Samsung Electronics Co., Ltd. Samsung Galaxy Note 4 Android 5 VPN Client (IVPNCPP14) Security Target

Version 1.1
2015/03/19

Prepared for:

Samsung Electronics Co., Ltd.

416 Maetan-3dong, Yeongtong-gu, Suwon-si, Gyeonggi-do, 443-742 Korea

Prepared By:

Gossamer Laboratories

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE consists of the Samsung Galaxy Note 4 Android 5 VPN Client provided by Samsung Electronics Co., Ltd.. The TOE is being evaluated as an IPsec VPN Client.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
  - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
  - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
  - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
  - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- The IVPNCPP uses an additional convention – the 'case' – which defines parts of an SFR that apply only when corresponding selections are made or some other identified conditions exist. Only the applicable cases are identified in this ST and they are identified using **bold** text.
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – Samsung Electronics Co., Ltd. Samsung Galaxy Note 4 Android 5 VPN Client (IVPNCPP14) Security Target

ST Version – Version 1.1

ST Date – 2015/03/19

1.2 TOE Reference

TOE Identification – Samsung Electronics Co., Ltd. Samsung Galaxy Note 4 Android 5 VPN Client.

TOE Developer – Samsung Electronics Co., Ltd.

Evaluation Sponsor – Samsung Electronics Co., Ltd.

1.3 TOE Overview

The Target of Evaluation (TOE) is Samsung Galaxy Note 4 Android 5 VPN Client. This ST focuses on the IPSEC VPN capabilities of the TOE. The IPSec VPN allows users the ability to have confidentiality, integrity, and protection of data in transit, even though it traverses a public network.

1.4 TOE Description

The TOE is a VPN client that runs on a mobile operating system based on Android 5.0.1 with modifications made to increase the level of security provided to end users and enterprises. The TOE is intended to be used as part of an enterprise messaging solution providing mobile staff with enterprise connectivity.

The model numbers of the mobile devices used during the evaluation are as follows:

Device Name	Model Number	Android Version	Kernel Version	Build Number
Galaxy Note 4 (Qualcomm)	SM-N910F	5.0.1	3.10.40	LRX22C

The TOE platform includes a Common Criteria mode (or "CC mode") that an administrator can invoke through the use of an MDM or through the installation and use of the administrative application, CCMode.apk (see the Guidance for instructions to obtain the application). The TOE platform must be configured as follows in order for an administrator to transition the TOE platform to CC mode.

- Require a screen lock password (swipe, PIN, pattern, or facial recognition screen locks are not allowed).
- The maximum password failure retry policy should be less than or equal to ten.
- Device encryption must be enabled.
- SDCard encryption must be enabled.
- Revocation checking must be enabled.

When CC mode has been enabled on the TOE platform, the TOE behavior is affected. The TOE behaves as follows.

- The TOE restricts the available VPN configurations to those evaluated as part of this evaluation.
- The TOE restricts the use of IKEv2/IPsec cipher suites to only those conformant with the requirements of the IVPNCPP14.

1.4.1 TOE Architecture

The TOE combines with a Mobile Device Management solution that enables the enterprise to watch, control and administer all deployed mobile devices, across multiple mobile service providers as well as facilitate secure communications through a VPN. This partnership provides a secure mobile environment that can be managed and controlled by the environment and reduce the risks that can be introduced through a Bring-Your-Own-Device (BYOD) model.

Data on the TOE is protected through the implementation of Samsung On-Device Encryption (ODE) which utilizes CAVP certified cryptographic algorithms to encrypt device and SD card storage. This functionality is combined with a number of on-device policies including local wipe, remote wipe, password complexity, automatic lock and privileged access to security configurations to prevent unauthorized access to the device and stored data.

The Samsung Enterprise Software Development Kit (SDK) builds on top of the existing Android security model by expanding the current set of security configuration of options to more than 390 configurable policies and including additional security functionality such as application whitelisting and blacklisting.

1.4.1.1 Physical Boundaries

The TOE is a multi-user operating system based on Android (5.0.1) that incorporates the Samsung Enterprise SDK. The TOE does not include the user applications that run on top of the operating system, but does include controls that limit application behavior. The method of use for the TOE is as a mobile messaging and VPN device for use within an enterprise environment where the configuration of the device is managed through a compliant device management solution.

The TOE communicates and interacts with 802.11-2012 Access Points and cellular networks to establish network connectivity.

