ATTACK DEFENSE

by PentesterAcademy

Name	AppArmor Profile
URL	https://attackdefense.com/challengedetails?cid=1831
Type	Privilege Escalation : AppArmor

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. Also, note that the IP addresses and domain names might be different in your lab.

Objective: Create an AppArmor profile for a copy of the cat utility and restrict (and audit) it while it tries to read the passwd file.

Theory

AppArmor profiles are simple text files used by AppArmor to enforce MAC policies on a program/binary.

Some important points:

- Absolute paths and wild cards are allowed (for file globbing) for specifying files.

- Supported access types for Files:
    - r (read)
    - w (write)
    - m (memory map as executable)
    - k (file locking)
    - l (creation hard links)
    - ix (to execute another program with the new program inheriting policy)
    - Px (execute under another profile, after cleaning the environment)
    - Cx (execute under a child profile, after cleaning the environment)
    - Ux (execute unconfined, after cleaning the environment)

- AppArmor also supports access controls of:
    - Linux capabilities
    - Network mount, remount and umount
    - Pivot_root
    - ptrace
    - Signal
    - DBus
    - UNIX domain sockets

- Variables can be defined in the profiles and can be manipulated from outside the profile. For example: @{PROC} and @{HOME} (add #include <tunables/global> to the profile file)

- Explicit deny rules are supported to override allow rules.

Apparmor_parser utility is used to load, unload, debug, remove, replace apparmor profile. It supports the following options:

- a Default Action to load a new profile in enforce mode.
- C Loading a new profile in complain mode.
- r Overwrite an existing profile.
- R Remove an existing profile in the kernel.
- V Display the profile version.
- h Display reference guide.

Creating a new AppArmor profile

There are two main ways to do this:

Option 1: Stand-Alone Profile Creation

The aa-genprof utility is used to create a profile affecting a single program/application that runs for a finite amount of time (e.g. web browser, mail client).

AppArmor's aa-genprof profile generating utility runs aa-autodep on the specified program/application to create an approximate profile, sets it to complain mode, reloads it into

AppArmor, marks the log, and prompts the user to execute the program and exercise its functionality.

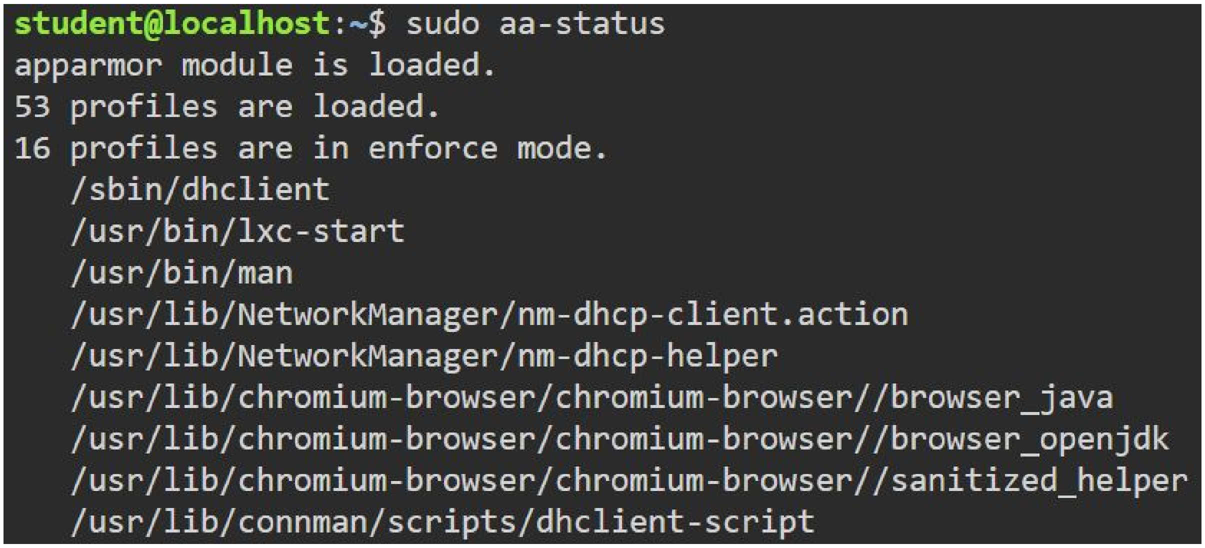
Option 2: Systemic Profile Creation

The aa-autodep utility is used to create a profile affecting multiple programs and/or applications that runs indefinitely or continuously across reboots (e.g. network server)

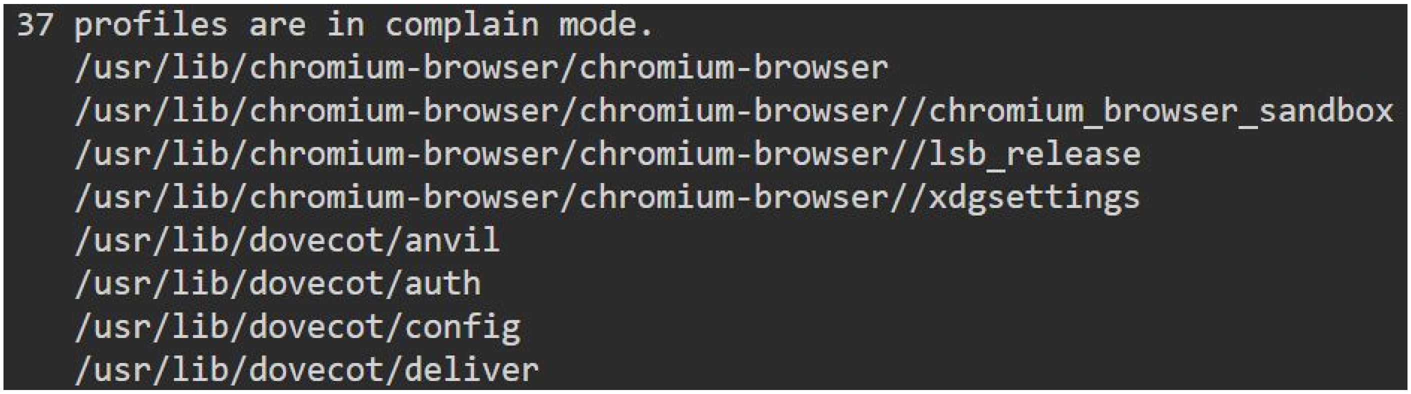
Solution:

Step 1: Check the AppArmor status.

