



# FIPS 140-2 Security Policy for Cisco 4402 and 4404 Wireless LAN Controllers

---

October 2, 2009  
Policy Version 1.7

## Contents

This security policy contains these sections:

- [Overview, page 2](#)
- [Physical Security Policy, page 3](#)
- [Secure Configuration, page 4](#)
- [Roles, Services, and Authentication, page 8](#)
- [Cryptographic Key Management, page 10](#)
- [Disallowed Security Functions, page 17](#)
- [Obtaining Documentation, Obtaining Support, and Security Guidelines, page 18](#)



---

Corporate Headquarters:  
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

© 2009 Cisco Systems, Inc. All rights reserved.

# Overview

The Cisco 4402 and 4404 Wireless LAN Controllers (collectively referred to as *the module*) support Control and Provisioning of Wireless Access Points (CAPWAP) and Wi-Fi Protected Access 2 (WPA2) security. CAPWAP uses DTLS to provide a secure link over which CAPWAP control messages are sent. DTLS is essentially TLS, but over datagram (UDP) transport. WPA2 is the approved Wi-Fi Alliance interoperable implementation of the IEEE 802.11i standard.

It automatically detects, authorizes and configures access points, setting them up to comply with the centralized security policies of the wireless LAN. In a wireless network operating in this mode, WPA2 protects all wireless communications between the wireless client and other trusted networked devices on the wired network with AES-CCMP encryption. CAPWAP protects all control and bridging traffic between trusted network access points and the module with AES-CCM encryption.

The module supports HTTPS using TLS, CAPWAP, WPA2 (802.11i), MFP, RADIUS KeyWrap (using AES key wrapping), IPSec, Local-EAP, EAP-FAST, TACACS+, and SNMP. HTTPS using TLS uses 1536 bit modulus RSA keys to wrap 128 bit AES symmetric keys, and RADIUS KeyWrap uses 128 bit AES keys as key encrypting keys to wrap AES keys of up to 128 bits. It is a multiple-chip standalone cryptographic module, compliant with all requirements of FIPS 140-2 Level 2. The cryptographic boundary of the module includes all hardware and firmware. The evaluated platform consists of model numbers 4402 and 4404, with firmware version 5.2.157.0 or 5.2.178.5, hardware revision A0, and opacity baffle version 1.0.

In the FIPS mode of operations, the module supports the following cryptographic algorithm implementations:

- AES (AES Cert. #960, vendor affirmed; key wrapping; key establishment methodology provides 128 bits of encryption strength)
- AES-CBC (firmware)
- AES-ECB (firmware)
- AES-CCM (firmware)
- SHA-1 (firmware)
- HMAC SHA-1 (firmware)
- FIPS 186-2 Random Number Generator (firmware)
- RSA signature verification (firmware)
- RSA (key wrapping; key establishment methodology provides 96 bits of encryption strength)
- TDES (firmware)
- Diffie-Hellman (key agreement; key establishment methodology provides 112 bits of encryption strength)

The module is interoperable with all FIPS 140-2 validated wireless LAN clients that support the ratified IEEE 802.11i standard.

This document details the security policy for the module. This security policy may be freely distributed.

# Physical Security Policy

Put tamper evident labels over the service port and over the right side of the face on the front panel as shown in [Figure 1](#). The tamper evident label placed on the right side of the front panel must wrap over onto the top cover of the chassis. Put tamper evident labels on the rear panel and the removable cover as shown in [Figure 2](#). Screw the opacity baffle onto the right side of the module, and affix tamper evident labels on the top and bottom as shown in [Figure 3](#) and [Figure 4](#). [Figure 5](#) shows the full cryptographic boundary of the module, which includes the modules case and the opacity baffle.

**Figure 1** Placement of Tamper-evident Labels (Front View)



**Figure 2** Placement of Tamper-evident Labels (Rear View)



**Figure 3** Placement of Opacity Baffle (Top View)



**Figure 4** *Placement of Opacity Baffle (Bottom View)*



**Figure 5** *Cryptographic Boundary*



## Secure Configuration

Initial configuration of the module shall be performed over a local link through the console connection. The Crypto Officer must ensure that the PC that is used for the console connection is a stand-alone or non-networked PC. After the first three steps below, remote access through HTTPS may be used for subsequent configuration. The service port shall not be used to configure the module. For connecting using HTTPS, the Crypto Officer shall configure their web browsers so that only TLS v1.0 is used. The HTTPS client must be configured to use AES\_128\_CBC\_SHA based cipher suites.