This evaluation does not include the underlying hardware and firmware or the device management application that is implemented on the device.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by the Samsung Galaxy Note 4 Android 5 VPN Client:

- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF
- Trusted path/channels

1.4.1.2.1 Cryptographic support

The IPsec implementation is the primary function of the TOE. IPSec is used by the TOE to protect communication between itself and a VPN Gateway over an unprotected network. With the exception of the IPsec implementation, the TOE relies upon its underlying platform (evaluated against the Protection Profile For Mobile Device Fundamentals) for the cryptographic services specified in this Security Target.

1.4.1.2.2 User data protection

The TOE ensures that residual information is protected from potential reuse in accessible objects such as network packets.

1.4.1.2.3 Identification and authentication

The TOE provides the ability to use, store, and protect X.509 certificates and pre-shared keys that are used for IPsec Virtual Private Network (VPN) connections.

1.4.1.2.4 Security management

The TOE provides all the interfaces necessary to manage the security functions identified throughout this Security Target. In particular, the IPsec VPN is fully configurable by a combination of functions provided directly by The TOE and those available to the associated VPN gateway.

1.4.1.2.5 Protection of the TSF

The TOE relies upon its underlying platform to perform self-tests that cover the TOE as well as the functions necessary to securely update the TOE.

1.4.1.2.6 Trusted path/channels

The TOE acts as a VPN client using IPsec to established secure channels to corresponding VPN gateways.

1.4.2 TOE Documentation

Samsung VPN Client on Galaxy Devices Guidance documentation, Version 2.1, February 13, 2015.

Samsung VPN Client on Galaxy Devices VPN User Guidance Documentation, Version 2.1, February 13, 2015.

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
  - Part 3 Conformant
- Protection Profile for IPsec Virtual Private Network (VPN) Clients, Version 1.4, 21 October 2013 (IVPNCPP14)
- Package Claims:
  - Assurance Level: EAL 1-conformant

2.1 Conformance Rationale

The ST conforms to the IVPNCPP14. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the IVPNCPP14 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The IVPNCPP14 offers additional information about the identified security objectives, but that has not been reproduced here and the IVPNCPP14 should be consulted if there is interest in that material.

In general, the IVPNCPP14 has defined Security Objectives appropriate for IPsec VPN Client and as such are applicable to the Samsung Galaxy Devices with Qualcomm Snapdragon Processors TOE.

3.1 Security Objectives for the Operational Environment

- **OE.NO_TOE_BYPASS** Information cannot flow onto the network to which the VPN client's host is connected without passing through the TOE.
- **OE.PHYSICAL** Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the operational environment.
- **OE.TRUSTED_CONFIG** Personnel configuring the TOE and its operational environment will follow the applicable security configuration guidance.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the IVPNCPP14. The IVPNCPP14 defines the following extended SFRs and SARs and since they are not redefined in this ST the IVPNCPP14 should be consulted for more information in regard to those CC extensions.

- FCS_CKM_EXT.2: Cryptographic Key Storage
- FCS_CKM_EXT.4: Cryptographic Key Zeroization
- FCS_IPSEC_EXT.1: Extended: Internet Protocol Security (IPsec) Communications
- FCS_RBG_EXT.1: Extended: Cryptographic operation (Random Bit Generation)
- FIA_X509_EXT.1: Extended: X.509 Certificate Validation
- FIA_X509_EXT.2: Extended: X.509 Certificate Use and Management
- FPT_TST_EXT.1: Extended: TSF Self Test
- FPT_TUD_EXT.1: Extended: Trusted Update
- FIA_PSK_EXT.1: Extended: Pre-Shared Key Composition

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the IVPNCPP14. The refinements and operations already performed in the IVPNCPP14 are not identified (e.g., highlighted) here, rather the requirements have been copied from the IVPNCPP14 and any residual operations have been completed herein. Of particular note, the IVPNCPP14 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the IVPNCPP14 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the IVPNCPP14 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The IVPNCPP14 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Samsung Galaxy Devices with Qualcomm Snapdragon Processors TOE.

