BROCADE
A Broadcom Inc. Company

USER GUIDE

# Brocade Fabric OS FIPS Cryptographic Module 8.2.x User Guide

## Supporting Fabric OS 8.2.x

*FOS-820-FIPS-Crypto-UG102*
*08 April 2019*

Copyright © 2019 Broadcom. All Rights Reserved. Broadcom, the pulse logo, Brocade, and the stylized B logo are among the trademarks of Broadcom in the United States, the EU, and/or other countries. The term “Broadcom” refers to Broadcom Inc. and/or its subsidiaries.

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.

The product described by this document may contain open source software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Broadcom products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit https://www.broadcom.com/.

# Contents

# Introduction



## About This Document

The Brocade Fabric OS® software uses the Brocade Fabric OS FIPS Cryptographic Module 8.2.x library to perform cryptographic functions. The module must be used in a Federal Information Processing Standards (FIPS) compliant operational environment along with the proper device configuration. This document provides the required conditions and configurations for a device to operate in a FIPS 140-2 compliant mode.

For additional information on the cryptographic module, refer to https://csrc.nist.gov/projects/cryptographic-module-validation-program/module-validation-lists.

## What Is New in This Document

The following section has been added or modified in this release:

- Updated Table 1 on page 4 with the FOS embedded switches.

## FIPS Compliant Operational Environment

FIPS certification in Fabric OS software releases prior to FOS 8.2.x referred to the entire operating system. In FOS 8.2.x, certification applies only to the cryptographic library of the cryptographic module and not to the operating system. The following table lists the FIPS-compliant operational environments for the Brocade Fabric OS® software and hardware platforms.

**TABLE 1** Compliant Operational Environment

| Hardware Platform | Operating System |
|---|---|
| Brocade G610 Switch | Fabric OS 8.2.0 |
| Brocade G620 Switch | Fabric OS 8.2.0 |
| Brocade G630 Switch | Fabric OS 8.2.0 |
| Brocade X6-4 Director | Fabric OS 8.2.0 |
| Brocade X6-8 Director | Fabric OS 8.2.0 |
| Brocade 6505 Switch | Fabric OS 8.2.0 |
| Brocade 6510 Switch | Fabric OS 8.2.0 |
| Brocade 6520 Switch | Fabric OS 8.2.0 |
| Brocade 7840 Extension Switch | Fabric OS 8.2.0 |
| Brocade DCX 8510-4 Backbone | Fabric OS 8.2.0 |
| Brocade DCX 8510-8 Backbone | Fabric OS 8.2.0 |
| Brocade 6543 16Gb FC Switch Blade for Huawei E9000 | Fabric OS 8.2.0a |

**TABLE 1** Compliant Operational Environment (continued)

| Hardware Platform | Operating System |
|---|---|
| Brocade 6547 FC5022 16Gb SAN Scalable Switch for Lenovo Flex System | Fabric OS 8.2.0a |
| Brocade 6558 16Gb FC Switch for HPE Synergy | Fabric OS 8.2.0a |
| Brocade M6505 FC Switch for Dell PowerEdge M1000e | Fabric OS 8.2.0a |
| Brocade G648 32Gb Fibre Channel SAN Switch Module for HPE Synergy | Fabric OS 8.2.0_GFT |
| Brocade G649 HPE Virtual Connect SE 32Gb FC Module for HPE Synergy** | Fabric OS 8.2.0_CBN |
| Brocade 7810 Extension Switch | Fabric OS 8.2.1 |

**This platform does not provide a CLI interface. To enable the FIPS 140-2 mode of operation, refer to the HPE documentation.

# Contacting Technical Support for Your Brocade® Product

For product support information and the latest information on contacting the Technical Assistance Center, go to https://www.broadcom.com/support/fibre-channel-networking/. If you have purchased Brocade® product support directly from Broadcom, use one of the following methods to contact the Technical Assistance Center 24x7.

| Online | Telephone |
|---|---|
| For nonurgent issues, the preferred method is to log in to myBroadcom at https://www.broadcom.com/mybroadcom. (You must initially register to gain access to the Customer Support Portal.) Once there, select **Customer Support Portal** > **Support Portal**. You will now be able to navigate to the following sites:<br>• **Knowledge Search**: Clicking the top-right magnifying glass brings up a search bar.<br>• **Case Management**: The legacy MyBrocade case management tool (MyCases) has been replaced with the Fibre Channel Networking case management tool.<br>• **DocSafe**: You can download software and documentation.<br>• **Other Resources**: Licensing Portal (top), SAN Health (top and bottom), Communities (top), Education (top). | Required for Severity 1 (critical) issues:<br>Please call Fibre Channel Networking Global Support at one of the numbers listed at https://www.broadcom.com/support/fibre-channel-networking/. |

If you purchased Brocade product support from a Broadcom OEM/solution provider, contact your OEM/solution provider for all your product support needs.

