Functional Package for Secure Shell (SSH)

Version: 1.0-draft
2021-03-11
National Information Assurance Partnership

Revision History

Version	Date	Comment
1.0 draft	2021-01-05	DRAFT: First draft as a Functional Package
1.0 draft	2021-03-11	DRAFT: Incorporated CCUF CWG input.

Contents

1 Introduction

1.1 Overview

Secure Shell (SSH) is a protocol for secure remote login and other secure network services over an untrusted network. SSH software can act as a client, server, or both.

This *Functional Package for Secure Shell* provides a collection of Secure Shell (SSH) protocol related SFRs and Evaluation Activities (EAs) covering audit, authentication, cryptographic algorithms, and protocol negotiation. The intent of this package is to provide PP, cPP, and PP-Module authors with a readily consumable collection of SFRs and EAs to be integrated into their documents.

1.2 Terms

The following sections list Common Criteria and technology terms used in this document.

1.2.1 Common Criteria Terms

Assurance	Grounds for confidence that a TOE meets the SFRs [CC].
Base Protection Profile (Base-PP)	Protection Profile used as a basis to build a PP-Configuration.
Common Criteria (CC)	Common Criteria for Information Technology Security Evaluation (International Standard ISO/IEC 15408).
Common Criteria Testing Laboratory	Within the context of the Common Criteria Evaluation and Validation Scheme (CCEVS), an IT security evaluation facility, accredited by the National Voluntary Laboratory Accreditation Program (NVLAP) and approved by the NIAP Validation Body to conduct Common Criteria-based evaluations.
Common Evaluation Methodology (CEM)	Common Evaluation Methodology for Information Technology Security Evaluation.
Distributed TOE	A TOE composed of multiple components operating as a logical whole.
Operational Environment	Hardware and software that are outside the TOE boundary that support the TOE functionality and security policy.

(OE)	
Protection Profile (PP)	An implementation-independent set of security requirements for a category of products.
Protection Profile Configuration (PP-Configuration)	A comprehensive set of security requirements for a product type that consists of at least one Base-PP and at least one PP-Module.
Protection Profile Module (PP-Module)	An implementation-independent statement of security needs for a TOE type complementary to one or more Base Protection Profiles.
Security Assurance Requirement (SAR)	A requirement to assure the security of the TOE.
Security Functional Requirement (SFR)	A requirement for security enforcement by the TOE.
Security Target (ST)	A set of implementation-dependent security requirements for a specific product.
TOE Security Functionality (TSF)	The security functionality of the product under evaluation.
TOE Summary Specification (TSS)	A description of how a TOE satisfies the SFRs in an ST.
Target of Evaluation (TOE)	The product under evaluation.

1.2.2 Technical Terms

Secure Shell (SSH)	Cryptographic network protocol for initiating text-based shell sessions on remote systems.
connection	The SSH transport layer between a client and a server. Within a connection there can be multiple sessions.
rekey	Where the connection renegotiates the shared secret and each session subsequently derives a new encryption key.
session	A discrete stream of data within a connection.

1.3 Compliant Targets of Evaluation

The Target of Evaluation (TOE) in this Functional Package (FP) is a product which acts as an SSH client, SSH server, or both. This FP describes the extended security functionality of SSH in terms of [CC].

The contents of this Functional Package must be appropriately incoporated into a PP, cPP, or PP-Module. When this package is so incorporated, the ST must include selection-based requirements in accordance with the selections or assignments indicated in the incorporating document.

The PP, cPP, or PP-Module that instantiates this Package must typically include the following components in order to satisfy dependencies of this Package. It is the responsibility of the PP, cPP, or PP-Module author who incorporates this FP to ensure that dependence on these components is satisfied, either by the TOE or by assumptions about its Operational Environment.

An ST must identify the applicable version of the PP, cPP, or PP-Module and this Functional Package in its conformance claims.

Component	Explanation
FCS_CKM.1	To support key generation for SSH, the incoporating document must include FCS_CKM.1 and specify the corresponding algorithm(s).
FCS_CKM.2	To support key establishment for SSH, the incoporating document must include FCS_CKM.2 and specify the corresponding algorithm(s).