Requirement Class	Requirement Component
FCS: Cryptographic support	FCS_CKM.1(1): Cryptographic Key Generation (Asymmetric Keys)
	FCS_CKM.1(2): Cryptographic Key Generation (for asymmetric keys - IKE)
	FCS_CKM_EXT.2: Cryptographic Key Storage
	FCS_CKM_EXT.4: Cryptographic Key Zeroization
	FCS_COP.1(1): Cryptographic Operation (Data Encryption/Decryption)
	FCS_COP.1(2): Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3): Cryptographic Operation (Cryptographic Hashing)
	FCS_COP.1(4): Cryptographic Operation (Keyed-Hash Message Authentication)
	FCS_IPSEC_EXT.1: Extended: Internet Protocol Security (IPsec) Communications
	FCS_RBG_EXT.1: Extended: Cryptographic operation (Random Bit Generation)
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection
FIA: Identification and authentication	FIA_PSK_EXT.1: Extended: Pre-Shared Key Composition
	FIA_X509_EXT.1: Extended: X.509 Certificate Validation
	FIA_X509_EXT.2: Extended: X.509 Certificate Use and Management
FMT: Security management	FMT_SMF.1(1): Specification of Management Functions
	FMT_SMF.1(2): Specification of Management Functions
FPT: Protection of the TSF	FPT_TST_EXT.1: Extended: TSF Self Test
	FPT_TUD_EXT.1: Extended: Trusted Update
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel

Table 1 TOE Security Functional Components

5.1.1 Cryptographic support (FCS)

5.1.1.1 Cryptographic Key Generation (Asymmetric Keys) (FCS_CKM.1(1))

FCS_CKM.1(1).1

Refinement: The [***TOE Platform***] shall generate asymmetric cryptographic keys used for key establishment in accordance with
- NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' for finite field-based key establishment schemes;
- NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' for elliptic curve-based key establishment schemes and implementing 'NIST curves' P-256, P-384 and [***P-521***] (as defined in FIPS PUB 186-4, 'Digital Signature Standard')
- [***NIST Special Publication 800-56B, 'Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography' for RSA-based key establishment schemes***]
and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits. See NIST Special Publication 800-57, 'Recommendation for Key Management' for information about equivalent key strengths.

5.1.1.2 Cryptographic Key Generation (for asymmetric keys - IKE) (FCS_CKM.1(2))

FCS_CKM.1(2).1

Refinement: The [***TOE Platform***] shall generate asymmetric cryptographic keys used for IKE peer authentication in accordance with a: [***FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4 for ECDSA schemes and implementing 'NIST curves' P-256, P-384 and [P-521]; ANSI X9.31-1998, Appendix A.2.4 Using AES for RSA schemes***]
and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.1.1.3 Cryptographic Key Storage (FCS_CKM_EXT.2)

FCS_CKM_EXT.2.1

The [***TOE Platform***] shall store persistent secrets and private keys when not in use in platform-provided key storage.

5.1.1.4 Cryptographic Key Zeroization (FCS_CKM_EXT.4)

FCS_CKM_EXT.4.1

Refinement: The [***TOE Platform***] shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

5.1.1.5 Cryptographic Operation (Data Encryption/Decryption) (FCS_COP.1(1))

FCS_COP.1(1).1

Refinement: The [***TOE Platform***] shall perform encryption and decryption in accordance with a specified cryptographic algorithm AES operating in GCM and CBC mode with cryptographic key sizes 128-bits and 256-bits that meets the following:
- FIPS PUB 197, 'Advanced Encryption Standard (AES)'
- NIST SP 800-38D, NIST SP 800-38A.

5.1.1.6 Cryptographic Operation (for cryptographic signature) (FCS_COP.1(2))

FCS_COP.1(2).1

Refinement: The [***TOE Platform***] shall perform cryptographic signature services in accordance with a specified cryptographic algorithm:

- [***FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.3 for RSA scheme, FIPS PUB 186-4, 'Digital Signature Standard', Appendix B.4 for ECDSA schemes and implementing 'NIST curves' P-256, P-384 and [P-521]***]
and cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.1.1.7 Cryptographic Operation (Cryptographic Hashing) (FCS_COP.1(3))

FCS_COP.1(3).1
Refinement: The [***TOE Platform***] shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-384, SHA-512***] and message digest sizes [***160, 256, 384, 512***] bits that meet the following: FIPS Pub 180-4, 'Secure Hash Standard.'

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

FCS_COP.1(4).1
Refinement: The [***TOE Platform***] shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC- [***SHA-1, SHA-256, SHA-384, SHA-512***], -key size [**any key sizes**], and message digest size of [***160, 256, 384, 512***] bits that meet the following: FIPS PUB 198-1, 'The Keyed-Hash Message Authentication Code', and FIPS PUB 180-4, 'Secure Hash Standard'.

5.1.1.9 Extended: Internet Protocol Security (IPsec) Communications (FCS_IPSEC_EXT.1)

FCS_IPSEC_EXT.1.1
The [***TOE***] shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2
The [***TOE***] shall implement [***tunnel mode***].

