FIPS 140-2 SECURITY POLICY FOR

SPECTRAGUARD® ENTERPRISE SERVER

August 31, 2011

FIPS 140-2 LEVEL-1 SECURITY POLICY FOR

AIRTIGHT NETWORKS' SPECTRAGUARD ENTERPRISE SERVER

1. Introduction

This document describes the Security Policy for the SpectraGuard® Enterprise Server cryptographic module from AirTight Networks, Inc. The Security Policy specifies the rules under which the module shall operate to meet Federal Information Processing Standard (FIPS) 140-2 Level 1 requirements.

FIPS 140-2, *Security Requirements for Cryptographic Modules*, describes the requirements for cryptographic modules. For more information about the FIPS 140-2 standard and the cryptographic module validation process see http://csrc.nist.gov/groups/STM/cmvp/index.html.

2. Module Specification

SpectraGuard® Enterprise Server version 6.5.35 (hereafter referred to as "Module") is a multi-chip standalone firmware cryptographic module from AirTight Networks, Inc. It has limited operational environment and it executes on a Linux operating system. Key components of the Module include Command Line Interface (CLI) which can either be accessed locally over console cable or remotely over the Secure Shell (SSH), Server application which is responsible for core intrusion detection function of the Module, web server which provides access to the Module's functionality from the Module's Graphical User Interface (GUI) and from the external third party applications (called API Clients) which can interact with the Module over HTTPS, database, high availability (HA) interface, interfaces to the external third party entities (syslog, SMTP, SNMP, OPSEC, LDAP), and cryptographic libraries. Additional applications cannot be installed on the Module during run time, since Module's user interfaces (CLI and GUI) do not allow such installation.

The Module is run on production-grade general purpose computer system, enterprise appliance, or on a virtual machine such as VMware.

The logical cryptographic boundary for the Module, and paths of data, control and status information flow are shown in the following figure.

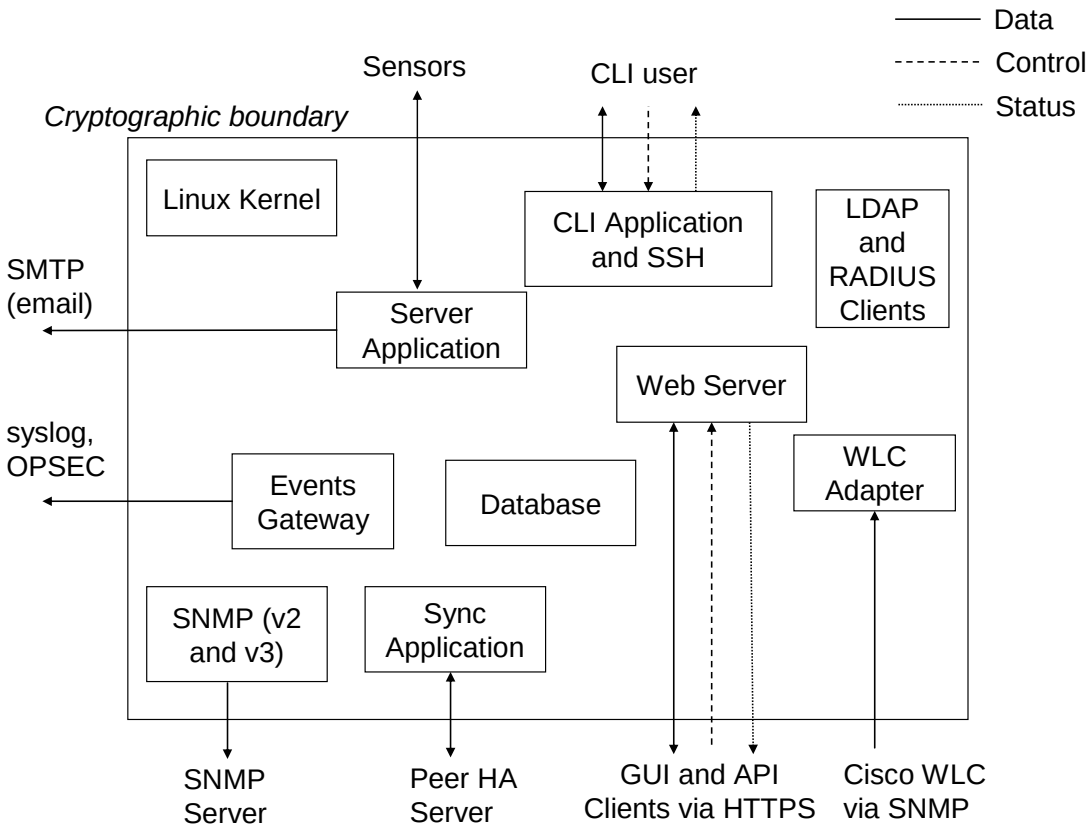

Figure 1: Cryptographic Module Schematic Diagram

In its operation, the Module generally receives, stores and processes information about wired and wireless devices reported by Sensor monitors and third party systems, in order to perform intrusion detection function. Several types of events are generated and notified by the Module based on this processing. The Module can also accept input related to any actions to be performed on the monitored devices.

