**Vulnerability Assessment Report**

**For**



**Dell Cloud**

**Date March 03 2022**

**Document Security Level:** Confidential

**Document Version:** 1.0

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Edit Report** | **Editor** |
| 1.0 | March 03 2022 | Creation | INET Managed Services |

|  |  |  |  |
| --- | --- | --- | --- |
| Dell Cloud | | INET Managed Services CO., LTD. | |
| Name |  | **Name** |  |
| Position |  | **Position** |  |
| Tel |  | **Tel** |  |
| Signature |  | **Signature** |  |

**Table of Contents**

[1. Restrictions on disclosure and use of information 4](#_Toc97201647)

[2. Operation Method 5](#_Toc97201648)

[3. Project Scope 6](#_Toc97201649)

[3.1 Infrastructure Vulnerability Assessment 6](#_Toc97201650)

[3.2 Web Application Vulnerability Assessment 6](#_Toc97201651)

[4. Testing Tools 7](#_Toc97201652)

[5. Infrastructure Vulnerability Assessment 7](#_Toc97201653)

[5.1 Target Information 8](#_Toc97201654)

[5.2 Executive summary 9](#_Toc97201655)

[5.2.1 Summary Vulnerability by Severity 9](#_Toc97201656)

[5.2.2 Vulnerability by Target 9](#_Toc97201657)

[5.3 Infrastructure Vulnerability Detail 11](#_Toc97201658)

[6. Web Application Vulnerability Assessment 12](#_Toc97201659)

[6.1 Target Information 13](#_Toc97201660)

[6.2 Executive summary 14](#_Toc97201661)

[6.2.1 Summary Vulnerability by Severity 14](#_Toc97201662)

[6.2.2 Vulnerability by Target 14](#_Toc97201663)

[6.3 Web Application Vulnerability Detail 15](#_Toc97201664)

[7. Port Discovery 18](#_Toc97201665)

[8. Appendix 19](#_Toc97201666)

[8.1 About Nessus 19](#_Toc97201667)

[8.1.1 Nessus vulnerabilities 19](#_Toc97201668)

[8.1.2 Nessus risk score 19](#_Toc97201669)

[8.2 About Acunetix 21](#_Toc97201670)

[8.2.1 Acunetix web vulnerabilities 21](#_Toc97201671)

[8.2.2 Acunetix risk score 22](#_Toc97201672)

# Restrictions on disclosure and use of information

Restriction on Disclosure and Use of Confidential Information. The Executive understands and agrees that the Confidential Information constitutes an asset of the Company and its affiliated entities and may not be converted to the Executive's own use. Accordingly, the Executive hereby agrees that the Executive shall not, directly, or indirectly, at any time, reveal, divulge, or disclose to any Person not expressly authorized by the Company any Confidential Information, and the Executive shall not, directly, or indirectly, use or make use of any Confidential Information in connection with any business activity other than that of the Company. The parties acknowledge and agree that this Agreement is not intended to, and does not, alter either the Company's rights or the Executive's obligations under any state or federal statutory or common law regarding trade secrets and unfair trade.

# Operation Method

* 1. Posture Review
  2. Information Gathering
  3. Enumeration
  4. Vulnerability Assessment
  5. Analyze & Evaluate Risk Value
  6. Report



Figure 1: Operation Method

# Project Scope

## **3.1 Infrastructure Vulnerability Assessment**

**Target / IP Address:**

| **No.** | **Domain / Server Name** | **Public IP Address** | **Private IP Address** | **OS/Model** | **Functions** | **Public Assessment** | **Private Assessment** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | DATABASE01 | - | 172.16.69.13 | Ubuntu 18 | Database Server 01 |  | ✓ |
| 2 | WEB01 | 123.123.123.123 | 172.16.69.14 | Ubuntu 20 | Web Server |  | ✓ |
| 3 | TERM | - | 172.16.69.52 | Windows Server 2016 | Terminal Server |  | ✓ |
| 4 | SMB01 | 12.12.12.12 | 172.16.69.53 | Windows Server 2019 | SMB Server |  | ✓ |
| 5 | DATABASE02 | - | 172.16.69.54 | Ubuntu 18 | Database Server 02 |  | ✓ |

## **3.2 Web Application Vulnerability Assessment**

**Target / IP Address:**

| **No.** | **Domain / Server Name** | **Public IP Address** | **Private IP Address** | **OS/Model** | **Functions** | **Public Assessment** | **Private Assessment** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | https://example.com/ | 123.123.123.123 | 172.16.69.14 | Ubuntu 20 | เว็บไซต์ขายของ | ✓ |  |

# Testing Tools

|  |  |
| --- | --- |
| **Tool Name** | **Testing Type** |
| Nmap | Host and Service Discovery |
| Nessus | Infrastructure Vulnerability Assessment |
| Acunetix | Web Application Vulnerability Assessment |

