Red Hat Enterprise Linux 5 OpenSSH Client
Cryptographic Module v1.1

FIPS 140-2 Security Policy

Version 1.3

Last Update: 2012-08-28
## Contents

Document History........................................................................................................................................3
1 Cryptographic Module Specification...........................................................................................................4
  1.1 Description of Module .........................................................................................................................4
  1.2 Description of Approved Mode..............................................................................................................4
  1.3 Cryptographic Module Boundary...........................................................................................................5
  1.3.1 Hardware Block Diagram .................................................................................................................6
  1.3.2 Software Block Diagram ...................................................................................................................7
2 Cryptographic Module Ports and Interfaces .............................................................................................7
3 Roles, Services and Authentication .............................................................................................................7
  3.1 Roles.....................................................................................................................................................8
  3.2 Services................................................................................................................................................8
  3.3 Operator Authentication.......................................................................................................................9
  3.4 Mechanism and Strength of Authentication .......................................................................................9
4 Physical Security........................................................................................................................................9
5 Operational Environment..........................................................................................................................9
  5.1 Policies..................................................................................................................................................10
6 Cryptographic Key Management................................................................................................................10
  6.2 Key Zeroization......................................................................................................................................11
  6.3 Random Number Generation .............................................................................................................11
7 Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC) ..................................................12
8 Self Tests ..................................................................................................................................................12
  8.1 Power-Up Tests....................................................................................................................................12
  8.1.1 Software Integrity Test Details ......................................................................................................12
9 Guidance..................................................................................................................................................13
  9.1 Crypto officer and User Guidance .......................................................................................................13
10 Glossary and Abbreviations....................................................................................................................13
11 References.................................................................................................................................................14

© 2012 Red Hat / atsec information security. This document can be reproduced and distributed only whole and intact, including this copyright notice.
## Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date of Change</th>
<th>Author</th>
<th>Changes to Previous Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2009-08-18</td>
<td>SHW - atsec</td>
<td>Initial</td>
</tr>
<tr>
<td>0.2</td>
<td>2009-10-14</td>
<td>SHW - atsec</td>
<td>Draft</td>
</tr>
<tr>
<td>1.0</td>
<td>2009-10-27</td>
<td>SHW – atsec</td>
<td>First release</td>
</tr>
<tr>
<td>1.1</td>
<td>2010-03-15</td>
<td>SHW – atsec</td>
<td>Version update</td>
</tr>
<tr>
<td>1.2</td>
<td>2010-05-18</td>
<td>SHW - atsec</td>
<td>Single User Mode</td>
</tr>
<tr>
<td>1.3</td>
<td>2010-06-18</td>
<td>ACH - atsec</td>
<td>Corrected RSA, and RNG cert. numbers</td>
</tr>
</tbody>
</table>
1 Cryptographic Module Specification

This document is the non-proprietary security policy for the OpenSSL FIPS Object Module, and was prepared as part of the requirements for conformance to Federal Information Processing Standard (FIPS) 140-2, Level 1.

The following section describes the module and how it complies with the FIPS 140-2 standard in each of the required areas.

1.1 Description of Module

The OpenSSH Client module is a software only, security level 1 cryptographic module, running on a multi-chip standalone platform. The module supplies cryptographic support the SSH protocol for the Red Hat Enterprise Linux user space. The RPM version for the validated module is 4.3p2-82.el5.

All cryptographic operations and the module integrity check are performed by the Red Hat Enterprise Linux OpenSSL Cryptographic Module for the OpenSSH module. The files that make up the module are:

For x86_64

```
/usr/bin/.ssh.hmac
/usr/bin/ssh
/usr/share/man/man1/ssh.1.gz

/lib64/.libcrypto.so.0.9.8e.hmac
/lib64/.libcrypto.so.6.hmac
/lib64/.libssl.so.0.9.8e.hmac
/lib64/.libssl.so.6.hmac
/lib64/libcrypto.so.0.9.8e
/lib64/libcrypto.so.6
/lib64/libssl.so.0.9.8e
/lib64/libssl.so.6

/usr/bin/.fipscheck.hmac
/usr/bin/fipscheck
```

