memory management and intercept critical events and inspect their impact on security before allowing them to get executed. Hence, RKP complements TIMA-PKM's periodic kernel integrity checking, which has limited effectiveness against attacks that can take place and properly hide their traces between periodic checks. TrustZone-RKP achieves three important security features:

- First, it completely prevents running unauthorized privileged code (i.e., code that has the kernel privilege) on the system
- Second, it prevents kernel data from being directly accessed by user processes.
- Third, RKP monitors some critical kernel data structures to verify that they are not exploited by attacks.

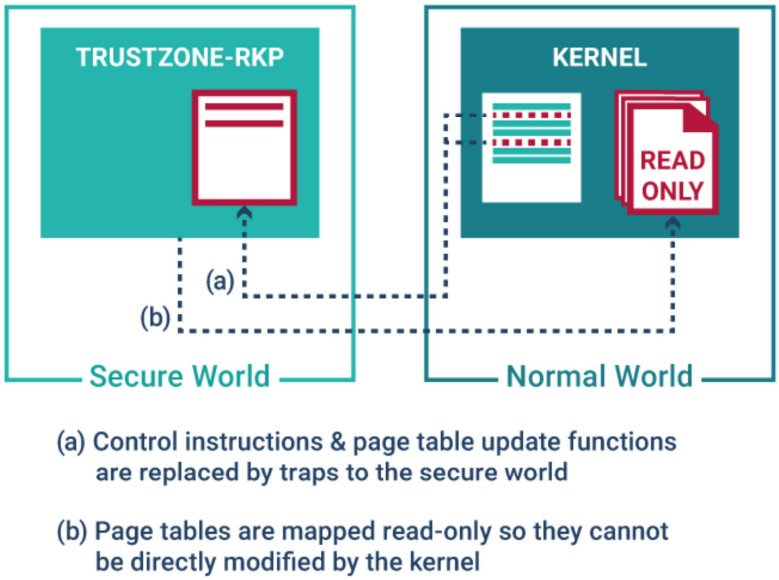

Figure 9: Real-time Kernel Protection

Remote Attestation

TIMA Attestation allows a device to attest facts about its state to a remote server, such as an MDM server. The attestation message contains state measurements that can be evaluated by a server, which can then decide whether to trust the device or not. The full attestation message is computed in the ARM TrustZone Secure World, and thus is accurate even if the entire normal world OS is compromised.

To ensure unforgeability, the attestation message is signed using the TIMA Attestation Key, which is traceable to Samsung's root key. Each Samsung device supporting TIMA attestation has a unique RSA key pair, the device root key (DRK). The DRK is generated during manufacture, is traceable to Samsung's root key using X.509 certificates, and is stored in TrustZone. The remote server can verify the integrity of this message using Samsung's root key. To ensure that the attacker of a compromised device cannot replay old valid attestation messages, the signature includes a server-generated cryptographic nonce, which is a random number used only once.

Depending on the attestation verdict and the data, any further action is determined by the enterprise's MDM security policy. The security policy might choose to detach from the device, erase the contents of the secure Workspace, ask for the location of the device, or any of many other possible security recovery procedures.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

2.2 PRODUCT USAGE

Activation

Samsung KNOX Workspace has to be activated by the end user IT admin through EMM or another MDM solution.

Configuration:

Once the IT administrator has activated Samsung KNOX, an end user can install it and set it up.

1. Following provisioning by the MDM, The Samsung KNOX Terms and Conditions are displayed to the user.
2. Agree to the Terms and Conditions, check “I agree” and “I accept all the terms above” and click Confirm.
3. Select a KNOX Workspace unlock method. Unlock methods can be password, PIN, pattern or Fingerprint (subject to Administrator policy).
4. Set the PIN, password, pattern or fingerprint which you will need to enter to access the Workspace; tap Continue.
5. Click Next when prompted to install a shortcut on the home screen. The Container is then setup including creating the secure file system and setting up applications within the Container.
6. When complete, the KNOX Workspace automatically launches.

Operation:

After configuration by the end user, the Samsung KNOX environment is ready to execute container applications and to allow installation of new applications. Container applications and data are isolated from the personal Android environment.

1. To log into the KNOX container, either:
    a. Tap the KNOX icon, or
    b. Swipe down the Notifications bar, and then tap KNOX Tap to start
2. Enter the KNOX password you selected when you set up the container.
3. Tap Done. The KNOX home screen is displayed.