# Infrastructure Vulnerability Assessment

**Vulnerability Assessment from Public Access (for public target)**

**Testing date:** March 30, 2021

**Tester IP Address:** 203.150.110.29

Diagram

Description automatically generated

Figure 2: Vulnerability Assessment from Public Access

**Vulnerability Assessment from Private Access (for private or restricted access target)**

**Testing date:** March 30, 2021

**Tester IP Address:** Private IP from VPN access

A picture containing diagram

Description automatically generated

Figure 3: Vulnerability Assessment from Private Access

## **5.1 Target Information**

| **No.** | **Domain / Server Name** | **IP Address** | **OS/Model** | **Port** |
| --- | --- | --- | --- | --- |
| 1 |  | 192.168.8.12 |  | TCP : 53, 88, 135, 139, 445, 464, 636, 3268, 3389, 5985, 49152, 49153, 49154, 49157, 49159, 49167, 49189, 60423, 60424, 60433 UDP : 53 |
| 2 |  | 192.168.8.13 |  | TCP : 135, 139, 445, 1433, 3389, 5985, 49152, 49153, 49154, 49155, 49156, 49170, 49173, 63422 |
| 3 |  | 192.168.8.14 |  | TCP : 389, 514 |
| 4 |  | 192.168.8.15 |  | TCP : 80, 443, 514, 636, 1514, 2012, 2014, 2015, 2020, 9443 |
| 5 |  | 192.168.8.17 |  | TCP : 22, 111, 53053 UDP : 111, 47591 |
| 6 |  | 192.168.8.31 |  | TCP : 22, 80 |
| 7 |  | 192.168.8.35 |  | TCP : 22, 80, 443 |
| 8 |  | 192.168.8.91 |  | TCP : 111, 135, 139, 445, 1063, 2049, 2805, 3389, 5985, 6160, 6161, 6162, 6190, 8190, 8191, 9392, 9393, 10001, 10002, 10003, 10005, 11731, 49152, 49153, 49154, 49155, 49156, 49161, 49164, 49166 UDP : 1434 |
| 9 |  | 192.168.8.136 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 10 |  | 192.168.8.137 |  | TCP : 5989 |
| 11 |  | 192.168.8.138 |  | TCP : 22, 8300 |
| 12 |  | 192.168.8.160 |  | TCP : 111, 135, 139, 445, 1063, 2049, 2500, 3389, 5985, 6160, 6161, 6162, 6190, 9392, 9393, 10001, 10002, 10003, 10005, 11731, 49152, 49153, 49154, 49155, 49156, 49160, 49161, 49163 UDP : 1434 |
| 13 |  | 192.168.8.166 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 14 |  | 192.168.8.167 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 15 |  | 192.168.8.168 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 16 |  | 192.168.8.171 |  | TCP : 111, 135, 139, 445, 1063, 2049, 3389, 5985, 6160, 6161, 6162, 6190, 9392, 9393, 10001, 10002, 10003, 10005, 11731, 49152, 49153, 49154, 49155, 49156, 49158, 49163, 49164, 49170 UDP : 1434 |
| 17 |  | 192.168.8.211 |  | TCP : 80, 135, 139, 445, 1433, 2383, 3389, 5985, 49152, 49153, 49154, 49155, 49160, 49265, 49267 |
| 18 |  | 192.168.8.212 |  | TCP : 3389 |
| 19 |  | 192.168.8.213 |  | TCP : 3389 |
| 20 |  | 192.168.8.231 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 21 |  | 192.168.8.232 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 22 |  | 192.168.8.245 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 23 |  | 192.168.8.246 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 24 |  | 192.168.10.1 |  | TCP : 53, 88, 135, 139, 389, 445, 464, 593, 636, 3268, 3269, 3389, 5985, 49664, 49665, 49666, 49667, 49670, 49672, 49675, 49693, 59240 UDP : 53 |
| 25 |  | 192.168.10.4 |  | TCP : 22, 80, 88, 389, 443, 514, 636, 1514, 2012, 2014, 2015, 2020, 9443 |
| 26 |  | 192.168.10.7 |  | TCP : 22, 80, 88, 389, 443, 514, 636, 1514, 2012, 2014, 2015, 2020, 9443 |
| 27 |  | 192.168.10.27 |  | TCP : 80, 135, 139, 445, 1433, 2383, 3389, 5985, 49152, 49153, 49154, 49155, 49156, 49266, 49268 |
| 28 |  | 192.168.10.191 |  | TCP : 80, 443, 5989, 8080 |
| 29 |  | 192.168.10.192 |  | TCP : 80, 8080 |
| 30 |  | 192.168.10.193 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 31 |  | 192.168.10.194 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 32 |  | 192.168.10.210 |  | TCP : 22, 80, 443 |
| 33 |  | 192.168.10.211 |  | TCP : 22 |
| 34 |  | 192.168.10.220 |  | TCP : 80, 135, 139, 445, 1433, 3389, 5985, 49152, 49153, 49154, 49155, 49156, 49188, 49189, 49194 |
| 35 |  | 192.168.10.221 |  | TCP : 22 |
| 36 |  | 192.168.10.222 |  | TCP : 22, 111, 56227 UDP : 111, 38788 |
| 37 |  | 192.168.10.224 |  | TCP : 22, 111 UDP : 111 |
| 38 |  | 192.168.10.225 |  | TCP : 22, 111 UDP : 111 |
| 39 |  | 192.168.10.226 |  | TCP : 80, 135, 139, 445, 1433, 3389, 5985, 49152, 49153, 49154, 49155, 49157, 49160, 49161 |
| 40 |  | 192.168.11.131 |  | TCP : 9080 |
| 41 |  | 192.168.11.132 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 42 |  | 192.168.11.134 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 43 |  | 192.168.11.135 |  | TCP : 5989, 8000, 8080 |
| 44 |  | 192.168.11.141 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 45 |  | 192.168.11.142 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 46 |  | 192.168.11.143 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 47 |  | 192.168.11.144 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 48 |  | 192.168.11.151 |  | TCP : 22, 80, 443, 902, 8000, 8080, 8182, 8300 |
| 49 |  | 192.168.11.152 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 8300, 9080 |
| 50 |  | 192.168.11.153 |  | TCP : 80, 443, 902, 5989, 8000, 8080, 8182, 9080 |
| 51 |  | 192.168.11.160 |  | TCP : 111, 135, 139, 445, 1063, 2049, 3389, 5985, 9392, 9393, 10001, 10002, 10003, 10005, 49275 UDP : 13, 111, 1434 |
| 52 |  | 192.168.11.176 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 53 |  | 192.168.11.177 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300 |
| 54 |  | 192.168.11.178 |  | TCP : 22, 80, 443, 5989, 8000, 8182, 8300, 9080 |
| 55 |  | 192.168.11.179 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8300, 9080 |
| 56 |  | 192.168.11.180 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 57 |  | 192.168.11.181 |  | TCP : 22, 902, 5989 |
| 58 |  | 192.168.11.182 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 59 |  | 192.168.11.183 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 60 |  | 192.168.11.184 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 61 |  | 192.168.11.185 |  | TCP : 22, 80, 443, 902, 5989, 8000, 8182, 8300, 9080 |
| 62 |  | 192.168.11.200 |  | TCP : 111, 135, 139, 445, 1063, 2049, 3389, 5985, 6160, 6161, 6162, 6190, 9392, 9393, 10001, 10002, 10003, 10005, 11731, 49152, 49153, 49154, 49155, 49156, 49159, 49202, 49209, 49211 UDP : 1434 |

## **5.2 Executive summary**

The purpose of this activity is to find the vulnerability on the target infrastructure.

