Brocade Communications Systems, Inc. Directors and Switches 8.1.0 using Fabric OS v8.1 (NDcPP10) Security Target

Version 0.3
06/01/17

Prepared for:

Brocade Communication Systems, Inc.

130 Holger Way
San Jose, CA 95134

Prepared By:

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is Directors and Switches provided by Brocade Communication Systems, Inc. The TOE is being evaluated as a network infrastructure device.

The Security Target contains the following additional sections:

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

Conventions

The following conventions have been applied in this document:

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

Acronyms and Terminology

This following acronyms and terms are used throughout this document.

DEK	Data Encryption Key
FC	Fibre Channel
FCIP	Fibre Channel over IP
HBA	Host Bus Adapter
JBOD	Stands for "Just a Bunch of Disks", and it a way of connecting together a series of hard drives, combining multiple drives and capacities, into one drive
LUN	Logical Unit Number, used to refer to a logical device within a chain.
SAN	Storage Area Network

1.1 Security Target Reference

ST Title – Brocade Communication Systems, Inc. Directors and Switches 8.1.0 using Fabric OS v8.1 Security Target

ST Version – Version 0.3

ST Date – 06/01/17

1.2 TOE Reference

TOE Identification – Brocade Communications Systems, Inc. Gen 5 and Gen 6 Directors and Switches operating with Fabric OS version 8.1.0, including the following series and models:

- Gen 5 Directors and Switches
  - Director Blade[1] Models: FC16-32, FC16-48, FC16-64, CP8, CR16-4, CR16-8, FX8-24,
  - Director Models: DCX 8510-4 and DCX 8510-8
  - Switch Appliance Models: 6510, 6520 and 7840
- Gen 6 Directors and Switches
  - Director Blade Models: FC32-48, CPX6, CR32-4, CR32-8 and SX6
  - Director Models: X6-4 and X6-8
  - Switch Appliance Model: G620

TOE Developer – Brocade Communication Systems, Inc.

Evaluation Sponsor – Brocade Communication Systems, Inc.

1.3 TOE Overview

The Target of Evaluation (TOE) is the Brocade Director and Switch family of products using Fabric OS v8.1.0 provided by Brocade Communications Systems, Inc. **Error! Reference source not found.** are hardware network devices that create what is called a 'Storage Area Network' or 'SAN'. SANs provide switched connections between servers connected to the SAN and storage devices such as disk storage systems and tape libraries that are also connected to the SAN. The TOE is being evaluated as a network devices as defined in the NDcPP.

1.4 TOE Description

The Target of Evaluation (TOE) is the **Error! Reference source not found.** running Fabric OS v8.1.0. The various models of the TOE identified below differ in performance, form factor and number of ports, but all run the same Fabric OS version 8.1.0 software. The TOE is available in two form factors:

1. a rack-mount Director chassis with a variable number of replaceable modules or ‘blades’, and
2. a self-contained network switching appliance device

Gen 5 Director models utilize blades of several types. A ‘director blade model’ can be a control blade (CP8), a core blade (CR16-4, CR16-8), and port blades (FC16-32, FC16-48, FC16-64) or application blades (FX8-24). Control blades provide the control plane for the chassis. A core switch blade contains the ASICs for switching between port blades. A port blade supports various numbers of ports and speeds. Application blades provide additional capabilities such as Fibre Channel (FC) over Ethernet. The DCX 8510-4 and DCX 8510-8 require at least one control blade and one core blade to make the director operational.

[1] A blade refers to a purpose-built component that is installed in a Brocade director.

Gen 6 Director models utilize blades of several types. A 'director blade model' is a control blade (CPX6), a core switch blade (CR32-4 or CR32-8), and port blades (FC32-48) or application blades (SX6 FC-IP). Control blades contain the control plane for the chassis. A core switch blade contains the ASICs for switching between port blades. A port blade supports various numbers of ports and speeds. Application blades provide additional capabilities such as Fibre Channel (FC) over Ethernet. The X64 and X68 require at least one control blade and one core blade to make the director operational.

Director Model	Blades
DCX 8510-4	CP8, CR16-4, FC16-32, FC16-48, FC16-64, FX8-24
DCX 8510-8	CP8, CR16-8, FC16-32, FC16-48, FC16-64, FX8-24
X6-4	CPX6, CR32-4, , FC32-48, SX6
X6-8	CPX6, CR32-8, FC32-48, SX6

Brocade Directors and Switches are hardware appliances that create a "SAN". SANs enable connectivity between machines in the environment containing a type of network card called a Fibre Channel Host Bus Adapter (HBA) that are located in the environment and storage devices such as disk storage systems and tape libraries that are also located in the environment. The network connection between the storage devices in the environment, the TOE, and HBAs in the environment use high-speed network hardware. SANs are optimized to transfer large blocks of data between HBAs and storage devices. SANs can be used to replace or supplement server-attached storage solutions, for example.

The basic concept of operations from a *user's* perspective is depicted below. Actual implementation may interconnect multiple instances of TOE models.

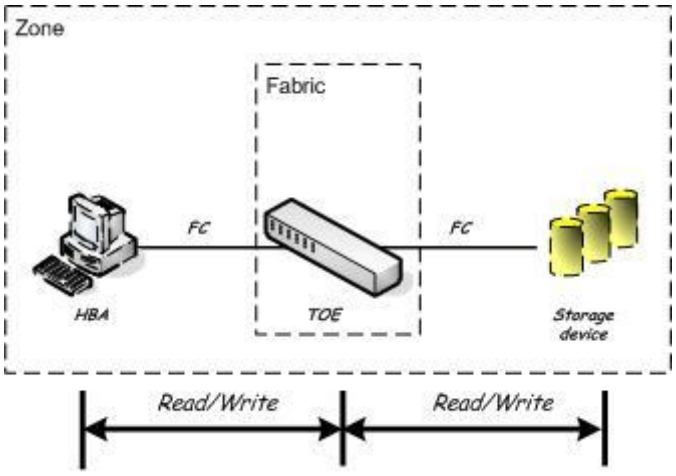

Figure 1: Host bus adapters can only access storage devices that are members of the same zone.

HBAs communicate with the TOE using FC or FC over IP (FCIP) protocols. Storage devices in turn are physically connected to the TOE using cabling connected to FC/FCIP interfaces.

1.4.1 TOE Architecture

The TOE provides the ability to centralize the location of storage devices in a network in the environment. Instead of attaching disks or tapes to individual hosts in the environment, or for example attaching a disk or tape directly to the network, storage devices can be physically attached to the TOE, which can then be physically attached to HBAs in the environment. HBAs that are connected to the TOE can then read from and write to storage devices that are

attached to the TOE according to TOE configuration. Storage devices in the environment appear to the operating system running on the machine that the HBA is installed in as local (i.e. directly-attached) devices.

More than one HBA can share one or more storage devices that are attached to the TOE according to TOE configuration. Scalability is achieved by interconnecting multiple instances of TOE directors and switches to form a fabric that supports different numbers of HBAs and storage devices.

Directors and switches both can be used by HBAs to access storage devices using the TOE. Switch appliances provide a fixed number of physical interfaces to hosts and storage devices in the environment. Directors provide a configurable number of physical interfaces using a chassis architecture that supports the use of blades that can be installed in and removed from the director chassis according to administrator configuration.

