# Juniper Networks, Inc.

# Juniper Express 4 MACsec Cryptographic Module

FIPS 140-3 Non-Proprietary Security Policy

Document Version: 1.2

Last update: 2026-02-26

**Prepared by:**

atsec information security corporation
4516 Seton Center Pkwy, Suite 250
Austin, TX 78759
www.atsec.com

**Prepared for:**

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
www.juniper.net

# Table of Contents

# List of Tables



# List of Figures

# 1 General

## 1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for the Juniper Express 4 MACsec Cryptographic Module. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for a Security Level 1 software-hybrid module.

This Security Policy has a one-to-one mapping to SP 800-140B starting with Section B.2.1 named "General" that maps to section 1 in this document and ending with section B.2.12 named "Mitigation of other attacks" that maps to section 12 in this document.



## 1.2 Security Levels

The following sections describe the cryptographic module and how it conforms to the FIPS 140-3 specification in each of the required areas.

| Section | Title | Security Level |
|---|---|---|
| 1 | General | 1 |
| 2 | Cryptographic module specification | 1 |
| 3 | Cryptographic module interfaces | 1 |
| 4 | Roles, services, and authentication | 1 |
| 5 | Software/Firmware security | 1 |
| 6 | Operational environment | 1 |
| 7 | Physical security | 1 |
| 8 | Non-invasive security | N/A |
| 9 | Sensitive security parameter management | 1 |
| 10 | Self-tests | 1 |
| 11 | Life-cycle assurance | 1 |
| 12 | Mitigation of other attacks | N/A |
| | Overall Level | 1 |

Table 1: Security Levels

## 1.3 Additional Information

The vendor has provided the non-proprietary Security Policy of the cryptographic module, which was further consolidated into this document by atsec information security together with other vendor-supplied documentation. In preparing the Security Policy document, the laboratory formatted the vendor-supplied documentation for consolidation without altering the technical statements therein contained. The further

refining of the Security Policy document was conducted iteratively throughout the conformance testing, wherein the Security Policy was submitted to the vendor, who would then edit, modify, and add technical contents. The vendor would also supply additional documentation, which the laboratory formatted into the existing Security Policy, and resubmitted to the vendor for their final editing.

# 2 Cryptographic Module Specification

## 2.1 Description

**Purpose and Use:**

The Juniper Express 4 MACsec Cryptographic Module (hereafter referred to as "the module") is a software-hybrid module.

The module is composed by the MACsec blocks in hardware (contained in the Juniper Networks® Express 4 processor), which provides the AES GCM and XPN algorithm implementations for encrypting and decrypting MACsec traffic, and a device driver in software, which provides the functionality to comply with FIPS 140-3 requirements (i.e. integrity test, self-tests), as well as the API to configure the hardware component.

The module is also bound to the following cryptographic modules:

- Junos® OS Evolved Kernel Cryptographic Module Version 2.0 (validated under FIPS certificate #5399), which provides the integrity check utility that is invoked by the module to check the integrity of the module's software component.
- Junos® OS Evolved OpenSSL Cryptographic Module Version 3.0.8 (validated under FIPS certificate #5400) to provide the algorithm implementation for the integrity test.

Sections of this Security Policy which refer to information from the bound module, also known as the Existing Validated Module (or EVM) are marked by [EVM] as per IG 1.A Resolution 5.

**Module Type**: : Software-hybrid

**Module Embodiment**: Multi-Chip Standalone

**Cryptographic Boundary:**

The diagram below shows the components that comprise the cryptographic module (in yellow), its cryptographic boundary (enclosed by dotted blue boxes), and the interfaces with the operational environment.

