Phio TX

FIPS 140-3 Non-Proprietary Security Policy

Document Version 1.7

October 16, 2024

Prepared for:

Quantum Xchange
7700 Old Georgetown Road, Suite 850
Bethesda, MD 20814
info@quantumxc.com

Prepared by:

KeyPair Consulting Inc.
987 Osos Street
San Luis Obispo, CA 93401
keypair.us
+1 805.316.5024

Table of Contents

List of Tables

List of Figures

1 General

This document defines the Security Policy for the Quantum Xchange Phio TX, hereafter denoted the Module.

The Module is a of type hardware, with a multi-chip standalone embodiment and is validated to FIPS 140-3 overall Level 2 requirements with security levels as follows:

Table 1: Security Levels

ISO/IEC 24759 Section 6. [Number Below]	FIPS 140-3 Section Title	Security Level
1	General	2
2	Cryptographic Module Specification	2
3	Cryptographic Module Interfaces	2
4	Roles, Services, and Authentication	2
5	Software/Firmware Security	2
6	Operational Environment	N/A
7	Physical Security	2
8	Non-Invasive Security	N/A
9	Sensitive Security Parameters Management	2
10	Self-tests	2
11	Life-cycle Assurance	2
12	Mitigation of Other Attacks	N/A

2 Cryptographic Module Specification

The Phio TX has a multi-chip standalone embodiment in [FIPS140-3] terminology. The Phio TX provides network and secure communications functionality to facilitate propagation of keys between Phio TX nodes and endpoint clients in a Phio Hive (a dynamic peer network).

The tested configurations are specified in Table 2 below.

Table 2: Cryptographic Module Tested Configuration

Model	Hardware [Part Number and Version]	Firmware Version	Distinguishing Features
Phio TX	F109158	3.2.3	Without QRNG hardware option.
Phio TX-Q	F109158-Q	3.2.3	With QRNG hardware option.

The optional QRNG is a quantum entropy source, with output used as additional entropy input that is not credited in the DRBG seeding strength rationale. As the QRNG is an optional feature, the Phio TX does not rely on it for assurance of key generation security strength in the [FIPS140-3] process. See the SSP section below for additional information.

2.1 Cryptographic Boundary

The physical form of the Module is depicted below. The cryptographic boundary is the metal chassis shown in Figure 1 and Figure 2; as such the Phio TX is validated as a hardware module in [FIPS140-3] terms. The figures represent both hardware versions listed in Table 2.

Figure 1: Module Cryptographic Boundary – Top/Front/Left Side View

Figure 2: Module Cryptographic Boundary – Bottom/Back/Right Side View

Figure 3: Front Panel

Figure 4: Rear Panel

The Module logical functionality (outlined in red) is shown below. All Module firmware is contained within the boundary.

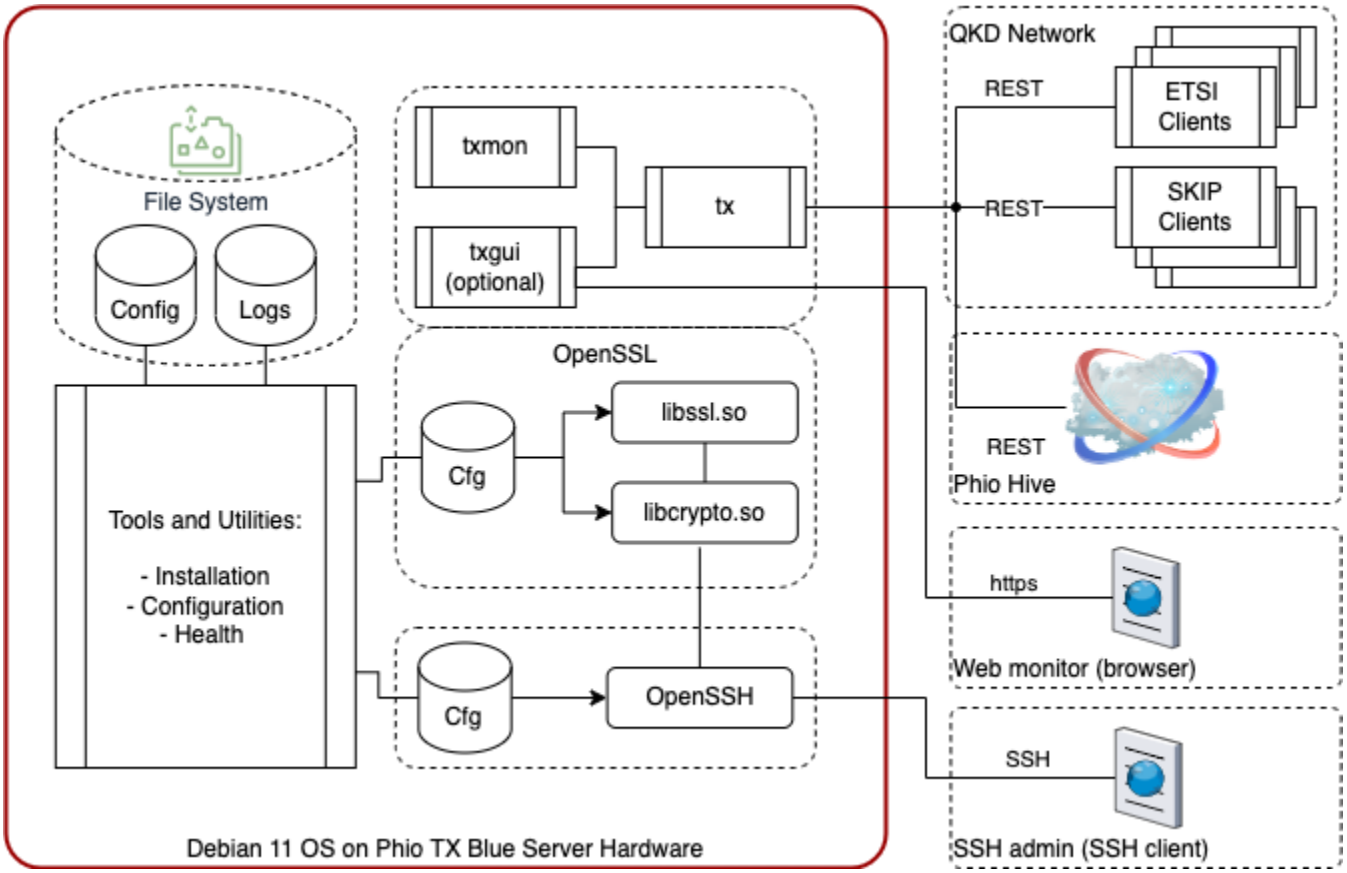

Figure 5: Module High Level Block Diagram

The following hardware/firmware components are excluded from the cryptographic boundary and thus the FIPS 140-3 requirements:

- TPM (use is optional): used to measure and track Module configuration.
- Whole disk encryption (use is optional): not security relevant, similar to [FIPS 140-3_IG] 2.4.A Scenario 2.
- PQC (use is optional): used for redundant communications channel obfuscation.
- QRNG (hardware option): used for redundant entropy input.
- Password hash: not security relevant, similar to [FIPS 140-3_IG] 2.4.A Scenario 1.

None of the excluded components listed above are necessary to meet FIPS 140-3 requirements.