There are administrative interfaces to manage TOE services that can be accessed using an Ethernet network, as well as interfaces that can be accessed using a directly-attached console as follows:

- Ethernet network-based web-based administrator console interfaces – Provides web-based administrator console interfaces called the "Brocade Advanced Web Tools."
- Ethernet network-based command-line administrator console interfaces – Provides command-line administrator console interfaces called the "FabricOS Command Line Interface."
- Serial terminal-based command-line administrator console interfaces – Provides command-line administrator console interfaces called the "FabricOS Command Line Interface."

There also exists administrative Ethernet network-based programmatic API interfaces that can be protected using SSL. The API interface is not supported in the evaluated configuration. Similarly, there exists a modem hardware component that is optional to the product that can be used in a similar manner as a serial console port, but it is disabled by virtue of not being physically installed during initial installation and configuration in the evaluated configuration.

The TOE can operate in either "Native Mode" or "Access Gateway Mode". Only Native mode is supported in the evaluated configuration. Access Gateway mode makes the switch function more like a "port aggregator" and in Access Gateway mode the product does not support the primary access control security functions (mainly zoning) claimed when operating in Native mode.

The basic concept of operations from an *administrator's* perspective is depicted below. While actual implementations may interconnect multiple instances of TOE models, each TOE device (i.e., instance of the TOE) is administered individually.

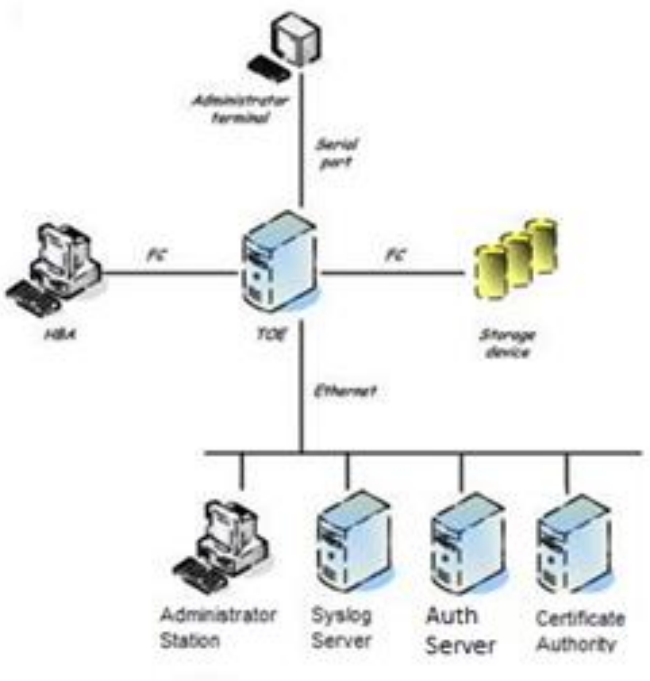

Figure 2: Administrators can access the TOE using a serial terminal or across a network. Audit records are sent to a syslog server.

Separate appliance ports are relied on to physically separate connected HBAs. The appliance's physical location between HBAs and storage devices is relied on to ensure TOE interfaces cannot be bypassed. The TOE encrypts commands sent from terminal applications by administrators using SSH for the command line interface and HTTPS for the Advanced Web Tools GUI interface. The TOE requires administrators to login before an SSH or HTTPS session is established.

1.4.1.1 Physical Boundaries

The TOE can be described in terms of the following components:

- Brocade Switch and Director Appliances – One or more of each type are supported in the evaluated configuration. The evaluated configuration also supports one or more blades per director, depending on the number supported by a given director model.
- Brocade FabricOS operating system – Linux-based operating system that runs on Brocade switches and directors. FabricOS is comprised of user-space programs, kernel daemons and kernel modules loaded as proprietary components into LINUX. The base features of LINUX, including the file system, memory management, processor and I/O support infrastructure for FOS user-space programs, daemons, and kernel modules. Interprocess communication is handled through commonly mapped memory or shared PCI memory and semaphores as well as IOCTL parameter passing. LINUX provides access to memory or to make a standard IOCTL call, and all the contents of the buffers and IOCTL message blocks or other message blocks are proprietary to the FOS user-space programs, kernel modules and daemons. The FabricOS operating system is considered to include the OpenSSL crypto engine version 1.0.2h as internal functionality supporting TOE operation.

In its most basic form, the TOE in its intended environment of the TOE is depicted in the figure below.

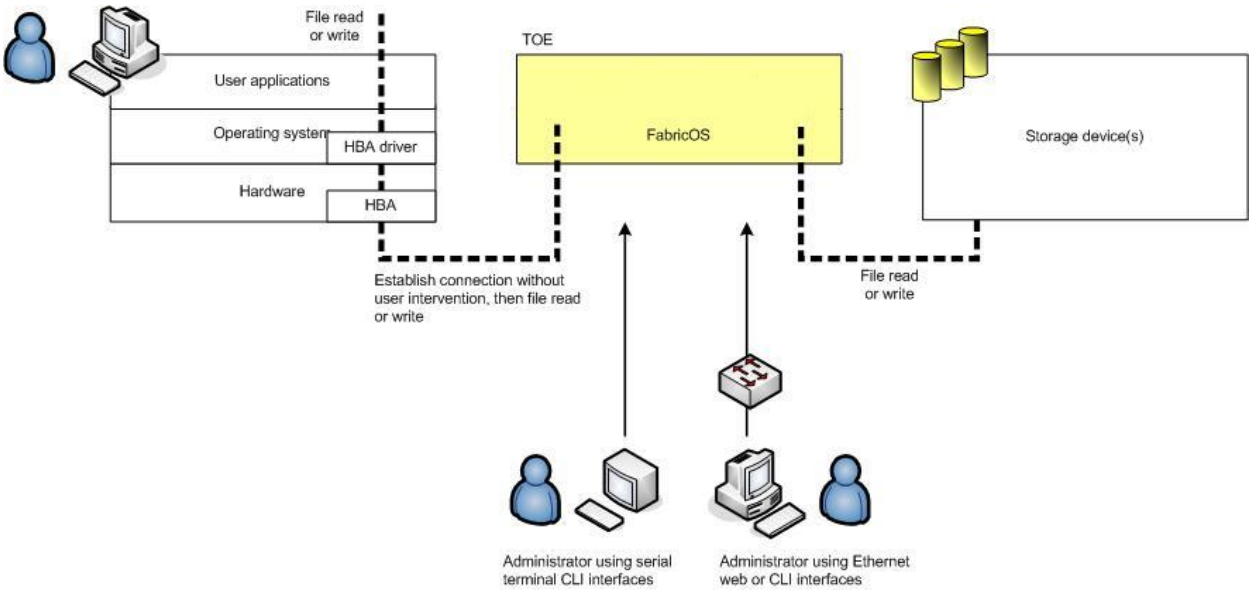

Figure 3: TOE and environment components.

The intended environment of the TOE can be described in terms of the following components:

- Host – A system in the environment that uses TOE Storage Area Network (SAN) services.
- Host Bus Adapters (HBAs) – Provides physical network interfaces from host machines in the environment to the TOE. HBA drivers provide operating system interfaces on host machines in the environment to storage devices in the environment. Storage devices in the environment appear to the host operating system as local (i.e. directly-attached) devices.
- Storage device – A device used to store data (e.g. a disk or tape) that is connected to the TOE using a FC/FCIP connection and is accessed by a host using the TOE.
- Terminal application – Provides a runtime environment for console-based (i.e. SSH) client administrator console interfaces.
- Web browser – Provides a runtime environment for web-based (i.e. HTTPS) client administrator console interfaces.
- Syslog server – Provides logging to record auditable event information generated by the TOE. The syslog server is expected to protect audit information sent to it by the TOE and make that data available to administrators of the TOE.
- Certificate Authority (CA) – Provides digital certificates TLS-based interfaces that are installed during initial TOE configuration. After installation, the CA no longer needs to be on the network for operation.
- Key management systems – Provide life cycle management for all data encryption keys (DEKs) created by the encryption engine. Key management systems are provided by third-party vendors and are not included in the scope of this evaluation.

