

Duale Hochschule Baden-Württemberg  
Mannheim

## **Pentest Report**

**Pentesting Project X**

### **Studiengang Cyber Security**

Verfasser:	Luka Tsipitsoudis
Matrikelnummer:	4110112
Kurs:	TINF20CS1
Abgabedatum:	18.04.2023
Betreuer:	Pr. Dr. Johannes Bauer

Unterschrift: \_\_\_\_\_

# **Abstract**

Deutsche Version

---

## Englische Version

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>v</b>
<b>Tabellenverzeichnis</b>	<b>vi</b>
<b>Abkürzungsverzeichnis</b>	<b>vii</b>
<b>1 Introduction CHECKS NOCH ENTFERNEN</b>	<b>1</b>
1.1 Scope . . . . .	1
1.2 Severities . . . . .	1
1.3 Classification . . . . .	1
1.4 Effort to Fix . . . . .	1
<b>2 Management Summary</b>	<b>2</b>
<b>3 Technical Summary</b>	<b>3</b>
3.1 Findings Overview . . . . .	3
3.2 Used Tools . . . . .	3
<b>4 Findings</b>	<b>4</b>
4.1 Finding 1 - Exact OpenSSH-Version can be determined . . . . .	4
4.1.1 Finding Description . . . . .	4
4.1.2 Finding Impact . . . . .	4
4.1.3 Finding Cause . . . . .	4
4.1.4 Finding Details . . . . .	5
4.1.5 Evaluation of Results . . . . .	5
4.2 Finding 2 - Vulnerable OpenSSH Version . . . . .	6
4.2.1 Finding Description . . . . .	6
4.2.2 Finding Impact . . . . .	6
4.2.3 Evaluation of Results . . . . .	6
4.3 Finding 3 - Exact Apache Version can be determined . . . . .	7
4.3.1 Finding Impact . . . . .	7
4.3.2 Finding Details . . . . .	7
4.3.3 Evaluation of Results . . . . .	8
4.4 Finding 4 - Vulnerable Apache Version . . . . .	8
4.4.1 Finding Impact . . . . .	8

4.4.2	Finding Details . . . . .	9
4.4.3	Evaluation of Results . . . . .	9
<b>5</b>	<b>Results</b>	<b>10</b>

# Abbildungsverzeichnis

Abbildung 4.1 Apache Version . . . . .	7
--	---

# Tabellenverzeichnis

# Abkürzungsverzeichnis

<b>DUT</b>	Device Under Testing
<b>AJP</b>	Apache JServ Protocol
<b>CVE</b>	Common Vulnerabilities and Exposures



# 1 Introduction CHECKS NOCH ENTFERNEN

## 1.1 Scope

Who gave the assignment? What is the Device under Test (DUT)? What have you received for the test (hardware, information, etc.)? Add email with hash sum of the mail to the report.

## 1.2 Severities

For each vulnerability you uncover in your testing, you typically provide: The likelihood of exploitation, taking into account how easy it is to discover and exploit The impact on the control you get once exploited Suggested remediation Suggested validation of remediation effectiveness

Low Moderate High Severe Critical

## 1.3 Classification

## 1.4 Effort to Fix

Low Moderate High

## 2 Management Summary

Short Summary for non technical people. what are key takeaways and recommendations? how urgent is acting necessary?

## 3 Technical Summary

summery for technical people

### 3.1 Findings Overview

table that contains the findings you'll describe later, sorted by severity Helps quickly triage results

### 3.2 Used Tools

## 4 Findings

### 4.1 Finding 1 - Exact OpenSSH-Version can be determined

**Classification:** Information Disclosure **CVE:** **Severity:** **Low**

#### 4.1.1 Finding Description

A nmap port scan reveals the exact version of the running OpenSSH-Server on the Device Under Testing (DUT). The version used on the DUT is **"OpenSSH 8.4p1 Debian 5+deb11u1"** and can be accessed via port 22.

#### 4.1.2 Finding Impact

This information can be used by an attacker to find known vulnerabilities in this specific OpenSSH-Version to exploit the DUT. Possible exploitations can be found in Finding 2.

#### 4.1.3 Finding Cause

This finding is caused by OpenSSH itself. There is no configuration option to hide the version of the SSH-Server. The version-banner can be found in the sshd binary.

### 4.1.4 Finding Details

```
1 $ nmap -A 172.16.0.29
2 Starting Nmap 7.91 ( https://nmap.org ) at 2023-03-06 09:30 CEST
3 Nmap scan report for 172.16.0.29
4 Host is up (0.00051s latency).
5 PORT STATE SERVICE VERSION
6 22/tcp open  ssh OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
```

### 4.1.5 Evaluation of Results

Effort to Fix: **Medium**

To fix this finding the OpenSSH Binary has to be changed. By default the binary can be found at '/usr/sbin/sshd'. Change the binary with hexedit and search for the version banner. After removing the version banner restart the ssh service wit 'systemctl restart sshd.service'. Due to the fact of the risk of working on the binary itself, this finding is rated as medium effort to fix.