2.2 Modes of Operation, Security Rules and Guidance

Configuration of the Module for conformance to this Security Policy requires installation of a valid Phio TX License with locked Approved mode enforcement, prior to doing so the Module is in a non-complaint state. Once configured for Approved operation, the Module does not require operator actions to operate in the approved mode and provides only Approved services and enforces the security rules listed next. The Module does not support a non-Approved or a degraded mode of operation.

1. No additional interface or service is implemented by the Module which would provide access to CSPs.
2. Data output is inhibited during key generation, self-tests, zeroisation, and error states.
3. All CSPs are zeroised by the zeroisation service (*tx_reset* command) with the exception of the FWIK, which can be destroyed if required by the secure sanitization process.
4. The Module does not support manual key entry.

5. The Module does not output plaintext CSPs or intermediate key values.
6. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the Module.
7. The Module can use only algorithms that have passed self-tests.
8. The Module prohibits changing to the Crypto Officer state from any other role other than the Crypto Officer.
9. The Module does not support multiple concurrent operators (over the serial console), a maintenance role, or a bypass capability.

The Module design corresponds to the Module security rules. Initialization and installation requirements of the Module have been specified in Section 11 of this document.

AES GCM is used to support TLS and SSH secure communications and adheres to the [FIPS140-3_IG] C.H Resolution 1a TLS 1.2 and 1d SSH protocol IV generation requirements.

AES-GCM IVs shall be used in compliance with [FIPS140-3_IG] C.H scenario 1a (TLS/DTLS 1.2, per [RFC5288]) and 1d (SSHv2, per [RFC5647]). The Module is compatible with TLS/DTLS 1.2 protocol and provides the primitives to support the AES GCM ciphersuites from [SP800-52r1] Section 3.3.1. The Module's implementation of AES-GCM is used together with one or more applications outside the Module's cryptographic boundary that implement the specified protocols; Per [FIPS140-3_IG] D.C, no parts of these protocols, other than the approved cryptographic algorithms and KDF, have been reviewed or tested by the CAVP and CMVP.

In each of the protocols, if the Module's power is lost and then restored, the key used for the AES GCM encryption/decryption shall be re-distributed. This condition is not enforced by the Module but is met implicitly. The Module does not retain any state across reset or power-cycles: AES-GCM key/IVs are not stored in non-volatile persistent memory (i.e., disk), hence no re-connection can occur without a fresh key establishment operation and the associated SSPs.

The Module explicitly ensures that the counter (the nonce_explicit part of the IV) does not exhaust the maximum number of possible values of 2^{64}-1 for a given session key. Connections are monitored, with a maximum lifetime of two hours or a maximum transaction count that is established to be less than the number of transactions prior to IV "rollover".

2.3 Approved and Allowed Cryptographic Functionality

The Module implements the Approved cryptographic functions listed in Table 3. Equivalent strength in bits is given for each key or algorithm type (as some algorithms do not use or produce keys). The term *s* is used throughout to indicate security strength, following the notation used in the majority of the sources (refer to the notes below Table 3). This table is referenced by Table 9 (SSPs). All references to the algorithm standards cited throughout this document can be found in the References section.

Table 3: Approved Algorithms

CAVP Cert.[1]	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2983	AES-CBC [FIPS197], [SP800-38A]	AES-CBC.	AES-128 (s = 128), AES-192 (s = 192), AES-256 (s = 256).	Encryption and decryption. CBC: TLS, SSH ciphers.
A2983	AES-ECB [FIPS197], [SP800-38A]	AES-ECB.	AES-128 (s = 128), AES-192 (s = 192), AES-256 (s = 256).	AES primitive used by CBC and CTR_DRBG.
A2983	AES-GCM [SP800-38D]	AES-GCM.	AES-128 (s = 128), AES-256 (s = 256).	Authenticated encryption and decryption. TLS, SSH cipher options.
Vendor Affirmed	CKG [SP800-133r2]	§4, §5.2, §6.2.1: Unmodified DRBG output.	N/A.	Asymmetric, symmetric key generation per [FIPS140-3_IG] D.H.
A2983	Counter DRBG [SP800-90Ar1]	AES CTR_DRBG: Instantiate, Generate, Reseed. DF: Uses block_cipher_df. Supports prediction resistance.	AES-256 (s = 256).	Random numbers used in key, IV and nonce generation.
A2983	DSA KeyGen [FIPS186-4]	FFC key generation.	L=2048, N=256 (s = 112); L=3072, N=256 (s = 128). See Note 4 and Note 6.	TLS DH/DHE key exchange.
A2983	ECDSA KeyGen [FIPS186-4]	ECC key generation.	P-256 (s ~= 128), P-384 (s ~= 192). See Note 2.	TLS, SSH key exchange and authentication.
A2983	ECDSA SigGen [FIPS186-4]	ECDSA signature generation (tested with SHA2-256, SHA2-384).	P-256 (s ~= 128), P-384 (s ~= 192). See Note 2.	TLS, SSH authentication. Certificate signing.
A2983	ECDSA SigVer [FIPS186-4]	ECDSA signature verification (tested with SHA2-256, SHA2-384).	P-256 (s ~= 128), P-384 (s ~= 192). See Note 2.	TLS, SSH authentication. Certificate verification.
N/A	ENT (NP) [SP800-90B]	Entropy source.	256 bits.	Used only to seed the approved DRBG.
A2983	HMAC-SHA-1 [FIPS198-1]	HMAC generation, verification with SHA-1.	SHA-1 (s = 160).	TLS message integrity.
A2983	HMAC-SHA2 [FIPS198-1]	HMAC generation, verification with the listed SHA2 modes.	SHA2-256 (s = 256), SHA2-384 (s = 384).	SSH, TLS message integrity. Firmware integrity.
A2983	KAS-ECC-SSC [SP800-56Ar3]	Scheme: Ephemeral Unified. Role: Initiator, Responder.	P-256 (s ~= 128), P-384 (s ~= 192). See Note 2 and Note 3.	TLS ECDH/ECDHE key exchange. SSH key exchange.
A2983	KAS-FFC-SSC [SP800-56Ar3]	Scheme: dhEphem. Role: Initiator, Responder. [FIPS140-3_IG] D.F Scenario 2, path 2) with TLS v1.2 KDF [RFC7627] and KDF SSH, per [FIPS140-3_IG] 2.4.B.	FB (s = 112), ffdhe2048 (s = 112), ffdhe3072 (112 ≤ s ≤ 128). See Note 5.	TLS DH/DHE key exchange.

[1] There are algorithms, modes, and key/moduli sizes that have been CAVP-tested but are not used by any approved service of the module. Only the algorithms, modes/methods, and key lengths/curves/moduli shown in this table are used by an approved service of the module.

