**BK Security Cybersecurity Services**

121 Look Ave.

Salt Lake City, UT 23434

USA

(343) 384 – 3474

[Info@bksec.com](mailto:Info@bksec.com)

Bksecurity.com



Table of Contents

[Executive Summary 2](#_Toc120445215)

[Introduction 2](#_Toc120445216)

[Attack Narrative/Scope 2](#_Toc120445217)

[Methodology 3](#_Toc120445218)

[Findings and Remediation 9](#_Toc120445219)

# Executive Summary

BK Security was awarded a contract to perform penetration testing services against internet facing information technology (IT) infrastructure to determine its vulnerabilities to cybersecurity risks and outside threats and provide advice on steps to take to mitigate them. This report contains outcomes pertaining to enumerations of target assets, hosts, and services, identification of sensitive data, and recommendations on remediation methods. This report also includes the detailed steps taken to arrive at the findings.

# Introduction

BK Security is a comprehensive cybersecurity services company that provides commitment to clients to leverage well known and modern methodologies and tools to identify cybersecurity vulnerabilities. Additionally, our mitigation recommendations are based on feasibility of our client. We believe that cybersecurity is a journey and every organization can take a variety of steps to working towards a more security IT architecture.

Our services include:

* Vulnerability Analysis
* Penetration Testing
* Restructuring Infrastructure
* Threat Intelligence Analysis
* Network Security Analysis
* Security Modernization

Our objective with a penetration test for your organization is to identify vulnerabilities no matter the gravity as we believe that the smallest overlooked detail could have devastating effects.

Our approach is to start from the basics of enumerating internet facing assets so no details are overlooked, and we are able to draw the full picture of the vulnerable infrastructure as a potential attacker might. We put ourselves in the attacker’s shoes to ensure that we cover every corner and make the organization aware of a vulnerability before an attacker finds it.

# Attack Narrative/Scope

Our target infrastructure for penetration testing for NBN included is a cloud server image that is used for online account management for customers and employee services, and an employee workstation that is used to access the server or perform customer service tasks.

The scope of our work includes penetration testing of the following areas:

1. External facing hosts
2. External facing services
3. External web applications
4. Internal facing hosts
5. Internal facing services

# Methodology

We used a Kali Linux computer to conduct our penetration testing as it comes equipped with efficient tools that are also easily updatable so that we’re leveraging the most modern methods.

The types of testing we conducted are:

* Network
* Web application

The steps taken from Penetration Testing Framework 0.59 were in the following categories:

* Discovery and Probing
* Enumeration
* Password cracking
* Vulnerability Assessment
* Server Specific Testing
* Final Report

The tools we used for each of the above framework categories are as follows:

* Discovery and Probing
  + Nmap
  + Netcat
  + Active Hosts
  + OS Fingerprint
* Enumeration
  + FTP
  + Password guessing
  + Hydra
  + SSH
* Password cracking
  + Metasploit
  + Hydra
* Vulnerability Assessment
  + ZAP
  + Nikto
  + Resources used: CVE, Exploit-DB, globalscape
* Server Specific Testing
  + Web Vulnerability Assessment

Since the server being tested hosted a web application, we used the Open Web Application Security Project (OWASP) testing guide to conduct passive and active testing in the following categories:

* Identity Management
* Authentication
* Authorization
* Input Validation
* Error Handling

We started by running a network scan using Network Mapper (Nmap) as shown in Figure 1. We ran nmap against all three known IP addresses of NBN. The client address of course did not return an open ports; all were in ignored states. The server address returned 4 common ports that were open and showed what service each port was running.

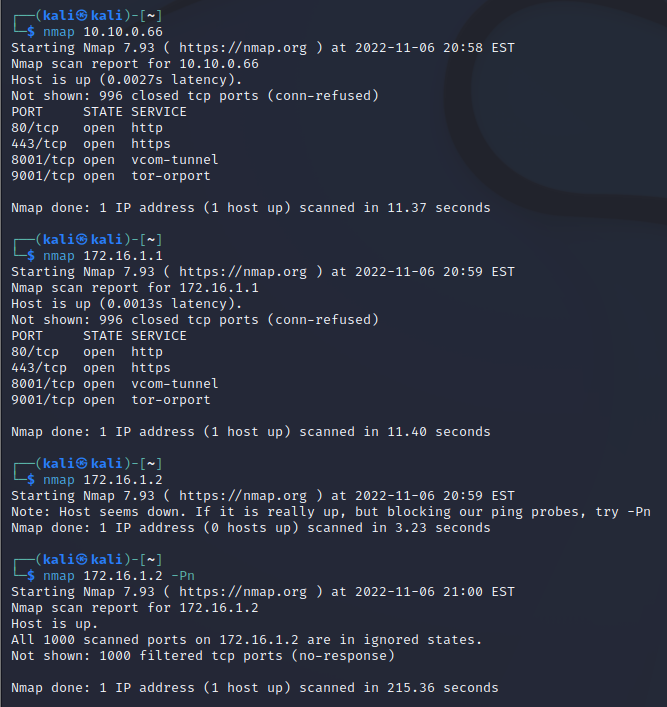


Figure 1

We used netcat (ncat) which is a network reading tool as shown in Figure 2. We ran ncat against the eth0 IP address of the NBN server. From this step we learned that port 443 is actually running the SSH service which isn’t the norm for SSH. We also got the version number which helped us try to find any known vulnerabilities associated with the specific version.

