NetMotion Mobility® 11.0
Security Target

Evaluation Assurance Level (EAL): EAL4+

Doc No: 1905-000-D102

Version: 1.8

29 September 2017

NetMotion Wireless, Inc.
701 N 34th Street
Seattle, Washington, USA
98103

Prepared by:

EWA-Canada
1223 Michael Street
Ottawa, Ontario, Canada
K1J7T2

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 4
2 CONFORMANCE CLAIMS 9
2.1 COMMON CRITERIA CONFORMANCE CLAIM 9
2.2 ASSURANCE PACKAGE CLAIM 9
2.3 PROTECTION PROFILE CONFORMANCE CLAIM 9
3 SECURITY PROBLEM DEFINITION 10
3.1 THREATS 10
3.2 ORGANIZATIONAL SECURITY POLICIES 10
3.3 ASSUMPTIONS 10
4 SECURITY OBJECTIVES 12
4.1 SECURITY OBJECTIVES FOR THE TOE 12
4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT 13
4.3 SECURITY OBJECTIVES RATIONALE 13
5 EXTENDED COMPONENTS DEFINITION 20
5.1 SECURITY FUNCTIONAL REQUIREMENTS 20
5.2 SECURITY ASSURANCE REQUIREMENTS 20
6 SECURITY REQUIREMENTS 21
6.1 CONVENTIONS 21
6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS 21
6.3 DEPENDENCY RATIONALE 30
6.4 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE 31
6.5 TOE SECURITY ASSURANCE REQUIREMENTS 34
7 TOE SUMMARY SPECIFICATION 37
7.1 TOE SECURITY FUNCTIONS 37

8 TERMINOLOGY AND ACRONYMS 45
8.1 TERMINOLOGY 45
8.2 ACRONYMS 45

LIST OF TABLES

Table 1 – Cryptographic Modules Used by the TOE 3
Table 2 –TOE-Supporting Hardware and Software 5
Table 3 –Operational Environment Hardware and Software 6
Table 4 – Logical Scope of the TOE 7
Table 5 – Threats 10
Table 6 – Assumptions 11
Table 7 – Security Objectives for the TOE 12
Table 8 – Security Objectives for the Operational Environment 13
Table 9 – Mapping Between Objectives, Threats, and Assumptions 14
Table 10 – Summary of Security Functional Requirements (SFRs) 22
Table 11 – Cryptographic Operations (Windows) 25
Table 12 – Cryptographic Operations (iOS and macOS) 26
Table 13 – Cryptographic Operations (Android) 27
Table 14 – Management of TSF Data 29
Table 15 – Functional Requirement Dependencies 31
Table 16 – Mapping of SFRs to Security Objectives 32
Table 17 – Functional Security Requirements Rationale 34
Table 18 – EAL 4 Assurance Requirements 36
Table 19 – Activity Log Information 38
Table 20 – Event Log Information 38
Table 21 – Cryptographic Module Validation Program Certificates 39
Table 22 – Cryptographic Primitive Usage 41
Table 23 – Terminology 45
Table 24 – Acronyms 47

LIST OF FIGURES

Figure 1 – NetMotion Mobility 11.0 3
Figure 2 – NetMotion Mobility 11.0 TOE Boundary 4

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 Security Target reference, the Target of Evaluation (TOE) reference, the TOE overview and the TOE description.

Section 2, Conformance Claims, describes how the ST conforms to the Common Criteria and Packages. The ST does not conform to a Protection Profile (PP).

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 Information Technology environment.

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

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

1.2 SECURITY TARGET REFERENCE

ST Title:	NetMotion Mobility® 11.0 Security Target
ST Version:	1.8
ST Date:	29 September 2017

1.3 TOE REFERENCE

TOE Identification:	NetMotion Mobility® 11.0 NetMotion Mobility 11.0 Server (11.04.21384)

	NetMotion Mobility 11.0 Client for Windows 8.1 (11.04.21384)
	NetMotion Mobility 11.0 Client for Windows 10 (11.04.21384)
	NetMotion Mobility 11.0 Client for Android (11.04.21376)
	NetMotion Mobility 11.0 Client for macOS (11.04.21579)
	NetMotion Mobility 11.0 Client for iOS (11.04.21379)
TOE Developer:	NetMotion Wireless, Inc.
TOE Type:	Virtual Private Network (VPN)

1.4 TOE OVERVIEW

The TOE is NetMotion Mobility 11.0, a client/server-based software Virtual Private Network (VPN) that secures communications between the enterprise network and the mobile environment. NetMotion Mobility allows TCP/IP network applications to operate reliably and confidentially, without modification, over wireless connections. NetMotion Mobility ensures that only authorized users are granted access to protected resources, and that all communications are protected using validated cryptographic standards.

When a mobile device goes out of range or suspends operation, NetMotion Mobility maintains the session status and resumes the protected session when the device returns to service. If the mobile device returns to service at a different point on the network or connects from a new location, the Mobility server relays data to the new location, even if it is on a different subnet or a different network.

The TOE is managed via the Mobility Console, which is a web-based configuration and management application used by an administrator to configure settings, monitor server status and client connections, monitor activity or event logs[1], and troubleshoot problems. The Mobility Console is part of the Mobility Server and is accessed via a web browser. Figure 1 shows the TOE components.

[1] An event log is a binary file updated by any Mobility component that logs event messages.

Figure 1 – NetMotion Mobility 11.0

The TOE calls cryptographic modules implemented within the server (Windows Server 2012 R2) and client platforms (Windows 8.1, Windows 10, iOS 10, macOS 10.12 and Android 6.0) to perform cryptographic operations. The implementation of the cryptographic primitives and related key management are not covered in the evaluation. These cryptographic modules have been Federal Information Processing Standards (FIPS) 140-2 validated by the National Institute of Standards and Technology (NIST) in the United States of America (USA) and Communications Security Establishment (CSE) in Canada. The certificate numbers are as follows:

Cryptographic Module	FIPS 140-2 Cryptographic Module Validation Program Certificate Number
Windows 8.1 and Windows Server 2012 R2, 64 bit	2356
Windows 10	2605
Apple iOS 10	2827
Apple macOS 10.12	2832
OpenSSL FIPS Object Module (on Android)	2398

Table 1 – Cryptographic Modules Used by the TOE

The TOE is a software only TOE.

1.5 TOE DESCRIPTION

1.5.1 Physical Scope

The TOE is the NetMotion Mobility 11.0 and is comprised of the NetMotion Mobility 11.0 Server and NetMotion Mobility 11.0 Clients for Windows and Android and NetMotion Mobility 11.0 Client for iOS and macOS. Figure 2 shows the TOE Boundary.

