Final Report
Network Penetration Test

For: ACME, Inc.

Submitted by:

MainNerve, LLC
201 E Pikes Peak Ave Suite 2025
Colorado Springs, CO 80903
www.mainnerve.com

September 11, 2023

Table of Contents

Summary of Changes **1**

1. How to Use This Report **2**

2. Executive Summary **3**

3. Scope of Services **4**

4. Network Penetration Testing Overview **4**

4.1 Testing Methodology 4

5. Primary Findings for External Testing **5**

5.1 Time-based Blind SQL Injection 5

5.2 Brute Force Authentication 9

5.3 Username Enumeration 13

6. Primary Findings for Internal Testing **15**

6.1 Dell OpenManage Server Administrator Authentication Bypass 15

6.2 Multiple Default Credentials 16

6.3 IPMI Hash Disclosure 19

6.4 End-of-life Windows Server 2008 r2 20

6.5 Out-of-date DNS Server (6.1.7601) 22

6.6 Out-of-date iDRAC Versions 22

6.7 End-of-life MSSQL Server Version 2014 12.00.5223 24

6.8 End-of-life Oracle Server Version 12.2.0.1.0 25

6.9 No Password Set for Admin 26

6.10 Out-of-date SSLv3 28

6.11 SNMP Public Default 29

7. Testing Tools **30**

8. Risk Rating Overview **31**

8.1 Risk Calculation 31

8.2 Risk Impact Rating 32

Summary of Changes

Change Information	Reason for Change	Date
Version 1.0	Initial Release	9/11/2023

1. How to Use This Report

This is the final report for the network penetration test performed by MainNerve. This document was prepared in accordance with security best practices published by organizations such as NIST. Remediation recommendations, where applicable, are found in line within each section. These recommendations specifically address the security concerns discussed in the respective sections. Wherever possible, screenshots are included to demonstrate methods and findings encountered during the period of performance of this test.

MainNerve has also provided a risk rating that describes the impact of each vulnerability or misconfiguration discovered during this test. This rating does not cover vulnerability scans or other assessments (if applicable) not directly part of this network penetration test. MainNerve used the rating that is based on the DREAD Threat Risk Modeling algorithm and is intended to assist with determining a proper course of action when performing remediation or analysis of the findings.

This report describes penetration testing that represents a point-in-time snapshot of the network security posture of the systems in question. In accordance with security best practices, regular security assessments should be commissioned especially after major changes to systems and/or networks.

2. Executive Summary

MainNerve completed penetration testing for ACME, Inc. ("Customer") on the external (Internet-facing) and internal (local area) networks. Using an approach found in NIST SP 800-115, MainNerve performed live host discovery, service enumeration, vulnerability scanning, and verification. Penetration testing was comprised primarily of manual testing. That is, a MainNerve penetration tester used hands-on methods while minimizing automated scanning to test the target systems. The goal of the testing was to identify vulnerabilities or misconfigurations inherent on the systems and attempt to exploit those vulnerabilities.

For this penetration test, MainNerve identified the hosts through various probes and service scans using active host discovery. Active host discovery involves attempts to interact with services exposed to the Internet or advertised within the internal network using specialized tools, then analyzing the results for weaknesses.

Over the course of the penetration test, it was discovered that the external system is vulnerable to time-based blind SQL injection. This means that an unauthenticated user is able to send crafted requests to a web application in order to execute SQL queries, which could steal information or execute code. It was also discovered that several weak credentials are being utilized, allowing for brute force attacks. Lastly, there are different messages for valid and invalid users for the password reset function, allowing for username enumeration.

The internal scope has several high severity vulnerabilities, including out-of-date software and operating systems which are susceptible to various publicly disclosed CVEs. There are also multiple instances of default credentials implemented, as well as several applications that have no password set. SSLv3 is running which is unsatisfactory for encryption, and one system is using SNMP with a public default string allowing information disclosure.

Based on thorough testing and analysis, MainNerve has determined the overall threat risk is High. This risk impact rating is based on the DREAD Threat Risk Modeling algorithm.

3. Scope of Services

The following services are described in this report. They were performed remotely from MainNerve's facility in Colorado Springs, Colorado and/or secure cloud server hosted by Cybercon.

Services	Scope
External Network Penetration Testing	304.129.75.785
Internal Network Penetration Testing	172.16.309.0/24 172.16.319.0/24 10.2.304.0/24

4. Network Penetration Testing Overview

Network penetration testing was performed on the systems identified in the previous paragraph. The goal of this test was to reveal vulnerabilities or misconfigurations that could be used to permit unauthorized access by malicious users and gain access to data hosted by the systems in question.

This penetration test, while thorough in nature, is considered a holistic view of the security of the system tested. It should not be considered absolute in nature as restrictions on time, economics, and resources can contribute to a limited perspective that an assessment of this type provides. Security should continue to be monitored beyond this penetration test.

4.1 Testing Methodology

For network-based testing and assessments, MainNerve employs a testing approach based on NIST SP 800-115, *Technical Guide to Information Security Testing and Assessment.* The publication describes methods for performing detailed assessments by providing a methodology for network enumeration, vulnerability identification, and documentation of results. A methodology based on this publication is one that divides penetration testing into four phases: 1) Planning, 2) Discovery, 3) Attack, and 4) Reporting. The details of each phase can be found in the project proposal and rules of engagement.

During the Planning phase of the penetration test, MainNerve and the customer established the dates, times, and specifc rules that govern the penetration test. All of the details were included in a rules of engagement. This penetration test involved the following:

- Identify services exposed to the Internet
- Reveal vulnerabilities using network vulnerability scanning tools
- Attempts to access sensitive data by attacking vulnerable system
- Use ethical hacking to exploit the most critical vulnerabilities or misconfigurations, if any
- Provide remediation recommendations as appropriate

5. Primary Findings for External Testing

MainNerve performed both manual and automated scanning in order to determine the accessibility of the external system. Using service and discovery scans with *nmap* and *Nessus Professional,* the system was scanned to reveal the types of services exposed to the Internet.

IP Addresses	Hostname/DNS	Notes
304.129.75.785	https://example.com	PORT STATE SERVICE 443/tcp open https

Table 1 – External system and respective services detected by MainNerve

5.1 Time-based Blind SQL Injection

Current Rating
HIGH

It was discovered through manual and automated testing that the web application is vulnerable to SQL injection. SQL injection vulnerabilities arise when user-controllable data is incorporated into database SQL queries in an unsafe manner. An attacker can supply crafted input to break out of the data context in which their input appears and interfere with the structure of the surrounding query.

A wide range of damaging attacks can often be delivered via SQL injection, including reading or modifying critical application data, interfering with application logic, escalating privileges within the database, and taking control of the database server.

In this case, the penetration tester was able to enumerate all databases and tables without authentication. The following URL and parameters are vulnerable to SQL injection.

https://example.com/EmployeeSelfService/Account/Register?class=form-horizontal
(AccountSystemEmployeeNo parameter)

Below are screenshots (Figures 1-2) which depict discovery of the vulnerability using Burp Suite Professional's repeater. Proxied traffic was modified to include select statements including a sleep command for 5, and 20 seconds. As demonstrated, the web server times out at 5.296 seconds and 20.31 seconds. This was performed multiple times to ensure reliability of the results.

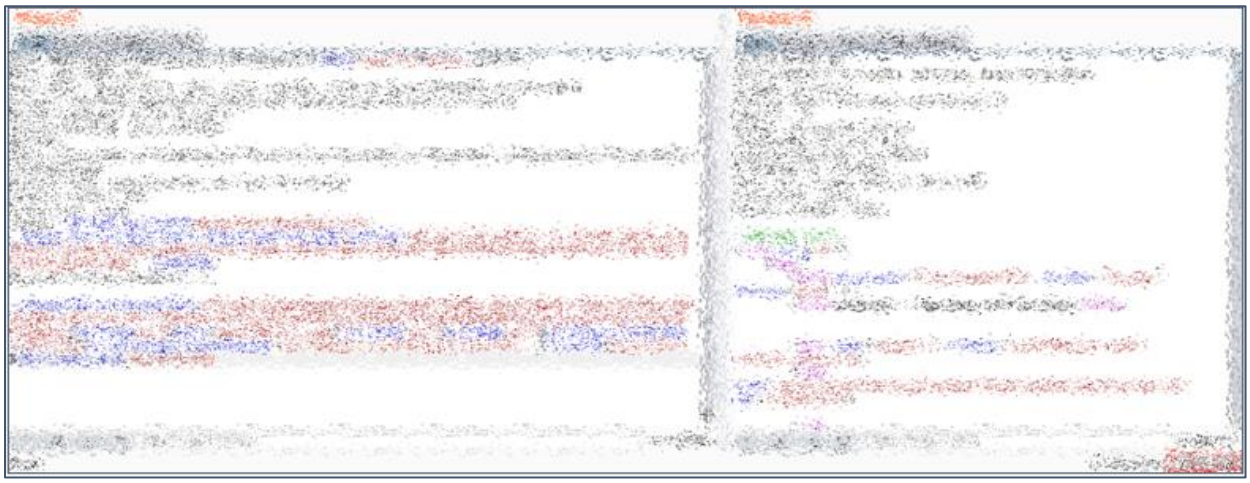

Figure 1 – Manual confirmation of SQL injection

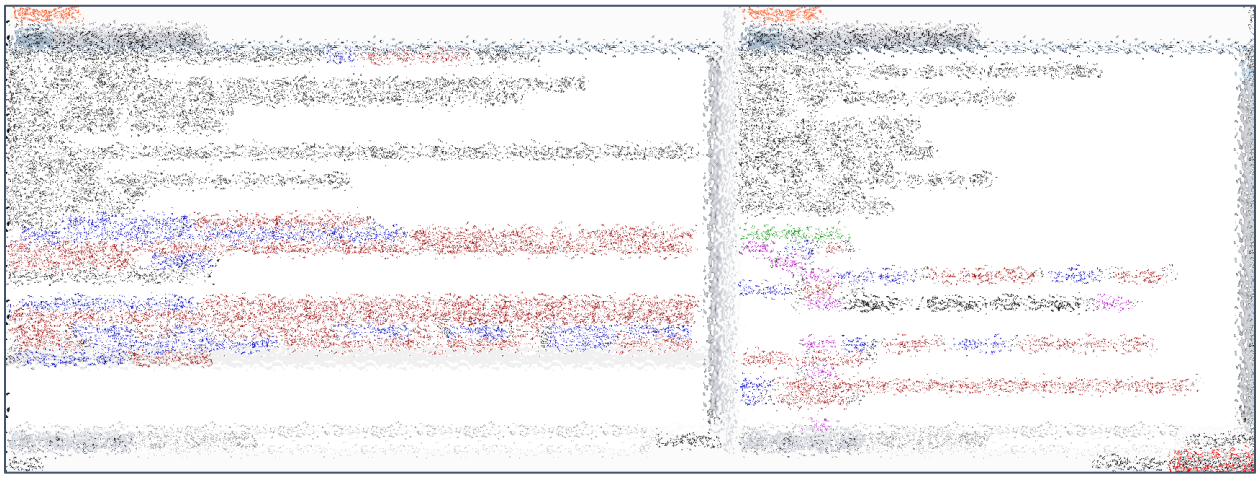

Figure 2 – Manual confirmation of SQL injection

The above process was automated by setting the injection points within Sqlmap and running it against the web application. This provided proof that it is possible to enumerate the databases, tables, and columns to extract sensitive information (see Figures 3-6).