### **5.2.1 Summary Vulnerability by Severity**

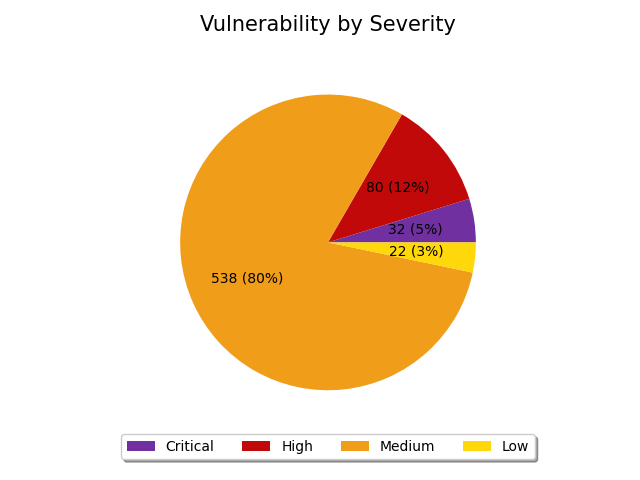


Figure 4: Summary by Severity of Infrastructure Vulnerability Assessment

### **5.2.2 Vulnerability by Target**

| **No.** | **Domain/Server Name** | **IP Address** | **Critical** | **High** | **Medium** | **Low** | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | - | 192.168.8.12 | 0 | 1 | 4 | 0 | 5 |
| 2 | - | 192.168.8.13 | 1 | 5 | 12 | 1 | 19 |
| 3 | - | 192.168.8.14 | 0 | 0 | 0 | 0 | 0 |
| 4 | - | 192.168.8.15 | 0 | 0 | 6 | 0 | 6 |
| 5 | - | 192.168.8.17 | 0 | 0 | 1 | 3 | 4 |
| 6 | - | 192.168.8.31 | 0 | 0 | 0 | 0 | 0 |
| 7 | - | 192.168.8.35 | 0 | 0 | 3 | 0 | 3 |
| 8 | - | 192.168.8.91 | 1 | 3 | 10 | 0 | 14 |
| 9 | - | 192.168.8.136 | 1 | 3 | 18 | 1 | 23 |
| 10 | - | 192.168.8.137 | 0 | 0 | 2 | 0 | 2 |
| 11 | - | 192.168.8.138 | 0 | 0 | 0 | 1 | 1 |
| 12 | - | 192.168.8.160 | 1 | 3 | 10 | 0 | 14 |
| 13 | - | 192.168.8.166 | 1 | 3 | 17 | 1 | 22 |
| 14 | - | 192.168.8.167 | 1 | 3 | 18 | 1 | 23 |
| 15 | - | 192.168.8.168 | 1 | 3 | 18 | 1 | 23 |
| 16 | - | 192.168.8.171 | 1 | 3 | 10 | 0 | 14 |
| 17 | - | 192.168.8.211 | 1 | 3 | 9 | 0 | 13 |
| 18 | - | 192.168.8.212 | 0 | 1 | 4 | 0 | 5 |
| 19 | - | 192.168.8.213 | 0 | 1 | 4 | 0 | 5 |
| 20 | - | 192.168.8.231 | 1 | 5 | 18 | 1 | 25 |
| 21 | - | 192.168.8.232 | 1 | 5 | 18 | 1 | 25 |
| 22 | - | 192.168.8.245 | 1 | 3 | 18 | 1 | 23 |
| 23 | - | 192.168.8.246 | 1 | 3 | 18 | 1 | 23 |
| 24 | - | 192.168.10.1 | 0 | 1 | 4 | 0 | 5 |
| 25 | - | 192.168.10.4 | 1 | 0 | 6 | 0 | 7 |
| 26 | - | 192.168.10.7 | 1 | 0 | 4 | 0 | 5 |
| 27 | - | 192.168.10.27 | 1 | 3 | 9 | 0 | 13 |
| 28 | - | 192.168.10.191 | 1 | 3 | 12 | 0 | 16 |
| 29 | - | 192.168.10.192 | 0 | 0 | 3 | 0 | 3 |
| 30 | - | 192.168.10.193 | 1 | 3 | 18 | 0 | 22 |
| 31 | - | 192.168.10.194 | 1 | 3 | 18 | 0 | 22 |
| 32 | - | 192.168.10.210 | 0 | 0 | 2 | 0 | 2 |
| 33 | - | 192.168.10.211 | 0 | 0 | 0 | 0 | 0 |
| 34 | - | 192.168.10.220 | 1 | 3 | 9 | 0 | 13 |
| 35 | - | 192.168.10.221 | 0 | 0 | 0 | 2 | 2 |
| 36 | - | 192.168.10.222 | 0 | 0 | 0 | 2 | 2 |
| 37 | - | 192.168.10.224 | 0 | 0 | 0 | 2 | 2 |
| 38 | - | 192.168.10.225 | 0 | 0 | 0 | 2 | 2 |
| 39 | - | 192.168.10.226 | 1 | 3 | 9 | 0 | 13 |
| 40 | - | 192.168.11.131 | 0 | 0 | 3 | 0 | 3 |
| 41 | - | 192.168.11.132 | 1 | 1 | 16 | 0 | 18 |
| 42 | - | 192.168.11.134 | 1 | 1 | 16 | 0 | 18 |
| 43 | - | 192.168.11.135 | 0 | 0 | 5 | 0 | 5 |
| 44 | - | 192.168.11.141 | 1 | 1 | 16 | 0 | 18 |
| 45 | - | 192.168.11.142 | 1 | 1 | 16 | 0 | 18 |
| 46 | - | 192.168.11.143 | 1 | 1 | 16 | 0 | 18 |
| 47 | - | 192.168.11.144 | 1 | 1 | 16 | 0 | 18 |
| 48 | - | 192.168.11.151 | 1 | 1 | 11 | 1 | 14 |
| 49 | - | 192.168.11.152 | 1 | 1 | 16 | 0 | 18 |
| 50 | - | 192.168.11.153 | 1 | 1 | 16 | 0 | 18 |
| 51 | - | 192.168.11.160 | 1 | 2 | 9 | 0 | 12 |
| 52 | - | 192.168.11.176 | 0 | 0 | 7 | 0 | 7 |
| 53 | - | 192.168.11.177 | 0 | 0 | 5 | 0 | 5 |
| 54 | - | 192.168.11.178 | 0 | 0 | 7 | 0 | 7 |
| 55 | - | 192.168.11.179 | 0 | 0 | 5 | 0 | 5 |
| 56 | - | 192.168.11.180 | 0 | 0 | 7 | 0 | 7 |
| 57 | - | 192.168.11.181 | 0 | 0 | 2 | 0 | 2 |
| 58 | - | 192.168.11.182 | 0 | 0 | 7 | 0 | 7 |
| 59 | - | 192.168.11.183 | 0 | 0 | 7 | 0 | 7 |
| 60 | - | 192.168.11.184 | 0 | 0 | 7 | 0 | 7 |
| 61 | - | 192.168.11.185 | 0 | 0 | 7 | 0 | 7 |
| 62 | - | 192.168.11.200 | 1 | 2 | 9 | 0 | 12 |
| **Total** | | | 32 | 80 | 538 | 22 | 672 |