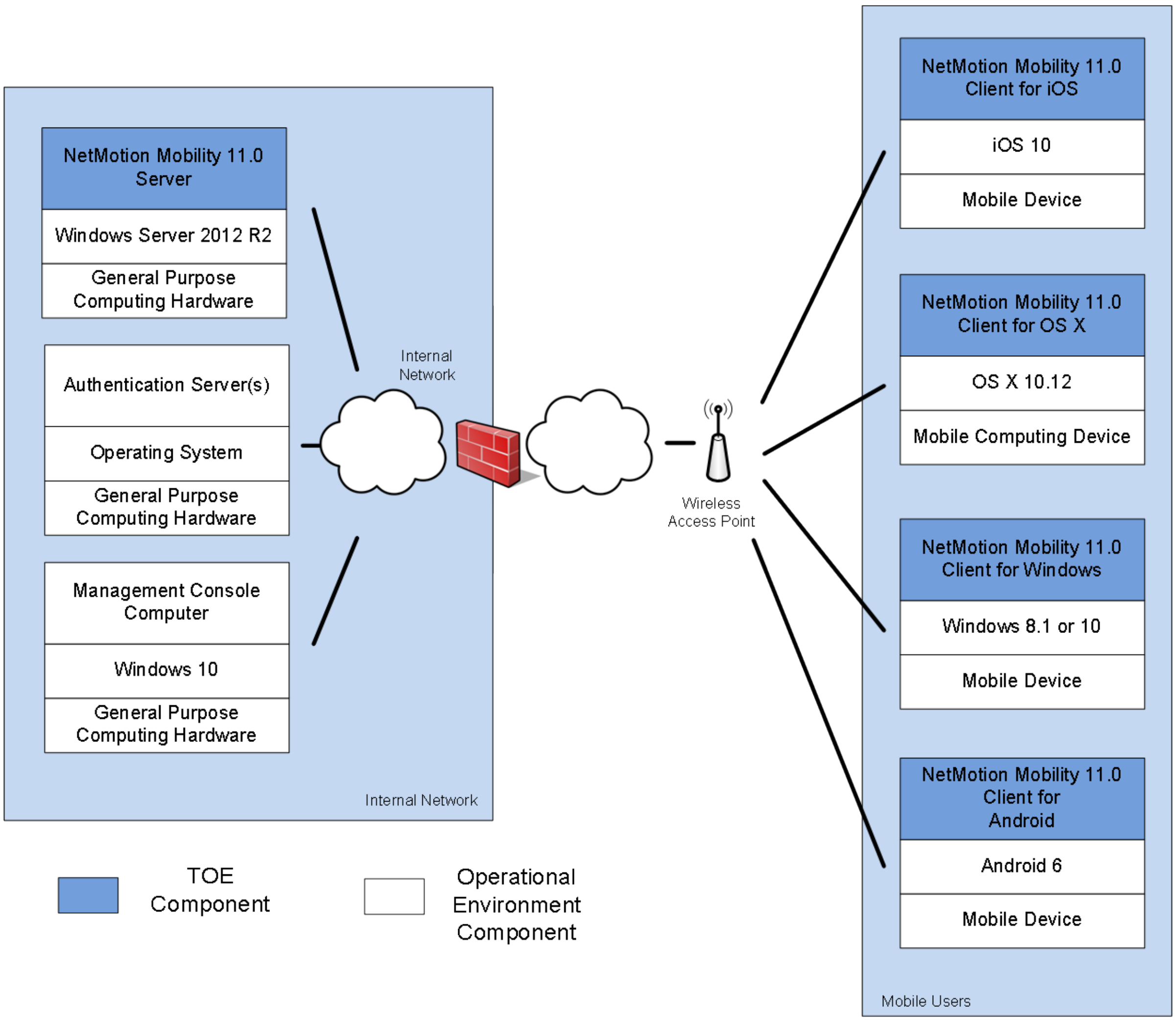

Figure 2 – NetMotion Mobility 11.0 TOE Boundary

The TOE consists of the following components:

- NetMotion Mobility 11.0 Server
- NetMotion Mobility 11.0 Client for Windows
- NetMotion Mobility 11.0 Client for Android
- NetMotion Mobility 11.0 Client for iOS
- NetMotion Mobility 11.0 Client for macOS

1.5.1.1 TOE Guidance

The TOE includes the following guidance documentation:

- http://www.netmotionwireless.com/support/docs/MobilityXG/1100/help/mobilityhelp.htm#page/Mobility%20Server/intro.01.02.html
- NetMotion Mobility 11.0 Common Criteria Guidance Supplement

1.5.2 Required Non-TOE Components

The following software and hardware components are required for operation of the TOE in the evaluated configuration.

TOE Component	Required Software	Required Hardware
NetMotion Mobility 11.0 Server	Windows Server 2012 R2	General Purpose Computing Platform with x64-compatible dual-core processor, 2.0 GHz, 4 GB RAM
NetMotion Mobility 11.0 Windows Client	Windows 8.1	General Purpose Computing Platform
NetMotion Mobility 11.0 Windows Client	Windows 10	General Purpose Computing Platform
NetMotion Mobility 11.0 Android Client	Android version 6.0	General Purpose Android device
NetMotion Mobility 11.0 iOS Client	iOS 10	iPad, iPhone (Apple device with A7 to A9X hardware)
NetMotion Mobility 11.0 macOS Client	macOS 10.12	Mac mini, iMac, MacPro or MacBook hardware

Table 2 –TOE-Supporting Hardware and Software

Operational Environment Component	Required Software	Required Hardware
Management Console Computer	Windows 10 Microsoft Edge 38	General Purpose Computing Platform
RADUIS Authentication server(s) (provided as a service to the TOE)	Software supporting PEAP MS-CHAPv2 and EAP-TLS	General Purpose Computing hardware that meets the requirements of the

Operational Environment Component	Required Software	Required Hardware
		authentication server
Public Key Infrastructure (provided as a service to the TOE)	Dependent upon the authentication server	General Purpose Computing hardware that meets the requirements of the authentication server
Firewall	Dependent upon the selected appliance	General Purpose Firewall appliance

Table 3 –Operational Environment Hardware and Software

The operational environment must provide an authentication server. For the purposes of the evaluation, end-user authentication is performed using a RADIUS based authentication server and an implementation of the PEAP MS-CHAPv2 protocol, or using EAP-TLS.

NTLMv2 over HTTPS is used to authenticate administrative users to the Mobility Console for the purposes of managing the TOE. Both the authentication server and the management console computer are located on a trusted internal network. It should be noted that physical access to the management console computer must be limited to authorized personnel. A firewall must be in place to ensure that the Mobility Console can only be accessed from the internal network. This configuration allows NTLMv2 over HTTPS to provide sufficient security to support the authentication of administrative users.

Support for NTLM authentication of administrators is provided by the Windows infrastructure in the operational environment. In order to support the implementation of PEAP MS-CHAPv2 and EAP-TLS, one or more RADIUS authentication servers are required in the operational environment. Additionally, the use of EAP-TLS requires that users be issued certificates. These may be issued in any number of ways, either from within the organization, or externally. Ultimately, a Public Key Infrastructure is required in the operational environment to issue certificates.

In the evaluated configuration, a firewall must be implemented to protect the internal resources, which include the Mobility Server, and authentication server, from the external network. The only access allowed from the wireless access point to the internal network is through a Mobility protected VPN connection. Note that for authentication, an encrypted tunnel is established between the Mobility Client and the authentication server. The ciphersuites are negotiated between the client operating system and the authentication server. Once authenticated, a VPN tunnel is established between the Mobility Client and the Mobility Server. The ciphersuites used are negotiated between the Mobility Client and the Mobility Server.

The connections between the Mobility Server and the authentication servers are protected by the operational environment. For example, these components may be physically secured in a server room, and may communicate on the same Local Area Network (LAN) segment.

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 4 summarizes the logical scope of the TOE.

Functional Classes	Description
Security Audit	The TOE generates audit records for security events. Only those roles that have been granted specific access to the audit trail have the ability to view the audit trail. For the purpose of this evaluation, only users in the Administrator role have been granted this access.
Cryptographic Support	The TOE supports secure communications between TOE components. This encrypted traffic prevents modification and disclosure of user information. Cryptographic functionality is provided by FIPS 140-2 validated modules in the operating systems of the server and client components. Cryptography also supports the authentication of users.
User Data Protection	The TOE provides an information flow security policy. The security policy limits access to internal protected resources based on policy settings. The TOE provides a secure connection between mobile users and the internal network. Traffic is protected from disclosure and modification.
Identification and Authentication	The TOE verifies that users are identified and authenticated before permitting access. Additionally, administrators must be identified and authenticated before access to administrative functions is permitted.
Security Management	The TOE provides security management functions through the Mobility Console. Administrators manage users, information flow policy, and audit.
Protection of the TSF	Reliable timestamps are provided in support of TOE functions, including the generation of audit records.

Table 4 – Logical Scope of the TOE

1.5.4 Functionality Excluded from the Evaluated Configuration

The following features are excluded from this evaluation:

- Analytics Module
- Web Services API
- Unattended authentication mode
- Mobility Client API
- Mobility Event Viewer (on the Mobility Client)
- Mobility Network Access Control Module

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 5, April 2017
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements CCMB-2012-09-003, Version 3.1, Revision 5, April 2017

As follows:

- CC Part 2 conformant
- CC Part 3 conformant

The Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017 [CEM] has to be taken into account.