The TOE relies on a syslog server in the environment to store and protect audit records that are generated by the TOE. The TOE does not rely on any other components in the environment to provide security-related services.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by Brocade Directors and Switches:

- Security audit
- Cryptographic support

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

1.4.1.2.1 Security audit

The TOE generates audit events for numerous activities including policy enforcement, system management and authentication. A syslog server in the environment is relied on to store audit records generated by the TOE. The TOE generates a complete audit record including the IP address of the TOE, the event details, and the time the event occurred. The time stamp is provided by the TOE appliance hardware. When the syslog server writes the audit record to the audit trail, it applies its own time stamp, placing the entire TOE-generated syslog protocol message contents into an encapsulating syslog record.

1.4.1.2.2 Cryptographic support

The TOE contains CAVP tested cryptographic implementations that provide key management, random bit generation, encryption/decryption, digital signature and secure hashing and key-hashing features in support of higher level cryptographic protocols including SSH and TLS.

1.4.1.2.3 Identification and authentication

The TOE authenticates administrative users. In order for an administrative user to access the TOE, a user account including a user name and password must be created for the user, and an administrative role must be assigned. Either the TOE performs the validation of the login credentials. The administrator can configure the order in which the external authentication provider and the local credentials are checked.

1.4.1.2.4 Security management

The TOE provides serial terminal (command line) and Ethernet network-based (command-line and web) management interfaces. Each of the three types of interfaces provides equivalent management functionality. The TOE provides administrative interfaces to set and reset administrator passwords.

1.4.1.2.5 Protection of the TSF

The TOE implements a number of features designed to protect itself to ensure the reliability and integrity of its security features.

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. It also provides its own timing mechanism to ensure that reliable time information is available (e.g., for log accountability).

Note that the TOE is a single appliance, and as such, no intra-TOE communication is subject to any risks that may require special protection (e.g., cryptographic mechanisms).

The TOE includes functions to perform self-tests so that it might detect when it is failing. It also includes mechanisms (i.e., verification of the digital signature of each new image) so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE.

1.4.1.2.6 TOE access

The TOE can be configured to display a message of the day banner when an administrator establishes an interactive session and subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session (local or remote) will be terminated.

1.4.1.2.7 Trusted path/channels

The TOE enforces a trusted path between the TOE administrators and the TOE using SSH and TLS/HTTPS connections for Ethernet connections from the Administrator terminal to the TOE. The TOE encrypts commands sent from terminal applications by administrators using SSH for the command line interface and TLS/HTTPS for the Advanced Web Tools GUI interface. The TOE provides a TLS protected communication channel between itself and remote audit and authentication servers.

1.4.2 TOE Documentation

Brocade offers a series of documents that describe the installation process for the TOE as well as guidance for subsequent use and administration of the applicable security features.

- Configuration Guide Fabric OS Common Criteria Supporting Fabric OS 8.1.0b, March 31, 2017

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
    - Part 3 Conformant
- ST conforms to the *collaborative Protection Profile for Network Devices*, Version 1.0, 27 February 2015 (NDcPP).

2.1 Conformance Rationale

The ST conforms to the NDcPP10. The security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDcPP10 and this section reproduces only the corresponding Security Objectives for reader convenience. The NDcPP10 offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP10 should be consulted if there is interest in that material.

In general, the NDcPP10 has defined Security Objectives appropriate for a network infrastructure device and as such are applicable to the Directors and Switches TOE.

3.1 Security Objectives for the TOE

O.DISPLAY_BANNER
The TOE will display an advisory warning regarding use of the TOE.

O.PROTECTED_COMMUNICATIONS
The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.

O.RESIDUAL_INFORMATION_CLEARING
The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.

O.SESSION_LOCK
The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.

O.SYSTEM_MONITORING
The TOE will provide the capability to generate audit data and send those data to an external IT entity.

O.TOE_ADMINISTRATION
The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.

O.TSF_SELF_TEST
The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

O.VERIFIABLE_UPDATES
The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.

3.2 Security Objectives for the Operational Environment

OE.ADMIN_CREDENTIALS_SECURE The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

OE.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

OE.UPDATES The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP10. The NDcPP10 defines the following extended requirements and since they are not redefined in this ST the NDcPP10 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- FAU_STG_EXT.1: Protected Audit Event Storage
- FCS_HTTPS_EXT.1: HTTPS Protocol
- FCS_RBG_EXT.1: Random Bit Generation
- FCS_SSHC_EXT.1: SSH Client Protocol
- FCS_SSHS_EXT.1: SSH Server Protocol
- FCS_TLSC_EXT.2: TLS Client Protocol with authentication
- FCS_TLSS_EXT.2: TLS Server Protocol with mutual authentication
- FIA_PMG_EXT.1: Password Management
- FIA_UAU_EXT.2: Password-based Authentication Mechanism
- FIA_UIA_EXT.1: User Identification and Authentication
- FIA_X509_EXT.1: X.509 Certificate Validation
- FIA_X509_EXT.2: X.509 Certificate Authentication
- FIA_X509_EXT.3: X.509 Certificate Requests
- FPT_APW_EXT.1: Protection of Administrator Passwords
- FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)
- FPT_TST_EXT.1: TSF testing
- FPT_TUD_EXT.1: Trusted update
- FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

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

The SFRs have all been drawn from the NDcPP10. The refinements and operations already performed in the NDcPP10 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP10 and any residual operations have been completed herein. Of particular note, the NDcPP10 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDcPP10 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the NDcPP10 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The NDcPP10 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by the Brocade Directors and Switches TOE.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User identity association
	FAU_STG.1: Protected audit trail storage
	FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation
	FCS_CKM.2: Cryptographic Key Establishment
	FCS_CKM.4: Cryptographic Key Destruction
	FCS_COP.1(1): Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1(2): Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1(3): Cryptographic Operation (Hash Algorithm)
	FCS_COP.1(4): Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1: HTTPS Protocol
	FCS_RBG_EXT.1: Random Bit Generation
	FCS_SSHS_EXT.1: SSH Server Protocol
	FCS_TLSC_EXT.2: TLS Client Protocol with authentication
	FCS_TLSS_EXT.1: TLS Server Protocol
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management
	FIA_UAU.7: Protected Authentication Feedback
	FIA_UAU_EXT.2: Password-based Authentication Mechanism
	FIA_UIA_EXT.1: User Identification and Authentication
	FIA_X509_EXT.1: X.509 Certificate Validation
	FIA_X509_EXT.2: X.509 Certificate Authentication
	FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	FMT_MOF.1(1): Management of security functions behaviour - TrustedUpdate
	FMT_MOF.1(3): Management of security functions behaviour - Audit
	FMT_MOF.1(4): Management of security functions behaviour - Audit
	FMT_MTD.1(1): Management of TSF Data
	FMT_MTD.1(2): Management of TSF data - Admin Action
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of	FPT_APW_EXT.1: Protection of Administrator Passwords