The user will be able to switch between the personal Android environment and the KNOX container environment without re-unlocking as long as the inactivity period does not exceed the timeout value (set by default to 10 minutes, but configurable) and that the end-user has not manually locked the container.

2.3 INTENDED OPERATING ENVIRONMENT

The operating environment for the KNOX container implementation is composed of:

- Samsung Android 5.0.x (Lollipop) operating system incorporating
    - SE for Android
    - Samsung KNOX 2.3 Framework components
- Secure boot and software integrity measurement mechanisms as described in Section 2.1;
- Preloaded or user installed applications in the base Android environment.
- MDM client and console to manage KNOX Containers and security policies on registered devices
- Optional VPN configuration to protect all Container network traffic, including application data, in transit

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

2.4 ENVIRONMENT ASSUMPTIONS

Final product usage:

The assumptions on the environment for the correct operation of the KNOX container during final product usage are as follows:

- **A.SECURE_BOOT**
  The Secure Boot mechanism verifies mobile firmware authenticity during boot, preventing unauthorized modifications to the boot chain.

- **A.TRUSTED_BOOT**
  The Trusted Boot mechanism only permits the TIMA Keystore to release keys used in the process of deriving container encryption keys if measurements taken of the bootloader and kernel images are deemed valid.

- **A.RUNTIME_INTEGRITY**
  TIMA PKM and RKP prevent modification of any kernel code and data from user processes and verifies at regular intervals that the security mechanisms of the SE for Android kernel are not tampered with. In particular, the TIMA mechanism verifies that SE for Android is operating in enforcing mode that does not allow bypassing rules of the security policy.

- **A.SECURE_FS**
  The Container file system is encrypted and suitably protects user data when the device is powered off such that files cannot feasibly be decrypted without the correct keys.

- **A.SECURE_FS_KEYS**
  Keys required to decrypt the Container file system are cryptographically tied to the device and user. The keys are protected such that it is not feasible for an unauthorized user or application to use them even if are accessed.

- **A.USER_AUTH_PASSWORD**
  The user authentication method to the Container is password. Pattern, PIN and fingerprint based authentication are not considered in this Security Target.

- **A.APPLICATION_INTEGRITY**
  Applications are checked for integrity during install, update and start up by the Android Package Manager. Applications that do not pass integrity checks are not permitted to be installed or used. When an application is updated, the fact that the update originates from the same author as the installed application is checked.

- **A.APPLICATION_SANDBOX**
  Application processes are isolated by Android's VM Sandboxing mechanism; where a separate Virtual Machine is started for each process and run under a per-application user. This provides runtime memory isolation and privilege control.

- **A.REMOTE_MANAGEMENT**
  MDM Policy is provisioned to the device via a secure and authenticated delivery mechanism. The MDM Policy is managed by a trusted IT Administrator.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

- **A.MAC_POLICY_UPDATE**
  The SE for Android Policy can only be remotely updated by Samsung servers, updates are signed to prove authenticity and integrity. The system will not accept invalid updates. The SE for Android Policy can only be updated by a system service on the device.

2.5 TYPICAL USERS

Typical users of Samsung KNOX Workspace are End Users and IT Administrators in Enterprise or Government organizations who require a managed mobile security solution.

2.6 EVALUATION PERIMETER

The evaluation focuses on the KNOX container technology executed on a platform such as described on Section 2.1.1. Evaluation perimeter is illustrated in Figure 10 by the components in red background. In particular, the components included in the perimeter are:

- SE for Android
- Samsung KNOX related SE Android security policy
- Samsung KNOX Framework components and services
  - Domain Separation, which provides separate user domains for applications and data
  - Device Management which provides policies to control applications and functionality available in the Container, provisioned through an MDM service.
  - Data Synchronisation, which allows data from authorised applications (Calendar and Contacts) data to be synchronised between the Personal world and a KNOX Container
  - Container Authentication component which provides user authentication and password management to control access to the Container
  - Relevant Samsung Android Framework components, modified to enforce MDM policies:
    - USB connectivity
    - SD Card access
    - Bluetooth
    - NFC