Logical Interfaces

The logical interfaces are protocol level and application level interfaces in the Module which input and/or output various types of information as described below.

Data Input	Reports from Sensors, input from GUI/API Clients, information from third party systems, synchronization data from peer Server in HA configuration, and database restoration over CLI.
Data Output	Commands to Sensors, output to GUI/API Clients, information to third party systems, synchronization data to peer Server in HA configuration, and logs and database backup over CLI.
Control Input	Operational parameters input from GUI/API Clients and from CLI users.
Status Output	Operation status output to GUI/API Clients and to CLI users.

Table 1: Logical Interfaces

Ports and Interfaces

The logical interfaces on the Module map on the physical ports of the computer system on which the module executes. Mapping of the logical interfaces to the physical ports is as follows.

Logical Interface	**Physical Port**
Data input	Network interface
Data output	Network interface
Control input	Network interface, serial port
Status output	Network interface, serial port
Power input	Power connector

Table 2: Physical Ports

The video and keyboard ports on the hardware platform which runs the Module do not support any data, control or status interfaces during the operation of the Module.

Modes of Operation

The Module is able to operate in FIPS (FIPS compliant) and non-FIPS (not compliant with FIPS) modes. Factory default setting is non-FIPS mode. Crypto Officer has to enable FIPS mode, whenever FIPS compliant operation is desired. Specific steps to turn the Module into FIPS mode are as follows:

- Crypto Officer logs into the Module over CLI (Command Line Interface). CLI login can be over console cable or SSH.
- Crypto Officer changes mode of operation by executing “set FIPS mode” command to turn on the FIPS mode (after this command is executed, the Module reboots and the shared secret key (K) used for communication with Sensor(s) is reset to factory default).
- Crypto Officer enters the new shared secret key (K) using “set communication key” command.

In order to turn the Module that is operating in FIPS mode to non-FIPS mode, following steps are required:

- Crypto Officer logs into the Module over CLI.
- Crypto Officer changes mode of operation by executing “set FIPS mode” command to turn off the FIPS mode (after this command is executed, the Module reboots and the shared secret key (K) used for communication with Sensor(s) is reset to factory default).
- Crypto Officer enters the new shared secret key (K) using “set communication key” command.

Whether the Module is running in FIPS mode or not can be checked by the Crypto Officer by running “get FIPS mode” command.

Compliance with FIPS Requirements

The Module meets FIPS 140-2 security requirements as follows.

Security Requirements Section	Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	1
Operational Environment	Not applicable
Cryptographic Key Management	1
Electromagnetic Interference/Electromagnetic Compatibility	1
Self Tests	1
Design Assurance	1
Mitigation of Other Attacks	Not applicable

Table 3: FIPS Compliance

3. Security Functions

The various security functions incorporated in the Module are described below.

a) Roles, Authentication, Services

The Module supports User and Crypto Officer roles. Any user logging into the Module via web server (HTTPS) has a User role. Any user logging into the Module via CLI (console cable or SSH) has a Crypto Officer role.

User Role

The User accesses the Module over HTTPS. The User can be a GUI user or an API Client. Each User has a separate username. At the time User account is created, one of the following rights must be assigned to the User: superuser, administrator, operator, or viewer. Thus, the Module can positively recognize right of the User currently logged in. The User of the Module is authenticated using one of the following options: a) password only, b) client certificate only, c) both password and client certificate. When the password is used in User authentication, the Module performs password verification either locally or by using the external LDAP/RADIUS service.

The Module can communicate with the LDAP server either in plaintext or in the encrypted TLS 1.0 session, depending upon the configuration. Communication with the RADIUS server occurs in plaintext.

User Authentication Strength for Passwords

For authentication options which use password, the strength of the password is enforced by the “password policy” setting in the user management menu. The password policy setting enables the superuser to set the minimum threshold on the number of characters in the password. The superuser must set this threshold to at least 6. In case the password verification is performed through LDAP/RADIUS service, the 6-character threshold shall be set in the LDAP/RADIUS server. This results in at least 308,915,776 combinations for the password (computed as 26 raised to the power 6). Thus, the possibility of correctly guessing the password is less than 1 in 1,000,000. The user management menu also includes “account locking” setting, which defines the period of time for which the User account will be temporarily locked (minimum settable lockout period being 5 minutes), if the authentication failure rate crosses a threshold (the fastest being 10 failures in 5 minutes). Thus login attempt rate of the User is limited to 2 attempts per minute. This in combination with the total possible combinations of the password ensures that multiple attempts to use the authentication mechanism during a one-minute period have probability of success less than one in 100,000.