FCS_COP.1	To support the cryptography needed for SSH communications, the incoporating document must include FCS_COP.1 (iterating as needed) to specify AES with corresponding key sizes and modes, digital signature generation and verification function (at least one of RSA or ECDSA), a cryptographic hash function, and a keyed-hash message authentication function. In particular, the incoporating document must support AES-CTR as defined in NIST SP 800-38A with key sizes of both 128 and 256 bits.
FCS_RBG_EXT.1	To support random bit generation needed for SSH key generation, the incorporating document must include a requirement that specifies the TOE's ability to invoke or provide random bit generation services, commonly identified as FCS_RBG_EXT.1.
FIA_X509_EXT.1	To support establishment of SSH communications using a public key algorithm that includes X.509, the incorporating document must include FIA_X509_EXT.1. Note however that support for X.509 is selectable and not mandatory.
FIA_X509_EXT.2	To support establishment of SSH communications using a public key algorithm that includes X.509, the incoporating document must include FIA_X509_EXT.2. Note however that support for X.509 is selectable and not mandatory.
FPT_STM.1	To support establishment of SSH communications using a public key algorithm that includes X.509, the incorporating document must include FPT_STM.1 or some other requirement that ensures reliable system time. Note however that support for time-based rekey thresholds is selectable and not mandatory.

2 Conformance Claims

Conformance Statement
An ST must claim exact conformance to this Package, as defined in the CC and CEM addenda for Exact Conformance, Selection-Based SFRs, and Optional SFRs (dated May 2017).

CC Conformance Claims
This Package is conformant to Parts 2 (extended) and 3 (conformant) of Common Criteria Version 3.1, Revision 5.

PP Claim
This Package does not claim conformance to any Protection Profile.

Package Claim
This Package does not claim conformance to any packages.

3 Security Functional Requirements

This chapter describes the security requirements which have to be fulfilled by the product under evaluation. Those requirements comprise functional components from Part 2 of [CC]. The following conventions are used for the completion of operations:

- **Refinement** operation (denoted by **bold text** or ~~strikethrough text~~): is used to add details to a requirement (including replacing an assignment with a more restrictive selection) or to remove part of the requirement that is made irrelevant through the completion of another operation, and thus further restricts a requirement.
- **Selection** (denoted by *italicized text*): is used to select one or more options provided by the [CC] in stating a requirement.
- **Assignment** operation (denoted by *italicized text*): is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets indicates assignment.
- **Iteration** operation: is indicated by appending the SFR name with a slash and unique identifier suggesting the purpose of the operation, e.g. "/EXAMPLE1."

3.1 Auditable Events for Mandatory SFRs

The auditable events specified in this Package are included in a Security Target if the incorporating PP, cPP, or PP-Module supports audit event reporting through FAU_GEN.1 and all other criteria in the incorporating PP or PP-Module are met.

Table 1: Auditable Events for Mandatory Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FCS_SSH_EXT.1	**[selection:** *Failure to establish SSH connection, None***]**	Reason for failure. **[selection:** *Non-TOE endpoint of attempted connection (IP Address) , None***]**
FCS_SSH_EXT.1	**[selection:** *Establishment of SSH connection, None***]**	**[selection:** *Non-TOE endpoint of connection (IP Address) , None***]**
FCS_SSH_EXT.1	**[selection:** *Termination of SSH*	**[selection:** *Non-TOE endpoint of connection*

	connection session, None]	*(IP Address) , None*]
FCS_SSH_EXT.1	[**selection:** *Dropping of packet(s) outside defined size limits, None*]	[**selection:** *Packet size , None*]

3.2 Cryptographic_Support_(FCS)

FCS_SSH_EXT.1 SSH Protocol

FCS_SSH_EXT.1.1

The TOE shall implement *SSH* acting as a [**selection**: *client*, *server*] in accordance with that complies with RFCs 4251, 4252, 4253, 4254, [**selection**: *4256*, *4344*, *5647*, *5656*, *6187*, *6668*, *8268*, *8308*, *8332*, *8709*, *8731*, *no other RFCs*] and [*no other standard*].