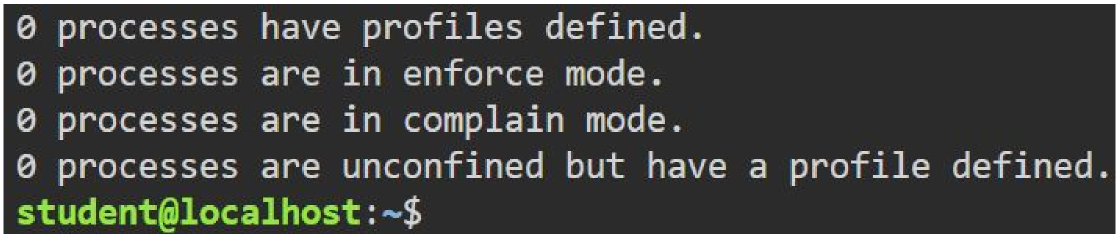
Command: sudo aa-status



```
student@localhost:~$ sudo aa-status
apparmor module is loaded.
53 profiles are loaded.
16 profiles are in enforce mode.
   /sbin/dhclient
   /usr/bin/lxc-start
   /usr/bin/man
   /usr/lib/NetworkManager/nm-dhcp-client.action
   /usr/lib/NetworkManager/nm-dhcp-helper
   /usr/lib/chromium-browser/chromium-browser//browser_java
   /usr/lib/chromium-browser/chromium-browser//browser_openjdk
   /usr/lib/chromium-browser/chromium-browser//sanitized_helper
   /usr/lib/connman/scripts/dhclient-script
```



```
37 profiles are in complain mode.
   /usr/lib/chromium-browser/chromium-browser
   /usr/lib/chromium-browser/chromium-browser//chromium_browser_sandbox
   /usr/lib/chromium-browser/chromium-browser//lsb_release
   /usr/lib/chromium-browser/chromium-browser//xdgsettings
   /usr/lib/dovecot/anvil
   /usr/lib/dovecot/auth
   /usr/lib/dovecot/config
   /usr/lib/dovecot/deliver
```

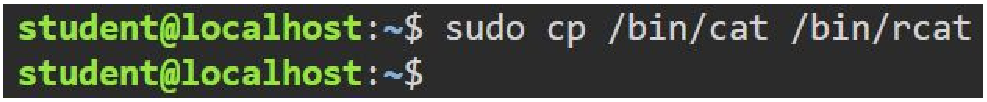


```
0 processes have profiles defined.
0 processes are in enforce mode.
0 processes are in complain mode.
0 processes are unconfined but have a profile defined.
student@localhost:~$
```

Making a copy of system utility

Step 2: Create a copy of the cat binary. Name it as per your liking. Here, it is named “rcat” (stands for restricted cat).

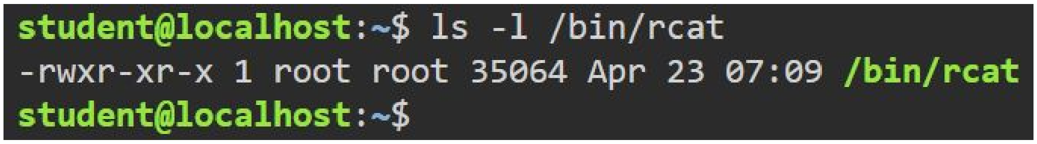
Command: sudo cp /bin/cat /bin/rcat



```
student@localhost:~$ sudo cp /bin/cat /bin/rcat
student@localhost:~$
```

Step 3: Check the file permissions on the rcat binary.

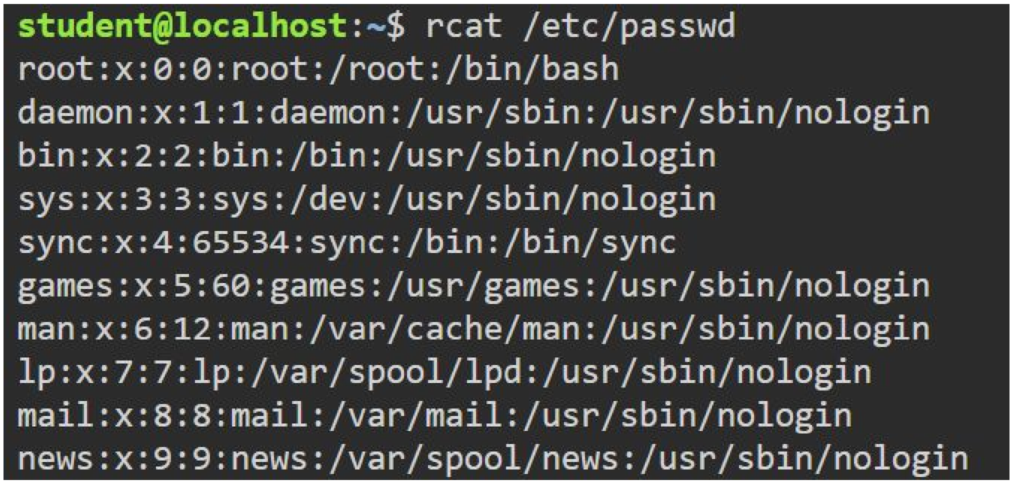
Command: ls -l /bin/rcat



```
student@localhost:~$ ls -l /bin/rcat
-rwxr-xr-x 1 root root 35064 Apr 23 07:09 /bin/rcat
student@localhost:~$
```

Step 4: Print the contents of /etc/passwd using rcat command. The /etc/passwd file is readable to all users, so rcat will succeed.