When the Module is power cycled, the User will have to re-authenticate. That is, the authentication state is forgotten after power cycle. When the Module is reset to factory default, all User accounts other than “admin” are deleted from the Module, and the password of the “admin” User is set to factory default value “admin”. The User passwords are not reset upon entering or exiting FIPS mode.

User Authentication Strength for Certificates

For the certificate-only authentication option (which does not use password) the strength of authentication is governed by the strength of certificate. Minimum key size in the certificate that is accepted by the Module in FIPS mode is 1024 bits.

As per SP 800-57, security strength of 1024 bit asymmetric RSA/DSA key is equivalent to 80 bit symmetric key. Since there are 2^{80} combinations for the 80 bit key, the possibility of correctly guessing the key is less than 1 in 1,000,000. Also, in case the client certificate is invalid, the Module returns error no quicker than 1 ms after the client certificate is entered. Thus, at most 60,000 certificates can be tried in 1 minute period. This in combination with the total possible combinations of the 80 bit key ensures that multiple attempts to use the authentication mechanism during a one-minute period have probability of success less than one in 100,000.

Crypto Officer Role

The Crypto Officer accesses the Module over CLI, either locally over console cable or remotely over SSH. Crypto Officer is authenticated via password. The Module performs password verification either locally or by using the external RADIUS service.

Crypto Officer Authentication Strength

The minimum threshold on the number of characters in the password as defined in the “password policy” setting in the user management menu described above also applies to the Crypto Officer. In case the password verification is performed through RADIUS service, this threshold shall be set in the RADIUS server. With 6-character minimum threshold, there are at least 308,915,776 combinations for the password (computed as 26 raised to the power 6). Hence, the possibility of correctly guessing the password is less than 1 in 1,000,000. Further, there is 2 second delay enforced after a failed login attempt before the next login attempt can be made. This in combination with the total possible combinations for the password ensures that multiple attempts to use the authentication mechanism during a one-minute period have probability of success less than one in 100,000.

When the Module is power cycled, the Crypto Officer will have to re-authenticate. That is, the authentication state is forgotten after power cycle. When the Module is reset to factory default, the password of the “config” Crypto Officer is set to the factory default value “config”. The Crypto Officer password is not reset to factory

default (zeroized) upon entering and exiting the FIPS mode of operation for the Module.

The following table summarizes the strength of authentication for User and Crypto Officer roles.

Role	Auth. Credentials	Minimum Length	Maximum Length	Strength per Attempt	Strength per Minute
User	Password	6 characters	15 characters	Probability of success less than 1 in 308,915,776	Probability of success less than 1 in 154,457,888
User	Certificate	1024 bits key	16384 bits key	Probability of success is 1 in 2^{80}	Probability of success less than 1 in 2^{64}
Crypto Officer	Password	6 characters	Unrestricted	Probability of success less than 1 in 308,915,776	Probability of success less than 1 in 10,297,192

User and Crypto Officer Services

The services available to the User and the Crypto Officer are shown in the table below.

Service	User	Crypto Officer	Description

CLI login	No	Yes	Log into the Module to access CLI (Command Line Interface), using the serial port (console cable) or over the network interface (SSH).
Bootstrap the Module	No	Yes	Configure network settings, time/date/country settings, web server certificate etc.
Start/stop services	Yes (partial)	Yes	Start/stop Server application, web server, SSH server and enable/disable HA.
Update firmware	Yes	Yes	Load new firmware. (Note: loading a unvalidated version of the firmware (that is, a version that has not been FIPS 140-2 validated) invalidates the module.)
Change mode of operation	No	Yes	Change mode of operation of the Module between FIPS and non-FIPS.
Change shared secret key (K)	No	Yes	Change shared secret key (K) used between the Module and the Sensors.
Show status	Yes (partial)	Yes	View status of operation of Server application, web server, HA interface, and SSH server. View mode of operation.
Set User authentication	Yes	No	Select the authentication option

option			for Users. One of the following four options can be set, which applies to all Users: password only, client certificate only, both password and client certificate, either password or client certificate. This service is only available to User with superuser role.
View self test result	No	Yes	Check result of self tests.
Perform on-demand self test (Reboot)	No	Yes	Perform self tests/reboot the Module.
Reset factory defaults (Zeroize)	No	Yes	Restore the Module to factory default state.
Database management	No	Yes	Perform database backup, restore and reset.
Reboot Sensor	Yes	No	Reboot the Sensor which is connected to the Module.
Web-based login	Yes	No	Log into the Module over web based interface using HTTPS.
Set operational parameters	Yes	No	Set channels, alarms, third party server identities, Sensor timeouts etc.

