**Vulnerability Assessment Report**

**For**



**SAC\_Revisit\_Private**

**March 22, 2022**

**Document Security Level:** Confidential

**Document Version:** 1.0

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Edit Report** | **Editor** |
| 1.0 | March 22, 2022 | Creation | INET Managed Services |

|  |  |  |  |
| --- | --- | --- | --- |
| SAC\_Revisit\_Private | | 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](#_Toc98507702)

[2. Operation Method 5](#_Toc98507703)

[3. Project Scope 6](#_Toc98507704)

[3.1 Infrastructure Vulnerability Assessment 6](#_Toc98507705)

[4. Testing Tools 7](#_Toc98507706)

[5. Infrastructure Vulnerability Assessment 7](#_Toc98507707)

[5.1 Target Information 8](#_Toc98507708)

[5.2 Executive summary 9](#_Toc98507709)

[5.2.1 Summary Vulnerability by Severity 9](#_Toc98507710)

[5.2.2 Vulnerability by Target 9](#_Toc98507711)

[5.3 Infrastructure Vulnerability Detail 10](#_Toc98507712)

[6. Appendix 11](#_Toc98507713)

[6.1 About Nessus 11](#_Toc98507714)

[6.1.1 Nessus vulnerabilities 11](#_Toc98507715)

[6.1.2 Nessus risk score 11](#_Toc98507716)

[6.2 About Burp Suite's web vulnerability scanner 13](#_Toc98507717)

[6.2.1 Burp Suite's web vulnerability scanner risk score 13](#_Toc98507718)

# 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 | - | - | - | - | - | - | ✓ |

# Testing Tools

|  |  |
| --- | --- |
| **Tool Name** | **Testing Type** |
| Nmap | Host and Service Discovery |
| Nessus Professional | Infrastructure Vulnerability Assessment |
| Burp Suite's web vulnerability scanner | Web Application Vulnerability Assessment |

# Infrastructure Vulnerability Assessment

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

**Testing date:** <<Date SCAN>>

**Tester IP Address:** <<IP Target>>

Diagram

Description automatically generated

Figure 2: Vulnerability Assessment from Public Access

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

**Testing date:** <<Date SCAN>>

**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 | - | 10.1.0.10 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 2 | - | 10.1.0.20 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 3 | - | 10.1.0.21 | - | TCP: 22, 80, 113, 443, 3306, 8008, 8010 |
| 4 | - | 10.1.0.25 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 5 | - | 10.1.0.251 | - | TCP: 80 |
| 6 | - | 10.1.0.254 | - | TCP: 113 |
| 7 | - | 10.1.1.2 | - | TCP: 53, 80, 113, 443, 8008, 8010 |
| 8 | - | 10.1.1.11 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 9 | - | 10.1.1.18 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 10 | - | 10.1.1.19 | - | TCP: 22, 80, 113, 443, 8008, 8010 |
| 11 | - | 10.1.1.244 | - | TCP: 22, 80, 113, 443, 3306, 8008, 8010 |

## **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**

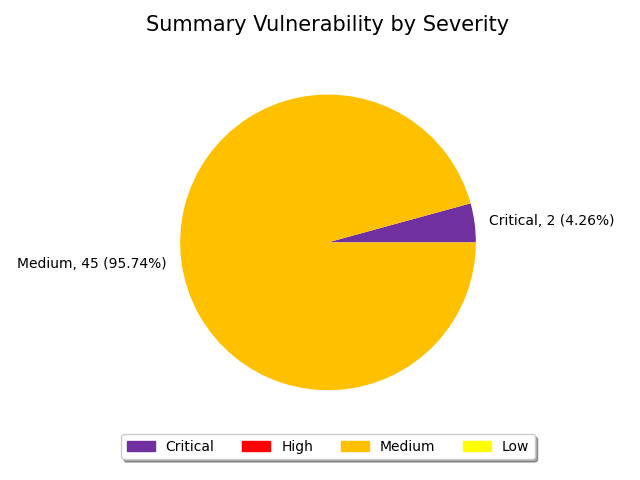


Figure 4: Summary by Severity of Infrastructure Vulnerability Assessment

### **5.2.2 Vulnerability by Target**

| **No.** | **Domain/Server Name** | **IP Address** | **Critical** | **High** | **Medium** | **Low** | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | - | 10.1.0.10 | 1 | 0 | 3 | 0 | 4 |
| 2 | - | 10.1.0.20 | 0 | 0 | 5 | 0 | 5 |
| 3 | - | 10.1.0.21 | 0 | 0 | 4 | 0 | 4 |
| 4 | - | 10.1.0.25 | 0 | 0 | 3 | 0 | 3 |
| 5 | - | 10.1.0.251 | 0 | 0 | 3 | 0 | 3 |
| 6 | - | 10.1.1.2 | 0 | 0 | 5 | 0 | 5 |
| 7 | - | 10.1.1.10 | 1 | 0 | 4 | 0 | 5 |
| 8 | - | 10.1.1.11 | 0 | 0 | 3 | 0 | 3 |
| 9 | - | 10.1.1.18 | 0 | 0 | 3 | 0 | 3 |
| 10 | - | 10.1.1.19 | 0 | 0 | 3 | 0 | 3 |
| 11 | - | 10.1.1.121 | 0 | 0 | 6 | 0 | 6 |
| 12 | - | 10.1.1.244 | 0 | 0 | 3 | 0 | 3 |
| **Total** | | | 2 | 0 | 45 | 0 | 47 |