Figure 3 – Sqlmap exploitation

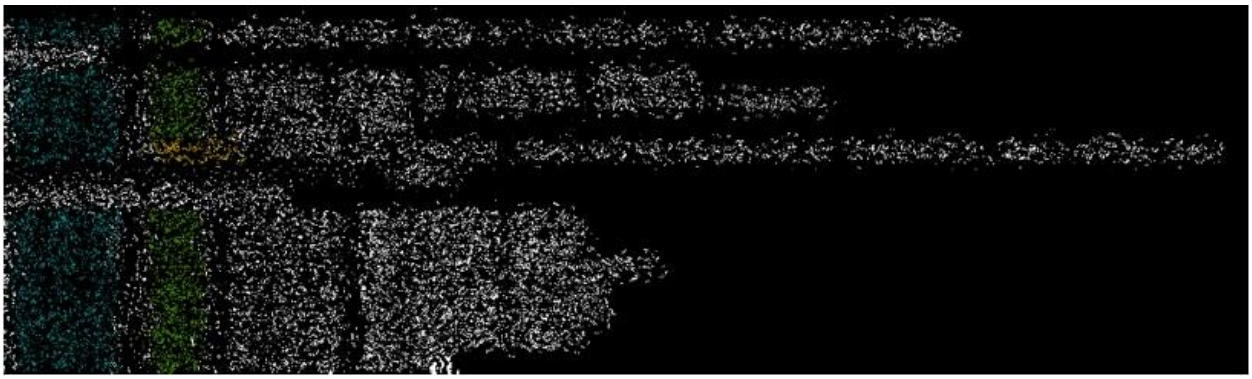

Figure 4 – Sqlmap exploitation

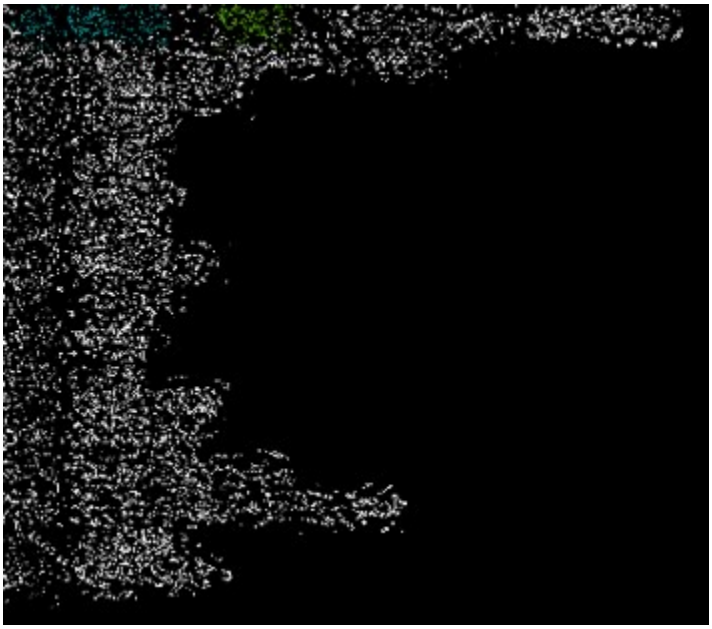

Figure 5 – Sqlmap exploitation

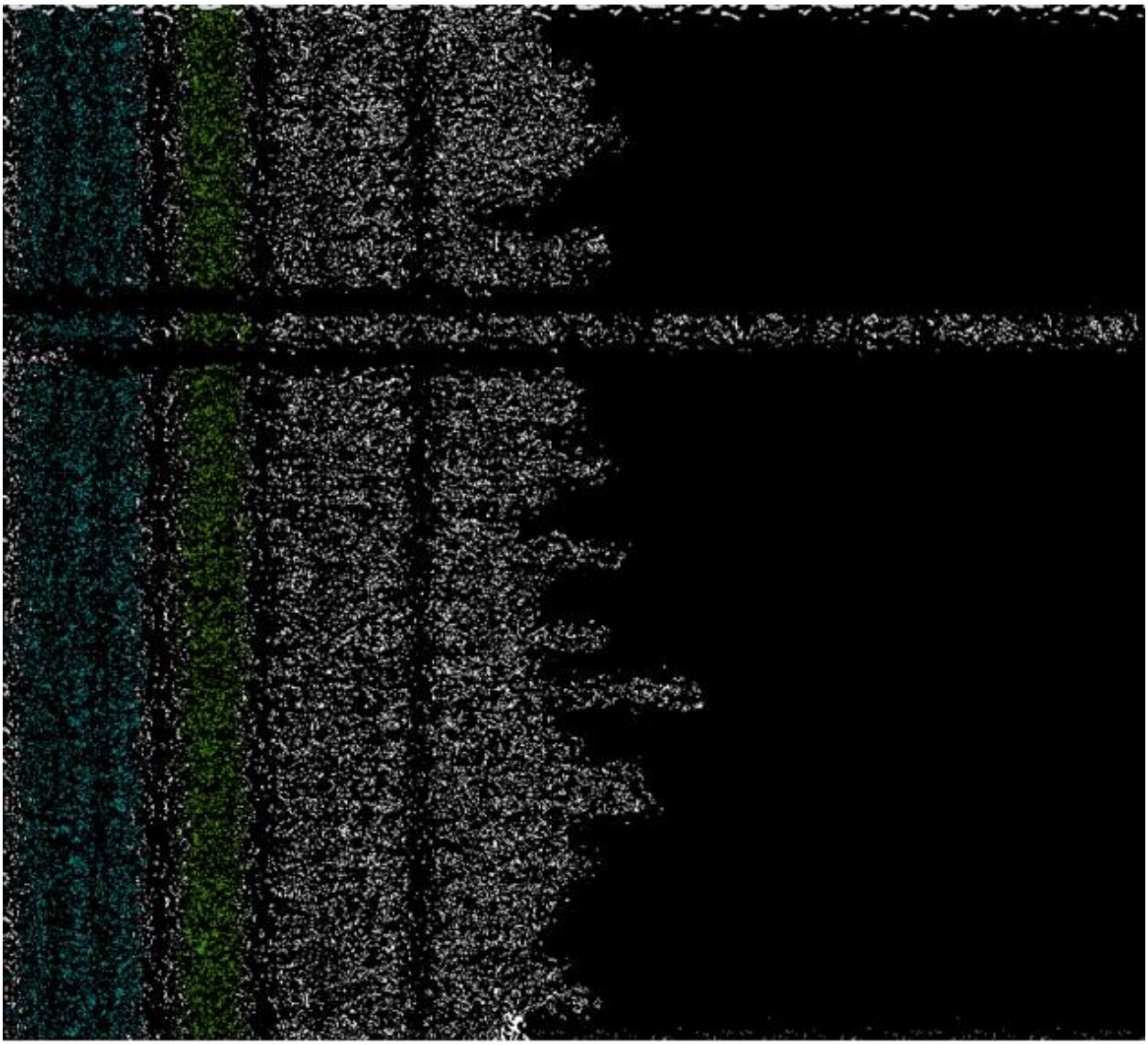

Figure 6 – Sqlmap exploitation

Recommendation:

The Microsoft SQL (MSSQL) configuration should be thoroughly analyzed in order to isolate the root cause of the SQL injection vulnerability, and ensure that it is swiftly remediated. It is also recommended that all database information is reviewed, and PostgreSQL logs are analyzed, as there is a chance that the database may have already been compromised.

The most effective way to prevent SQL injection attacks is to use parameterized queries (also known as prepared statements) for all database access. This method uses two steps to incorporate potentially tainted data into SQL queries: first, the application specifies the structure of the query, leaving placeholders for each item of user input; second, the application specifies the contents of each placeholder. Because the structure of the query has already been defined in the first step, it

is not possible for malformed data in the second step to interfere with the query structure. Review the documentation for the database and application platform to determine the appropriate APIs which can be used to perform parameterized queries. It is strongly recommended that MSSQL is configured to parameterize every variable data item that is incorporated into database queries, even if it is not obviously tainted, to prevent oversights occurring and avoid vulnerabilities being introduced by changes elsewhere within the code base of the application.

Some commonly employed and recommended mitigations for SQL injection vulnerabilities are not always effective:

- One common defense is to double up any single quotation marks appearing within user input before incorporating that input into a SQL query. This defense is designed to prevent malformed data from terminating the string into which it is inserted. However, if the data being incorporated into queries is numeric, then the defense may fail, because numeric data may not be encapsulated within quotes, in which case only a space is required to break out of the data context and interfere with the query. Further, in second-order SQL injection attacks, data that has been safely escaped when initially inserted into the database is subsequently read from the database and then passed back to it again. Quotation marks that have been doubled up initially will return to their original form when the data is reused, allowing the defense to be bypassed.
- Another often cited defense is to use stored procedures for database access. While stored procedures can provide security benefits, they are not guaranteed to prevent SQL injection attacks. The same kinds of vulnerabilities that arise within standard dynamic SQL queries can arise if any SQL is dynamically constructed within stored procedures. Further, even if the procedure is sound, SQL injection can arise if the procedure is invoked in an unsafe manner using user-controllable data.

References:

https://cwe.mitre.org/data/definitions/89.html
https://cwe.mitre.org/data/definitions/94.html
https://cwe.mitre.org/data/definitions/116.html

5.2 Brute Force Authentication

Current Rating
HIGH

The penetration tester performed a brute force password spray attack, which resulted in successful account compromise. While it was observed that brute force login countermeasures are in place

in the form of temporary account lockouts, the penetration tester was able to exploit a weak password policy using a small password set as a workaround.

The below screenshot (Figure 7) shows a Burp Suite Professional Intruder configuration, using a large wordlist of usernames in both the UserName and Password parameters.