- OEM/solution providers are trained and certified by Broadcom to support Brocade products.
- Broadcom provides backline support for issues that cannot be resolved by the OEM/solution provider.
- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information on this option, contact Broadcom or your OEM.
- For questions regarding service levels and response times, contact your OEM/solution provider.

# Document Feedback

Quality is our first concern. We have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission or if you think that a topic needs further development, we want to hear from you. Send your feedback to documentation.pdl@broadcom.com. Provide the publication title, publication number, topic heading, page number, and as much detail as possible.

# Overview



# Overview of FIPS Mode Implementation

The FIPS mode implementation defines the cryptographic boundary of FIPS evaluation to include only the cryptographic library. FIPS 140-2 is evaluated at level 1.

This evaluation focuses on secure access and the validated cryptographic boundary within switches that are running Fabric OS 8.2.x firmware. Cryptographic functionality for the switches is provided by the OpenSSL crypto library.

**TABLE 2** Checklist for FIPS Compliance

| Item | Reference |
|---|---|
| FIPS-compliant operational environment | Table 1 on page 4 |
| Algorithms and protocols for FIPS mode | Table 3 on page 8 and Table 4 on page 9 |
| Self-tests | Status for Self-Tests on page 10 |
| Zeroization | Table 5 on page 11 |
| List of critical security parameters | Critical Security Parameters and Public Keys on page 18 |
| Third-party software | Third-party software |

# Fabric OS Implementation

With the Fabric OS 8.2.x implementation of FIPS, FIPS 140-2 is evaluated at level 1 for the cryptographic library, where the boundary for FIPS evaluation is limited to only the cryptographic library and not to the entire switch. The following are the prerequisites for the evaluation:

- The module or library within the FIPS boundary must run self-tests for algorithms and perform an integrity check for the library during load or bootup.
- Self-tests must be run every time the library is loaded.
- An integrity check must be run on the library.

The following configurations are supported for FIPS mode:

- FIPS mode enabled: With FIPS mode enabled, self-tests for algorithms and an integrity test of the cryptographic library are run every time the library is loaded.
- FIPS-9.xx enabled: This mode supports the IG 9.xx recommendation. Self-tests are run only once when the library is loaded after the switch powers up. The integrity test is run every time the library is loaded.
- FIPS mode/9.xx disabled: This mode turns off either FIPS mode or the FIPS-9.xx configuration. Neither algorithm self-tests nor an integrity test is performed at library load.

The Fabric OS 8.2.x implementation addresses the following:

- Support for a cryptographic key of security strength as per SP800-131A
- Algorithm and protocol validation tests to check for compliance
- Zeriozation methods

- Command line interface (CLI)
  - To configure algorithms and protocols
  - To restrict nonapproved/supported features/algorithms
  - To configure FIPS mode compliance

# Compliant Services

Fabric OS 8.2.x uses an embedded, FIPS-validated cryptographic module to support the security-relevant services. The embedded module is a software library API for cryptography functionality including DRBG output. Fabric OS 8.2.x services that use the embedded, approved cryptographic module include:

- SSH (Client and Server)
- TLS (HTTPS, SYSLOG, LDAP, and RADIUS)
- SNMP

The FIPS 140-2 validation boundary includes the underlying cryptographic library and not the entire switch. The following table lists the approved and nonapproved algorithms.

