Cyber Security Services

Security Penetration Testing

Penetration Testing report for AIVMPT-5244 DMIB - OCP Migration - 3.11 to 4.10

Version 1.0

Date of Issue: 08-Dec-2022

CONFIDENTIAL

Document Control

Title	Penetration Testing report for AIVMPT-5244 DMIB - OCP Migration - 3.11 to 4.10
Version	1.0
Date of issue	08-Dec-2022
Document Author(s)	**Ireneusz Wolny**
Document Contributor(s)	N/A
Document Reviewer(s) / Approved by	**Vidya Baduru**
Document Owner	Krystian Szybis
Repository (link where it is stored)	https://jira.global.standardchartered.com/browse/AIVMPT-5244

Document Amendment Details / Revision History

Version Number	Revision Date	Sections affected / Change summary	Author
0.1	07-Dec-2022	New	**Ireneusz Wolny**
0.2	08-Dec-2022	Review	**Vidya Baduru**
1.0	08-Dec-2022	Final	**Ireneusz Wolny**

Distribution List

Sl. No.	Distributed to	Remarks (if any)
1	Vishal Gaurav Vishal.Gaurav1@sc.com	**JIRA requestor**
2	Pentest.Communication@sc.com Klaudia Kaminska Klaudia.Kaminska@sc.com	**SPOC**
3	A, Devanand Devanand.Appar@sc.com Singaraj, Rathakrishnan	**Allowed Users**

Sl. No.	Distributed to	Remarks (if any)
	Rathakrishnan.Singaraj@sc.com E, Jayakumar Jayakumar.E@sc.com P, Karthikeyan Karthikeyan.P@sc.com	
4	Carvalho, Vanessa Jane Vanessajane.Carvalho@sc.com Liniado, Dor Dor.Liniado@sc.com Tan, Mathew Mathew.Tan@sc.com Abkhalim, Natasha NatashaBinti.ABKhalim@sc.com	**Business Users**

Table of Contents

1 Executive Summary

The Security Penetration Testing team was tasked with conducting a penetration test of DMIB infrastructure. All activities were conducted in a manner that simulates a malicious actor engaged in a targeted attack against targets defined in scope section. During engagement 3 vulnerabilities have been detected, including 3 low vulnerabilities.

Detected vulnerabilities may allow to:

- Decrypt network traffic due to usage of weak TLS protocols and ciphers.
- Decrypt network traffic due to usage of weak SSH protocols and ciphers.

The assessment was conducted with the level of access that a SCB employee would have and according to the Standard Chartered Penetration Testing Standard and the Penetration Testing Methodology.

Highest Rated Findings

Low	SSL Issues - Weak Cypher Suites Supported *Server supports weak cipher suites making it prone to MITM attacks and not compliant with SCB Cryptography Standards.*
Low	SSL Issues - Missing Server-side Order of Cypher Suites *The server does not have cipher suites ordering, making it easier to break the encryption of TLS channel by not negotiating the best available ciphersuite.*
Low	SSH Configuration Weaknesses *By abusing weak algorithms, an attacker may recover the plaintext message from the ciphertext.*

Recommendations Summary

Project needs to implement at least following recommendations to increase overall security posture:

- Reconfigure TLS profile to support strong TLS cipher suites and follow SCB cryptography standard.
- Disable deprecated SSH algorithms.

All findings should be reviewed and fixed, after a fix implementation it is recommended to perform retest assessment to confirm full remediation.