For IA64

```
/usr/bin/.ssh.hmac
/usr/bin/ssh
/usr/share/man/man1/ssh.1.gz

/lib/.libcrypto.so.0.9.8e.hmac
/lib/.libcrypto.so.6.hmac
/lib/.libssl.so.0.9.8e.hmac
/lib/.libssl.so.6.hmac
/lib/libcrypto.so.0.9.8e
/lib/libcrypto.so.6
/lib/libssl.so.0.9.8e
/lib/libssl.so.6

/usr/bin/.fipscheck.hmac
/usr/bin/fipscheck
```
The following table shows the overview of the security level for each of the eleven sections of the validation.

<table>
<thead>
<tr>
<th>Security Component</th>
<th>Security Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cryptographic Module Specification</td>
<td>1</td>
</tr>
<tr>
<td>Cryptographic Module Ports and Interfaces</td>
<td>1</td>
</tr>
<tr>
<td>Roles, Services and Authentication</td>
<td>1</td>
</tr>
<tr>
<td>Finite State Model</td>
<td>1</td>
</tr>
<tr>
<td>Physical Security</td>
<td>N/A</td>
</tr>
<tr>
<td>Operational Environment</td>
<td>1</td>
</tr>
<tr>
<td>Cryptographic Key Management</td>
<td>1</td>
</tr>
<tr>
<td>EMI/EMC</td>
<td>1</td>
</tr>
<tr>
<td>Self Tests</td>
<td>1</td>
</tr>
<tr>
<td>Design Assurance</td>
<td>1</td>
</tr>
<tr>
<td>Mitigation of Other Attacks</td>
<td>N/A</td>
</tr>
</tbody>
</table>

*Table 1, Security Level of the Module*

The module has been tested on the following multi-chip standalone platforms:

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Model</th>
<th>O/S &amp; Ver.</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP</td>
<td>HP Integrity Server RX2660</td>
<td>Red Hat Enterprise Linux 5.4 and 5.8 (Single User Mode)</td>
</tr>
<tr>
<td>HP</td>
<td>HP ProLiant Server DL585</td>
<td>Red Hat Enterprise Linux 5.4 and 5.8 (Single User Mode)</td>
</tr>
</tbody>
</table>

*Table 2, Tested Platforms*

1.2 Description of Approved Mode

When in FIPS 140-2 approved mode, the contents of the file /proc/sys/crypto/fips_enabled will be '1'.

In Approved mode the module will support the following Approved and allowed functions/protocols:

- Triple-DES (Certs. #839, #840 and #841)
- AES (Certs. #1160, #1161 and #1162)
- DSA (Certs. #378, #379 and #380)
- RNG (ANSI X9.31) (Certs. #642, #643 and #644)
- HMAC-SHA1, HMAC-SHA256 (Certs. #661, #662 and #663)
- RSA (Certs. #549, #550 and #552)

In Approved mode the module will support the following Non-Approved functions:

- RSA (encrypt, decrypt) *(see caveat below)*
- Diffie-Hellman (key agreement; key establishment methodology) *(see caveat below)*

Note: The Red Hat Enterprise Linux 5 OpenSSH Client Cryptographic Module will use the Red Hat Enterprise
Linux OpenSSL Cryptographic Module (FIPS 140-2 Validation #1320) for standard cryptographic operations and will require that a copy of a FIPS 140-2 level 1 validated version of Red Hat Enterprise Linux OpenSSL Cryptographic Module be installed on the system for the Red Hat Enterprise Linux 5 OpenSSH Client Cryptographic Module to operate in a validated mode.

The Red Hat Enterprise Linux 5 OpenSSH Client Cryptographic Module itself implements the SSHv2 protocol.

The module integrity check is performed by the Red Hat Enterprise Linux OpenSSL Cryptographic Module utility fipscheck. The version is 1.2.0-1.el5, and fipscheck-lib version is 1.2.0-1.el5 HMAC/SHA-256 (Certs #661, #662 and #663 from the Red Hat Enterprise Linux OpenSSL Cryptographic Module.)