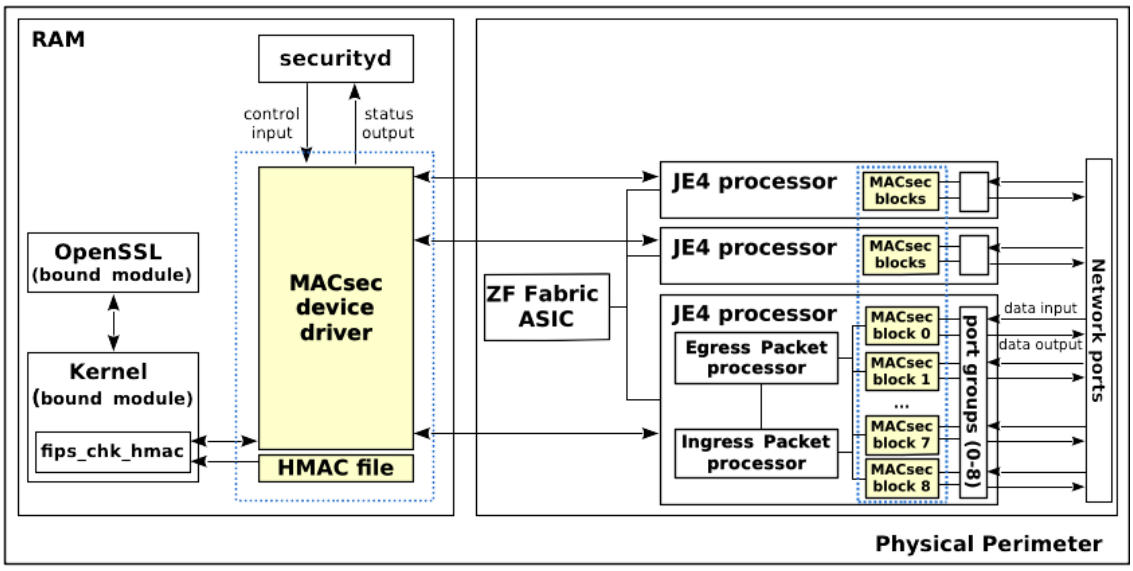


Figure 1: Block Diagram

**Tested Operational Environment's Physical Perimeter (TOEPP):**

The tested operating environment's physical perimeter is the general-purpose computer on which the module is running.

## 2.2 Tested and Vendor Affirmed Module Version and Identification

**Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):**

| Package or File Name | Software/ Firmware Version | Features | Integrity Test |
|---|---|---|---|
| /usr/lib64/libmacsecv2.so and /usr/lib64/.libmacsecv2.so.hmac on Juniper Networks® Packet Transport Router Model PTX10001-36MR | 1.0 | N/A | HMAC-SHA2-256 |

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

**Tested Module Identification – Hybrid Disjoint Hardware:**

| Model and/or Part Number | Hardware Version | Firmware Version | Processors | Features |
|---|---|---|---|---|
| Juniper Express 4 MACsec Blocks | JTAG ID 20611361 | N/A | Juniper Express 4 | N/A |

Table 3: Tested Module Identification – Hybrid Disjoint Hardware

**Tested Operational Environments - Software, Firmware, Hybrid:**

| Operating System | Hardware Platform | Processors | PAA/PAI | Hypervisor or Host OS | Version(s) |
|---|---|---|---|---|---|
| Junos® OS Evolved version 22.4 | Juniper Networks® Packet Transport Router Model PTX10001-36MR | Intel® Xeon® D-2163IT | No | N/A | 1.0 |

Table 4: Tested Operational Environments - Software, Firmware, Hybrid

## 2.3 Excluded Components

The module does not have any excluded components.