2 Scope of Work

The following has been confirmed prior to Penetration Testing as Statement of Work:

- Type of Assessment: Infrastructure
- Testing Information Provided: Black-Box
- Environment: Preprod
- Target Address:
  - HKLVATAPQ310.hk.standardchartered.com 10.7.29.144
  - HKLVATAPQ311.hk.standardchartered.com 10.7.29.145
  - HKLVATAPQ312.hk.standardchartered.com 10.7.29.146
  - HKLVATAPQ313.hk.standardchartered.com 10.7.29.147
  - HKLVATAPQ314.hk.standardchartered.com 10.7.29.148
  - HKLVATAPQ315.hk.standardchartered.com 10.7.29.149
  - HKLVATAPQ316.hk.standardchartered.com 10.7.29.150
  - HKLVATAPQ317.hk.standardchartered.com 10.7.29.151
  - HKLVATAPQ318.hk.standardchartered.com 10.7.29.152
  - HKLVATAPQ319.hk.standardchartered.com 10.7.29.153
  - HKLVATAPQ320.hk.standardchartered.com 10.7.29.154
  - HKLVATAPQ321.hk.standardchartered.com 10.7.29.155
  - HKLVATAPQ322.hk.standardchartered.com 10.7.29.156
  - HKLVATAPQ323.hk.standardchartered.com 10.7.29.157
  - HKLVATAPQ324.hk.standardchartered.com 10.7.29.158
  - HKLVATAPQ325.hk.standardchartered.com 10.7.29.159
- Testing Duration: 8 days (from 2022-11-28 to 2022-12-7)
- Testing hours: Any time.
- Exclusions: Denial of Service (network based)
- Man-days: 8

Test Objectives

The objective of the penetration test was to enable the Standard Chartered Bank to better understand the current IT security risk profile of the DMIB infrastructure and to provide recommendations to help reduce any identified risks before the servers are placed in a production environment. This penetration test was designed to replicate the position of an unauthenticated user of the DMIB with the intention of gaining access to the customers data.

Methodology

The Penetration Testing Methodology version used for delivery of this penetration testing assessment is available at:
https://confluence.global.standardchartered.com/display/AIVM/Penetration+Testing+Methodology

Detailed Methodology execution checklist can be found in section 5 “Methodology execution checklist”.

3 Findings Summary

Ref	Rating	Title
4.1	**3.1**	Low: SSL Issues - Weak Cypher Suites Supported
4.2	**3.1**	Low: SSL Issues - Missing Server-side Order of Cypher Suites
4.3	**3.1**	Low: SSH Configuration Weaknesses

4 Detailed findings

4.1. SSL Issues - Weak Cypher Suites Supported

Status: Open	Category: A2	Rating: **Low – <u>3.1</u>** https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:A/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Description

The server supports weak or obsolete cipher suites in TLS negotiation, which are considered to be insecure.

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
- TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_128_CCM
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_256_CCM
- TLS_RSA_WITH_AES_256_GCM_SHA384

Attack scenario

In case when one of weak or vulnerable ciphers are negotiated during TLS handshake, a suitably positioned attacker could capture the TLS network traffic for later decryption due to weak encryption algorithm used in communication.