CAVP Cert.[1]	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
CVL A2983	TLS v1.2 KDF [RFC7627]	TLS [RFC7627] key derivation with Extended Master Secret (EMS) support, using the listed hash algorithms.	SHA2-256 (s = 256), SHA2-384 (s = 384).	TLS key derivation.
CVL A2983	KDF SSH [SP800-135r1]	SSH v2 KDF using the listed hash algorithms.	SHA2-256 (s = 256), SHA2-384 (s = 384).	SSH key derivation.
A2983	RSA SigGen [FIPS186-4]	PSS signature generation (tested with the listed moduli and the following hash algorithms: SHA2-256, SHA2-384).	k=2048 (s ~= 112), k=3072 (s ~= 128). See Note 4 and Note 7.	SSH, TLS authentication. Certificate signing.
A2983	RSA SigVer [FIPS186-4]	PSS signature verification (tested with the listed moduli and the following hash algorithms: SHA2-256, SHA2-384).	K=2048 (s ~= 112), k=3072 (s ~= 128). See Note 4 and Note 7.	SSH, TLS authentication. Certificate verification.
A2983	SHA-1 [FIPS180-4]	N/A.	SHA-1 (s = 160).	Primitive used by HMAC.
A2983	SHA2 [FIPS180-4]	SHA2 modes listed at right.	SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512) See Note 1.	Message digest generation. Primitive used by HMAC, ECDSA, RSA and KDFs.
A2983	KTS-1	AES-CBC, AES-GCM, HMAC-SHA2, HMAC-SHA-1.	SP 800-38D and SP 800-38F. KTS (key wrapping) per IG D.G. 128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength.	TLS key establishment.
A2983	KTS-2	AES-CBC, AES-GCM, HMAC-SHA2.	SP 800-38D and SP 800-38F. KTS (key wrapping) per IG D.G. 128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength.	SSH key establishment.
A2983	KAS-1	Schemes: Ephemeral Unified. Roles: Initiator, Responder. KAS-ECC-SSC curves: P-256, P-384. TLS v1.2 KDF [RFC7627].	SP 800-56Arev3. KAS-ECC per IG D.F Scenario 2 path (2) option 2. P-256 and P-384 curves providing 128 or 192 bits of encryption strength.	TLS key agreement.
A2983	KAS-2	Schemes: Ephemeral Unified. Roles: Initiator, Responder. KAS-ECC-SSC curves: P-256, P-384. KDF SSH.	SP 800-56Arev3. KAS-ECC per IG D.F Scenario 2 path (2) option 2. P-256 and P-384 curves providing 128 or 192 bits of encryption strength.	SSH key agreement.
A2983	KAS-3	Schemes: dhEphem. Roles: Initiator, Responder. KAS-FFC-SSC safe prime groups: ffdhe2048. ffdhe3072. TLS v1.2 KDF [RFC7627].	SP 800-56Arev3. KAS-FFC per IG D.F Scenario 2 path (2) option 2.	TLS key agreement.

CAVP Cert.[1]	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
			2048-bit and 3072-bit keys providing 112 or 128 bits of encryption strength.	

Note 1: Preimage resistance strength applies to hash algorithms used in DRBG, KDFs. Described also in [SP800-57P1r5] Table 3.

Note 2: Elliptic curve strengths are annotated as approximate (i.e., s ~=) since [SP800-186] Table 1 provides approximate security strengths.

Note 3: Approved elliptic curves for ECC key agreement are given in [SP800-56Ar3] Table 24.

Note 4: In Digital Signature applications, security strength is primarily associated with the asymmetric key pair specification. The hash function used must have equivalent strength equal to or greater than the security strength of the associated key pair.

Note 5: Approved key types for FFC key agreement are given in [SP800-56Ar3] Tables 25, 26. The group notation of Table 26 is used for consistency with CAVP algorithm listings and ACVP capability registration.

Note 6: Security strength for L=2048/N=256 is determined in accordance with [FIPS140-3_IG] D.B Strength of SSP Establishment Methods as y = min(x, N/2), where x is 112 and therefore y = min(112, 128) = 112.

Note 7: Estimated security strengths of common RSA moduli are given in [SP800-56Br2] Table 4. IFC key types approved for Digital Signature Generation and Verification are given in [SP800-57P1r5] Table 2. Equivalent strengths are annotated as approximate (i.e., s ~=) since [SP800-56Br2] Table 4 provides approximate security strengths.

Reference sources for the strengths provided in Table 3 are as follows:

- AES (AES-128, AES-192, AES-256): [SP800-57P1r5] Table 2.
- ECC (P-256, P-384): [SP800-186] Table 1.
- FFC (L=2048/N=256, L=3072/N=256): [SP800-57P1r5] Table 2.
- FFC (FB, ffdhe2048, ffdhe3072): [SP800-56Ar3] Tables 1 and 26.
- IFC (k=2048, k=3072): [SP800-56Br2] Table 4.

The Module does not implement the following:

- Non-Approved Algorithms Allowed in the Approved Mode of Operation.
- Non-Approved Algorithms Allowed in the Approved Mode of Operation with No Security Claimed.
- Non-Approved Algorithms Not Allowed in the Approved Mode of Operation.

The Module supports the following ciphersuites for use with Module TLS v1.2 primitives[2]:

- TLS_DH_RSA_WITH_AES_256_CBC_SHA256 [RFC5246] (Hex Enum: 0x00,0x69)
  - KEx: DHE
  - Signature: RSA
  - PRF: HMAC-SHA2-256
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 [RFC5246] (Hex Enum: 0x00,0x6B)
  - KEx: DHE
  - Signature: RSA
  - PRF: HMAC-SHA2-256
  - Cipher: AES-256
  - Authentication: HMAC

[2] Specified according to IETF cipher suite enumeration conventions.

- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 [RFC5288] (Hex Enum: 0x00,0x9F)
  - KEx: DHE
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_DH_RSA_WITH_AES_256_GCM_SHA384 [RFC5288] (Hex Enum: 0x00,0xA1)
  - KEx: DH
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384 [RFC5289] (Hex Enum: 0xC0,0x26)
  - KEx: ECDH
  - Signature: ECDSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 [RFC5289] (Hex Enum: 0xC0,0x28)
  - KEx: ECDHE
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384 [RFC5289] (Hex Enum: 0xC0,0x2A)
  - KEx: ECDH
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384 [RFC5289] (Hex Enum: 0xC0,0x2E)
  - KEx: ECDH
  - Signature: ECDSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 [RFC5289] (Hex Enum: 0xC0,0x30)
  - KEx: ECDHE
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM

- TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384 [RFC5289] (Hex Enum: 0xC0,0x32)
  - KEx: ECDH
  - Signature: RSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 [RFC5289] (Hex Enum: 0xC0,0x2C)
  - KEx: ECDHE
  - Signature: ECDSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 [RFC5289] (Hex Enum: 0xC0,0x24)
  - KEx: ECDHE
  - Signature: ECDSA
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_PSK_WITH_AES_256_CBC_SHA [RFC4279] (Hex Enum: 0x00,0x8D)
  - KEx: PSK
  - Signature: N/A
  - PRF: HMAC-SHA-1
  - Cipher: AES-256
  - Authentication: HMAC
- TLS_PSK_WITH_AES_256_GCM_SHA384 [RFC5487] (Hex Enum: 0x00,0xA9)
  - KEx: PSK
  - Signature: N/A
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: GCM
- TLS_PSK_WITH_AES_256_CBC_SHA384 [RFC5487] (Hex Enum: 0x00,0xAF)
  - KEx: PSK
  - Signature: N/A
  - PRF: HMAC-SHA2-384
  - Cipher: AES-256
  - Authentication: HMAC

The Module supports the following SSH parameters for use with Module SSH primitives (the SSH protocol allows any combination):

Key Exchange

- ecdh-sha2-nistp256
- ecdh-sha2-nistp384
- ecdh-sha2-nistp521

Client Authentication

- ecdsa-sha2-nistp256
- ecdsa-sha2-nistp384

- ecdsa-sha2-nistp521
- rsa-sha2-256
- rsa-sha2-512

Cipher

- aes128-cbc
- aes192-cbc
- aes256-cbc
- aes128-gcm
- aes256-gcm

MAC

- hmac-sha2-256
- hmac-sha2-384

3 Cryptographic Module Interfaces

The Module supports the physical ports shown in **Error! Reference source not found.** and **Error! Reference source not found.** in Section 2 of this document. The ports and corresponding logical interfaces are described in Table 4.

