Cantina Lorenzo

TUSCAN / BASQUE FUSION BAR AND CANTINA

Penetration Test for Cantina Lorenzo

Conducted by

Gruosso Francesco

As a submission for

CM3109 - Ethical Hacking

Date: 8 Dec 2021

Supervisor: Dr Shamal Faily

Abstract

This penetration test report, commissioned by Lorenzo Segrè, identifies and proposes countermeasures to vulnerabilities found in a Virtual Machine Lorenzo has set up to manage his cantina’s inventory. It focuses on three significant weaknesses, including relevant terminology and background or context for each one.

Relevant terminology

To allow a better understanding of how this penetration test was executed, below is a list of relevant keywords and utilised tools.

Keywords

- OS: Operating System
- VM: Virtual Machine
- (Distributed) Denial of Service (DDoS and DoS): Overwhelming a system with data, rendering it challenging to access for actual users.
- Scanning: Detecting what systems are alive and reachable via the internet and discovering what services they offer.
    - Host discovery: Scanning a network to identify which ports are open.
    - Passive OS scanning: Identifying the OS of a machine by reading its network traffic.
- Enumeration: Intrusive probing to identify valuable resources.
    - Banner grabbing: Gain information about a device over a network and what services it runs on its open ports.
    - Stack fingerprinting: Determine what OS the machine has by looking at how it responds to different types of crafted packets.
    - Vulnerability scanning: Scanning a network to identify specific exploits.

Tools

- Metasploit: A framework that contains scripts of several known exploits for different operating systems and services.
    - RHOST(S): Remote host, which is the target machine
- Nessus: Vulnerability scanner used by penetration testers to identify relevant security risks.
- Nmap: Network scanner. It works by sending specific packets and analysing their responses to determine certain information, such as running OS, open ports, etc.
    - -O flag: Enables OS detection.
    - -sS flag: Runs a stealth scan (sends a packet to the port but does not complete the full TCP connection, making it harder to detect).
    - -sV flag: Enables a service and version detection.
    - -T4 flag: Determines how quickly to perform the scan
- Hydra: Network login cracker used to carry out brute-force attacks (guessing a password) on several vulnerable servers.
    - -t flag: Number of threads (more is faster)
    - -l flag: Username to try to guess the password for
    - -P flag: Path to password dictionary
    - -vV flag: Verbose – Show login and password of each attempt

Scanning and Enumeration

To verify that Lorenzo's Virtual Machine (VM) is up and running, I pinged its IP address as a first step – Figure 1.1.

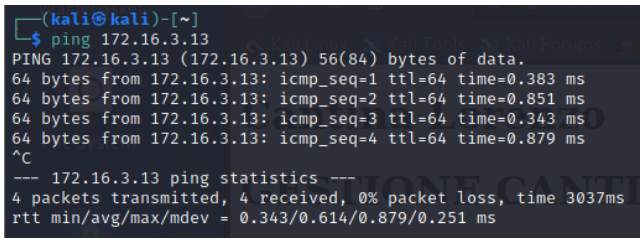


Figure 1.1: Simple ping

To get an initial overview of what vulnerabilities might be present, let us run an advanced scan with Nessus – Figure 1.2.

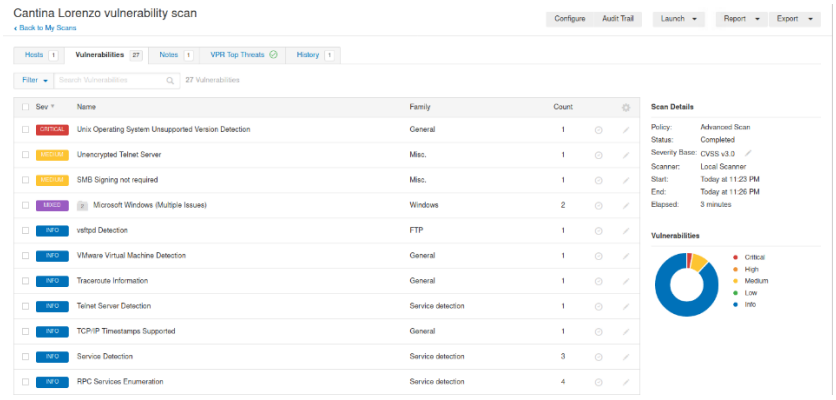


Figure 1.2: Nessus vulnerability scan results

We can already see a few issues with this VM.