Affected hosts

10.7.29.144 / 1936 / tcp

10.7.29.144 / 9001 / tcp

10.7.29.145 / 1936 / tcp

10.7.29.145 / 9001 / tcp

10.7.29.146 / 1936 / tcp

10.7.29.146 / 9001 / tcp

10.7.29.147 / 1936 / tcp

10.7.29.147 / 9001 / tcp

10.7.29.148 / 9001 / tcp

10.7.29.148 / 9099 / tcp

10.7.29.148 / 9641 / tcp

10.7.29.148 / 9642 / tcp

10.7.29.148 / 9643 / tcp

10.7.29.148 / 9644 / tcp

10.7.29.148 / 9979 / tcp

10.7.29.149 / 9001 / tcp

10.7.29.149 / 9641 / tcp

10.7.29.149 / 9642 / tcp

10.7.29.149 / 9643 / tcp

10.7.29.149 / 9644 / tcp

10.7.29.149 / 9979 / tcp

10.7.29.149 / 17697 / tcp

10.7.29.149 / 22623 / tcp

10.7.29.150 / 9001 / tcp

10.7.29.150 / 9641 / tcp

10.7.29.150 / 9642 / tcp

10.7.29.150 / 9643 / tcp

10.7.29.150 / 9644 / tcp

10.7.29.150 / 9979 / tcp

10.7.29.151 / 9001 / tcp

10.7.29.151 / 9641 / tcp

10.7.29.151 / 9642 / tcp

10.7.29.151 / 9643 / tcp

10.7.29.151 / 9644 / tcp

10.7.29.151 / 9979 / tcp

10.7.29.151 / 17697 / tcp

10.7.29.151 / 22623 / tcp

10.7.29.152 / 9642 / tcp

10.7.29.152 / 9643 / tcp

10.7.29.152 / 9644 / tcp

10.7.29.152 / 9979 / tcp

10.7.29.152 / 17697 / tcp

10.7.29.152 / 22623 / tcp

10.7.29.153 / 9644 / tcp

10.7.29.154 / 9001 / tcp

10.7.29.157 / 9001 / tcp

10.7.29.158 / 9001 / tcp

10.7.29.159 / 9001 / tcp

Recommendations

Disable weak cipher suites highlighted in description section.

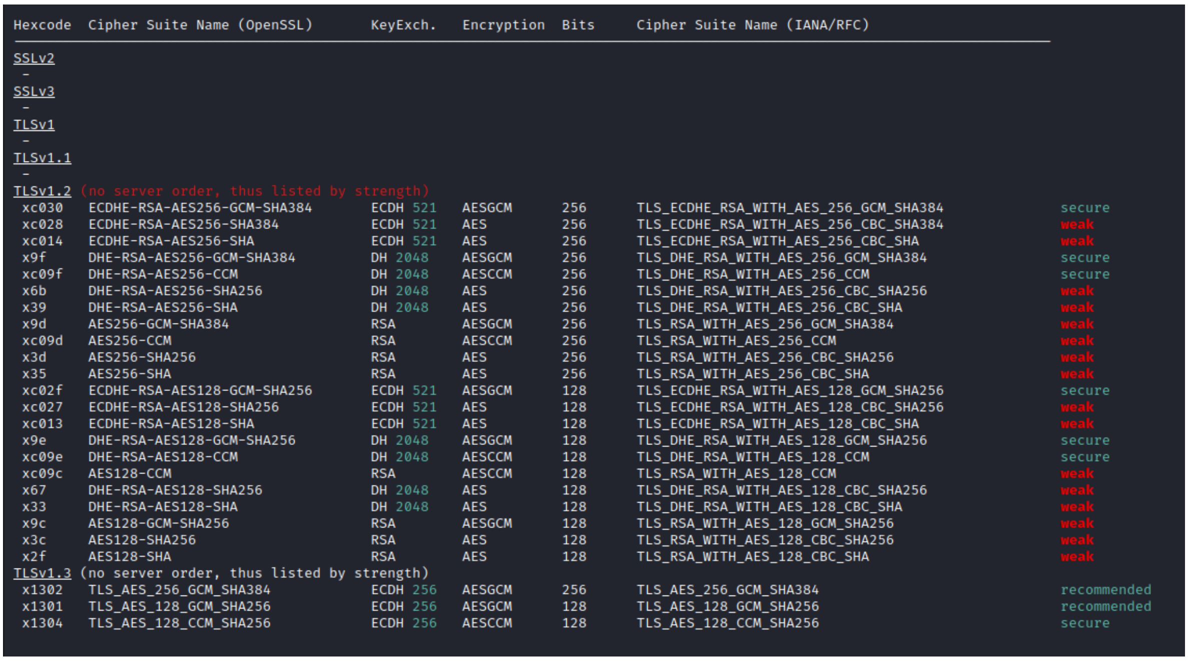
References

- SCB Cryptography Standard: https://rv2.global.standardchartered.com/govpoint-ui/#/govpoint/viewDocument?documentNumber=STD00028
- CWE-327: https://cwe.mitre.org/data/definitions/327.html
- CWE-326: https://cwe.mitre.org/data/definitions/326.html

Evidence

The below screenshot is the output example of testssl tool run against 10.7.29.148:9641.