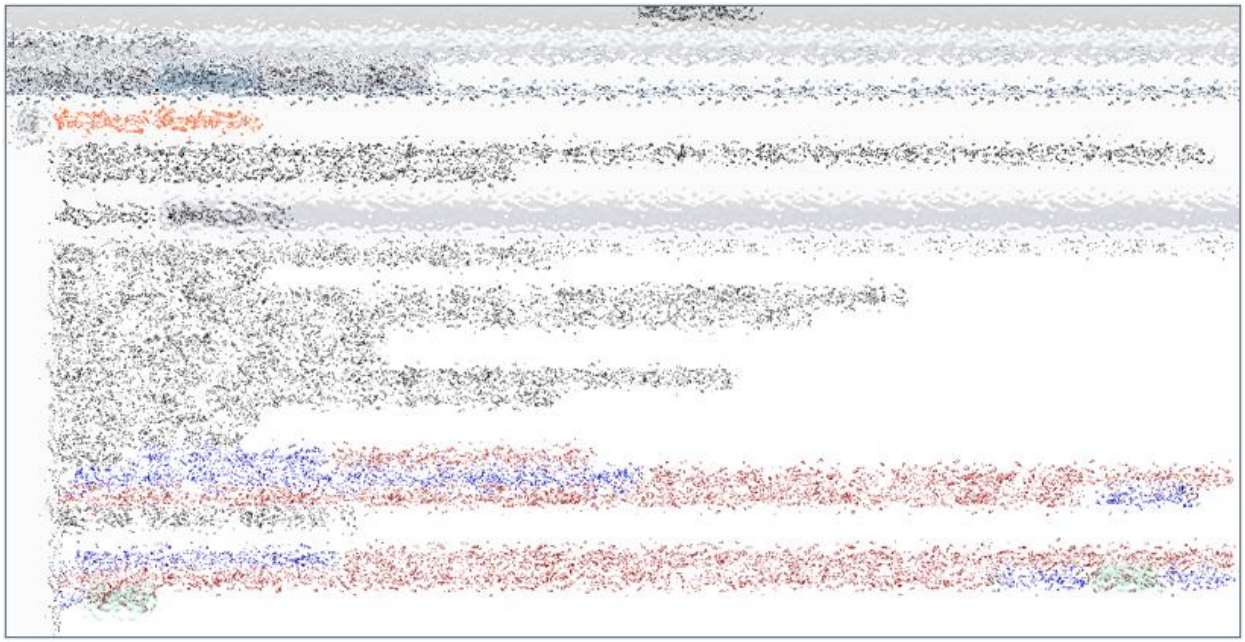

Figure 7 – Burp Suite Professional Intruder configuration

After several thousand requests, it was discovered that one of the users "jane" was successfully authenticated. With more time dedicated towards this attack vector, it is likely that many more accounts would have been compromised. Below are screenshots which depict proof of concept (Figures 8-11).

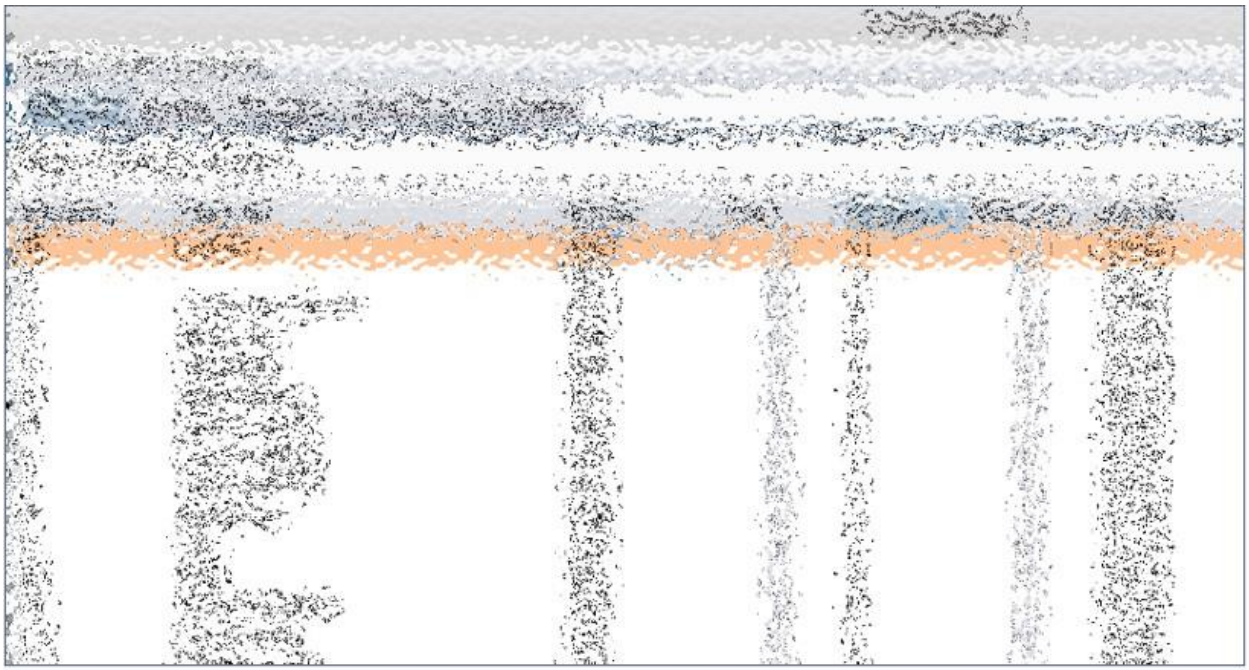

Figure 8 – Successful authentication for jane

The screenshots below show successful authentication into the jane user account.

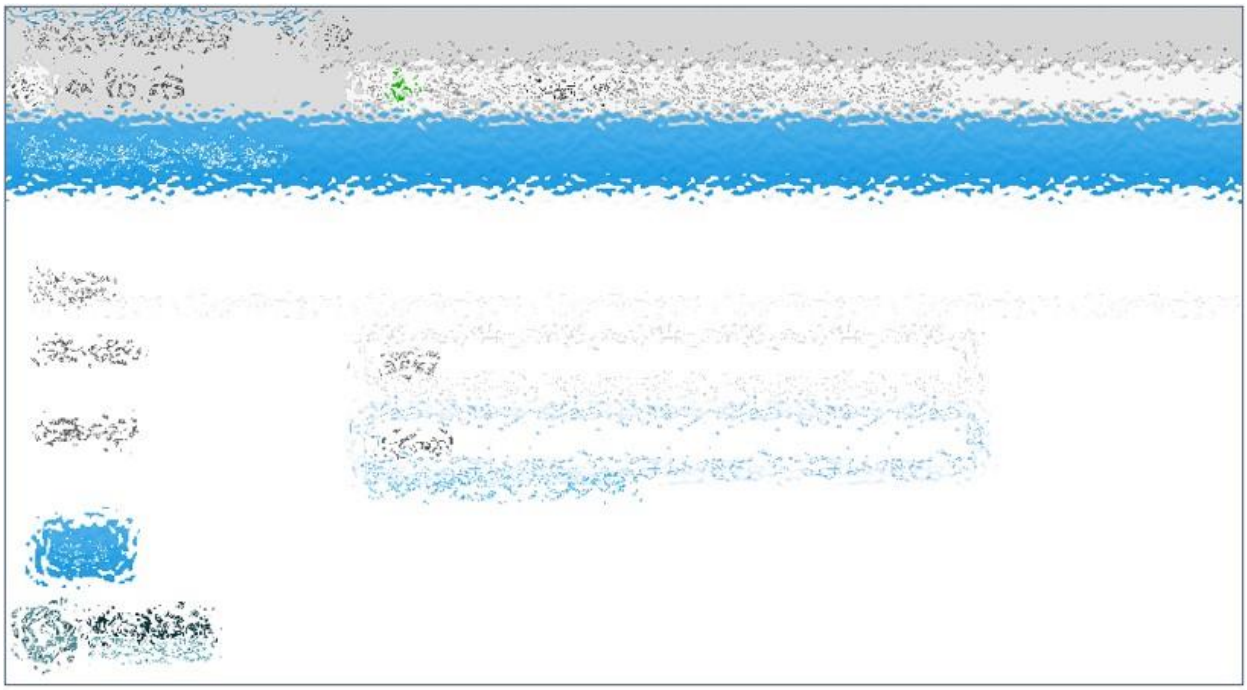

Figure 9 – Login attempt for jane

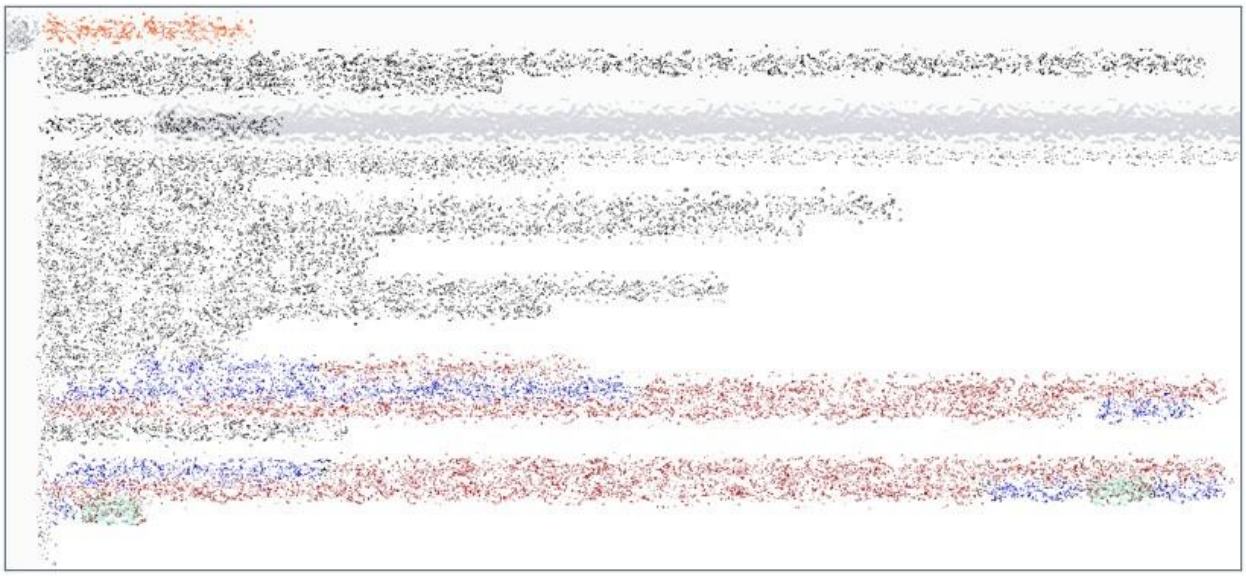

Figure 10 – Login attempt for jane

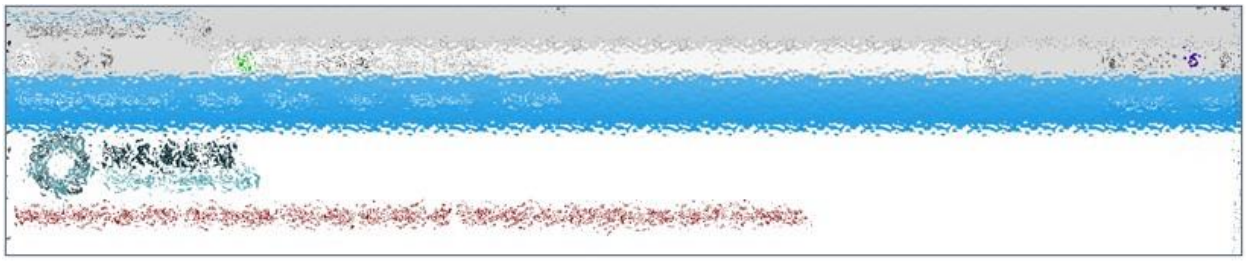

Figure 11 – Successful login for jane

Recommendation:

The current password policy states that it requires a minimum of 8 characters. However, it was observed that the password that was compromised is 6 characters. It is recommended that a password policy is implemented and technically enforced to require passwords of 8 characters minimum with a maximum length of 64 characters or higher. For higher password strength, a requirement could also be added to require a combination of 1 capital letter, one number, and one special character. Additionally, it is recommended that jane's password is changed immediately to a strong password and all other users are forced to reset password to a strong password.