CAVEAT: The Module will support the following non-approved functions:
1) RSA (key wrapping; key establishment methodology provides between 80 and 150 bits of encryption strength)
2) Diffie-Hellman (key agreement; key establishment methodology provides between 80 and 192 bits of encryption strength).

1.3 Cryptographic Module Boundary

The physical module boundary is the surface of the case of the test platform. The logical module boundary is depicted in the software block diagram and is embodied by the SSH client application found at /usr/bin/ssh and the OpenSSL shared library module.
1.3.1 Hardware Block Diagram

![Hardware Block Diagram](image)

*Figure 1, Hardware Block Diagram*

1.3.2 Software Block Diagram
2 Cryptographic Module Ports and Interfaces

<table>
<thead>
<tr>
<th>Function</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command In</td>
<td>Keyboard, Network, Configuration File ~/.ssh/config, Command Line Options</td>
</tr>
<tr>
<td>Status Out</td>
<td>Display and Network</td>
</tr>
<tr>
<td>Data In</td>
<td>Keyboard, Configuration File ~/.ssh/known_hosts, Network</td>
</tr>
<tr>
<td>Data Out</td>
<td>Display, Network</td>
</tr>
</tbody>
</table>

Table 3, Ports and Interfaces

3 Roles, Services and Authentication

This section defines the roles, services and authentication mechanisms and methods with respect to the applicable FIPS 140-2 requirements.

3.1 Roles
### 3.2 Services

The module supports services that are available to users in the various roles. All of the services are described in detail in the module’s user documentation. The following table shows the services available to the various roles and the access to cryptographic keys and CSPs resulting from services.

- **R** – The item is read or referenced by the service.
- **W** – The item is written or updated by the service.
- **Z** – The persistent item is zeroized by the service.

All of the ciphers are from the Red Hat Enterprise Linux OpenSSL Cryptographic Module validated cryptographic module. The Red Hat Enterprise Linux 5 OpenSSH Client Cryptographic Module performs SSH v2 functions only, and passes all cryptographic operations to Red Hat Enterprise Linux OpenSSL Cryptographic Module.

<table>
<thead>
<tr>
<th>Service</th>
<th>Category</th>
<th>Function</th>
<th>Role</th>
<th>Cryptographic Keys and CSPs Accessed</th>
<th>Access Type (RWZ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Establish &amp; Maintain SSH Session</td>
<td></td>
<td>Encrypt/Decrypt, Keyed-Hash, Key Exchange, Sign/Verify</td>
<td>User, crypto officer</td>
<td>RSA or DSA Client private key/public key</td>
<td>RWZ</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Server Public Key</td>
<td>RW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DH private and public parameters, Session Encryption and Data Authentication Key s</td>
<td>RW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DRNG Seed and Seed Key</td>
<td>RW</td>
</tr>
</tbody>
</table>
### Table 5, Services

<table>
<thead>
<tr>
<th>Service</th>
<th>Category</th>
<th>Function</th>
<th>Role</th>
<th>Cryptographic Keys and CSPs Accessed</th>
<th>Access Type (RWZ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close SSH Session</td>
<td>None</td>
<td>Zeroize</td>
<td>User, crypto officer</td>
<td>DH private and public parameters, Session Encryption and Data Authentication Keys, DRNG Seed and Seed Key</td>
<td>Z</td>
</tr>
<tr>
<td>Terminate SSH Application</td>
<td>None</td>
<td>Zeroize</td>
<td>User, crypto officer</td>
<td>DH private and public parameters, Session Encryption and Data Authentication Keys, DRNG Seed and Seed Key</td>
<td>Z</td>
</tr>
<tr>
<td>Self-Tests</td>
<td>Self Test (includes Integrity and known answer tests)</td>
<td>Invoked by restarting the module</td>
<td>User, crypto officer</td>
<td>Software Integrity Key</td>
<td>R</td>
</tr>
<tr>
<td>Show Status</td>
<td>Status</td>
<td>Via verbose mode and exit codes</td>
<td>User, crypto officer</td>
<td>None</td>
<td>N/A</td>
</tr>
</tbody>
</table>

#### 3.3 Operator Authentication

There is no operator authentication, the role is implicit by action.