The affected ports share the similar testssl output.



```
Hexcode  Cipher Suite Name (OpenSSL)       KeyExch.   Encryption  Bits     Cipher Suite Name (IANA/RFC)
-----------------------------------------------------------------------------------------------------------------------------
SSLv2
 -
SSLv3
 -
TLSv1
 -
TLSv1.1
 -
TLSv1.2 (no server order, thus listed by strength)
 xc030   ECDHE-RSA-AES256-GCM-SHA384       ECDH 521   AESGCM      256      TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384              secure
 xc028   ECDHE-RSA-AES256-SHA384           ECDH 521   AES         256      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384              weak
 xc014   ECDHE-RSA-AES256-SHA              ECDH 521   AES         256      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA                 weak
 x9f     DHE-RSA-AES256-GCM-SHA384         DH 2048    AESGCM      256      TLS_DHE_RSA_WITH_AES_256_GCM_SHA384                secure
 xc09f   DHE-RSA-AES256-CCM                DH 2048    AESCCM      256      TLS_DHE_RSA_WITH_AES_256_CCM                       secure
 x6b     DHE-RSA-AES256-SHA256             DH 2048    AES         256      TLS_DHE_RSA_WITH_AES_256_CBC_SHA256                weak
 x39     DHE-RSA-AES256-SHA                DH 2048    AES         256      TLS_DHE_RSA_WITH_AES_256_CBC_SHA                   weak
 x9d     AES256-GCM-SHA384                 RSA        AESGCM      256      TLS_RSA_WITH_AES_256_GCM_SHA384                    weak
 xc09d   AES256-CCM                        RSA        AESCCM      256      TLS_RSA_WITH_AES_256_CCM                           weak
 x3d     AES256-SHA256                     RSA        AES         256      TLS_RSA_WITH_AES_256_CBC_SHA256                    weak
 x35     AES256-SHA                        RSA        AES         256      TLS_RSA_WITH_AES_256_CBC_SHA                       weak
 xc02f   ECDHE-RSA-AES128-GCM-SHA256       ECDH 521   AESGCM      128      TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256              secure
 xc027   ECDHE-RSA-AES128-SHA256           ECDH 521   AES         128      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256              weak
 xc013   ECDHE-RSA-AES128-SHA              ECDH 521   AES         128      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA                 weak
 x9e     DHE-RSA-AES128-GCM-SHA256         DH 2048    AESGCM      128      TLS_DHE_RSA_WITH_AES_128_GCM_SHA256                secure
 xc09e   DHE-RSA-AES128-CCM                DH 2048    AESCCM      128      TLS_DHE_RSA_WITH_AES_128_CCM                       secure
 xc09c   AES128-CCM                        RSA        AESCCM      128      TLS_RSA_WITH_AES_128_CCM                           weak
 x67     DHE-RSA-AES128-SHA256             DH 2048    AES         128      TLS_DHE_RSA_WITH_AES_128_CBC_SHA256                weak
 x33     DHE-RSA-AES128-SHA                DH 2048    AES         128      TLS_DHE_RSA_WITH_AES_128_CBC_SHA                   weak
 x9c     AES128-GCM-SHA256                 RSA        AESGCM      128      TLS_RSA_WITH_AES_128_GCM_SHA256                    weak
 x3c     AES128-SHA256                     RSA        AES         128      TLS_RSA_WITH_AES_128_CBC_SHA256                    weak
 x2f     AES128-SHA                        RSA        AES         128      TLS_RSA_WITH_AES_128_CBC_SHA                       weak
TLSv1.3 (no server order, thus listed by strength)
 x1302   TLS_AES_256_GCM_SHA384            ECDH 256   AESGCM      256      TLS_AES_256_GCM_SHA384                             recommended
 x1301   TLS_AES_128_GCM_SHA256            ECDH 256   AESGCM      128      TLS_AES_128_GCM_SHA256                             recommended
 x1304   TLS_AES_128_CCM_SHA256            ECDH 256   AESCCM      128      TLS_AES_128_CCM_SHA256                             secure
```

4.2. SSL Issues - Missing Server-side Order of Cypher Suites

Status: Open	Category: A2	Rating: **Low – 3.1** https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:A/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Description

The server does not present cipher suites order preference during TLS negotiation, which may impact encryption strength and security of established TLS channel.