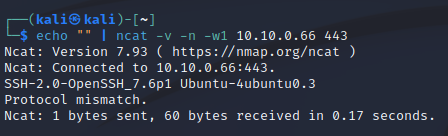


Figure 2

We used Wireshark, a packet analyzer, as shown in Figure 3 to see what the communication looks like between our client and NBN’s server to understand what we go through and what the endpoint may look like. We didn’t gather too much useful information from this step but it is always good practice to analyze the communication pathway to a potentially vulnerable host.

Calendar

Description automatically generated with medium confidence

Figure 3

To get a bit more information about the open ports on the NBN server, we filtered our nmap command as shown in Figure 4. This step was a bit more enlightening as it showed us that there is a text file on the http service, and there are other entries. So we decided to enter 10.10.0.66/internal on a browser. That returned a blank page. We further inspected that page and we found a note as shown in Figure 5. The note didn’t expose any sensitive information however it did show the developer’s notes which in a worse case scenario could potentially expose information that a bad actor could leverage. For example, from the note “research and mitigate possible injection vulns” can suggest that there may be sql injection vulnerabilities. That’s a great hint for a bad actor to take advantage of. From the note “ensure that we’re not leaking anything sensitive in CEO’s metadata” could suggest that somewhere in that address there is some metadata related to the CEO that could be accessed by signing in. That could motivate bad actors to persistently try to penetrate that page.

Text

Description automatically generated

Figure 4

Graphical user interface, text, application

Description automatically generated

Figure 5

We then went to 10.10.0.66/data/ and found a decent amount of content that shouldn’t be there such as a customer list. We also found another flag: **flag1{away\_we\_go}**. There was another flag there but it was password protected along with an image file as shown in Figure 6 and 7.

Graphical user interface, text, application, email

Description automatically generated

Figure 6

Graphical user interface, text, application, email

Description automatically generated

Figure 7

We filtered our nmap command a bit more as shown in Figure 6. In this step we were confirmed that ssh was running on port 443 and we were able to see the ssh hostkeys. Another interesting result of this step was regarding port 9001 where we can see that FTP is running. We were able to see really helpful ftp server information including the FTP server address and the directory. Since anonymous FTP is allowed, we were able to log in and view the file in the “gibson” directory. In the directory we found a flag: **flag3{brilliantly\_lit\_boulevard}** as shown in Figure 7.

Text

Description automatically generated

Figure 6

Text

Description automatically generated

Figure 7

The result in Figure 6 returned some good details that helped us research for known vulnerabilities that may exist. For example potential denial of service (DOS) attack by making repeated connections to VSFTPD. "VSFTPD only lets a certain amount of connections to be made to the server, so, by repeatedly making new connections to the server, you can block other legitimate users from making a connection to the server, if the connections/ip isn't limited. (if it's limited, just run this script from different proxies using proxy chains, and it will work)" ([vsftpd 3.0.3 - Remote Denial of Service - Multiple remote Exploit (exploit-db.com)](https://www.exploit-db.com/exploits/49719))

We also looked into exploits related FTP for example: [Top 4 FTP Exploits Used by Hackers | FTP Risks (globalscape.com)](https://www.globalscape.com/blog/top-4-ftp-exploits-used-hackers):

We looked into the version of apache server being used and found some common vulnerability information related to that version for example: [Apache Http Server version 2.4.29 : Security vulnerabilities (cvedetails.com)](https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-576122/Apache-Http-Server-2.4.29.html):

We used the OWASP Zed Attack Proxy (ZAP) tool to try to find vulnerabilities in the web application being hosted on the NBN server address of 10.10.0.66. ZAP was able to write to the customer list page as shown in figure 8, which should be prevented. However the best approach is to not have the customer list exposed.

Text

Description automatically generated

Figure 8

ZAP also found additional vulnerabilities with the NBN web application such as cross site scripting (DOM based, persistent, and reflect), remote OS command injection, application error disclosure, directory browsing, vulnerable JS library, and absence of Anti-CSRF tokens which confirmed some of our findings from earlier for example application error disclosure. As shown in Figure 9, the login failed message of the web application gave us three pieces of information:

1. There is a sql table in the back
2. One of the tables is called users
3. Another table is called password

Error messages or failure messages should be hidden or written in a way that does not over-explain the error so that bad actors can learn from fake sign in attempts exactly what is going on in the background and what they should do to mimic a successful login.

Graphical user interface, text, application, email

Description automatically generated

Figure 9

We used Nikto to an analysis of the target and found the following vulnerabilities as shown in Figure 10:

Text

Description automatically generated

Figure 10

# Findings and Remediation

At the conclusion of this test, we identified some steps that NBN should take to protect its information from the Internet such as:

1. Not allow anonymous FTP login
2. Prevent your open ports from being visible
3. Not put any information in the robots.txt file
4. Not put customer list in a public space
5. Put in place preventative measures to be able to write your web page
6. Prevent SQL injections from sign in pages
7. Encourage users to use strong passwords

In this report we delivered the parts of the IT infrastructure that we investigated, the methods we used to explore each part of the network that was visible to us, and exploited any vulnerabilities to fully test the gravity of them. We provided some mitigation steps to take to secure the internet facing infrastructure.

The format of this report includes an executive summary in the beginning which lays out the scope this contract. The most important information is included in the Methodology section which iterates through every step we took for this penetration testing project and the findings we came across at each step. Even if we didn’t exploit a certain vulnerability, we noted where extra security measures should be applied.