BlackBerry Smartphones with OS 10.3.3 VPN Client

Security Target

Doc No: 1958-003-D102
Version: 1.5
24 January 2016

BlackBerry
2200 University Ave. East
Waterloo, Ontario, Canada
N2K 0A7

Prepared by:
EWA-Canada
1223 Michael Street, Suite 200
Ottawa, Ontario, Canada
K1J7T2

Enabling a More Secure Future

CONTENTS

1 SECURITY TARGET INTRODUCTION 1
1.1 DOCUMENT ORGANIZATION 1
1.2 SECURITY TARGET REFERENCE 1
1.3 TOE REFERENCE 1
1.4 TOE OVERVIEW 2
1.5 TOE DESCRIPTION 2
1.5.1 Physical Scope 2
1.5.2 TOE Guidance 2
1.5.3 Logical Scope 3
2 CONFORMANCE CLAIMS 4
2.1 COMMON CRITERIA CONFORMANCE CLAIM 4
2.2 ASSURANCE PACKAGE CLAIM 4
2.3 PROTECTION PROFILE CONFORMANCE CLAIM 4
3 SECURITY PROBLEM DEFINITION 5
3.1 THREATS 5
3.2 ASSUMPTIONS 6
4 SECURITY OBJECTIVES 7
4.1 SECURITY OBJECTIVES FOR THE TOE 7
4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT 7
4.3 SECURITY OBJECTIVES CORRESPONDENCE 8
5 EXTENDED COMPONENTS 9
6 SECURITY REQUIREMENTS 10
6.1 CONVENTIONS 10
6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS 10
6.2.1 Cryptographic Support (FCS) 11
6.2.2 User Data Protection (FDP) 16
6.2.3 Identification and Authentication (FIA) 16
6.2.4 Security Management (FMT) 17
6.2.5 Protection of the TSF (FPT) 17
6.2.6 Trusted Path/Channels (FTP) 18

6.3 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE 18
6.4 DEPENDENCY RATIONALE 19
6.5 TOE SECURITY ASSURANCE REQUIREMENTS 21

7 TOE SUMMARY SPECIFICATION 23
7.1 CRYPTOGRAPHIC SUPPORT 23
7.1.1 Asymmetric Key Generation 23
7.1.2 Cryptographic Key Storage and Zeroization 23
7.1.3 Cryptographic Operation 25
7.1.4 IPSec Processing 26
7.1.5 Random Bit Generation 31
7.2 USER DATA PROTECTION 31
7.2.1 Information Flow Control 31
7.2.2 Residual Information Protection 31
7.3 IDENTIFICATION AND AUTHENTICATION 32
7.3.1 Certificate Validation and Use 32
7.4 SECURITY MANAGEMENT 32
7.5 PROTECTION OF THE TSF 33
7.5.1 TSF Self Test 33
7.5.2 Trusted Update 34
7.6 TRUSTED PATH / CHANNELS 34

8 ACRONYMS 36
8.1 ACRONYMS 36

LIST OF TABLES

Table 1 – Logical Scope of the TOE 3
Table 2 – Threats 5
Table 3 – Assumptions 6
Table 4 – Security Objectives for the TOE 7
Table 5 – Security Objectives for the Operational Environment 8
Table 6 – Mapping Between Objectives, Threats, and Assumptions 8
Table 7 – Extended Security Functional Requirements 9
Table 8 – Summary of Security Functional Requirements 11

Table 9 – Mapping of SFRs to Security Objectives 19
Table 10 – Functional Requirement Dependencies.. 21
Table 11 – Security Assurance Requirements 22
Table 12 – VPN Key Storage and Zeroization.. 25
Table 13 – HMAC Functions.. 25
Table 14 – NIAP-compatible format 29
Table 15 – Diffie-Hellman Group Security Values 29
Table 16 – Gateway Auth ID type and Certificate Identifier 30
Table 17 – Security Management Functions and VPN Profile Entries 33
Table 18 – Acronyms 38

1 SECURITY TARGET INTRODUCTION

This Security Target (ST) defines the scope of the evaluation in terms of the assumptions made, the intended environment for the Target of Evaluation (TOE), the Information Technology (IT) security functional and assurance requirements to be met, and the level of confidence (evaluation assurance level) to which it is asserted that the TOE satisfies its IT security requirements. This document forms the baseline for the Common Criteria (CC) evaluation.

1.1 DOCUMENT ORGANIZATION

Section 1, ST Introduction, provides the ST reference, the TOE reference, the TOE overview and the TOE description.

Section 2, Conformance Claims, describes how the ST conforms to the Common Criteria and Packages.

Section 3, Security Problem Definition, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.

Section 4, Security Objectives, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment in response to the problem defined by the security problem definition.

Section 5, Extended Components Definition, defines the extended components which are then detailed in Section 6.

Section 6, Security Requirements, specifies the security functional and assurance requirements that must be satisfied by the TOE and the IT environment.

Section 7, TOE Summary Specification, describes the security functions that are included in the TOE to enable it to meet the IT security functional requirements.

Section 8 Acronyms, defines the acronyms used in this ST.

1.2 SECURITY TARGET REFERENCE

ST Title:	BlackBerry Smartphones with OS 10.3.3 VPN Client Security Target
ST Version:	1.5
ST Date:	24 January 2016

1.3 TOE REFERENCE

TOE Identification:	BlackBerry OS 10.3.3.1668 VPN Client

TOE Developer:	BlackBerry
TOE Type:	VPN Client

1.4 TOE OVERVIEW

A Virtual Private Network (VPN) Client allows remote users to use client resources to establish an encrypted Internet Protocol Security (IPsec) tunnel across an unprotected public network to a private network. The TOE platform is a BlackBerry smartphone device running Operating System (OS) 10.3.3. The TOE is the VPN Client software that loads the VPN Profile which provides protection between itself and another VPN endpoint, such as a VPN Gateway. The TOE VPN Client, together with the TOE platform, protect the data between the device and a VPN Gateway, providing confidentiality, integrity, and protection of data in transit, even though it traverses a public network.

The TOE is a software only TOE.