Reference:

https://pages.nist.gov/800-63-3/sp800-63b.html

5.3 Username Enumeration

Current Rating
LOW

The web application is vulnerable to username enumeration. This is made possible by observing the difference responses from the web application when providing valid and invalid usernames within the reset password functionality. The below screenshots (Figures 12-13) display an invalid username being entered (test123), prompting a message which states "User does not exist."

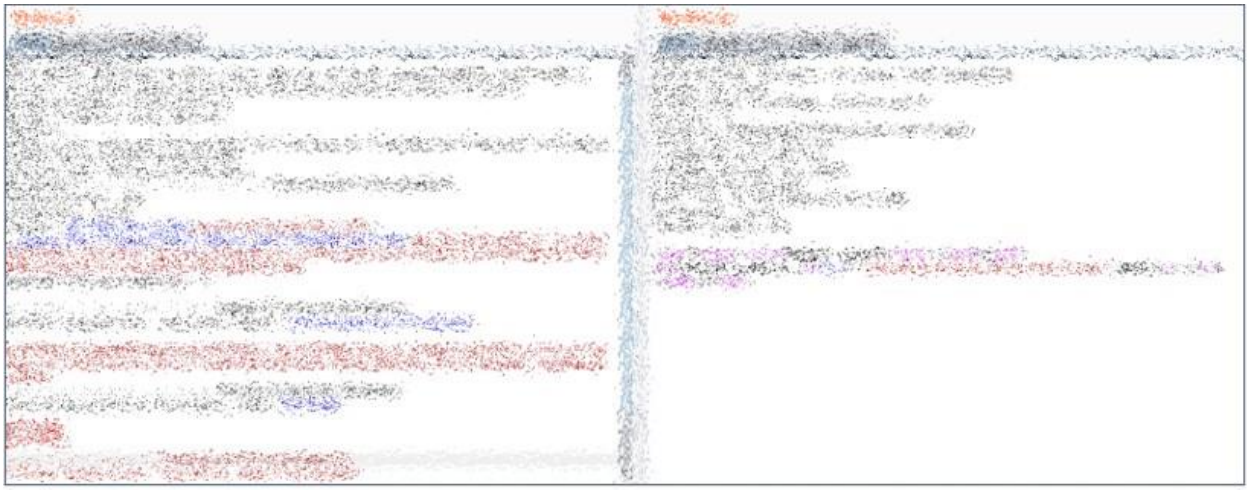

Figure 12 – Invalid username attempt

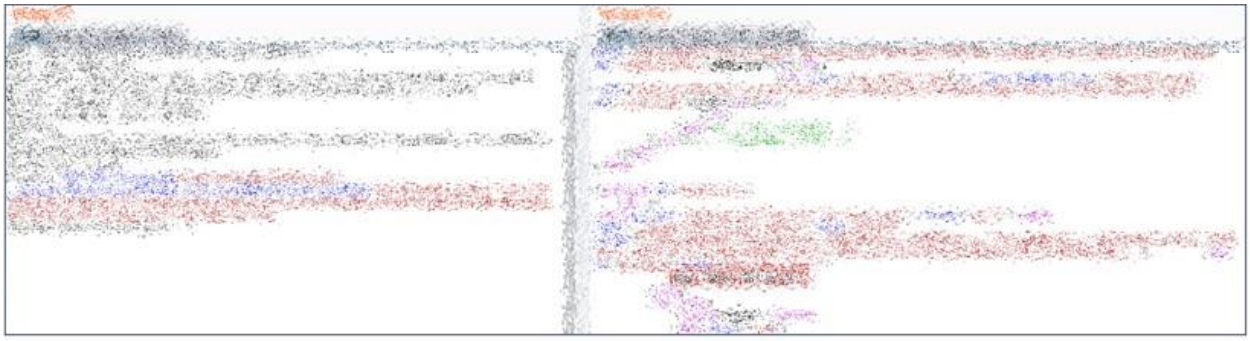

Figure 13 – Invalid username attempt

Subsequently, a known valid username was entered (John.Doe) (see Figures 14-15), and the web application stated, “Mail Sent with password reset token.”

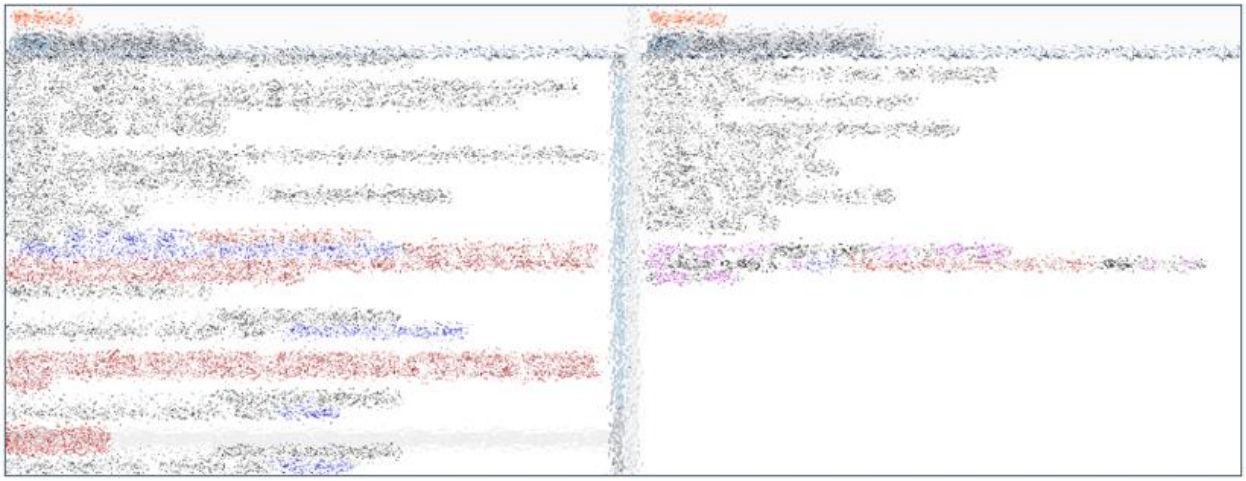

Figure 14 – Valid username attempt

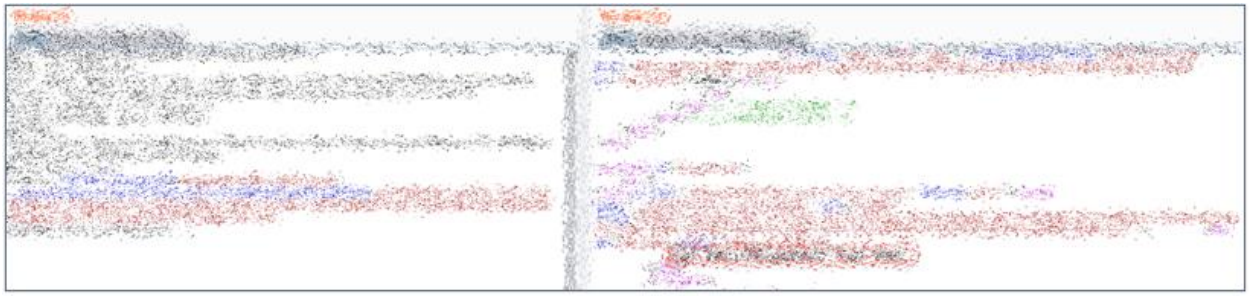

Figure 15 – Valid username attempt

Recommendation:

The response from the web application should not state whether or not the email was correct. Instead, there should be a generic message. For example, "If a valid e-mail was entered, you will receive an e-mail shortly."

Reference:

https://blog.rapid7.com/2017/06/15/about-user-enumeration/#:~:text=User%20enumeration%20is%20when%20a,system%20that%20requires%20user%20authentication

6. Primary Findings for Internal Testing

MainNerve discovered 81 systems on the internal network with open ports, of which, a representative sample was chosen for testing. System selection was based on potential criticality of vulnerabilities present and the role of the system in the network infrastructure.

6.1 Dell OpenManage Server Administrator Authentication Bypass

Current Rating
HIGH

Internal systems 172.16.309.100, 172.16.309.102, and 172.16.309.73 are running out-of-date versions of Dell OpenManage Server Administrator. These versions are prone to several publicly known Common Vulnerabilities and Exposures (CVE). During the course of the penetration test, the tester was able to successfully exploit the systems via authentication bypass (CVE-2021-21513).

Below is a screenshot which shows an out-of-date version of Dell OpenManage Server Administrator (Figure 16).

Figure 16 – Discovery of Dell EMC OpenManage Version

By sending a crafted API call to the web application, it was possible to log in as the administrator as depicted in the below screenshots (Figures 17-18).