FCS_IPSEC_EXT.1.3
The [***TOE***] shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

FCS_IPSEC_EXT.1.4
The [***TOE***] shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms AES-GCM-128, AES-GCM-256 as specified in RFC 4106, [***AES-CBC-128 (specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC, AES-CBC-256 (specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC***].

FCS_IPSEC_EXT.1.5
The [***TOE***] shall implement the protocol: [***IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and [no other RFCs for hash functions]***].

FCS_IPSEC_EXT.1.6
The [***TOE***] shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and [***AES-GCM-128, AES-GCM-256 as specified in RFC 5282***].

FCS_IPSEC_EXT.1.7
The [***TOE***] shall ensure that IKEv1 Phase 1 exchanges use only main mode

FCS_IPSEC_EXT.1.8
The [***TOE***] shall ensure that [***IKEv2 SA lifetimes can be configured by [VPN Gateway] based on [length of time, where the time values can be limited to: 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs]***].

FCS_IPSEC_EXT.1.9
The [***TOE***] shall generate the secret value x used in the IKE Diffie-Hellman key exchange ('x' in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [**(224, 256, or 384)**] bits .

FCS_IPSEC_EXT.1.10

The [***TOE***] shall generate nonces used in IKE exchanges in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^[**(112, 128, or 192)**] .

FCS_IPSEC_EXT.1.11

The [***TOE***] shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), 19 (256-bit Random ECP), and [***5 (1536-bit MODP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP)***].

FCS_IPSEC_EXT.1.12

The [***TOE***] shall ensure that all IKE protocols perform peer authentication using a [***RSA, ECDSA***] that use X.509v3 certificates that conform to RFC 4945 and [***Pre-Shared Keys***].

FCS_IPSEC_EXT.1.13

The TSF shall support peer identifiers of the following types: [***IP address, Fully Qualified Domain Name (FQDN), Distinguished Name (DN)***] and [***no other reference identifier type***].

FCS_IPSEC_EXT.1.14

The [***VPN Gateway***] shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 IKE_SA***] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 CHILD_SA***] connection.

5.1.1.10 Extended: Cryptographic operation (Random Bit Generation) (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1

The [***TOE Platform***] shall perform all deterministic random bit generation services in accordance with [***NIST Special Publication 800-90A using [CTR _DRBG(AES)]***].

FCS_RBG_EXT.1.2

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

5.1.2 User data protection (FDP)

5.1.2.1 Full Residual Information Protection (FDP_RIP.2)

FDP_RIP.2.1

The [***TOE***] shall enforce that any previous information content of a resource is made unavailable upon the [***allocation of the resource to***] all objects.

5.1.3 Identification and authentication (FIA)

5.1.3.1 Extended: Pre-Shared Key Composition (FIA_PSK_EXT.1)

FIA_PSK_EXT.1.1

The [***TOE***] shall be able to use pre-shared keys for IPsec.

FIA_PSK_EXT.1.2

The [***TOE***] shall be able to accept text-based pre-shared keys that:
- are 22 characters and [***[up to 64 characters]***];
- composed of any combination of upper and lower case letters, numbers, and special characters (that include: '!', '@', '#', '$', '%', '^', '&', '*', '(', and ')').

FIA_PSK_EXT.1.3

The [***TOE***] shall [***be able to [accept] bit-based pre-shared keys***].

5.1.3.2 Extended: X.509 Certificate Validation (FIA_X509_EXT.1)

FIA_X509_EXT.1.1

The [***TOE***] shall validate certificates in accordance with the following rules:
- Perform RFC 5280 certificate validation and certificate path validation.
- Validate the revocation status of the certificate using [***the Online Certificate Status Protocol (OCSP) as specified in RFC 2560***].
- Validate the certificate path by ensuring the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates.
- Validate the extendedKeyUsage field according to the following rules:

o Certificates used for [***no other purpose***] shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3).

FIA_X509_EXT.1.2

The [***TOE***] shall only treat a certificate as a CA certificate if the following is met: the basicConstraints extension is present and the CA flag is set to TRUE.

5.1.3.3 Extended: X.509 Certificate Use and Management (FIA_X509_EXT.2)

FIA_X509_EXT.2.1

The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec exchanges, and [***no additional uses***].

FIA_X509_EXT.2.2

When a connection to determine the validity of a certificate cannot be established, the [***TOE***] shall [***not accept the certificate***].