1.5 TOE DESCRIPTION

1.5.1 Physical Scope

The VPN Client runs on the mobile device. The mobile device is defined as the BlackBerry 10.3.3 software running on one of the following BlackBerry smartphone devices:

- Passport
- Classic
- Leap
- Z30
- Z10
- Q10
- P’9982 (Porsche Design)
- P’9983 (Porsche Design)

The VPN Client is the portion of the BlackBerry 10.3.3 software that loads the VPN Profile. The smartphone hardware and the remainder of the OS are considered to be the TOE platform.

1.5.2 TOE Guidance

The primary guidance for the TOE is the Administration Guide BES12, Version 12.5 (SWD-20160824100629873 Published: 2016-08-24). The section on creating the VPN Profiles describes the VPN Client options. The guidance also includes the BlackBerry Smartphones with OS 10.3.3 Common Criteria Guidance Supplement version 1.1.

1.5.3 Logical Scope

The logical boundary of the TOE includes all interfaces and functions within the physical boundary. The logical boundary of the TOE may be broken down by the security function classes described in Section 6. The following breakdown also provides the description of the security features of the TOE, and loosely follows the security functional classes described in Section 6. Table 1 summarizes the logical scope of the TOE.

Functional Classes	Description
Cryptographic Support	Cryptographic functionality is provided in support of random bit generation, asymmetric key generation, key establishment, key storage, key destruction and cryptographic operations for the TOE. Specified cryptographic functions include the IPsec protocol.
User Data Protection	All IP traffic must pass through the IPsec VPN client. Data contained in protected resources is made unavailable when the resource is reallocated.
Identification and Authentication	X.509 certificates are used to support authentication of IPsec exchanges and are validated in accordance with policies.
Security Management	The TOE provides management capabilities for the configuration of VPN connections, IKE protocol, and authentication techniques. It also provides the ability to update the TOE and verify the updates.
Protection of the TSF	Self-tests must be run at start up. Trusted updates may be performed by authorized administrators.
Trusted Path/Channel	The TOE uses IPsec to provide a trusted communication channel between itself and a VPN Gateway.

Table 1 – Logical Scope of the TOE

2 CONFORMANCE CLAIMS

2.1 COMMON CRITERIA CONFORMANCE CLAIM

This Security Target claims to be conformant to Version 3.1 of Common Criteria for Information Technology Security Evaluation according to:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components CCMB-2012-09-003, Version 3.1, Revision 4, September 2012

As follows:

- CC Part 2 extended
- CC Part 3 conformant

The Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012 [CEM] has to be taken into account.

2.2 ASSURANCE PACKAGE CLAIM

This Security Target does not claim conformance to an Assurance Package, but conforms to the Security Assurance Requirements described in Section 4.3 of the Protection Profile for IPsec Virtual Private Network (VPN) Clients, Version 1.4, dated 21 October 2013.

2.3 PROTECTION PROFILE CONFORMANCE CLAIM

This Security Target claims exact conformance with the NIAP Protection Profile for IPsec Virtual Private Network (VPN) Clients, Version 1.4, dated 21 October 2013, and Technical Decisions TD0037, TD0042, TD0053 and TD0079.

3 SECURITY PROBLEM DEFINITION

3.1 THREATS

Table 2 lists the threats addressed by the TOE. Potential threat agents are authorized TOE users, and unauthorized persons. The level of expertise of both types of attacker is assumed to be unsophisticated. TOE users are assumed to have access to the TOE, extensive knowledge of TOE operations and to possess a high level of skill. They have moderate resources to alter TOE parameters, but are assumed not to be wilfully hostile. Unauthorized persons have little knowledge of TOE operations, a low level of skill, limited resources to alter TOE parameters and no physical access to the TOE.

Mitigation to the threats is through the objectives identified in Section 4.1 Security Objectives.

Threat	Description
T.TSF_CONFIGURATION	Failure to allow configuration of the TSF may prevent its users from being able to adequately implement their particular security policy, leading to a compromise of user information.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNAUTHORIZED_ ACCESS	A user may gain unauthorized access to the TOE data. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_ UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender because it is not rendered inaccessible after it is done being used.

Table 2 – Threats

3.2 ASSUMPTIONS

The assumptions required to ensure the security of the TOE are listed in Table 3.

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

Table 3 – Assumptions

4 SECURITY OBJECTIVES

The purpose of the security objectives is to address the security concerns and to show which security concerns are addressed by the TOE, and which are addressed by the environment. Threats may be addressed by the TOE or the security environment or both. Therefore, the CC identifies two categories of security objectives:

- Security objectives for the TOE
- Security objectives for the environment

4.1 SECURITY OBJECTIVES FOR THE TOE

Table 4 identifies and describes the security objectives that are to be addressed by the TOE.

Security Objective	Description
O.VPN_TUNNEL	The TOE will provide a network communication channel protected by encryption that ensures that the VPN client communicates with an authenticated VPN gateway.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to allow administrators to be able to configure the TOE.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.

Table 4 – Security Objectives for the TOE

4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

Table 5 identifies and describes the security objectives that are to be addressed by the IT domain or by non-technical or procedural means.

Security Objective	Description
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.

Table 5 – Security Objectives for the Operational Environment

4.3 SECURITY OBJECTIVES CORRESPONDENCE

Table 6 maps the security objectives to the assumptions and threats identified for the TOE.

	T.TSF_CONFIGURATION	T.TSF_FAILURE	T.UNAUTHORIZED_ ACCESS	T.UNAUTHORIZED_ UPDATE	T.USER_DATA_REUSE	A.NO_TOE_BYPASS	A.PHYSICAL	A.TRUSTED_CONFIG
O.VPN_TUNNEL			X					
O.RESIDUAL_ INFORMATION_CLEARING					X			
O.TOE_ADMINISTRATTION	X							
O.TSF_SELF_TEST		X						
O.VERFIFIABLE_UPDATES				X				
OE.NO_TOE_BYPASS						X		
OE.PHYSICAL							X	
OE.TRUSTED_CONFIG								X

Table 6 – Mapping Between Objectives, Threats, and Assumptions

5 EXTENDED COMPONENTS

This section specifies the extended Security Functional Requirements (SFRs) used in this ST and defined in the Protection Profile for IPsec Virtual Private Network (VPN) Clients. The following table identifies the extended SFRs that have been created to address additional security features of the TOE:

Class	Family	Component
FCS: Cryptographic Support	FCS_CKM_EXT: Cryptographic Key Management	FCS_CKM_EXT.2: Cryptographic Key Storage
		FCS_CKM_EXT.4: Cryptographic Key Zeroization
	FCS_IPSEC_EXT: Internet Protocol Security	FCS_IPSEC_EXT.1: Internet Protocol Security (IPsec) Communications
	FCS_RBG_EXT: Cryptographic Operation (Random Bit Generation)	FCS_RBG_EXT.1: Cryptographic Operation (Random Bit Generation)
FDP: User Data Protection	FDP_IFC_EXT: Information Flow Control Policy	FDP_IFC_EXT.1 Subset Information Flow Control
FIA: Identification and Authentication	FIA_X509_EXT: X509 Certificates	FIA_X509_EXT.1: X.509 Certificate Validation
		FIA_X509_EXT.2: X.509 Certificate Use and Management
FPT: Protection of the TSF	FPT_TST_EXT: TSF Self Test	FPT_TST_EXT.1: TSF Self Test
	FPT_TUD_EXT: Trusted Update	FPT_TUD_EXT.1: Trusted Update

Table 7 – Extended Security Functional Requirements

6 SECURITY REQUIREMENTS

This section provides security functional and assurance requirements that must be satisfied by a compliant TOE. These requirements consist of functional components from Part 2 of the CC and extended requirements as described in the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

6.1 CONVENTIONS

The CC permits four types of operations to be performed on functional requirements: selection, assignment, refinement, and iteration. These operations, when performed on requirements that derive from CC Part 2, are identified in this ST in the following manner:

- Selection: Indicated by surrounding brackets, e.g., [selected item].
- Assignment: Indicated by surrounding brackets and italics, e.g., [*assigned item*].
- Refinement: Refined components are identified by using **bold** for additional information, or ~~strikeout~~ for deleted text.
- Iteration: Indicated by assigning a number in parenthesis to the end of the functional component identifier as well as by modifying the functional component title to distinguish between iterations, e.g., 'FCS_CKM.1(1), Cryptographic key generation (asymmetric keys)' and 'FCS_CKM.1(2) Cryptographic key generation (for asymmetric keys – IKE)'.

6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for this ST consist of the following components from Part 2 of the CC and extended components identified in Section 5, summarized in Table 8 - Summary of Security Functional Requirements.

Class	Identifier	Name
Cryptographic Support (FCS)	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)

Class	Identifier	Name
	FCS_COP.1(3)	Cryptographic operation (cryptographic hashing)
	FCS_COP.1(4)	Cryptographic operation (keyed-hash message authentication)
	FCS_IPSEC_EXT.1	Internet protocol security (IPsec) communications
	FCS_RBG_EXT.1	Cryptographic operation (random bit generation)
User Data Protection (FDP)	FDP_IFC_EXT.1	Subset Information Flow Control
	FDP_RIP.2	Full residual information protection
Identification and Authentication (FIA)	FIA_X509_EXT.1	X.509 certificate validation
	FIA_X509_EXT.2	X.509 certificate use and management
Security Management (FMT)	FMT_SMF.1 (1)	Specification of management functions (TOE)
	FMT_SMF.1 (2)	Specification of management functions (TOE Platform)
Protection of the TSF (FPT)	FPT_TST_EXT.1	TSF self test
	FPT_TUD_EXT.1	Trusted update
Trusted path/channels (FTP)	FTP_ITC.1	Inter-TSF trusted channel

Table 8 – Summary of Security Functional Requirements

6.2.1 Cryptographic Support (FCS)

6.2.1.1 FCS_CKM.1(1) Cryptographic key generation (asymmetric keys)

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution,
or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1(1) The **[TOE platform]** ~~TSF~~ shall generate **asymmetric** cryptographic keys **used for key establishment** in accordance with ~~a specified cryptographic key generation algorithm~~ [

- *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 [no other curves] (as defined in FIPS PUB 186-4, "Digital Signature Standard")*
- *[no other]*]

and specified cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 112 bits*]~~ that meet the following: [assignment: *list of standards*]~~.

6.2.1.2 FCS_CKM.1(2) Cryptographic key generation (for asymmetric keys - IKE)

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1(2) The **[TOE platform]** ~~TSF~~ shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a ~~specified cryptographic key generation algorithm~~:

[

- *FIPS PUB 186-4, "Digital Signature Standards (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" p-256, p-384 and [P-521]*]

and specified cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 112 bits*] ~~that meet the following: [assignment: *list of standards*]~~.

6.2.1.3 FCS_CKM_EXT.2 Cryptographic Key Storage

Hierarchical to: No other components.

Dependencies: No dependencies

FCS_CKM_EXT.2.1 The [TOE platform] shall store persistent secrets and private keys when not in use in platform-provided key storage.

6.2.1.4 FCS_CKM_EXT.4 Cryptographic Key Zeroization

Hierarchical to: No other components.

Dependencies: No dependencies

FCS_CKM_EXT.4.1 The [TOE platform] shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

6.2.1.5 FCS_COP.1(1) Cryptographic operation (data encryption/ decryption)

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(1) The **[TOE platform]** ~~TSF~~ 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 meet**s** the following:[

- *FIPS PUB 197, "Advanced Encryption Standard (AES)"*
- *NIST SP 800-38D, NIST SP 800-38A.*]

6.2.1.6 FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(2) The **[TOE platform]** ~~TSF~~ 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 (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" p-256, p-384 and [no other curve]*]

and cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 112 bits*] ~~that meet the following: [assignment: *list of standards*]~~.

6.2.1.7 FCS_COP.1(3) Cryptographic operation (cryptographic hashing)

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(3) The **[TOE platform]** ~~TSF~~ shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm [*SHA-1, SHA-256, SHA-384, SHA-512*] and **message digest** ~~cryptographic key~~ sizes [*160, 256, 384, 512 bits*] that meet the following: [*FIPS Pub 180-4, "Secure Hash Standard."*]

6.2.1.8 FCS_COP.1(4) Cryptographic operation (keyed-hash message authentication)

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(4) The **[TOE platform]** ~~TSF~~ shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm [*HMAC* [*SHA-1, SHA-256, SHA-384*]] ~~and cryptographic~~ key size [*256 bit key sizes, and message digest size of* [*160, 256, 384*] *bits*] that meet the following: [*FIPS PUB 198-1, "The Keyed-Hash Message Authentication Code", and FIPS PUB 180-4, "Secure Hash Standard".*]

6.2.1.9 FCS_IPSEC_EXT.1 Internet Protocol Security (IPsec) Communications

Hierarchical to: No other components.