Table 4: Ports and Interfaces

Physical Port	Logical Interface	Data that passes over port/interface
Front Panel (left to right)		
LED (qty. 1): "PWR".	SO.	Phio TX power status (green=on).
LCD screen (qty. 1): Basic information display.	SO.	Status output, two lines of 20 characters. Line 1: Phio TX status. Line 2: Rotating display of temperature, fan speed and IP addresses for SFP/Ethernet connections.
Control button (qty. 4): Pushbuttons.	CI.	Unused.
LED (qty. 4): Indicators (details at right).	SO.	Activity LEDs (green = OK; red = not OK). "A": Peer (incoming) activity. "B": Server (outgoing) activity. "C": Client activity. "D": Parcel (information exchange) activity.
Reset button (qty. 1): Recessed paper clip reset button.	CI.	Push button reset.
Serial console (qty. 1): RJ-45 connector.	DI, DO, CI, SO.	System console, local administration TTY.
USB (qty. 2): USB 3.0 ports.	DI, DO, CI, SO.	System update.
Ethernet (qty. 6): RJ-45 connectors (numbered 1 - 6).	DI, DO, CI, SO.	Phio Hive and client endpoint traffic.
Ethernet activity LED (qty. 12): Activity/rate indicators on each RJ-45.	SO.	Left LED: activity. Right LED: connection rate.
SFP1 (qty. 1): Fiber optic networking connector.	DI, DO, CI, SO.	Phio Hive and client endpoint traffic.
SFP2 (qty. 1): Fiber optic networking connector.	DI, DO, CI, SO.	Phio Hive and client endpoint traffic.
Back Panel (left to right)		
Ground (qty. 1): Banana plug for ESD wrist strap.	N/A.	N/A – power.
Ground (qty. 1): Chassis ground screw.	N/A.	N/A – power (Note: left of AC power switch).
Power switch (qty. 1): AC power rocker switch.	CI.	N/A – power.
Alarm reset (qty. 1): Power supply alarm reset.	SO.	Unused. Present only for supplies with audible alarms.
LED (qty. 2): Power indicators (on power supplies).	SO.	Individual power supply indicators.
Power (qty. 2): Power supply (2x) NEMA connectors.	N/A.	N/A – power.

DI: Data in; DO = Data out; CI = Control in; CO = Control out (N/A); SO = Status out. Power and ground are not logical interfaces, and as such are marked as N/A in the logical interface and data columns.

4 Roles, Services and Authentication

The Module supports three roles using role-based authentication: the Cryptographic Officer (CO) role, a limited privilege Admin role (abbreviated as 'A' in tables below) and the User role. Each role is implicitly identified by the service requested, with authentication as shown in Table 6.

The *Status* service may be used to determine the current status of the Module, as well as provide the CMVP listing information for the Module (identifiers and version).

Table 5: Roles, Service Commands, Input and Output

Role	Service	Input	Output
CO	CLI: Console connect.	Username, password.	Status; CLI shell prompt on success.
CO, A	CLI: SSH connect.	Username, password. SSH handshake input.	Status; CLI shell prompt on success. SSH handshake output.
CO, A	CLI: Miscellaneous.	Phio TX CLI command and arguments.	Status; CLI shell prompt on success.
CO, A	CLI: Certificate management.	Phio TX CLI command and arguments.	Status; CLI shell prompt on success.
CO, A	CLI: User management.	Phio TX CLI command and arguments.	Status; CLI shell prompt on success.
CO, A	CLI: Tests.	Phio TX CLI command and arguments.	Status; CLI shell prompt on success.
CO, A	CLI: Status.	*tx_status* command line.	Status; module identifier & version; prompt.
CO, A	CLI: Run CASTs.	*txfips* command line.	Status; CAST results; prompt.
CO, A	CLI: Run FW integrity tests.	*txscan* command line.	Status; prompt.
CO, A	CLI: Zeroise.	*tx_reset* command.	Status; prompt.
CO, A	Web Monitor.	https / TLS handshake input. Mouse and keyboard events.	https / TLS handshake output. Status; Web monitor page display.
User	ETSI REST Services.	JSON GET/POST/PUT messages.	Status; JSON response messages.
User	SKIP REST Services.	JSON GET/POST/PUT messages.	Status; JSON response messages.
User	Phio Hive REST Services.	JSON GET/POST/PUT messages.	Status; JSON response messages.

Commands corresponding to [FIPS140-3] required services are italicized: *tx_status* (output module identifier and version, output status); *txfips* and *txscan* (user callable self-tests); *tx_reset* (zeroisation).

Table 6: Roles and Authentication

Role	Authentication Method	Authentication Strength	
		One time	Within 1 minute
CO, A	Debian login (memorized secret).	6.6E+15	1.1E+08
CO, A	OpenSSL client authentication (based on ECDSA P-384 SHA2-384 signature verification).	6.3E+57	1.0E+50
User	JSON Web Token authentication (based on RSA 2048).	5.2E+33	8.7E+25

For an operator in the CO or Admin role, connected via the console or SSH (Debian login (memorized secret)):

- The 8-character minimum password length is enforced by the following setting in /etc/pam.d/common-password: password [success=2 default=ignore] pam_unix.so obscure sha512 **minlen=8**
- The one-time strength of this authentication method (assuming the 95 printable characters) is 95^8 = 6.6E+15.
- The one-minute security strength (based on a minimum 1 µsec authentication time) is $95^8/(60*1E+6)$ = 1.1E+08.
- The use of SSH during the process of authentication is permitted per [FIPS140-3_IG] 4.1.A Resolution d.

For an operator in the CO or Admin role using Web GUI [RFC7519]:

- The one-time strength of this authentication method is based on ECDSA P-384 SHA2-384 signature verification, providing 192 bits of security strength or 2^{192} = 6.3E+57.
- The one-minute security strength (based on a minimum 1 µsec authentication time) is $2^{192}/(60*1E+6)$ = 1.0E+50.

For an operator in the User role via REST API:

- TLS client authentication using RSA signature verification (as the minimum strength ciphersuite option), providing 112 bits of security or 2^{112} = 5.2E+33.

- The one-minute security strength (based on a minimum 1 µsec authentication time) is $2^{112}/(60*1E+6) = 8.7E+25$.

4.1 Services and Access to SSPs

Table 7 describes all Module services and service access to SSPs. The '>' character in the Indicator column refers to updates in indication depending on outcome. The CLI indication in the Service column indicates a command line interface interaction via either the Console connection or an SSH connection. The SSH† row entry applies to all services marked with SSH†. The TLS‡ row entry applies to all services marked with TLS‡.