- The Unix Operating System is outdated.
    - This can cause several problems. Outdated operating systems usually have a history of known vulnerabilities, making it easy for an attacker to exploit.
- The Telnet server is not encrypted
    - Telnet is not a very secure network protocol, as it is unencrypted by default, making all the data transmitted over it readable in plaintext.

Let us run Nmap to find which ports are open for a connection, grab their banner, and fingerprint the OS – Figure 1.3.

- To run this, I used Zenmap, a simple User Interface for Nmap.

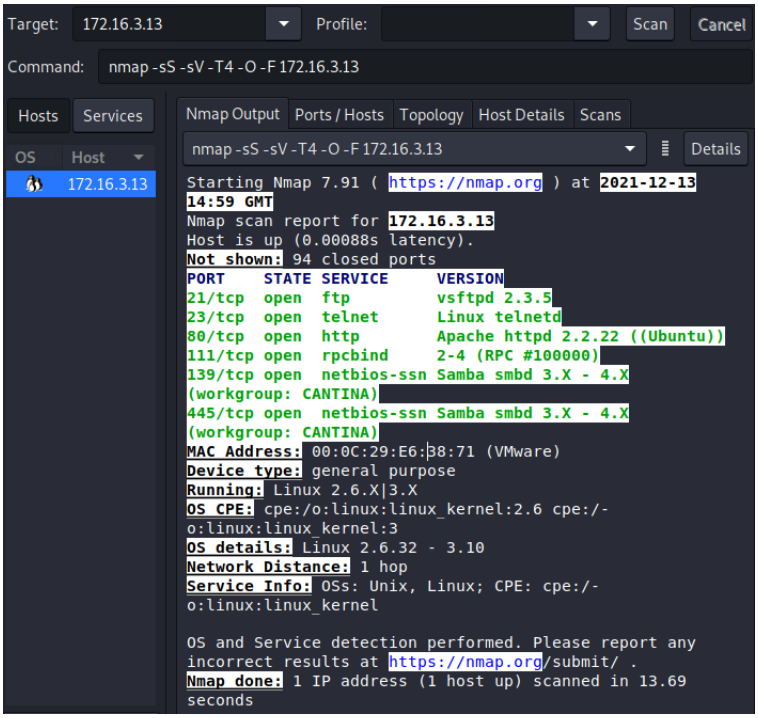


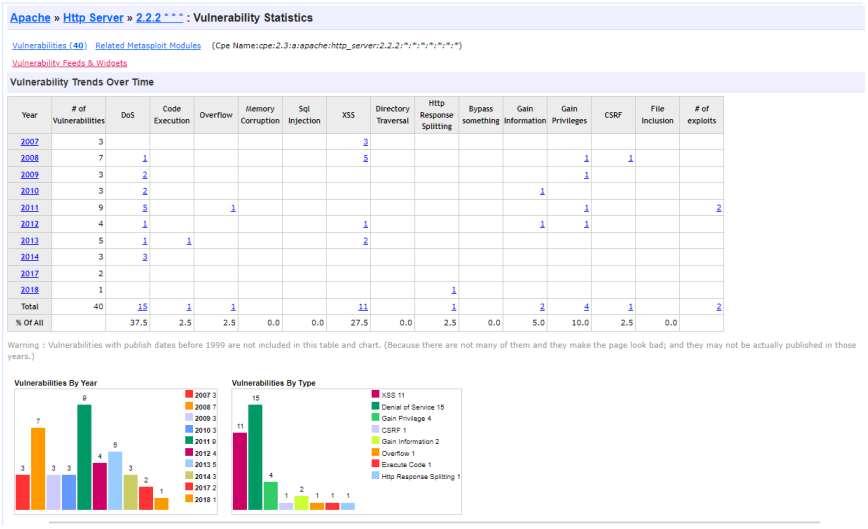
Figure 1.3: Nmap's results show each service's version and the OS information

These results give a good starting point to find ways to interfere with the cellar management machine.