Dependencies: No dependencies

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

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

FCS_IPSEC_EXT.1.3 The [TOE platform] 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 platform] 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, AES-CBC-256 (both specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC].

FCS_IPSEC_EXT.1.5 The [TOE platform] shall implement the protocol: [IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and [RFC 4868 for hash functions]].

FCS_IPSEC_EXT.1.6 The [TOE platform] 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 platform] shall ensure that IKEv1 Phase 1 exchanges use only main mode.

FCS_IPSEC_EXT.1.8 The [TOE platform] shall ensure that [IKEv2 SA lifetimes can be configured by [an Administrator] 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 platform] 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 [192, 224, 256, and 384] bits.

FCS_IPSEC_EXT.1.10 The [TOE platform] 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^ [*512*].

FCS_IPSEC_EXT.1.11 The [TOE platform] 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), no other DH groups].

FCS_IPSEC_EXT.1.12 The [TOE platform] shall ensure that all IKE protocols perform peer authentication using a [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [no other method].

FCS_IPSEC_EXT.1.13a The [TOE platform] shall support peer identifiers of the following types: [Fully Qualified Domain Name (FQDN), user FQDN Distinguished Name (DN)] and [no other reference identifier type].

FCS_IPSEC_EXT.1.13b The [TOE platform] shall not establish an SA if the presented identifier does not match the configured reference identifier of the peer.

FCS_IPSEC_EXT.1.14 The [TOE platform] 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.

Application Note: FCS_IPSEC_EXT.1.7 is included for completeness; however, IKEv1 is not supported in the evaluated configuration.

6.2.1.10 FCS_RBG_EXT.1 Cryptographic operation (random bit generation

Hierarchical to: No other components.

Dependencies: No dependencies

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 [HMAC_DRBG (any)]].

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 an hashes that it will generate.

6.2.2 User Data Protection (FDP)

6.2.2.1 FDP_IFC_EXT.1

Hierarchical to: FDP_RIP.1 Subset residual information protection

Dependencies: No dependencies.

FDP_IFC_EXT.1.1 The TSF shall ensure that all IP traffic (other than IP traffic required to establish the VPN connection) to flow through the IPsec VPN client.

6.2.2.2 FDP_RIP.2 Full residual information protection

Hierarchical to: FDP_RIP.1 Subset residual information protection

Dependencies: No dependencies.

FDP_RIP.2.1 The **[TOE platform]** ~~TSF~~ shall **enforce** ~~ensure~~ that any previous information content of a resource is made unavailable upon the [deallocation of the resource from] all objects.

6.2.3 Identification and Authentication (FIA)

6.2.3.1 FIA_X509_EXT.1 X.509 certificate validation

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_X509_EXT.1.1 The [TOE platform] 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:
  - 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 platform] 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.

6.2.3.2 FIA_X509_EXT.2 Certificate use and management

Hierarchical to: No other components.

Dependencies: No dependencies

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 platform] shall [not accept the certificate].

FIA_X509_EXT.2.3 The [TOE platform] shall not establish an SA if a certificate or certificate path is deemed invalid.

6.2.4 Security Management (FMT)

6.2.4.1 FMT_SMF.1(1) Specification of management functions (TOE)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1(1) The **TOE** ~~TSF~~ shall be capable of performing the following management functions: [

- *Specify VPN Gateways to use for connections,*
- *Specify client credentials to be used for connections*].

6.2.4.2 FMT_SMF.1(2) Specification of management functions (TOE Platform)

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1(2) The **[TOE Platform]** ~~TSF~~ 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 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 updates,*
- *Ability to configure all security management functions identified in other sections of this PP,*
- *Configure the reference identifier for the peer*
- [*no other actions*]].

6.2.5 Protection of the TSF (FPT)

6.2.5.1 FPT_TST_EXT.1 TSF self test

Hierarchical to: No other components.

Dependencies: No dependencies.

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 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 [*a digital signature in accordance with FCS_COP.1(2) and FCS_COP.1(3) using a hardware-protected asymmetric key*].

6.2.5.2 FPT_TUD_EXT.1 Trusted update

Hierarchical to: No other components.

Dependencies: No dependencies.

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

FPT_TUD_EXT.1.2 The [TOE Platform] shall provide 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.

6.2.6 Trusted Path/Channels (FTP)

6.2.6.1 FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

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

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

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

6.3 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

Table 9 provides a mapping between the SFRs and Security Objectives. Additional mappings are provided to complete those provided in the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

	O.VPN_TUNNEL	O.RESIDUAL_ INFORMATION_CLEARING	O.TOE_CONFIGURATION	O.TSF_SELF_TEST	O.VERIFIABLE_UPDATES
FCS_CKM.1(1)	X				
FCS_CKM.1(2)	X				
FCS_CKM_EXT.2	X				
FCS_CKM_EXT.4	X				
FCS_COP.1(1)	X				
FCS_COP.1(2)	X				X
FCS_COP.1(3)	X				X
FCS_COP.1(4)	X				
FCS_IPSEC_EXT.1	X				
FCS_RBG_EXT.1	X				
FDP_IFC_EXT.1	X				
FDP_RIP.2		X			
FIA_X509_EXT.1	X				X
FIA_X509_EXT.2	X				
FMT_SMF.1(1)			X		
FMT_SMF.1(2)			X		
FPT_TST_EXT.1				X	
FPT_TUD_EXT.1					X
FTP_ITC.1	X				

Table 9 – Mapping of SFRs to Security Objectives

6.4 DEPENDENCY RATIONALE

Table 10 identifies the Security Functional Requirements from Part 2 of the CC and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency.