Only the CAPWAP firmware version 5.2.157.0 or 5.2.178.5 may be loaded on the wireless LAN controllers for distribution to access points.

Follow these steps to prepare the secure configuration for the module:

1. [Enable FIPS Mode of Operations](#)
2. [Disable Boot Break](#)
3. [Configure HTTPS Certificate](#)
4. [Configure Authentication Data](#)
5. [Configure Communications with RADIUS](#)
6. [Configure Pre-shared Keys for 802.11i](#)
7. [Configure Ciphersuites for 802.11i](#)
8. [Configure SNMP](#)
9. [Configure TACACS+ secret](#)
10. [Configure MFP \(Management Frame Protection\)](#)
11. [Configure Local EAP](#)

12. [Configure EAP-FAST](#)
13. [Configure EAP-TLS](#)
14. [Save and Reboot](#)

## Enable FIPS Mode of Operations

The following CLI command places the controller in FIPS mode of operations, enabling all necessary self tests and algorithm restrictions:

```
> config switchconfig fips-prerequisite enable
```

## Disable Boot Break

The following CLI command prevents breaking out of the boot process. It must be executed after enabling FIPS mode of operations.

```
> config switchconfig boot-break disable
```

## Configure HTTPS Certificate

The following command configures the controller to use the manufacture-installed Cisco device certificate for the HTTPS server. It must be executed after enabling FIPS mode of operations:

```
> config certificate use-device-certificate webadmin
```

## Configure Authentication Data

All users shall have a password containing 8 or more characters, including numbers and letters. A crypto officer can use the following CLI command to set user passwords:

```
>config mgmtuser password username password
```

Note that this and all subsequent configuration steps may also be performed through HTTPS. However, only the CLI commands are included in this document.

## Configure Communications with RADIUS

Communications between the controller and RADIUS may be configured for RADIUS KeyWrap or IPSec.

### RADIUS KeyWrap and MACK Keys

The following CLI commands configure the RADIUS secret and AES-key wrap KEK and MACK:

```
> config radius auth add index ip-address port hex secret
> config radius auth keywrap add hex kek mack index
> config radius auth keywrap enable
```

## IPSec

Optionally, the controller may be configured to communicate with RADIUS via IPSec. Refer to the document at the following link for additional instructions:

[http://www.cisco.com/en/US/products/ps6366/products\\_tech\\_note09186a0080a829b8.shtml](http://www.cisco.com/en/US/products/ps6366/products_tech_note09186a0080a829b8.shtml)

## Configure Pre-shared Keys for 802.11i

WPA2 Pre-shared key (WPA2-PSK) is an optional mode permitted by this security policy. Generation of pre-shared keys is outside the scope of this security policy, but they should be entered as 64 hexadecimal values (256 bits) by the following command syntax:

```
> config wlan security wpa akm psk set-key hex key index
> config wlan security wpa akm psk enable index
```

Refer to Cisco Wireless LAN Controller Configuration Guide for additional instructions.

## Configure Ciphersuites for 802.11i

The following CLI commands create a wireless LAN, configure it to use WPA2, associate it with a RADIUS server, and enable it:

```
> config wlan create index profile_name ssid
> config wlan radius_server auth add index radius-server-index
> config wlan enable index
```

## Configure SNMP

Non-security related remote monitoring and management of the Controller can be done via SNMP. No CSPs are entered or output via SNMP. Only SNMPv3 with HMAC-SHA-1 is permitted by this security policy. The user passwords shall be selected to be 8 or more characters, including numbers and letters.