Attack scenario

If it happens that implementation of client or server will choose a weak cipher suite and the attacker positioned in the same network will be able to capture the encrypted traffic, it might be easier for them to break the encryption as not the strongest cipher suite was chosen.

Affected hosts

10.7.29.148 / 2379 / tcp

10.7.29.148 / 2380 / tcp

10.7.29.148 / 9641 / tcp

10.7.29.148 / 9642 / tcp

10.7.29.148 / 9643 / tcp

10.7.29.148 / 9644 / tcp

10.7.29.148 / 9978 / tcp

10.7.29.148 / 9979 / tcp

10.7.29.149 / 2379 / tcp

10.7.29.149 / 2380 / tcp

10.7.29.149 / 9641 / tcp

10.7.29.149 / 9642 / tcp

10.7.29.149 / 9643 / tcp

10.7.29.149 / 9644 / tcp

10.7.29.149 / 9978 / tcp

10.7.29.149 / 9979 / tcp

10.7.29.150 / 2379 / tcp

10.7.29.150 / 2380 / tcp

10.7.29.150 / 9641 / tcp

10.7.29.150 / 9642 / tcp

10.7.29.150 / 9643 / tcp

10.7.29.150 / 9644 / tcp

10.7.29.150 / 9978 / tcp

10.7.29.150 / 9979 / tcp

10.7.29.151 / 2379 / tcp

10.7.29.151 / 2380 / tcp

10.7.29.151 / 9641 / tcp

10.7.29.151 / 9642 / tcp

10.7.29.151 / 9643 / tcp

10.7.29.151 / 9644 / tcp

10.7.29.151 / 9978 / tcp

10.7.29.151 / 9979 / tcp

10.7.29.152 / 9642 / tcp

10.7.29.152 / 9643 / tcp

10.7.29.152 / 9644 / tcp

10.7.29.152 / 9978 / tcp

10.7.29.152 / 9979 / tcp

10.7.29.153 / 9644 / tcp

10.7.29.153 / 9978 / tcp

Recommendations

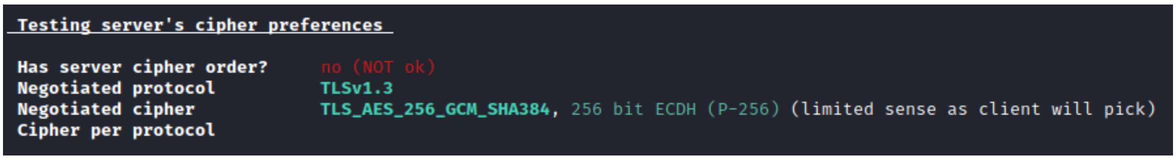
Enable TLS cipher suite ordering on the server.

References

- SCB Cryptography Standard - https://rv2.global.standardchartered.com/govpoint-ui/#/govpoint/viewDocument?documentNumber=STD00028
- CWE-327: https://cwe.mitre.org/data/definitions/327.html

Evidence

The below screenshot is the output example of the testssl tool displaying server side cipher order misconfigurations. The displayed misconfiguration is the same for all affected ports.



```
 Testing server's cipher preferences 

 Has server cipher order?     no (NOT ok)
 Negotiated protocol          TLSv1.3
 Negotiated cipher            TLS_AES_256_GCM_SHA384, 256 bit ECDH (P-256) (limited sense as client will pick)
 Cipher per protocol
```

4.3. SSH Configuration Weaknesses

Status: Open	Category: I2	Rating: **Low – 3.1** https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:A/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Description

The devices support the following weak key exchange algorithms:

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

The devices support the following weak host-key algorithms:

- ecdsa-sha2-nistp256

The devices support the following weak encryption algorithms (ciphers):

- aes256-cbc aes128-cbc

Attack scenario

By abusing weak algorithms, an attacker may recover the plaintext message from the ciphertext.