Table 4: Roles and Services

b) Controlling Access to the Module for the First time

By default, there is always one Crypto Officer account with the username “config” with the factory default password “config”. The username “config” cannot be

changed or deleted. The Crypto Officer should enable FIPS mode and manually input the shared secret key (K). The Crypto Officer should change the password for “config” account. The new password should be at least 6 characters in length.

By default, there is always one User with the username “admin” and the factory default password “admin”. This User has superuser privilege. The username “admin” cannot be changed or deleted. However, password of “admin” can be changed and should be changed upon first login. The new password should be at least 6 characters in length. The “admin” User should also set the “password policy” in the user management menu for the minimum password character threshold of at least 6. If the external authentication service (LDAP/RADIUS) is used for password verification, the minimum password character threshold of at least 6 should be set in the LDAP/RADIUS server. The “admin” User should also set the authentication option to be used by all Users of the Module. Additional Users (with usernames other than “admin”) can then be added. Valid license is required to be applied before User login prompt can be displayed for the first time.

c) Encryption/Decryption

Various encryption/decryption functions are described below:

c.1) Communication with Sensors

Certain data output/input to/from each Sensor is encrypted. Before the Module establishes data communication channel with the Sensor, mutual authentication is performed.

Mutual Authentication

The mutual authentication is performed using the shared secret key (K). During the mutual authentication, each side sends a challenge to the other side, which the other side returns as AES-CBC encrypted with the key K. The challenging side decrypts the response and verifies that the original challenge is found therein.

Derivation and Transport of Session Key (SK)

After mutual authentication, the session key (SK) is randomly generated by the Module and transported to the Sensor encrypted with the key K using AES-CBC encryption. The session key is 128 bits in length.

Derivation of Message Encryption and Authentication Keys (MAK, MEK)

The message encryption key (MEK) and message authentication key (MAK) are derived via HMAC-SHA1 of the predefined text and the randomly generated key ID, using the session key SK as the secret key. The MEK is 128 bits in length and the MAK is 160 bits in length. The MAK is used for per-message HMAC-SHA1 authentication and the MEK is used for per-message AES-CBC encryption between the Module and the Sensor. There is different pair of (MAK, MEK) in each direction – Module to Sensor and Sensor to Module. These key derivation procedures are in compliance with NIST Special Publication 800-108, *Recommendation for Key Derivation Using Pseudorandom Functions,* October 2009.

Shared Secret Key (K) Management

After the Crypto Officer enables FIPS mode, the Crypto Officer is required to change the key K from its factory default setting. The Crypto Officer has to manually input the new value of key K is the Module. The key K resets to factory default (zeroized) upon entering or exiting the FIPS mode. The Crypto Officer is required to zeroize the key K either by resetting the module to factory default or by exiting the FIPS mode at such times as the Module is to be discarded. The key K is never outputted in plaintext from the Module.

c.2) SSH

The Module includes SSH version 2. The SSH server supports secure remote access to CLI for the Crypto Officer. It also supports securely receiving (using SCP (secure copy)) the HA synchronization data from the peer server. In FIPS mode, the SSH server is restricted to use the following approved algorithms: RSA (2048), DSA (1024), AES-CBC (128, 192, 256), SHA1 and HMAC-SHA1. It also uses Diffie-Hellman (1024 or greater) algorithm.

The SSH client supports securely outputting database backup data and securely receiving database restore data. It also supports securely outputting the HA synchronization data. In FIPS mode, the SSH client is restricted to use the following approved algorithms: RSA (1024 or greater), DSA (1024), AES-CBC (128, 192, 256), SHA1 and HMAC-SHA1. It also uses Diffie-Hellman (1024 or greater) algorithm. In FIPS mode, the database backup data output, the database restore data input and the HA synchronization data output are all performed using SCP (secure copy) and they all contain the shared secret key (K).