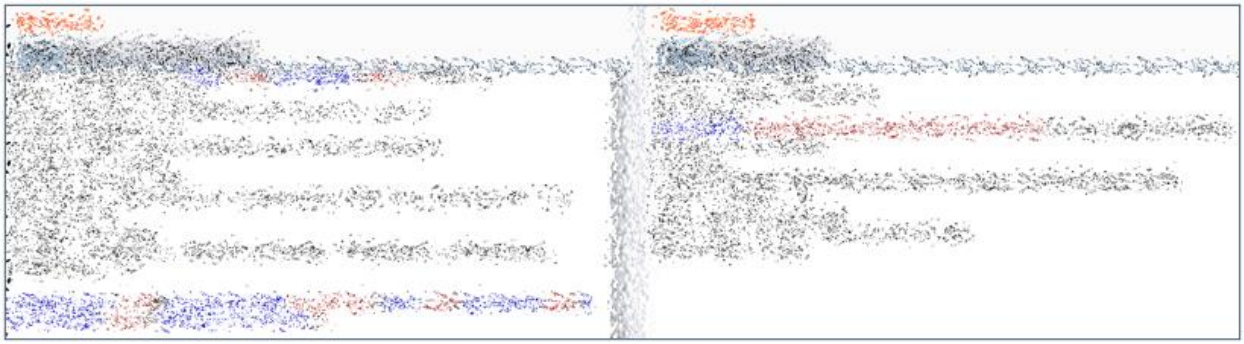

Figure 17 - CVE-2021-21513 exploit

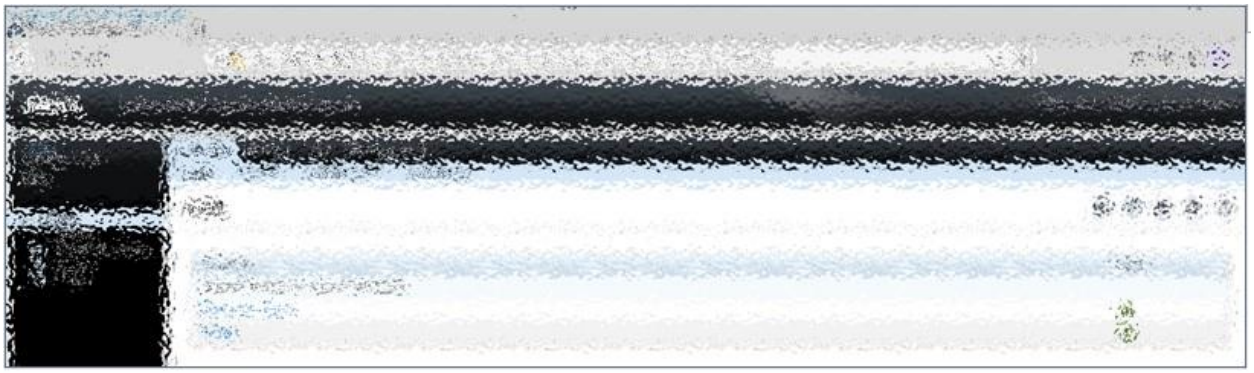

Figure 18 - CVE-2021-21513 exploit

Recommendation:

It is recommended that the version is upgraded to Dell EMC OpenManage Server Administrator 9.4.0.3 / 9.5.0.1 or later.

Reference:

https://www.dell.com/support/kbdoc/en-us/000183670/dsa-2021-040-dell-emc-openmanage-server-administrator-omsa-security-update-for-multiple-vulnerabilities

6.2 Multiple Default Credentials

Current Rating
HIGH

Several internal systems have web applications with default credentials implemented which are publicly known. This is a serious issue because any user within the internal network could authenticate regardless of authorization. Moreover, the accounts provide the ability to perform

administrative configuration actions. Below are the URLs which accept publicly known default credentials.

- https://172.16.309.193 (Integrated Dell Remote Access Controller 8)
- http://172.16.309.10 (Konica Minolta)
- http://172.16.309.65 (Konica Minolta)
- http://172.16.309.73 (Konica Minolta)

The following screenshots show successful authentication using the default credentials for the above web applications (Figures 19-21).

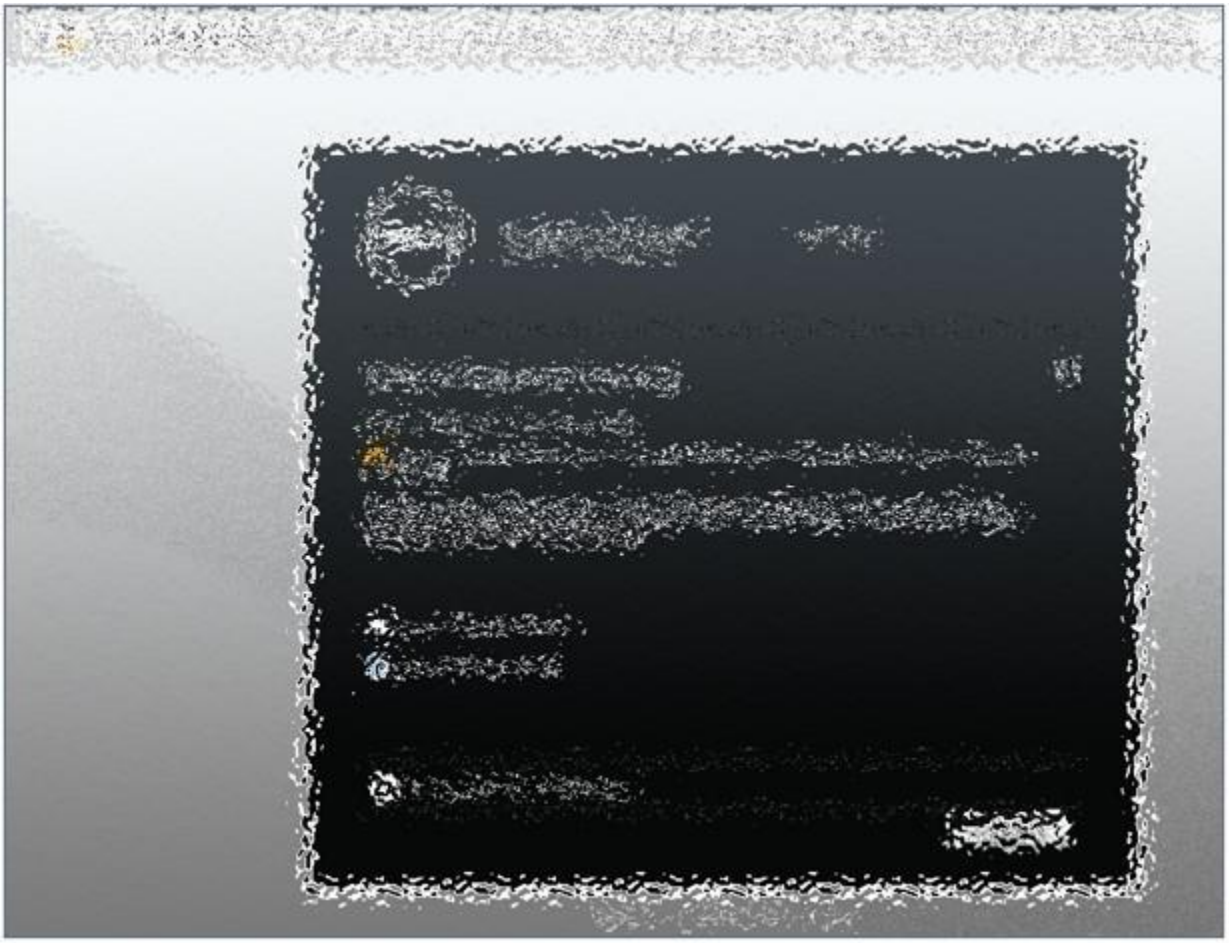

Figure 19 – Default password for iDRAC

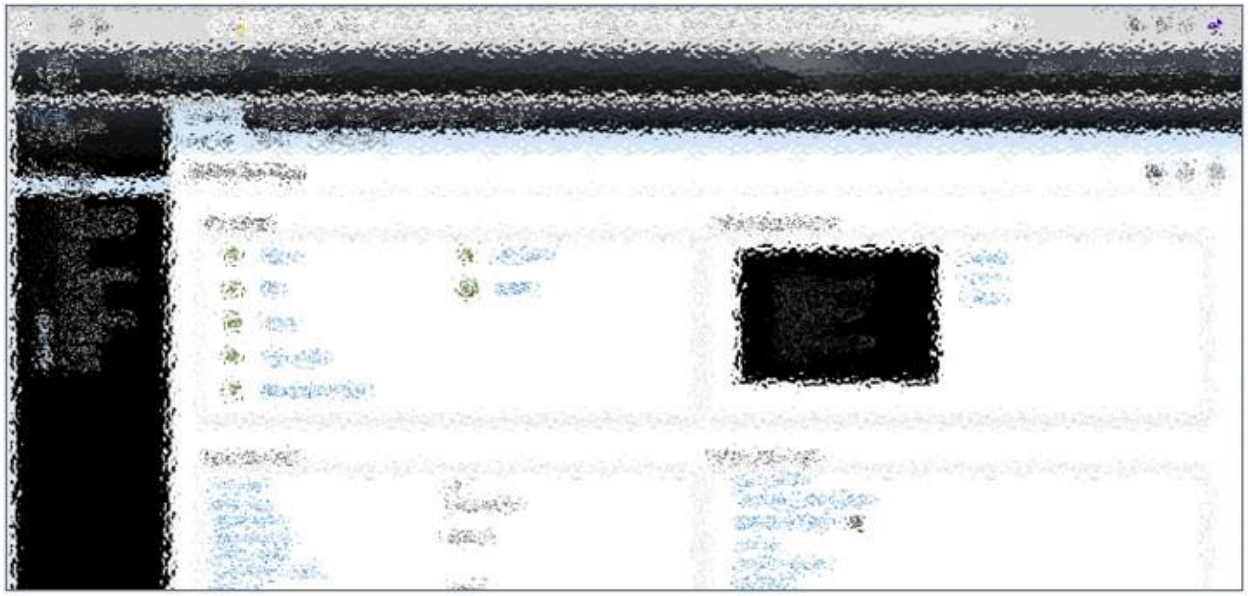

Figure 20 – Successful login with default credentials

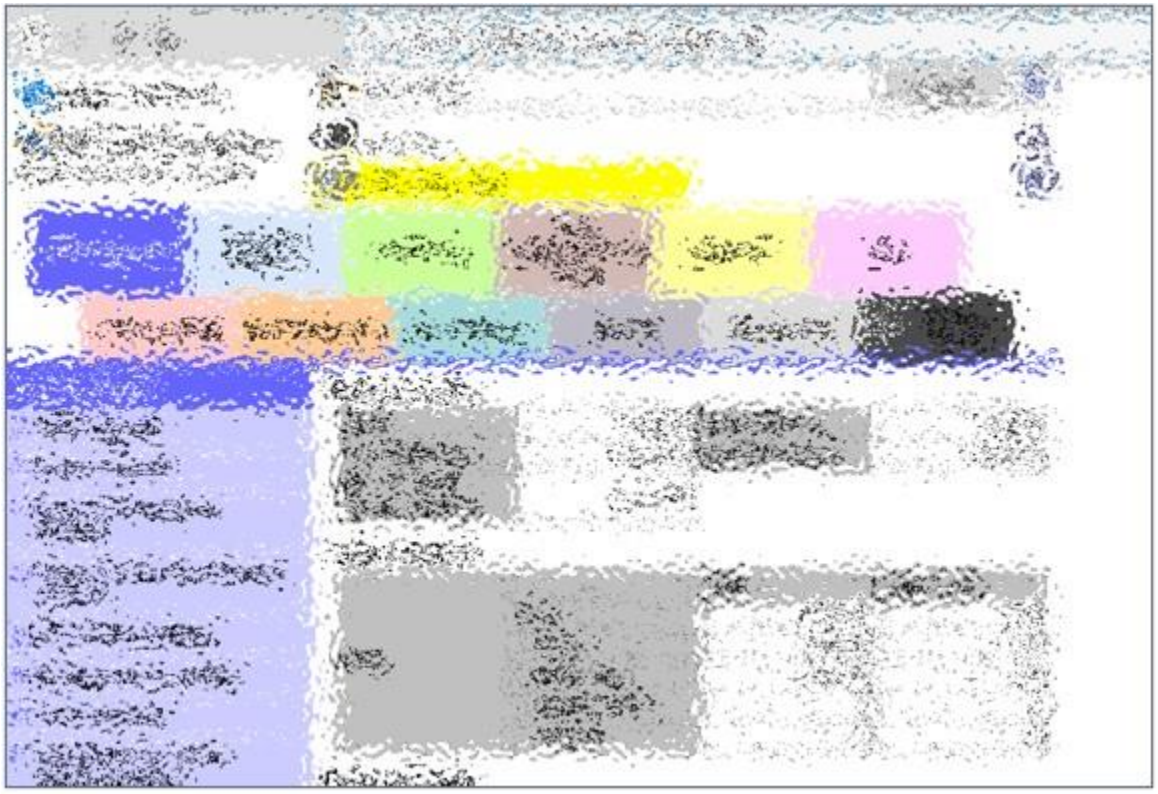

Figure 21 – Default credentials for Konica Minolta

Recommendation:

It is recommended that the default passwords are changed to a strong password. This should be consistent with NIST's standards of password strength and complexity (e.g. minimum of 8

characters, no dictionary words, and a combination of capital/lowercase alphanumeric and special characters).

References:

https://pages.nist.gov/800-63-3/sp800-63b.html#memsecret
https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-Authentication_Testing/02-Testing_for_Default_Credentials

6.3 IPMI Hash Disclosure

Current Rating
HIGH

One internal host (172.16.309.40) is susceptible to Intelligent Platform Management Interface (IPMI) information disclosure due to the support of RMCP+ Authenticated Key-Exchange Protocol (RAKP) authentication. This makes it possible for anyone within the internal network to acquire password hashes for legitimate users via the HMAC from a RAKP message 2 response from a BMC. The penetration tester leveraged the available Metasploit auxiliary module for proof of concept as depicted below (Figure 22).

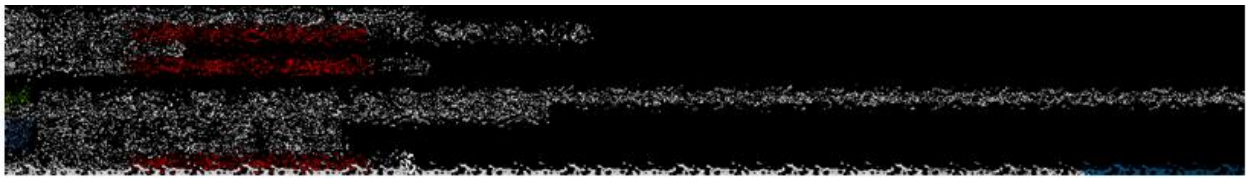

Figure 22 – IPMI hash discovery

Recommendation:

There is currently no security patch or workaround for this vulnerability due to it being an inherent problem with the specification for IPMI 2.0. Suggested mitigation approaches include the following.

- Disabling IPMI over LAN if it is not needed.
- Using strong passwords to limit the successfulness of off-line dictionary attacks.
- Using Access Control Lists (ACLs) or isolated networks to limit access to the IPMI management interfaces.

References:

https://www.tenable.com/plugins/nessus/80101

https://www.rapid7.com/blog/post/2013/07/02/a-penetration-testers-guide-to-ipmi
https://pages.nist.gov/800-63-3/sp800-63b.html#memsecret
https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-Authentication_Testing/02-Testing_for_Default_Credentials

6.4 End-of-life Windows Server 2008 r2

Current Rating
HIGH

Through manual analysis of several internal systems, it was discovered that a Windows operating system is being used which is considered end-of-life. This means that important security updates will not be provided, leaving these systems at risk of current and future vulnerabilities associated with the particular operating systems. Below are the affected systems.

- 172.16.309.18
- 172.16.309.19

The following screenshot (Figure 23) shows enumeration of one of the end-of-life operating systems.

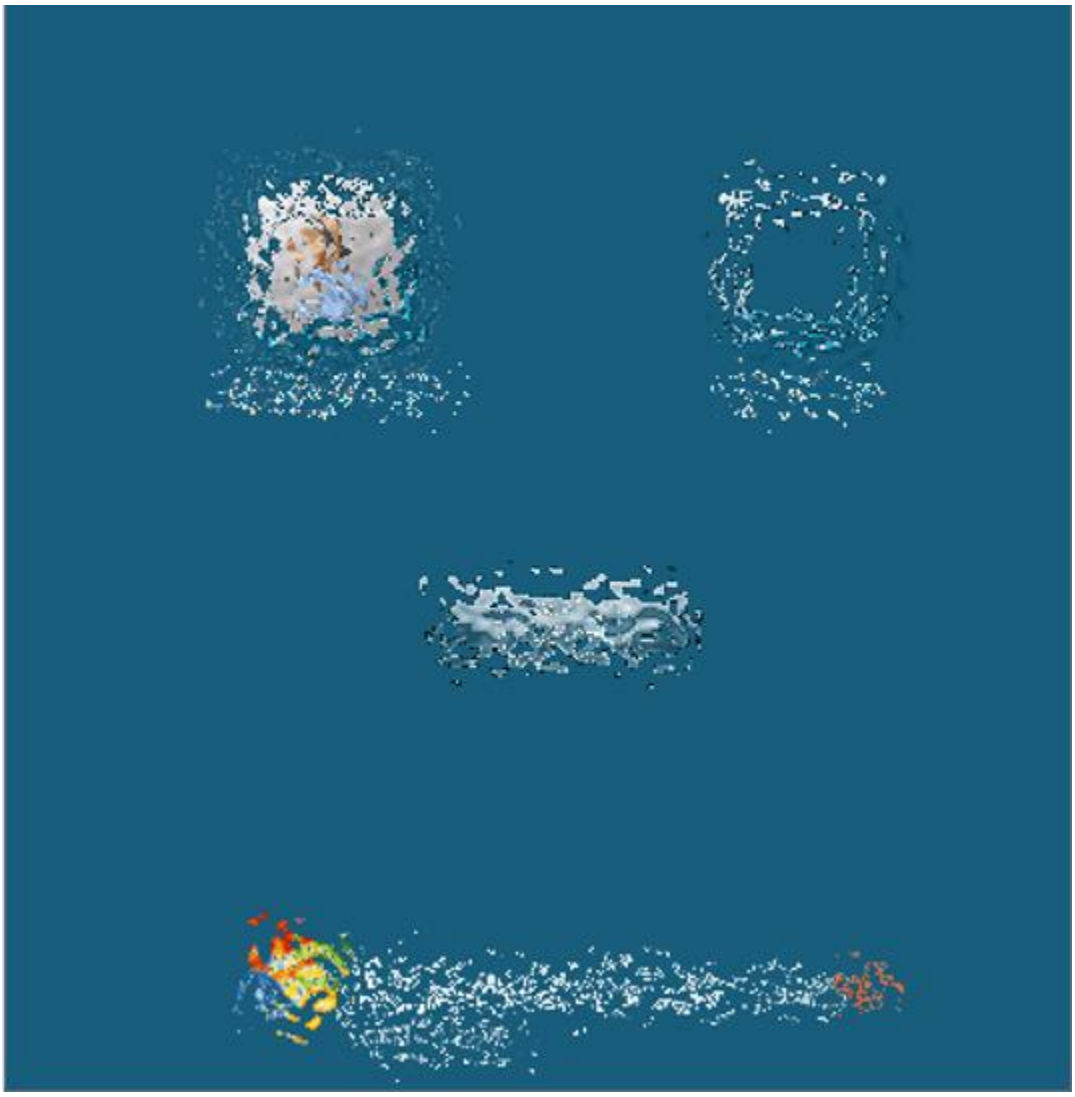

Figure 23 – Operating system discovery

Recommendation:

The Microsoft Windows system should be upgraded to a current supported version.

Reference:

https://learn.microsoft.com/en-us/troubleshoot/windows-server/windows-server-eos-faq/end-of-support-windows-server-2008-2008r2

6.5 Out-of-date DNS Server (6.1.7601)

Current Rating
HIGH

The DNS server version of one of the internal systems (172.16.309.19) is associated with CVE-2020-1350. This is a publicly known vulnerability which can result in remote code execution if locally exploited. The penetration tester was not able to exploit this vulnerability due to lack of local system access to this particular system. Below is a screenshot (Figure 24) which shows discovery of the aforementioned DNS server.

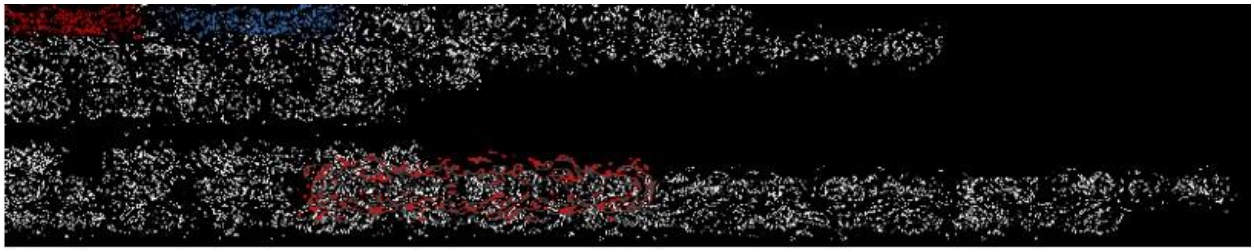

Figure 24 – DNS server enumeration

Recommendation:

The Microsoft DNS server should be upgraded to the most recent version. Alternatively, Microsoft has released a set of patches for Windows Server which could be implemented.

Reference:

https://support.microsoft.com/en-us/topic/kb4569509-guidance-for-dns-server-vulnerability-cve-2020-1350-6bdf3ae7-1961-2d25-7244-cce61b056569

6.6 Out-of-date iDRAC Versions

Current Rating
HIGH

It was discovered that several internal systems have out-of-date versions of Integrated Dell Remote Access Controller (iDRAC) implemented. These versions of iDRAC are associated with numerous Common Vulnerabilities and Exposures (CVE). The penetration tester attempted to exploit these vulnerabilities, one of which appeared to be successful. However, there was no reverse shell acquired. It is suspected that this is due to an issue with the network route. That being said, this finding is based solely upon the discovered versions. Below is a screenshot which shows discovery of the iDRAC version and exploitation attempt.