## **5.3 Infrastructure Vulnerability Detail**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 1 | **Finding** | Microsoft SQL Server Unsupported Version Detection (remote check) |
| **Severity** | Critical | **Port** | TCP: 1433, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.13(1433), 192.168.8.91(49164), 192.168.8.160(49163), 192.168.8.171(49170), 192.168.8.211(1433), 192.168.10.27(1433), 192.168.10.220(1433), 192.168.10.226(1433), 192.168.11.160(49275), 192.168.11.200(49202) | | |
| **Detail** | According to its self-reported version number, the installation of Microsoft SQL Server on the remote host is no longer supported. Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it is likely to contain security vulnerabilities. | | |
| **Solution** | Upgrade to a version of Microsoft SQL Server that is currently supported. | | |
| **Remark** | http://www.nessus.org/u?d4418a57 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 2 | **Finding** | VMware vCenter Server 6.5 / 6.7 Multiple Vulnerabilities (VMSA-2021-0027) |
| **Severity** | Critical | **Port** | TCP: 443 |
| **Target** | 192.168.10.4(443), 192.168.10.7(443) | | |
| **Detail** | The version of VMware vCenter Server installed on the remote host is 6.5 prior to 6.5 U3r or 6.7 prior to 6.7 U3p. It is, therefore, affected by multiple vulnerabilities:  - An arbitrary file read vulnerability exists in the vSphere web client. An unauthenticated, remote attacker  can exploit this, via HTTPS, to gain access to sensitive information. (CVE-2021-21980)  - A server side request forgery vulnerability exists in the vSAN Web Client plug-in. An unauthenticated,  remote attacker can exploit this, via HTTPS, to cause the server to access internal services or access  sites outside of vCenter. (CVE-2021-22049) Note that Nessus has not tested for these issues but has instead relied only on the application's self-reported version  number. Nessus has also not tested for the presence of a workaround. | | |
| **Solution** | Upgrade to VMware vCenter Server 6.5 U3r, 6.7 U3p, or later. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2021-0027.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 3 | **Finding** | ESXi 6.5 / 6.7 XSS (VMSA-2020-0008) |
| **Severity** | Critical | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443), 192.168.11.132(443), 192.168.11.134(443), 192.168.11.141(443), 192.168.11.142(443), 192.168.11.143(443), 192.168.11.144(443), 192.168.11.151(443), 192.168.11.152(443), 192.168.11.153(443) | | |
| **Detail** | The remote VMware ESXi host is version 6.5 or 6.7 and is affected by a cross-site scripting (XSS) vulnerability in virtual machine attributes due to improper validation of user-supplied input before returning it to users. An authenticated, remote attacker with access to modify the system properties of a virtual machine from inside the guest OS can exploit this, by inserting script-related HTML in the system properties and having a user view the system properties from the ESXi Host Client, to execute arbitrary script code in a user's ESXi Host Client session. Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2020-0008.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 4 | **Finding** | SSL Medium Strength Cipher Suites Supported (SWEET32) |
| **Severity** | High | **Port** | TCP: 1433, 3389, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.12(3389), 192.168.8.13(1433, 3389), 192.168.8.91(3389, 49164), 192.168.8.160(3389, 49163), 192.168.8.171(3389, 49170), 192.168.8.211(1433, 3389), 192.168.8.212(3389), 192.168.8.213(3389), 192.168.10.1(3389), 192.168.10.27(1433, 3389), 192.168.10.220(1433, 3389), 192.168.10.226(1433, 3389), 192.168.11.160(3389, 49275), 192.168.11.200(3389, 49202) | | |
| **Detail** | The remote host supports the use of SSL ciphers that offer medium strength encryption. Nessus regards medium strength as any encryption that uses key lengths at least 64 bits and less than 112 bits, or  else that uses the 3DES encryption suite. Note that it is considerably easier to circumvent medium strength encryption if the attacker is on the same physical network. | | |
| **Solution** | Reconfigure the affected application if possible to avoid use of medium strength ciphers. | | |
| **Remark** | https://www.openssl.org/blog/blog/2016/08/24/sweet32/ https://sweet32.info | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 5 | **Finding** | SSL Certificate Signed Using Weak Hashing Algorithm |
| **Severity** | High | **Port** | TCP: 1433, 3389 |
| **Target** | 192.168.8.13(1433, 3389), 192.168.8.211(1433), 192.168.10.27(1433), 192.168.10.220(1433), 192.168.10.226(1433) | | |
| **Detail** | The remote service uses an SSL certificate chain that has been signed using a cryptographically weak hashing algorithm (e.g. MD2, MD4, MD5, or SHA1). These signature algorithms are known to be vulnerable to collision attacks. An attacker can exploit this to generate another certificate with the same digital signature, allowing an attacker to masquerade as the affected service. Note that this plugin reports all SSL certificate chains signed with SHA-1 that expire after January 1, 2017 as vulnerable. This is in accordance with Google's gradual sunsetting of the SHA-1 cryptographic hash algorithm. Note that certificates in the chain that are contained in the Nessus CA database (known\_CA.inc) have been ignored. | | |
| **Solution** | Contact the Certificate Authority to have the SSL certificate reissued. | | |
| **Remark** | https://tools.ietf.org/html/rfc3279 http://www.nessus.org/u?9bb87bf2 http://www.nessus.org/u?e120eea1 http://www.nessus.org/u?5d894816 http://www.nessus.org/u?51db68aa http://www.nessus.org/u?9dc7bfba | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 6 | **Finding** | SSL Version 2 and 3 Protocol Detection |
| **Severity** | High | **Port** | TCP: 1433, 49163, 49164, 49170 |
| **Target** | 192.168.8.13(1433), 192.168.8.91(49164), 192.168.8.160(49163), 192.168.8.171(49170) | | |
| **Detail** | The remote service accepts connections encrypted using SSL 2.0 and/or SSL 3.0. These versions of SSL are affected by several cryptographic flaws, including:  - An insecure padding scheme with CBC ciphers.  - Insecure session renegotiation and resumption schemes. An attacker can exploit these flaws to conduct man-in-the-middle attacks or to decrypt communications between the affected service and clients. Although SSL/TLS has a secure means for choosing the highest supported version of the protocol (so that these versions will be used only if the client or server support nothing better), many web browsers implement this in an unsafe way that allows an attacker to downgrade a connection (such as in POODLE). Therefore, it is recommended that these protocols be disabled entirely. NIST has determined that SSL 3.0 is no longer acceptable for secure communications. As of the date of enforcement found in PCI DSS v3.1, any version of SSL will not meet the PCI SSC's definition of 'strong cryptography'. | | |
| **Solution** | Consult the application's documentation to disable SSL 2.0 and 3.0. Use TLS 1.2 (with approved cipher suites) or higher instead. | | |
| **Remark** | https://www.schneier.com/academic/paperfiles/paper-ssl.pdf http://www.nessus.org/u?b06c7e95 http://www.nessus.org/u?247c4540 https://www.openssl.org/~bodo/ssl-poodle.pdf http://www.nessus.org/u?5d15ba70 https://www.imperialviolet.org/2014/10/14/poodle.html https://tools.ietf.org/html/rfc7507 https://tools.ietf.org/html/rfc7568 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 7 | **Finding** | ESXi 6.5 / 6.7 / 7.0 Multiple Vulnerabilities (VMSA-2020-0026) |
| **Severity** | High | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443), 192.168.11.132(443), 192.168.11.134(443), 192.168.11.141(443), 192.168.11.142(443), 192.168.11.143(443), 192.168.11.144(443), 192.168.11.151(443), 192.168.11.152(443), 192.168.11.153(443) | | |