Command: rcat /etc/passwd

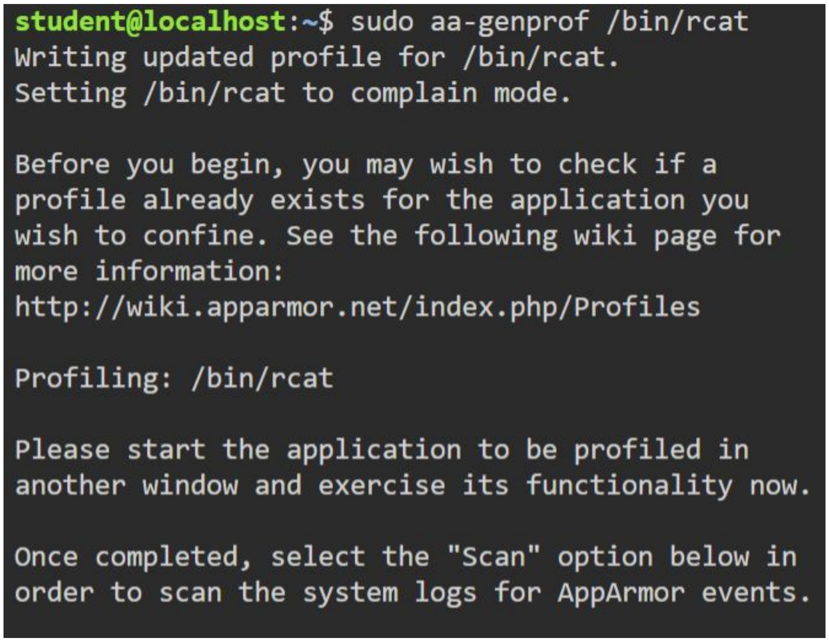


```
student@localhost:~$ rcat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
```

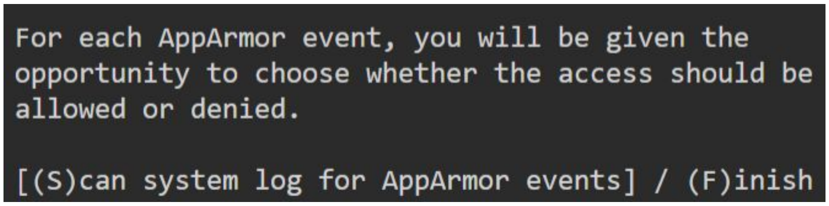
Generating profile for rcat

Step 5: To confine the rcat using the apparmor, a profile is needed. The profile can be generated using aa-genprof command.

Command: sudo aa-genprof /bin/rcat



```
student@localhost:~$ sudo aa-genprof /bin/rcat
Writing updated profile for /bin/rcat.
Setting /bin/rcat to complain mode.

Before you begin, you may wish to check if a
profile already exists for the application you
wish to confine. See the following wiki page for
more information:
http://wiki.apparmor.net/index.php/Profiles

Profiling: /bin/rcat

Please start the application to be profiled in
another window and exercise its functionality now.

Once completed, select the "Scan" option below in
order to scan the system logs for AppArmor events.
```

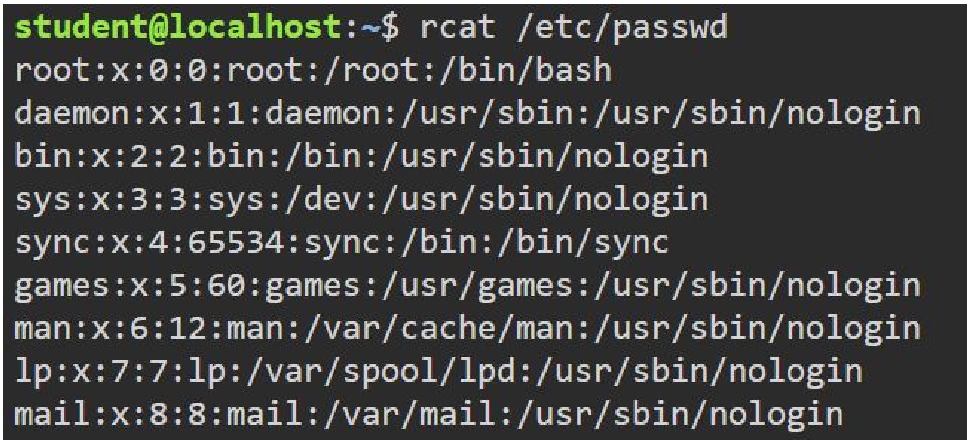


```
For each AppArmor event, you will be given the
opportunity to choose whether the access should be
allowed or denied.

[(S)can system log for AppArmor events] / (F)inish
```

This command is now monitoring the rcat binary (/bin/rcat) for any action that it might perform. These actions will be then used to create the profile. Once the action is performed, S key (case-insensitive) needs to be pressed.

Step 6: Open a new terminal (T2) and again print the contents of /etc/passwd using rcat.

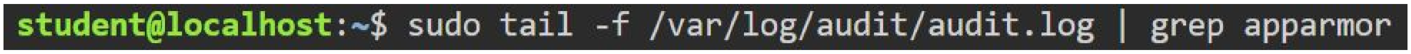
Command: rcat /etc/passwd



```
student@localhost:~$ rcat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
```