2.2 ASSURANCE PACKAGE CLAIM

This Security Target claims conformance to Evaluation Assurance Level (EAL) 4 augmented with ALC_FLR.1 Basic flaw remediation.

2.3 PROTECTION PROFILE CONFORMANCE CLAIM

The TOE for this ST does not claim conformance with any Protection Profile.

3 SECURITY PROBLEM DEFINITION

3.1 THREATS

Table 5 lists the threats addressed by the TOE. The threat agents to the TOE are considered to be unauthorized users with public knowledge of how the TOE operates and who possess the skills and resources to alter TOE configuration settings, or parameters, or both. The threat agents do not have physical access to the TOE.

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

Threat	Description
T.ACCESS	An unauthorized individual on an external network may access and exploit protected application data resources on an internal network.
T.NOAUTH	An unauthorized individual may gain access to the TOE security management functions and use this to allow unauthorized access to application data protected by the TOE.
T.SENSDATA	An unauthorized individual may be able to view or alter sensitive application data passed between a client and a server.
T.UNAUTH	An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and application data.

Table 5 – Threats

Application data refers to the data that passes between the end user and the server when the end user accesses the application remotely. The application may be any client/server application written for an IP network. Application data includes, but is not limited to, documents, spreadsheets, images, and XML data. Application data is considered to be at risk when it is transmitted on an external public network.

3.2 ORGANIZATIONAL SECURITY POLICIES

There are no Organizational Security Policies applicable to this TOE.

3.3 ASSUMPTIONS

The assumptions required to ensure the security of the TOE are listed in Table 6.

Assumptions	Description
A.AUTH	The operational environment provides authentication services to the TOE.
A.CERTIFICATE	A Public Key Infrastructure is available to issue certificates to

Assumptions	Description
	users and servers. Root trust exists for the certificated chain.
A.INTERNAL	The internal network and its assets are protected from unauthorized access. A firewall must be in place to ensure that only authorized connections from Mobility Clients to the Mobility Server are permitted.
A.MANAGE	A management console computer is available on the internal protected network for the purposes of managing the TOE. Administrators will access the Mobility Console only from a management console computer on the internal network.
A.NOEVIL	Authorized administrators are non-hostile and follow all administrator guidance.
A.OS	The services, including cryptographic services, provided by the underlying operating system work correctly, and the operating system does not introduce any negative side effects to the TSF.
A.PHYSEC	The server resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.
A.SECCOM	The communications between the TOE (Mobility server) and the authentication services, and between the TOE (Mobility server) and the management console computer are secured on an internal network.

Table 6 – 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

This section identifies and describes the security objectives that are to be addressed by the TOE.

Security Objective	Description
O.ACCESS	The TOE must regulate the flow of information from users on an external network to resources on an internal network, and must ensure that rules for granting access are enforced.
O.ACCOUN	Users must be accountable for their use of the TOE and authorized administrators must be accountable for their use of TOE security management functions.
O.AUDREC	The TOE must record security relevant events, the date and time the events occurred, and provide a means to read this information.
O.ENCRYP	The TOE must protect the confidentiality and integrity of information that flows between distributed TOE components.
O.IDENTIFY	The TOE must ensure that users are identified and authenticated using an approved authentication mechanism prior to allowing access to TOE functions and data.
O.SECFUN	The TOE must provide functionality that enables authorized administrators authenticated in the operational environment to manage the security functions of the TOE, and must ensure that only authorized administrators are able to access this functionality.

Table 7 – Security Objectives for the TOE

4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

This section 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.ADMTRA	Authorized administrators are carefully screened during the selection process. All selected administrators are trained to appropriately install, configure, and maintain the TOE in its evaluated configuration according to the TOE guidance documentation.
OE.CONSOLEOS	The operating system on the management console computer that supports access to the Mobility Console will provide user authentication services.
OE.IDAUTH	The operational environment provides authentication services based on standard PEAP MS-CHAPv2 and EAP-TLS protocols, supported by trusted certificates.
OE.INTNETWORK	The operational environment will protect the internal network and its assets from access by unauthorized external entities through use of a firewall.
OE.PHYSEC	Those responsible for the TOE must ensure that those parts of the TOE critical to the enforcement of security are protected from any physical attack.
OE.SECCOM	The operational environment will protect the communications between the TOE (Mobility server) and the authentication servers, and between the TOE (Mobility Console on the Mobility server) and the management console computer on a protected internal network.
OE.SUPPORT	The underlying operating system provides the correct services (including cryptographic services) to the TOE without introducing negative side effects on the TSF.

Table 8 – Security Objectives for the Operational Environment

4.3 SECURITY OBJECTIVES RATIONALE

The following table maps the security objectives to the assumptions and threats identified for the TOE.

	T.ACCESS	T. NOAUTH	T.SENSDATA	T.UNAUTH	A.AUTH	A.CERTIFICATE	A.INTERNAL	A.MANAGE	A.NOEVIL	A.OS	A.PHYSEC	A.SECCOM
O.ACCESS	X											
O.ACCOUN		X										
O.AUDREC		X										
O.ENCRYP			X									
O.IDENTIFY	X	X		X								
O.SECFUN				X								
OE.ADMTRA									X			
OE.CONSOLEOS				X								
OE.IDAUTH		X			X	X						
OE.INTNETWQRK							X					
OE.PHYSEC											X	
OE.SECCOM								X				X
OE.SUPPORT										X		

Table 9 – Mapping Between Objectives, Threats, and Assumptions

4.3.1 Security Objectives Rationale Related to Threats

The security objectives rationale related to threats traces the security objectives for the TOE and the operational environment back to the threats addressed by the TOE.

Threat: T.ACCESS	An unauthorized individual on an external network may access and exploit protected resources on an internal network.	
Objectives:	O.ACCESS	The TOE must regulate the flow of information from users on an external network to resources on an internal network, and must ensure that rules for granting access are enforced.
	O.IDENTIFY	The TOE must ensure that users are identified and authenticated using an approved authentication mechanism prior to allowing access to TOE functions and data.

Rationale:	O.ACCESS counters this threat by ensuring that the rules for access to protected resources are enforced. O.IDENTIFY ensures that users are identified and authenticated before gaining access to the TOE, thereby mitigating the risk of unauthorized access to protected resources.	

Threat: T.NOAUTH	An unauthorized individual may gain access to the TOE security management functions and use this to allow unauthorized access to information protected by the TOE.	
Objectives:	O.ACCOUN	Users must be accountable for their use of the TOE and authorized administrators must be accountable for their use of TOE security management functions.
	O.AUDREC	The TOE must record security relevant events, the date and time the events occurred, and provide a means to read this information.
	O.IDENTIFY	The TOE must ensure that users are identified and authenticated using an approved authentication mechanism prior to allowing access to TOE functions and data.
	OE.IDAUTH	The operational environment provides authentication services based on standard PEAP MS-CHAPv2 and EAP-TLS protocols, supported by trusted certificates.
Rationale:	O.ACCOUN ensures that users are held accountable for their use of the TOE. O.AUDREC ensures that the audit records supporting accountability are recorded, time stamped, and may be read using the TOE. O.IDENTIFY ensures that users are identified and authenticated prior to being allowed access to TOE functions. OE.IDAUTH ensures that the operational environment identifies and authenticates users, and provides this information to the TOE in support of the TOE objectives.	

Threat: T.SENSDATA	An unauthorized individual may be able to view or alter sensitive data passed between a client and a server.	
Objectives:	O.ENCRYP	The TOE must protect the confidentiality and integrity of information that flows between

		distributed TOE components.
Rationale:	O.ENCRYP ensures that data that flows between components may not be read or altered by an unauthorized individual.	

Threat: T.UNAUTH	An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.	
Objectives:	O.IDENTIFY	The TOE must ensure that users are identified and authenticated using an approved authentication mechanism prior to allowing access to TOE functions and data.
	O.SECFUN	The TOE must provide functionality that enables authorized administrators authenticated in the operational environment to manage the security functions of the TOE, and must ensure that only authorized administrators are able to access this functionality.
	OE.CONSOLEOS	The operating system on the management console computer that supports access to the Mobility Console will provide user authentication services.
Rationale:	O.IDENTIFY ensures that users are identified and authenticated using an approved authentication mechanism before gaining access, thereby mitigating the risk of bypass of TOE security. O.SECFUN ensures that only authorized administrators are able to access TOE security management functionality. OE.CONSOLEOS ensures that the operational environment provides authentication services in support of Mobility Console access.	

