Network Device Interpretation # 201669

SSH Server Encryption Algorithms

Status: ☒ *Active* □ *Inactive*

Date: *21-Feb-2017*

Type of Document: ☒ *Technical Decision* □ *Technical Recommendation*

Approved by: ☒ *Network iTC Interpretations Team* ☒ *Network iTC*

Affected Document(s): *NDcPP V1.0, FWcPP V1.0*

Affected Section(s): *FCS_SSHC_EXT.1.4, FCS_SSHS_EXT.1.4*

Superseded Interpretation(s): *None*

Issue:

FCS_SSHC_EXT.1.4/FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: aes128-cbc, aes256-cbc, [selection: AEAD_AES_128_GCM, AEAD_AES_256_GCM, no other algorithms].

Issue

According to the SFR, compliant TOEs should offer mandatory support aes128-cbc, aes256-cbc, and select: AEAD_AES_128_GCM, AEAD_AES_256_GCM, no other algorithms. The SFR also requires the TOE to reject all other encryption algorithms.

A. Many vendors have disabled the use of AES-CBC algorithms over SSH due to concerns that their products will be flagged as vulnerable when scanned with common network vulnerability scanning tools.

Many tools flag SSH servers offering AES-CBC as a security risk due to a legacy vulnerability regarding SSH with AES-CBC encryption (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-5161, http://www-01.ibm.com/support/docview.wss?uid=swg21971424, https://access.redhat.com/solutions/420283)

There are several examples of vendors running into problems when their products offer the algorithms required by the SFR, despite the fact that the original CVE has been remediated in the currently implemented version of SSH:

- *http://www.accella.net/knowledgebase/ask-the-sysadmin-fixing-cipher-and-mac-ssh-security-problems/*
- *http://community.arubanetworks.com/t5/Wireless-Access/SSH-and-AES-CBC/td-p/248919*

B. In Version 1.1 draft of the collaborative Protection Profile for Network Devices, FCS_SSHC_EXT.1.4/FCS_SSHS_EXT.1.4 do not list the AES-CBC algorithms as mandatory, and includes them as selectable algorithms together with aes128-ctr, aes256-ctr.

Resolution

A technical decision on the following inquiries is requested:

a. In light of the future cPP support for AES-128-CTR, AES-196-CTR and AES-256-CTR selections, would a TOE be permitted to offer support for those encryption algorithms and meet the SFR?

b. In light of the future cPP version making all SSH encryption algorithms selectable (no mandatory algorithm support), could a TOE exclude support for AES-128-CBC and AES-256-CBC and meet the SFR?

Resolution:

The NIT acknowledges the security related concerns regarding AES-CBC mode and supports making AES-128-CBC and AES-256-CBC optional. It is outside of the scope of the NIT to add cryptographic algorithms to existing SFRs, though. FCS_SSHC_EXT.1.4 and FCS_SSHS_EXT.1.4 shall therefore be modified as follows:

"The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: *[selection: aes128-cbc, aes256-cbc, AEAD_AES_128_GCM, AEAD_AES_256_GCM]*."

The corresponding application notes shall be modified as follows:

"RFC 5647 specifies the use of the AEAD_AES_128_GCM and AEAD_AES_256_GCM algorithms in SSH. As described in RFC 5647, AEAD_AES_128_GCM and AEAD_AES_256_GCM can only be chosen as encryption algorithms when the same algorithm is being used as the MAC algorithm. Corresponding FCS_COP entries are included in the ST for the algorithms selected here."

Rationale:

As stated in the 'Issue' section.

Further Action:

None

Action by Network iTC:

None