## 2.4 Modes of Operation

**Modes List and Description:**

The module supports only the approved mode of operation. When the module starts up successfully, after passing all the pre-operational self-tests and conditional cryptographic algorithm self-tests (CASTs), the module is operating in the approved mode of operation.

| Mode Name | Description | Type | Status Indicator |
|---|---|---|---|
| Approved mode | Automatically entered whenever an approved service is requested | Approved | Service API returns 1 |

Table 5: Modes List and Description

## 2.5 Algorithms

**Approved Algorithms:**

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| AES-GCM | A4089 | Direction - Decrypt, Encrypt<br>IV Generation - External<br>IV Generation Mode - 8.2.1<br>Key Length - 128, 256<br>Tag Length - 128<br>IV Length - IV Length: 96<br>Payload Length - Payload Length: 128, 256, 120, 248<br>AAD Length - AAD Length: 128, 256, 120, 0 | SP 800-38D |
| AES-XPN | A4089 | Direction - Decrypt, Encrypt<br>Key Length - 128, 256<br>Payload Length - Payload Length: 128, 256, 120, 248 | SP 800-38D |

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| | | AAD Length - AAD Length: 128, 256, 120, 0<br>Tag Length - 128<br>IV Generation - External<br>IV Generation Mode - 8.2.1<br>Salt Generation - External | |

Table 6: Approved Algorithms -

[EVM]

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| HMAC-SHA2-256 | A4246 | MAC - MAC: 256<br>Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |

Table 7: Approved Algorithms - [EVM]

## 2.6 Security Function Implementations

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| Authenticated symmetric encryption | BC-Auth | Authenticated symmetric encryption | | AES-GCM: (A4089)<br>AES-XPN: (A4089) |
| Authenticated symmetric decryption | BC-Auth | Authenticated symmetric decryption | | AES-GCM: (A4089)<br>AES-XPN: (A4089) |
| Integrity test | MAC | [EVM] Integrity test | Provided by:Bound OpenSSL and Kernel modules | HMAC-SHA2-256: (A4246) |

Table 8: Security Function Implementations

## 2.7 Algorithm Specific Information

AES GCM is used within the context of the MACsec protocol.  Please see Section 2.11.1 for details.

## 2.8 RBG and Entropy

This module does not support random bit generation and does not contain an entropy source.

## 2.9 Key Generation

The module does not provide key generation mechanisms.

## 2.10 Key Establishment

The module does not provide key establishment mechanisms.

## 2.11 Industry Protocols

### 2.11.1 AES GCM IV

The AES GCM IV generation is compliant with IEEE 802.1AE and shall only be used for the MACsec protocol to be compliant with FIPS 140-3 IG IG C.H, provision 1.c (“MACsec protocol IV generation”).

The module is part of the Juniper Packet Transport Router Model PTX10001-36MR, which supports MACsec using static connectivity association key (CAK) security mode. In this mode, a pre-shared key (PSK) is exchanged between the devices on each end of the point-to-point Ethernet link. Each appliance plays the role of either the Peer or the Authenticator in the context of the MACsec protocol. No authentication server is involved.

When supporting the MACsec protocol in the approved mode, the module should only be used together with the same appliance or other appliances that are also FIPS 140-3 validated and operating in the approved mode. In addition, the link between the Peer and the Authenticator should be secured to prevent the possibility for an attacker to introduce foreign equipment into the local area network.

In line with the MACsec protocol, the IV has a length of 96 bits and it is constructed externally by concatenating:

- For AES-128-GCM and AES-256-GCM: the 64-bit Secure Channel Identifier (SCI) and the 32-bit packet number (PN)
- For AES-128-XPN and AES-256-XPN: the 32-bit Short Secure Channel Identifier (SSCI), the 64-bit extended packet number (XPN) and a 96-bit salt.

In case the module's power is lost and then restored, the key used for the AES GCM encryption or decryption shall be redistributed.

# 3 Cryptographic Module Interfaces

## 3.1 Ports and Interfaces

Ports and interfaces implemented are shown in the following table. The Control Output interface is omitted because the module does not implement it.

Network ports and the power supply port correspond to the Juniper Express 4 processor where the MACsec blocks are contained.