## **5.3 Infrastructure Vulnerability Detail**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 1 | **Finding** | Unix Operating System Unsupported Version Detection |
| **Severity** | **Critical** | **Port** | TCP: 0 |
| **Target** | 10.1.0.10(0), 10.1.1.10(0) | | |
| **Detail** | According to its self-reported version number, the Unix operating system running 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 the Unix operating system that is currently supported. | | |
| **Remark** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 2 | **Finding** | SSL Certificate Cannot Be Trusted |
| **Severity** | **Medium** | **Port** | TCP: 3389, 8010 |
| **Target** | 10.1.0.10(8010), 10.1.0.20(8010), 10.1.0.21(8010), 10.1.0.25(8010), 10.1.0.251(8010), 10.1.1.2(8010), 10.1.1.10(8010), 10.1.1.11(8010), 10.1.1.18(8010), 10.1.1.19(8010), 10.1.1.121(3389, 8010), 10.1.1.244(8010) | | |
| **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.** | 3 | **Finding** | SSL Self-Signed Certificate |
| **Severity** | **Medium** | **Port** | TCP: 3389, 8010 |
| **Target** | 10.1.0.10(8010), 10.1.0.20(8010), 10.1.0.21(8010), 10.1.0.25(8010), 10.1.0.251(8010), 10.1.1.2(8010), 10.1.1.10(8010), 10.1.1.11(8010), 10.1.1.18(8010), 10.1.1.19(8010), 10.1.1.121(3389, 8010), 10.1.1.244(8010) | | |
| **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.** | 4 | **Finding** | TLS Version 1.0 Protocol Detection |
| **Severity** | **Medium** | **Port** | TCP: 8010 |
| **Target** | 10.1.0.10(8010), 10.1.0.20(8010), 10.1.0.21(8010), 10.1.0.25(8010), 10.1.0.251(8010), 10.1.1.2(8010), 10.1.1.10(8010), 10.1.1.11(8010), 10.1.1.18(8010), 10.1.1.19(8010), 10.1.1.121(8010), 10.1.1.244(8010) | | |
| **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.** | 5 | **Finding** | Apache Server ETag Header Information Disclosure |
| **Severity** | **Medium** | **Port** | TCP: 80 |
| **Target** | 10.1.0.20(80), 10.1.0.21(80) | | |
| **Detail** | The remote web server is affected by an information disclosure vulnerability due to the ETag header providing sensitive information that could aid an attacker, such as the inode number of requested files. | | |
| **Solution** | Modify the HTTP ETag header of the web server to not include file inodes in the ETag header calculation. Refer to the linked Apache documentation for more information. | | |
| **Remark** | http://httpd.apache.org/docs/2.2/mod/core.html#FileETag | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 6 | **Finding** | HSTS Missing From HTTPS Server (RFC 6797) |
| **Severity** | **Medium** | **Port** | TCP: 8010 |
| **Target** | 10.1.0.20(8010), 10.1.1.2(8010), 10.1.1.10(8010), 10.1.1.121(8010) | | |
| **Detail** | The remote web server is not enforcing HSTS, as defined by RFC 6797.  HSTS is an optional response header that can be configured on the server to instruct  the browser to only communicate via HTTPS. The lack of HSTS allows downgrade attacks, SSL-stripping man-in-the-middle attacks, and weakens cookie-hijacking protections. | | |
| **Solution** | Configure the remote web server to use HSTS. | | |
| **Remark** | https://tools.ietf.org/html/rfc6797 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID.** | 7 | **Finding** | DNS Server Cache Snooping Remote Information Disclosure |
| **Severity** | **Medium** | **Port** | UDP: 53 |
| **Target** | 10.1.1.2(53) | | |
| **Detail** | The remote DNS server responds to queries for third-party domains that do not have the recursion bit set.   This may allow a remote attacker to determine which domains have recently been resolved via this name server, and therefore which hosts have been recently visited.   For instance, if an attacker was interested in whether your company utilizes the online services of a particular financial institution, they would be able to use this attack to build a statistical model regarding company usage of that financial institution. Of course, the attack can also be used to find B2B partners, web-surfing patterns, external mail servers, and more.  Note: If this is an internal DNS server not accessible to outside networks, attacks would be limited to the internal network. This may include employees, consultants and potentially users on a guest network or WiFi connection if supported. | | |
| **Solution** | Contact the vendor of the DNS software for a fix. | | |
| **Remark** | http://cs.unc.edu/~fabian/course\_papers/cache\_snooping.pdf | | |



# Appendix

## **6.1 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

### **6.1.1 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

### **6.1.2 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 |

## **6.2 About Burp Suite's web vulnerability scanner**

The web vulnerability scanner behind Burp Suite's popularity has more to it than most. Burp Scanner uses PortSwigger's world-leading research to help its users find a wide range of vulnerabilities in web applications, automatically. Sitting at the core of both Burp Suite Enterprise Edition and Burp Suite Professional, Burp Scanner is the weapon of choice for over 60,000 users across more than 15,000 organizations.

Reference: https://portswigger.net/burp/vulnerability-scanner

### **6.2.1 Burp Suite's web vulnerability scanner risk score**

The level of severity for an issue that was found by a scan. The higher the severity level, the larger the impact is likely to be if an attacker is able to exploit this vulnerability. Note that the severity level is only a rough approximation based on a typical website. You should use your knowledge of the purpose and context of the associated functionality to determine how serious each issue is in your individual case.

Reference: https://portswigger.net/burp/extensibility/enterprise/graphql-api/severity.html

| **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. |