Step 7: Open another terminal (T3) and start tail on the audit.log

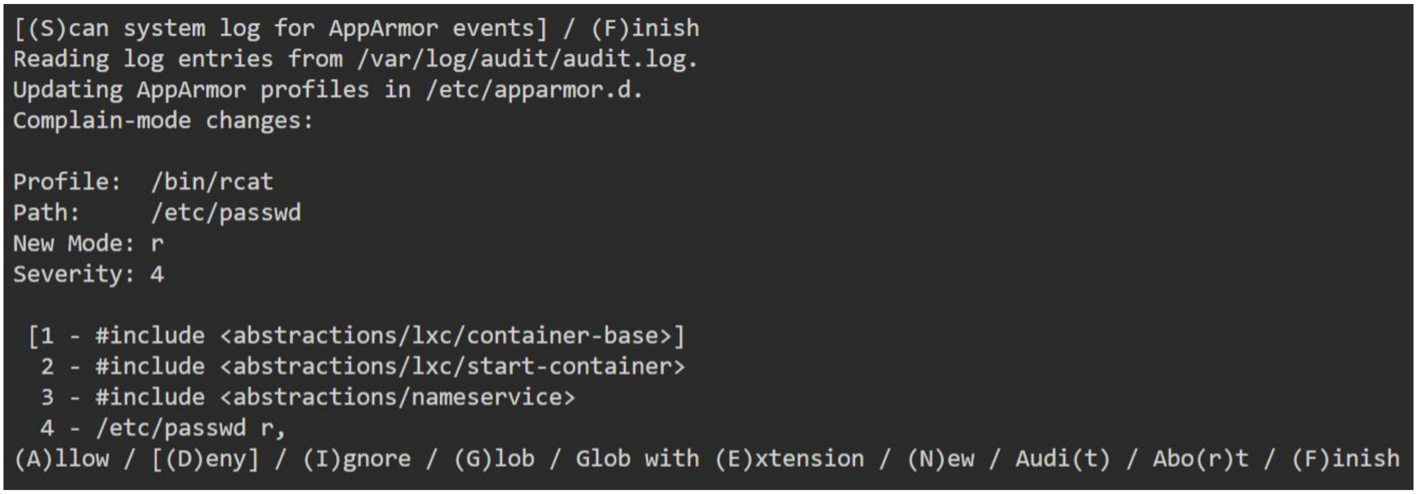
Command: sudo tail -f /var/log/audit/audit.log | grep apparmor



```
student@localhost:~$ sudo tail -f /var/log/audit/audit.log | grep apparmor
```

Step 8: Switch back to terminal T1 and press the s key.

Key pressed: s

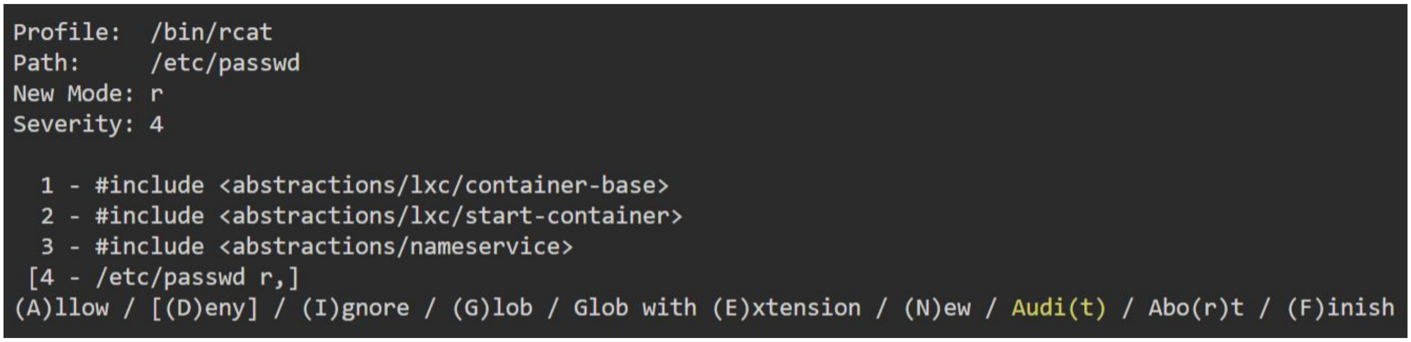


```
[(S)can system log for AppArmor events] / (F)inish
Reading log entries from /var/log/audit/audit.log.
Updating AppArmor profiles in /etc/apparmor.d.
Complain-mode changes:

Profile:  /bin/rcat
Path:     /etc/passwd
New Mode: r
Severity: 4

 [1 - #include <abstractions/lxc/container-base>]
  2 - #include <abstractions/lxc/start-container>
  3 - #include <abstractions/nameservice>
  4 - /etc/passwd r,
(A)llow / [(D)eny] / (I)gnore / (G)lob / Glob with (E)xtension / (N)ew / Audi(t) / Abo(r)t / (F)inish
```

This utility will check the entries from audit.log and detect the action (i.e. printing /etc/passwd) and list those here. The point of interest is number 4.

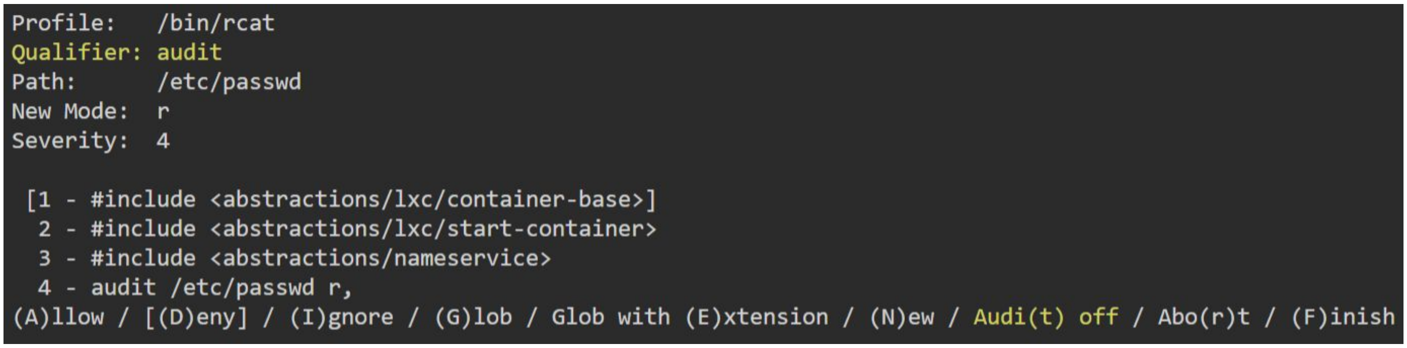
Step 9: Move to point 4 using arrow keys and put this in audit mode by pressing “t” key.