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

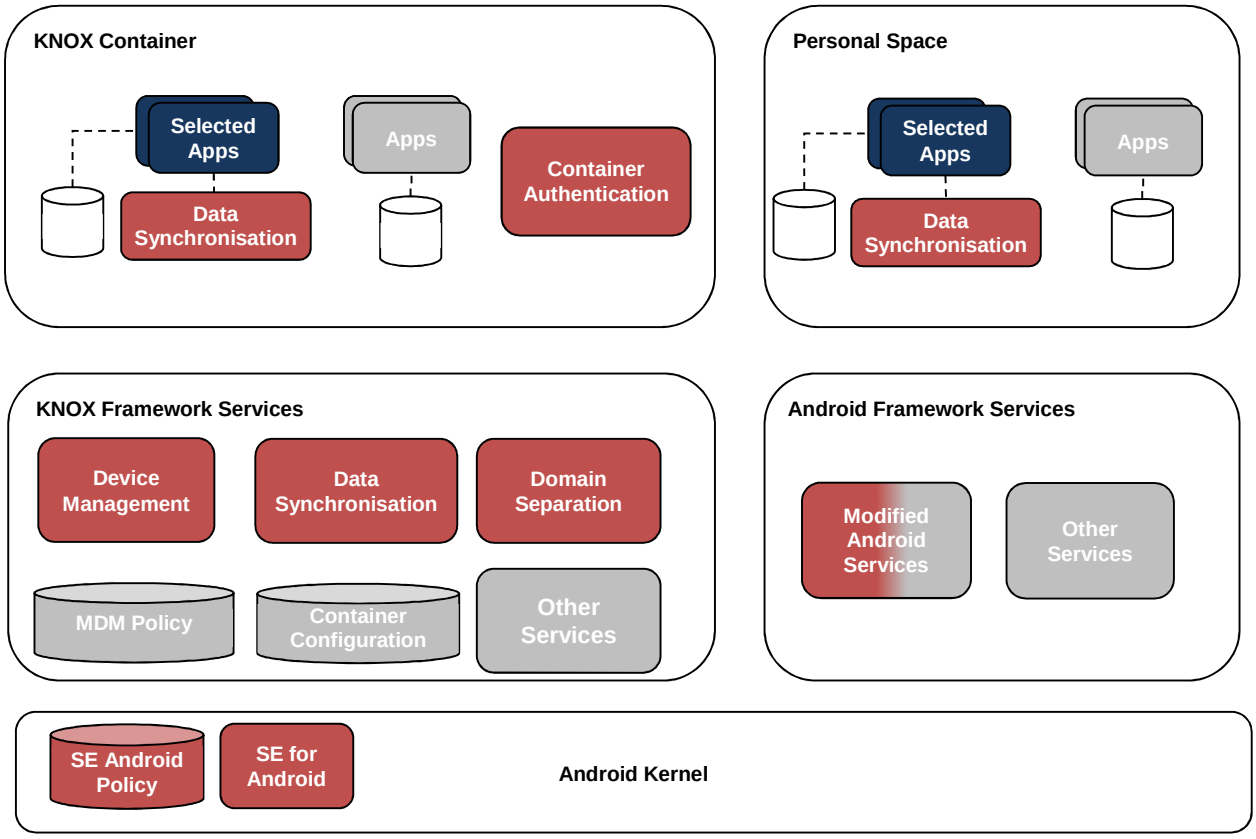

Figure 10: Evaluation perimeter

2.7 TEST PLATFORM

The test platform is composed of:

- Samsung Galaxy S5 and Note 4 smartphones, with the KNOX version identified in Section 1.1;
- Software tool allowing Container creation and configuration of MDM policies

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

3 Product Assets

3.1 SECURITY CRITERIA

The security criteria for assets are:

- Confidentiality: An asset is protected in confidentiality when it can only be accessible to authorized users.
- Integrity: An asset is protected in integrity when it can only be modified by authorized users.
- Authenticity: An asset is protected in authenticity when a forbidden modification is detected before the asset is used (and that a security violation occurs).

The relevant security criteria are described in the following for each asset protected by KNOX.

3.2 ASSETS PROTECTED BY THE TOE

The assets protected by the TOE are:

- D.USER_DATA: Container user data

  Set of user data managed by the applications present in the container. It may be banking information, credit card statements, financial investments, etc.