Application Note: The following mapping is provided as a guide to ST authors to ensure the appropriate RFC selections are made based on applicable selections in subsequent SFRs:

- RFC 4256: Select for keyboard-interactive authentication
- RFC 4344: Select for AES-128-CTR or AES-256-CTR
- RFC 5647: Select for AEAD_AES_128_GCM, AEAD_AES_256_GCM, or aes*-gcm@openssh.com
- RFC 5656: Select for elliptic curve cryptography
- RFC 6187: Select for X.509 certificate use
- RFC 6668: Select for HMAC-SHA-2 algorithms
- RFC 8268: Select for FFC DH groups with SHA-2
- RFC 8308: Select if RFC 8332 is selected
- RFC 8332: Select if SHA-2 is available with ssh-rsa
- RFC 8709: Select if ed25519 or ed448 is used as a public key algorithm
- RFC 8731: Select if curve25519 or curve448 is used for key exchange

The ST author selects which of the additional RFCs to which conformance is being claimed. An SSH product can implement additional RFCs, but only those listed in the selection can be claimed as conformant under CC. The RFC selections for this requirement must be consistent with selections in later elements of this Functional Package (e.g., cryptographic algorithms permitted).

For the purposes of this package (and subsequent integration into cPPs) only the claimed algorithms listed in the package must be enabled for use.

RFC 4253 indicates that certain cryptographic algorithms are "REQUIRED." This means that from the Internet Engineering Task Force's (IETF's) perspective the implementation must include support, not that the algorithms must be enabled for use. For the purposes of this SFR's evaluation activity and this Functional Package overall, it is not necessary to ensure that algorithms listed as "REQUIRED" by the RFC but not listed in later elements of this Functional Package are actually implemented.

RFC 4344 must be selected if aes128-ctr or aes256-ctr is selected in FCS_SSH_EXT.1.4.

RFC 4356 must be selected if "keyboard-interactive" is selected in FCS_SSH_EXT.1.2.

RFC 5647 must be selected when AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com, or aes256-gcm@openssh.com is selected as an encryption algorithm in FCS_SSH_EXT.1.4 and when AEAD_AES_128_GCM or AEAD_AES_256_GCM is selecyed as MAC algorithm in FCS_SSH_EXT.1.5.

RFC 5656 must be selected when ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521 is selected as a public key algorithm in FCS_SSH_EXT.1.2, or when ecdh-sha2-nistp256, ecdh-sha2-nistp384, or ecdh-sha2-nistp521 is selected as a key exchange algorithm in FCS_SSH_EXT.1.6, or when "RFC 5656" is selected in FCS_SSH_EXT.1.7.

RFC 6187 must be selected when x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, or x509v3-rsa2048-sha256 is selected as a public key algorithm in FCS_SSH_EXT.1.2.

RFC 6668 must be selected when hmac-sha2-256 or hmac-sha2-512 is selected as a MAC algorithm in FCS_SSH_EXT.1.5.

RFC 8268 must be selected when diffie-hellman-group14-sha256, diffie-hellman-group15-sha512, diffie-hellman-group16-sha512, diffie-hellman-group17-sha512, or diffie-hellman-group18-sha512 is selected as a key exchange algorithm in FCS_SSH_EXT.1.6.

RFC 8332 must be selected when rsa-sha2-256 or rsa-sha2-512 is selected as a public key algorithm in FCS_SSH_EXT.1.2.

RFC 8709 must be selected when ssh-ed25519 or ssh-ed448 is selected as a public key algorithm in FCS_SSH_EXT.1.2.

RFC 8731 must be selected when curve25519-sha256 or curve448-sha512 is selected as a key exchange algorithm in FCS_SSH_EXT.1.6.

If "client" is selected, then the ST must include FCS_SSHC_EXT.1.

If "server" is selected, then the ST must include FCS_SSHS_EXT.1.