Key pressed: t



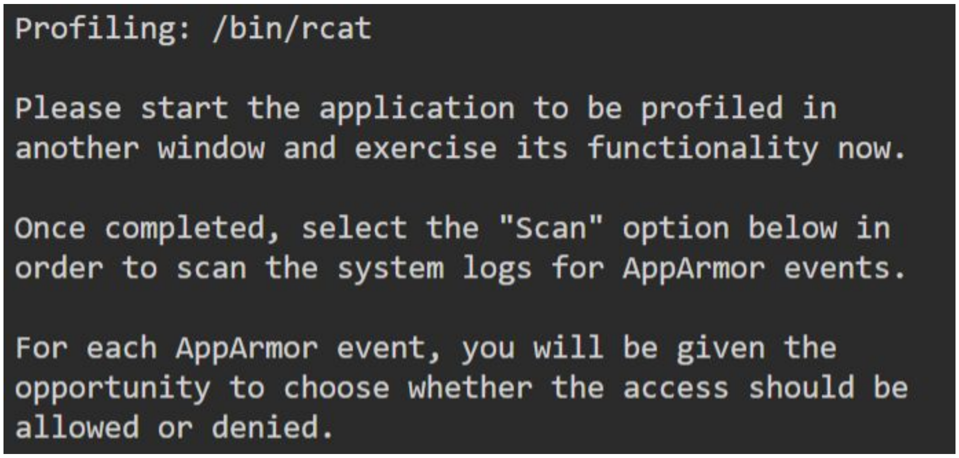
```
Profile:  /bin/rcat
Path:     /etc/passwd
New Mode: r
Severity: 4

  1 - #include <abstractions/lxc/container-base>
  2 - #include <abstractions/lxc/start-container>
  3 - #include <abstractions/nameservice>
 [4 - /etc/passwd r,]
(A)llow / [(D)eny] / (I)gnore / (G)lob / Glob with (E)xtension / (N)ew / Audi(t) / Abo(r)t / (F)inish
```

After pressing the key, the Qualifier field will be set for this entry.



```
Profile:   /bin/rcat
Qualifier: audit
Path:      /etc/passwd
New Mode:  r
Severity:  4

 [1 - #include <abstractions/lxc/container-base>]
  2 - #include <abstractions/lxc/start-container>
  3 - #include <abstractions/nameservice>
  4 - audit /etc/passwd r,
(A)llow / [(D)eny] / (I)gnore / (G)lob / Glob with (E)xtension / (N)ew / Audi(t) off / Abo(r)t / (F)inish
```

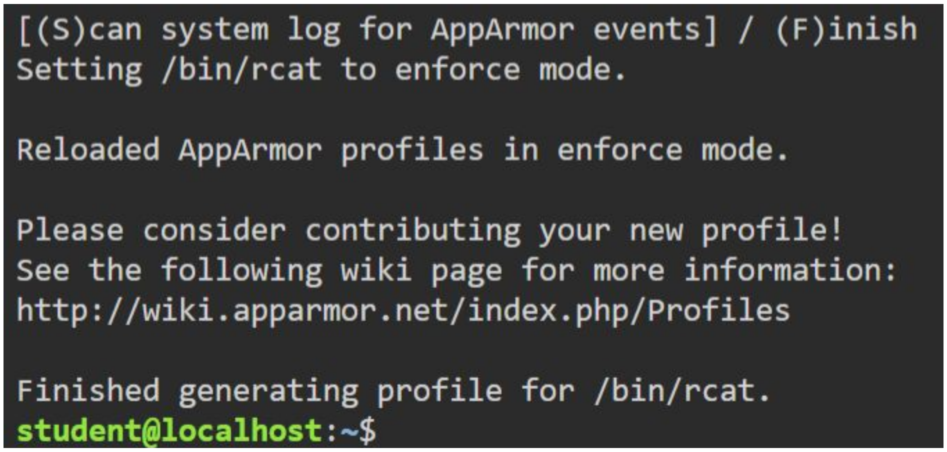
To save the changes, finish the process by pressing the “F” button.



```
Profiling: /bin/rcat

Please start the application to be profiled in
another window and exercise its functionality now.

Once completed, select the "Scan" option below in
order to scan the system logs for AppArmor events.

For each AppArmor event, you will be given the
opportunity to choose whether the access should be
allowed or denied.
```

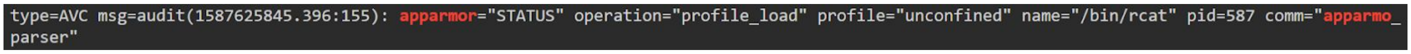
Step 10: Again press F button to finish the process and save the changes

Key pressed: f



```
[(S)can system log for AppArmor events] / (F)inish
Setting /bin/rcat to enforce mode.

Reloaded AppArmor profiles in enforce mode.

Please consider contributing your new profile!
See the following wiki page for more information:
http://wiki.apparmor.net/index.php/Profiles

Finished generating profile for /bin/rcat.
student@localhost:~$
```

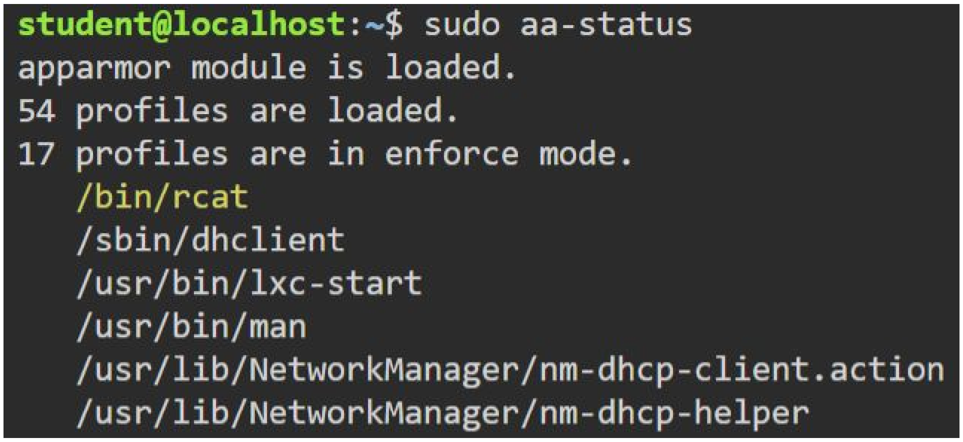
Step 11: In T3, the logs of adding the profile for rcat will be printed.



```
type=AVC msg=audit(1587625845.396:155): apparmor="STATUS" operation="profile_load" profile="unconfined" name="/bin/rcat" pid=587 comm="apparmo_
parser"
```

Step 12: Check the AppArmor status.