The following CLI commands enable SNMPv3 with HMAC-SHA1:

```
> config snmp version v1 disable
> config snmp version v2c disable
> config snmp version v3 enable
> config snmp v3user create username <ro|rw> hmacsha authkey encryptkey
```

## Configure TACACS+ secret

The crypto officer may configure the module to use TACACS+ for authentication, authorization and accounting. Configuring the module to use TACACS+ is optional. If the module is configured to use TACACS+, the Crypto-Officer must define TACACS+ shared secret keys that are at least 8 characters long. The following CLI command configures TACACS+ for authentication (auth), authorization (athr) and accounting (acct):

```
> config tacacs <auth|athr|acct> add index ip port <ascii|hex> secret
```

Refer to the Cisco Wireless LAN Controller Configuration Guide for additional instructions.

## Configure MFP (Management Frame Protection)

Infrastructure MFP enables one access point to validate a neighboring Access Point's management frames. Configuring the module to use MFP is optional. The following CLI command is used to enable infrastructure MFP:

```
> config wps mfp infrastructure enable
```

Client MFP is used to encrypt and sign management frames between the AP and the client. The following CLI command is used to enable client MFP:

```
> config wlan mfp client enable index required
```

Refer to the Cisco Wireless LAN Controller Configuration Guide for additional instructions.

## Configure Local EAP

The module can be optionally configured as a local EAP authentication server to authenticate wireless clients. Both EAP-TLS and EAP-FAST are supported and permitted by this security policy.

Refer to the Cisco Wireless LAN Controller Configuration Guide for instructions on configuring Local EAP server to authenticate wireless clients without a RADIUS server.

## Configure EAP-FAST

EAP-FAST is an Extensible Authentication protocol and can be used as an authentication method between the Controller and the wireless client. When a RADIUS server is used to authenticate clients, no extra EAP-FAST configuration is required.

The following CLI command is used by the crypto officer to enter a new EAP-FAST server key, where hex-key can be up to 32 hex digits or 16 bytes.

```
> config local-auth method fast server-key hex-key
```

Refer to the Cisco Wireless LAN Controller Configuration Guide for instructions on configuring Local EAP server with EAP-FAST as the authentication method for the wireless clients.

## Configure EAP-TLS

EAP-TLS is an Extensible Authentication protocol and can be used as an authentication method between the Controller and the wireless client. It requires configuration based on certificates issued from a PKI. Refer to the *Cisco EAP-TLS Deployment Guide for Wireless LAN Networks* configuration instructions to use EAP-TLS as the authentication method for the wireless clients.

Click this URL for an example configuration:

[http://www.cisco.com/en/US/tech/tk722/tk809/technologies\\_configuration\\_example09186a0080851b42.shtml](http://www.cisco.com/en/US/tech/tk722/tk809/technologies_configuration_example09186a0080851b42.shtml)

## Save and Reboot

After executing the above commands, you must save the configuration and reboot the system:

```
> save config
> reset system
```

## Roles, Services, and Authentication

This section describes the roles, services, and authentication types in the security policy.

### Roles

The module supports these four roles:

- AP Role—This role is filled by an access point associated with the controller.
- Client Role—This role is filled by a wireless client associated with the controller.
- User Role—This role performs general security services including cryptographic operations and other approved security functions. The product documentation refers to this role as a management user with read-only privileges.
- Crypto Officer (CO) Role—This role performs the cryptographic initialization and management operations. In particular, it performs the loading of optional certificates and key-pairs and the zeroization of the module. The product documentation refers to this role as a management user with read-write privileges.

The module does not support a maintenance role.

### Services

The services provided are summarized in [Table 1](#).

**Table 1**      *Module Services*

Service	Role	Purpose
Self Test and Initialization	CO	Cryptographic algorithm tests, firmware integrity tests, module initialization.
System Status	User or CO	The LEDs show the network activity and overall operational status and the command line status commands output system status.
Key Management	CO	Key and parameter entry, key output, key zeroization.
Module Configuration	CO	Selection of non-cryptographic configuration settings.
SNMPv3	CO	Non security related monitoring by the CO using SNMPv3.