Affected hosts

10.7.29.144 / 22 / tcp

10.7.29.145 / 22 / tcp

10.7.29.146 / 22 / tcp

10.7.29.147 / 22 / tcp

10.7.29.148 / 22 / tcp

10.7.29.149 / 22 / tcp

10.7.29.150 / 22 / tcp

10.7.29.151 / 22 / tcp

10.7.29.152 / 22 / tcp

10.7.29.153 / 22 / tcp

10.7.29.154 / 22 / tcp

10.7.29.155 / 22 / tcp

10.7.29.156 / 22 / tcp

10.7.29.157 / 22 / tcp

10.7.29.158 / 22 / tcp

10.7.29.159 / 22 / tcp

Recommendations

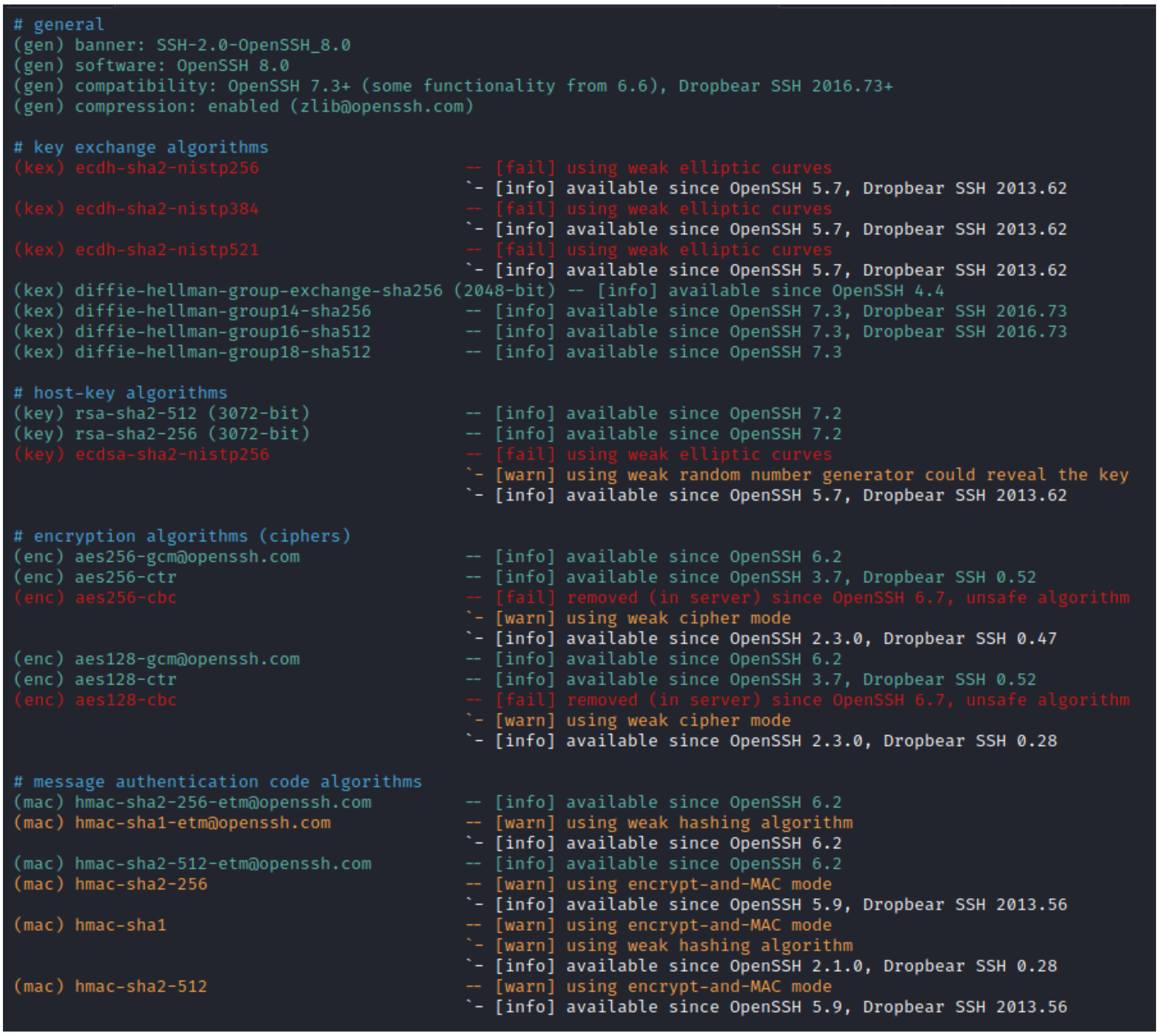
Disable weak MAC and KEX algorithms for the affected SSH services.