4.3.2 Security Objectives Rationale Related to Assumptions

The security objectives rationale related to assumptions traces the security objectives for the operational environment back to the assumptions for the TOE's operational environment.

Assumption: A.AUTH	The operational environment provides authentication services to the TOE.	
Objectives:	OE.IDAUTH	The operational environment provides authentication services based on standard PEAP MS-CHAPv2 and EAP-TLS protocols, supported by trusted certificates.

Rationale:	OE.IDAUTH supports this assumption by ensuring that the operational environment provides authentication services.	

Assumption: A.CERTIFICATE	A Public Key Infrastructure is available to issue certificates to users and servers. Root trust exists for the certificated chain.	
Objectives:	OE.IDAUTH	The operational environment provides authentication services based on standard PEAP MS-CHAPv2 and EAP-TLS protocols, supported by trusted certificates.
Rationale:	OE.IDAUTH supports this assumption by ensuring that trusted certificates are available to support authentication.	

Assumption: A.INTERNAL	The internal network and its assets are protected from unauthorized access. A firewall must be in place to ensure that only authorized connections from Mobility Clients to the Mobility Server are permitted.	
Objectives:	OE.INTNETWORK	The operational environment will protect the internal network and its assets from access by unauthorized external entities through use of a firewall.
Rationale:	OE.INTNETWORK supports this assumption by ensuring that the operational environment includes a firewall to protect the internal network and its assets against unauthorized access.	

Assumption: A.MANAGE	A management console computer is available on the internal protected network for the purposes of managing the TOE. Administrators will access the Mobility Console only from a management console computer on the internal network.	
Objectives:	OE.SECCOM	The operational environment will protect the communications between the TOE (Mobility server) and the authentication servers, and between the TOE (Mobility Console on the Mobility server) and the management console computer on a protected internal network.
Rationale:	OE.SECCOM supports this assumption by ensuring that communications between the TOE and administrators using the management console computer and between the TOE and the authentication server are protected.	

Assumption: A.NOEVIL	Authorized administrators are non-hostile and follow all administrator guidance.	
Objectives:	OE.ADMTRA	Authorized administrators are carefully screened during the selection process. All selected administrators are trained to appropriately install, configure, and maintain the TOE in its evaluated configuration according to the TOE guidance documentation.
Rationale:	OE.ADMTRA upholds the assumption A.NOEVIL by providing carefully selected administrators with the training required to be able to follow the administrative guidance.	

Assumption: A.OS	The services, including cryptographic services, provided by the underlying operating system work correctly, and the operating system does not introduce any negative side effects to the TSF.	
Objectives:	OE.SUPPORT	The underlying operating system provides the correct services (including cryptographic services) to the TOE without introducing negative side effects on the TSF.
Rationale:	OE.SUPPORT upholds the assumption A.OS by providing reliable services, including cryptographic services, to the TOE without the introduction of negative side effects.	

Assumption: A.PHYSEC	The server resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	
Objectives:	OE.PHYSEC	Those responsible for the TOE must ensure that those parts of the TOE critical to the enforcement of security are protected from any physical attack.
Rationale:	OE.PHYSEC ensures that TOE server resources are protected from physical attack, thereby supporting the assumption of protection of the resources.	

Assumption: A.SECCOM	The communications between the TOE (Mobility server) and the authentication services, and between the TOE (Mobility server) and the management console computer are secured on an internal network.

Objectives:	OE.SECCOM	The operational environment will protect the communications between the TOE (Mobility server) and the authentication servers, and between the TOE (Mobility Console on the Mobility server) and the management console computer on a protected internal network.
Rationale:	OE.SECCOM ensures that communications between the TOE (Mobility server) and the authentication servers, and between the TOE (Mobility Console on the Mobility server) and the management console computer are appropriately protected, as required by the assumption A.SECCOM.	

5 EXTENDED COMPONENTS DEFINITION

5.1 SECURITY FUNCTIONAL REQUIREMENTS

This ST does not include extended Security Functional Requirements.

5.2 SECURITY ASSURANCE REQUIREMENTS

This ST does not include extended Security Assurance Requirements.

6 SECURITY REQUIREMENTS

Section 6 provides security functional and assurance requirements to be satisfied by the TOE. These requirements consist of functional components from Part 2 of the CC, and an EAL that contains assurance components from Part 3 of the CC.

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 **bolding** 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., 'FDP_ACC.1(1), Subset access control (administrators)' and 'FDP_ACC.1(2) Subset access control (devices)'.

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 defined in Section 5, summarized in Table 10.

Class	Identifier	Name
Security Audit (FAU)	FAU_GEN.1	Audit data generation
	FAU_SAR.1	Audit review
Cryptographic Support (FCS)	FCS_CKM.1(1)	Cryptographic key generation (3DES and AES)
	FCS_CKM.1(2)	Cryptographic key generation (RSA)
	FCS_CKM.1(3)	Cryptographic key generation (ECDH)
	FCS_CKM.1(4)	Cryptographic key generation (DH)
	FCS_CKM.4	Cryptographic key Destruction (Windows, iOS, macOS and Android)

Class	Identifier	Name
	FCS_COP.1(1)	Cryptographic operation (Windows)
	FCS_COP.1(2)	Cryptographic operation (iOS and macOS)
	FCS_COP.1(3)	Cryptographic operation (Android)
User Data Protection (FDP)	FDP_IFC.1	Subset information flow control
	FDP_IFF.1	Simple security attributes
	FDP_ITT.1	Basic internal transfer protection
Identification and Authentication (FIA)	FIA_UAU.2	User authentication before any action
	FIA_UAU.5	Multiple authentication mechanisms
	FIA_UID.2	User identification before any action
Security Management (FMT)	FMT_MTD.1	Management of TSF data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.1	Security roles
Protection of the TSF (FPT)	FPT_STM.1	Reliable time stamps

Table 10 – Summary of Security Functional Requirements (SFRs)

6.2.1 Security Audit (FAU)

6.2.1.1 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the [not specified] level of audit; and

c) [*establishment of an encrypted session, client connection events, administrator logon to the Mobility Console*].

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*device name*].

6.2.1.2 FAU_SAR.1 Audit review

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.1.1 The TSF shall provide [*Administrators*] with the capability to read [*all audit information*] from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

6.2.2 Cryptographic Support (FCS)

6.2.2.1 FCS_CKM.1(1) Cryptographic key generation (3DES and AES)

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 TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*Counter mode Deterministic Random Bit Generator (CTR-DRBG)*] and specified cryptographic key sizes [*128, 168 or 256 bit*] that meet the following: [*NIST Special Publication 800-90A*].

6.2.2.2 FCS_CKM.1(2) Cryptographic key generation (RSA)

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 TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*Deterministic Random Bit Generator (DRBG)*] and specified cryptographic key sizes [*2048 bit*] that meet the following: [*NIST Special Publication 800-56A (Windows), NIST Special Publication 800-90A (iOS and macOS), ANSI 9.31 (Android)*].

6.2.2.3 FCS_CKM.1(3) Cryptographic key generation (ECDH)

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(3) The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*Random Bit*

Generator (RBG)] and specified cryptographic key sizes [*P-256 and P-521*[2]] that meet the following: [*FIPS 186-4 as referenced in NIST Special Publication 800-56A*].

6.2.2.4 FCS_CKM.1(4) Cryptographic key generation (DH)

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(4) The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*Random Bit Generator (RBG)*] and specified cryptographic key sizes [*112,128 bits of security strength*] that meet the following: [*FIPS 186-4 as referenced in NIST Special Publication 800-56A*].

6.2.2.5 FCS_CKM.4 Cryptographic key destruction (Windows, iOS, macOS and Android)

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.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [*zeroization*] that meets the following: [*FIPS 140-2 zeroization standards*].