FCS_SSH_EXT.1.2

The TSF shall ensure that the SSH protocol implementation supports the following authentication methods: [**selection**:

- *“password” (RFC 4252),*
- *“keyboard-interactive” (RFC 4256),*
- *“publickey” (RFC 4252): [**selection**:*
  - *ssh-rsa (RFC 4253),*
  - *rsa-sha2-256 (RFC 8332),*
  - *rsa-sha2-512 (RFC 8332),*
  - *ecdsa-sha2-nistp256 (RFC 5656),*
  - *ecdsa-sha2-nistp384 (RFC 5656),*
  - *ecdsa-sha2-nistp521 (RFC 5656),*
  - *ssh-ed25519 (RFC 8709),*
  - *ssh-ed448 (RFC 8709),*
  - *x509v3-ecdsa-sha2-nistp256 (RFC 6187),*
  - *x509v3-ecdsa-sha2-nistp384 (RFC 6187),*
  - *x509v3-ecdsa-sha2-nistp521 (RFC 6187),*
  - *x509v3-rsa2048-sha256 (RFC 6187)*

  ]

] and no other methods.

Application Note: Within SSH there are two types of authentication: user authentication and peer authentication. This SFR deals with the options supported for user authentication. Peer authentication is covered in FCS_SSHS_EXT.1.1 (for servers) and FCS_SSHC_EXT.1.1 (for clients).

FCS_SSH_EXT.1.3

The TSF shall ensure that, as described in RFC 4253, packets greater than [**assignment**: *number of bytes between 35,000 and 1 GB (inclusive)*] in an SSH transport connection are dropped.

Application Note: RFC 4253 (section 6.1) provides for the acceptance of “large packets” with the caveat that the packets should be of “reasonable length” or dropped. The assignment should be filled in by the ST author with the maximum packet size accepted, thus defining “reasonable length” for the TOE.

The upper bound on the packet size is driven by the size identified in FCS_SSH_EXT.1.8.

FCS_SSH_EXT.1.4

The TSF shall protect data in transit from unauthorised disclosure using the following mechanisms: [**selection**:

- *aes128-ctr (RFC 4344),*
- *aes256-ctr (RFC 4344),*
- *aes128-cbc (RFC 4253),*
- *aes256-cbc (RFC 4253),*
- *AEAD_AES_128_GCM (RFC 5647),*
- *AEAD_AES_256_GCM (RFC 5647),*
- *aes128-gcm@openssh.com (RFC 5647),*
- *aes256-gcm@openssh.com (RFC 5647)*

] and no other mechanisms.

Application Note: As described in RFC 5647, AEAD_AES_128_GCM and AEAD_AES_256_GCM need the corresponding MAC algorithm to be selected in FCS_SSH_EXT.1.5.

FCS_SSH_EXT.1.5

The TSF shall protect data in transit from modification, deletion, and insertion using: [**selection**:

- *hmac-sha2-256 (RFC 6668),*
- *hmac-sha2-512 (RFC 6668),*
- *AEAD_AES_128_GCM (RFC 5647),*
- *AEAD_AES_256_GCM (RFC 5647),*
- *implicit*

] and no other mechanisms.

Application Note: As described in RFC 5647, AEAD_AES_128_GCM and AEAD_AES_256_GCM need the corresponding encryption algorithm to be selected. In AES-GCM mode, integrity is not provided using a MAC, it is implicit in AES-GCM mode itself. There is no need for a corresponding FCS_COP element. The FCS_COP element for AES would already cover this.

If the negotiated encryption algorithm is one of the aes*-gcm@openssh.com algorithms, then the MAC field is ignored during negotiation and implicitly selects AES-GCM for the MAC. “implicit” is not an SSH identifier and will not be seen on the wire; however, the negotiated MAC might be decoded as “implicit”.

FCS_SSH_EXT.1.6

The TSF shall establish a shared secret with its peer using: [**selection**:

- *diffie-hellman-group14-sha256 (RFC 8268)*,
- *diffie-hellman-group15-sha512 (RFC 8268)*,
- *diffie-hellman-group16-sha512 (RFC 8268)*,
- *diffie-hellman-group17-sha512 (RFC 8268)*,
- *diffie-hellman-group18-sha512 (RFC 8268)*,
- *ecdh-sha2-nistp256 (RFC 5656)*,
- *ecdh-sha2-nistp384 (RFC 5656)*,
- *ecdh-sha2-nistp521 (RFC 5656)*,
- *curve25519-sha256 (RFC 8731)*,
- *curve448-sha512 (RFC 8731)*

] and no other mechanisms.