Access column	**Indicator column**
G = Generate: The Module generates or derives the SSP. **R = Read**: The SSP is read from the Module (e.g., the SSP is output). **W = Write**: The SSP is updated, imported, or written to the Module. **E = Execute**: The Module uses the SSP for a cryptographic operation. **Z = Zeroise**: The Module zeroises the SSP.	**LCD-NR**: Front panel LCD "NOT running". **LCD-R:** Front panel LCD "Running". **LCD-STR:** Front panel LCD "Selftest Running". **LCD-IF:** Integrity Fail. **LCD-CF:** CAST Fail. **SH-CMD:** Phio TX shell command. **SH-LOGIN:** Phio TX shell login, prompt, and response text. **SH-STAT:** *tx_status* command response. **SH-CAST:** *txfips* command response. **SH-FWIT:** *txscan* command response. **SH-RST:** *tx_reset* command response.

Table 7: Approved Services

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access Rights to Keys and/or SSPs	Indicator
CLI: Console connect.	Console login, password (memorized secret) authentication.	N/A.	CO-IPW CO-RPW	CO	WEZ E	LCD-R SH-LOGIN
CLI: SSH connect.	SSH login, password (memorized secret) authentication. SSH†.	SSH†.	CO-IPW CO-RPW SSH†	CO, A	WEZ E SSH†	LCD-R SH-LOGIN
CLI: Miscellaneous. No security function or CSP usage.	CO administrative commands, in the following categories: - Power-off or reboot; - Configuration.	N/A.	N/A	CO, A	N/A	LCD-R SH-LOGIN
CLI: Certificate management.	Commands to generate, install, set or remove certificates and keys.	DSA KeyGen, ECDSA KeyGen, ECDSA SigGen, ECDSA SigVer, RSA SigGen, RSA SigVer, CKG.	SSH-HSK-Pri SSH-HSK-Pub SSH-KEX-Pri SSH-KEX-Pub TLS-HSK-Pri TLS-HSK-Pub TLS-KEX-Pri TLS-KEX-Pub	CO, A	GWEZ GWEZ GWEZ GWEZ GWEZ GWEZ GWEZ GWEZ	LCD-R SH-CMD
CLI: User management.	Add, remove or update users.	N/A.	CO-IPW CO-RPW	CO, A	W W	LCD-R
CLI: Tests.	Check: - a Phio Hive connection. - e-mail problem notification setup. SSH† (command invocation). TLS‡ (peer interaction).	SSH† TLS‡.	SSH† TLS‡	CO, A	SSH† TLS†	LCD-R

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access Rights to Keys and/or SSPs	Indicator
CLI: Status.	Report Phio TX identifier, version, and other health and status information.	N/A.	N/A	CO, A	N/A	LCD-R SH-STAT
CLI: Run CASTs.	Run CASTs on demand (all conditional self-tests in Section 10, for self-test only).	SSH†.	SSH†	A	SSH†	LCD-R > LCD-CF SH-CAST
CLI: Run FW integrity tests.	Run firmware integrity test on demand.	SSH†. HMAC-SHA2-384.	N/A SSH†	CO, A	N/A SSH†	LCD-R > LCD-CF SH-FWIT
CLI: Zeroise.	Factory reset zeroises Phio TX SSPs.	N/A.	JWT-Pub CO-IPW CO-RPW SSH-HSK-Pri SSH-HSK-Pub SSH-KEX-Pri SSH-KEX-Pub SSH-EDK SSH-MK SSH-SS TLS-EDK TLS-HSK-Pri TLS-HSK-Pub TLS-KEX-Pri TLS-KEX-Pub TLS-MK TLS-MS TLS-SS	CO, A	Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z	LCD-R > LCD-NR SH-RST
Web Monitor.	Visual display of Phio Hive connections. TLS‡ (web server interaction).	TLS‡.	TLS‡ CO-IPW CO-RPW	CO, A	TLS† W E	LCD-R
ETSI REST Services.	Connect to an endpoint using ETSI. Respond to ETSI REST API functions. TLS‡ (REST interaction).	TLS‡.	TLS‡	User	TLS†	LCD-R
SKIP REST Services.	Connect to an endpoint using SKIP. Respond to SKIP REST API functions. TLS‡ (REST interaction).	TLS‡.	TLS‡	User	TLS†	LCD-R
Phio Hive REST Services.	Connect to a Phio TX peer. Respond to Phio TX REST API functions. TLS‡ (REST interaction).	TLS‡.	TLS‡	User	TLS†	LCD-R
SSH† = SSH Usage.	This entry indicates all security functions and SSPs associated with SSH usage. SSH is used for Admin shell functions.	AES-CBC, AES-GCM, ECDSA KeyGen, ECDSA SigGen, ECDSA SigVer, HMAC-SHA2-256, HMAC-SHA2-384, KAS-ECC-SSC, RSA SigGen, RSA SigVer, KDF SSH, CKG.	SSH-HSK-Pri SSH-HSK-Pub SSH-KEX-Pri SSH-KEX-Pub SSH-EDK SSH-MK SSH-SS	CO, A	GEZ	LCD-R SH-LOGIN

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access Rights to Keys and/or SSPs	Indicator
TLS‡ = TLS usage.	This entry indicates all security functions and SSPs associated with TLS usage. TLS is used for Web Monitor (CO) and ETSI, SKIP and PHIO REST services (User).	AES-CBC, AES-GCM, DSA KeyGen, ECDSA KeyGen, ECDSA SigGen, ECDSA SigVer, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA-1, KAS-ECC-SSC, KAS-FFC-SSC, RSA SigGen, RSA SigVer, TLS v1.2 KDF [RFC7627], CKG.	TLS-HSK-Pri TLS-HSK-Pub TLS-KEX-Pri TLS-KEX-Pub TLS-SS TLS-MS TLS-EDK TLS-MK	CO, A, User	E ER E ER GEZ GEZ GEZ GEZ	LCD-R

The Module supports self-initiated cryptographic output capability in the context of the TLS v1.2 protocol. The following two actions are performed prior to activation of the self-initiated cryptographic output capability:
Action 1: installation of credentials (PKIs), using tx_generate_tx_csr followed by tx_install_tx_crt.
Action 2: registration of the peer in the config file, specifying name (matching the peer CN) and optionally IP address.
For self-initiated cryptography to function, both actions are required on each peer (both sides of the connection).

The Module supports a partial firmware load from an external source and this functionality is limited to the Crypto Officer role thus ensuring that unauthorised access to and use of the Module is not feasible. The firmware load test using HMAC-SHA2-384 performed prior to installation of the loaded firmware ensures an isolation of code.

5 Software/Firmware Security

The Module uses HMAC-SHA2-384 performed over all Module firmware as the integrity technique. The operator can initiate the integrity test on demand via the *txscan* command.The Phio TX application is delivered in a single executable binary, as a self-extracting installation package which can be installed on any system on which the base OS has been installed.

The Module supports firmware loading (partial update). HMAC-SHA2-384 with (256-bit HMAC key) is performed on updated components.

6 Operational Environment

The Module is classified in [FIPS140-3] terms as a limited operational environment.

7 Physical Security

The Phio TX is a hardware Module (multichip standalone embodiment) packaged in a metal chassis as shown in Figure 1 and Figure 2. The enclosure is protected by four (4) tamper-evident seals as shown in Figure 6, is production grade and opaque in the visible spectrum. The Crypto Officer role is responsible for securing and having control at all times of any unused seals, and the direct control and observation of any changes to the Module such as reconfigurations where the

tamper evident seals or security appliances are removed or installed to ensure the security of the Module is maintained during such changes and the Module is returned to the Approved state.