6.2.2.6 FCS_COP.1(1) Cryptographic operation (Windows)

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 The TSF shall perform [*the cryptographic operations in Table 11*] in accordance with a specified cryptographic algorithm [*in Table 11*] and cryptographic key sizes [*in Table 11*] that meet the following: [*list of standards in Table 11*].

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
Encryption and Decryption	AES CBC AES GCM	128, 256	2832 (8.1) 3497 (10 and Server 2012)	FIPS 197 SP 800-38A SP 800-38D
Secure Hash	Secure Hash Algorithm	Not applicable	2373 (8.1) 2886 (10 and	FIPS 180-3

[2] P-256 is used with EAP-TLS authentication and National Institute of Standards and Technology (NIST) curve P-521 is used by the Mobility encrypted tunnel.

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
	SHA-256, SHA-384		Server 2012)	
Keyed-Hash Message Authentication Code (HMAC)	HMAC	256, 384	1773 (8.1) 2233 (10 and Server 2012)	FIPS 198
Asymmetric cryptography	RSA (RSASSA-PKCS1_V1_5)	2048	1493 (8.1) 1783 (10 and Server 2012)	PKCS#1 v1.5 FIPS 186-4
Encryption and Decryption	3DES CBC	168	1692 (8.1) 1969 (10 and Server 2012)	FIPS 46-3, ISO 18033-3 SP 800-38A

Table 11 – Cryptographic Operations (Windows)

Application note: FCS_COP.1(1) applies to TOE components installed on Windows 8.1, and Windows 10 and Windows Server 2012 R2 using the Microsoft Kernel Mode Cryptographic Primitives Library (CNG.SYS) Software Version: 6.3.9600, Cryptographic Module Validation Program (CMVP) certificate number 2356, and on Windows 10 using the Microsoft Kernel Mode Cryptographic Primitives Library (CNG.SYS) Software Version: 10.0.10240, CMVP certificate number 2605. AES GCM is only used in conjunction with user authentication.

6.2.2.7 FCS_COP.1(2) Cryptographic operation (iOS and macOS)

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 The TSF shall perform [*the cryptographic operations in Table 12*] in accordance with a specified cryptographic algorithm [*in Table 12*] and cryptographic key sizes [*in Table 12*] that meet the following: [*list of standards in Table 12*].

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
Encryption and Decryption	AES CBC AES GCM	128, 256	For iOS: 4156, 4165, and 4180 For macOS: 4193, 4200, 4209, 4217 and 4225	FIPS 197 SP 800-38A SP 800-38D
Secure Hash	Secure Hash Algorithm	Not applicable	For iOS: 3421 and	FIPS 180-3

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
	SHA-256		3431 For macOS: 3446, 3454, 3462 and 3498	
Keyed-Hash Message Authentication Code (HMAC)	HMAC	256	For iOS: 2723 and 2733 For macOS: 2748, 2756, 2764 and 2797	FIPS 198
Asymmetric cryptography	RSA (RSASSA-PKCS1_V1_5)	2048	For iOS: 2264 For macOS: 2277	PKCS#1 v1.5 FIPS 186-4
Encryption and Decryption	3DES CBC	168	For iOS: 2272 For macOS: 2285	FIPS 46-3, ISO 18033-3 SP 800-38A

Table 12 – Cryptographic Operations (iOS and macOS)

Application note: FCS_COP.1(2) applies to TOE components installed on iOS 10 using the Apple iOS CoreCrypto Module, v7.0, CMVP certificate number 2827, or the Apple macOS CoreCrypto Module, v7.0, CMVP certificate number 2832. AES GCM is only used in conjunction with user authentication.

6.2.2.8 FCS_COP.1(3) Cryptographic operation (Android)

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 The TSF shall perform [*the cryptographic operations in Table 13*] in accordance with a specified cryptographic algorithm [*in Table 13*] and cryptographic key sizes [*in Table 13*] that meet the following: [*list of standards in Table 13*].

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
Encryption and Decryption	AES CBC AES GCM	128, 256	3751	FIPS 197 SP 800-38A SP 800-38D
Secure Hash	Secure Hash Algorithm SHA-256, SHA-	Not applicable	3121	FIPS 180-3

Operation	Algorithm	Key Size (bits)	CAVP Certificate	Standard
	384			
Keyed-Hash Message Authentication Code (HMAC)	HMAC	256, 384	2452	FIPS 198
Asymmetric cryptography	RSA (RSASSA-PKCS1_V1_5)	2048	1928	PKCS#1 v1.5 FIPS 186-4
Encryption and Decryption	3DES CBC	168	2086	FIPS 46-3, ISO 18033-3 SP 800-38A

Table 13 – Cryptographic Operations (Android)

Application note: FCS_COP.1(3) applies to TOE components installed on Android 6.0 using the OpenSSL FIPS Object Module SE, Software Version 2.0.12 CMVP certificate number 2398. AES GCM is only used in conjunction with user authentication.

6.2.3 User Data Protection (FDP)

6.2.3.1 FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1 The TSF shall enforce the [*secure information flow control SFP*] on [*Subjects: Mobility client users*
Information: application data[3]
Operations: send, receive].

6.2.3.2 FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control

FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1 The TSF shall enforce the [*secure information flow control SFP*] based on the following types of subject and information security attributes: [*Subjects: Mobility client users*
Subject attributes: identity, authentication status
Information: application data
Information attributes: none].

[3] Application data refers to the data that passes between the end user and the server when the end user accesses the application remotely.

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [*The mobility client user is successfully identified and authenticated*, *and is an authorized user in the NetMotion Mobility implementation*].

FDP_IFF.1.3 The TSF shall enforce the [*no additional information flow control SFP rules*].

FDP_IFF.1.4 The TSF shall explicitly authorise an information flow based on the following rules: [*no additional rules*].

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following rules: [*no additional rules*].

6.2.3.3 FDP_ITT.1 Basic internal transfer protection

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

FDP_ITT.1.1 The TSF shall enforce the [*secure information flow control SFP*] to prevent the [disclosure, modification] of user data when it is transmitted between physically-separated parts of the TOE.

Application Note: 'User data' refers to the application data described in FDP_IFC.1 and FDP_IFF.1.

6.2.4 Identification and Authentication (FIA)

6.2.4.1 FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

6.2.4.2 FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.5.1 The TSF shall provide [*PEAP MS-CHAPv2, EAP-TLS*] to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the [*the authentication mechanism utilized for that particular implementation*].

6.2.4.3 FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

6.2.5 Security Management (FMT)

6.2.5.1 FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1 The TSF shall restrict the ability to [perform the functions listed in Table 14] the [*data listed in Table 14*] to [*Administrators*].

Data	Allowed Actions
Secure Information Flow Control Data	Change_default, query, modify, delete
Audit Logs	Query, clear
User Account Attributes	Query, modify, delete

Table 14 – Management of TSF Data

6.2.5.2 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:
[*a. Create, delete, modify, and view information flow security policy data that permits user access;*
b. Review the audit logs; and
c. Query, modify, and delete user information].

6.2.5.3 FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles [*Administrator and Mobility Client User*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

6.2.6 Protection of the TSF (FPT)

6.2.6.1 FPT_STM.1 Reliable time stamps

Hierarchical To: No other components.

Dependencies: No dependencies.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

6.3 DEPENDENCY RATIONALE

Table 15 – Functional Requirement Dependencies 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	Dependencies	Dependency Satisfied / Rationale
FAU_GEN.1	FPT_STM.1	Satisfied.
FAU_SAR.1	FAU_GEN.1	Satisfied.
FCS_CKM.1(1)	FCS_CKM.2 or FCS_COP.1, and FCS_CKM.4	This dependency is satisfied by FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3) and FCS_CKM.4.
FCS_CKM.1(2)	FCS_CKM.2 or FCS_COP.1, and FCS_CKM.4	This dependency is satisfied by FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3) and FCS_CKM.4.
FCS_CKM.1(3)	FCS_CKM.2 or FCS_COP.1, and FCS_CKM.4	This dependency is satisfied by FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3) and FCS_CKM.4.
FCS_CKM.1(4)	FCS_CKM.2 or FCS_COP.1, and FCS_CKM.4	This dependency is satisfied by FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3) and FCS_CKM.4.
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	This dependency is satisfied by FCS_CKM.1(1), FCS_CKM.1(2), and FCS_CKM.1(3).
FCS_COP.1(1)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 and FCS_CKM.4	This dependency is satisfied by FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.1(3) and FCS_CKM.4.
FCS_COP.1(2)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 and FCS_CKM.4	This dependency is satisfied by FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.1(3) and FCS_CKM.4.
FCS_COP.1(3)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 and FCS_CKM.4	This dependency is satisfied by FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.1(3) and FCS_CKM.4.
FDP_IFC.1	FDP_IFF.1	Satisfied.