## 4.2 Finding 2 - Vulnerable OpenSSH Version

**Classification:** Vulnerable Software Version      **Severity:** **Medium**  
**CVE:** CVE-2021-28041, CVE-2021-41617

### 4.2.1 Finding Description

The DUT is running a vulnerable OpenSSH version (8.4p1). This version is vulnerable to the following CVEs: CVE-2021-28041, CVE-2021-41617.

### 4.2.2 Finding Impact

Following exploits can be used to gain access to the DUT:

**CVE-2021-28041:** This vulnerability enables an attacker to carry out unauthorized code execution on a target system remotely. The vulnerability stems from an error in the ssh-agent, where a remote attacker can lure the victim to connect to a server where the attacker has root access.

**CVE-2021-41617:** When OpenSSH is used with non default configurations privilege escalation is possible. (Check configuration)

### 4.2.3 Evaluation of Results

**Effort to Fix:** **Low**

Update to newer OpenSSH version. This can be done by running the following command:

```
1 $ sudo apt update
2 $ sudo apt install openssh-server
```

## 4.3 Finding 3 - Exact Apache Version can be determined

**Classification:** Information Disclosure **Severity:** Low

**CVE:** Null





Visiting port 80 of the DUT in a web browser with the path `"/home"` reveals the exact version of Apache that is running on the DUT. The version of Apache that is running on the DUT is **"Apache/2.4.54 (Debian)"**. This Finding has a low severity, because it should be more important to use a newer version of Apache to prevent exploits of known vulnerabilities.

### 4.3.1 Finding Impact

This can be used to find known vulnerabilities in the version of Apache that is running on the DUT. These vulnerabilities can be found in chapter 4.

### 4.3.2 Finding Details

## Index of /home

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 <a href="#">Parent Directory</a>		-	
 <a href="#">bingo/</a>	2023-02-12 20:48	-	
 <a href="#">bluey/</a>	2023-02-12 20:48	-	
 <a href="#">root/</a>	2023-02-12 20:48	-	

**Apache/2.4.54 (Debian) Server at 172.16.0.30 Port 80**

Abbildung 4.1: Apache Version

### 4.3.3 Evaluation of Results

Effort to Fix: **Medium**

How do you judge the individual technical findings (severity, likelihood)? What is your suggested remediation, if there is one? How can the customer validate their remediation is effective once implemented?

## 4.4 Finding 4 - Vulnerable Apache Version

**Classification:** Vulnerable Software Version **Severity:** **Medium**

**CVE:** CVE-2023-25690, CVE-2023-27522, CVE-2006-20001,  
CVE-2022-36760, CVE-2022-37436

On port 80 the DUT is running a vulnerable Apache version ("Apache 2.4.54"). This version has multiple vulnerabilities and shouldn't be used in production. The following vulnerabilities are known from Common Vulnerabilities and Exposures (CVE) but haven't been exploited on the DUT. Some of these vulnerabilities may only be exploitable with specific configurations. Nevertheless, all of these vulnerabilities are shown to provide transparency and to show the possible impact of the vulnerabilities.

### 4.4.1 Finding Impact

**CVE-2023-25690:** When the mod\_proxy configuration is enabled a HTTP smuggling attack is possible, which could bypass the access controls.

**CVE-2023-27522:** This vulnerability allows an attacker to send a origin header which contains special characters to the server. This could be used truncate/split the response forwarded to the client.

**CVE-2006-20001:** This vulnerability allows an attacker to send a specific if request to the server, which could be used to crash the process.

**CVE-2022-36760:** Due to an inconsistent interpretation of HTTP requests of



the server it could be possible for attackers to smuggle HTTP requests to the Apache JServ Protocol (AJP) server.

**CVE-2022-37436:** A malicious backend has the ability to terminate the response headers prematurely, leading to certain headers being integrated into the response body. Following headers which serve a security function, they will not be comprehended by the client.

#### 4.4.2 Finding Details

```
1 $ nmap -A 172.16.0.29
2
3 Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-06 09:30 CET
4 Nmap scan report for 172.16.0.29
5 Host is up (0.00051s latency).
6
7 PORT STATE SERVICE VERSION
8 80/tcp open  http  Apache httpd 2.4.54 ((Debian))
```

#### 4.4.3 Evaluation of Results

Effort to Fix: **Low**

To fix this vulnerability the Apache Server has to be updated to a newer version. This could be done with the following command:

```
1 $ apt update && apt install apache2
```

## 5 Results