Exploit One: Distributed Denial of Service (DDoS) attack

One bit of information that seemed relevant was the port 80's HTTP version.

A quick search on a vulnerabilities database (CVE)[1] for Apache v2.2.22 shows that most versions of 2.2.x are very susceptible to Denial of Service attacks – Figure 2.1.



Apache » Http Server » 2.2.2 * * * : Vulnerability Statistics

Vulnerabilities (40) Related Metasploit Modules (Cpe Name:cpe:2.3:a:apache:http_server:2.2.2:*:*:*:*:*:*:*)

Vulnerability Feeds & Widgets

Vulnerability Trends Over Time

Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
2007	3						3								
2008	7	1					5					1	1		
2009	3	2										1			
2010	3	2									1				
2011	9	5		1								1			2
2012	4	1					1				1	1			
2013	5	1	1				2								
2014	3	3													
2017	2														
2018	1								1						
Total	40	15	1	1			11		1		2	4	1		2
% Of All		37.5	2.5	2.5	0.0	0.0	27.5	0.0	2.5	0.0	5.0	10.0	2.5	0.0	

Warning : Vulnerabilities with publish dates before 1999 are not included in this table and chart. (Because there are not many of them and they make the page look bad; and they may not be actually published in those years.)

Vulnerabilities By Year

Vulnerabilities By Type

Figure 2.1: In 11 years, there have been 15 known DoS vulnerabilities with v2.2.x

I searched Metasploit for DDoS attacks and decided to use Slowloris[2] – Figure 2.2.

Slowloris sends legit HTTP requests to the server, making sure to read them extra slowly. The target machine, thinking that the connections will be completed, does not block incoming connections.

```
msf6 > search denial of service attack slowloris

Matching Modules
================

   #  Name                                Disclosure Date  Rank    Check  Description
   -  ----                                ---------------  ----    -----  -----------
   0  auxiliary/dos/http/slowloris        2009-06-17       normal  No     Slowloris Denial of Service Attack
```

Figure 2.2: Slowloris exploit in Metasploit

We can see in figure 2.3 that Lorenzo's machine has no incoming traffic.

```
cantina@ubuntu-virtual-machine:~$ netstat -ntu|awk '{print $5}'|cut -d: -f1 -s|sort|uniq -c|sort -nk1 -r
cantina@ubuntu-virtual-machine:~$
```

Figure 2.3: This script prints out a list of connected IPs and their currently active connections

Let us send 1000 sockets (Slowloris sends HTTP headers) and see how the target machine reacts – Figures 2.4 and 2.5.

[1] https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-411922/Apache-Http-Server-2.2.2.html