SFR	Dependencies	Dependency Satisfied / Rationale
FDP_IFF.1	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1 is included. FMT_MSA.3 is not included because there are no configurable security attributes associated with the secure information flow control SFP, and therefore, no default values provided by the TOE. The security attributes for the secure information flow control SFP are Mobility client user identity, which is provided by the user, and authentication status, which is provided by the authentication server.
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	This dependency is satisfied by FDP_IFC.1.
FIA_UAU.2	FIA_UID.1	This dependency is satisfied by FIA_UID.2, which is hierarchical to FIA_UID.1.
FIA_UAU.5	None	Not applicable.
FIA_UID.2	None	Not applicable.
FMT_MTD.1	FMT_SMR.1 and FMT_SMF.1	Satisfied.
FMT_SMF.1	None	Not applicable.
FMT_SMR.1	FIA_UID.1	This dependency is satisfied by FIA_UID.2, which is hierarchical to FIA_UID.1.
FPT_STM.1	None	Not applicable.

Table 15 – Functional Requirement Dependencies

6.4 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

The following Table provides a mapping between the SFRs and Security Objectives.

	O.ACCESS	O.ACCOUN	O.AUDREC	O.ENCRYP	O.IDENTIFY	O.SECFUN
FAU_GEN.1		X	X			
FAU_SAR.1			X			
FCS_CKM.1(1)				X		
FCS_CKM.1(2)				X		

	O.ACCESS	O.ACCOUN	O.AUDREC	O.ENCRYP	O.IDENTIFY	O.SECFUN
FCS_CKM.1(3)				X		
FCS_CKM.1(4)				X		
FCS_CKM.4				X		
FCS_COP.1(1)				X		
FCS_COP.1(2)				X		
FCS_COP.1(3)				X		
FDP_IFC.1	X					
FDP_IFF.1	X					
FDP_ITT.1				X		
FIA_UAU.2	X				X	X
FIA_UAU.5					X	
FIA_UID.2	X	X			X	X
FMT_MTD.1						X
FMT_SMF.1						X
FMT_SMR.1						X
FPT_STM.1		X	X			

Table 16 – Mapping of SFRs to Security Objectives

The following rationale traces each SFR back to the security objectives for the TOE.

SFR	Rationale
FAU_GEN.1	This requirement supports O.ACCOUN by ensuring that audit records are produced with sufficient detail to hold users accountable for their actions related to the use of the TOE, including use of the TOE security management functions. This requirement supports O.AUDREC by ensuring that security relevant events are recorded.
FAU_SAR.1	This requirement supports O.AUDREC by providing a means to read the records of security relevant events.

SFR	Rationale
FCS_CKM.1(1) FCS_CKM.1(2) FCS_CKM.1(3) FCS_CKM.1(4)	These requirements support O.ENCRYP by ensuring that cryptographic keys are appropriately generated to support the use of the encryption that provides confidentiality and integrity of the information that flows between distributed TOE components, and between the TOE and authentication services in the operational environment.
FCS_CKM.4	This requirement supports O.ENCRYP by ensuring that encryption keys are properly destroyed, mitigating the risk of improper use, and thereby supporting the confidentiality and integrity of the information that flows between distributed TOE components, and between the TOE and authentication services in the operational environment.
FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3)	These requirements support O.ENCRYP by providing the encryption services that ensure the confidentiality and integrity of the information that flows between distributed TOE components, and between the TOE and authentication services in the operational environment.
FDP_IFC.1	This requirement supports O.ACCESS by identifying the secure information flow control SFP between TOE users on an external network, to the application data resources on an internal network to which the TOE controls access.
FDP_IFF.1	This requirement identifies the user attributes and the secure information flow control rules. This requirement supports O.ACCESS by enforcing the rules for the information flow between TOE users on an external network, to the TOE controlled resources on an internal network.
FDP_ITT.1	This requirement supports O.ENCRYP by ensuring that the confidentiality and integrity of data is preserved as it passes between parts of the TOE.
FIA_UAU.2	This requirement supports O.IDENTIFY by ensuring that users are authenticated prior to being allowed access to TOE functions and data. In turn, the requirement supports O.ACCESS by ensuring that the information required to enforce the secure information flow control SFP is available. This requirement supports O.SECFUN by providing the authentication mechanism that ensures that only authorized administrators may access security management functionality.
FIA_UAU.5	This requirement supports O.IDENTIFY by ensuring that users are authenticated using an approved authentication mechanism before being granted access to TOE functions or data.

SFR	Rationale
FIA_UID.2	This requirement supports O.IDENTIFY by ensuring that users are identified prior to being allowed access to TOE functions and data. In turn, the requirement supports O.ACCESS by ensuring that the information required to enforce the secure information flow control SFP is available. It also supports O.ACCOUN to identify users so they may be held accountable for their use of the TOE. This requirement supports O.SECFUN by providing the identification mechanism that ensures that only authorized administrators may access security management functionality.
FMT_MTD.1	This requirement supports O.SECFUN by ensuring that only authorized administrators have access to security management functions.
FMT_SMF.1	This requirement supports O.SECFUN by providing the functionality that enables authorized administrators to manage the security functions of the TOE.
FMT_SMR.1	This requirement supports O.SECFUN by providing the roles that are used to ensure that only authorized administrators are able to access security management functions.
FPT_STM.1	This requirement supports O.ACCOUN by ensuring that audit records are produced with a reliable time stamp to hold users accountable for their actions related to the use of the TOE, including the use of the TOE security management functions. This requirement supports O.AUDREC by providing a time stamp for all auditable events.

Table 17 – Functional Security Requirements Rationale

6.5 TOE SECURITY ASSURANCE REQUIREMENTS

The TOE assurance requirements for this ST consist of the requirements corresponding to the EAL 4 level of assurance, as defined in the CC Part 3, augmented by the inclusion of Basic Flaw Remediation (ALC_FLR.1). EAL 4 was chosen because it is based upon good commercial development practices with thorough functional testing. EAL 4 provides customers with a moderate level of independently assured security in a conventional commercial TOE. The threat of malicious attacks is not greater than low, the security environment provides physical protection, and the TOE itself offers a very limited, purpose-built interface.

The assurance requirements are summarized in Table 18 – EAL 4 Assurance Requirements.

Assurance Class	Assurance Components	
	Identifier	Name

Assurance Class	Assurance Components	
	Identifier	**Name**
Development	ADV_ARC.1	Security architecture description
	ADV_FSP.4	Complete functional specification
	ADV_IMP.1	Implementation representation of the TSF
	ADV_TDS.3	Basic modular design
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-cycle support	ALC_CMC.4	Production support, acceptance procedures and automation
	ALC_CMS.4	Problem tracking CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_DVS.1	Identification of security measures
	ALC_FLR.1	Basic flaw remediation
	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.1	Well-defined development tools
Security Target Evaluation	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.2	Security objectives
	ASE_REQ.2	Derived security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE summary specification

Assurance Class	Assurance Components	
	Identifier	**Name**
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: basic design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	AVA_VAN.3	Focused vulnerability analysis

Table 18 – EAL 4 Assurance Requirements

7 TOE SUMMARY SPECIFICATION

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

7.1 TOE SECURITY FUNCTIONS

A description of each of the TOE security functions follows.