| **Detail** | According to its self-reported version number, the remote VMware ESXi host is version 6.5, 6.7 or 7.0 and is affected by multiple vulnerabilities.   - A use-after-free error exists in the XHCI USB controller. An unauthenticated, local attacker with local  administrative privileges on a virtual machine can exploit this, to execute code as the virtual machine's  VMX process running on the host. (CVE-2020-4004)  - A privilege escalation vulnerability exists in ESXi due to how certain system calls are managed. An  authenticated, local attacker with privileges within the VPM process can exploit this, when chained with  CVE-2020-4004, to obtain escalated privileges. (CVE-2020-4005) Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2020-0026.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 8 | **Finding** | ESXi 6.0 / 6.5 / 6.7 Out-of-Bounds Read Vulnerability (VMSA-2018-0026) (Remote Check) |
| **Severity** | High | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443) | | |
| **Detail** | The remote VMware ESXi host is version 6.0, 6.5, or 6.7 and is missing a security patch. It is, therefore, vulnerable to an out-of-bounds read vulnerability in SVGA devices. An attacker with access to a guest system may be able to execute code on the host system by leveraging this vulnerability. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2018-0026.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 9 | **Finding** | ESXi 6.0 / 6.5 / 6.7 Multiple Vulnerabilities (VMSA-2018-0027) (Remote Check) |
| **Severity** | High | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443) | | |
| **Detail** | The remote VMware ESXi host is version 6.0, 6.5, or 6.7 and is missing a security patch. It is, therefore, vulnerable to multiple vulnerabilities. Leveraging the most severe of these vulnerabilities could allow an attacker to execute arbitrary code on the host from the security context of an unprivileged user on the guest system. Note: CVE-2018-6982 only applies to ESXi 6.5 and 6.7 installations.  ESXi 6.0 installations are not affected. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2018-0027.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 10 | **Finding** | ESXi 5.5 / 6.0 / 6.5 / Multiple Vulnerabilities (VMSA-2017-0021) (VMSA-2018-0002) (Spectre) (remote check) |
| **Severity** | High | **Port** | TCP: 443 |
| **Target** | 192.168.8.231(443), 192.168.8.232(443) | | |
| **Detail** | The remote VMware ESXi host is version 5.5, 6.0, or 6.5 and is missing a security patch. It is, therefore, affected by multiple vulnerabilities that can allow code execution in a virtual machine via the authenticated VNC session as well as cause information disclosure from one virtual machine to another virtual machine on the same host. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2017-0021.html https://www.talosintelligence.com/vulnerability\_reports/TALOS-2017-0369 https://www.vmware.com/us/security/advisories/VMSA-2018-0002.html https://meltdownattack.com/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 11 | **Finding** | ESXi 6.5 Build 6765664 Heap Buffer Overflow (VMSA-2017-0021) (remote check) |
| **Severity** | High | **Port** | TCP: 443 |
| **Target** | 192.168.8.231(443), 192.168.8.232(443) | | |
| **Detail** | The version of the remote VMware ESXi 6.5 host is prior to build 6765664. It is, therefore, affected by a heap buffer overflow vulnerability that can be triggered by a specially crafted set of VNC packets. | | |
| **Solution** | Apply patch ESXi650-201710401-BG according to the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2017-0021.html https://www.talosintelligence.com/vulnerability\_reports/TALOS-2017-0368 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 12 | **Finding** | SSL Certificate Cannot Be Trusted |
| **Severity** | Medium | **Port** | TCP: 443, 636, 1433, 1514, 3389, 5989, 8080, 8182, 9080, 9443, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.12(3389), 192.168.8.13(1433, 3389), 192.168.8.15(443, 636, 1514, 9443), 192.168.8.35(443), 192.168.8.91(3389, 49164), 192.168.8.136(443, 5989, 8080, 8182, 9080), 192.168.8.137(5989), 192.168.8.160(3389, 49163), 192.168.8.166(443, 5989, 8080, 8182, 9080), 192.168.8.167(443, 5989, 8080, 8182, 9080), 192.168.8.168(443, 5989, 8080, 8182, 9080), 192.168.8.171(3389, 49170), 192.168.8.211(1433, 3389), 192.168.8.212(3389), 192.168.8.213(3389), 192.168.8.231(443, 5989, 8080, 8182, 9080), 192.168.8.232(443, 5989, 8080, 8182, 9080), 192.168.8.245(443, 5989, 8080, 8182, 9080), 192.168.8.246(443, 5989, 8080, 8182, 9080), 192.168.10.1(3389), 192.168.10.4(443, 636, 1514, 9443), 192.168.10.7(443, 636, 1514, 9443), 192.168.10.27(1433, 3389), 192.168.10.191(443, 5989, 8080), 192.168.10.192(8080), 192.168.10.193(443, 5989, 8080, 8182, 9080), 192.168.10.194(443, 5989, 8080, 8182, 9080), 192.168.10.210(443), 192.168.10.220(1433, 3389), 192.168.10.226(1433, 3389), 192.168.11.131(9080), 192.168.11.132(443, 5989, 8080, 8182, 9080), 192.168.11.134(443, 5989, 8080, 8182, 9080), 192.168.11.135(5989, 8080), 192.168.11.141(443, 5989, 8080, 8182, 9080), 192.168.11.142(443, 5989, 8080, 8182, 9080), 192.168.11.143(443, 5989, 8080, 8182, 9080), 192.168.11.144(443, 5989, 8080, 8182, 9080), 192.168.11.151(443, 8080, 8182), 192.168.11.152(443, 5989, 8080, 8182, 9080), 192.168.11.153(443, 5989, 8080, 8182, 9080), 192.168.11.160(3389, 49275), 192.168.11.176(443, 5989, 8182, 9080), 192.168.11.177(443, 5989, 8182), 192.168.11.178(443, 5989, 8182, 9080), 192.168.11.179(443, 5989, 9080), 192.168.11.180(443, 5989, 8182, 9080), 192.168.11.181(5989), 192.168.11.182(443, 5989, 8182, 9080), 192.168.11.183(443, 5989, 8182, 9080), 192.168.11.184(443, 5989, 8182, 9080), 192.168.11.185(443, 5989, 8182, 9080), 192.168.11.200(3389, 49202) | | |
| **Detail** | The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below :  - First, the top of the certificate chain sent by the  server might not be descended from a known public  certificate authority. This can occur either when the  top of the chain is an unrecognized, self-signed  certificate, or when intermediate certificates are  missing that would connect the top of the certificate  chain to a known public certificate authority.  - Second, the certificate chain may contain a certificate  that is not valid at the time of the scan. This can  occur either when the scan occurs before one of the  certificate's 'notBefore' dates, or after one of the  certificate's 'notAfter' dates.  - Third, the certificate chain may contain a signature  that either didn't match the certificate's information  or could not be verified. Bad signatures can be fixed by  getting the certificate with the bad signature to be  re-signed by its issuer. Signatures that could not be  verified are the result of the certificate's issuer  using a signing algorithm that Nessus either does not  support or does not recognize. If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and  identity of the web server. This could make it easier to carry out  man-in-the-middle attacks against the remote host. | | |
| **Solution** | Purchase or generate a proper SSL certificate for this service. | | |
| **Remark** | https://www.itu.int/rec/T-REC-X.509/en https://en.wikipedia.org/wiki/X.509 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 13 | **Finding** | SSL Self-Signed Certificate |
| **Severity** | Medium | **Port** | TCP: 443, 1433, 3389, 5989, 8080, 8182, 9080, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.12(3389), 192.168.8.13(1433, 3389), 192.168.8.35(443), 192.168.8.91(3389, 49164), 192.168.8.136(443, 8080, 8182, 9080), 192.168.8.160(3389, 49163), 192.168.8.166(443, 8080, 8182, 9080), 192.168.8.167(443, 8080, 8182, 9080), 192.168.8.168(443, 8080, 8182, 9080), 192.168.8.171(3389, 49170), 192.168.8.211(1433, 3389), 192.168.8.212(3389), 192.168.8.213(3389), 192.168.8.231(443, 8080, 8182, 9080), 192.168.8.232(443, 8080, 8182, 9080), 192.168.8.245(443, 8080, 8182, 9080), 192.168.8.246(443, 8080, 8182, 9080), 192.168.10.1(3389), 192.168.10.27(1433, 3389), 192.168.10.191(443, 8080), 192.168.10.192(8080), 192.168.10.193(443, 8080, 8182, 9080), 192.168.10.194(443, 8080, 8182, 9080), 192.168.10.210(443), 192.168.10.220(1433, 3389), 192.168.10.226(1433, 3389), 192.168.11.131(9080), 192.168.11.132(443, 8080, 8182, 9080), 192.168.11.134(443, 8080, 8182, 9080), 192.168.11.135(8080), 192.168.11.141(443, 8080, 8182, 9080), 192.168.11.142(443, 8080, 8182, 9080), 192.168.11.143(443, 8080, 8182, 9080), 192.168.11.144(443, 8080, 8182, 9080), 192.168.11.151(443, 8080, 8182), 192.168.11.152(443, 8080, 8182, 9080), 192.168.11.153(443, 8080, 8182, 9080), 192.168.11.160(3389, 49275), 192.168.11.176(5989, 8182, 9080), 192.168.11.177(5989, 8182), 192.168.11.178(5989, 8182, 9080), 192.168.11.179(5989, 9080), 192.168.11.180(5989, 8182, 9080), 192.168.11.181(5989), 192.168.11.182(5989, 8182, 9080), 192.168.11.183(5989, 8182, 9080), 192.168.11.184(5989, 8182, 9080), 192.168.11.185(5989, 8182, 9080), 192.168.11.200(3389, 49202) | | |