**TABLE 3** Cryptographic Algorithms

| Cryptographic Algorithms | Approved (Library) | Nonapproved |
|---|---|---|
| Encryption | **AES**: **ECB** (e/d; 128, 192, 256); **CBC** (e/d; 128, 192, 256)<br>**CFB128** (e/d; 128)<br>**CTR** (int only; 128, 192, 256) | **AES-GCM** (256 & 128)<br>**Camellia** (256 & 128)<br>**Seed** (128)<br>**RC4** (128)<br>**Triple DES**: **TCBC** KO-1 |
| Key Exchange | **ECC CDH**: Curves tested: P-256, P-384, P-512<br>**FFC**: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG, DPV, KPG) SCHEMES: Ephem: (KARole: Initiator / Responder) FC ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG, DPV, KPG) SCHEMES: EphemUnified: (KARole: Initiator / Responder) EC: P-256; ED: P-384; EE: P-512 | — |
| Secure Hash | **SHA**: **SHA-1** (BYTE-only)<br>**SHA-224** (BYTE-only)<br>**SHA-256** (BYTE-only)<br>**SHA-384** (BYTE-only)<br>**SHA-512** (BYTE-only) | **MD5** |
| Random Number Generation | **AES_CTR_DRBG**<br>Prediction Resistance Tested: Enabled; BlockCipher_Use_df: (AES-256) | — |
| Keyed Hash Message Authentication Code | **HMAC-SHA1** (Key Size Ranges: KS<BS)<br>**HMAC-SHA224** (Key Size Ranges: KS<BS)<br>**HMAC-SHA256** (Key Size Ranges: KS<BS)<br>**HMAC-SHA384** (Key Size Ranges: KS<BS)<br>**HMAC-SHA512** (Key Size Ranges: KS<BS) | **HMAC-MD5** |

**TABLE 3** Cryptographic Algorithms (continued)

| Cryptographic Algorithms | Approved (Library) | Nonapproved |
|---|---|---|
| Key-based Derivation Functions (KDF) | **TLS** (TLS1.0/1.1 TLS1.2 (SHA 256))<br>**SSH** (SHA 1 , 224 , 256 , 384 , 512)<br>**SNMP** SHA1 | — |
| Host Key Authentication | **DSA**: PQG(gen): [(2048,256)SHA(256);] PQG(ver): [(2048,256) SHA(256);] KeyPairGen: [(2048,256)]<br>**RSA**: KEY(gen): FIPS186-4_Fixed_e (65537); PGM(ProbPrimeCondition): 2048 , 3072 PPTT: C.2) ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(224, 256, 384, 512)) (3072 SHA(224, 256, 384, 512)) SIG(Ver) (1024 SHA( 224, 256, 384, 512)) (2048 SHA(224, 256, 384, 512)) (3072 SHA(224, 256, 384, 512))<br>**ECDSA**<br>PKG: CURVES(P-256 P-384 TestingCandidates) PKV: CURVES(P-256 P-384) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384))) | — |

Cryptographic modules on the switches support cryptographies that cater to multiple requirements. The following configurations are recommended (not mandated) in FIPS mode.

**TABLE 4** Cryptographic Compliance Protocol

| Protocol | FIPS-Approved Configuration |
|---|---|
| TLS Protocol | TLSv1.0/1.1 and TLSv1.2 |
| TLS Server (HTTPS) | !ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM |
| TLS Client (LDAP and RADIUS) | !ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM |
| PKI | Certificates (CA and Identity) should be of RSA 2048 key size and signed with SHA256. A CA certificate is required for the client applications (LDAP and RADIUS). |
| Encryption SSHv2 Server and Client | aes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, aes192-cbc, aes256-cbc |
| Key-Exchange (SSHv2 Server and Client) | ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp512, diffie-hellman-group-exchange-sha256 |
| MAC (SSHv2 Server and Client) | hmac-sha1, hmac-sha2-256, hmac-sha2-512 |
| SNMPv3 | SHA-1, AES-128 |

**NOTE**

Cipher suites for the TLS cipher strings mentioned above are explained below. This is output from OpenSSL and SSLv3 and indicates the minimum strings in which the cipher suite can be supported. However, SSLv3 is not supported in Fabric OS firmware.

```
openssl ciphers -V '!ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM'

AES256-SHA256 TLSv1.2 Kx=RSA  Au=RSA  Enc=AES(256)  Mac=SHA256
AES256-SHA    SSLv3   Kx=RSA  Au=RSA  Enc=AES(256)  Mac=SHA1
AES128-SHA256 TLSv1.2 Kx=RSA  Au=RSA  Enc=AES(128)  Mac=SHA256
AES128-SHA    SSLv3   Kx=RSA  Au=RSA  Enc=AES(128)  Mac=SHA1
DES-CBC3-SHA  SSLv3   Kx=RSA  Au=RSA  Enc=3DES(168) Mac=SHA1
```

## Validation Tests

Self-tests for algorithms must be executed during every load of the FIPS 140-2 certified cryptographic module to ensure that the implementation is working as evaluated. To enable algorithm self-tests, enable FIPS mode or 9.xx mode.