**Table 1**      **Module Services (continued)**

Service	Role	Purpose
TACACS+	User or CO	User & CO authentication to the module using TACACS+.
IPSec	User or CO	Secure communications between controller and RADIUS
CAPWAP	AP	Establishment and subsequent data transfer of a CAPWAP session for use between the module and an access point. <sup>1</sup>
MFP	AP	Generation and subsequent distribution of MFP key to the AP over a CAPWAP session.
TLS	CO	Establishment and subsequent data transfer of a TLS session for use between the module and the CO.
Local EAP Authenticator	Client	Establishment of EAP-TLS or EAP-FAST based authentication between the client and the Controller.
802.11i	AP	Establishment and subsequent data transfer of an 802.11i session for use between the client and the access point.
RADIUS KeyWrap	Any	Establishment and subsequent receive 802.11i PMK from the RADIUS server.

1. CAPWAP uses RSA Key wrapping which provides 96 bits of effective key strength.

The module does not support a bypass capability in the approved mode of operations.

## Ports and Interfaces

The module has the following physical ports and interfaces:

- Service and Utility Ethernet interfaces (these interfaces are not used in FIPS mode of operations)
- Console serial port
- Two (4402) or four (4404) Small Form-factor Pluggable (SFP) interfaces
- Power port
- LEDs
  - Link and activity indicators for the Ethernet and SFP interfaces
  - PS1/PS2 power supply status indicators; Red indicates a power supply error
  - Status LED to indicate the module is operating normally
  - Alarm LED; Red indicates a module system error

## User and CO Authentication

When a user first connects to the module via console port, the module prompts the user to enter a username and password. The user is authenticated based on the password provided. Once the user has been authenticated, the module provides services to that user based on whether they have read-only privileges (the user role) or read-write privileges (the CO role). The "\*" characters are used to mask user password when the users authenticate. If the incorrect password is entered, the module will re-prompt the user to login again. After the module power cycles, a user must reauthenticate.

The module supports password based local authentication for access via the CLI or HTTPS, as well as remote authentication using RADIUS and TACACS+. The module also supports remote access via SNMPv3. All SNMP traffic to and from the module is considered unprotected. RADIUS, TACACS+ and SNMPv3 may be used in the FIPS mode.

The security policy stipulates that all user passwords must be 8 alphanumeric characters, so the password space is 2.8 trillion possible passwords. The possibility of randomly guessing a password is thus far less than one in one million. To exceed a one in 100,000 probability of a successful random password guess in one minute, an attacker would have to be capable of 28 million password attempts per minute, which far exceeds the operational capabilities of the module to support.

## AP Authentication

The module performs mutual authentication with an access point through the CAPWAP protocol, using an RSA key pair with 1536 bit modulus, which has an equivalent symmetric key strength of 96 bits. An attacker would have a 1 in  $2^{96}$  chance of randomly obtaining the key, which is much stronger than the one in a million chance required by FIPS 140-2. To exceed a one in 100,000 probability of a successful random key guess in one minute, an attacker would have to be capable of approximately  $7.9 \times 10^{23}$  attempts per minute, which far exceeds the operational capabilities of the module to support.

## Client Authentication

The module performs mutual authentication with a wireless client through EAP-TLS or EAP-FAST protocols. EAP-FAST is based on EAP-TLS and uses EAP-TLS key pair and certificates. The RSA key pair for the EAP-TLS credentials has modulus size of 1024 bit to 2048 bit, thus providing between 80 bits and 112 bits of strength. Assuming the low end of that range, an attacker would have a 1 in  $2^{80}$  chance of randomly obtaining the key, which is much stronger than the one in a million chance required by FIPS 140-2. To exceed a one in 100,000 probability of a successful random key guess in one minute, an attacker would have to be capable of approximately  $1.8 \times 10^{21}$  attempts per minute, which far exceeds the operational capabilities of the modules to support.

## Cryptographic Key Management

Cryptographic keys are stored in plaintext form, in flash for long-term storage and in SDRAM for active keys. The AES key wrap KEK, AES key wrap MAC keys, and the Pre shared key (PSK) are input by the CO in plaintext over a local console connection. The PMK and NSK are input from the RADIUS server encrypted with the AES key wrap protocol or via IPsec. RSA public keys are output in plaintext in the form of X.509 certificates. The CAPWAP session key is output wrapped with the AP's RSA key, and the MFP MIC key and 802.11i PTK, 802.11i GTK are output encrypted with the CAPWAP session key. PAC

key is output wrapped with the Client's RSA key. Asymmetric key establishment (RSA key transport) is used in the creation of session keys during EAP-TLS and EAP-FAST. Any keys not explicitly mentioned are not input or output.