FIA_X509_EXT.2.3

The [***TOE***] shall not establish an SA if a certificate or certificate path is deemed invalid.

5.1.4 Security management (FMT)

5.1.4.1 Specification of Management Functions (FMT_SMF.1(1))

FMT_SMF.1(1).1

The TOE shall be capable of performing the following management functions:
- Specify VPN gateways to use for connections,
- Specify client credentials to be used for connections,
- [**no additional management functions**].

5.1.4.2 Specification of Management Functions (FMT_SMF.1(2))

FMT_SMF.1(2).1

The [***TOE, VPN Gateway, or TOE Platform***] shall be capable of performing the following management functions:
- Configuration of IKE protocol version(s) used,
- Configure IKE authentication techniques used,
- Configure the cryptoperiod for the established session keys. The unit of measure for configuring the cryptoperiod shall be no greater than an hour,
- Configure the reference identifier for the peer
- Configure certificate revocation check,
- Specify the algorithm suites that may be proposed and accepted during the IPsec exchanges,
- load X.509v3 certificates used by the security functions in this PP,
- ability to update the TOE, and to verify the updates,
- ability to configure all security management functions identified in other sections of this PP,
- [***[no additional management functions]***].

Application Note: *For TOEs that support only IP address and FQDN identifier types, configuration of the reference identifier may be the same as configuration of the peer's name for the purposes of connection.*

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Extended: TSF Self Test (FPT_TST_EXT.1)

FPT_TST_EXT.1.1
The [***TOE Platform***] shall run a suite of self -tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2
The [***TOE, TOE Platform***] shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the [**cryptographic signature and hash for integrity**].

5.1.5.2 Extended: Trusted Update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1
The [***TOE Platform***] shall provide authorized administrators the ability to query the current version of the TOE firmware/software

FPT_TUD_EXT.1.2
The [***TOE Platform***] shall provide authorized administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3
The [***TOE Platform***] shall provide a means to verify firmware/software updates to the TOE using a digital signature mechanism and [***no other functions***] prior to installing those updates.

5.1.6 Trusted path/channels (FTP)

5.1.6.1 Inter-TSF trusted channel (FTP_ITC.1)

FTP_ITC.1.1
Refinement: The [***TOE***] shall use IPsec to provide a trusted communication channel between itself and a VPN Gateway that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2
The [***TOE***] shall permit the TSF to initiate communication via the trusted channel.

FTP_ITC.1.3
The [***TOE***] shall initiate communication via the trusted channel for all traffic traversing that connection.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the EAL 1 components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic functional specification
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures

ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM coverage
ATE: Tests	ATE_IND.1: Independent testing - conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability survey

Table 2 EAL 1 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic functional specification (ADV_FSP.1)

ADV_FSP.1.1d

The developer shall provide a functional specification.

ADV_FSP.1.2d

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

ADV_FSP.1.1c

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

ADV_FSP.1.2c

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

ADV_FSP.1.3c

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

ADV_FSP.1.4c

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

ADV_FSP.1.1e

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

ADV_FSP.1.2e

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

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1d

The developer shall provide operational user guidance.

AGD_OPE.1.1c

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

AGD_OPE.1.2c

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

AGD_OPE.1.3c

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

AGD_OPE.1.4c

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

AGD_OPE.1.5c

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

AGD_OPE.1.6c

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

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

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

5.2.2.2 Preparative procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1c

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

AGD_PRE.1.2c

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

AGD_PRE.1.1e

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

AGD_PRE.1.2e

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

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d

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

ALC_CMC.1.1c

The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e

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

5.2.3.2 TOE CM coverage (ALC_CMS.1)

ALC_CMS.1.1d

The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c

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

ALC_CMS.1.2c

The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e

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

5.2.4 Tests (ATE)

5.2.4.1 Independent testing - conformance (ATE_IND.1)

ATE_IND.1.1d

The developer shall provide the TOE for testing.

ATE_IND.1.1c

The TOE shall be suitable for testing.

ATE_IND.1.1e

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

ATE_IND.1.2e

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

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability survey (AVA_VAN.1)

AVA_VAN.1.1d

The developer shall provide the TOE for testing.

AVA_VAN.1.1c

The TOE shall be suitable for testing.