SFR	Dependency	Dependency Satisfied	Rationale
FCS_CKM.1(1)	FCS_CKM.2 or FCS_COP.1	✓	
	FCS_CKM.4	✓	
FCS_CKM.1(2)	FCS_CKM.2 or FCS_COP.1	✓	
	FCS_CKM.4	✓	
FCS_CKM_EXT.2	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FCS_CKM_EXT.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
FCS_COP.1(1)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FCS_COP.1(2)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FCS_COP.1(3)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FCS_COP.1(4)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FCS_IPSEC_EXT.1	None	N/A	
FCS_RBG_EXT.1	None	N/A	
FDP_IFC_EXT.1	None	N/A	
FDP_RIP.2	None	N/A	
FIA_X509_EXT.1	None	N/A	

SFR	Dependency	Dependency Satisfied	Rationale
FIA_X509_EXT.2	None	N/A	
FMT_SMF.1(1)	None	N/A	
FMT_SMF.1(2)	None	N/A	
FPT_TST_EXT.1	None	N/A	
FPT_TUD_EXT.1	None	N/A	
FTP_ITC.1	None	N/A	

Table 10 – Functional Requirement Dependencies

6.5 TOE SECURITY ASSURANCE REQUIREMENTS

The TOE assurance requirements for this ST consist of the requirements corresponding to the Security Assurance Requirement specified in the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

The assurance requirements are summarized in Table 11.

Assurance Class	Assurance Components	
	Identifier	Name
Development (ADV)	ADV_FSP.1	Basic functional specification
Guidance Documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-Cycle Support (ALC)	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage

Assurance Class	Assurance Components	
	Identifier	Name
Security Target Evaluation (ASE)	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_TSS.1	TOE summary specification
Tests (ATE)	ATE_IND.1	Independent testing - sample
Vulnerability Assessment (AVA)	AVA_VAN.1	Vulnerability survey

Table 11 – Security Assurance Requirements

7 TOE SUMMARY SPECIFICATION

This section provides a description of the security functions and assurance measures of the TOE and TOE platform that meet the security requirements.

A description of each of the security functions follows.

7.1 CRYPTOGRAPHIC SUPPORT

This is a description of how the TOE and TOE platform meet the cryptographic support security objectives.

7.1.1 Asymmetric Key Generation

The key establishment functionality for asymmetric keys used for key establishment is invoked through initiation of the IPsec protocol.

TOE Security Functional Requirements addressed: FCS_CKM.1(1), FCS_CKM.1(2).

7.1.2 Cryptographic Key Storage and Zeroization

The persistent secrets and keys used in support of the VPN Client are listed in Table 12.

Private keys are stored encrypted in the CertStore database on the TOE platform.

Persistent Secrets and Keys	Generation and Usage	Storage Location	Zeroization
VPN Identity Certificate Private Key	Generated using the platform based cryptographic functions whenever a new SCEP profile is received. Used in support of VPN client identification	Stored encrypted in the CertStore database Stored in main memory (RAM) during use.	The private key may be cleared for a workspace wipe, or at the request of the user or the BES administrator. It is overwritten by zeroes. Overwritten by zeroes in RAM immediately after use.

Persistent Secrets and Keys	Generation and Usage	Storage Location	Zeroization
VPN Client Certificate	Generated using the platform based cryptographic functions whenever a new SCEP profile is received. Used to encrypt/ decrypt in support of VPN client operations	Stored in the CertStore database	The certificate may be cleared for a workspace wipe, or at the request of the user or the BES administrator. It is overwritten by zeroes.
CA Certificate	Used to verify the identity of the VPN Gateway	Stored in the CertStore database	The values are not secret. The certificate is discarded when replaced by an updated certificate.
DH Group parameters	Provided with the TSF and used in the Diffie-Hellman key exchange portion of the IPsec protocol	TOE software	The values are not secret and are never cleared.
IKEv2/IPsec Diffie-Hellman Secrets	Provided with the TSF and used to generate keys to support the Diffie-Hellman key exchange portion of the IKEv2 and IPsec protocol	Held in memory as required and never stored.	Overwritten by zeroes when the SA is dropped.
IKEv2 SA encryption key AES-CBC and AES-GCM	Generated as part of the establishment of the security association and used to encrypt and decrypt VPN protected traffic	Held in memory as required and never stored.	Overwritten by zeroes when the SA is dropped.
IKEv2 SA MAC key HMAC-SHA	Generated as part of the establishment of the security association and used to ensure the integrity of VPN protected traffic	Held in memory as required and never stored.	Overwritten by zeroes when the SA is dropped.

Persistent Secrets and Keys	Generation and Usage	Storage Location	Zeroization
IKEv2 Child SA encryption key AES-CBC and AES-GCM	Generated as part of the establishment of the child security association and used to encrypt and decrypt VPN protected traffic	Held in memory as required and never stored.	Overwritten by zeroes when the child SA is dropped.
IKEv2 Child SA MAC key HMAC-SHA	Generated as part of the establishment of the child security association and used to ensure the integrity of VPN protected traffic	Held in memory as required and never stored.	Overwritten by zeroes when the child SA is dropped.

Table 12 – VPN Key Storage and Zeroization

TOE Security Functional Requirements addressed: FCS_CKM_EXT.2, FCS_CKM_EXT.4.

7.1.3 Cryptographic Operation

All cryptographic functionality is invoked via the Security Builder API as specified in the developer documentation found at: https://developer.blackberry.com/native/reference/core/com.qnx.doc.crypto.lib_ref/topic/manual/intro.html.

The TOE supports AES-CBC and AES-GCM for encryption/decryption within IPsec.

The TOE supports both RSA and ECDSA signature schemes.

The TOE supports SHA-1, SHA-256, SHA-384 and SHA-512.

The TOE supports HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-384 as described in Table 13.

HMAC	Key Length (bits)	Hash Function	Block Size	Output MAC
HMAC-SHA-1	160	SHA-1	512	160
HMAC-SHA-256	256	SHA-256	512	256
HMAC-SHA-384	384	SHA-384	1024	384

Table 13 – HMAC Functions

TOE Security Functional Requirements addressed: FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4).

7.1.4 IPSec Processing

In the evaluated configuration, the NIAP-compatible profile must be used. To put the device into this configuration, the profile parameters shown in Table 14 must be selected, as shown for each the authentication mechanism, in the VPN Profile. This profile is configured in the BlackBerry Enterprise Service (BES).