Rear right tamper seal | Rear left tamper seal | Left front tamper seal | Right front tamper seal

Figure 6: Module Seal Application Locations

Table 8: Physical Security Inspection Guidelines

Physical Security Mechanism	Recommended Frequency of Inspection/Test	Inspection/Test Guidance Details
Tamper-evident seals (qty. 4) over chassis screws.	Seals should be inspected during physical maintenance operations and when circumstances dictate (e.g., if tampering is suspected).	Inspect seals for evidence of lifted edges or excessive wear.

8 Non-Invasive Security

The Module does not implement non-invasive security measures.

9 Sensitive Security Parameters Management

Table 9 summarizes the SSPs implemented by the Module. System keys handled by the Phio TX are not classified as SSPs as they are not used by the Phio TX. All such data crosses the boundary via TLS connections, such that system secrets are TLS protected.

Generation, inclusive of derivation (Key-entity association)
G1: Generated by on-chip [SP800-90B] ENT (NP) (memory map).
G2: Derived using [SP800-90Ar1] with Block_cipher_df (memory map).
G3: FFC (DSA) or ECC (ECDSA) or RSA (memory map).

Establishment (equated to Key Agreement [FIPS140-3_IG] D.F)
E1: Calculated using KAS-ECC-SSC or KAS-FFC-SSC.
E2: Calculated using approved TLS KDF.
E3: Calculated using approved SSH KDF.

Storage (associated to entity by memory mapping)
S1: Stored in plaintext in RAM, dynamic storage.
S2: Stored in PEM format in hard drive media, static storage.

Zeroisation
Z1: Overwritten by zeros after use (Module initiated), shutdown or loss of power (operator initiated).
Z2: Overwritten by zeros by Zeroisation service (operator initiated).

Import/Export (equated to Key Transport [FIPS140-3_IG] D.G)
IE1: Provided in plaintext to the Module (message parameter).
IE2: Imported and exported in ciphertext (positional parameter).
IE3: Established in Quantum Xchange manufacturing facility.

Table 9: SSPs

Key/SSP Name/Type	Strength[3]	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroisation	Use and Related Keys
DRBG-EI (CSP)	256	ENT (NP).	G1	--	--	S1	Z1	Entropy input – see detail below in Table 10.
DRBG-Seed (CSP)	384	ENT (NP), Counter DRBG #A2983.	G2	--	--	S1	Z1	Seed derived using the entropy input.
DRBG-State (CSP)	256	Counter DRBG #A2983.	G2	--	--	S1	Z1	DRBG internal state (V and C).
JWT-Pub (PSP)	128	ECDSA P-256 #A2983.	--	IE2 AD /EE	--	S2	Z2	JSON Web Token authentication (only GUI).
CO-IPW (CSP)	64	Login.	--	IE1 MD /DE or AD /EE	--	S1	Z1	CO role password input.
CO-RPW (CSP)	64	Login.	--	IE3	--	S2	Z2	CO role stored password reference.
SSH-HSK-Pri (CSP)	128 or 192	ECDSA SigGen #A2983. RSA SigGen #A2983. CKG.	G3	IE3	--	S2	Z2	ECDSA (P-256, P-384) or RSA (2048, 3072) private key - signature generation.
SSH-HSK-Pub (PSP)	128 or 192	ECDSA SigVer #A2983. RSA SigVer #A2983. CKG.	G3	IE3	--	S2	Z2	ECDSA (P-256, P-384) or RSA (2048, 3072) public key - signature verification.
SSH-KEX-Pri (CSP)	128 or 192	KAS-ECC-SSC #A2983. ECDSA KeyGen #A2983 CKG.	G3	IE3	--	S1 S2	Z1 Z2	ECDSA (P-256, P-384) private key for SSH key exchange.
SSH-KEX-Pub (PSP)	128 or 192	KAS-ECC-SSC #A2983.	--	IE1 AD /EE	--	S1 S2	Z1 Z2	ECDSA (P-256, P-384) public key for SSH key exchange.
SSH-EDK (CSP)	128, 192 (CBC only) or 256	AES-CBC #A2983, AES-GCM #A2983 KDF SSH #A2983 CKG.	--	--	E3	S1	Z1	SSH Session Encrypt Decrypt Key (cipher key).
SSH-MK (CSP)	256 or 384	HMAC-SHA2-256 #A2983, HMAC-SHA2-384 #A2983 KDF SSH #A2983. CKG.	--	--	E3	S1	Z1	SSH Session MAC Key (for non-AEAD suites).
SSH-SS (CSP)	128, 192 or 256	KAS-ECC-SSC #A2983. CKG.	--	--	E1	S1	Z1	SSH Shared secret: derive SSH key block.
TLS-EDK (CSP)	256	AES-CBC #A2983, AES-GCM #A2983. CKG.	--	--	E2	S1	Z1	TLS Session Encrypt Decrypt Key (cipher key).
TLS-HSK-Pri (CSP)	128 or 192 112 or 128	ECDSA SigGen #A2983. RSA SigGen #A2983.	G3	IE3	--	S2	Z2	ECDSA (P-256, P-384) or RSA (2048, 3072) private key for digital signature generation.

[3] Strength is provided in bits. Please refer to Table 3 and the notes below it for the strength provenance (traceability to applicable standards and special publications).

		CKG.						
TLS-HSK-Pub (PSP)	128 or 192 112 or 128	ECDSA SigVer #A2983. RSA SigVer #A2983. CKG.	G3	IE3	--	S2	Z2	ECDSA (P-256, P-384) or RSA (2048, 3072) public key for digital signature verification.
TLS-KEX-Pri (CSP)	112 or 128 128 or 192	KAS-FFC-SSC #A2983. KAS-ECC-SSC #A2983. ECDSA KeyGen #A2983. DSA KeyGen #A2983. CKG	G3	IE3	--	S1 S2	Z1 Z2	ECDSA (P-256, P-384) or FFC (L=2048, N=224, 256; L=3072, N=256) private key for TLS KAS SSC.
TLS-KEX-Pub (PSP)	112 or 128 128 or 192	KAS-FFC-SSC #A2983. KAS-ECC-SSC #A2983.	--	IE1 AD /EE	--	S1 S2	Z1 Z2	ECDSA (P-256, P-384) or FFC (L=2048, N=224, 256; L=3072, N=256) public key for TLS KAS SSC.
TLS-MK (CSP)	256 or 384	HMAC-SHA2-256 #A2983, HMAC-SHA2-384 #A2983 HMAC-SHA-1 #A2983. CKG.	--	--	E2	S1	Z1	TLS Session MAC Key (for non-AEAD suites).
TLS-MS (CSP)	256 or 384	TLS v1.2 KDF [RFC7627] #A2983. CKG.	--	--	E2	S1	Z1	TLS Master Secret: derive key block; finalize.
TLS-SS (CSP)	128 or 192 112 or 128	KAS-ECC-SSC #A2983. KAS-FFC-SSC #A2983. CKG.	--	--	E1	S1	Z1	TLS Shared Secret: derive TLS Master Secret.

-- = not applicable.

The firmware integrity key (FWIK, formally not a SSP) is a 256-bit HMAC key used with HMAC-SHA2-384.