All data output via data output interface is inhibited when the module is performing self-tests or zeroization, or when the module is in the error state.

| Physical Port | Logical Interface(s) | Data That Passes |
|---|---|---|
| N/A | Data Input | Decrypted/encrypted data received from network ports |
| N/A | Data Output | Encrypted/decrypted data sent to network ports |
| N/A | Control Input | API function calls, API input parameters for control input. |
| N/A | Status Output | API return codes, API output parameters for status output. |
| N/A | Power | N/A |

Table 9: Ports and Interfaces

# 4 Roles, Services, and Authentication

## 4.1 Authentication Methods

The module does not support authentication.

## 4.2 Roles

The module supports the Crypto Officer role only. This sole role is implicitly assumed by the operator of the module when performing a service. The module does not support concurrent operators.

| Name | Type | Operator Type | Authentication Methods |
|---|---|---|---|
| Crypto Officer | Role | Crypto Officer | None |

Table 10: Roles

## 4.3 Approved Services

The module provides API functions to access the registers of the MACsec blocks used to perform encryption (macsecv2_drv_rx_reg_port_config_rd) and decryption (macsecv2_drv_rx_reg_port_config_rd) services. The value 1 in the register position corresponding to the MACsec block determines the service indicator.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| Authenticated symmetric encryption | Perform AES encryption | macsecv2_drv_tx_reg_port_config_rd() returns 1 in bit corresponding to the MACsec block. | Plaintext, IV, AES key | Ciphertext | Authenticated symmetric encryption | Crypto Officer - AES key: W,E |
| Authenticated symmetric decryption | Perform AES decryption | macsecv2_drv_rx_reg_ port_config_rd() returns 1 in bit corresponding to the MACsec block. | Ciphertext, IV, AES key | Plaintext | Authenticated symmetric decryption | Crypto Officer - AES key: W,E |
| Show status | Return the module status | None | None | Module status | None | Crypto Officer |

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| Self-test | Perform the CASTs and the integrity test (integrity test provided by bound OpenSSL and Kernel modules) | None | None | Pass/fail | Authenticated symmetric encryption Authenticated symmetric decryption Integrity test | Crypto Officer |
| Zeroization | Zeroize all SSPs | None | Port group ID | Pass/fail | None | Crypto Officer - AES key: Z |
| Show module name and version | Return module name and version information | None | None | Module name/version | None | Crypto Officer |

Table 11: Approved Services

## 4.4 Non-Approved Services

The module does not implement any non-approved services.

## 4.5 External Software/Firmware Loaded

The module does not support the loading of external software/firmware.

# 5 Software/Firmware Security

## 5.1 Integrity Techniques

The integrity of the module is ensured with the HMAC-SHA2-256 value stored in the corresponding /usr/lib64/.libmacsecv2.so.hmac file that is computed at build time. During Pre-Operational Self-Tests, the module invokes the fips_chk_hmac utility provided by the bound Kernel module (relying on the HMAC service provided by the bound OpenSSL module) to calculate the HMAC value of the shared library, and then compares it with the pre-stored one. If the two HMAC values do not match, the test fails and the module enters the error state.

The integrity of the fips_chk_hmac utility itself is performed before the integrity tests of the module, and ensured with the HMAC-SHA2-256 value stored in the corresponding .hmac file that is computed at build time of the utility. The fips_chk_hmac utility calculates the HMAC value, and then compares it with the prestored value. If the two HMAC values do not match, the test fails and the module enters the error state.

The HMAC key is stored within the fips_chk_hmac utility binary.

## 5.2 Initiate on Demand

Integrity tests are performed as part of the pre-operational self-tests, which are executed when the module is initialized. The integrity tests can be invoked on demand by unloading and subsequently re-initializing the module, which will perform (among others) the software integrity tests.

# 6 Operational Environment

## 6.1 Operational Environment Type and Requirements

The module operates in a modifiable operational environment per FIPS 140-3 level 1 specifications. The module runs on a commercially available general-purpose operating system executing on the hardware specified in Table 2.