AVA_VAN.1.1e

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

AVA_VAN.1.2e

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

AVA_VAN.1.3e

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

6. TOE Summary Specification

This chapter describes the security functions:

- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF
- Trusted path/channels

6.1 Cryptographic support

The TOE implements the IPsec protocol as specified in RFC 4301; however, the TOE presents as few configuration options as possible to the User in order to minimize the possibility of misconfiguration and relies upon the Gateway to enforce organizational policies (for things like the specific cipher suites and selection of traffic to protect). For this reason, the TOE does not support editing of its SPD entries. The TOE will insert a PROTECT rule to IPsec encrypt and send all TOE traffic to the VPN GW (as the TOE ignores the IKEv2 Traffic Selector negotiated between the client and gateway and always sends all traffic).

The TOE routes all packets through the kernel's IPsec interface (ipsec0) when the VPN is active. The kernel compares packets routed through this interface to the SPDs configured for the VPN to determine whether or PROTECT, BYPASS, or DISCARD each packet. The vendor designed the TOE's VPN, when operating in CC Mode, to allow no configuration and to always force all traffic through the VPN. The TOE ignores any IKEv2 traffic selector negotiations with the VPN GW and will always create an SPD PROTECT rule that matches all traffic. Thus, the kernel will match all packets, subsequently encrypt those packets, and finally forward them to the VPN Gateway. The TOE supports tunnel mode for its IPsec connections. The TOE provides IKEv2 key establishment as part of its IPsec implementation. The IKEv2 implementation is conformant with RFCs 5996 and 4307 and supports NAT traversal.

The TOE Platform implements RFC 4106 conformant AES-GCM-128 and AES-GCM-256, and RFC 3602 conformant AES-CBC-128, and AES-CBC-256 as encryption algorithms. The TOE Platform also implements SHA-1, SHA-256, SHA-384, and SHA-512 in addition to HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 as integrity/authentication algorithms as well as Diffie-Hellman Groups 5, 14, 19, 20 and 24. The encrypted payload for IKEv2 uses AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and AES-GCM-128 and AES-GCM-256 as specified in RFC 5282. The TOE relies upon the VPN Gateway to ensure that by default the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the IKEv2 CHILD_SA connection. The TOE utilizes the cryptographic algorithm implementation of the TOE Platform by linking against the native OpenSSL cryptographic library. In this way, then TOE can invoke the cryptographic operations provided by the TOE Platform (including the key establishment, key generation, encryption/decryption, random bit generation, digital signatures, hashing, and keyed hashing). The TOE Platform's OpenSSL cryptographic library ensures association/use of lower-level cryptographic algorithms as part of higher ones (for example, use of SHA hashing as part of RSA and ECDSA signature generation and verification) while the TOE itself ensure association/use of OpenSSL's cryptographic algorithms as part of the IPsec and IKEv2 protocols.

An administrator can configure the VPN Gateway to limit SA lifetimes based on length of time to values that include 24 hours for IKE SAs and 8 hours for IPsec SAs. The TOE includes hardcoded limits of 10 hours for an IKE SA and 3 hours for an IPsec SA. The TOE and VPN Gateway will rekey their IKE and IPsec SAs after the shorter of either 10 hours and 3 hours respectively (the TOE's fixed lifetimes) or the administrator specified values that are configured on the VPN Gateway.

The TOE generates the secret value x used in the IKEv2 Diffie-Hellman key exchange ('x' in g^x mod p) using the FIPS validated RBG specified in FCS_RBG_EXT.1 and having possible lengths of 224, 256, or 384 bits. When a random number is needed for a nonce, the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^{112}, 2^{128}, or 2^{192}.

The TOE implements peer authentication using RSA certificates or ECDSA certificates that conform to RFC 4945 and FIPS 186-4, or pre-shared keys. If certificates are used, the TOE ensures that the IP Address[1] or Fully Qualified Distinguished Name (FQDN) contained in a certificate matches the expected IP address or FQDN for the entity attempting to establish a connection and ensures that the certificate has not been revoked (using the Online Certificate Status Protocol [OCSP] in accordance with RFC 2560).

Pre-shared keys can include any letter from a-z, A-Z, the numbers 0 – 9, and any special character located above the numbers on the keyboard. The specific length of 22 characters required by the VPNCPP14 is supported by the TOE. The TOE does not perform any processing on pre-shared keys. The TOE simply uses the pre-shared key that was entered by the administrator.

The following table describes the keys and secrets utilized by the TOE.