## Pairwise Consistency Tests

Pairwise consistency (ECDSA and RSA) tests are executed for the following scenarios:

1. An RSA key pair is used for encryption/decryption.
2. RSA keys are used for signing and verification of digital signatures.
3. ECDSA keys are tested for pairwise consistency for signing and verifying digital signatures.

## Continuous Random Number Generator Tests

Separate continuous random generator tests are performed for NDRNG (bits for /dev/random) and DRNG (bits from DRBG). The DRBG runs an internal check every time it is executed. The results of the self-test failures are logged to the console whenever they occur. The conditional tests are executed each time prior to using the random number provided by the random number generator.

## Self-Tests (Known Answer Tests - KATs)

Self-tests are invoked when the FIPS-approved cryptographic library is loaded.

## Status for Self-Tests

If the self-test fails, the respective application will fail, and the system will try to reboot. If the system goes into a reboot loop because the cryptography of the switch can no longer be validated or trusted, the Fabric OS firmware must be reinstalled to recover.

## Status for Integrity Tests

If the integrity test fails, the respective application will fail, and the system will try to reboot. If the system enters into a reboot loop because the cryptography of the switch can no longer be validated or trusted, the Fabric OS firmware must be reinstalled to recover.

# Zeroization Functions



# Overview of Zeroization

Zeroization is a method of erasing electronically stored data, cryptographic keys, and critical security parameters (CSPs) by altering or deleting the contents of the data storage to prevent the recovery of the data. Zeroization erases all potentially sensitive information in the switch memory. This includes erasing the main memory, cache memories, and any memory locations that may contain security data, including NVRAM and flash memory.

The restrictions of zeroization are:

- Zeroization parameters cannot be configured.
- Zeroization is invoked only through the command line interface (CLI).
- Zeroization is performed only by a local operator who has physical control of the cryptographic module, with all network connections physically disconnected.

# Zeroization Keys

Zeroization can be performed at the discretion of the security administrator to clear passwords, shared secrets, and other security modules.

Zeroization functions are implemented for the keys mentioned in the table below. A CLI command is used to zeroize the keys.

**TABLE 5** Zeroization Keys

| Keys | Zeroization | Remarks |
|---|---|---|
| DH private keys | Session termination or `fipscfg --zeroize` | DH private keys are zeroized within the code before they are released from memory. |
| SSH session key | Session termination or `fipscfg --zeroize` | This key is generated for each SSH session that is established with the host. It automatically zeroizes upon session termination. |
| SSH RSA private key | Session termination or `fipscfg --zeroize` | Key-based SSH authentication is not used for an SSH session. |
| DRBG seed material | Session termination or `fipscfg --zeroize` | /dev/random is used as the initial source for DRBG. The DRBG seed key is changed (zeroized) upon every random number generation within the FIPS-compliant OpenSSL DRBG implementation. |
| Passwords | `passwdDefault` or `fipscfg --zeroize` | The `passwdDefault` command removes user-defined accounts and restores the default passwords for the default root, admin, and user accounts. However, only the root account has permissions for this command. Users with securityadmin and admin permissions must use `fipscfg--zeroize`, which, in addition to removing user accounts and resetting passwords, also performs the complete zeroization of the system. |