[Table 2](#) lists the secret and private cryptographic keys and CSPs used by the module. [Table 3](#) lists the public keys used by the module. [Table 4](#) lists the access to the keys by service.

**Table 2** *Secret and Private Cryptographic Keys and CSPs*

Name	Algorithm	Storage	Description
PRNG seed key	FIPS 186-2	Flash	This is the seed key for the PRNG. It is statically stored in the code.
PRNG seed	FIPS 186-2	SDRAM	This is the seed for the PRNG. It is generated using an un-approved RNG based on the controller's /dev/urandom device.
DTLS Pre-Master Secret	Shared secret	SDRAM	Shared secret generated by approved RNG for generating the DTLS encryption key.
DTLS Encryption Key (CAPWAP Session Key)	AES-CBC	SDRAM	Session key used to encrypt and decrypt CAPWAP control messages.
DTLS Integrity Key	HMAC- SHA-1	SDRAM	Session key used for integrity checks on CAPWAP control messages.
AAA Shared Secret	TDES	Flash	Used to derive IPsec encryption keys and IPsec HMAC keys.
RADIUSOverIPsec EncryptionKey	TDES	SDRAM	TDES encryption/decryption key, used in IPsec tunnel between module and RADIUS to encrypt/decrypt EAP keys.
RADIUSOverIPsec IntegrityKey	HMAC	SDRAM	Integrity/authentication key, used in IPsec tunnel between module and RADIUS.
User Password	Shared secret	Flash	Identity-based authentication data for a user.
SNMPv3 Password	Shared secret	Flash	This secret is used to derive HMAC-SHA1 key for SNMPv3 authentication.
TACACS+ authentication secret	Shared secret	Flash	This TACACS+ shared secret is used to obfuscate the Crypto-Officer's authentication requests and responses between the module and the TACACS+ server. Entered by the Crypto-Officer in plaintext form and stored in plaintext form. Note that encryption algorithm is not FIPS compliant and the Crypto-Officer must ensure a strong user password.

**Table 2**      **Secret and Private Cryptographic Keys and CSPs (continued)**

Name	Algorithm	Storage	Description
TACACS+ authorization secret	Shared secret	Flash	This TACACS+ shared secret is used to obfuscate the Crypto-Officers' operation's authorization requests and responses between the module and the TACACS+ server. Entered by the Crypto-Officer in plaintext form and stored in plaintext form. Note that encryption algorithm is not FIPS compliant.
TACACS+ accounting secret	Shared secret	Flash	This TACACS+ shared secret is used to obfuscate accounting requests and responses between the module and the TACACS+ server. Entered by the Crypto-Officer in plaintext form and stored in plaintext form. Note that encryption algorithm is not FIPS compliant.
bsnOldDefaultIdCert	RSA	Flash	1536-bit RSA private key used to authenticate to the access point, generated during the manufacturing process.
bsnDefaultIdCert	RSA	Flash	1536-bit RSA private key, not used in FIPS mode.
bsnSslWebadminCert	RSA	Flash	1536-bit RSA private key used for HTTPS-TLS, generated during the manufacturing process.
bsnSslWebauthCert	RSA	Flash	1024-bit RSA private key, not used in FIPS mode.
VendorDeviceCert	RSA	Flash	Certificate to authenticate controller to EAP clients during EAP authentication. It may be used in EAP-TLS or EAP-FAST authentication method.
HTTPS TLS Pre-Master Secret	Shared secret	SDRAM	Shared secret created using asymmetric cryptography from which new HTTPS session keys can be created.
HTTPS TLS Encryption Key	AES-CBC	SDRAM	AES key used to encrypt HTTPS data.
HTTPS TLS Integrity Key	HMAC- SHA-1	SDRAM	HMAC-SHA-1 key used for HTTPS integrity protection.
Infrastructure MFP MIC Key	AES-CMAC	Flash	This 128-bit AES key is generated in the controller using FIPS 186-2 approved RNG. This key is sent to the AP encrypted with the DTLS encryption key. This key is used by the AP to sign management frames when infrastructure MFP is enabled.
Pre-Shared Key (PSK)	AES-CCM	Flash	The 802.11i pre shared key (PSK). This key is optionally used as a PMK.