Key Name:	Origin/Purpose:	Storage Location:	Cleared upon:	Type of clearing:
DH Group Parameters (DH 14, 19, 5, 24, 20)	RFC defined parameters hardcoded into the TSF/used in the ephemeral Diffie-Hellman key exchange	Executable Image in Flash	N/A – Public values	N/A – Public values
IPsec Pre-Shared Keys	Entered by the user/used for peer authentication	TOE Platform Keystore	On wipe function	Crypto erase
User IPsec X.509v3 Certs (RSA/ECDSA)	Entered by the user/used for client authentication	TOE Platform Keystore	On wipe function	Crypto erase
CA IPsec X.509v3 Certs (RSA/ECDSA)	Entered by the user/used to authenticate the gateway	TOE Platform Keystore	N/A – public values	N/A = public values
IKEv2 IKE_SA Enc Keys (AES CBC or GCM)	Generated as part of IKEv2 IKE_SA establishment/used to encipher/decipher traffic	Memory/RAM	No longer needed by trusted channel	Zero overwrite
IKEv2 IKE_SA MAC Keys (HMAC-SHA)	Generated as part of IKEv2 IKE_SA establishment/used for traffic integrity	Memory/RAM	No longer needed by trusted channel	Zero overwrite
IKEv2 CHILD_SA Enc Keys (AES CBC or GCM)	Generated as part of IKEv2 CHILD_SA establishment/ used to encipher/decipher traffic	Memory/RAM	No longer needed by trusted channel	Zero overwrite
IKEv2 CHILD_SA MAC Keys (HMAC-SHA)	Generated as part of IKEv2 CHILD_SA establishment/ used for traffic integrity	Memory/RAM	No longer needed by trusted channel	Zero overwrite

Table 3 TOE Keys and Secrets

The TOE supports a number of different Diffie-Hellman (DH) groups for use in SA negotiation including DH Groups 5 (1536-bit MODP), 14 (2048-bit MODP), 19 (256-bit Random ECP), 20 (384-bit Random ECP), and 24 (2048-bit MODP with 256-bit POS). The TOE selects the DH group by selecting the largest group configured by an administrator that is offered by the VPN gateway.

During the Peer Authentication stage of IPsec, the TOE Platform will verify the authenticity of the VPN gateway's X.509v3 certificate by validating the certificate, validating the certificate path, validating the certificates revocation status using OCSP, validating that the certificate path terminates in a trusted CA certificate, and validating that the CA certificate has the basicConstraints extension present and the CA flag set to true.

The TOE relies upon the VPN Gateway to ensure that the cryptographic algorithms and key sizes negotiated during the IKEv2 negotiation ensure that the security strength of the IKE_SA are greater than or equal to that of the CHILD_SA.

[1] The TOE only accepts IPv4 addresses when matching a certificate IP address

The Cryptographic support function is designed to satisfy the following security functional requirements:

- FCS_CKM.1(1): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_CKM.1(2): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_CKM_EXT.2: This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_CKM_EXT.4: This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_COP.1(1): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_COP.1(2): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_COP.1(3): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_COP.1(4): This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FCS_IPSEC_EXT.1: The TOE implements IPsec in accordance with FCS_IPSEC_EXT.1 as described above.
- FCS_RBG_EXT.1: This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).

6.2 User data protection

The TOE has been designed to ensure that no residual information exists in network packets. When the TOE allocates a new buffer for either an incoming or outgoing a network packet, the new packet data will be used to overwrite any previous data in the buffer. If an allocated buffer exceeds the size of the packet, and additional space will be overwritten (padded) with zeros before the packet is forwarded (to the external network or delivered to the appropriate, internal application).

The User data protection function is designed to satisfy the following security functional requirements:

- FDP_RIP.2: The TOE ensures that previous information contents of resources used for new objects are not discernible in any new object, such as network packets, as described above.

6.3 Identification and authentication

The TOE supports the use of pre-shared keys (the TOE allows 22 to 64 character PSKs) for IPsec VPNs. The usage of pre-shared keys within the IPsec implementation is described in Section 6.1.