**TABLE 5** Zeroization Keys (continued)

| Keys | Zeroization | Remarks |
|---|---|---|
| | | **NOTE**<br>In a dual-CP system, executing the `passwdDefault` command in the active CP synchronizes passwords with the standby CP. This causes user-defined accounts to be removed from both the active and standby CPs, and only the default accounts (root, factory, admin, and user) are retained. The passwords for these accounts are then set to the factory defaults.<br><br>**ATTENTION**<br>To maintain FIPS 140-2 compliance, passwords for the default accounts (admin and user) must be changed after every zeroization operation. |
| TLS private keys | `seccertmgmt generate` | The `seccertmgmt generate` command is available to zeroize these keys. Before generating new keys, it deletes old keys. |
| TLS pre-master secret | Session termination or `fipscfg --zeroize` | This is automatically zeroized upon session termination. |
| TLS master secret | Session termination or `fipscfg --zeroize` | This is automatically zeroized upon session termination. |
| TLS session key | Session termination or `fipscfg --zeroize` | This is automatically zeroized upon session termination. |
| TLS authentication key | Session termination or `fipscfg --zeroize` | This is automatically zeroized upon session termination. |
| RADIUS secret | `aaaconfig --remove` | The `aaaconfig -add` command configures the RADIUS server, and the `aaaconfig --remove` command zeroizes the secret and deletes the configured server. |
| SSH public keys | `sshutil delpubkeys` | The `delpubkeys` option zeroizes the SSH public keys. Only a public key of size 2048 is allowed. |
| LDAP CA certificate | `seccertmgmt delete -ldapcacert <certname>` | The specified LDAP certificate file is zeroized and deleted from the module. |
| RADIUS CA certificate | `seccertmgmt delete -ca <certname>` | The specified RADIUS certificate file is zeroized and deleted from the module. |
| SFTP session keys | No CLI session termination or `fipscfg --zeroize` | This is automatically zeroized upon session termination. All SFTP sessions are terminated upon zeroization. |
| ECDSA private key | Session termination or `fipscfg --zeroize` | This is deleted as part of the zeroize operation. |
| ECDSA K random value | Session termination or `fipscfg --zeroize` | This is deleted as part of the zeroize operation. |
| SNMPv3 auth and priv key | Session termination or `fipscfg --zeroize` | This deletes the auth and priv key and passwords of default users. |

# FIPS Mode Configuration



# Overview of FIPS Mode Configuration

Based on the FIPS mode that is configured on the device, one of the following tests is executed within the FIPS cryptographic module:

- FIP mode: Self-tests and integrity tests are run upon library load for every single instance.
- FIPS 9.xx mode: Integrity tests are run upon a library load for every single instance, and a self-test is run only once after power-up.

If either self-tests or integrity tests fail, an error message is generated and the system reboots. In the case of a chassis, only the failed CP will reboot. In the case of continuous failure, the switch enters into a rolling reboot.

# Enabling FIPS Mode

Perform the following procedure on the active CP to enable FIPS mode on the CP.

1. Log in to the switch (to the active CP in case of a chassis) as an authorized user.
2. Verify the firmware version using the `firmwareshow` command.

   ```
   firmwareshow
   ```

3. (Recommended) Zeroize the switch using the `fipscfg --zeroize` command.

   ```
   fipscfg --zeroize
   ```

4. Reboot the switch.
   - On a chassis: Execute the `reboot` command on both CPs to explicitly reboot the chassis.
   - On a fixed-port switch: Execute the `reboot` command.

5. Execute the `seccryptocfg` command to configure ciphers for SSH, TLS, RADIUS, and LDAP.
   a) Export the default_strong template from the switch using the `seccryptocfg` command.

```
seccryptocfg --export default_strong -server <IP address> -name <username> -proto scp -file
<remote file name>
```

   b) Edit the template to include only the ciphers as mentioned in Table 3 on page 8.

   Templates follow the `<name>:<value>` pair format. See the following examples to edit some of the template cipher configurations:

```
Enc:aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,aes192-cbc,aes256-cbc
Kex:ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256
Mac:hmac-sha1,hmac-sha2-256,hmac-sha2-512
Ciphers:!ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM
RAD_Ciphers:!ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM
LDAP_Ciphers:!ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM
Syslog_Ciphers:!ECDH:!DH:HIGH:-MD5:!CAMELLIA:!SRP:!PSK:!AESGCM
```

   c) Download the template and enable it using the `seccryptocfg` command.

```
seccryptocfg --import <custom template name> -server <server IP address> -name <username> -proto
scp -file <remote file name>
seccryptocfg --apply <custom template name>
```

   **NOTE**

   - When applying a template for AAA authentication, ensure that only valid ciphers listed in Table 3 on page 8 are used.
   - On directors, reboot the standby CP after the `seccryptocfg` command is issued on the active CP to configure cryptographic parameters.

6. (Recommended) Change the default password for admin and user by pressing `Enter` instead of `Ctrl+C` after logging in as admin.

   **NOTE**

   Passwords for default accounts (admin and user) should be changed after every zeroization to maintain FIPS 140-2 compliance.