Requirement Class	Requirement Component
the TSF	FPT_FLS.1: Failure with preservation of secure state
	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM.1: Reliable Time Stamps
	FPT_TST_EXT.1: TSF testing
	FPT_TST_EXT.2: Self tests based on certificates
	FPT_TUD_EXT.1: Trusted update
	FPT_TUD_EXT.2: Trusted Update based on certificates
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination
	FTA_SSL.4: User-initiated Termination
	FTA_SSL_EXT.1: TSF-initiated Session Locking
	FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel
	FTP_TRP.1: Trusted Path

Table 1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (FAU_GEN.1)

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:
a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions comprising:
- Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).
- Security related configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- Starting and stopping services (if applicable)
- [***no other actions***];

d) Specifically defined auditable events listed in **Table 2 Auditable Events.**

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, 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, information specified in column three of **Table 3 Auditable Events.**

Requirement	Auditable Events	Additional Content
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG.1	None	None
FAU_STG_EXT.1	None	None
FAU_STG_EXT.2	None	None
FAU_STG_EXT.3	Warning about low storage space for audit events.	None
FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None

Requirement	Auditable Events	Additional Content
FCS_COP.1(1)	None	None
FCS_COP.1(2)	None	None
FCS_COP.1(3)	None	None
FCS_COP.1(4)	None	None
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
FCS_RBG_EXT.1	None	None
FCS_SSHS_EXT.1	Failure to establish an SSH session. Successful SSH rekey.	Reason for failure. Non-TOE endpoint of connection (IP Address).
FCS_TLSC_EXT.1	Failure to establish a TLS Session.	Reason for failure.
FCS_TLSS_EXT.1	Failure to establish a TLS Session.	Reason for failure.
FIA_PMG_EXT.1	None	None
FIA_UAU.7	None	None
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_X509_EXT.1	Unsuccessful attempt to validate a certificate.	Reason for failure.
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None	None
FMT_MOF.1(1)	Any attempt to initiate a manual update.	None
FMT_MOF.1(3)	Modification of the behaviour of the transmission of audit data to an external IT entity.	None
FMT_MOF.1(4)	Modification of the behaviour of the handling of audit data.	None
FMT_MTD.1(1)	All management activities of TSF data.	None
FMT_MTD.1(2)	Modification, deletion, generation/import of cryptographic keys.	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None
FPT_APW_EXT.1	None	None
FPT_SKP_EXT.1	None	None
FPT_STM.1	Changes to time.	The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None	None
FPT_TST_EXT.2	Failure of self-test.	Reason for failure (including identifier of invalid certificate).
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
FTA_SSL.4	The termination of an interactive session.	None
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	None
FTA_TAB.1	None	None
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted path. Termination	Identification of the claimed user

Requirement	Auditable Events	Additional Content
	of the trusted path. Failure of the trusted path functions.	identity.

Table 4 Auditable Events

5.1.1.2 User identity association (FAU_GEN.2)

FAU_GEN.2.1

For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.1.1.3 Protected audit trail storage (FAU_STG.1)

FAU_STG.1.1

The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.1.2

The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

5.1.1.4 Protected Audit Event Storage (FAU_STG_EXT.1)

FAU_STG_EXT.1.1

The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

FAU_STG_EXT.1.2

The TSF shall be able to store generated audit data on the TOE itself.

FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [overwrite oldest records first]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (FCS_CKM.1)

FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm:

[***- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.3,***

- ECC schemes using 'NIST curves' [P-256, P-384] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4,

- FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.1].

5.1.2.2 Cryptographic Key Establishment (FCS_CKM.2)

FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method:

[***- RSA-based key establishment schemes that meets the following: NIST Special Publication 800-56B, 'Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography',***

- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography',

- Finite field-based key establishment schemes that meets the following: NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography'].

5.1.2.3 Cryptographic Key Destruction (FCS_CKM.4)

FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

[- ***For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes,]]];***

[- For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [logically addresses the storage location of the key and performs a [single-pass] overwrite consisting of [zeroes]]

that meets the following: No Standard. (TD0130 applied)

5.1.2.4 Cryptographic Operation (AES Data Encryption/Decryption) (FCS_COP.1(1))

FCS_COP.1(1).1

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC***] mode and cryptographic key sizes [***128 bits and 256 bits***] that meet the following:

- AES as specified in ISO 18033-3,
- [***CBC as specified in ISO 10116***].

5.1.2.5 Cryptographic Operation (Signature Generation and Verification) (FCS_COP.1(2))

FCS_COP.1(2).1

The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm

[***- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [assignment: 2048 bits or greater],***

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [assignment: 256 bits or greater]]

that meet the following:

[***-For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5, ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3 (TD0116 applied),***

- For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' P-256, and P-384, and [no other curves] , ISO/IEC 14888-3, Section 6.4].

5.1.2.6 Cryptographic Operation (Hash Algorithm) (FCS_COP.1(3))

FCS_COP.1(3).1

The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-512***] that meet the following: ISO/IEC 10118-3:2004.

5.1.2.7 Cryptographic Operation (Keyed Hash Algorithm) (FCS_COP.1(4))

FCS_COP.1(4).1

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512***] and

cryptographic key sizes [**equal to the input block size**] and message digest sizes [***160, 256, 512***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.8 HTTPS Protocol (FCS_HTTPS_EXT.1)

FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3
The TSF shall establish the connection only if [***the peer initiates handshake***]. (TD0125 applied)

.

5.1.2.9 Random Bit Generation (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].

FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[2 software-based noise source] software-based noise source, [1] hardware-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.10 SSH Server Protocol (FCS_SSHS_EXT.1)

FCS_SSHS_EXT.1.1
The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [***no other RFCs***].

FCS_SSHS_EXT.1.2
The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based.

FCS_SSHS_EXT.1.3
The TSF shall ensure that, as described in RFC 4253, packets greater than [**256k**] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4
The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [***aes128-cbc, aes256-cbc***]. (TD0189 applied)

FCS_SSHS_EXT.1.5
The TSF shall ensure that the SSH transport implementation uses [***ssh-rsa, ecdsa-sha2-nistp256***] and [***no other public key algorithms***] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6
The TSF shall ensure that the SSH transport implementation uses [***hmac-sha1, hmac-sha2-256, hmac-sha2-512***] and [***no other MAC algorithms***] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7
The TSF shall ensure that [***diffie-hellman-group14-sha1, ecdh-sha2-nistp256***] and [***ecdh-sha2-nistp384***] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8
The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than onegigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.(TD0167 applied)

5.1.2.11 TLS Client Protocol with authentication (FCS_TLSC_EXT.2)

FCS_TLSC_EXT.2.1

The TSF shall implement [***TLS 1.2 (RFC 5246)***] supporting the following ciphersuites:
Mandatory Ciphersuites:
- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268

Optional Ciphersuites:
[***- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,***
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246].

FCS_TLSC_EXT.2.2

The TSF shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.2.3

The TSF shall only establish a trusted channel if the peer certificate is valid.

FCS_TLSC_EXT.2.4

The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [***none***] and no other curves.

FCS_TLSC_EXT.2.5

The TSF shall support mutual authentication using X.509v3 certificates.

5.1.2.12 TLS Server Protocol (FCS_TLSS_EXT.1)

FCS_TLSS_EXT.1.1

The TSF shall implement [***TLS 1.2 (RFC 5246***] supporting the following ciphersuites:
Mandatory Ciphersuites:
- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268

Optional Ciphersuites:
[***- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,***
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246].

FCS_TLSS_EXT.1.2

The TSF shall deny connections from clients requesting SSL 1.0, SSL 2.0, SSL 3.0, TLS 1.0, and [***TLS 1.1***].