**Table 2**      **Secret and Private Cryptographic Keys and CSPs (continued)**

Name	Algorithm	Storage	Description
802.11i Pairwise Master Key (PMK)	Shared secret	SDRAM	The PMK is a secret shared between an 802.11 supplicant and authenticator, and is used to establish the other 802.11i keys.
802.11i Key Confirmation Key (KCK)	HMAC- SHA-1	SDRAM	The KCK is used by IEEE 802.11i to provide data origin authenticity in the 4-Way Handshake and Group Key Handshake messages.
802.11i Key Encryption Key (KEK)	AES-KeyWrap	SDRAM	The KEK is used by the EAPOL-Key frames to provide confidentiality in the 4-Way Handshake and Group Key Handshake messages.
802.11i Pairwise Transient Key (PTK)	AES-CCM	SDRAM	The PTK, also known as the CCMP key, is the 802.11i session key for unicast communications.
802.11i Group Temporal Key (GTK)	AES-CCM	SDRAM	The GTK is the 802.11i session key for broadcast communications.
RADIUS AES KeyWrap KEK	AES-KeyWrap	Flash	The key encrypting key used by the AES Key Wrap algorithm to protect the PMK for the 802.11i protocol.
RADIUS AES KeyWrap MACK	AES-KeyWrap	Flash	The MAC key used by the AES Key Wrap algorithm to authenticate RADIUS conversation.
EAP-TLS Pre-Master Secret	Shared secret	SDRAM	Shared secret created using asymmetric cryptography from which new EAP-TLS session keys can be created.
EAP-TLS Encryption Key	AES-CBC	SDRAM	AES key used to encrypt EAP-TLS session data.
EAP-TLS Integrity Key	HMAC- SHA-1	SDRAM	HMAC-SHA-1 key used for EAP-TLS integrity protection.
EAP-TLS Peer Encryption Key	Shared secret	SDRAM	This 32-byte key is master session key of the EAP-TLS authentication algorithm. It is the PMK for 802.11i.
EAP-FAST Server Key	AES-CCM	Flash	EAP-FAST server master key to generate client protected access credential (PAC).
EAP-FAST PAC-Key	Shared secret	SDRAM	Shared secret between the local EAP authenticator and the wireless client. For EAP-FAST authentication. It is created by PRNG and is used to derive EAP-FAST tunnel master secret.
EAP-FAST tunnel master secret	Shared Secret	SDRAM	This is the master secret for EAP-FAST. It is used to derive EAP-FAST Encryption key, EAP-FAST Integrity key, EAP-FAST Session Key Seed.
EAP-FAST Encryption Key	AES-CBC	SDRAM	Encryption Key for EAP-FAST tunnel.

**Table 2** *Secret and Private Cryptographic Keys and CSPs (continued)*

Name	Algorithm	Storage	Description
EAP-FAST Integrity Key	HMAC-SHA-1	SDRAM	Integrity Key for EAP-FAST tunnel.
EAP-FAST Session-Key Seed	Shared Secret	SDRAM	This secret is used to derive the EAP-FAST master session key by mixing with the EAP-FAST Inner Method Session Key.
EAP-FAST Inner Method Session Key	Shared Secret	SDRAM	This 32-byte key is the session key generated by the EAP handshake inside the EAP-FAST tunnel.
EAP-FAST Master Session Key	Shared Secret	SDRAM	This 64-byte key is the session key generated by the EAP-FAST authentication method. It is then used as PMK for 802.11i.

**Table 3** *Public Keys*

Name	Algorithm	Storage	Description and Zeroization
bsnOldDefaultCaCert	RSA	Flash	Verification certificate, used for CAPWAP authentication.
bsnDefaultRootCaCert	RSA	Flash	Verification certificate, used to validate the controller's firmware image.
bsnDefaultCaCert	RSA	Flash	Verification certificate, used for CAPWAP authentication.
bsnDefaultBuildCert	RSA	Flash	Verification certificate, used to validate the controller's firmware image.
cscsDefaultNewRootCaCert	RSA	Flash	Verification certificate, used with CAPWAP to validate the certificate that authenticates the access point.
cscsDefaultMfgCaCert	RSA	Flash	Verification certificate, used with CAPWAP to authenticate the access point.
cscsDefaultDevCaCert	RSA	Flash	Verification certificate, used with CAPWAP to authenticate the access point.
cscsDefaultR3CaCert	RSA	Flash	Verification certificate, not used in FIPS mode of operations.
bsnOldDefaultIdCert	RSA	Flash	Authentication certificate, used to authenticate to the access point.