The TOE can also use X.509 certificates for authentication. The TOE requires that for each VPN connection, the user specify the client certificate the TOE will use (the user must have previously loaded such a certificate into the keystore) and specify the CA certificate to which the server's certificate must chain. The TOE thus uses the specified certificate when attempting to establish that VPN connection. The TOE validates authentication certificates (including the full path) and checks their revocation station using OCSP. The TOE processes a VPN connection to a server by first comparing the Identification (ID) Payload received from the server against the certificate sent by the server, and if the DN of the certificate does not match the ID, then the TOE does not establish the connection. Assuming the server's certificate matches the ID, the TOE then validates that it can construct a certificate path from the server's certificate through any intermediary CAs to the CA certificate specified by the user in the VPN configuration. If the TOE can successfully build the certificate path, then the TOE will next check the validity of the certificates (e.g., checking its validity dates and that the CA flag is present in the basic constraints section for all CA certs). Assuming the certificates are valid, the TOE finally checks the revocation status of all certificates (starting the server's certificate and working up the chain). The TOE will reject any certificate for which it cannot determine the validity and reject the connection attempt. Section 6.1 describes how the TOE uses certificates in its IPsec architecture.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_PSK_EXT.1: The TOE supports the use of pre-shared IPsec keys used to create IPsec connections. The pre-shared keys can be composed as required and as described above.
- FIA_X509_EXT.1: This requirement is satisfied by the TOE, which performs all needed certificate validation (including the certificate, its path, and its revocation status).
- FIA_X509_EXT.2: The TOE uses X.509v3 certificates for authentication in IPsec exchanges and rejects any certificates that cannot be validated as described above.

6.4 Security management

The following security management functions are provided directly by the TOE, implemented in the VPN gateway and/or implemented by the TOE platform as indicated below:

- The TOE provides functions allowing the user to select VPN gateway and credentials used to connect to those gateways.
- The VPN Gateway (acting as an administrator) to which the TOE connects selects IKE protocols and authentication techniques.
- The VPN Gateway (acting as an administrator) to which the TOE connects selects the algorithms to be used in IPsec exchanges.
- The VPN gateway provides the ability to configure the cryptoperiod for the established IPsec session keys (including the ability to conjure crypto periods less than an hour in duration).
- The TOE platform provides the ability to configure certificate revocation checking.
- The TOE platform provides the ability to load X.509v3 certificates used for VPN connections using IPsec.
- The TOE provides users the ability to specify an X.509v3 certificate (previously loaded into the TOE Platform's key store) for the TOE to use to authenticate to the VPN gateway during IPsec peer authentication as well as an X.509v3 certificate to use as the CA certificate. The TOE alternatively provides users the ability to enter a Pre-Shared Key to be used in lieu of an X.509v3 certificate during IPsec peer authentication.
- The TOE platform provides the ability to update the TOE, and to verify the updates.
- The TOE platform provides the functions necessary for all other security functions identified in this Security Target,

The Security management function is designed to satisfy the following security functional requirements:

- FMT_SMF.1(1): The TOE provides the functions necessary to specify VPN gateways and the corresponding credentials used to establish VPN connections as described above.
- FMT_SMF.1(2): A combination of the TOE platform, the TOE, and the VPN gateway provide the functions necessary to manage the security functions described in this security target as described above.

6.5 Protection of the TSF

The TOE is a system service in the context of its platform (i.e., the TOE platform). As such, it is subject to the self-tests and trusted update features of the TOE Platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).

The TOE cryptographically verifies the integrity of its executable code (the files /system/bin/charon, /system/lib/libcharon.so, /system/lib/libstrongswan.so, and /system/lib/libhydra.so) upon load and prior to execution (the TOE Platform loads the executable whenever a VPN connection is requested/attempted) by requesting checking from the Security Manager daemon (part of the TOE platform). The Security Manager verifies (using an embedded RSA-2048 public key) a PKCS#1 (using SHA-256) signature of the TOE executable code to ensure that it has not been modified or corrupted. If the Security Manager's check of the TOE fails, the TOE will fail to establish a VPN connection.

The TOE platform performs known answer power on self-tests (POST) on its cryptographic algorithms to ensure that they are functioning correctly. The kernel itself performs known answer tests on its cryptographic algorithms to ensure they are working correctly and the Samsung security manager service invokes self-test of the OpenSSL module at start to ensure that those cryptographic algorithms are working correctly..

Should the TOE platform fail its power-up tests fails, the TOE platform will lock itself, preventing login.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_TST_EXT.1: This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).
- FPT_TUD_EXT.1: This requirement is satisfied by the TOE platform (evaluated against the Protection Profile For Mobile Device Fundamentals, Version 1.0, 21 October 2013).

6.6 Trusted path/channels

See section 6.1 for a description of how the TOE can establish IPsec VPN connections with configured VPN gateways. The resulting VPNs ensure that both ends of the channel are authenticated and the channel protects data from disclosure and modification.

The Trusted path/channels function is designed to satisfy the following security functional requirements:

- FTP_ITC.1: The TOE uses IPsec to provide a protected communication channel between itself and an IPsec VPN gateway. The channel provides assurance identification of the end points and protects transmitted data from disclosure and modification.