Parameter	Policy	Notes
Name	An administrator-defined name %name%	The administrator defined name is entered as a string
Server Address:	The fully qualified domain name of the server %FQDN%	The fully qualified domain name of the server is entered as a string
Gateway Type:	NIAP-compliant[1] IKEv2 VPN server	This must be selected in the evaluated configuration
Authentication Type:	PKI	PKI must be chosen in the evaluated configuration
EAP Identity	%Username%	Not applicable when the authentication type is set to PKI
Gateway Authentication Type:	PKI	This must be selected in the evaluated configuration
Enable OSCP and CRL checks on the Certificate from the VPN:	Checked	This must be selected in the evaluated configuration
Authentication ID Type:	Fully Qualified Domain Name, Email address, Identity certificate distinguished name	One of these selections must be made in the evaluated configuration
Authentication ID or Group username:	Username %UserName%	The authentication ID or Group username is entered as a string
Gateway authentication ID type:	Fully Qualified Domain Name, Email address, Identity certificate distinguished name	One of these selections must be made in the evaluated configuration
Gateway authentication ID:	Gateway Identification %GW ID%	The Gateway ID is entered as a string
Send requested gateway ID in message 1 of IKEv2 protocol:	Unchecked	This must remain unchecked in the evaluated configuration
Allow personal	Checked	This selection is mandatory in

[1] When NIAP-compliant is selected as the Gateway type, IKEv2 is automatically selected.

Parameter	Policy	Notes
apps to use work networks:		the evaluated configuration
Untrusted certificate action:	Prompt	This selection is mandatory in the evaluated configuration
Client certificate source:	Other	This selection is mandatory in the evaluated configuration
IKE lifetime:	minimum 60 seconds, maximum 2147483647 seconds [60, 2147483647]	The recommended value for the evaluated configuration is 86400 seconds
IKE threshold:	percentage [0-100]	The recommended value for the evaluated configuration is 90%
IPSec lifetime	minimum 60 seconds, maximum 2147483647 seconds [60, 2147483647]	The recommended value for the evaluated configuration is 10800 seconds
IPSec threshold:	percentage [0-100]	The recommended value for the evaluated configuration is 90%
Allow VPN extensions	Checked	This must be selected in the evaluated configuration
VPN extensions	TestCertCheckExtension	This is required to test the bar file
Require vendor ID extension	Unchecked	This must remain unchecked in the evaluated configuration
Require certificate validation extension:	Checked	This must be selected in the evaluated configuration
Enable hash- and URL format certificate payload during IKE:	Checked	This must be selected in the evaluated configuration (Certificate URL field may be left blank)
Enable strict enforcement of approved algorithms:	Checked	This must be selected in the evaluated configuration
Manual algorithms selection:	Checked	This must be selected in the evaluated configuration
IKE DH group:	5, 14, 19, 20, 24	
IKE cipher:	AES (128-bit key), AES (256-bit key), AES128_ICV16_GCM (128-bit key), AES256_ICV16_GCM	

Parameter	Policy	Notes
	(256-bit key)	
IKE hash:	SHA1, SHA256, SHA384, SHA512, AES128_ICV16_GCM, AES256_ICV16_GCM	
IKE PRF:	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384	
IPSec DH group:	5, 14, 19, 20, 24	A DH Group must be selected in the evaluated configuration: 5, 14, 19, 20 or 24
IPSec cipher:	AES (128-bit key), AES (256-bit key), AES-ICV16-GCM (128-bit key), AES-ICV16-GCM (256-bit key)	
IPSec hash:	SHA1, SHA256, SHA384, SHA512, AES128_ICV16_GCM (128-bit key), AES256_ICV16_GCM (256-bit key)	
Trusted certificate source:	Trusted certificate store	This must be selected in the evaluated configuration
Automatically determine IP:	Checked	This is optional in the evaluated configuration
Automatically determine DNS:	Checked	This is optional in the evaluated configuration
Perfect forward secrecy:	Checked	This must be selected in the evaluated configuration
NAT keepalive:	blank	This is an optional selection, but is left blank in the evaluated configuration. When blank, this defaults to 30 seconds
DPD frequency:	blank	This is an optional selection, but is left blank in the evaluated configuration. When blank, this defaults to 240 seconds
User can edit:	Credentials Only	This must be selected in the evaluated configuration
Display VPN information on device:	Visible	
Data security level:	Always Available	This must be selected in the evaluated configuration
Associated profiles: Associated proxy profile:	This must be left as “ – Select – ”	No selection is to be made in the evaluated configuration

Table 14 – NIAP-compatible format

VPN connections are established to operate in tunnel mode.

The TOE platform implements a Security Policy Database (SPD) in accordance with RFC 4301 that provides the rules for how a packet is processed through the use of the VPN profile. If a packet does not meet the required conditions for passing the packet, the packet will be discarded.

AES-CBC-128, AES-CBC-256, AES-GCM-128 and AES-GCM-256 are supported for payload encryption. The IPSec Hash may be SHA1, SHA256, SHA384, SHA512, AES128_ICV16_GCM, or AES256_ICV16_GCM.

Only IKEv2 is supported in the evaluated configuration. Once 'NIAP-compatible' is selected as the Gateway type, IKEv2 becomes the only available option.

When selecting the IPSec cipher, the administrator must first select 'manual algorithm selection'. Then one of the following algorithms may be selected: AES (128-bit key), AES (256-bit key), AES-ICV16-GCM (128-bit key) or AES-ICV16-GCM (256-bit key).

IKEv2 is supported with mandatory support for NAT traversal. The IKEv2 payload is encrypted using AES-CBC-128 and AES-CBC-256, as selected in the VPN profile. AES-GCM-128 and AES-GCM-256 may be specified as indicated in Table 14.

IKEv1 is not supported. Therefore, IKEv1 Phase 1 aggressive mode can never be used.

Only the BES administrator may configure the IKEv2 Phase 1 (IKE lifetime in the Administration Guide) and IKEv2 Phase 2 (IPsec lifetime in the Administration Guide) SA lifetime. This is configured in the VPN profile, and pushed to the mobile device. It may not be changed on the mobile device. Possible lifetime values are from 1 to 2,147,483,647 seconds (1 second to approximately 68 years).

Diffie-Hellman Groups 5, 14, 19, 20 and 24 are supported. This corresponds to the bits of security and values of 'x', the secret value used in the exchange, as shown in Table 15.

Diffie-Hellman Group	Value of 'x'	Bits of Security
5	192	90
14	224	112
19	256	128
20	384	192
24	256	112

Table 15 – Diffie-Hellman Group Security Values

The process for generating x uses the hu_ECCKeyGen and hu_IDLCKeyGen functions from the Certicom Security Builder Government Solution Edition (SBGSE) and uses the TOE platform DRBG. The key generation process is validated, and is consistent with the FIPS 186-4 standard.