7.1.1 Security Audit

The activity log records connection events. The log file is written in a comma-delimited format (CSV). The activity log records the following information:

Event	Event Description	Audit Record
Connect	The Mobility client connects to the Mobility server	• Date and time connection was made • Device Name • User Name • Virtual IP assigned by the Mobility server • Point of Presence (PoP) address of the Mobility client device at its current location • The number of bytes sent and received during the connection period
Disconnect	The Mobility client was disconnected from the Mobility server.	• Date/Time • Device Name • User Name • Reason for disconnect
Unreachable	The Mobility server could not reach a client device that was connected to the server. An unreachable state occurs when the following occurs: • A client's network connection fails • The device leaves the area network and no other network is available • The device goes into standby or hibernation. • The server receives no confirmation from the client device that it must disconnect from the server.	• Date and time the client was unavailable • Device Name • User Name
Reachable	The Mobility client that was unreachable has reconnected.	• Date and time when the connection was re-established • Device Name • User Name

Event	Event Description	Audit Record
Roaming	A Mobility client moved to a different network or the network connection changed.	• Date and time of when the connection change was made • Device Name • User Name • Virtual IP assigned by the Mobility server • PoP address of the Mobility client device at its current location • The number of bytes sent and received during the connection period

Table 19 – Activity Log Information

The activity log page displays the connection events, and automatically refreshes the display as it records new events.

The event log collects information on events that occur on the server, including:

- All decisions on requests for information flow
- When an encrypted session is established
- Startup and shutdown of audit function (as indicated by startup/shutdown of TOE)
- Administrator logon

The event log is a binary file updated by Mobility components. By default, Mobility events are logged to *nmevent.nel* in the Windows folder. They may be viewed using the Mobility Console. The event log records the following information:

Event	Description
Type	The message types include Trace, Debug, Information, Warning and Error. An administrator may specify which event types are logged.
Date/Time	This is the date and time of the event.
Source	This is the module that produced the event.
Data	This is the event log message.

Table 20 – Event Log Information

TOE Security Functional Requirements addressed: FAU_GEN.1, FAU_SAR.1.

7.1.2 Cryptographic Support

Mobility encrypts data transmitted between the Mobility server and the Mobility client using FIPS 140-2 validated cryptographic modules. The modules used are as follows:

Operating System	Description	CMVP Certificate
Windows 8.1 and Windows Server 2012 R2	Microsoft Kernel Mode Cryptographic Primitives Library (CNG.SYS) Software Version: 6.3.9600	2356
Windows 10	Microsoft Kernel Mode Cryptographic Primitives Library (CNG.SYS) Software Version: 10.0.10240	2605
Apple iOS 10	Apple iOS CoreCrypto Module, v7.0 (Hardware Version: A7 through A9X)	2827
Apple macOS 10.12	Apple macOS CoreCrypto Module, v7.0 (Hardware Version: Mac mini, MacPro and MacBook)	2832
Android 6.0	OpenSSL FIPS Object Module SE Version: 2.0.12	2398

Table 21 – Cryptographic Module Validation Program Certificates

The Microsoft and Apple cryptographic modules pre-exist in their respective operating systems. Since no suitable cryptographic module exists in the Android operating system, the OpenSSL FIPS Object Module software is delivered to the device with the NetMotion client software.

For the Mobility Server, keys are created by the cryptographic module in the Windows Server 2012 R2 operating system, in accordance with the security policy. Likewise, for Mobility Client on Windows, the keys are created by the cryptographic module in the Windows 10 or Windows 8.1 operating system, in accordance with the security policies. Windows uses a NIST SP800-90A RBG (CTR_DRBG) for symmetric keys and a NIST SP800-56 A RBG (DRBG) for asymmetric keys in the CNG.SYS library.

For Mobility Client on iOS, the keys are created by the cryptographic module in the iOS operating system, in accordance with the security policy. The platforms to which CMVP certificate 2827 apply are listed in the Apple iOS CoreCrypto Module, v7.0 FIPS 140-2 Non-Proprietary Security Policy. iOS uses a NIST SP800-90A RBG (CTR_DRBG) for symmetric and asymmetric key generation.

For Mobility Client on macOS, the keys are created by the cryptographic module in the macOS operating system, in accordance with the security policy. The platforms to which CMVP certificate 2832 apply are listed in the Apple macOS CoreCrypto Module, v7.0 FIPS 140-2 Non-Proprietary Security Policy. MacOS uses a NIST SP800-90A RBG (CTR_DRBG) for symmetric and asymmetric key generation.

For Mobility Client on Android, symmetric keys are generated using a Counter Mode Deterministic Random Bit Generator (DRBG) (using AES) in the OpenSSL library. This DRBG has been validated in accordance with NIST Special Publication 800-90, Recommendation for Random Number generation Using

Deterministic Random Bit Generators. Asymmetric keys are generated in accordance with ANSI 9.31.

Elliptic Curve Diffie-Hellman (ECDH) using NIST approved curves is used for key agreement operations in support of the establishment of a VPN tunnel. Windows clients use CNG.SYS in support of the ECDH primitives. Windows servers use CNG.SYS when communicating with Windows clients, or Miracl (Multiprecision Integer and Rational Arithmetic Cryptographic Library) primitives when communicating with Android, iOS or macOS clients. Android clients use OpenSSL primitives, and iOS and macOS clients use Miracl primitives. Each party generates an ephemeral key pair. Diffie-Hellman Key Agreement (DH) is used by each of the mobile devices in support of PEAP MS-CHAPv2 authentication. DH is implemented in accordance with NIST SP-800-56A for Finite Field Cryptography (FFC), and provides 112 or 128 bits of security strength.

For all of the operating systems, the keys are zeroized by the cryptographic module in accordance with FIPS 140-2 zeroization standards. Calls are made by the Mobility component to the cryptographic module in support of all cryptographic operations. All of the cryptographic operations are completed using Cryptographic Algorithm Validation Program validated algorithms. The certificate numbers for these algorithms are shown in Table 11, Table 12, and Table 13.

The Mobility server determines the algorithm used to encrypt client connections. In the evaluated configuration, only FIPS 140-2 validated cryptographic modules are permitted. The client cannot negotiate a lower encryption level. If a client does not support the encryption method set on the server, it cannot connect and posts a message in the client event log.

If a client accepts the requested encryption method, but then tries to establish an unencrypted connection or a connection using a different encryption method, the Mobility server disconnects the session and posts a warning in the server event log.

7.1.2.1 Use of Cryptographic Primitives

The following table identifies the usage of the claimed cryptographic primitives:

Function	Supporting Algorithms
PEAP MS-CHAPv2 Authentication	Diffie Hellman Key Agreement and ECDH P-256
	RSA
	3DES CBC
	AES CBC 128, 256; AES GCM 128, 256
	SHA-256, SHA 384
EAP-TLS	ECDH P-256

Function	Supporting Algorithms
	3DES CBC
	AES CBC 128, 256; AES GCM 128, 256
	RSA
	SHA-256, SHA-384
	HMAC 256, HMAC 384
Encrypted Tunnel	ECDH P-521
	AES CBC 128, 256
	SHA-256, SHA-384
	HMAC 256, HMAC 384

Table 22 – Cryptographic Primitive Usage

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

7.1.3 User Data Protection

The TOE enforces the secure information flow control SFP between the Mobility Server and the Mobility Client by means of end-user authentication. For the purposes of the evaluation, the TOE uses a RADIUS (RFC 2865) based authentication server and a standards compliant implementation of the PEAP MS-CHAPv2 or EAP-TLS.

When the Mobility Server is configured to use RADIUS for authentication, it acts as a Network Access Server (NAS) in the RADIUS security system. Authentication is performed using a database that supports the PEAP MS-CHAPv2 format, such as Microsoft Active Directory (AD) (PEAPv0, draft-josefsson-pppext-eap-tls-eap-06).

Although standards compliance is claimed, modern RADIUS authenticators do not enforce use of the mandatory ciphersuite in PEAP. Cisco RADIUS servers disable RC4 by default and Microsoft allows use of RC4 to be disabled. In the evaluated configuration, use of RC4 and MD5 must be disabled. However, it should be noted that even without this configuration, Mobility will default to PEAP authentication using a ciphersuite with RSA, AES 256 and SHA 384.