FCS_TLSS_EXT.1.3

The TSF shall generate key establishment parameters using RSA with key size 2048 bits and [***no other size***] and [***no other***]. (TD0191 applied)

5.1.3 Identification and authentication (FIA)

5.1.3.1 Password Management (FIA_PMG_EXT.1)

FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [***!, @, #, $, %, ^, &, *, (,)***];

b) Minimum password length shall be settable by the Security Administrator, and shall support passwords of 15 characters or greater.

5.1.3.2 Protected Authentication Feedback (FIA_UAU.7)

FIA_UAU.7.1

The TSF shall provide only obscured feedback to the administrative user while the authentication is in progress at the local console.

5.1.3.3 Password-based Authentication Mechanism (FIA_UAU_EXT.2)

FIA_UAU_EXT.2.1

The TSF shall provide a local password-based authentication mechanism, [***[SSH public-key-based authentication mechanism]***] to perform administrative user authentication.

5.1.3.4 User Identification and Authentication (FIA_UIA_EXT.1)

FIA_UIA_EXT.1.1

The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:
- Display the warning banner in accordance with FTA_TAB.1;
- [***[network routing and SAN services]***].

FIA_UIA_EXT.1.2

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

5.1.3.5 X.509 Certificate Validation (FIA_X509_EXT.1)

FIA_X509_EXT.1.1

The TSF shall validate certificates in accordance with the following rules: - RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [***the Online Certificate Status Protocol (OCSP) as specified in RFC 2560,***].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - o Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.8.0.1.3) in the extendedKeyUsage field.
  - o Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.8.0.1.1) in the extendedKeyUsage field. o Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.8.0.1.2) in the extendedKeyUsage field.
  - o OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.8.0.1.9) in the extendedKeyUsage field.

FIA_X509_EXT.1.2

The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.1.3.6 X.509 Certificate Authentication (FIA_X509_EXT.2)

FIA_X509_EXT.2.1

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

FIA_X509_EXT.2.2

When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [***not accept the certificate***].

5.1.3.7 X.509 Certificate Requests (FIA_X509_EXT.3)

FIA_X509_EXT.3.1

The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name***].

FIA_X509_EXT.3.2

The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behaviour - TrustedUpdate (FMT_MOF.1(1))

FMT_MOF.1(1).1

The TSF shall restrict the ability to enable the functions to perform manual update to Security Administrators.

5.1.4.2 Management of security functions behaviour - Audit (FMT_MOF.1(3))

FMT_MOF.1(3).1

The TSF shall restrict the ability to determine the behaviour of, modify the behaviour of the functions transmission of audit data to an external IT entity to Security Administrators.

5.1.4.3 Management of security functions behaviour - Audit (FMT_MOF.1(4))

FMT_MOF.1(4).1

The TSF shall restrict the ability to determine the behaviour of, modify the behaviour of the functions handling of audit data to Security Administrators.

5.1.4.4 Management of TSF Data (FMT_MTD.1(1))

FMT_MTD.1(1).1

The TSF shall restrict the ability to manage the TSF data to Security Administrators.

5.1.4.5 Management of TSF data - Admin Action (FMT_MTD.1(2))

FMT_MTD.1(2).1

The TSF shall restrict the ability to modify, delete, generate/import the cryptographic keys to Security Administrators.

5.1.4.6 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions:
- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates (Per TD0090);
- [***-No other capabilities***].].

5.1.4.7 Restrictions on Security Roles (FMT_SMR.2)

FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions
- The Security Administrator role shall be able to administer the TOE locally;
- The Security Administrator role shall be able to administer the TOE remotely
are satisfied.

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Protection of Administrator Passwords (FPT_APW_EXT.1)

FPT_APW_EXT.1.1

The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext passwords

.

5.1.5.2 Protection of TSF Data (for reading of all symmetric keys) (FPT_SKP_EXT.1)

FPT_SKP_EXT.1.1

The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.5.3 Reliable Time Stamps (FPT_STM.1)

FPT_STM.1.1

The TSF shall be able to provide reliable time stamps.

5.1.5.4 TSF testing (FPT_TST_EXT.1)

FPT_TST_EXT.1.1

The TSF shall run a suite of the following self-tests [***periodically during normal operation***] to demonstrate the correct operation of the TSF: [**cryptographic algorithm known answer tests, firmware integrity tests using RSA signature verification and conditional self-tests for PRNG, pair-wise consistency tests on generation of RSA keys, and a firmware load test (RSA signature verification)**].

5.1.5.5 Trusted update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1

The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software as well as the most recently installed version of the TOE firmware/software.

FPT_TUD_EXT.1.2

The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [***no other update mechanism***].

FPT_TUD_EXT.1.3

The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.6 TOE access (FTA)

5.1.6.1 TSF-initiated Termination (FTA_SSL.3)

FTA_SSL.3.1

Refinement: The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.6.2 User-initiated Termination (FTA_SSL.4)

FTA_SSL.4.1

Refinement: The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.6.3 TSF-initiated Session Locking (FTA_SSL_EXT.1)

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [***terminate the session***] after a Security Administrator-specified time period of inactivity.

5.1.6.4 Default TOE Access Banners (FTA_TAB.1)

FTA_TAB.1.1

Refinement: Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Inter-TSF trusted channel (FTP_ITC.1)

FTP_ITC.1.1

The TSF shall be capable of using [***TLS***] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [***authentication server***] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2

The TSF shall permit the TSF, or the authorized IT entities to initiate communication via the trusted channel.

FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [**transfer of audit records, verification of user identity via remote authentication server**].

5.1.7.2 Trusted Path (FTP_TRP.1)

FTP_TRP.1.1

The TSF shall be capable of using [***SSH, HTTPS***] to provide a communication path between itself and authorized remote administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and provides detection of modification of the channel data.

FTP_TRP.1.2

The TSF shall permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3

The TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

5.2 TOE Security Assurance Requirements

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

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

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

Table 3 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic functional specification (ADV_FSP.1)

ADV_FSP.1.1d
The developer shall provide a functional specification.

ADV_FSP.1.2d
The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c
The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c
The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c
The functional specification shall provide rationale for the implicit categorisation of interfaces as SFR-non-interfering.

ADV_FSP.1.4c
The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e
The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1d
The developer shall provide operational user guidance.

AGD_OPE.1.1c
The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c
The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c
The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c
The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c
The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6c
The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c
The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative procedures (AGD_PRE.1)

AGD_PRE.1.1d
The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1c
The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c
The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e
The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d
The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c
The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM coverage (ALC_CMS.1)

ALC_CMS.1.1d
The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c
The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c
The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent testing - conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

ATE_IND.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e
The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

AVA_VAN.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2e
The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e
The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

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

6.1 Security audit

The TOE generates audit records for start-up and shutdown of the TOE, all administrator actions, and for an unspecified level of audit (see table below for specific events). Audit records include date and time of the event, type of event, user identity that caused the event to be generated, and the outcome of the event. The TOE maintains a local audit log buffer that retains the last 1024 messages persistently, overwriting the oldest events as necessary, and is only accessible by TOE administrators after logging in. The TOE sends audit records to a configured syslog server in the environment. The environment is relied on to provide interfaces to read from the audit trail. The auditable events include those listed in **Table 5 Auditable Events**.

Syslog protocol messages containing audit records have three parts. The first part is called the PRI, the second part is the HEADER, and the third part is the MSG. The TOE generates syslog audit records as follows:

- The TOE generates a complete audit record including the IP address of the TOE, the event details, and the time the event occurred. The time stamp is provided by the underlying TOE appliance hardware.