References

- SCB Cryptography Standard: https://rv2.global.standardchartered.com/govpoint-ui/#/govpoint/viewDocument?documentNumber=STD00028
- CWE-327: https://cwe.mitre.org/data/definitions/327.html
- CWE-310: https://cwe.mitre.org/data/definitions/310.html
- CWE-326: https://cwe.mitre.org/data/definitions/326.html

Evidence

The example screenshot below presents the weak host-key algorithym. The below setting is present on all affected hosts.



```
# general
(gen) banner: SSH-2.0-OpenSSH_8.0
(gen) software: OpenSSH 8.0
(gen) compatibility: OpenSSH 7.3+ (some functionality from 6.6), Dropbear SSH 2016.73+
(gen) compression: enabled (zlib@openssh.com)

# key exchange algorithms
(kex) ecdh-sha2-nistp256                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) ecdh-sha2-nistp384                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) ecdh-sha2-nistp521                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) diffie-hellman-group-exchange-sha256 (2048-bit) -- [info] available since OpenSSH 4.4
(kex) diffie-hellman-group14-sha256         -- [info] available since OpenSSH 7.3, Dropbear SSH 2016.73
(kex) diffie-hellman-group16-sha512         -- [info] available since OpenSSH 7.3, Dropbear SSH 2016.73
(kex) diffie-hellman-group18-sha512         -- [info] available since OpenSSH 7.3

# host-key algorithms
(key) rsa-sha2-512 (3072-bit)               -- [info] available since OpenSSH 7.2
(key) rsa-sha2-256 (3072-bit)               -- [info] available since OpenSSH 7.2
(key) ecdsa-sha2-nistp256                   -- [fail] using weak elliptic curves
                                            `- [warn] using weak random number generator could reveal the key
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62

# encryption algorithms (ciphers)
(enc) aes256-gcm@openssh.com                -- [info] available since OpenSSH 6.2
(enc) aes256-ctr                            -- [info] available since OpenSSH 3.7, Dropbear SSH 0.52
(enc) aes256-cbc                            -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                            `- [warn] using weak cipher mode
                                            `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.47
(enc) aes128-gcm@openssh.com                -- [info] available since OpenSSH 6.2
(enc) aes128-ctr                            -- [info] available since OpenSSH 3.7, Dropbear SSH 0.52
(enc) aes128-cbc                            -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                            `- [warn] using weak cipher mode
                                            `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.28

# message authentication code algorithms
(mac) hmac-sha2-256-etm@openssh.com         -- [info] available since OpenSSH 6.2
(mac) hmac-sha1-etm@openssh.com             -- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 6.2
(mac) hmac-sha2-512-etm@openssh.com         -- [info] available since OpenSSH 6.2
(mac) hmac-sha2-256                         -- [warn] using encrypt-and-MAC mode
                                            `- [info] available since OpenSSH 5.9, Dropbear SSH 2013.56
(mac) hmac-sha1                             -- [warn] using encrypt-and-MAC mode
                                            `- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 2.1.0, Dropbear SSH 0.28
(mac) hmac-sha2-512                         -- [warn] using encrypt-and-MAC mode
                                            `- [info] available since OpenSSH 5.9, Dropbear SSH 2013.56
```

5 Methodology execution checklist

Infrastructure Penetration Testing Methodology has been fulfilled in **100%**. For potential exclusions from the scope see Limitations section.