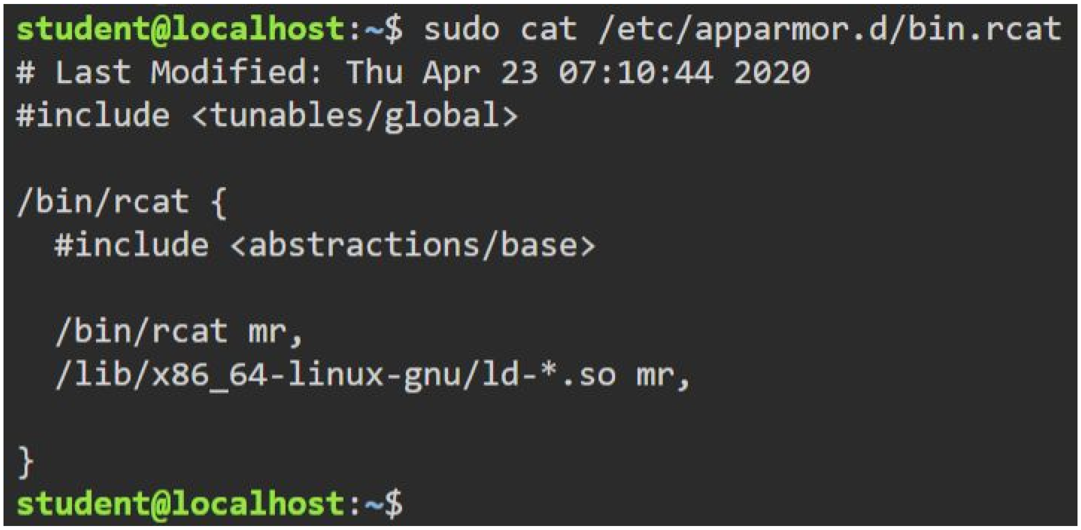
Command: sudo aa-status



```
student@localhost:~$ sudo aa-status
apparmor module is loaded.
54 profiles are loaded.
17 profiles are in enforce mode.
   /bin/rcat
   /sbin/dhclient
   /usr/bin/lxc-start
   /usr/bin/man
   /usr/lib/NetworkManager/nm-dhcp-client.action
   /usr/lib/NetworkManager/nm-dhcp-helper
```

Step 13: Check the AppArmor profile created for rcat.

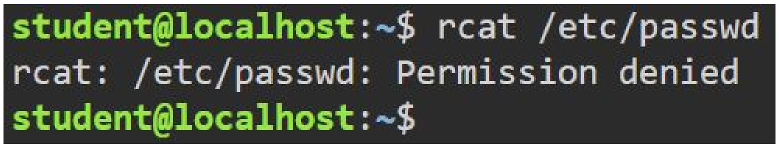
Command: sudo cat /etc/apparmor.d/bin.rcat



```
student@localhost:~$ sudo cat /etc/apparmor.d/bin.rcat
# Last Modified: Thu Apr 23 07:10:44 2020
#include <tunables/global>

/bin/rcat {
  #include <abstractions/base>

  /bin/rcat mr,
  /lib/x86_64-linux-gnu/ld-*.so mr,

}
student@localhost:~$
```

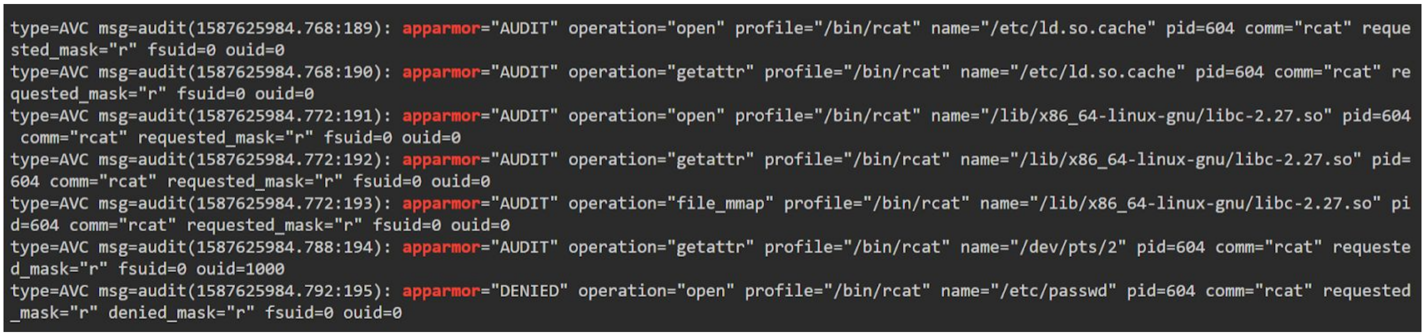
Step 14: Try to print the contents of the /etc/passwd file.

Command: rcat /etc/passwd



```
student@localhost:~$ rcat /etc/passwd
rcat: /etc/passwd: Permission denied
student@localhost:~$
```

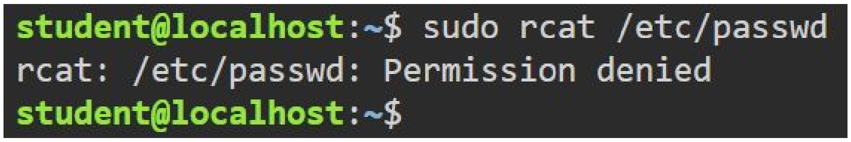
The attempt failed. And, the corresponding logs can be checked in T3.



```
type=AVC msg=audit(1587625984.768:189): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/etc/ld.so.cache" pid=604 comm="rcat" reque
sted_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.768:190): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/etc/ld.so.cache" pid=604 comm="rcat" re
quested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:191): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=604
 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:192): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=
604 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:193): apparmor="AUDIT" operation="file_mmap" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pi
d=604 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.788:194): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/dev/pts/2" pid=604 comm="rcat" requeste
d_mask="r" fsuid=0 ouid=1000
type=AVC msg=audit(1587625984.792:195): apparmor="DENIED" operation="open" profile="/bin/rcat" name="/etc/passwd" pid=604 comm="rcat" requested
_mask="r" denied_mask="r" fsuid=0 ouid=0
```