[2] https://www.rapid7.com/db/modules/auxiliary/dos/http/slowloris/

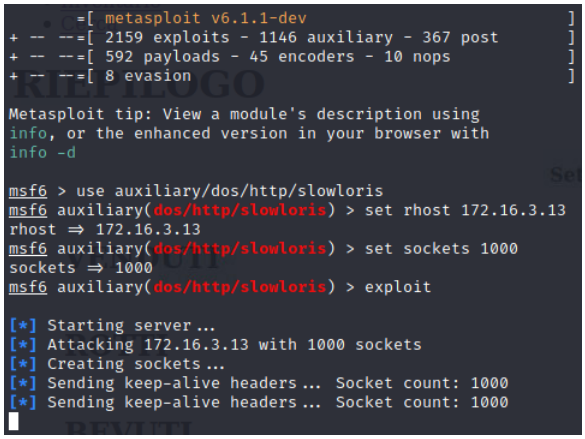


Figure 2.4: Start the exploit.

```
cantina@ubuntu-virtual-machine:~$ netstat -ntu|awk '{print $5}'|cut -d: -f1 -s|sort|uniq -c|sort -nk1 -r
    649 172.16.3.11
cantina@ubuntu-virtual-machine:~$ netstat -ntu|awk '{print $5}'|cut -d: -f1 -s|sort|uniq -c|sort -nk1 -r
    677 172.16.3.11
cantina@ubuntu-virtual-machine:~$ netstat -ntu|awk '{print $5}'|cut -d: -f1 -s|sort|uniq -c|sort -nk1 -r
    714 172.16.3.11
```

Figure 2.5: Lorenzo's VM has 714 (and increasing) active connections with the attacking machine

This renders Lorenzo's VM very slow, making it difficult for an actual user to utilise and resulting in a Denial of Service – Figure 2.6.

Figure 2.6: Connecting to the Cellar Management system becomes challenging.

Preventive measures for this type of attack should be implemented, as having a slow system can affect both business and customer experience, resulting in a substantial loss of money.
To avoid a DDoS attack, limiting the number of connections an IP address can have simultaneously is advisable[3]. It is also good practice to insert a timeout after a determined connection time.

[3] https://www.netscout.com/what-is-ddos/slowloris-attacks

Exploit Two: Brute-force attack

Improper restriction of excessive authentication attempts can make a system vulnerable to brute-force attacks. These attacks are automated “password guessing” scripts that try every combination of characters in an effort to guess the user’s credentials (often starting from a publicly available dictionary of commonly used passwords).

For this security test, I was only provided with the password for the user “cantina”, but there are two more users. I decided to test whether the password utilised for user “lorenzo" was secure enough.
I started by downloading the most popular password dictionary available on the internet, “rockyou.txt”, which comes from a data breach of the social network RockYou that happened in 2009.

I decided to use Hydra, a popular brute-force network attacker – Figure 3.1.

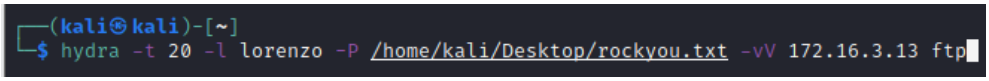


Figure 3.1: Using the rockyou.txt dictionary to guess the user credentials

After less than 10 minutes, Hydra found the VM’s password – Figure 3.2.

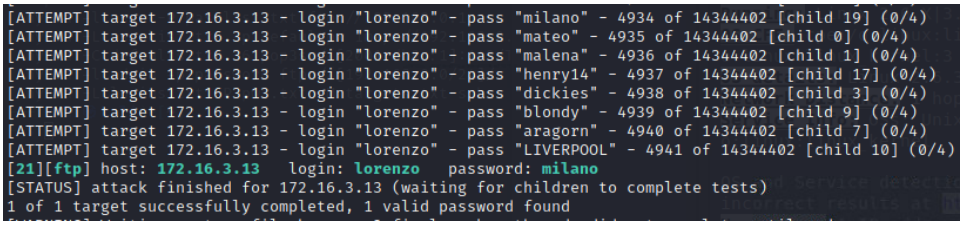


Figure 3.2: The password was easily guessable. It contains no symbols, and it is a simple word

With this password, I was able to connect remotely through ftp or telnet and verified that I had reading\writing permissions – Figure 3.3.

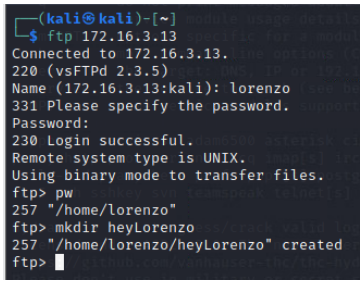


Figure 3.3: Creating a directory after a remote ftp login

Not using a strong password comes with many risks. I was able to access an administrator account in less than 10 minutes. This allows an attacker to do anything they want with the system, as we will see in the third exploit.
To prevent an attacker from brute-forcing a system, other than using a stronger password, the best way would be only allowing whitelisted IPs. Two-factor authentication is also a solid choice.