FCS_SSH_EXT.1.7

The TSF shall use *SSH KDF* as defined in [**selection**:

- *RFC 4253 (Section 7.2)*,
- *RFC 5656 (Section 4)*

] to derive the following cryptographic keys from a shared secret: *session keys*.

Application Note: RFC 4253 must be selected when the key establishment scheme (selected in FCS_SSH_EXT.1.6) uses finite field cryptography (FFC) and RFC 5656 when it uses elliptic curve cryptography (ECC).

RFC 4253 section 7.2 defines two KDFs for FFC based key establishment schemes. Therefore RFC 4253 should be selected if any of the RFC 4253 or RFC 8268 key establishment schemes are selected.

RFC 5656 section 4 defines KDFs used in ECC key establishment schemes and should be selected when RFC 5656 or RFC 8731 key establishment schemes are selected.

FCS_SSH_EXT.1.8

The TSF shall ensure that [**selection**:

- *a rekey of the session keys*,
- *connection termination*

] occurs when any of the following thresholds are met:

- one hour connection time
- no more than one gigabyte of transmitted data, or
- no more than one gigabyte of received data.

Application Note: This SFR defines three thresholds that need to be implemented. These thresholds were arrived at to ensure that the cryptographic key space for the symmetric session keys isn’t exhausted (more detail can be found in RFC 4344 and RFC 4253). A rekey or connection termination needs to be performed whenever a threshold is reached for a given connection. The rekey applies to all session keys (encryption, integrity protection) for incoming and outgoing traffic.

It is acceptable for a TOE to implement lower thresholds than the maximum values defined in the SFR. If a threshold is configurable, the guidance documentation needs to specify how to configure that threshold.

It is possible that hardware limitation may prevent reaching data transfer threshold in less than one hour. In cases where data transfer threshold could not be reached due to hardware limitations it is acceptable to omit testing of this (SSH rekeying based on data transfer threshold). See Evaluation Activities for details.

Evaluation Activities

FCS_SSH_EXT.1.1

TSS

The evaluator shall ensure that the selections indicated in the ST are consistent with selections in this and subsequent components. Otherwise, this SFR is evaluated by activities for other SFRs.

Guidance

There are no guidance evaluation activities for this component. This SFR is evaluated by activities for other SFRs

Tests

There are no test evaluation activities for this component. This SFR is evaluated by activities for other SFRs

FCS_SSH_EXT.1.2

TSS

The evaluator shall check to ensure that the authentication methods listed in the TSS are identical to those listed in this SFR component; and, ensure if password-based authentication methods have been selected in the ST then these are also described; and, ensure that if keyboard-interactive is selected, it describes the multifactor authentication mechanisms provided by the TOE.

Guidance

The evaluator shall check the guidance documentation to ensure the configuration options, if any, for authentication mechanisms provided by the TOE are described.

Tests

- ***Test 1:*** *[conditional] If the TOE is acting as SSH Server:*
    a. *The evaluator shall use a suitable SSH Client to connect to the TOE, enable debug messages in the SSH Client, and examine the debug messages to determine that only the configured authentication methods for the TOE were offered by the server.*
    b. *[conditional] If the SSH server supports X509 based Client authentication options:*
        a. *The evaluator shall initiate an SSH session from a client where the username is associated with the X509 certificate. The evaluator shall verify the session is successfully established.*
        b. *Next the evaluator shall use the same X509 certificate as above but include a username not associated with the certificate. The evaluator shall verify that the session does not establish.*
        c. *Finally, the evaluator shall use the correct username (from step a above) but use a different X509 certificate which is not associated with the username. The evaluator shall verify that the session does not establish.*
- ***Test 2:*** *[conditional] If the TOE is acting as SSH Client, the evaluator shall test for a successful configuration setting of each authentication method as follows:*
    a. *The evaluator shall initiate a SSH session using the authentication method configured and verify that the session is successfully established.*
    b. *Next, the evaluator shall use bad authentication data (e.g. incorrectly generated certificate or incorrect password) and ensure that the connection is rejected.*

  Steps a-b shall be repeated for each independently configurable authentication method supported by the server.
- ***Test 3:*** *[conditional] If the TOE is acting as SSH Client, the evaluator shall verify that the connection fails upon configuration mismatch as follows:*
    a. *The evaluator shall configure the Client with an authentication method not supported by the Server.*
    b. *The evaluator shall verify that the connection fails.*

If the Client supports only one authentication method, the evaluator can test this failure of connection by configuring the Server with an authentication method not supported by the Client. In order to facilitate this test, it is acceptable for the evaluator to configure an authentication method that is outside of the selections in the SFR.