  User data is protected by the Domain Separation, Data Synchronisation and SE for Android components of the TOE and by assumptions A.SECURE_FS and A.APPLICATION_SANDBOX.

  Protection: *Confidentiality, Integrity*

- D.CONTAINER_CONFIG: Container configuration data.

  Set of configuration data that determines the run-time operation of the Container. This includes:

  - MDM device and container policies
  - Container State

  Container configuration data is stored outside the container encrypted file system and is protected by SE for Android. It may be optionally protected for Confidentiality by enabling ODE.

  MDM policies are managed by the Device Management component of the TOE and by assumption A.REMOTE_MANAGEMENT.

  Container State is managed by the Domain Separation component of the TOE.

  Protection: *Confidentiality, Integrity*

- D.CONTAINER_PASSWORD: Container password.

  Password set by the user during container configuration. The container password is used:

  - During derivation of Container file system encryption keys

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

- When the user authenticates to the KNOX container (unlock, change settings)

Management of this password is performed by the Container Authentication component of the TOE.

The Container password is verified by Samsung Key Management Module by decrypting part of the Container file system encryption key hierarchy. Additionally, the hash and salt value of the password is stored in a database on the device. This database is accessible by the system user only.

The Container Password is protected by SE for Android and assumption A.APPLICATION_SANDBOX.

Protection: *Confidentiality, Integrity*

- D.MAC_POLICY: SE for Android security policy.

  Set of SE for Android security policy rules that control access to system objects and files.

  These rules correspond to files loaded in memory and that can only modified through an update sent by Samsung servers.

  This policy is protected by SE for Android and A.MAC_POLICY_UPDATE. The SE for Android enforcing status is protected by TIMA PKM.

  The policy may be optionally protected for Confidentiality by enabling ODE.

  Protection: *Confidentiality, Integrity, Authenticity*

3.3 ASSETS OUTSIDE THE TOE PERIMETER

The assets stored outside TOE perimeter but used for nominal operation of the TOE are:

- D.CONTAINER_APPLICATION: Container applications.

  Set of KNOX applications preloaded in the container or installed by the user. They are stored alongside the personal android applications, but are executed within the container SELinux domain.

  Applications are integrity and authenticity protected at install, update and system start by assumption A.APPLICATION_INTEGRITY.

  Protection: *Integrity, Authenticity*

- D.FS_KEY: Encrypted file system cryptographic key.

  This AES256 key is derived from container password set by the user and from hardware dependent data. It protects the encrypted file system used by the container.

  File system keys are protected by assumptions A.SECURE_FS_KEYS.

  Protection: *Confidentiality, Integrity*

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

4 Threats

Considered threats for the evaluation perimeter defined in Section 2.6 against the assets of Section 3.2 are:

- **T.UNAUTHORISED_USER. Unauthorized user access:**
  An unauthorised user accesses applications and container data using the user interface.

  Related Assets: D.USER_DATA
  Coverage: FS.USER_AUTH, FS.USER_AUTH_LOCK, FS.USER_AUTH_FAIL
  Related Assumptions: A.USER_AUTH_PASSWORD

- **T.ACCESS_RIGHTS. Container data access:**
  An attacker uses malicious software to gain access or modify data stored in the mounted container file system or container configuration data stored outside the container.

  Related Assets: D.MAC_POLICY, D. USER_DATA
  Coverage: FS.DATA_ISOLATION
  Related Assumptions: A.SECURE_BOOT, A.TRUSTED_BOOT, A.RUNTIME_INTEGRITY, A.SECURE_FS, A.SECURE_FS_KEYS, A.MAC_POLICY_UPDATE

- **T.CREDENTIAL. Container credentials access:**
  An attacker, e.g. through memory analysis or an application in the usual Android environment, succeeds in obtaining the container credentials and in accessing container data.

  Related Assets: D.CONTAINER_PASSWORD, D. USER_DATA
  Coverage: FS.USER_AUTH
  Related Assumptions: A.RUNTIME_INTEGRITY, A.APPLICATION_SANDBOX

- **T.LEAKAGE. Information leakage from container applications:**
  A container application that is not explicitly permitted by MDM data sharing policy (Section 2.1.1) succeeds in exchanging information with software outside the container through services, file system or run-time memory.