Exploit Three: Fork bomb

Now that we have remote access to an Administrator user of Lorenzo's machine, we can also read and write scripts remotely.

Pivoting off the brute-force attack, we can now run a local DoS attack. I created a Python script that when executed, keeps cloning itself recursively, overloading the system's CPU and memory. This is called a fork bomb.

First, let us check Lorenzo's VM normal CPU usage – Figure 4.1.

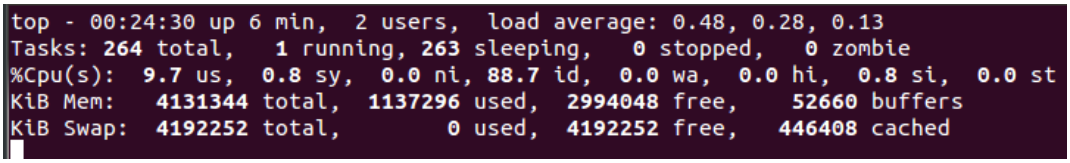


Figure 4.1: The normal usage with no open processes (in this environment) is always around 10%.

Let us access the server with Telnet and the credentials we obtained and create a standard fork bomb python script – Figures 4.2 and 4.3.

Figure 4.2: Creating the forkBomb.py script

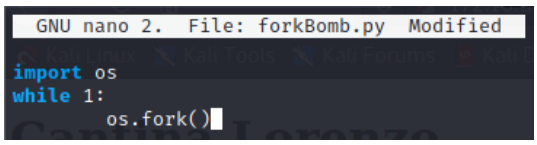


Figure 4.3: Structure of the script

This script imports the OS Python module and keeps cloning itself (forking) recursively until the machine's CPU and memory resources are exhausted.

When we run this script (Figure 4.4), the target VM freezes and will need a reset (to kill all the processes). It is impossible to verify the CPU usage after running the script, as nothing on that machine will work.

```
lorenzo@ubuntu-virtual-machine:~$ sudo nano forkBomb.py
[sudo] password for lorenzo:
lorenzo@ubuntu-virtual-machine:~$ ls
Desktop     examples.desktop  index.php  Public
Documents   forkBomb.py       Music      Templates
Downloads   heyLorenzo        Pictures   Videos
lorenzo@ubuntu-virtual-machine:~$ python forkBomb.py
```

Figure 4.4: Running the fork bomb remotely

If we restart the server, we can see that the file was created and is still on the Desktop – Figure 4.5.

Figure 4.5: The fork bomb

It is self-evident that preventing fork bombs is very important. This can be done by limiting the number of maximum processes a user can create and keeping active simultaneously.

Further possible vulnerabilities

Entering Lorenzo's VM IP address in the browser of my attacking machine gave me direct unprotected access to the Cellar's management system's database – Figure 5.1.