FCS_SSH_EXT.1.3

TSS

The evaluator shall check that the TSS describes how "large packets" are detected and handled.

Tests

- ***Test 1:*** *The evaluator shall demonstrate that the TOE accepts the maximum allowed packet size.*
- ***Test 2:*** *This test is performed to verify that the TOE drops packets that are larger than size specified in the component.*
    a. *The evaluator shall establish a successful SSH connection with the peer.*
    b. *Next the evaluator shall craft a packet that is one byte larger than the maximum size specified in this component and send it through the established SSH connection to the TOE.*
    c. *Evaluator shall verify that the packet was dropped by the TOE by reviewing the TOE audit log for a dropped packet audit.*

FCS_SSH_EXT.1.4

TSS

The evaluator will check the description of the implementation of SSH in the TSS to ensure the encryption algorithms supported are specified. The evaluator will check the TSS to ensure that

the encryption algorithms specified are identical to those listed for this component.

Guidance

The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.

Tests

The evaluator shall perform the following tests.

If the TOE can be both a client and a server, these tests must be performed for both roles.

- ***Test 1:*** *The evaluator must ensure that only claimed algorithms and cryptographic primitives are used to establish an SSH connection. To verify this, the evaluator shall establish an SSH connection with a remote endpoint. The evaluator shall capture the traffic exchanged between the TOE and the remote endpoint during protocol negotiation (e.g. using a packet capture tool or information provided by the endpoint, respectively). The evaluator shall verify from the captured traffic that the TOE offers only the algorithms defined in the ST for the TOE for SSH connections. The evaluator shall perform one successful negotiation of an SSH connection and verify that the negotiated algorithms were included in the advertised set. If the evaluator detects that not all algorithms defined in the ST for SSH are advertised by the TOE or the TOE advertises additional algorithms not defined in the ST for SSH, the test shall be regarded as failed.*

  The data collected from the connection above shall be used for verification of the advertised hashing and shared secret establishment algorithms in FCS_SSH_EXT.1.5 and FCS_SSH_EXT.1.6 respectively.
- ***Test 2:*** *For the connection established in Test 1, the evaluator shall terminate the connection and observe that the TOE terminates the connection.*
- ***Test 3:*** *The evaluator shall configure the remote endpoint to only allow a mechanism that is not included in the ST selection. The evaluator shall attempt to connect to the TOE and observe that the attempt fails.*

FCS_SSH_EXT.1.5

TSS

The evaluator will check the description of the implementation of SSH in the TSS to ensure the hashing algorithms supported are specified. The evaluator will check the TSS to ensure that the hashing algorithms specified are identical to those listed for this component.

Guidance

The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.

Tests

- ***Test 1:*** *The evaluator shall use the test data collected in FCS_SSH_EXT.1.4, Test 1 to verify that appropriate mechanisms are advertised.*
- ***Test 2:*** *The evaluator shall configure an SSH peer to allow only a hashing algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection and observe that the connection is rejected.*

FCS_SSH_EXT.1.6

TSS

The evaluator will check the description of the implementation of SSH in the TSS to ensure the shared secret establishment algorithms supported are specified. The evaluator will check the TSS to ensure that the shared secret establishment algorithms specified are identical to those listed for this component.

Guidance

The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.

Tests

- ***Test 1:*** *The evaluator shall use the test data collected in FCS_SSH_EXT.1.4, Test 1 to verify that appropriate mechanisms are advertised.*
- ***Test 2:*** *The evaluator shall configure an SSH peer to allow only a key exchange method that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection and observe that the connection is rejected.*

FCS_SSH_EXT.1.7

TSS

The evaluator will check the description of the implementation of SSH in the TSS to ensure the KDFs supported are specified. The evaluator will check the TSS to ensure that the KDFs specified are identical to those listed for this component.