By default, the TOE generates 64 bytes of nonce using the TOE platform DRBG. Assuming no bias in the random number generator, nonce values will be repeated with probability 1/2^512.

Diffie-Hellman groups 5, 14, 19, 20 and 24 are implemented in the IKE protocol. DH Group negotiation is performed in accordance with RFC 5996. No negotiation is permitted. Both the client and the gateway must select the same option or the tunnel will not be created.

Peer authentication may be performed using either RSA or ECDSA. The choice is made by selecting the certificate with which to authenticate.

During authentication, the Gateway Authentication ID is compared against one identifier in a certificate based on the Gateway Authentication ID Type. The certificate identifier used is as follows:

Gateway Authentication ID Type	Certificate Identifier
FQDN	dNSName
Email	rfc822name
Certificate Subject Name	Distinguished Name

Table 16 – Gateway Auth ID type and Certificate Identifier

The BES administrator may configure the FQDN, DN or email address to be used as the reference identifier. Upon receipt of a certificate from the VPN Gateway, the TOE will ensure that the certificate has not been revoked as per the OCSP protocol, and that the configured reference identifier matches the dNSName, rfc822Name or the certificate subject name presented in the certificate before establishing a connection.

On the VPN Client, the 'Authentication ID Type' field determines the format of the authentication ID the client sends to the gateway for authentication. The field 'Gateway Authentication ID Type' determines the expected format of the authentication ID that the gateway sends to the client during authentication.

For example, if the VPN Client Authentication ID Type is FQDN, and the Gateway Authentication ID Type is 'Email Address', then the client will check the Email address in the server’s certificate, and the VPN server will be expected to check the FQDN of the client’s certificate. The client will fail to authenticate with the server if the 'Email Address' provided does not match the Subject Alt Name in the certificate, the server will fail to authenticate with the client if the presented FQDN does not match the Subject Alt Name in the server certificate.

When establishing a Child SA, the TOE verifies that the key length of the IKE SA is greater than or equal to the key length of the Child SA. This is a simple check of the number of bits in the key length in the evaluated configuration where only AES-CBC and AES-GCM are supported.

TOE Security Functional Requirements addressed: FCS_IPSEC_EXT.1.

7.1.5 Random Bit Generation

The BlackBerry OS Cryptographic Kernel implements HMAC_DRBG in accordance with SP 800-90 and DRBG certificate #81. The DRBG implementations are seeded by entropy with sources from software and hardware based noise sources with a minimum of 256 bits of entropy. The VPN client may request and be provided with random bits from these DRBG implementations. All requests are invoked via the Security Builder API as specified in the developer documentation found at: https://developer.blackberry.com/native/reference/core/com.qnx.doc.crypto.lib_ref/topic/manual/intro.html.

TOE Security Functional Requirements addressed: FCS_RBG_EXT.1.

7.2 USER DATA PROTECTION

7.2.1 Information Flow Control

The TOE platform ensures that when an IPsec VPN connection is established, all IP traffic flows through the VPN client.

Once a trusted, native VPN connection is established, the TOE platform routes all IP traffic over the native VPN, with the exception of internet connectivity control information. All other traffic is blocked using packet filter rules.

Internet connectivity detection traffic is routed outside the native VPN as it used in attempts to verify external connectivity of each baseband, as well as for captive portal detection. This Internet connectivity check is used to determine which baseband is used to establish the VPN. An HTTP request is sent to public web server to verify Internet connectivity. Responses from the web server are used to verify Internet connectivity over the specified baseband. For the purposes of FDP_IFC_EXT.1, this is considered to be "traffic required to establish the VPN connection".

In the absence of an established native VPN connection, no IP traffic other than this noted exception is allowed to leave or enter the TOE. This is the same for all baseband (LTE and Wi-Fi) protocols.

TOE Security Functional Requirements addressed: FDP_IFC_EXT.1.

7.2.2 Residual Information Protection

The TOE’s VPN client includes both an inbound buffer and an outbound buffer. Traffic may be held temporarily in the inbound buffer while waiting for a user prompt or certificate check operation. Once the processing of the packet is complete, the inbound buffer is zeroized by overwriting the buffer with zeroes. The outbound buffer is zeroized after the packet is sent. The outbound buffer is overwritten with zeroes once when the client responds to the gateway, and the inbound buffer is overwritten three times when the client is awaiting a gateway response.

TOE Security Functional Requirements addressed: FDP_RIP.2.

7.3 IDENTIFICATION AND AUTHENTICATION

7.3.1 Certificate Validation and Use

The TOE platform validates a certificate using the certificate path validation. The certificate verification process is initiated on the TOE platform by invoking the CertMgr API. The TOE platform can also be configured to validate the revocation status of the certificate by using the Online Certificate Status Protocol (OCSP). If certificate revocation checking is enabled, and the OCSP responder cannot be reached, the certificate is rejected.

The BB10 Certificate Manager performs the certificate path validation in the Trust Anchor Database using the certificate path validation algorithm, which performs the following checks:

1. Ensure the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates;
2. Use Online Certificate Status Protocol (OCSP) as specified in RFC 2560 to verify revocation status;
3. Validates the extendedKeyUsage field according to the following rule:
    - The validation chain terminates in a trusted root certificate.

The TOE platform only treats certificates as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

The certificate to be used by the TOE is identified in the VPN profile.

TOE Security Functional Requirements addressed: FIA_X509_EXT.1, FIA_X509_EXT.2.

7.4 SECURITY MANAGEMENT

All of the management functionality is configured in the BES, in the VPN Profile. This is passed to the TOE platform (BBOS 10.3.3) where the configured options are enforced on the TOE (VPN Client).

Table 17 provides a mapping between the security management functions and the entries in the VPN Profile that are used to configure the function.