7.1.3.1 PEAP MS-CHAPv2 Authentication

Protected Extensible Authentication Protocol (PEAP) is used in the evaluated configuration in support of user login. PEAP encapsulates the Extensible Authentication Protocol (EAP) within an encrypted and authenticated Transport Layer Security (TLS) tunnel. The inner authentication protocol is Microsoft's Challenge Handshake Authentication Protocol Version 2 (RFC 2759).

The encrypted tunnel is set up as follows:

Step 1 – PEAP Stage 1

The client contacts the Mobility Server. The authentication server presents its certificate. The client verifies that the certificate is valid, was signed by a Certification Authority (CA) trusted by the client, and was issued to a known server. The client and authentication server continue to establish a TLS tunnel, using a cryptosuite that is supported by both the client and the authentication server. This step uses Diffie-Hellman (DH) key agreement to set up the outer tunnel.

Step 2 – PEAP Stage 2

A PEAP MS-CHAPv2 exchange occurs within the tunnel to authenticate the user using a challenge response. (This exchange is described in RFC 2759.) The user is authenticated to the authentication server.

Step 3 – Elliptic Curve Diffie-Hellman (ECDH)

ECDH key agreement occurs between the Mobility client and the Mobility server to create the symmetric keys that will be used to encrypt the session. This exchange is authenticated using HMAC-MD5 and the key generated during the PEAP MS-CHAPv2 exchange (RFC 2548 paragraph 2.4.1).

Step 4 – Encrypted tunnel

Data passes between the client and the application server, through the Mobility server. The data is encrypted with the symmetric keys between the client and the mobility server.

7.1.3.2 EAP-TLS Authentication

The other user authentication option is Extensible Authentication Protocol Transport Layer Security (EAP-TLS) (RFC 5216). This is very similar to the PEAP implementation, except that user certificates are presented instead of a username and password.

The encrypted tunnel is set up as follows:

Step 1 – EAP Stage 1

The client contacts the authentication server. The authentication server presents its certificate. The client verifies that the certificate is valid, was signed by a Certification Authority (CA) trusted by the client, and was issued to a known server.

Step 2 – EAP Stage 2

The user presents a user certificate which may be verified by the authentication server. The user certificate may be installed on a smart card, or in the user's personal certificate store. A password may be required to access the user certificate.

Step 3 – Elliptic Curve Diffie-Hellman (ECDH)

ECDH key agreement occurs between the Mobility client and the Mobility server to create the symmetric keys that will be used to encrypt the session. This exchange is authenticated using HMAC-MD5 and the key generated during the EAP-TLS exchange (RFC 5216; see paragraph 2.3 Master Session Key (MSK)).

Step 4 – Encrypted tunnel

Data passes between the client and the application server, through the Mobility server. The data is encrypted with the symmetric keys between the client and the mobility server.

Once the Authentication server confirms that the user is identified and authenticated, the Mobility Server allows data to flow from the Mobility Client, through the Mobility Server to the protected internal resource. The Mobility client secures all of the application data destined for the TCP/IP network stack, including TCP, UDP and ICMP ping using a VPN tunnel between the Mobility client and the Mobility server. This ensures that the confidentiality and integrity of any data is protected using FIPS-140-2 validated cryptographic modules.

Local traffic such as loopback traffic and data destined for local applications is not sent through the encrypted tunnel.

TOE Security Functional Requirements addressed: FDP_IFC.1, FDP_IFF.1, FDP_ITT.1.

7.1.4 Identification and Authentication

Users of the Mobility Console must provide Windows credentials that are valid for an account on the Mobility server operating system or in the domain in which the server participates. The Mobility Console user enters a username and password, which are verified in the operating system. The user is only granted access to the Mobility Console if the username and password are correct, and the user is a member of a group which has a Mobility Console role.

Users of the Mobility Client must be identified and authenticated before being allowed access to protected resources, as described in Section 7.1.3. The authentication mechanism may be either PEAP MS-CHAPv2 or EAP-TLS. Users will authenticate using the mechanism that has been implemented in their environment.

Administrators authenticate to the Mobility Server from the management console computer. Both the authentication server and the management console computer are located on a trusted internal network.

TOE Security Functional Requirements addressed: FIA_UAU.2, FIA_UAU.5 and FIA_UID.2.

7.1.5 Security Management

The Mobility Console provides the functionality to manage NetMotion Mobility. This functionality includes:

a. Creating, deleting, modifying and viewing the information flow security policy rules that permit or deny information flows, and changing default values for, querying, modifying and deleting the data required to establish those rules;

b. Reviewing and clearing audit logs; and

c. Querying, modifying and deleting user account attributes.

Users of the Mobility Console must be a member of the Administrator group or be made an administrator on the Mobility Console. The Mobility server is installed with the built-in Administrator role, which is assigned to the Administrators group in Active Directory. To access the Mobility Console, users must provide Windows credentials that are valid for an account on the Mobility server or in the domain in which the server participates.

Mobility Client Users do not have access to Mobility Console functions.

TOE Security Functional Requirements addressed: FMT_MTD.1, FMT_SMF.1, FMT_SMR.1.

7.1.6 Protection of the TSF

All events on the Mobility server (identified in Table 18 above) are logged and stamped with the date and time, which is retrieved from the operating system. All log timestamps are written as *yyyyMMdd* and *hh:mm:ss*.

Both Ipv4 and Ipv6 addressing are supported.

TOE Security Functional Requirements addressed: FPT_STM.1.

8 TERMINOLOGY AND ACRONYMS

8.1 TERMINOLOGY

The following terminology is used in this ST:

Term	Description
Activity log	The activity log records connection events and is written in a comma-delimited format for export and analysis. To view the activity log, the administrator opens the Mobility Console and selects the 'Activity log' link at the top of the page.
Application data	Application data refers to the data that passes between the end user and the server when the end user accesses the application remotely. The application may be any client/server application written for an IP network. Application data includes, but is not limited to, documents, spreadsheets, images, and XML data. Application data is considered to be at risk when it is transmitted on an external public network.
Event log	The event log is a binary file updated by any Mobility component that logs event messages. By default, events are logged to nmevent.nel in the Windows folder. To view the event log, the administrator opens the Mobility Console and selects the 'Event log' link at the top of the page.
Management Console Computer	This is the hardware that supports use of the Mobility Console through a browser. It is not part of the TOE.
Mobility Console	The Mobility Console is a web-based configuration and management utility that an administrator can use to configure settings, create and apply client policies, monitor server status and client connections, monitor activity or event logs, and troubleshoot problems. It is part of the TOE.

Table 23 – Terminology

8.2 ACRONYMS

The following acronyms are used in this ST:

Acronym	Definition
AD	Active Directory
AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certification Authority

Acronym	Definition
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria
CMVP	Cryptographic Module Validation Program
CSE	Communications Security Establishment (Canada)
CSEC	Swedish Certification Body for IT Security
CSV	Comma separated values
CTR-DRBG	Counter mode Deterministic Random Bit Generator
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
EAL	Evaluation Assurance Level
EAP	Extensible Authentication Protocol
ECDH	Elliptic Curve Diffie-Hellman
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standards
GCM	Galois/Counter Mode
HMAC	Hash Message Authentication Code
HTTPS	Hypertext transfer protocol secure
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
MS-CHAP	Microsoft Challenge-Handshake Authentication Protocol
MSK	Master Session Key
NAS	Network Access Server
NIST	National Institute of Standards and Technology (USA)

Acronym	Definition
PEAP	Protected Extensible Authentication Protocol
PoP	Point of Presence
PP	Protection Profile
RADIUS	Remote Authentication Dial In User Service
RAM	Random Access Memory
RBG	Random Bit Generator
RSASSA	RSA Signature Scheme with Appendix
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
ST	Security Target
TCP/IP	Transmission Control Protocol / Internet Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
UDP	User Datagram Protocol
USA	United States of America
VPN	Virtual Private Network

Table 24 – Acronyms