7. (Recommended): Disable the root account using the `userconfig` command.

```
userconfig –change root –e no
```

8. Enable FIPS mode or FIPS 9.xx mode using the `fipscfg` command.

   To enable FIPS mode:

   ```
   fipscfg --enable fipsinside
   ```

   The following output displays.

   ```
   You are enabling fipsinside.
   Do you want to continue? (yes, y, no, n) [no]: y
   FIPS inside mode has been set to : Enabled
   ```

   To enable FIPS 9.xx mode:

   ```
   fipscfg --enable fipsinside -9.xx
   You are enabling fipsinside.
   Do you want to continue? (yes, y, no, n) [no]: y
   FIPS inside mode has been set to : Enabled(9.xx)
   admin> fipscfg --show
   FIPS mode is : Disabled
   FIPS Inside is : Enabled(9.xx)
   FIPS Selftests mode/status is : Enabled/None
   diffie-hellman-group-exchange-sha256 is : Disabled
   ```

   **NOTE**

   The output line `FIPS mode is : Disabled` means that the legacy FIPS mode is disabled.

9. Power-cycle the device or reboot the active node. On a dual-CP system, reboot the standby as well using the `fipscfg` command.

   ```
   fipscfg --show
   FIPS mode is :Disabled
   FIPS Inside is:Enabled
   ```

   The following is sample output if you entered a DP mode for the various switches.

   - X6-4 or X6-8

     ```
     fipscfg --show
     FIPS mode is : Disabled
     FIPS Inside is : Enabled(9.xx)
     FIPS Selftests mode/status is : Disabled/None
     diffie-hellman-group-exchange-sha256 is : Disabled
     ```

   - 7840

     ```
     fipscfg --show
     FIPS mode is : Disabled
     FIPS Inside is : Enabled
     FIPS Selftests mode/status is : Enabled/None
     ```

# Fabric OS Feature Configuration in FIPS Mode

The following table lists the typical configuration of switches that are running Fabric OS firmware in FIPS mode.

**TABLE 6** FIPS Mode Configuration

| Configuration | FIPS Mode (Supported) |
|---|---|
| Root account | It is recommended to disable. |
| Boot PROM | It is recommended to disable. |

**TABLE 6** FIPS Mode Configuration (continued)

| Configuration | FIPS Mode (Supported) |
|---|---|
| Ipfilter rules | It is recommended to block the unsecure ports or applications. |
| Certificates | It is recommended to use the approved key size, signature size, and hash type. |
| Disabling nonapproved services (for example, FCAP, inflight encryption) | N/A |
| Self-test suite run at switch boot time | It is recommended but it is not considered as a part of FIPS mode. |
| Passwords of default accounts | It is strongly recommended to modify the passwords to nondefault values at the strongest level. |

**NOTE**

FIPS-9.xx mode behaves the same as FIPS mode except that the self-tests run only once for the library. See Fabric OS Implementation on page 7.

The following table lists the `fipscfg` command and its options.

**TABLE 7** Command for Configuring FIPS Mode

| Command | Arguments and Parameters | Description |
|---|---|---|
| `fipscfg` | `--help` | Displays the help screen. |
| | `--show` | Displays the current FIPS configuration. |
| | `--enable [fipsinside [-9.xx] \| fips \| selftests \| bootprom \| simulate \| dh] [-nowarn] [-dp] [enable/disable]` | Enables the FIPS inside mode, FIPS mode, self-test mode, boot PROM access mode, simulation mode, or diffiehellman mode. |
| | `--disable [fipsinside \| selftests \| bootprom \| simulate \| dh] [-nowarn] [-dp] [enable/disable]` | Disables the FIPS inside mode, self-test mode, boot PROM access mode, simulation mode, or diffiehellman mode. |
| | `--zeroize [-nowarn] [-dp]` | Zeroizes the system. |
| | `--verify fips[-dp]` | States the results of the FIPS prerequisite tests. |

The following are examples of how to use the arguments and parameters of the `fipscfg` command.