**Table 3**      *Public Keys (continued)*

Name	Algorithm	Storage	Description and Zeroization
bsnSslWebadminCert	RSA	Flash	Server certificate used for HTTPS-TLS.
VendorCACert	RSA	Flash	Certificate to validate wireless clients certificates during EAP authentication. It may be used in EAP-TLS or EAP-FAST authentication method.

**Table 4**      *Key/CSP Access by Service*

Service	Key Access
Self Test and Initialization	<ul style="list-style-type: none"> <li>Initializes PRNG seed</li> </ul>
System Status	<ul style="list-style-type: none"> <li>None</li> </ul>
Key Management	<ul style="list-style-type: none"> <li>Read/Write PSK, AAA Shared Secret, PSK, RADIUS AES KeyWrap KEK, RADIUS AES KeyWrap MACK, EAP-FAST Server Key</li> <li>Destroy all keys (with Key Zeroization command)</li> </ul>
Module Configuration	<ul style="list-style-type: none"> <li>Modify user passwords</li> <li>Modify TACACS+ shared secret</li> </ul>
SNMPv3	<ul style="list-style-type: none"> <li>Authenticate using SNMPv3 user password</li> </ul>
TACACS+	<ul style="list-style-type: none"> <li>Authenticate, authorize and accounting using TACACS+ shared secrets</li> </ul>
IPSec	<ul style="list-style-type: none"> <li>Use AAA Shared Secret, RADIUSOverIPSecEncryptionKey, and RADIUSOverIPSecIntegrityKey</li> </ul>
CAPWAP	<ul style="list-style-type: none"> <li>Verify with ciscoDefaultNewRootCaCert and ciscoDefaultMfgCaCert</li> <li>Sign with bsnOldDefaultIdCert Private Key</li> <li>Read (and transmit) bsnOldDefaultIdCert Certificate</li> <li>Establish and then encrypt/decrypt with CAPWAP Session Key</li> </ul>
MFP	<ul style="list-style-type: none"> <li>Derive Infrastructure MFP MIC key from PRNG and distribute to connected APs</li> </ul>
HTTPS (TLS)	<ul style="list-style-type: none"> <li>Sign with bsnSslWebadminCert Private Key</li> <li>Read (and transmit) bsnSslWebadminCert Public Key</li> <li>Establish TLS Pre-Master Key</li> <li>Establish and then perform cryptographic operations with TLS Encryption Key and TLS Integrity Key</li> </ul>

**Table 4**      **Key/CSP Access by Service**

Service	Key Access
Local EAP Authenticator (EAP-TLS)	<ul style="list-style-type: none"> <li>• Sign with VendorDeviceCert Private Key</li> <li>• Read (and transmit) VendorCACert</li> <li>• Establish EAP-TLS tunnel Pre-master secret</li> <li>• Derives EAP-TLS Master secret and tunnel encryption &amp; integrity keys</li> <li>• Derives EAP-TLS peer encryption Key</li> </ul>
Local EAP Authenticator (EAP-FAST)	<p>In-band PAC Provisioning without certificates:</p> <ul style="list-style-type: none"> <li>• Establish EAP-TLS pre-master secret using anonymous Diffie Hellman key exchange</li> <li>• Derive EAP-TLS master secret and EAP-TLS tunnel encryption and integrity keys</li> <li>• Read EAP-FAST Server Key and generate EAP-FAST PAC-Key for the client</li> </ul> <p>In-band PAC Provisioning with certificates:</p> <ul style="list-style-type: none"> <li>• Sign with VendorDeviceCert Private Key</li> <li>• Read (and transmit) VendorCACert</li> <li>• Read and verify Client certificate</li> <li>• Establish EAP-TLS pre-master secret using authenticated Diffie Hellman key exchange</li> <li>• Derive EAP-TLS master secret and EAP-TLS tunnel encryption and integrity keys</li> <li>• Read EAP-FAST Server Key and generate EAP-FAST PAC-Key for the client</li> </ul> <p>EAP-FAST Tunnel Establishment:</p> <ul style="list-style-type: none"> <li>• Read EAP-Fast Server Key</li> <li>• Decrypt client PAC to recover client EAP-FAST PAC-Key</li> <li>• Derive EAP-FAST Master secret and tunnel encryption/integrity keys and EAP-FAST Session-Key Seed.</li> </ul> <p>Authentication:</p> <ul style="list-style-type: none"> <li>• Derive EAP-FAST Inner Method Session Key according to the inner EAP algorithm</li> <li>• Derive EAP-FAST Master Session Key using the Session-Key Seed and Inner Method Session Key(s).</li> </ul>