#### 3.4 Mechanism and Strength of Authentication

No authentication is required at security level 1, authentication is implicit by assumption of the role.

#### 4 Physical Security

This is a level one software module with no physical security.
5 Operational Environment

This module will operate in a modifiable operational environment per the FIPS 140-2 definition.

5.1 Policies

The operating system shall be restricted to a single operator mode of operation (i.e., concurrent operators are explicitly excluded).

The operator that makes use of the cryptographic module is the single user.

In the FIPS approved mode the ptrace(2) system call, the debugger (gdb(1)) and strace(1) shall not be used.

6 Cryptographic Key Management

The following table identifies the Cryptographic Keys and Critical Security Parameters (CSPs) used within the module. Cryptographic keys and CSPs are never output from the module in plaintext. An Approved key generation method is used to generate keys that are generated by the module via OpenSSL.

6.1 Key life cycle table:

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Generation</th>
<th>Establishment</th>
<th>Access by Service</th>
<th>Entry and output method</th>
<th>Storage</th>
<th>Zeroization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Private Keys</td>
<td>DSA or RSA keys</td>
<td>N/A</td>
<td>N/A</td>
<td>Establish &amp; Maintain SSH Session</td>
<td>N/A</td>
<td>Plaintext</td>
<td>Immediately after use</td>
</tr>
<tr>
<td>Client Public Keys</td>
<td>DSA or RSA keys</td>
<td>N/A</td>
<td>N/A</td>
<td>Establish &amp; Maintain SSH Session</td>
<td>Exported</td>
<td>Plaintext</td>
<td>N/A</td>
</tr>
<tr>
<td>(not a CSP)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Server Public Key</td>
<td>DSA or RSA key</td>
<td>N/A</td>
<td>N/A</td>
<td>Establish &amp; Maintain SSH Session</td>
<td>Imported</td>
<td>Plaintext</td>
<td>N/A</td>
</tr>
<tr>
<td>Key</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Session Data</td>
<td>HMAC SHA-1</td>
<td>N/A</td>
<td>Established during the SSH handshake through DH</td>
<td>Establish &amp; Maintain SSH Session</td>
<td>N/A</td>
<td>Ephemeral</td>
<td>Close SSH Session or Terminate SSH Application</td>
</tr>
<tr>
<td>Authentication Keys</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Session Encryption</td>
<td>AES or Triple-DES</td>
<td>N/A</td>
<td>Established during the SSH handshake through DH</td>
<td>Establish &amp; Maintain SSH Session</td>
<td>N/A</td>
<td>Ephemeral</td>
<td>Close SSH Session or Terminate SSH Application</td>
</tr>
<tr>
<td>Keys</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Integrity</td>
<td>HMAC SHA-256</td>
<td>N/A</td>
<td>N/A</td>
<td>Self-Tests</td>
<td>N/A</td>
<td>Plaintext within the OpenSSL and fipscheck libraries</td>
<td>Close SSH Session or Terminate SSH Application</td>
</tr>
<tr>
<td>Key</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The module ships without containing any keys and CSPs. When the module is configured, the crypto officer generates a DSA or RSA private/public key pair that is stored in plaintext form in keystore files in the filesystem.

A crypto officer or user adds server public keys to the ~/.ssh/known_hosts file using the ssh-keyscan utility or by manually adding the public keys. If the ssh_keyscan utility is used, the crypto officer or user must verify the keys are correct to prevent a man-in-the-middle attack.

The public key is associated with the correct entity as each key is associated with its relevant hostname, or IPv4 or IPv6 address as a lookup index. Moreover using an incorrect key results in failure to establish a valid session as the corresponding private key cannot correctly authenticate against an incorrect public key.