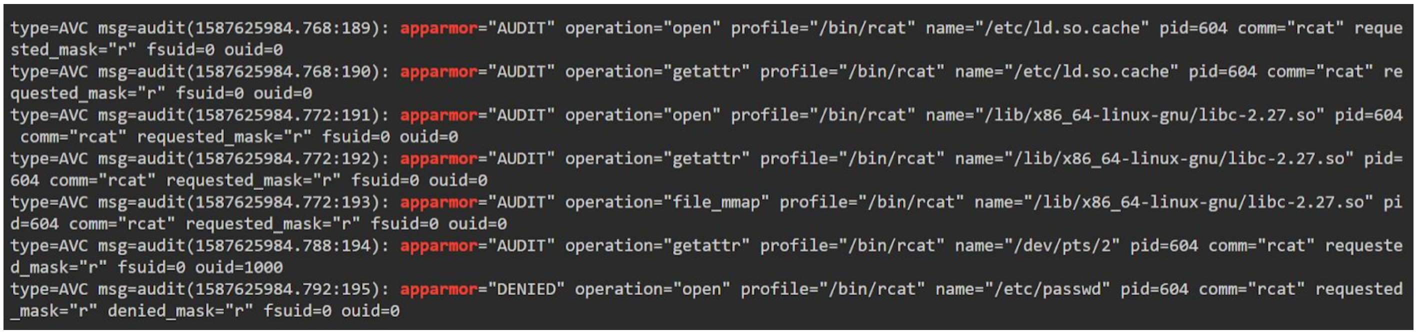
Step 15: Try to print the contents of the /etc/passwd file. This time using sudo.

Command: rcat /etc/passwd



```
student@localhost:~$ sudo rcat /etc/passwd
rcat: /etc/passwd: Permission denied
student@localhost:~$
```

Again, the attempt failed and the corresponding logs can be checked in T3.



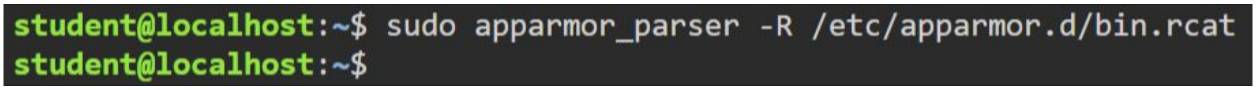
```
type=AVC msg=audit(1587625984.768:189): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/etc/ld.so.cache" pid=604 comm="rcat" reque
sted_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.768:190): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/etc/ld.so.cache" pid=604 comm="rcat" re
quested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:191): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=604
 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:192): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=
604 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.772:193): apparmor="AUDIT" operation="file_mmap" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pi
d=604 comm="rcat" requested_mask="r" fsuid=0 ouid=0
type=AVC msg=audit(1587625984.788:194): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/dev/pts/2" pid=604 comm="rcat" requeste
d_mask="r" fsuid=0 ouid=1000
type=AVC msg=audit(1587625984.792:195): apparmor="DENIED" operation="open" profile="/bin/rcat" name="/etc/passwd" pid=604 comm="rcat" requested
_mask="r" denied_mask="r" fsuid=0 ouid=0
```

So, as the action is restricted and enforced, any user can’t print the /etc/passwd using rcat.

Removing the profile

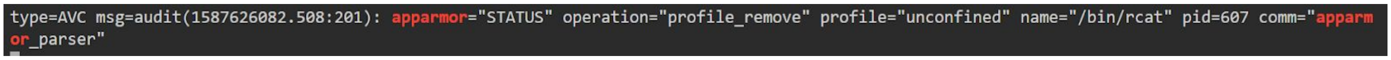
Step 16: Remove the profile from enforcement mode to disable mode. In simple words, disable the profile.

Command: sudo apparmor_parser -R /etc/apparmor.d/bin.rcat



```
student@localhost:~$ sudo apparmor_parser -R /etc/apparmor.d/bin.rcat
student@localhost:~$
```

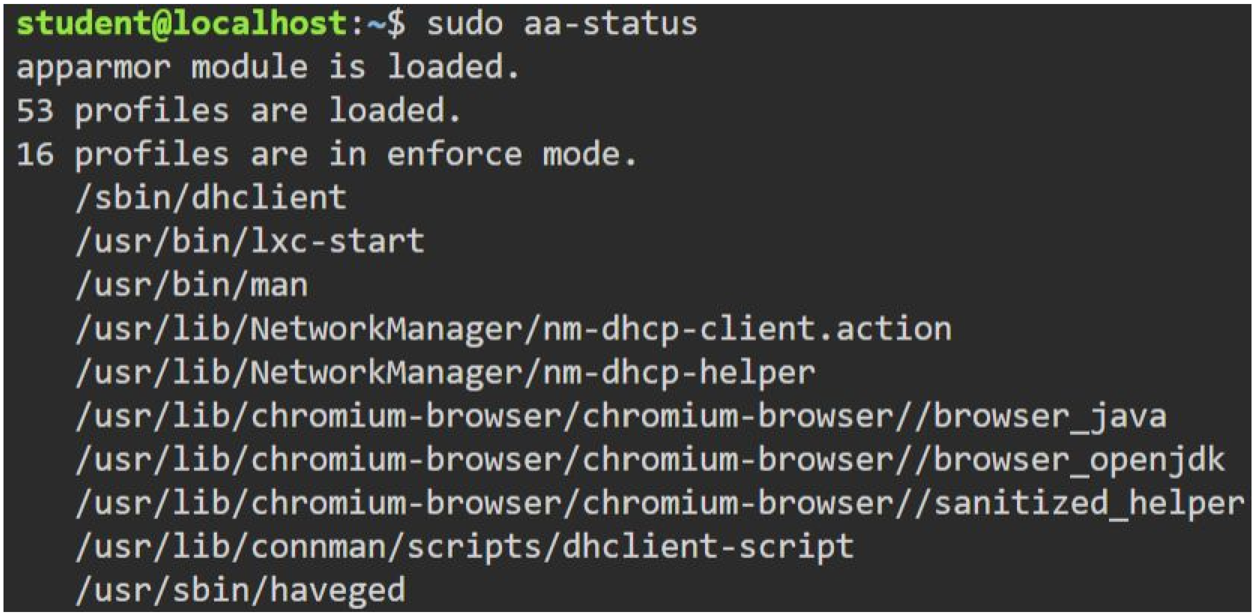
The corresponding logs can be checked in T3.



```
type=AVC msg=audit(1587626082.508:201): apparmor="STATUS" operation="profile_remove" profile="unconfined" name="/bin/rcat" pid=607 comm="apparm
or_parser"
```

Step 17: Also check the apparmor status to verify.