**Type of Operational Environment**: Modifiable

**How Requirements are Satisfied**:

The operating system provides process isolation and memory protection mechanisms that ensure appropriate separation for memory access among the processes on the system. Each process has control over its own data and uncontrolled access to the data of other processes is prevented.

## 6.2 Configuration Settings and Restrictions

The module shall be installed as stated in Section 11. If properly installed, the operating system provides process isolation and memory protection mechanisms that ensure appropriate separation for memory access among the processes on the system. Each process has control over its own data and uncontrolled access to the data of other processes is prevented.

Instrumentation tools like the ptrace system call, gdb and strace utilities, userspace live patching, as well as other tracing mechanisms offered by the Linux environment such as ftrace or systemtap, shall not be used in the operational environment. The use of any of these tools implies that the cryptographic module is running in a non-tested operational environment.

# 7 Physical Security

The Juniper Express 4 processor (an ASIC silicon) is the hardware that contains the MACsec blocks that constitute the hardware component of the module. The embodiment of the chip is a single chip consisting of production-grade components. The coating is a standard sealing coat applied over the single chip. The module provides no additional physical security techniques.

Figure 2 - Juniper Express 4 processor

The module inherits the physical characteristics of the host running it. Figure 3 illustrates the Juniper Networks® Packet Transport Router Model PTX10001-36MR that represents the testing platform and includes the hardware component of the cryptographic module. The router includes three Juniper Express 4 ASIC chips to provide forwarding traffic and supporting MACsec for its 36 multi-rate ports.


Figure 3 - Packet Transport Router Model PTX10001-36MR

# 8 Non-Invasive Security

This module does not implement any non-invasive security mechanism and therefore this section is not applicable.

# 9 Sensitive Security Parameters Management

## 9.1 Storage Areas

| Storage Area Name | Description | Persistence Type |
|---|---|---|
| RAM | Temporary storage for SSPs used by the module as part of service execution. The module does not perform persistent storage of SSPs | Dynamic |

Table 12: Storage Areas

The module does not perform persistent storage of SSPs. The SSPs are temporarily stored in the RAM in plaintext form. SSPs are provided to the module by the calling process and are destroyed when released by the appropriate zeroization function calls.

## 9.2 SSP Input-Output Methods

| Name | From | To | Format Type | Distribution Type | Entry Type | SFI or Algorithm |
|---|---|---|---|---|---|---|
| API input parameters | Operator calling application (TOEPP) | Cryptographic module | Plaintext | Manual | Electronic | |

Table 13: SSP Input-Output Methods

The module only supports SSP entry from the calling application running on the same operational environment. This corresponds to manual distribution, electronic entry/output ("CM Software to/from App via TOEPP Path") per FIPS 140-3 IG 9.5.A Table 1. There is no entry of cryptographically protected SSPs.

The module does not output any SSPs.

## 9.3 SSP Zeroization Methods

| Zeroization Method | Description | Rationale | Operator Initiation |
|---|---|---|---|
| Wipe and Free memory block allocated | Zeroizes the SSPs contained within the cipher handle. | Memory occupied by SSPs is overwritten with zeroes and then it is released, which renders the SSP values irretrievable. The completion of the zeroization routine indicates that the zeroization procedure succeeded. | By calling the cipher related zeroization API function: macsecv2_drv_port_tx_sa_delete() for AES keys |

| Zeroization Method | Description | Rationale | Operator Initiation |
|---|---|---|---|
| Automatic | Automatically zeroized by the module when no longer needed | Memory occupied by SSPs is overwritten with zeroes, which renders the SSP values irretrievable. | N/A |
| Module Reset | De-allocates the volatile memory used to store SSPs | Volatile memory used by the module is overwritten within nanoseconds when power is removed. | By unloading and reloading the module |

Table 14: SSP Zeroization Methods

The module also zeroizes the registers when the MACsec block is no longer used when the macsecv2_drv_port_tx_sa_delete() function is invoked. The completion of this function will indicate that the zeroization of the SSPs included in the MACsec block finished successfully.