  Each audit record contains the following fields:

  > *AUDIT, <Timestamp generated by TOE>, <Event Identifier>, <Severity>, <Event Class>, <Username>/<Role>/<IP address>/<Interface>/<Application name>, <Admin Domain>/<Switch name>, <Reserved field for future expansion>, <Message>*

  For example:

  > *AUDIT, 2006/12/10-09:54:03 (GMT), [SEC-1000], WARNING, SECURITY, JohnSmith/root/192.168.132.10/Telnet/CLI, Domain A/JohnsSwitch, , Incorrect password during login attempt*

- The audit record is packaged into a syslog protocol message. The complete audit record is packaged into the syslog MSG part. The PRI and HEADER are then added.
- A network connection is established with the syslog server in the environment and the audit record is sent.

When the syslog server writes the audit record to the audit trail, it applies its own time stamp, placing the entire TOE-generated syslog protocol message contents into an encapsulating syslog record, as depicted below.

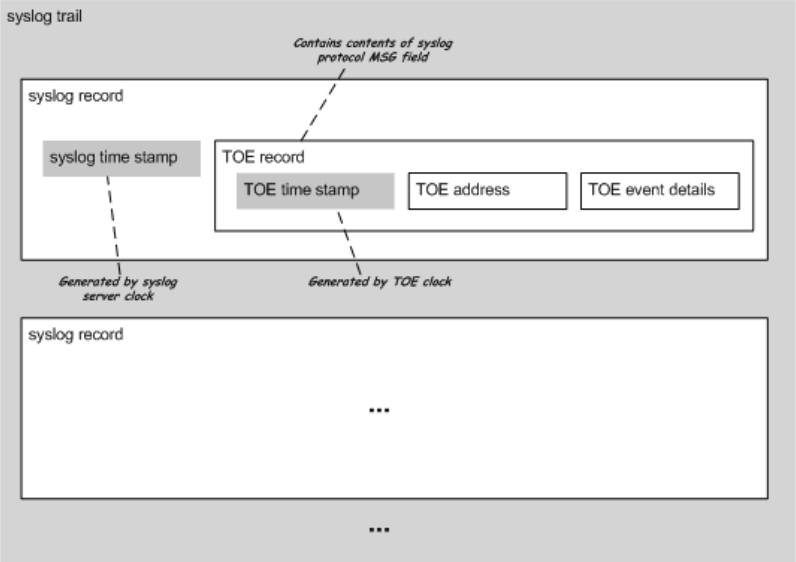

Figure 4: TOE and environment audit record components.

Since the time stamp applied by the TOE was included as part of the event details, the time stamp in the event details can be used to determine the order in which events occurred on the TOE. Similarly, the instance of the TOE that generated the record can be determined by examining the field containing the IP address of the TOE.

For example:

> *Jun 20 11:07:11 [10.33.8.20.2.2] raslogd: AUDIT, 2006/12/10-09:54:03 (GMT), [SEC-1000], WARNING, SECURITY, JohnSmith/root/192.168.132.10/Telnet/CLI, Domain A/JohnsSwitch, , Incorrect password during login attempt.*

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

- FAU_GEN.1: The TOE generates audit events for the not specified level of audit. A syslog server in the environment is relied on to store audit records generated by the TOE.
- FAU_GEN.2: The TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- FAU_STG.1: The TOE protects the audit trail from modification and deletion by not allowing direct access to the audit log files. Only adminsitarors are permitted to log onto the TOE and the TOE does not offer any administrative interfaces to the audit log other than to view and clear.
- FAU_STG_EXT.1: The TOE can be configured to export audit records to an external SYSLOG server. This communication is protected with the use of TLS.

6.2 Cryptographic support

The TOE includes a CAVP tested cryptomodule providing supporting cryptographic functions. The evaluated configuration requires that the TOE be configured in FIPS mode to ensure CAVP tested functions are used. The following table provides a mapping between the models and the processors. This mapping can be used to reference the CAVP certificates referenced in the next table.

Model	Processor
DCX 8510-4	MPC8548
DCX 8510-8	MPC8548
Brocade 6510	PPC440EPX

Model	Processor
Brocade 6520	MPC8548
Brocade 7840	P3041
X6-4	P4080
X6-8	P4080
Brocade G620	T1022

Table 6 Processor Mapping

The following functions have been CAVP tested in accordance with the identified standards.

Functions	Requirement	Cert				
		MPC8548	PPC440EPX	T1022	P4080	P3041
Encryption/Decryption						
AES CBC (128 and 256 bits)	FCS_COP.1(1)	4247	4246	4244	4243	4242
Cryptographic signature services						
RSA Digital Signature Algorithm (rDSA) (modulus 2048)	FCS_COP.1(2)	2292	2291	2290	2289	2288
ECDSA Digital Signature Algorithm (P-256, P-384)	FCS_COP.1(2)	987	986	984	983	982
Cryptographic hashing						
SHA-1/256/512 (digest sizes 160, 256, and 512 bits)	FCS_COP.1(3)	3484	3483	3481	3480	3479
Keyed-hash message authentication						
HMAC-SHA-1, HMAC_SHA2-256, HMAC-SHA2-512 (digest sizes 160, 256, and 512 bits)	FCS_COP.1(4)	2785	2784	2782	2781	2780
Random bit generation						
RNG with sw based noise sources	FCS_RBG_EXT.1	1326	1325	1323	1322	1321
Key Generation						
RSA Key Generation	FCS_CKM.1	2292	2291	2290	2289	2288
ECDSA Key Generation	FCS_CKM.1	987	986	984	983	982
DSA Key Generation	FCS_CKM.1	1137	1136	1134	1133	1132
Key Establishment						
ECC FFC CVL	FCS_CKM.2	999	996	993	990	987
Key Derivation Functions						

Functions	Requirement	Cert				
		MPC8548	PPC440EPX	T1022	P4080	P3041
TLS and SSH		1000	997	994	991	988

Table 7 Cryptographic Functions

The TOE generally fulfills all of the NIST SP 800-56A and SP 800-56B requirements without extensions. The TOE does not perform any operations marked as "shall not" or "should not" and performs all operations marked as "shall" or "should". For elliptic curve and finite-field based key establishment, the TOE implements the following sections of SP 800-56A: 5.6 and all subsections. For RSA key establishment, the TOE implements the following sections of SP 800-56B: 6 and all subsections. The TOE acts as both a sender and receipt for RSA-based key establishment schemes as it acts as a client for SYSLOG and a server for it TLS management interface.

The TOE uses an SP 800-90A AES-256 CTR_DRBG when operating in the FIPS mode. AES-256 is used in conjunction with a minimum of 256 bits of entropy accumulated from the timing of disk I/O completion events and the low-order bits from the CPU clock.

Additionally, the TOE is designed to zeroize secret and private keys when they are no longer required by the TOE. This function has also been subject to FIPS 140 certification. Note that zeroization occurs as follows: 1) when deleted from FLASH, the previous value is overwritten once with zeroes; 2) when added or changed in FLASH, any old value is overwritten completely with the new value; and, 3) the zeroization of values in RAM is achieved by overwriting once with zeroes.