Security Management Function	VPN Profile Entry
Specify VPN Gateways to use for connections	Server Address
Specify client credentials to be used for connections	Authentication ID type and Authentication ID
Configuration of IKE protocol version(s) used	Gateway type. When 'NIAP-compliant IKEv2 VPN server' is selected as the Gateway Type, IKEv2 is automatically selected and cannot be changed.
Configure IKE authentication techniques used	Authentication Type
Configure the cryptoperiod for the established session keys	IKE lifetime, IPsec lifetime
Configure certificate revocation check	Enabled via the ‘Enable OCSP checks on certificate’ checkbox.
Specify the algorithm suites that may be proposed and accepted during the IPsec exchanges	The ‘Manual algorithm selection’ checkbox must be selected. The IKE DH group, IKE Cipher, IKE Hash, IKE PRF, IPSec DH Group, IPSec Cipher and IPSec Hash are used to select algorithms.
Load X.509v3 certificates	Certificates are not configured as part of the VPN Profile. They must be loaded onto the smartphone device.
Ability to configure all security management functions identified in other sections of this PP	As indicated in Table 14
configure the reference identifier for the peer	Gateway Authentication ID type, Gateway authentication ID

Table 17 – Security Management Functions and VPN Profile Entries

The ability to update the TOE and verify updates is not included in the VPN Profile. Updates to the BBOS software may be sent by the Administrator, and verified and installed on the smartphone device. These software updates may be generated to include updates to the VPN Client.

The type of client credential used by the mobile device for establishing a connection is specified in the user’s VPN profile. This credential is used by the TOE VPN client to authenticate to the VPN gateway.

TOE Security Functional Requirements addressed: FMT_SMF.1.

7.5 PROTECTION OF THE TSF

7.5.1 TSF Self Test

The BlackBerry OS Cryptographic Library performs a power-on self test (POST) to ensure the cryptographic library is not modified and all the cryptographic functions perform correctly.

The self-tests are initiated automatically by the module at start-up, as follows:

1. Known Answer Tests (KATs): KATs are performed on TDES, AES, AES GCM, SHS (using HMAC-SHS), HMAC-SHS, DRBG, RSA Signature Algorithm, and KDF. For DSA and ECDSA, a Pair-wise Consistency Test is used. For DH, ECDH, ECMQV, the underlying arithmetic implementations are tested using DSA and ECDSA tests.

2. Software Integrity Test: The software integrity test deploys ECDSA signature validation to verify the integrity of the module.

The secure boot process includes the authentication of the processor software by verifying the digital signature using the Code Signing Public Key. This digital signature is applied to the OS 10.3.3 software, which includes the VPN Client. The processor is bound to the Code Signing key through hardware.

The device powers on when the self test completes successfully. If the self test does not complete successfully, the device indicates that an error has occurred.

TOE Security Functional Requirements addressed: FPT_TST_EXT.1.

7.5.2 Trusted Update

The VPN Client is embedded in the operating system of the BlackBerry Smartphone with OS 10.3.3 device and may only be updated as part of an operating system update.

The TOE platform verifies software updates to the Application Processor using digital signatures. A hardware-protected public key held in the Trust Anchor Database is used to verify the key used for signatures on software updates. The TOE platform only allows the boot integrity hash to be updated during a software update process.

The calculation of the hash values FWIR is SHA-512. The public key from the Trust Anchor Database is used to verify the author certificates that are used to sign the BAR files. A Certicom X.509 certificate decoder is used to perform the signature verification function. The Package-Author-Certificate-Hash is held in the MANIFEST.MF file.

BlackBerry maintains the chain of trust by controlling the package generation, and by using hash algorithms (as part of the digital signature) to confirm that the BAR file has not been tampered with. If the checks fail at any point, the BAR file cannot be installed.

The TOE platform is described in the BlackBerry Smartphones with OS 10.3.3 Security Target.

TOE Security Functional Requirements addressed: FPT_TUD_EXT.1.

7.6 TRUSTED PATH / CHANNELS

The TOE uses IPsec to provide a communication channel between itself and a VPN Gateway. Packets are processed in accordance with RFC 4301. Packets are processed against the SPD as described in Section 7.1.4, in accordance with the cryptographic protocols described in Section 6.2.1.9. The channel provides assured identification of the end points, detects any modification of the data and protects the data from disclosure and modification. The cryptographic protocols are as described for FCS_IPSEC_EXT.1.

The VPN profile must be configured as described in Section 7.1.4. The user may initiate the connection to the VPN Gateway by tapping on the appropriate VPN Profile in the VPN Settings page on the mobile device. An auto-connect feature re-establishes the connection to the VPN Gateway should the connection be unintentionally lost. This feature may be configured by the user on the VPN settings page.

TOE Security Functional Requirements addressed: FTP_ITC.1.

8 ACRONYMS

8.1 ACRONYMS

The following acronyms are used in this ST:

Acronym	Definition
AES	Advanced Encryption Standard
BES	BlackBerry Enterprise Service
CA	Certification Authority
CBC	Cipher Block Chaining
CC	Common Criteria
CEM	Common Methodology for Information Technology Security Evaluation
CM	Configuration Management
CRL	Certificate Revocation List
CSP	Critical Security Parameter
DH	Diffie-Hellman
DN	Distinguished Name
DNS	Domain Name System
DRBG	Deterministic Random Bit Generator
DSS	Digital Signature Standard
EAP	Extensible Authentication Protocol
ECDSA	Elliptic Curve Digital Signature Algorithm
ECP	Elliptic Curve modulo Prime
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standards
FQDN	Fully qualified domain name
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
IKE	Internet Key Exchange
IT	Information Technology
IPsec	Internet Protocol Security

Acronym	Definition
MAC	Message Authentication Code
MODP	Modular Exponential
MSCHAP	Microsoft Challenge-Handshake Authentication Protocol
NAT	Network Address Translation
NIAP	National Information Assurance Partnership (US)
NIST	National Institute of Standards and Technology
N/A	Not Applicable
IT	Information Technology
IP	Internet Protocol
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OS	Operating System
OTA	Over-the-air
POST	Power-on self test
PP	Protection Profile
PRF	Pseudo Random Function
PUB	Publication
RBG	Random Bit Generator
RFC	Request for Comments
RSA	Rivest, Shamir and Adleman
SA	Security Association
SBGSE	Certicom Security Builder Government Solution Edition
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SP	Special Publication
SPD	Security Policy Database
ST	Security Target
TLS	Transport Layer Security

Acronym	Definition
TOE	Target of Evaluation
TSF	TOE Security Functionality
URL	Universal Resource Locator
VPN	Virtual Private Network

Table 18 – Acronyms