All data output is inhibited during zeroization.

## 9.4 SSPs

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| AES key | AES key used for authenticated symmetric encryption/decryption | 128, 256 bits - 128, 256 bits | Symmetric key - CSP | | | Authenticated symmetric encryption<br>Authenticated symmetric decryption |

Table 15: SSP Table 1

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| AES key | API input parameters | RAM:Plaintext | | Wipe and Free memory block allocated<br>Module Reset | |

Table 16: SSP Table 2

# 10 Self-Tests

The module performs the pre-operational and conditional cryptographic algorithms self-tests automatically when the module is loaded into memory. These self-tests ensure that the module is not corrupted and that the cryptographic algorithms work as expected. While the module is executing the pre-operational and the conditional cryptographic algorithms self-tests, services are not available, and input and output are inhibited. The module is not available for use by the calling application until the self-tests are completed successfully. If any of the self-tests fails, an error message is returned and the module transitions to error state.

## 10.1 Pre-Operational Self-Tests

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details |
|---|---|---|---|---|---|
| HMAC-SHA2-256 | HMAC-SHA2-256 | MAC tag verification | SW/FW Integrity | Module becomes operational | [EVM] Integrity test for module |

Table 17: Pre-Operational Self-Tests

The module performs a pre-operational software integrity test automatically when the module is powered on before the module transitions into the operational state. The details of the integrity test are specified in Section 5.1.

## 10.2 Conditional Self-Tests

Table 18 lists the cryptographic algorithm self-tests (CASTs). The CASTs include the KATs for the integrity mechanism that is run prior to performing the integrity test. The details of the integrity test are provided in Section 5.1.

Each KAT includes comparison of the calculated output with the expected known answer, hard coded as part of the test vectors used in the test. Data output through the data output interface is inhibited during the self-tests. If the values do not match, the KAT fails and the module transitions to the error state.

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| AES-GCM Encrypt | 128, 256-bit keys, encrypt | KAT | CAST | Module becomes operational | Symmetric operation | Test runs at power-on before module becomes operational |
| AES-GCM Decrypt | 128, 256-bit keys, decrypt | KAT | CAST | Module becomes operational | Symmetric operation | Test runs at power-on before module becomes operational |

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| AES-XPN Encrypt | 128, 256-bit keys, encrypt | KAT | CAST | Module becomes operational | Symmetric operation | Test runs at power-on before module becomes operational |
| AES-XPN Decrypt | 128, 256-bit keys, decrypt | KAT | CAST | Module becomes operational | Symmetric operation | Test runs at power-on before module becomes operational |
| HMAC-SHA2-256 (A4246) | SHA2-256 | KAT | CAST | Module becomes operational | [EVM] Message authentication | Test runs at power-on before the integrity test |

Table 18: Conditional Self-Tests

KATs for the HMAC algorithm used in this module are performed by the respective bound modules.

## 10.3 Periodic Self-Test Information

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| HMAC-SHA2-256 | MAC tag verification | SW/FW Integrity | On Demand | Manually |

Table 19: Pre-Operational Periodic Information

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| AES-GCM Encrypt | KAT | CAST | On Demand | Manually |
| AES-GCM Decrypt | KAT | CAST | On Demand | Manually |
| AES-XPN Encrypt | KAT | CAST | On Demand | Manually |
| AES-XPN Decrypt | KAT | CAST | On Demand | Manually |
| HMAC-SHA2-256 (A4246) | KAT | CAST | On Demand | Manually |

Table 20: Conditional Periodic Information

On-demand self-tests can be invoked by powering-off and reloading the module which cause the module to run the pre-operational and conditional cryptographic algorithms self-tests.