The database backup and restore are formatted as .zip file. The .zip encryption is considered equivalent to plaintext.

c.3) Output over SNMPv3, SMTP, Syslog, OPSEC Interfaces and Log Output

None of these contains any CSPs or keys for the Module. They are considered (equivalent to) plaintext.

c.4) Web Server

The web server is responsible for interaction between the Module and GUI or API Clients. HTTPS protocol is used for communication between the web server and the GUI/API Clients. HTTPS runs over TLS version 1.0. In FIPS mode, the web server is restricted to use the following algorithms: RSA (1024 or greater), DSA (1024), AES-CBC (128, 192, 256), Triple-DES (168), SHA-1 and HMAC-SHA1. It also uses Diffie-Hellman (1024 or greater) algorithm. The web server also uses MD5 algorithm to the extent its use is allowed in TLS 1.0.

c.5) LDAP Client

The Module optionally supports communication with the external LDAP server over HTTPS which runs over TLS version 1.0. In FIPS mode, the LDAP client in the Module is restricted to use the following algorithms: RSA (1024 or greater), DSA (1024), AES-CBC (128, 192, 256), Triple-DES (168), SHA-1 and HMAC-SHA1. It also uses Diffie-Hellman (1024 or greater) algorithm. The LDAP client also uses MD5 algorithm to the extent its use is allowed in TLS 1.0.

d) Summary of Passwords, Keys and Critical Security Parameters (CSPs)

Passwords, keys and CSPs used in the Module are summarized below, along with access control to them. The “read” access (“R”) means ability to read value, the “write” (“W”) access means ability to change value (including zeroization/reset to factory default, creation and deletion), and the “execute” (“E”) access means ability to use value for performing some function in obtaining a specific service.

The keys and CSPs used in the Module are not shared between non-FIPS and FIPS modes. Persistent keys and CSPs are stored in the non-volatile memory in plaintext and ephemeral keys and CSPs are stored in the volatile memory in plaintext. All the keys and CSPs are zeroized upon entering/exiting FIPS mode or upon reset to factory defaults. The keys and CSPs are either never output from the Module, or if output, they are never output in plaintext.

Password/Key /CSP	**Description**	**Access to CSP** (U: User, CO: Crypto Officer; Access right denoted in parentheses)
User password	Used to authenticate the User. The User password may be shared between non-FIPS and FIPS modes. The password is stored in hashed form in non-volatile memory.	“CLI login” service: CO (W). Crypto Officer can reset the password of User with username “admin” to factory default. “Web-based login” service: U (E, W). Superuser can change password of any user. Non-superuser can

		change only his own password.
Crypto Officer password	Used to authenticate the Crypto Officer. The Crypto Officer password may be shared between non-FIPS and FIPS modes. The password is stored in hashed form in non-volatile memory.	“CLI login” service: CO (E, W). Crypto Officer can change his own password.
Shared secret key (K)	Used for mutual authentication with Sensors. This key is 128 bits in length. This key is same for all the Sensors. The key K is manually entered in the Module by the Crypto Officer. This key is persistent across power cycle and rebooting of the Module.	“Change shared secret key (K)”, “Change mode”, and “Reset factory defaults” services: CO (W).
Session key (SK) for Sensor communication	The session key SK is generated by the Module and securely transmitted to the Sensor. There is separate SK for each Sensor. SK is used to derive inbound/outbound encryption and authentication key pairs (MEK, MAK) for communication with the Sensor. The key SK is 128 bits in length. This key is ephemeral and it is deleted upon termination of the session between the Module and the Sensor.	“Reboot”, “Change mode”, “Database management”, “Start/stop services”, and “Reset factory defaults” services: CO (W). “Reboot Sensor”, “Set operation parameters”, “Start/stop services” services: U (W).
Outbound and inbound	The key MAK is used for authentication of messages between	“Reboot”, “Change mode”, “Database management”,

Message Authentication Keys (MAKs) for Sensor communication	the Module and the Sensor. There is a different key MAK in outbound and inbound direction. Each key MAK is 160 bits in length. The MAKs are ephemeral and they are deleted upon termination of the session between the Module and the Sensor.	"Start/stop services", and "Reset factory defaults" services: CO (W). "Reboot Sensor", "Set operation parameters", "Start/stop services" services: U (W).
Outbound and inbound Message Encryption Keys (MEKs) for Sensor communication	The key MEK is used for encryption of messages between the Module and the Sensor. There is a different key MEK in outbound and inbound direction. Each key MEK is 128 bits in length. The MEKs are ephemeral and they are deleted upon termination of the session between the Module and the Sensor.	"Reboot", "Change mode", "Database management", "Start/stop services", and "Reset factory defaults" services: CO (W). "Reboot Sensor", "Set operation parameters", "Start/stop services" services: U (W).
RSA and DSA private keys in SSH server	RSA (2048 bits) and DSA (1024 bits) private keys are used for host authentication in SSH. They are generated when the Module is first booted, and thereafter on entering/exiting FIPS mode or on reset to factory default in FIPS mode. They are persistent across power cycle and rebooting of the Module.	"CLI login" service over SSH: CO (E). "Change mode" and "Reset factory defaults" services: CO (W).
Diffie-Hellman private key in SSH server	Diffie-Hellman private key (1024 bits or greater) is generated and used for key agreement at the time of	"CLI login" service over SSH: CO (E). "CLI login" over SSH,