| **Detail** | The X.509 certificate chain for this service is not signed by a recognized certificate authority. If the remote host is a public host in production, this nullifies the use of SSL as anyone could establish a man-in-the-middle attack against the remote host.  Note that this plugin does not check for certificate chains that end in a certificate that is not self-signed, but is signed by an unrecognized certificate authority. | | |
| **Solution** | Purchase or generate a proper SSL certificate for this service. | | |
| **Remark** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 14 | **Finding** | SSL RC4 Cipher Suites Supported (Bar Mitzvah) |
| **Severity** | Medium | **Port** | TCP: 1433, 3389, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.12(3389), 192.168.8.13(1433, 3389), 192.168.8.91(3389, 49164), 192.168.8.160(3389, 49163), 192.168.8.171(3389, 49170), 192.168.8.211(1433, 3389), 192.168.8.212(3389), 192.168.8.213(3389), 192.168.10.1(3389), 192.168.10.27(1433, 3389), 192.168.10.220(1433, 3389), 192.168.10.226(1433, 3389), 192.168.11.160(3389, 49275), 192.168.11.200(3389, 49202) | | |
| **Detail** | The remote host supports the use of RC4 in one or more cipher suites. The RC4 cipher is flawed in its generation of a pseudo-random stream of bytes so that a wide variety of small biases are introduced into the stream, decreasing its randomness. If plaintext is repeatedly encrypted (e.g., HTTP cookies), and an attacker is able to obtain many (i.e., tens of millions) ciphertexts, the attacker may be able to derive the plaintext. | | |
| **Solution** | Reconfigure the affected application, if possible, to avoid use of RC4 ciphers. Consider using TLS 1.2 with AES-GCM suites subject to browser and web server support. | | |
| **Remark** | https://www.rc4nomore.com/ http://www.nessus.org/u?ac7327a0 http://cr.yp.to/talks/2013.03.12/slides.pdf http://www.isg.rhul.ac.uk/tls/ https://www.imperva.com/docs/HII\_Attacking\_SSL\_when\_using\_RC4.pdf | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 15 | **Finding** | TLS Version 1.0 Protocol Detection |
| **Severity** | Medium | **Port** | TCP: 443, 636, 1433, 3389, 5989, 8080, 8182, 9080, 49163, 49164, 49170, 49202, 49275 |
| **Target** | 192.168.8.12(3389), 192.168.8.13(1433, 3389), 192.168.8.15(443, 636), 192.168.8.35(443), 192.168.8.91(3389, 49164), 192.168.8.136(443, 5989, 8080, 8182, 9080), 192.168.8.137(5989), 192.168.8.160(3389, 49163), 192.168.8.166(443, 8080, 8182, 9080), 192.168.8.167(443, 5989, 8080, 8182, 9080), 192.168.8.168(443, 5989, 8080, 8182, 9080), 192.168.8.171(3389, 49170), 192.168.8.211(1433, 3389), 192.168.8.212(3389), 192.168.8.213(3389), 192.168.8.231(443, 5989, 8080, 8182, 9080), 192.168.8.232(443, 5989, 8080, 8182, 9080), 192.168.8.245(443, 5989, 8080, 8182, 9080), 192.168.8.246(443, 5989, 8080, 8182, 9080), 192.168.10.1(3389), 192.168.10.4(443, 636), 192.168.10.27(1433, 3389), 192.168.10.191(443, 5989, 8080), 192.168.10.192(8080), 192.168.10.193(443, 5989, 8080, 8182, 9080), 192.168.10.194(443, 5989, 8080, 8182, 9080), 192.168.10.220(1433, 3389), 192.168.10.226(1433, 3389), 192.168.11.131(9080), 192.168.11.132(443, 5989, 8080, 8182, 9080), 192.168.11.134(443, 5989, 8080, 8182, 9080), 192.168.11.135(5989, 8080), 192.168.11.141(443, 5989, 8080, 8182, 9080), 192.168.11.142(443, 5989, 8080, 8182, 9080), 192.168.11.143(443, 5989, 8080, 8182, 9080), 192.168.11.144(443, 5989, 8080, 8182, 9080), 192.168.11.151(443, 8080, 8182), 192.168.11.152(443, 5989, 8080, 8182, 9080), 192.168.11.153(443, 5989, 8080, 8182, 9080), 192.168.11.160(3389, 49275), 192.168.11.200(3389, 49202) | | |
| **Detail** | The remote service accepts connections encrypted using TLS 1.0. TLS 1.0 has a number of cryptographic design flaws. Modern implementations of TLS 1.0 mitigate these problems, but newer versions of TLS like 1.2 and 1.3 are designed against these flaws and should be used whenever possible. As of March 31, 2020, Endpoints that aren’t enabled for TLS 1.2 and higher will no longer function properly with major web browsers and major vendors. PCI DSS v3.2 requires that TLS 1.0 be disabled entirely by June 30, 2018, except for POS POI terminals (and the SSL/TLS termination points to which they connect) that can be verified as not being susceptible to any known exploits. | | |
| **Solution** | Enable support for TLS 1.2 and 1.3, and disable support for TLS 1.0. | | |
| **Remark** | https://tools.ietf.org/html/draft-ietf-tls-oldversions-deprecate-00 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 16 | **Finding** | SMB Signing not required |
| **Severity** | Medium | **Port** | TCP: 445 |
| **Target** | 192.168.8.13(445), 192.168.8.91(445), 192.168.8.160(445), 192.168.8.171(445), 192.168.8.211(445), 192.168.10.27(445), 192.168.10.220(445), 192.168.10.226(445), 192.168.11.160(445), 192.168.11.200(445) | | |
| **Detail** | Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server. | | |
| **Solution** | Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details. | | |
| **Remark** | http://www.nessus.org/u?df39b8b3 http://technet.microsoft.com/en-us/library/cc731957.aspx http://www.nessus.org/u?74b80723 https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html http://www.nessus.org/u?a3cac4ea | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 17 | **Finding** | SSLv3 Padding Oracle On Downgraded Legacy Encryption Vulnerability (POODLE) |
| **Severity** | Medium | **Port** | TCP: 1433, 49163, 49164, 49170 |
| **Target** | 192.168.8.13(1433), 192.168.8.91(49164), 192.168.8.160(49163), 192.168.8.171(49170) | | |
| **Detail** | The remote host is affected by a man-in-the-middle (MitM) information disclosure vulnerability known as POODLE. The vulnerability is due to the way SSL 3.0 handles padding bytes when decrypting messages encrypted using block ciphers in cipher block chaining (CBC) mode. MitM attackers can decrypt a selected byte of a cipher text in as few as 256 tries if they are able to force a victim application to repeatedly send the same data over newly created SSL 3.0 connections. As long as a client and service both support SSLv3, a connection can be 'rolled back' to SSLv3, even if TLSv1 or newer is supported by the client and service. The TLS Fallback SCSV mechanism prevents 'version rollback' attacks without impacting legacy clients; however, it can only protect connections when the client and service support the mechanism. Sites that cannot disable SSLv3 immediately should enable this mechanism. This is a vulnerability in the SSLv3 specification, not in any particular SSL implementation. Disabling SSLv3 is the only way to completely mitigate the vulnerability. | | |
| **Solution** | Disable SSLv3. Services that must support SSLv3 should enable the TLS Fallback SCSV mechanism until SSLv3 can be disabled. | | |
| **Remark** | https://www.imperialviolet.org/2014/10/14/poodle.html https://www.openssl.org/~bodo/ssl-poodle.pdf https://tools.ietf.org/html/draft-ietf-tls-downgrade-scsv-00 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 18 | **Finding** | MS16-047: Security Update for SAM and LSAD Remote Protocols (3148527) (Badlock) (uncredentialed check) |
| **Severity** | Medium | **Port** | TCP: 49155, 63422 |
| **Target** | 192.168.8.13(49155, 63422) | | |
| **Detail** | The remote Windows host is affected by an elevation of privilege vulnerability in the Security Account Manager (SAM) and Local Security Authority (Domain Policy) (LSAD) protocols due to improper authentication level negotiation over Remote Procedure Call (RPC) channels. A man-in-the-middle attacker able to intercept communications between a client and a server hosting a SAM database can exploit this to force the authentication level to downgrade, allowing the attacker to impersonate an authenticated user and access the SAM database. | | |