This enables FIPS inside mode.

```
fipscfg --enable fipsinside
```

This disables the configuration of the FIPS KAT and conditional tests at switch bootup.

```
fipscfg --disable selftests
```

This enables the configuration of the FIPS KAT and conditional tests at system bootup. The KAT and conditional tests will be run even when FIPS mode is not configured.

```
fipscfg --enable selftests
```

This disables the boot PROM. Only root can execute this command, and it cannot be executed in FIPS mode.

```
fipscfg --disable bootprom
```

This enables the boot PROM access. Only root, admin, and security admin can execute this command.

```
fipscfg --enable bootprom
```

This command removes the Diffie-Hellman-specific SSH configuration in both the SSH server and the client configuration and restarts the SSH server.

```
fipscfg --disable dh
```

This command configures the Diffie-Hellman-specific SSH configuration in both the SSH server and the client configuration and restarts the SSH server.

```
fipscfg --enable dh
```

This zeroizes all the CSPs.

```
fipscfg --zeroize
```

This displays the current FIPS configuration.

```
fipscfg --show
```

# Critical Security Parameters and Public Keys

The following information is representative of critical security parameters (CSPs) in platforms that support both CP and DP with IPsec capability.

**TABLE 8** IPsec Support

| Platform | IPsec Support |
|---|---|
| X6 | Yes |
| G620 | No |
| 7840 | No |
| 7810 | No |

The following CSPs and public keys supported on the platforms are:

1. DH Private Keys for use with the 2048-bit modulus.
2. SSHv2/SCP/SFTP Encryption Keys.
3. SSHv2/SCP/SFTP Authentication Key.
4. SSHv2 KDF Internal State.
5. SSHv2 DH Shared Secret Key (2048-bit).
6. SSHv2 ECDH Shared Secret Key (P-256, P-384, and P-512).
7. SSHv2 ECDH Private Key (P-256, P-384, and P-512).
8. SSHv2 ECDSA Private Key (P-256).
9. Value of K during SSHv2 P-256 ECDSA session.
10. TLS Private Key (RSA 2048).
11. TLS Pre-Master Secret.
12. TLS Master Secret.
13. TLS KDF Internal State.
14. TLS Session Keys - 128-, 256-bit AES CBC.
15. TLS Authentication Key for HMAC-SHA-1 (160 bits) and HMAC-SHA-256.
16. CP DRBG Seed Material.
17. CP DRBG Internal State (V and Key).
18. Passwords.
19. RADIUS Secret.
20. DH Private Key (256 bits) (Used in IKEv2).
21. DH Shared Secret (2048 bits) (Used in IKEv2).
22. IKEv2 AES-256 Encrypt/Decrypt Keys.

23. ESP AES-256-GCM Encrypt/Decrypt Keys.
24. IKEv2 KDF State.
25. IKEv2 Authentication Key (PSK).
26. IKEv2 ECDH P-384 Private Key.
27. IKEv2 ECDSA P-384 Private Key.
28. IKEv2 Integrity Key (HMAC-SHA-384).
29. DRBG Internal State (V and Key) (On Cavium).
30. Entropy Data (On Cavium).
31. SNMPv3 Auth and Priv password.
32. SNMPv3 KDF Internal State.
33. SNMPv3 Auth and Priv Secrets.
34. DH Public Key (2048-bit modulus).
35. DH Peer Public Key (2048-bit modulus).
36. TLS Public Key (RSA 2048).
37. TLS Peer Public Key (RSA 2048).
38. FW Download Public Key (RSA 2048).
39. SSHv2 ECDSA Public Key (P-256).
40. SSHv2 ECDSA Peer Public Key (P-256).
41. LDAP ROOT CA certificate (RSA 2048).
42. RADIUS ROOT CA certificate (RSA 2048).
43. DH Public Key (2048 bits) (Used in IKEv2).
44. DH Peer Public Key (2048 bits) (Used in IKEv2).
45. IKEv2 ECDH P-384 Public Key.
46. IKEv2 ECDH P-384 Peer Public Key.
47. IKEv2 ECDSA P-384 Public Key.
48. IKEv2 ECDSA P-384 Peer Public Key.
49. SSHv2 ECDH Public Key (P-256, P-384, and P-512).
50. SSHv2 ECDH Peer Public Key (P-256, P-384, and P-512).

# Revision History

## FOS-820-FIPS-Crypto-UG102; 08 April 2019

- Updated Table 1 on page 4 with the FOS embedded switches.

## FOS-820-FIPS-Crypto-UG101; 04 January 2019

- Updated Table 1 on page 4 with the Brocade 7810 Extension Switch and FOS embedded switches.

## FOS-820-FIPS-Crypto-UG100; 20 September 2018

- Initial release.