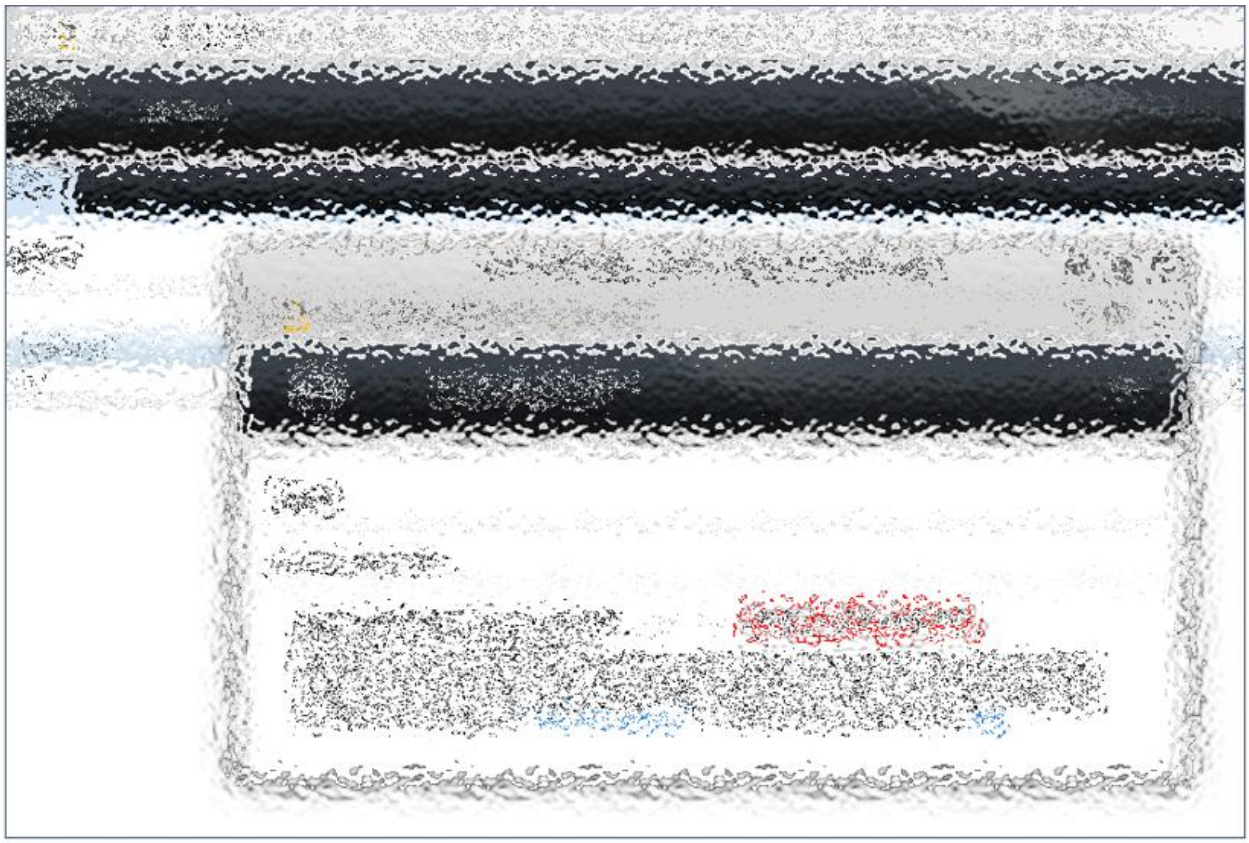

Figure 25 – iDRAC version discovery

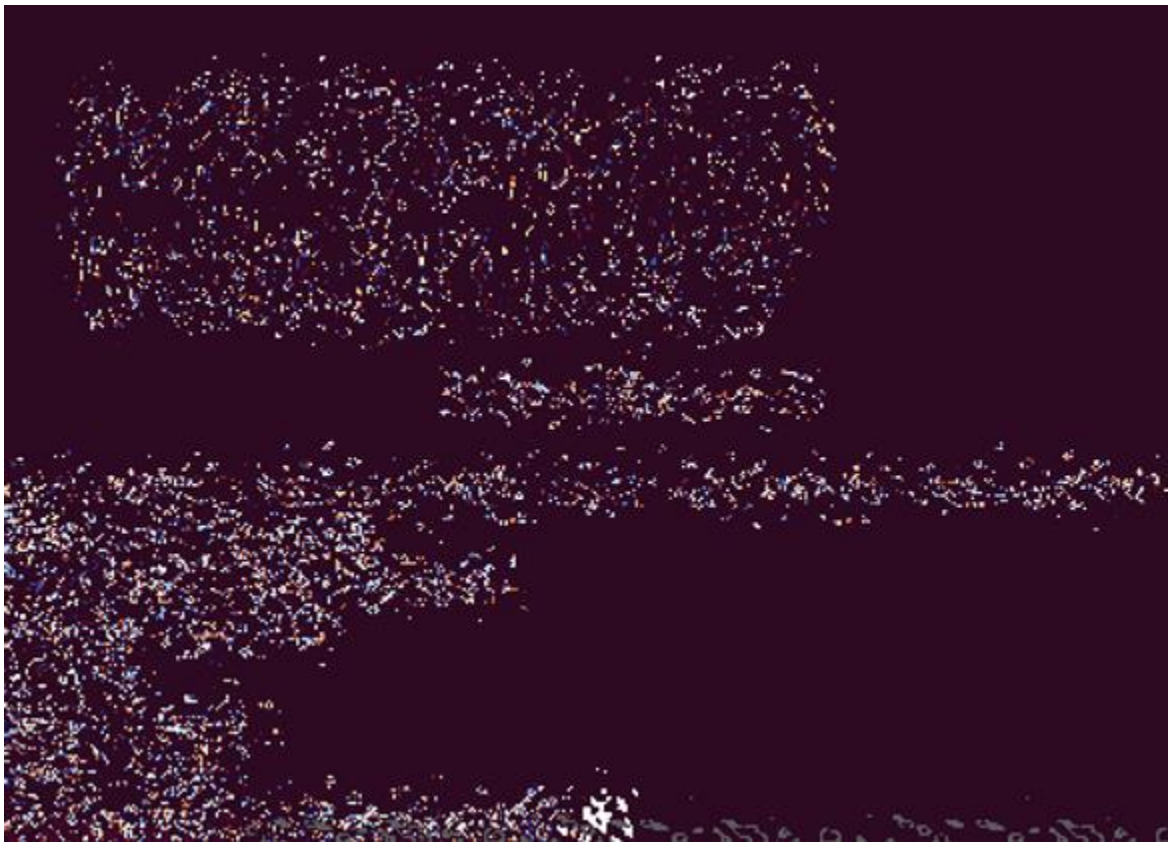

Figure 26 – Exploitation attempt

Recommendation:

The firmware for this iDRAC implementation should be upgraded to the most recent version.

Reference:

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-5685

6.7 End-of-life MSSQL Server Version 2014 12.00.5223

Current Rating
HIGH

It was discovered that several of the internal systems are using end-of-life versions of Microsoft Structured Query Language (MSSQL) Server. This means that import security updates will not be provided, leaving these systems at risk of current and future vulnerabilities associated with the particular MSSQL versions. Below are the affected systems.

- 172.16.309.102
- 172.16.309.18
- 172.16.309.19

- 172.16.309.6
- 172.16.309.70

The below screenshot (Figure 27) shows discovery of the MSSQL version for one of the above systems.

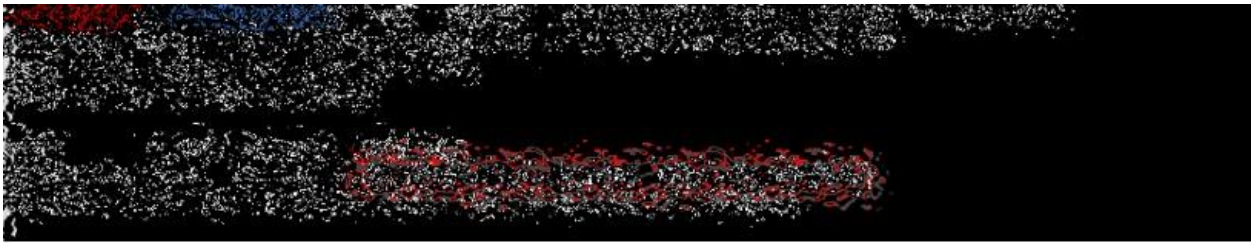

Figure 27 – MSSQL version discovery

Recommendation:

It is recommended that MSSQL is upgraded to the most recent version, or at a minimum one that is currently supported by the vendor.

Reference:

https://learn.microsoft.com/en-us/lifecycle/products/sql-server-2014

6.8 End-of-life Oracle Server Version 12.2.0.1.0

Current Rating
HIGH

It was discovered that one of the internal systems is using end-of-life versions of Oracle Server. This means that import security updates will not be provided, leaving the system at risk of current and future vulnerabilities associated with the particular Oracle version. Below is the affected system.

- 172.16.309.84

Below is a screenshot depicting the discovery (Figure 28).

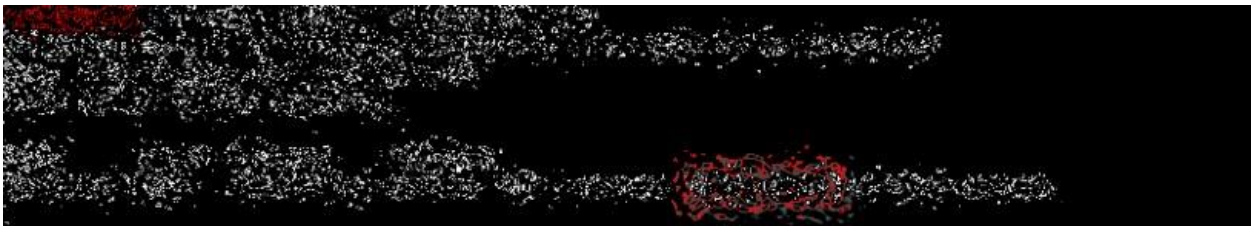

Figure 28 – Oracle version discovery

Recommendation:

It is recommended that Oracle Server is upgraded to the most recent version, or at a minimum one that is currently supported by the vendor.

Reference:

https://www.oracle.com/us/assets/lifetime-support-technology-069183.pdf

6.9 No Password Set for Admin

Current Rating
HIGH

Through manual discovery and analysis of several internal systems, it was discovered that multiple systems have no password set for HP LaserJet over ports 80, 443, and 23. It was also discovered that these systems provide administrative permissions. This means that anyone on the network can log in as administrator and perform actions in the context of admin, regardless of authorization. Below are screenshots depicting discovery (Figures 29-30).