| **Solution** | Microsoft has released a set of patches for Windows Vista, 2008, 7, 2008 R2, 2012, 8.1, RT 8.1, 2012 R2, and 10. | | |
| **Remark** | http://www.nessus.org/u?52ade1e9 http://badlock.org/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 19 | **Finding** | ESXi 6.0 / 6.5 / 6.7 Multiple Vulnerabilities (VMSA-2019-0005) (Remote Check) |
| **Severity** | Medium | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443), 192.168.11.132(443), 192.168.11.134(443), 192.168.11.141(443), 192.168.11.142(443), 192.168.11.143(443), 192.168.11.144(443), 192.168.11.151(443), 192.168.11.152(443), 192.168.11.153(443) | | |
| **Detail** | The remote VMware ESXi host is version 6.0, 6.5, or 6.7 and is missing a security patch. It is, therefore, vulnerable to multiple vulnerabilities, including:  - An out-of-bounds read/write vulnerability and a Time-of-check  Time-of-use (TOCTOU) vulnerability in the virtual USB 1.1 UHCI  (Universal Host Controller Interface). Exploitation of these  issues requires an attacker to have access to a virtual machine  with a virtual USB controller present. These issues may allow a  guest to execute code on the host. (CVE-2019-5518, CVE-2019-5519) | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2019-0005.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 20 | **Finding** | ESXi 6.5 / 6.7 / 7.0 DoS (VMSA-2020-0018) |
| **Severity** | Medium | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443), 192.168.11.132(443), 192.168.11.134(443), 192.168.11.141(443), 192.168.11.142(443), 192.168.11.143(443), 192.168.11.144(443), 192.168.11.151(443), 192.168.11.152(443), 192.168.11.153(443) | | |
| **Detail** | The remote VMware ESXi host is version 6.5, 6.7 or 7.0 and is affected by a denial of service (DoS) vulnerability in the authentication service. An unauthenticated, remote attacker can exploit this issue to exhaust memory resources  resulting in a degradation of performance condition while the attack is sustained. Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2020-0018.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 21 | **Finding** | ESXi 5.5 / 6.0 / 6.5 / 6.7 Speculative Execution Side Channel Vulnerability (Foreshadow) (VMSA-2018-0020) (remote check) |
| **Severity** | Medium | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443) | | |
| **Detail** | The remote VMware ESXi host is version 5.5, 6.0, 6.5, or 6.7 and is missing a security patch. It is, therefore, vulnerable to a speculative execution side channel attack known as L1 Terminal Fault (L1TF). An attacker who successfully exploited L1TF may be able to read privileged data across trust boundaries. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2018-0020.html https://foreshadowattack.eu/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 22 | **Finding** | VMware ESXi 5.5 / 6.0 / 6.5 / 6.7 DoS (VMSA-2018-0018) (remote check) |
| **Severity** | Medium | **Port** | TCP: 443 |
| **Target** | 192.168.8.136(443), 192.168.8.166(443), 192.168.8.167(443), 192.168.8.168(443), 192.168.8.231(443), 192.168.8.232(443), 192.168.8.245(443), 192.168.8.246(443), 192.168.10.191(443), 192.168.10.193(443), 192.168.10.194(443) | | |
| **Detail** | The remote VMware ESXi host is version 5.5, 6.0, 6.5, or 6.7 and is missing a security patch. It is, therefore, vulnerable to a denial of service vulnerability. The vulnerability exists in the RPC handler due to a NULL pointer dereference issue. An authenticated, remote attacker can exploit this issue to cause VMs to stop responding. | | |
| **Solution** | Apply the appropriate patch as referenced in the vendor advisory. | | |
| **Remark** | https://www.vmware.com/security/advisories/VMSA-2018-0018.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 23 | **Finding** | SSH Weak Algorithms Supported |
| **Severity** | Medium | **Port** | TCP: 22 |
| **Target** | 192.168.8.17(22) | | |
| **Detail** | Nessus has detected that the remote SSH server is configured to use the Arcfour stream cipher or no cipher at all. RFC 4253 advises against using Arcfour due to an issue with weak keys. | | |
| **Solution** | Contact the vendor or consult product documentation to remove the weak ciphers. | | |
| **Remark** | https://tools.ietf.org/html/rfc4253#section-6.3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 24 | **Finding** | SSH Server CBC Mode Ciphers Enabled |
| **Severity** | Low | **Port** | TCP: 22 |
| **Target** | 192.168.8.17(22), 192.168.8.136(22), 192.168.8.138(22), 192.168.8.166(22), 192.168.8.167(22), 192.168.8.168(22), 192.168.8.231(22), 192.168.8.232(22), 192.168.8.245(22), 192.168.8.246(22), 192.168.10.221(22), 192.168.10.222(22), 192.168.10.224(22), 192.168.10.225(22), 192.168.11.151(22) | | |
| **Detail** | The SSH server is configured to support Cipher Block Chaining (CBC) encryption. This may allow an attacker to recover the plaintext message from the ciphertext.   Note that this plugin only checks for the options of the SSH server and does not check for vulnerable software versions. | | |
| **Solution** | Contact the vendor or consult product documentation to disable CBC mode cipher encryption, and enable CTR or GCM cipher mode encryption. | | |
| **Remark** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 25 | **Finding** | SSH Weak Key Exchange Algorithms Enabled |
| **Severity** | Low | **Port** | TCP: 22 |
| **Target** | 192.168.8.17(22), 192.168.10.221(22), 192.168.10.222(22), 192.168.10.224(22), 192.168.10.225(22) | | |
| **Detail** | The remote SSH server is configured to allow key exchange algorithms which are considered weak. This is based on the IETF draft document Key Exchange (KEX) Method Updates and Recommendations for Secure Shell (SSH) draft-ietf-curdle-ssh-kex-sha2-20. Section 4 lists guidance on key exchange algorithms that SHOULD NOT and MUST NOT be enabled. This includes:  diffie-hellman-group-exchange-sha1  diffie-hellman-group1-sha1  gss-gex-sha1-\*  gss-group1-sha1-\*  gss-group14-sha1-\*  rsa1024-sha1 Note that this plugin only checks for the options of the SSH server, and it does not check for vulnerable software versions. | | |
| **Solution** | Contact the vendor or consult product documentation to disable the weak algorithms. | | |
| **Remark** | http://www.nessus.org/u?b02d91cd https://datatracker.ietf.org/doc/html/rfc8732 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 26 | **Finding** | SSL Certificate Chain Contains RSA Keys Less Than 2048 bits |
| **Severity** | Low | **Port** | TCP: 1433 |
| **Target** | 192.168.8.13(1433) | | |
| **Detail** | At least one of the X.509 certificates sent by the remote host has a key that is shorter than 2048 bits. According to industry standards set by the Certification Authority/Browser (CA/B) Forum, certificates issued after January 1, 2014 must be at least 2048 bits. Some browser SSL implementations may reject keys less than 2048 bits after January 1, 2014. Additionally, some SSL certificate vendors may revoke certificates less than 2048 bits before January 1, 2014. Note that Nessus will not flag root certificates with RSA keys less than 2048 bits if they were issued prior to December 31, 2010, as the standard considers them exempt. | | |
| **Solution** | Replace the certificate in the chain with the RSA key less than 2048 bits in length with a longer key, and reissue any certificates signed by the old certificate. | | |
| **Remark** | https://www.cabforum.org/wp-content/uploads/Baseline\_Requirements\_V1.pdf | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 27 | **Finding** | SSH Weak MAC Algorithms Enabled |
| **Severity** | Low | **Port** | TCP: 22 |
| **Target** | 192.168.8.17(22) | | |
| **Detail** | The remote SSH server is configured to allow either MD5 or 96-bit MAC algorithms, both of which are considered weak. Note that this plugin only checks for the options of the SSH server, and it does not check for vulnerable software versions. | | |
| **Solution** | Contact the vendor or consult product documentation to disable MD5 and 96-bit MAC algorithms. | | |
| **Remark** | - | | |