	establishment of each SSH session. The Diffie-Hellman private key is ephemeral and it is deleted upon termination of the SSH session.	“Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
Outbound and inbound message authentication keys in SSH server	There is a separate message authentication key (of size 160 bits) in each direction (outbound and inbound) for each SSH session. These keys are ephemeral and are deleted upon termination of the SSH session.	“CLI login” service over SSH: CO (E). “CLI login” over SSH, “Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
Outbound and inbound message encryption keys in SSH server	There is a separate message encryption key (of size 128 bits, 192 bits or 256 bits) in each direction (outbound and inbound) for each SSH session. These keys are ephemeral and are deleted upon termination of the SSH session.	“CLI login” service over SSH: CO (E). “CLI login” over SSH, “Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
RSA private key in SSH client	RSA (2048 bits) private key is used for user authentication by the SSH client in the Module to the SSH server on the peer server in HA configuration. It is generated when HA configuration is enabled and deleted when HA configuration is disabled. It is persistent across power cycle and rebooting of the Module.	“Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
Diffie-Hellman private key in	Diffie-Hellman private key (1024 bits or greater) is generated and used for	“Database management” service: CO (E).

SSH client	key agreement at the time of establishment of each SSH session. The Diffie-Hellman private key is ephemeral and it is deleted upon termination of the SSH session.	“Database management”, “Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
Outbound and inbound message authentication keys in SSH client	There is a separate message authentication key (of size 160 bits) in each direction (outbound and inbound) for each SSH session. These keys are ephemeral and are deleted upon termination of the SSH session.	“Database management” service: CO (E). “Database management”, “Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
Outbound and inbound message encryption keys in SSH client	There is a separate message encryption key (of size 128 bits, 192 bits or 256 bits) in each direction (outbound and inbound) for each SSH session. These keys are ephemeral and are deleted upon termination of the SSH session.	“Database management” service: CO (E). “Database management”, “Reboot”, “Change mode”, “Start/stop services” and “Reset factory defaults” services: CO (W).
RSA private key in web server	This RSA private key (2048 bits) is used for authenticating the web server to the GUI/API Clients. It is generated when the Module is first booted, and thereafter on entering/exiting FIPS mode, on new certificate request by the Crypto Officer, or on reset to factory default in FIPS mode. It is persistent across power cycle and rebooting of the Module.	“Web-based login” service: U (E). “Change mode”, “CLI login”, and “Reset factory defaults” services: CO (W). CO (W) in “CLI login” service occurs when the CO requests generation of new certificates via CLI command.

Diffie-Hellman private key in web server	Diffie-Hellman private key (1024 bits or greater) is generated and used for key agreement at the time of establishment of a TLS session with the web server. The Diffie-Hellman private key is ephemeral and is deleted upon termination of the TLS session.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Temporary RSA private key in web server	This temporary RSA private key (1024 bits or greater) is generated at the time of establishment of a TLS session with the web server. It is used to decrypt the premaster secret sent by the client to the web server that has been encrypted with the public key corresponding to the temporary RSA private key. This key is ephemeral and is deleted upon termination of the TLS session.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Premaster secret in web server	Premaster secret (of size 48 bytes) is agreed between the client and the web server at the time of establishment of the TLS session. It is either randomly generated by the client and sent to the web server encrypted with the temporary RSA public key of the web server, or it is agreed upon via Diffie-Hellman exchange. The premaster secret is used to establish master secrets for	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).

	TLS connections within the TLS session. The premaster secret is ephemeral and is deleted upon termination of the TLS session.	
Master secret in web server	Master secret (of size 48 bytes) is generated for each TLS connection within the TLS session, using the premaster secret for the TLS session. The master secret is used to derive message authentication and encryption keys for the TLS connection. The master secret is ephemeral and is deleted upon termination of the TLS connection.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Outbound and inbound message authentication keys in web server	There is a separate message authentication key (of size 160 bits) in each direction (outbound and inbound) for each TLS connection (the TLS connection is inside the TLS session). These keys are ephemeral and are deleted upon termination of the TLS connection.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Outbound and inbound message encryption keys in web server	There is a separate message encryption key (for AES-128, AES-256 or Triple-DES-168 encryption) in each direction (outbound and inbound) for each TLS connection (the TLS connection is inside the TLS session). These keys are ephemeral and are deleted upon	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).