The only key management operations during initial configuration include generating the client public-private key pair and storing client public keys in the ~/.ssh/, which are out of scope for this validation. At runtime, public keys may be added, removed or updated from the ~/.ssh/known_hosts file or a new client public-private key pair may be generated and deployed as needed.

Diffie-Hellman key agreement transpires at the beginning of a session and with sessions after each 1 GB of data transfer or 1 hour of operation, whichever occurs first.

Persistently stored secret and private keys are out of scope, but may be zeroized using the a FIPS140-2 approved mechanism to clear data on hard disks.

### 6.2 Key Zeroization

For volatile memory, memset is included in deallocation operations. There are no restrictions when zeroizing any cryptographic keys and CSPs.

### 6.3 Random Number Generation

A FIPS 140-2, ANSI X9.31 approved pseudo random number generation mechanism will be used in the module, called from OpenSSL, which is seeded by the kernel.

The kernel uses /dev/urandom as a source of random numbers for RNG seeds. The Linux kernel initializes this
pseudo device at system startup.

The kernel performs continual tests on the random numbers it uses to ensure that the seed and seed key input to the Approved RNG do not have the same value. The kernel also performs continual tests on the output of the approved RNG to ensure that consecutive random numbers do not repeat.

7 Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)

Product Name and Model: HP ProLiant Server DL585 Series
Regulatory Model Number: HSTNS-1025
Product Options: All
Conforms to the following Product Specifications and Regulations:
EMC: Class A
CISPR 22:2005
EN 55022:2006
EN 61000-3-2:2006

Product Name and Model: HP Integrity Server rx2660
Regulatory Model Number: RSVLA-0503
Product Options: All
Conforms to the following Product Specifications and Regulations:
EMC: Class A
EN 61000-3-2:2000
EN 61000-3-3:1995

8 Self Tests

FIPS 140-2 requires that the module perform self tests to ensure the integrity of the module and the correctness of the cryptographic functionality at start up. In addition some functions require continuous verification of function, such as the random number generator. All of these tests are listed and described in this section.

8.1 Power-Up Tests

Software Integrity Test. All cryptographic function tests are performed by the Red Hat Enterprise Linux OpenSSL module before it will perform cryptographic operations for the OpenSSH Client module.

8.1.1 Software Integrity Test Details

OpenSSH userspace modules have their integrity verified at startup by the software integrity test.

The integrity check is performed by the Red Hat Enterprise Linux OpenSSL module utility fipscheck. The version is 1.2.0-1.el5, and fipscheck-lib version is 1.2.0-1.el5 HMAC/SHA-256.

When the module starts, it exercises the power-on self-test including the software integrity test. The software integrity test (HMAC-SHA256) constitutes a known answer test for the HMAC-SHA256 algorithm.

The user space integrity verification is performed as follows:

The OpenSSH client application links with the library libfipscheck.so which is intended to execute fipscheck to verify the integrity of the calling application file using HMAC SHA-256. Upon calling the
FIPSCHECK_verify() function provided with libfipscheck.so, the fipscheck application is loaded and executed, and the following steps are performed.

- OpenSSL as loaded by fipscheck performs the integrity check of the OpenSSL library files using HMAC SHA-256.
- The application fipscheck performs the integrity check of its application file using HMAC SHA-256 provided by OpenSSL.
- The fipscheck application performs the integrity check of the calling application. The fipscheck computes the HMAC-SHA-256 checksum of the file from the command line and compares the computed value to the value stored inside the /path/to/application/.<applicationfilename>.hmac checksum file. The fipscheck application returns the appropriate exit value based on the comparison result (zero if the checksum is OK – which is enforced by the libfipscheck.so library).

9 Guidance

NOTE: All cryptographic functions for the Red Hat Enterprise Linux 5 OpenSSH Client Cryptographic Module will be provided by a copy of a FIPS 140-2 validated version of the Red Hat OpenSSL cryptographic module.

9.1 Crypto Officer and User Guidance

The version of the RPM containing the validated module is stated in section 1 above. The integrity of the RPM is automatically verified during the installation and the crypto officer shall not install the RPM file if the RPM tool indicates an integrity error.