  Related Assets: D. USER_DATA, D.MAC_POLICY
  Coverage: FS.DATA_ISOLATION, FS.APPLICATIONS_INSTALL, FS.EXT_INTERFACES
  Related Assumptions: A.RUNTIME_INTEGRITY, A.APPLICATION_SANDBOX

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

5 Security Functions

Product security functions are:

- **FS.DATA_ISOLATION**

  Applications can only share data or interact with applications running in the same container through standard Android mechanisms (intents, content providers etc). Applications in the Personal world cannot share data or interact with applications running inside a KNOX Container. The Domain Separation component of the TOE provides this isolation by enabling the creation and management of KNOX Containers. SE for Android enforces the isolation.

  The Data Synchronisation component of the TOE allows data from selected applications (Calendar, Contacts etc.) to be synchronised between the Personal world and a KNOX Container. Sharing of data between these applications is fully managed using MDM policies, authorized by the administrator.

  Third party applications cannot use Data Synchronisation, they are restricted to applications that Samsung has permitted.

- **FS.APPLICATIONS_INSTALL**

  Applications which are permitted for installation into the KNOX container are defined by the MDM Server.
  The MDM policy may be configured in the following ways:
  - Allow/Disallow any application installation in the container
  - Whitelist of applications – the user cannot install any additional applications not on the whitelist
  - Blacklist of applications – the user cannot install any applications that are on the blacklist
  - Enable or disable KNOX App Store
  - Enable or disable Google Play®
  - Application push by IT Admin (subject to white/blacklist)

  Application installation is handled by the Android Package Management system, not part of the TOE, as stated in A.APPLICATION_INTEGRITY.

  The Application Installation policy from the MDM controls which applications are available in the Container.

- **FS.USER_AUTH**

  The Container Authentication component provides user authentication for KNOX container access, using Samsung's Key Management Module in Trustzone. Additionally in the KNOX Workspace, the user is required to re-authenticate when changing its authentication method or credentials.

- **FS.USER_AUTH_LOCK**
  The KNOX Container can be locked by:
  - User pressing the container lock button from the notification bar or
  - On a configurable timeout/action:
    - KNOX Container inactive for a configurable time or
    - Screen timeout or

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.

- Device restart

When locked, the Container Authentication component of the TOE displays a Keyguard to prevent any Container application from being visible to the user .

When the KNOX container is locked, notifications are still displayed if permitted by MDM Policy. The user cannot perform any related actions based upon these container notifications (e.g. they cannot read emails, calendar appointments, or text messages) without first authenticating to the Container Authentication component of the TOE.

The Container inactivity timeout is independent from the device lock screen timeout.

- **FS.USER_AUTH_FAIL**
  The Container Authentication component of the TOE maintains, for each user, the number of failed logins since the last successful login. When the maximum number of incorrect logins is reached, the KNOX Workspace can either perform a full wipe of data protected by KNOX Workspace (i.e. data inside the container) or it lock the container until unlocked by the administrator.

  The Administrator can set the number of unsuccessful authentication attempts allowed to a value between 1 and 100 (default=10), via MDM Policy.

- **FS.CONTAINER_STATE**
  The Domain Separation component of the TOE maintains the state machine of the Container. The state of the container cannot be changed directly by other TOE components but is triggered to change by events (for instance locking by administrator). The state machine enforces how the state can change in response to these events and what actions should be performed.

  The Container state controls subsequent access to the container; e.g. locking the container when a failure occurs, locking by administrator and forcing a password reset.

- **FS.EXT_INTERFACES**
  External data interfaces can be controlled by MDM policies and are in some cases disabled entirely for applications in the Container. These restrictions are controlled via the Device Management component of the TOE and enforced by the Modified Android Services component of the TOE.

  USB MTP and USB Mass Storage are not available to the user or applications in the Container.

  External SD Card access can be disabled entirely or a whitelist of applications that can access the SD Card can be defined by MDM Policy.

  Bluetooth connectivity can be restricted by MDM policy at the device level. Applications are limited to accessing CAC Smartcards via Samsung's SmartCard Framework only if Bluetooth Secure Mode is enabled. The MDM policy can enable/disable Bluetooth Secure Mode.

  NFC can be enabled/disabled by MDM policy at the device level. Applications in the container cannot access NFC.

Copyright © 2015 Samsung Electronics Co, Ltd. All Rights Reserved.