	termination of the TLS connection.	
Diffie-Hellman private key in LDAP client	Diffie-Hellman private key (1024 bits or greater) is generated and used for key agreement at the time of establishment of an LDAP session over TLS. The Diffie-Hellman private key is ephemeral and is deleted upon termination of the LDAP TLS session.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Premaster secret in LDAP client	Premaster secret (of size 48 bytes) is agreed between the LDAP client and the LDAP server at the time of establishment of the TLS session with the LDAP server. It is either randomly generated by the client and sent to the LDAP server encrypted with the temporary RSA public key of the LDAP server, or it is agreed upon via Diffie-Hellman exchange. The premaster secret is used to establish master secrets for TLS connections within the TLS session. The premaster secret is ephemeral and is deleted upon termination of the TLS session.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Master secret in LDAP client	Master secret (of size 48 bytes) is generated for each TLS connection within the TLS session with the LDAP server, using the premaster secret for the TLS session. The	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults"

	master secret is used to derive message authentication and encryption keys for the TLS connection. The master secret is ephemeral and is deleted upon termination of the TLS connection.	services: CO (W).
Outbound and inbound message authentication keys in LDAP client	There is a separate message authentication key (of size 160 bits) in each direction (outbound and inbound) for each TLS connection with the LDAP server (the TLS connection is inside the TLS session). These keys are ephemeral and are deleted upon termination of the TLS connection with the LDAP server.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Outbound and inbound message encryption keys in LDAP client	There is a separate message encryption key (for AES-128, AES-256 or Triple-DES-168 encryption) in each direction (outbound and inbound) for each TLS connection with the LDAP server (the TLS connection is inside the TLS session). These keys are ephemeral and are deleted upon termination of the TLS connection with the LDAP server.	"Web-based login" service: U (W) and U (E). "Reboot", "Change mode", "Start/stop services" and "Reset factory defaults" services: CO (W).
Seed and Seed Key	Seed (128 bits) and seed key (256 bits) are used for random number generation. There are separate	"Reboot", "Change mode", "Start/stop services" and "Reset factory defaults"

	instances of seed and seed key for different cryptographic services/applications within the Module. Random number generator obtains its seed and seed key by reading bytes from the /dev/urandom device. They are deleted from memory on power cycle, reboot, entering/exiting FIPS mode or reset to factory default.	services: CO (W). "Start/stop services" service: U (W).

Table 5: Passwords, Keys, CSPs

Public Keys:

The following public keys belong to the Module and are used in FIPS mode:

- RSA (2048 bits) and DSA (1024 bits) public keys in SSH server: They are the public key counterparts of the corresponding private keys used for host authentication in SSH server. They are not shared between non-FIPS and FIPS modes. These keys are stored in plaintext in non-volatile memory. These keys are generated when the Module is first booted, and thereafter on entering or exiting FIPS mode, or on reset to factory default.

- Diffie-Hellman public key (1024 bits or greater) in SSH server: It is the public key component of the private key used for key agreement at the time of establishment of SSH session. It is not shared between non-FIPS and FIPS modes. This key is stored in plaintext in volatile memory. It is deleted upon termination of the SSH session, power cycle, reboot, entering/exiting FIPS mode, or reset to factory default.

- Diffie-Hellman public key (1024 bits or greater) in SSH client: It is the public key component of the private key used for key agreement at the time of establishment of SSH session. It is not shared between non-FIPS

and FIPS modes. This key is stored in plaintext in volatile memory. It is deleted upon termination of the SSH session, power cycle, reboot, entering/exiting FIPS mode, or reset to factory default.

- RSA public key (2048 bits) in web server: It is the public key counterpart of the web server private key. This key is not shared between non-FIPS and FIPS modes. It is stored in plaintext in non-volatile memory. This key is generated when the Module is first booted, and thereafter on entering or exiting FIPS mode, upon new certificate request by the Crypto Officer, or on reset to factory default.

- Diffie-Hellman (1024 bits or greater) and temporary RSA (1024 bits or greater) public keys in web server: They are the public key counterparts of the corresponding private keys used for premaster secret derivation or transport in the web server. They are not shared between non-FIPS and FIPS modes. These keys are stored in plaintext in volatile memory. They are deleted upon termination of the SSH session, power cycle, reboot, entering/exiting FIPS mode, or reset to factory default.

The following public keys do not belong to the Module, but are stored in the module and are used in the FIPS mode:

- Public keys of the root CA, the intermediate CA(s) and the client certificate, which are used during the client certificate based User authentication,
- Public keys of the root CA, the intermediate CA(s) and the LDAP server certificate, which are used during accessing the LDAP server over TLS,
- Public key of the peer SSH server, which is used by the SSH client in the Module during database backup and restore,
- Public key of the peer SSH server/client, which is used by the SSH client/server in the Module during the HA synchronization data output/input process,

- Ephemeral public keys of the peer entities in the TLS and SSH sessions.