The following Critical Security Parameters are contained in the module:

- DH Private Keys for use with 2048 bit modulus in SSHv2 (FLASH)
- SSH Session Keys- 128 and 256 bit AES CBC (RAM)
- SSH Authentication Keys - 2048 bit RSA private/public key pair (FLASH)
- SSH KDF Internal State (RAM)
- SSH DH Shared Secret Key – 2048 bit key size (RAM)
- TLS Private Key (RSA 1024) (FLASH)
- TLS Pre-Master Secret – 48 byte key size (RAM)
- TLS Master Secret – 48 byte key size (RAM)
- TLS PRF Internal State (RAM)
- TLS Session Key – 128 bit AES (RAM)
- TLS Authentication Key for HMAC-SHA-1 (RAM)
- Approved RNG Seed Material (RAM)
- ANSI X9.31 DRNG Internal State (RAM)
- Passwords (FLASH)

These supporting cryptographic functions are included to support the SSHv2 (compliant with RFCs 4251, 4252, 4253, and 4254) and TLSv1.2 (compliant with RFC 5246) secure communication protocols.

The TOE supports TLSv1.2 with client (mutual) authentication. The TOE rejects older versions of TLS and SSL. The TOE does not support certificate pinning. TLS v1.2 is supported with AES (CBC) 128 or 256 bit ciphers, in conjunction with SHA-1 and SHA-256 and RSA. The following cipher suites are implemented by the TOE:

- TLS_RSA_WITH_AES_128_CBC_SHA,
- TLS_RSA_WITH_AES_256_CBC_SHA,
- TLS_RSA_WITH_AES_128_CBC_SHA256, and
- TLS_RSA_WITH_AES_256_CBC_SHA256.

The TOE supports SSHv2 with AES (CBC) 128 or 256 bit ciphers, in conjunction with HMAC-SHA-1, HMAC-SHA2-256, and HMAC-SHA2-512 and RSA and ECDH using the following key exchange methods:

- diffie-hellman-group14-sha1,
- ecdh-sha2-nistp256,
- ecdh-sha2-nistp256,
- ecdh-sha2-nistp384

The TOE also supports SSH_RSA and ecdsa-sha2-nistp256 for user and server authentication. While other ciphers and hashes are implemented in the product, they are disabled while the TOE is operating in FIPS mode.

The SSHv2 supports both public-key and password based authentication can be configured; and packets are limited to 256K bytes. Whenever the timeout period, one gigabyte limit or authentication retry limit is reached, the TOE closes the applicable TCP connection and releases the SSH session resources. As SSH packets are being received, the TOE uses a buffer to build all packet information. Once complete, the packet is checked to ensure it can be appropriately decrypted. However, if it is not complete when the buffer becomes full (256K bytes) the packet will be dropped.

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

- FCS_CKM.1: See table above.
- FCS_CKM.2: See table above
- FCS_CKM.4: All memory is cleared by overwriting it with zeroes.
- FCS_COP.1(1): See table above.
- FCS_COP.1(2): See table above.
- FCS_COP.1(3): See table above.
- FCS_COP.1(4): See table above.
- FCS_HTTPS_EXT.1: The TOE implements HTTPS using TLS and compliant with RFC 2818.
- FCS_RBG_EXT.1: See table above.
- FCS_SSH_EXT.1: The TOE supports SSHv2 interactive command-line secure administrator sessions as indicated above.
- FCS_TLSC_EXT.2: The TOE acts as a TLS client when exporting audit logs to an external server.
- FCS_TLSS_EXT.1: The TOE provides TLS to support its web-based administrator interface.

6.3 Identification and authentication

The TOE defines administrative users in terms of:

- user identity; and
- password; and
- role.

Role permissions determine the functions that administrators may perform. Ten roles, each with a fixed set of permissions, are supported: Root, Factory, Admin, FabricAdmin, SecurityAdmin, SwitchAdmin, BasicSwitchAdmin, ZoneAdmin, Operator and User. There are four pre-defined administrator accounts called “root”, “factory”, “admin” and “user”, each of which is assigned the respective role of the same name, e.g. the “admin” account is assigned the Admin role. Note that neither the account called “user” nor any account that is assigned the User role, corresponds to a HBA that is attempting to access a storage device, rather a User-role account corresponds to an administrative user that can view but not change configuration settings. The internal

FabricOS root and factory accounts are disabled during TOE configuration, since they allow access to the operating system. Note that this FabricOS root account is not the same as the "Root" role.

The TOE authenticates administrative users accessing the TOE via the local console, SSH or web interface (HTTPS) in the same manner using either its own authentication mechanism. The TOE provides its own password authentication mechanism to authenticate administrative users. In order for an administrative user to access the TOE (i.e., to perform any functions except to see a configured login banner or to access network or SAN services), a user account including a user name and password must be created for the user, and an administrative role must be assigned. The TOE password authentication mechanism enforces password composition rules. Passwords must be between 8 and 40 characters; they can contain an alphabetic (upper or lower case) character; they can include numeric characters and special characters such as !, @, #, $, %, ^, &, *, (, and); and they are case-sensitive. The TOE supports several password policies which apply only to accounts defined within the local user database. Among these policies is a minimum length setting that allows an administrator to configure a minimum password length (from 8 to 40 characters) that will be enforced by the TOE when passwords are changed. Additionally, the SSH interface support public key authentication. Users are associated with a public key that must be provided during the authentication process.

The TOE uses only RSA certificates as its TLS server certificate on the HTTPS protected web management interface. Once the TOE is configured with a TLS certificate, user's must authenticate to the TOE using password-based authentication. Root certificates are loaded into the TOE during its initial configuration and the TOE uses those when making certificate checks. Likewise client certificates are loaded during its initial configuration and the TOE uses those for SYSLOG connections.

When authentication succeeds, the TOE looks up the user's defined privilege level, assigns that to the user's session, and presents the user with a command prompt.

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

- FIA_PMG_EXT.1: The TOE supports passwords comprising upper and lower case alphabetic characters, numbers, and a set of special characters identified above. The TOE also allows administrator to define a minimum password length between 8 and 40 characters.
- FIA_UAU.7: The TOE does not echo passwords as they are entered; rather either '*' or no characters are echoed when entering passwords.
- FIA_UAU_EXT.2: The TOE offers no TSF-mediated functions except display of a login banner and network and SAN services until the user is identified and authenticated.
- FIA_UIA_EXT.1: The TOE provides a password-based authentication mechanism, as well as public-key authentication for SSH. The order in which these authentication providers are checked is determined by an administrator.
- FIA_X509_EXT.1: OCSP is supported for X509v3 certificate validation. Certificates are validated as part of the authentication process when they are presented to the TOE and when they are loaded into the TOE. The following fields are verified as appropriate:
  - Expiration – certificate cannot be expired
  - Common Name - Needs to be FQN device name or IP. Wildcards are allowed only for 1 level of sub-domain and not allowed for the main domain.
  - CA Field – must be true if CA certificate
  - Key Usage - Need to have "Certificate Sign" incase of CA certificates and "Digital Signature" in case of identity certificates.
  - X509v3 Extended Key Usage - Need to rightly indicate whether it is for use as "server" certificate or "client" certificate. If incorrect, connection is not allowed

  - Authority Information Access - Must have valid OCSP server respond affirmatively. If this field is not present, OCSP check is not performed.
  - Subject Alternative Method - Not a mandatory attribute. If present, the values stored in this will take priority over the CN in Subject attribute.
  - Basics Constraints - Attribute must be present and must have CA Field

- FIA_X509_EXT.2: Certificates are checked and if found not valid are not accepted if the OCSP server cannot be contacted for validity checks, then the certificate is not accepted..
- FIA_X509_EXT.3: The TOE generates certificate requests and validates the CA used to sign the certificates