# Web Application Vulnerability Assessment

**Vulnerability Assessment from Public Access (for public target)**

**Testing date:** March 30, 2021

**Tester IP Address:** 203.150.79.252

Diagram

Description automatically generated

Figure 5: Vulnerability Assessment from Public Access

## **6.1 Target Information**

| **No.** | **Domain / Server Name** | **IP Address** | **OS/Model** | **Port** |
| --- | --- | --- | --- | --- |
| 1 | https://example.com | 123.123.123.123 | Ubuntu 20 | TCP 22, 53, 80, 113, 123, 443, 2000, 4118, 4119, 4120, 4121, 4122, 4444, 5000, 5060, 8008, 8082 |

## **6.2 Executive summary**

The purpose of this activity is to find the vulnerability on the target web application.

### **6.2.1 Summary Vulnerability by Severity**

Figure 6: Summary by Severity of Web Application Vulnerability Assessment

### **6.2.2 Vulnerability by Target**

| **No.** | **Domain/Server Name** | **IP Address** | **Critical** | **High** | **Medium** | **Low** | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | https://example.com | 123.123.123.123 | 0 | 0 | 1 | 3 | 4 |
| **Total** | | | **0** | **0** | **1** | **3** | **4** |

## **6.3 Web Application Vulnerability Detail**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 1 | **Finding** | Clickjacking: X-Frame-Options header |
| **Severity** | **Low** | **Port** | TCP 443 |
| **Target** | https://example.com/  https://example.com/sitemap.xml  https://example.com/sitemap.xml.gz  https://example.com/login  https://example.com/backend/  https://example.com/backend/api/v1/  https://example.com/backend/api/  https://example.com/backend.bak  https://example.com/backend.7z  https://example.com/backend.cfg  https://example.com/backend.csv  https://example.com/backend.dump  https://example.com/backend.ini  https://example.com/backend.jar  https://example.com/backend.old  https://example.com/backend.ost  https://example.com/backend.pst  https://example.com/backend.sh  https://example.com/backend.sln  https://example.com/backend.tar  https://example.com/backend.war | | |
| **Detail** | Clickjacking (User Interface redress attack, UI redress attack, UI redressing) is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information, or taking control of their computer while clicking on seemingly innocuous web pages.  The server did not return an X-Frame-Options header with the value DENY or SAMEORIGIN, which means that this website could be at risk of a clickjacking attack. The X-Frame-Options HTTP response header can be used to indicate whether a browser should be allowed to render a page inside a frame or iframe. Sites can use this to avoid clickjacking attacks, by ensuring that their content is not embedded into untrusted sites. | | |
| **Impact** | The impact depends on the affected web application. | | |
| **Solution** | Configure your web server to include an X-Frame-Options header and a CSP header with frame-ancestors directive. Consult Web references for more information about the possible values for this header. | | |
| **Remark** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 2 | **Finding** | HTTP Strict Transport Security (HSTS) not implemented |
| **Severity** | **Low** | **Port** | TCP 443 |
| **Target** | https://example.com/  https://example.com/sitemap.xml  https://example.com/sitemap.xml.gz  https://example.com/login  https://example.com/backend/  https://example.com/backend/api/v1/  https://example.com/backend/api/  https://example.com/backend.bak  https://example.com/backend.7z  https://example.com/backend.cfg  https://example.com/backend.csv  https://example.com/backend.dump  https://example.com/backend.ini  https://example.com/backend.jar  https://example.com/backend.old  https://example.com/backend.ost  https://example.com/backend.pst  https://example.com/backend.sh  https://example.com/backend.sln  https://example.com/backend.tar  https://example.com/backend.war | | |
| **Detail** | HTTP Strict Transport Security (HSTS) tells a browser that a web site is only accessable using HTTPS. It was detected that your web application doesn't implement HTTP Strict Transport Security (HSTS) as the Strict Transport Security header is missing from the response. | | |
| **Impact** | HSTS can be used to prevent and/or mitigate some types of man-in-the-middle (MitM) attacks | | |
| **Solution** | It's recommended to implement HTTP Strict Transport Security (HSTS) into your web application. Consult web references for more information | | |
| **Remark** | |  | | --- | | https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 3 | **Finding** | Sensitive pages could be cached |
| **Severity** | **Low** | **Port** | TCP 443 |
| **Target** | https://example.com/?password=g00dPa$$w0rDusername=pHqghUme | | |
| **Detail** | One or more pages contain possible sensitive information (e.g., a password parameter) and could be potentially cached. Even in secure SSL channels sensitive data could be stored by intermediary proxies and SSL terminators. To prevent this, a Cache-Control header should be specified. | | |
| **Impact** | Possible sensitive information disclosure. | | |
| **Solution** | Prevent caching by adding "Cache Control: No-store" and "Pragma: no-cache" to the HTTP response header. | | |
| **Remark** | - | | |

# Port Discovery

| **Port** | **Protocol** | **Service** |
| --- | --- | --- |
| 22 | TCP | ssh |
| 80 | TCP | http |
| 110 | TCP | pop3 |
| 143 | TCP | imap-proxy |
| 443 | TCP | https |
| 465 | TCP | ssl/smtp |
| 587 | TCP | smtp |
| 993 | TCP | ssl/imap-proxy |
| 995 | TCP | ssl/pop3 |
| 8443 | TCP | https-alt? |
| 9071 | TCP | ssl/http |

# Appendix

## **8.1 About Nessus**

Nessus is a proprietary vulnerability scanner developed by Tenable, Inc. Nessus is trusted by more than 30,000 organizations worldwide as one of the most widely deployed security technologies on the planet - and the gold standard for vulnerability assessment.

Reference: https://www.tenable.com/products/nessus

### **8.1.1 Nessus vulnerabilities**

As information about new vulnerabilities are discovered and released into the public domain, Tenable, Inc. research staff designs programs to enable Nessus to detect them. These programs are named plugins, and are written in the Nessus proprietary scripting language, called Nessus Attack Scripting Language (NASL). Plugins contain vulnerability information, a generic set of remediation actions, and the algorithm to test for the presence of the security issue.

Reference: https://www.tenable.com/plugins

### **8.1.2 Nessus risk score**

There are four risk levels in this document: Critical, High, Medium, and Low. There are methods for determining the risk level. Based on the Common Vulnerability Scoring System (CVSS), a standard for assessing the severity of vulnerabilities in computer systems. Regarded by the NIAC (National Infrastructure Advisory Council), expert assessments are measured in a range of 0 – 10

| **Severity** | **Description** | **Score** |
| --- | --- | --- |
| Critical | Vulnerabilities that score in the critical range usually have most of the following characteristics:   * Exploitation of the vulnerability likely results in root-level compromise of servers or infrastructure devices. * Exploitation is usually straightforward, in the sense that the attacker does not need any special authentication credentials or knowledge about individual victims, and does not need to persuade a target user, for example via social engineering, into performing any special functions.   For critical vulnerabilities, is advised that you patch or upgrade as soon as possible, unless you have other mitigating measures in place. For example, a mitigating factor could be if your installation is not accessible from the Internet. | 9.0 – 10.0 |
| High | Vulnerabilities that score in the high range usually have some of the following characteristics:   * The vulnerability is difficult to exploit. * Exploitation could result in elevated privileges. * Exploitation could result in a significant data loss or downtime. | 7.0 – 8.9 |
| Medium | Vulnerabilities that score in the medium range usually have some of the following characteristics:   * Vulnerabilities that require the attacker to manipulate individual victims via social engineering tactics. * Denial of service vulnerabilities that are difficult to set up. * Exploits that require an attacker to reside on the same local network as the victim. * Vulnerabilities where exploitation provides only very limited access. * Vulnerabilities that require user privileges for successful exploitation. | 4.0 – 6.9 |
| Low | Vulnerabilities in the low range typically have very little impact on an organization's business. Exploitation of such vulnerabilities usually requires local or physical system access. | 0.1 – 3.9 |

## **8.2 About Acunetix**

Acunetix by Invicti Security is an application security testing tool built to help small mid-size organizations around the world take control of their web security. Acunetix is built to evolve and stay ahead of cybersecurity changes. Acunetix industry-leading dynamic and interactive application security testing (DAST and IAST) technology automates vulnerability management and empowers security teams to uncover more vulnerabilities, reduce false positives, increase productivity, and simplify remediation efforts.

Reference: https://www.acunetix.com/product/, https://www.acunetix.com/about/

### **8.2.1 Acunetix web vulnerabilities**

The following reference link is a list of known web application vulnerabilities that can be automatically detected by Acunetix.

Reference: https://www.acunetix.com/vulnerabilities/web/

### **8.2.2 Acunetix risk score**

Severity is a metric for classifying the level of risk which a security vulnerability poses. The severity level of a vulnerability is assigned based on the security risk posed to an organization should the vulnerability be exploited, as well as the degree of difficulty involved in exploiting it. The result of a successful attack by exploiting a vulnerability could vary from denial of service and information disclosure to a complete compromise of applications or systems. The following provides a description of what the results in this analysis consider to be the impact of each vulnerability severity level.

| **Severity** | **Description** |
| --- | --- |
| High | An attacker can **fully** compromise the confidentiality, integrity, or availability, of a target system without specialized access, user interaction or circumstances that are beyond the attacker’s control. Very likely to allow lateral movement and escalation of attack to other systems on the internal network of the vulnerable application. |
| Medium | An attacker can **partially** compromise the confidentiality, integrity, or availability, of a target system. Specialized access, user interaction, or circumstances that are beyond the attacker’s control may be required for an attack to succeed. Very likely to be used in conjunction with other vulnerabilities to escalate an attack. |
| Low | An attacker can **limitedly** compromise the confidentiality, integrity, or availability, of a target system. Specialized access, user interaction, or circumstances that are beyond the attacker’s control is required for an attack to succeed. Needs to be used in conjunction with other vulnerabilities to escalate an attack. |