☒ Performed □ Not performed ■ Not applicable

Methodology Requirements		Execution Detailed Information
Intelligence Gathering		
☒	Identify Devices Types, Platforms and Operating Systems	
☒	Identify Open Ports and Accessible Services	
☒	Identify Services Version	
■	Identify Metadata	
☒	Enumerate Hosted Application on Web Servers	
☒	Identify Defence Technologies	
☒	Gather Operating Systems, Services, Applications Default Credentials	
Vulnerability Analysis		
☒	Scan Vulnerabilities with Automatic Scanners	
☒	Check Exploit Databases and Framework Modules	
☒	Check Common Misconfigurations	
■	Perform Reverse Engineering on Available Binaries	No custom binaries found.
☒	Perform Fuzzing	
■	Validate Detected Vulnerabilities	No vulnerability found which require validating.
Exploitation		
☒	Test Weak Authentication and Default Credentials Usage	
■	Tailor Exploits	No exploit tailoring required.
■	Exploit Detected Vulnerabilities	No working exploits were executed.
Post Exploitation		
■	Identify Device Misconfiguration, Sensitive Data, Available User Information	No vulnerability was exploited, therefore post-exploitation phase was not conducted.
■	Perform Password Cracking	
■	Identify Weak Encryption Usage	
■	Test Exfiltration Paths	
■	Enumerate Accessible Devices from Exploited Device	
■	Perform Privilege Escalation	
■	Identify Services Available Locally	
Cleanup		

■	Move all Copied/Generated Files from a Device to Evidence Folder	No vulnerability was exploited, therefore cleanup phase was not conducted.
■	Return to Original Values System and Application Settings	
■	Delete all Accounts Created by Pentester	

6 Severity Rating Scale and Vulnerability Categories

Standard Chartered Security Penetration Testing Team uses vanilla Common Vulnerability Scoring System v3.1Metrics and Equations to calculate risk rating.

Rating	CVSS v3.1Score
Critical	9.0 - 10.0
High	7.0 - 8.9
Medium	4.0 - 6.9
Low	0.1 - 3.9
None	0.0

Vulnerability Categories

Application				Operating System	
Broken Access Control	A1	Improper Platform Usage	M1	Weak, Guessable, or Hardcoded Passwords	I1
Cryptographic Failures	A2	Insecure Data Storage	M2	Insecure Network Services	I2
Injection	A3	Insecure Communication	M3	Insecure Ecosystem Interfaces	I3
Insecure Design	A4	Insecure Authentication	M4	Lack of Secure Update Mechanism	I4
Security Misconfiguration	A5	Insufficient Cryptography	M5	Use of Insecure or Outdated Components	I5
Vulnerable and Outdated Components	A6	Insecure Authorization	M6	Insufficient Privacy Protection	I6
Identification and Authentication Failures	A7	Poor Code Quality	M7	Insecure Data Transfer and Storage	I7
Software and Data Integrity Failures	A8	Code Tampering	M8	Lack of Device Management	I8
Security Logging and Monitoring Failures	A9	Reverse Engineering	M9	Insecure Default Settings	I9
Server-Side Request Forgery (SSRF)	A10	Extraneous Functionality	M10	Lack of Physical Hardening	I10

7 References and Templates

Separate / list down all the SOPs and other STS related documents which support this process

Name	Description	Owner	Location
Penetration Testing Methodology	Describes how Penetration Testing is delivered for each of its subservices	Krystian Szybis	SPT - Public
SPT Service Catalogue	For list of services	Krystian Szybis	Service Catalogue
Penetration Testing Artefacts	Report template, SoW, etc.	Krystian Szybis	Internal Confluence page
Security Remediation	Security Remediation	Hariharan Bala	Remediation Sharepoint