**Table 4**      *Key/CSP Access by Service*

Service	Key Access
802.11i	<ul style="list-style-type: none"> <li>• Compute 802.11i KCK, 802.11i KEK and 802.11i PTK from 802.11i PMK or 802.11i PSK</li> <li>• Generate 802.11i GTK</li> <li>• Encrypt/decrypt using 802.11i KEK</li> <li>• Authenticate data using 802.11i KCK</li> </ul>
RADIUS	<ul style="list-style-type: none"> <li>• Decrypt 802.11i PMK using KeyWrap KEK</li> <li>• Authenticate data using KeyWrap MACK</li> </ul>

## Key Zeroization

All keys in the module may be zeroized by entering this CLI command:

```
> config switchconfig key-zeroize controller
```

After this step, power cycle the module and hold down the escape key to initiate a memory test that will clear any residual keys from the RAM.

## Disallowed Security Functions

These cryptographic algorithms are not approved and may not be used in FIPS mode of operations:

- RC4
- MD5
- HMAC MD5
- AES-CTR
- CCKM

## Self Tests

The following self tests are performed by the module:

- Firmware integrity test
- Power on self test of AES-ECB, AES-CCM, SHA-1, HMAC SHA-1, RNG, TDES, EAP-FAST KDF, and RSA algorithms
- Continuous random number generator test for Approved and non-Approved RNGs

Self Tests are performed automatically when power is applied to the module. Self Tests may be run on-demand at any time by cycling power to the module.

## Mitigation of Attacks

The module provides mitigation against the following attacks:

- Protection against wireless denial of service attacks due to forged 802.11 management frames. When wireless clients and wireless infrastructure are enabled with MFP (Management Frame Protection) the system is protected against DoS attacks from exploited 802.11 management frames.
- Protection against rogue or unauthorized APs in joining the trusted network. The Cisco APs and Controllers support mutual authentication via x.509 certificates that are installed from the factory.
- Protection against MiTM attacks against AP control traffic. All control and bridging traffic between Controllers and APs is protected with AES-CCM encryption.

## Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New* in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at this URL:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

CCDE, CCENT, CCSI, Cisco Eos, Cisco HealthPresence, Cisco IronPort, the Cisco logo, Cisco Lumin, Cisco Nexus, Cisco Nurse Connect, Cisco Pulse, Cisco StackPower, Cisco StadiumVision, Cisco TelePresence, Cisco Unified Computing System, Cisco WebEx, DCE, Flip Channels, Flip for Good, Flip Mino, Flipshare (Design), Flip Ultra, Flip Video, Flip Video (Design), Instant Broadband, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn, Cisco Capital, Cisco Capital (Design), Cisco:Financed (Stylized), Cisco Store, and Flip Gift Card are service marks; and Access Registrar, Aironet, AllTouch, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, Continuum, EtherFast, EtherSwitch, Event Center, Explorer, Fast Step, Follow Me Browsing, FormShare, GainMaker, GigaDrive, HomeLink, iLYNX, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, Laser Link, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerKEY, PowerPanels, PowerTV, PowerTV (Design), PowerVu, Prisma, ProConnect, ROSA, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0908R)

© 2009 Cisco Systems, Inc. All rights reserved.