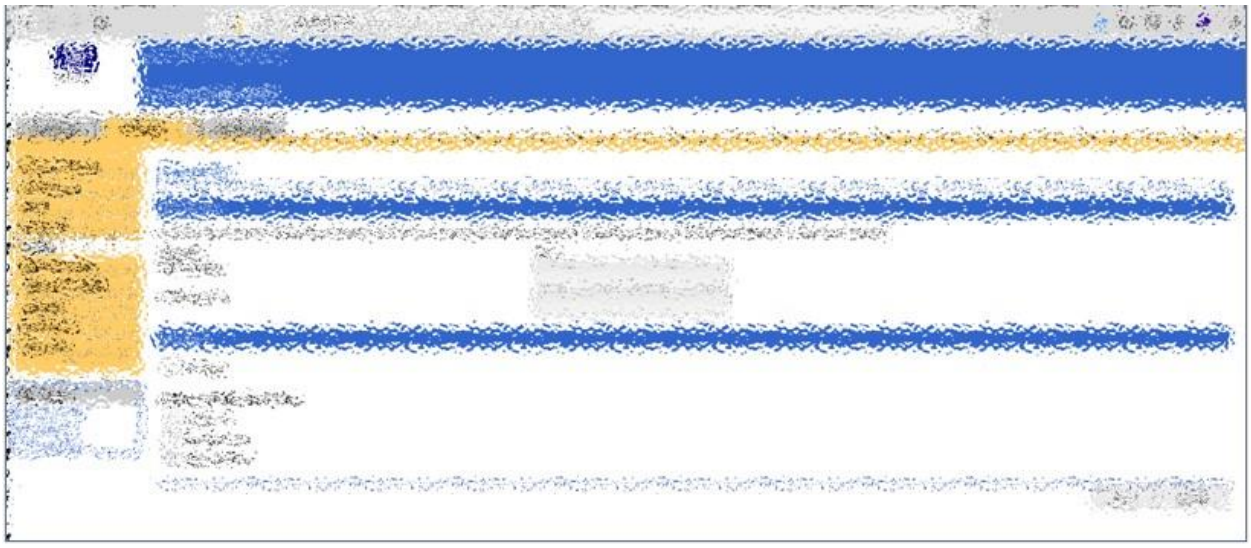

Figure 29 – No password set

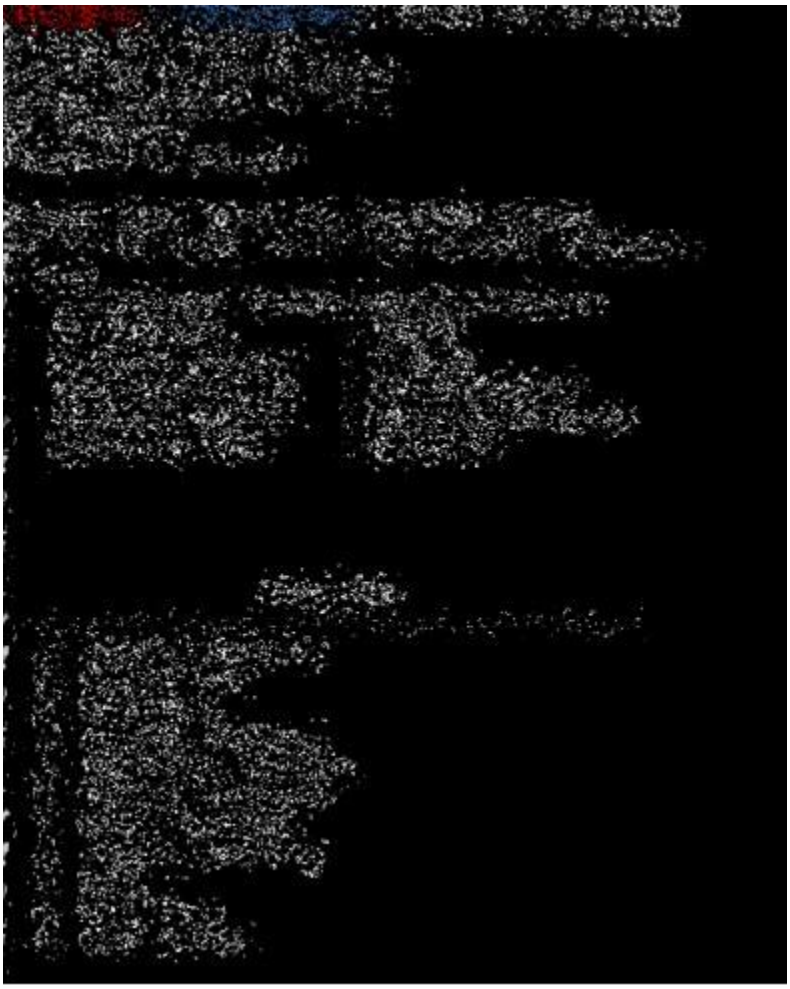

Figure 30 – No password set

Recommendation:

It is recommended that the default passwords are changed to a strong password. This should be consistent with NIST's standards of password strength and complexity (e.g. minimum of 8 characters, no dictionary words, and a combination of capital/lowercase alphanumeric and special characters).

References:

https://pages.nist.gov/800-63-3/sp800-63b.html#memsecret
https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-Authentication_Testing/02-Testing_for_Default_Credentials

6.10 Out-of-date SSLv3

Current Rating
MEDIUM

Several systems are utilizing outdated and vulnerable versions of SSL (version 3.0). SSL version 3.0 is considered an unsatisfactory means of cryptographically securing communications by contemporary cyber security standards. The vulnerability inherent within SSL version 3.0 may allow the POODLE attack to take place if an attacker is positioned suitably to act as a Man-in-the-Middle (MITM), and has the ability to control portions of the client side of the SSL connection. The POODLE attack takes advantage of the protocol version negotiation feature built into SSL/TLS to force the use of SSL 3.0 and then leverages this new vulnerability to decrypt select content within the SSL session.

- 172.16.309.6
- 172.16.309.16

The below screenshot (Figure 31) shows enumeration of SSL, TLS, and underlying ciphers.

Figure 31 – SSL version 3.0 discovery

Recommendation:

Consult the application's documentation to disable SSL versions 2.0 and 3.0. Also disable TLS versions 1.0 and 1.1. Use TLS 1.2 (with approved cipher suites) or higher instead.

Reference:

https://www.openssl.org/~bodo/ssl-poodle.pdf

6.11 SNMP Public Default

Current Rating
LOW

SNMP is configured on internal system 10.23.304.3 over UDP port 161 with the default community name of "public", which can allow anyone within the internal network to enumerate sensitive information from each affected system. Below is a screenshot showing enumeration using SNMP.

Figure 32 – SNMP enumeration

Recommendation:

Unless required for operations, disable SNMP. Otherwise, change the default community string to something which cannot be easily guessed.

References:

https://www.rapid7.com/db/vulnerabilities/SNMP-READ-0001/
https://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0186

7. Testing Tools

Throughout this entire assessment, MainNerve used the following tools to scan, enumerate, and attempt to exploit vulnerabilities and/or misconfigurations on the systems.

Tool Name	Description/Purpose
Nessus Professional	Vulnerability identification and verification
Nmap	Host/Service discovery and enumeration
Metasploit	Exploitation and scanning
Burp Suite Professional	Web-based vulnerability identification and verification
Responder	Attempted credential harvesting
OpenVAS	Vulnerability identification and verification

8. Risk Rating Overview

MainNerve uses the DREAD threat modeling algorithm to determine the amount of risk posed to a system or infrastructure by vulnerabilities discovered during an assessment.

Damage Potential	Reproducability	Exploitability	Affected Users	Discoverability
•If a threat exploit occurs, how much damage will be caused?	•How difficult is it to reproduce the attack?	•What is needed to exploit this threat?	•How many users will be affected?	•How easy is it to discover this threat?

Rating (Value)	High (10)	Medium (5)	Low (0)
Damage Potential	An attacker can gain privileged access; full system/network breach is possible	Some sensitive data may be accessed by an attacker, but full system breach is improbable	No sensitive data or information leakage
Reproducibility	An attacker can easily circumvent security controls	An attacker can bypass security controls under certain conditions	Very hard or impossible, even for administrators of the system
Exploitability	Requires little and/or novice-level skills	Requires moderate skills and malware is publicly available	Requires very high skill level with custom or advanced attack tools
Affected Users	All users. Critical processes significantly affected	Some users are affected. Critical processes operational	Very little to no impact on users or critical processes
Discoverability	No detection controls or measures in place	Details are already in the public domain and are easily discovered using a search engine	Security controls in place that detect and block attacks; vulnerability not overt

8.1 Risk Calculation

MainNerve assigns a Risk Value to each relevant finding discovered during this assessment using the following calculation.

To determine the priority level, MainNerve uses the following table of values. The *overall* Risk Rating is calculated by the highest priority rating. That is, if an organization has four (4) Medium priorities and one (1) High, the overall Threat Risk Impact Rating would be High.

Risk Value	Priority
0 - 4	Low
5 – 7	Medium
8 and greater	High

Table 2 – Risk impact table of values

8.2 Risk Impact Rating

Based on the findings and analysis by MainNerve, the risks are calculated in the following table. It should be noted that the DREAD algorithm, as calculated below, is used to compute a risk value, which is an average of all five categories.

Finding	D	R	E	A	D	Value	Priority
Time-based Blind SQL Injection	10	10	10	10	10	50/5=10	High
Brute Force Authentication	10	10	10	10	10	50/5=10	High
Username Enumeration	2	2	2	2	2	10/5=2	Low
Dell OpenManage Server Administrator Authentication Bypass	10	10	10	10	10	50/5=10	High
Multiple Default Credentials	10	10	10	10	10	50/5=10	High
IPMI Hash Disclosure	8	8	8	8	8	40/5=8	High
End-of-life Windows Server 2008 r2	8	8	8	8	8	40/5=8	High
Out-of-date DNS Server (6.1.7601)	8	8	8	8	8	40/5=8	High
Out-of-date iDRAC Versions	8	8	8	8	8	40/5=8	High
End-of-life MSSQL Server Version 2014	8	8	8	8	8	40/5=8	High
End-of-life Oracle Server Version 12.2.0.1.0	8	8	8	8	8	40/5=8	High
No Password Set for Admin	8	8	8	8	8	40/5=8	High
Out-of-date SSLv3	5	5	5	5	5	25/5=5	Medium
SNMP Public Default	1	1	1	1	1	5/5=1	Low

Based on these findings, MainNerve has determined that the overall risk impact is High. As a matter of due diligence, the customer should assume that the findings of this report are a point-in-time assessment and review the findings in this report with the recommendations to reduce its risk exposure through effective remediation. Periodic assessments should also be completed as a matter of best practice while ensuring a higher level of security.