## 10.4 Error States

When the module fails any pre-operational self-test, the module will enter the Error state. Any further cryptographic operation is inhibited. The calling application can obtain the module state by calling the

macsecv2_drv_fips_kat_ok() and the macsecv2_drv_integrity_chk_ok() API functions. The function will return a boolean code indicating whether the CAST or the integrity tests passed or failed.
The Crypto Officer can recover from the Error state by restarting the hardware platform on which the module is running.

| Name | Description | Conditions | Recovery Method | Indicator |
|---|---|---|---|---|
| Error | General-purpose error state | Failure of CAST<br>Failure of integrity tests | Power cycle | macsecv2_drv_fips_kat_ok() returns false (CAST failure), macsecv2_drv_integrity_chk_ok() returns false (integrity test failure) |

Table 21: Error States

## 10.5 Operator Initiation of Self-Tests

The operator can initiate the pre-operational self-tests and the cryptographic algorithms self-tests by powering off and subsequently reloading the module.

# 11 Life-Cycle Assurance

## 11.1 Installation, Initialization, and Startup Procedures

The binary of the module is contained in the base Junos Evolved installation image. The Crypto Officer shall follow this Security Policy to configure the operational environment and install the module to be operated as a FIPS 140-3 validated module.

## 11.2 Administrator Guidance

The module is already pre-installed on the image file (junos-evo-install-ptx-fixed-x86-64-22.4R2.11-S1-EVO.iso). The crypto officer is responsible to verify the correct installation of the module by executing the following command:

```
show macsec drv version
```

Verify that the command returns the following name and version of the software and hardware components of the module:

```
Junos OS Evolved MACsec Cryptographic Driver Library, version 1.0 : ASIC JTAG ID 20611361
```

The Junos OS Evolved OpenSSL Cryptographic Module version 3.0.8 and the Junos OS Evolved Kernel Cryptographic Module are bound modules that shall also be installed and configured as described in section 11.1 of their corresponding Security Policies. The administrator shall follow the steps to install the modules and verify their version.

## 11.3 End of Life

As the module does not persistently store SSPs, secure sanitization of the module consists of unloading the module. This will zeroize all SSPs in volatile memory.

# 12 Mitigation of Other Attacks

The module does not implement any additional mitigation mechanism.

# Appendix A. Glossary and Abbreviations

| | |
|---|---|
| **AES** | Advanced Encryption Standard |
| **API** | Application Program Interface |
| **CAST** | Cryptographic Algorithm Self-Test |
| **CAVP** | Cryptographic Algorithm Validation Program |
| **CBC** | Cipher Block Chaining |
| **CMVP** | Cryptographic Module Validation Program |
| **CSP** | Critical Security Parameter |
| **ECB** | Electronic Code Book |
| **EE** | Electronic Entry |
| **FIPS** | Federal Information Processing Standards Publication |
| **GCM** | Galois Counter Mode |
| **HMAC** | Hash Message Authentication Code |
| **IG** | Implementation Guidance |
| **KAT** | Known Answer Test |
| **MD** | Manual Distribution |
| **NIST** | National Institute of Science and Technology |
| **PAA** | Processor Algorithm Acceleration |
| **PCT** | Pair-wise Consistency Test |
| **SHA** | Secure Hash Algorithm |
| **SCI** | Secure Channel Identifier |
| **SSCI** | Short Secure Channel Identifier |
| **SSP** | Sensitive Security Parameter |
| **XPN** | Extended Packet Numbering |

# Appendix B. References


| | |
|---|---|
| FIPS 140-3 | **FIPS PUB 140-3 - Security Requirements For Cryptographic Modules**<br>March 2019<br>https://doi.org/10.6028/NIST.FIPS.140-3 |
| FIPS 140-3 IG | **Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program**<br>October 2022<br>https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-ig-announcements |
| SP 800-38D | **Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC**<br>November 2007<br>https://doi.org/10.6028/NIST.SP.800-38D |
| IEEE 802.1AE | **MAC Security (MACsec)**<br>December 2018<br>https://1.ieee802.org/security/802-1ae/ |