Additional Keys:

The following additional keys are used in FIPS mode. They are not considered CSPs.

- A hardcoded key with which a Sensor password is encrypted when sent from the Module to the Sensor even within the AES tunnel. In other words, the password is double encrypted, once with inner encryption (with a factory-defined fixed key) and then with outer encryption (with key MEK). The inner encrypted password is considered equivalent to plaintext. The factory-defined inner encryption key cannot be changed or zeroized.

e) Summary of Cryptographic Algorithms

The Module implements the following FIPS approved or allowed algorithms.

Algorithm	**Validation Certificate**	**Usage**	**Keys/CSPs**	**Key Sizes (bits)**
AES	#1545	Encrypt/decrypt	K, MEK, SSH per-session encryption and decryption keys, TLS per-session encryption and decryption keys	128, 192, 256
Triple-DES	#1015	Encrypt/decrypt	TLS per-session encryption and decryption keys	168
SHA1	#1370	Hashing	None	Not Applicable

HMAC-SHA1	#896	Message integrity, key derivation	Shared secret key (K), session key (SK), message authentication key (MAK), SSH per-session message authentication key, TLS premaster secret, TLS master secret, TLS per-session message authentication key	16, 20
RSA	#748	Digital signature, encryption	RSA private key in SSH and TLS, temporary RSA private key	Key generation: 2048; Signature generation: 2048; Signature verification: 1024 – 4096; Encryption: 1024 – 16384.
DSA	#477	Digital signature	DSA private key in SSH	1024
PRNG	#833	Random number generation	Seed and seed key	256

Diffie-Hellman	Non-approved, but allowed for key establishment	Key establishment	Diffie-Hellman keys in SSH and TLS	1024 – 6144
MD5	Not approved, but allowed in TLS 1.0 during key exchange	Key derivation	Premaster and master secrets	Not Applicable

Table 6: Cryptographic Algorithms in FIPS Mode

Additionally, only in non-FIPS mode, the Module implements following cryptographic algorithms:

Digital Signature	RSA (key sizes smaller than 1024 bits), DSA (key sizes smaller than 1024 bits)
Message Integrity	HMAC-SHA1-96, HMAC-MD5-96, HMAC-MD5
Encryption	AES-CTR, ARC4, Blowfish-CBC, CAST128, ARC4-256, ARC4-128, CAST128-CBC, RC2, RC4, DES, IDEA, RSA (key sizes smaller than 1024 bits)
Hashing	MD5, UMAC-64 (RFC4418), RIPEMD-160

Table 7: Additional Cryptographic Algorithms in non-FIPS Mode

4. Self Tests

The Module always reboots when FIPS mode is entered. At boot time firmware integrity check is done using MD5 checksum of the firmware. If it passes, the

module performs power-up self tests for SSH server, web server and Server application as given in the following table. These self tests are also performed whenever an instance of SSH client is initiated.

Algorithm	**Test**
AES	KAT
Triple-DES	KAT
DSA	KAT, sign/verify
RSA	KAT
PRNG	KAT
HMAC-SHA-1	KAT
SHA-1	KAT

Table 8: Power-up Self Tests

If any of the above tests fails, the Module goes to Error State.

During operation, the module performs following conditional self tests:

Algorithm	**Test**
DSA	Pairwise consistency
RSA	Pairwise consistency
PRNG	Continuous

Table 9: Conditional Self Tests

The Module also performs firmware load test (using RSA 2048 bits digital signature) at the time of updating the Module with the new firmware. If the firmware load test fails, the new firmware image is not loaded onto the Module and the Module continues operation with the existing firmware image. If any other of the above tests fails, the Module enters the Error State.

In Error State, the Module does not output any data on the Data Output interface. The results of the above tests can be viewed by the Crypto Officer by accessing the Module over CLI.

When Crypto Officer attempts changing the shared secret key (K), manual key entry test is performed. If the test succeeds, new key is accepted. Else, new key is rejected.

It is also possible to perform on-demand self test by rebooting the Module.

5. Physical Security

The Module has limited operational environment and it executes on a Linux operating system. Access to the operating system operations is restricted by the Module.

Firmware integrity check is performed using MD5 checksum at the time Module boots up. When new firmware is loaded in the Module, any new firmware image's integrity is verified using RSA digital signature.

The Module is run on production-grade general purpose computer system, enterprise appliance, or on a virtual machine such as VMware. In such environment, the Module is entirely contained within a metal or hard plastic production-grade enclosure that blocks physical access to the Module.