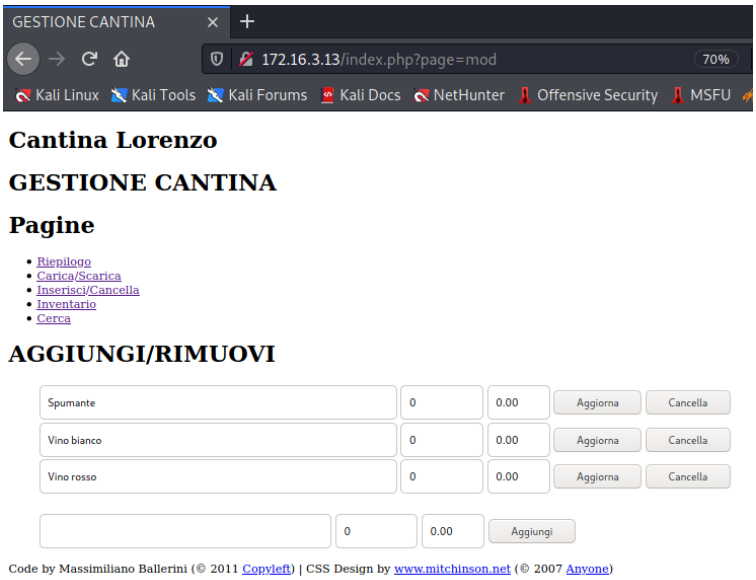


Figure 5.1: Adding to and updating the database does not require authentication

This allows me to change Lorenzo's stock directly without entering user credentials.
This is a significant security risk, as it may cause issues and result in counterfeit stock numbers. To mitigate this, an authentication and login structure (or some whitelisting) should be implemented.

I tried to exploit the available inputs with an SQL injection to break the code, but it did not yield relevant results. However, this should be looked at more in-depth, as appending SQL syntax to the link seems to be read by the system and causes an error - Figure 5.2.

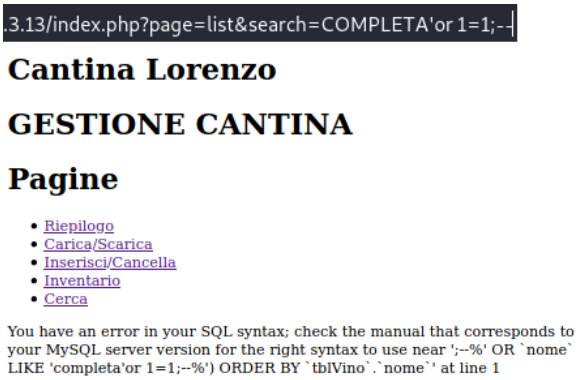


Figure 5.2: Appending SQL code to the link results in an SQL syntax error

Another possible vulnerability (once gained access to one of the Administrator accounts) is the ability by an attacker to decrypt the rest of the users' credentials by running */etc/passwd* and */etc/shadow* with John the Ripper, a password cracking tool. I did not attempt this, but a simple *locate /etc/passwd* and *locate /etc/shadow* shows that these files are present, so they could also be exploited.

Conclusion

Lorenzo's VM should be updated to the latest Unix version, but it could also benefit from a complete upgrade to a more modern Operating System.
Preventive measures for common types of attacks should also be integrated, regardless of which operating system the machine is running. Only certain IP addresses should be allowed to access the stock's management page so that only authorised personnel can modify it.

References

Apache Http Server version 2.2.2 : Security vulnerabilities, 2021. [online]. Available from: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-411922/Apache-Http-Server-2.2.2.html [Accessed 15 December 2021].

Common Password List (rockyou.txt), 2021. [online]. Available from: https://www.kaggle.com/wjburns/common-password-list-rockyoutxt [Accessed 13 December 2021].

CVE-2007-6422 : The balancer_handler function in mod_proxy_balancer in the Apache HTTP Server 2.2.0 through 2.2.6, when a threaded Multi, 2021. [online]. Available from: https://www.cvedetails.com/cve/CVE-2007-6422/ [Accessed 13 December 2021].

CWE - CWE-307: Improper Restriction of Excessive Authentication Attempts (4.6), 2021. [online]. Available from: https://cwe.mitre.org/data/definitions/307.html [Accessed 14 December 2021].

Slowloris Denial of Service Attack, 2021. [online]. Available from: https://www.rapid7.com/db/modules/auxiliary/dos/http/slowloris/ [Accessed 13 December 2021].

What is a Fork Bomb (Rabbit Virus) | DDoS Attack Glossary | Imperva, 2021. [online]. Available from: https://www.imperva.com/learn/ddos/fork-bomb/ [Accessed 14 December 2021].

What is a Slowloris DDoS Attack?, 2021. [online]. Available from: https://www.netscout.com/what-is-ddos/slowloris-attacks [Accessed 14 December 2021].