6.4 Security management

The TOE defines the following administrative roles all of which are considered an 'authorized administrator', albeit with differing actual capabilities, for the purpose of evaluation:

- admin – can perform all administrative commands
- switchAdmin – can perform administrative commands except for those related to user management and zoning configuration commands
- operator – can perform administrative commands that do not affect security settings
- zoneAdmin – can perform administrative commands that only affect zoning configuration
- fabricAdmin – can perform administrative commands except for those related to user management
- basicSwitchAdmin – can be used to monitor system activity
- securityAdmin – can perform security-related configuration including user management and security policy configuration
- root – can perform all administrative commands and access the OS; this user account is disabled during TOE configuration
- factory – can perform all administrative commands

The TOE administrative interfaces consist of an Ethernet network-based interface and a serial terminal-based interface. Ethernet interfaces use a command-line interface called the "FabricOS Command Line Interface" or an HTTPS based interface known as Web Tools. The FabricOS Command Line Interface is reached using SSH or the serial interface, while Web Tools supports the use of hypertext transfer protocol over secure socket layer (HTTPS). Both network-based and terminal-based interfaces provide equivalent management functionality. The Ethernet (i.e., SSH) and serial terminal interfaces support the same command-line interface commands after a session has been established.

Once authenticated (none of these functions is available to any user before being identified and authenticated), authorized administrators have access to the following security functions:

- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- Ability to configure a login banner as well as network routing and SAN functions;
- Ability to configure the cryptographic functionality.

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

- FMT_MOF.1(1): Only the authorized administrator can update the TOE.

- FMT_MOF.1(3): Only the authorized administrator can configure the audit trail for export to an external server.
- FMT_MOF.1(4): Only the authorized administrator can configure the audit logging functions.
- FMT_MTD.1(1): Only the authorized administrator can configure TSF-related functions.
- FMT_MTD.1(2): Cryptographic key related management is restricted to the authorized administrator.
- FMT_SMF.1: The TOE provides administrative interfaces to perform the functions identified above.
- FMT_SMR.2: The TOE maintains administrative user roles.

6.5 Protection of the TSF

The TOE is an appliance and as such is designed to work independent of other components to a large extent. Secure communication with third-party peers is addressed in section 6.8. As such, no additional protection (e.g., encryption) should be necessary in most operational environments.

While the administrative interface is function rich, the TOE is designed specifically to not provide access to locally stored passwords (which are protected using MD-5 hashing) and also, while cryptographic keys can be entered, the TOE does not disclose any cryptographic keys stored in the TOE. The TOE is a hardware appliance that includes a hardware-based real-time clock. The TOE's embedded OS manages the clock and exposes administrator clock-related functions. The TOE can be configured to periodically synchronize its clock with a time server, but the TOE can only ensure its own reliability and not that of an external time mechanism. The TOE also implements the timing elements through timeout functionality due to inactivity for terminating both local and remote sessions. Note that the clock is used primarily to provide timestamp for audit records, but is also used to support timing elements of cryptographic functions.

The TOE includes a number of built in diagnostic tests that are run during start-up to determine whether the TOE is operating properly. An administrator can configure the TOE to reboot or to stop, with errors displayed, when an error is encountered. When configured, the power-on self-tests comply with the FIPS 140-2 requirements for self-testing. The module performs cryptographic algorithm known answer tests, firmware integrity tests using RSA signature verification and conditional self-tests for PRNG, pair-wise consistency tests on generation of RSA keys, and a firmware load test (RSA signature verification). Upon failing any of its FIPS mode power-on self-tests, the TOE will refuse to boot.

The TOE supports loading a new software image manually by the administrator using CLI commands. From the CLI, an administrator can use the firmwareDownload command in order to download a new firmware image, and the TOE, prior to actually installing and using the new software image, will verify its digital certificate using the public key in the certificate configured in the TOE. An unverified image cannot be installed. When a new firmware is downloaded, the new firmware always replaces the public key file on the switch with what is in the new firmware.

The TOE generates time stamps to support the auditing function.

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

- FPT_APW_EXT.1: The TOE does not offer any functions that will disclose to any user a plain text password. Furthermore, locally defined passwords are not stored in plaintext form.
- FPT_SKP_EXT.1: The TOE does not offer any functions that will disclose to any users a stored cryptographic key.
- FPT_STM.1: The TOE generates time stamps for use in audit records.
- FPT_TST_EXT.1: The TOE includes a number of power-on diagnostics that will serve to ensure the TOE is functioning properly. The tests ensure the TOE is functioning as expected.
- FPT_TUD_EXT.1: The TOE provides function to query the version and upgrade the software embedded in the TOE appliance. When installing updated software, digital signatures are used to authenticate the update to ensure it is the update intended and originated by Brocade.

6.6 TOE access

The TOE can be configured to display an administrator-configured message of the day and banner that will be displayed before authentication is completed. In the case of the console and SSH, the message of the day is displayed before entering the user password and the banner is displayed afterwards. In the case of the web interface, the banner is displayed when connected to a session.

The TOE can be configured by an administrator to set a session timeout value (with 0 disabling the timeout and no timeout by default). Note that there are two timeout values – one applies to the console and SSH and the other applies to the web interface. A session (local console or remote SSH or Web/HTTPS) that is inactive (i.e., no commands issuing from the remote client) for the defined timeout value will be terminated. Upon exceeding the session timeout (if set), the TOE logs the user off, but leaves the user's console displaying the last contents in the case of the console or SSH (web sessions change to indicate an invalid session).

The user will be required to login in after any session has been terminated due to inactivity or after voluntary termination. Of course, administrators can logout of local or remote sessions at any time.

The TOE access function is designed to satisfy the following security functional requirements:

- FTA_SSL.3: The TOE terminates remote sessions that have been inactive for an administrator-configured period of time.
- FTA_SSL.4: The TOE provides the function to logout (or terminate) from the both local and remote user sessions as directed by the user.
- FTA_SSL_EXT.1: The TOE terminates local sessions that have been inactive for an administrator-configured period of time.
- FTA_TAB.1: The TOE can be configured to display administrator-defined advisory banners when administrators successfully establish interactive sessions with the TOE.

6.7 Trusted path/channels

The TOE provides a trusted path for its remote administrative users accessing the TOE via the Ethernet ports provided on the TOE using either the command line interface using SSH or Advanced Web Tools using TLS/HTTPS. Note that local administrator access via the serial port is also allowed for command line access. However this access is protected by physical protection of the serial interface along with the TOE itself.

When an administrator attempts to connect to the TOE remotely, the TOE attempts to negotiate a session. If the session cannot be negotiated, the connection is dropped. When negotiating a TLS/HTTPS or SSH session, the TOE and the client application (SSH client or web browser) used by the administrator will negotiate the most secure algorithms available at both ends to protect that session. The available algorithms are identified in section 6.2 above.

Remote connections to third-party SYSLOG servers are supported for exporting audit records to an external audit server and for external user authentication. Communication with those external servers is protected using TLS (as specified earlier).

In all cases, the endpoints are assured by virtue of the certificates installed, trusted, and reviewable when connecting and by virtue of user authentication.

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

- FTP_ITC.1: In the evaluated configuration, the TOE must be configured to use TLS to ensure that any authentication operations and exported audit records are sent only to the configured server so they are not subject to inappropriate disclosure or modification.
- FTP_TRP.1: The TOE uses SSH and HTTPS to provide a trusted path for remote management interfaces to protect the communication from disclosure and modification.