FCS_SSH_EXT.1.8

TSS

The evaluator shall check the TSS to ensure that if the TOE enforces connection rekey or termination limits lower than the maximum values that these lower limits are identified.

In cases where hardware limitation will prevent reaching data transfer threshold in less than one hour, the evaluator shall check the TSS to ensure it contains:

a. *An argument describing this hardware-based limitation and*
b. *Identification of the hardware components that form the basis of such argument.*

***For example**, if specific Ethernet Controller or Wi-Fi radio chip is the root cause of such limitation, these subsystems shall be identified.*
Guidance
The evaluator shall check the guidance documentation to ensure that if the connection rekey or termination limits are configurable, it contains instructions to the administrator on how to configure the relevant connection rekey or termination limits for the TOE.
Tests
The test harness needs to be configured so that its connection rekey or termination limits are greater than the limits supported by the TOE -- it is expected that the test harness should not be initiating the connection rekey or termination.

- ***Test 1:** Establish an SSH connection. Wait until the identified connection rekey limit is met. Observed that a connection rekey or termination is initiated. This may require traffic to periodically be sent, or connection keep alive to be set, to ensure that the connection is not closed due to an idle timeout.*
- ***Test 2:** Establish an SSH connection. Transmit data from the TOE until the identified connection rekey or termination limit is met. Observe that a connection rekey or termination is initiated.*
- ***Test 3:** Establish an SSH connection. Send data to the TOE until the identified connection rekey limit or termination is met. Observe that a connection rekey or termination is initiated.*

FCS_SSHC_EXT.1 SSH Protocol - Client

This is a selection-based component. Its inclusion depends upon selection from FCS_SSH_EXT.1.1.

FCS_SSHC_EXT.1.1

The TSF shall authenticate its peer (SSH server) using: [**selection**:

- *using a local database by associating each host name with a public key corresponding to the following list: [**selection**:*
  - *ssh-rsa (RFC 4253),*
  - *rsa-sha2-256 (RFC 8332),*
  - *rsa-sha2-512 (RFC 8332),*
  - *ecdsa-sha2-nistp256 (RFC 5656),*
  - *ecdsa-sha2-nistp384 (RFC 5656),*
  - *ecdsa-sha2-nistp521 (RFC 5656),*
  - *ssh-ed25519 (RFC 8709),*
  - *ssh-ed448 (RFC 8709)*

  ] ,
- *a list of trusted certification authorities when the public key is in the following formats: [**selection**:*
  - *x509v3-ecdsa-sha2-nistp256 (RFC 6187),*
  - *x509v3-ecdsa-sha2-nistp384 (RFC 6187),*
  - *x509v3-ecdsa-sha2-nistp521 (RFC 6187),*
  - *x509v3-rsa2048-sha256 (RFC 6187)*

  ]

] as described in RFC 4251 section 4.1.

Application Note: The local database may be implemented using any equivalent local storage mechanism.

Evaluation Activities

FCS_SSHC_EXT.1:
TSS
No activities.
Guidance
The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Tests
The evaluator shall perform the following tests:

- ***Test 1:** [conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall configure the TOE with only a single host name and corresponding public key in the local database. The evaluator shall verify that the*

TOE can successfully connect to the host identified by the host name.

- ***Test 2:*** *[conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall configure the TOE with only a single host name and non-corresponding public key in the local database. The evaluator shall verify that the TOE fails to connect to a host not identified by the host name.*
- ***Test 3:*** *[conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall try to connect to a host not configured in the local database. The evaluator shall verify that the TOE either fails to connect to a host identified by the host name or there is a prompt provided to store the public key in the local database.*
- ***Test 4:*** *[conditional] If using a list of trusted certification authorities, the evaluator shall configure the TOE with only a single trusted certification authority corresponding to the host. The evaluator shall verify that the TOE can successfully connect to the host identified by the host name.*
- ***Test 5:*** *[conditional] If using a list of trusted certification authorities, the evaluator shall configure the TOE with only a single trusted certification authority that does not correspond to the host. The evaluator shall verify that the TOE fails to the host identified by the host name.*

FCS_SSHS_EXT.1 SSH Protocol - Server

This is a selection-based component. Its inclusion depends upon selection from FCS_SSH_EXT.1.1.