Table 10: Non-Deterministic Random Number Generation Specification

Entropy sources	Minimum number of bits of entropy	Details
Jent	[SP800-90Ar1] *min_length*: 256 bits. [SP800-90Ar1] *seedlen*: 384 bits.	[FIPS140-3_IG] 9.3.A: The Module generates ENT within the Module boundary: option 1(a) using a [SP800-90B] compliant ENT (NP). Per [SP800-90Ar1] Table 2, the AES CTR_DRBG requires 384 bits of entropy in the *DRBG_Seed* value. As input to the [SP800-90Ar1] *Block_cipher_df*, the Module provides 512 bits of entropy, well in excess of the minimum requirement.

10 Self-tests

The Module automatically invokes all tests listed below on each power-on or reset. The FW integrity test is run periodically as a scheduled system service (nominally every eight hours) and may also be run on demand by running *txscan* on an SSH or Console connection.

The complete set of ACVP tests used to achieve the CAVP listings are run at power-on (as listed under Conditional Self-tests below), are also run periodically as a scheduled system service (nominally every eight hours) and may also be run on demand by running *txfips* on an SSH or Console connection. All cryptographic algorithm self-tests (CASTs) must complete successfully prior to any other use of cryptography by the Module. If the firmware integrity test fails, the Module enters the *ERR_TXSCAN* state (indicated by LCD-IF, see Table 7). If one of the CASTs fails, the Module enters the *ERR_TXFIPS* state (indicated by LCD-CF, see Table 7).

The *ERR_TXFIPS* and *ERR_TXSCAN* error states are persistent, cleared by successful completion of *txfips* or *txscan*, respectively. When in an error state, only self-test and status services are available. All attempts to use the Module's services result in the return of the error state indicator (*Failed TXFIPS* or *Failed TXSCAN*).

Pre-operational Self-tests

- Pre-operational firmware integrity test: HMAC-SHA2-384 with (256-bit HMAC key) digest verification performed over all Module firmware.

Conditional Self-tests

- txfips Conditional Cryptographic Algorithm tests:
    - AES-ECB, AES-CBC #A2983: CASTs - complete set of AES ECB and CBC ACVP test vectors.
        - AES-CBC-128 encrypt KAT.
        - AES-CBC-128 decrypt KAT.
        - AES-CBC-192 encrypt KAT.
        - AES-CBC-192 decrypt KAT.
        - AES-CBC-256 encrypt KAT.
        - AES-CBC-256 decrypt KAT.
    - AES-GCM #A2983: CASTs - complete set of AES GCM ACVP test vectors.
        - AES-GCM-128 encrypt KAT.
        - AES-GCM-128 decrypt KAT.
        - AES-GCM-192 encrypt KAT.
        - AES-GCM-192 decrypt KAT.
        - AES-GCM-256 encrypt KAT.
        - AES-GCM-256 decrypt KAT.
    - Counter DRBG #A2983: CASTs - complete set of DRBG ACVP test vectors. Instantiate, Generate and Reseed functions (per Section 11.3 of [SP800-90Ar1]) for AES-256 Counter DRBG with derivation function.
    - ECDSA SigGen #A2983: CASTs - complete set of ECDSA ACVP test vectors.
        - ECDSA SigGen KAT using P-256 and SHA2-256.
        - ECDSA SigGen KAT using P-256 and SHA2-384.
        - ECDSA SigGen KAT using P-384 and SHA2-256.
        - ECDSA SigGen KAT using P-384 and SHA2-384.
    - ECDSA SigVer #A2983: CASTs - complete set of ECDSA ACVP test vectors.
        - ECDSA SigVer KAT using P-256 and SHA2-256.
        - ECDSA SigVer KAT using P-256 and SHA2-384.
        - ECDSA SigVer KAT using P-384 and SHA2-256.

        - ECDSA SigVer KAT using P-384 and SHA2-384.
    - HMAC-SHA-1 #A2983: Complete set of HMAC-SHA-1 ACVP test vectors.HMAC-SHA2 #A2983: Complete set of HMAC-SHA2-256 and HMAC-SHA2-384 ACVP test vectors (performed prior to FW Integrity).
        - HMAC-SHA2-256 KAT.
        - HMAC-SHA2-384 KAT.
    - KAS-ECC-SSC #A2983: CAST - complete set of KAS-ECC-SSC ACVP test vectors.
        - [SP800-56Ar3] Section 6 Ephemeral Unified Shared Secret (Z) Computation using P-256.
        - [SP800-56Ar3] Section 6 Ephemeral Unified Shared Secret (Z) Computation using P-384.
    - KAS-FFC-SSC #A2983: CAST - complete set of KAS-FFC-SSC ACVP test vectors.
        - [SP800-56Ar3] Section 6 dhEphem Shared Secret (Z) Computation using L=2048/N=256.
        - [SP800-56Ar3] Section 6 dhEphem Shared Secret (Z) Computation using L=3072/N=256.
    - KDF SSH #A2983: CAST - complete set of SSH KDF ACVP test vectors.
    - RSA SigGen #A2983: CASTs - complete set of RSA ACVP test vectors.
        - PSS SigGen KAT using k=2048, SHA2-256.
        - PSS SigGen KAT using k=2048, SHA2-384.
        - PSS SigGen KAT using k=3072, SHA2-256.
        - PSS SigGen KAT using k=3072, SHA2-384.
    - RSA SigVer #A2983: CASTs - complete set of RSA ACVP test vectors.
        - PSS SigVer KAT using k=2048, SHA2-256.
        - PSS SigVer KAT using k=2048, SHA2-384.
        - PSS SigVer KAT using k=3072, SHA2-256.
        - PSS SigVer KAT using k=3072, SHA2-384.
    - SHA2 #A2983: CAST - complete set of SHS (SHA2) ACVP test vectors.
        - SHA2-256 KAT.
        - SHA2-384 KAT.
        - SHA2-512 KAT.
    - TLS v1.2 KDF [RFC7627] #A2983: CAST - complete set of TLS KDF ACVP test vectors.
    - ENT (NP) NIST SP 800-90B Health Tests
- Conditional pair-wise consistency tests:
    - DSA PCT (key agreement): Per [SP800-56Ar3] §5.6.2.1.4, the Phio TX generates the static and ephemeral key pairs for use in TLS.
    - ECDSA PCT (key agreement): Per [SP800-56Ar3] §5.6.2.1.4, the Phio TX generates the static and ephemeral key pairs for use in TLS and SSH.
    - ECDSA PCT (digital signature): The Phio TX performs a sign-verify PCT on every key pair generated.
- Conditional software/firmware load test: HMAC-SHA2-384 with (256-bit HMAC key) test performed on updated components.

11 Life-cycle Assurance

Installation, initialization, configuration, and provisioning are managed using Quantum Xchange provided scripts as well as the administrative tools of the Phio TX shell available via SSH or the console. This involves installation of a valid Phio TX License with locked Approved mode enforcement. Once configured, the Approved mode of operation persists.

Quantum Xchange recommends the use of the *tx_reset* command to reset the Phio TX to factory defaults.

The Module Guidance Documentation [GD] provides detailed procedures for secure installation, initialization, configuration, provisioning, decommissioning, and sanitization of the Module. No maintenance requirements are defined for the Phio TX. The Phio TX application is delivered as a single executable binary. The binary must be run with the root account to install the upgrade. The firmware load test and integrity tests implemented ensure protection of the firmware during delivery.