The RPM package of the module can be installed by standard tools recommended for the installation of RPM packages on a Red Hat Enterprise Linux system (for example, yum, rpm, and the RHN remote management tool).

For proper operation of the in-module integrity verification, the prelink has to be disabled. This can be done by setting PRELINKING=no in the /etc/sysconfig/prelink configuration file.

To bring the module into FIPS mode, the crypto officer has to regenerate the initrd by using the following command:

For the x86_64 platform, the command is:

```
mkinitrd --with-fips -f /boot/initrd-$(uname -r).img $(uname -r)
```

For the IA64, the command is:

```
mkinitrd --with-fips -f /boot/efi/efi/redhat/initrd-$(uname -r).img $(uname -r)
```

After regenerating the initrd, the crypto officer has to append the following string to the kernel command line by changing the setting in the boot loader:

```
fips=1
```

In addition to the configuration of the kernel, the OpenSSH client configuration ~/.ssh/config should contain:

- Either no "Ciphers" option or the option with a subset out of "aes128 ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc";
- Either no "MACs" option or the option with "hmac-sha1";
- "Protocol 2" must be specified.
10 Glossary and Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>Advanced Encryption Specification</td>
</tr>
<tr>
<td>CAVP</td>
<td>Cryptographic Algorithm Validation Program</td>
</tr>
<tr>
<td>CBC</td>
<td>Cypher Block Chaining</td>
</tr>
<tr>
<td>CCM</td>
<td>Counter with Cipher Block Chaining-Message Authentication Code</td>
</tr>
<tr>
<td>CFB</td>
<td>Cypher Feedback</td>
</tr>
<tr>
<td>CMT</td>
<td>Cryptographic Module Testing</td>
</tr>
<tr>
<td>CMVP</td>
<td>Cryptographic Module Validation Program</td>
</tr>
<tr>
<td>CSP</td>
<td>Critical Security Parameter</td>
</tr>
<tr>
<td>CVT</td>
<td>Component Verification Testing</td>
</tr>
<tr>
<td>DES</td>
<td>Data Encryption Standard</td>
</tr>
<tr>
<td>DH</td>
<td>Diffie-Hellman</td>
</tr>
<tr>
<td>DSA</td>
<td>Digital Signature Algorithm</td>
</tr>
<tr>
<td>ECB</td>
<td>Electronic Code Book</td>
</tr>
<tr>
<td>FSM</td>
<td>Finite State Model</td>
</tr>
<tr>
<td>HMAC</td>
<td>Hash Message Authentication Code</td>
</tr>
<tr>
<td>LDAP</td>
<td>Lightweight Directory Application Protocol</td>
</tr>
<tr>
<td>MAC</td>
<td>Message Authentication Code</td>
</tr>
<tr>
<td>NIST</td>
<td>National Institute of Science and Technology</td>
</tr>
<tr>
<td>NVLAP</td>
<td>National Voluntary Laboratory Accreditation Program</td>
</tr>
<tr>
<td>OFB</td>
<td>Output Feedback</td>
</tr>
<tr>
<td>O/S</td>
<td>Operating System</td>
</tr>
<tr>
<td>RNG</td>
<td>Random Number Generator</td>
</tr>
<tr>
<td>RSA</td>
<td>Rivest, Shamir, Addleman</td>
</tr>
<tr>
<td>SAP</td>
<td>Service Access Points</td>
</tr>
<tr>
<td>SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>SHA</td>
<td>Secure Hash Algorithm</td>
</tr>
<tr>
<td>SHS</td>
<td>Secure Hash Standard</td>
</tr>
<tr>
<td>SOF</td>
<td>Strength of Function</td>
</tr>
<tr>
<td>SSH</td>
<td>Secure Shell</td>
</tr>
<tr>
<td>TDES</td>
<td>Triple DES</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
</tbody>
</table>

Table 7, Abbreviations
11 References

[1] OpenSSH Client user guide (provided with installation RPM, see section 1.1 Description of Module for version)