FCS_SSHS_EXT.1.1

The TSF shall authenticate itself to its peer (SSH Client) using: [**selection**:

- *ssh-rsa (RFC 4253),*
- *rsa-sha2-256 (RFC 8332),*
- *rsa-sha2-512 (RFC 8332),*
- *ecdsa-sha2-nistp256 (RFC 5656),*
- *ecdsa-sha2-nistp384 (RFC 5656),*
- *ecdsa-sha2-nistp521 (RFC 5656),*
- *x509v3-ecdsa-sha2-nistp256 (RFC 6187),*
- *x509v3-ecdsa-sha2-nistp384 (RFC 6187),*
- *x509v3-ecdsa-sha2-nistp521 (RFC 6187),*
- *x509v3-rsa2048-sha256 (RFC 6187),*
- *ssh-ed25519 (RFC 8709),*
- *ssh-ed448 (RFC 8709)*

].

Application Note: These requirements relate to Server authenticating to the Client. The Client authenticating to the Server is covered in FCS_SSHC_EXT.1.1.

Evaluation Activities

FCS_SSHS_EXT.1:
TSS
No activites.
Guidance
The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Tests
The evaluator shall repeat Test 1 and Test 2 from FCS_SSH_EXT.1.4 for each of the authentication mechanisms supported by the TOE.

Next the evaluator shall configure the remote peer to only allow an authentication mechanism that is not included in the ST selection. The evaluator shall attempt to connect to the TOE and observe that the attempt fails.

Appendix A - Implementation-Dependent Requirements

Implementation-Dependent Requirements are dependent on the TOE implementing a particular function. If the TOE fulfills any of these requirements, the vendor must either add the related SFR or disable the functionality for the evaluated configuration.

Appendix B - Selection Rules

This rules in this appendix define which combinations of selections are considered valid. An ST is considered conforming only if it satisfies all rules.

Appendix C - Acronyms

Acronym	Meaning
Base-PP	Base Protection Profile
CC	Common Criteria
CEM	Common Evaluation Methodology
OE	Operational Environment
PP	Protection Profile
PP-Configuration	Protection Profile Configuration
PP-Module	Protection Profile Module
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification

Appendix D - Bibliography

Identifier	Title
[AppPP]	Protection Profile for Application Software
[GPOSPP]	Protection Profile for General Purpose Operating Systems
[MDMPP]	Protection Profile for Mobile Device Management
[RFC 4251]	The Secure Shell (SSH) Protocol Architecture
[RFC 4252]	The Secure Shell (SSH) Authentication Protocol
[RFC 4253]	The Secure Shell (SSH) Transport Layer Protocol
[RFC 4254]	The Secure Shell (SSH) Connection Protocol
[RFC 4256]	Generic Message Exchange Authentication for the Secure Shell Protocol (SSH)
[RFC 4344]	The Secure Shell (SSH) Transport Layer Encryption Modes
[RFC 5647]	AES Galois Counter Mode for the Secure Shell Transport Layer Protocol
[RFC 5656]	Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer
[RFC 6187]	X.509v3 Certificates for Secure Shell Authentication
[RFC 6668]	SHA-2 Data Integrity Verification for the Secure Shell (SSH) Transport Layer Protocol
[RFC 8268]	More Modular Exponentiation (MODP) Diffie-Hellman (DH) Key Exchange (KEX) Groups for Secure Shell (SSH)
[RFC 8308]	Extension Negotiation in the Secure Shell (SSH) Protocol
[RFC 8332]	Use of RSA Keys with SHA-256 and SHA-512 in the Secure Shell (SSH) Protocol
[RFC 8709]	Ed25519 and Ed448 public key algorithms for the Secure Shell (SSH) protocol
[RFC 8731]	Secure Shell (SSH) Key Exchange Method using Curve25519 and Curve448
[VirtPP]	Protection Profile for Virtualization
[openssh-portable/ PROTOCOL]	OpenSSH's deviations and extensions (1.6 transport: AES-GCM)
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017. • Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017. • Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.