12 Mitigation of Other Attacks

The Module does not implement mitigations of other attacks outside the scope of [FIPS140-3].

References

- [FIPS140-3]: FIPS 140-3, Security Requirements for Cryptographic Modules, 22-Mar-2019
- [SP800-140_DTR]: NIST SP 800-140, FIPS 140-3 Derived Test Requirements (DTR): CMVP Validation Authority Updates to ISO/IEC 24759, 20-Mar-2020
- [SP800-140A]: NIST SP 800-140A, CMVP Documentation Requirements: CMVP Validation Authority Updates to ISO/IEC 24759, 20-Mar-2020
- [SP800-140B]: NIST SP 800-140B, CMVP Security Policy Requirements: CMVP Validation Authority Updates to ISO/IEC 24759 and ISO/IEC 19790 Annex B, 20-Mar-2020
- [SP800-140Cr2]: NIST SP 800-140C Rev. 2, Cryptographic Module Validation Program (CMVP)-Approved Security Functions: CMVP Validation Authority Updates to ISO/IEC 24759, 25-Jul-2023
- Supplemental Information: SP 800-140C: Approved Security Functions, 25-Jul-2023
- [SP800-140Dr2]: NIST SP 800-140D Rev. 2, Cryptographic Module Validation Program (CMVP)-Approved Sensitive Security Parameter Generation and Establishment Methods: CMVP Validation Authority Updates to ISO/IEC 24759, 25-Jul-2023
- Supplemental Information: SP 800-140D: Approved SSP Generation and Establishment Methods, 25-Jul-2023
- [SP800-140E]: NIST SP 800-140E, CMVP Approved Authentication Mechanisms: CMVP Validation Authority Requirements for ISO/IEC 19790 Annex E and ISO/IEC 24579 Section 6.17, 20-Mar-2020
- [SP800-140F]: NIST SP 800-140F, CMVP Approved Non-Invasive Attack Mitigation Test Metrics: CMVP Validation Authority Updates to ISO/IEC 24759, 20-Mar-2020
- [FIPS140-3_IG]: Implementation Guidance for FIPS 140-3 and the Cryptographic Module Validation Program, 1-Aug-2023
- [I19790]: ISO/IEC 19790:2012 Information technology -- Security techniques -- Security requirements for cryptographic modules, 1-Nov-2015
- [I24759]: ISO/IEC 24759:2017 Information technology -- Security techniques -- Test requirements for cryptographic modules, 1-Mar-2017
- [FIPS180-4]: FIPS 180-4, Secure Hash Standard (SHS), 4-Aug-2015
- [FIPS186-4]: FIPS 186-4, Digital Signature Standard (DSS), 19-Jul-2013
- [FIPS197]: FIPS 197, Advanced Encryption Standard (AES), 9-May-2023
- [FIPS198-1]: FIPS 198-1, The Keyed Hash Message Authentication Code (HMAC), 16-Jul-2008
- [SP800-38A]: NIST SP 800-38A, Recommendation for Block Cipher Modes of Operation: Methods and Techniques, 1-Dec-2001
- [SP800-38D]: NIST SP 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, 28-Nov-2007
- [SP800-52r2]: NIST SP 800-52 Rev. 2, Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations, 29-Aug-2019
- [SP800-56Ar3]: NIST SP 800-56A Rev. 3, Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, 16-Apr-2018
- [SP800-56Br2]: NIST SP 800-56B Rev. 2, Recommendation for Pair-Wise Key-Establishment Using Integer Factorization Cryptography, 21-Mar-2019

- [SP800-57P1r5]: NIST SP 800-57 Part 1 Rev. 5, Recommendation for Key Management: Part 1 – General, 4-May-2020
- [SP800-90Ar1]: NIST SP 800-90A Rev. 1, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, 24-Jun-2015
- [SP800-90B]: NIST SP 800-90B, Recommendation for the Entropy Sources Used for Random Bit Generation, 10-Jan-2018
- [SP800-133r2]: NIST SP 800-133 Rev. 2, Recommendation for Cryptographic Key Generation, 4-Jun-2020
- [SP800-135r1]: NIST SP 800-135 Rev. 1, Recommendation for Existing Application-Specific Key Derivation Functions, 23-Dec-2011
- [SP800-186]: NIST SP 800-186, Recommendations for Discrete Logarithm-based Cryptography: Elliptic Curve Domain Parameters, 3-Feb-2023
- [RFC4279]: RFC 4279, Pre-Shared Key Ciphersuites for Transport Layer Security (TLS), Dec-2005
- [RFC5246]: RFC 5246, The Transport Layer Security (TLS) Protocol Version 1.2, Aug-2008
- [RFC5288]: RFC 5288, AES Galois Counter Mode (GCM) Cipher Suites for TLS, Aug-2008
- [RFC5289]: RFC 5289, TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM), Aug-2008
- [RFC5487]: RFC 5487, Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode, Mar-2009
- [RFC7519]: RFC 7519, JSON Web Token (JWT), May-2015
- [RFC7627]: RFC 7627, Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension, Sept-2015
- [GD]: Quantum Xchange Phio TX FIPS 140-3 Guidance Documentation, 1-Sep-2022

Acronyms and Definitions

- ACVP: Automated Cryptographic Validation Program
- ACVT: Automated Cryptographic Validation Testing
- AES: Advanced Encryption Standard, see [FIPS197]
- API: Application Programming Interface
- CAST: Cryptographic Algorithm Self-Test
- CKG: Cryptographic Key Generation
- CMVP: Cryptographic Module Validation Program
- CO: Cryptographic Officer
- CSP: Critical Security Parameter, see [FIPS140-3]
- CPU: Central Processing Unit
- CRC: Cyclic Redundancy Check
- CCCS: Canadian Centre of Cybersecurity
- CVL: Component Validation List
- DRAM: Dynamic Random-Access Memory

- DRBG: Deterministic Random Number Generator, see [SP800-90Ar1]
- DSA: Digital Signature Algorithm, see [FIPS186-4]
- DTR: Derived Test Requirements
- ECB: Electronic Code Book
- ECC: Elliptic Curve Cryptography
- ECDSA: Elliptic Curve Digital Signature Algorithm, see [FIPS186-4]
- EMS: Extended Master Secret
- ETSI: European Telecommunications Standards Institute
- FFC: Finite Field Cryptography
- FIPS: Federal Information Processing Standard
- GCM: Galois/Counter Mode
- HMAC: Keyed-Hash Message Authentication Code, see [FIPS198-1]
- IG: Implementation Guidance, see [FIPS140-3_IG]
- IV: Initialization Vector
- KAS: Key Agreement Scheme
- KDF: Key Derivation Function
- MAC: Message Authentication Code
- NIST: National Institute of Standards and Technology
- OE: Operating Environment
- PKI: Public Key Infrastructure
- PRF: Pseudo-Random Function
- PSK: Pre-Shared Key
- PSP: Public Security Parameter
- QKD: Quantum Key Distribution
- RSA: Rivest, Shamir and Adleman algorithm, see [FIPS186-4]
- SHA/SHS: Secure Hash Algorithm/Standard, see [FIPS180-4]
- SKIP: Secure Key Integration Protocol, Cisco
- SP: NIST Special Publication
- SSC: Shared Secret Computation
- SSP: Sensitive Security Parameter
- TX: Trusted Exchange (Phio TX)