Command: sudo aa-status

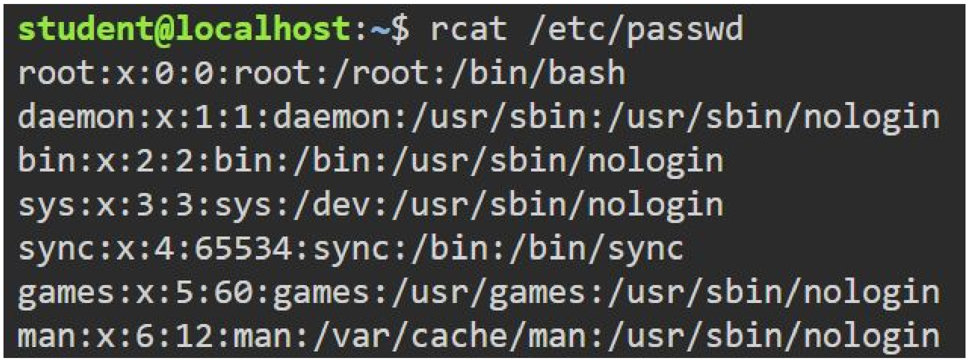


```
student@localhost:~$ sudo aa-status
apparmor module is loaded.
53 profiles are loaded.
16 profiles are in enforce mode.
   /sbin/dhclient
   /usr/bin/lxc-start
   /usr/bin/man
   /usr/lib/NetworkManager/nm-dhcp-client.action
   /usr/lib/NetworkManager/nm-dhcp-helper
   /usr/lib/chromium-browser/chromium-browser//browser_java
   /usr/lib/chromium-browser/chromium-browser//browser_openjdk
   /usr/lib/chromium-browser/chromium-browser//sanitized_helper
   /usr/lib/connman/scripts/dhclient-script
   /usr/sbin/haveged
```

The entry for /bin/rcat is removed.

Step 18: Try to print the contents of the /etc/passwd file.

Command: rcat /etc/passwd



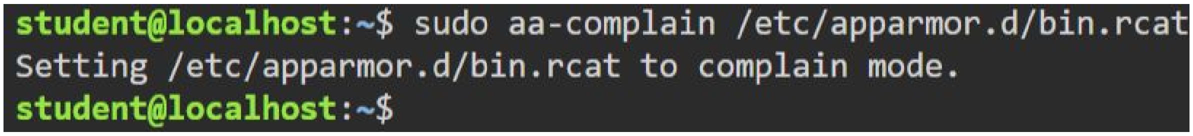
```
student@localhost:~$ rcat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
```

The attempt succeeded as the apparmor profile is disabled

Adding the profile to complain mode

Step 19: Add the rcat profile to complain mode.

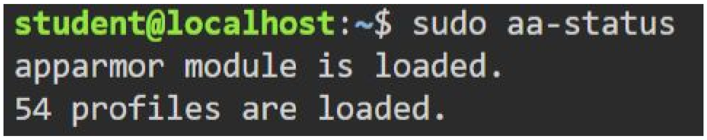
Command: sudo aa-complain /etc/apparmor.d/bin.rcat



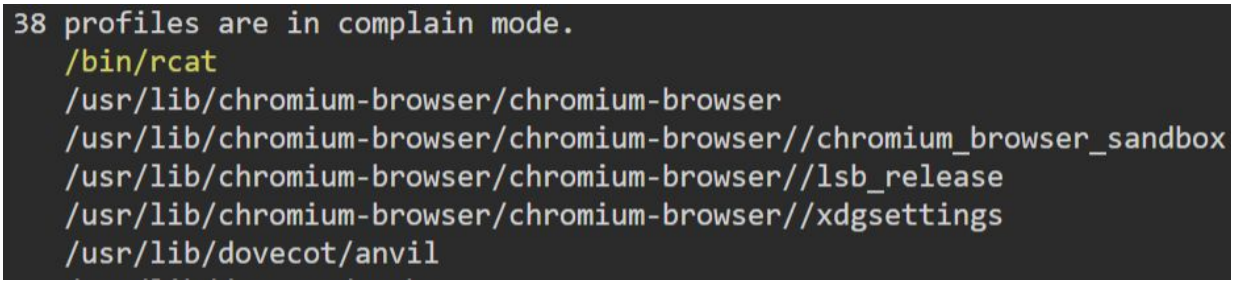
```
student@localhost:~$ sudo aa-complain /etc/apparmor.d/bin.rcat
Setting /etc/apparmor.d/bin.rcat to complain mode.
student@localhost:~$
```

Step 20: Check the apparmor status to verify the change.

Command: sudo aa-status

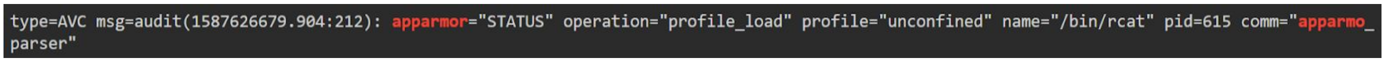


```
student@localhost:~$ sudo aa-status
apparmor module is loaded.
54 profiles are loaded.
```



```
38 profiles are in complain mode.
   /bin/rcat
   /usr/lib/chromium-browser/chromium-browser
   /usr/lib/chromium-browser/chromium-browser//chromium_browser_sandbox
   /usr/lib/chromium-browser/chromium-browser//lsb_release
   /usr/lib/chromium-browser/chromium-browser//xdgsettings
   /usr/lib/dovecot/anvil
```

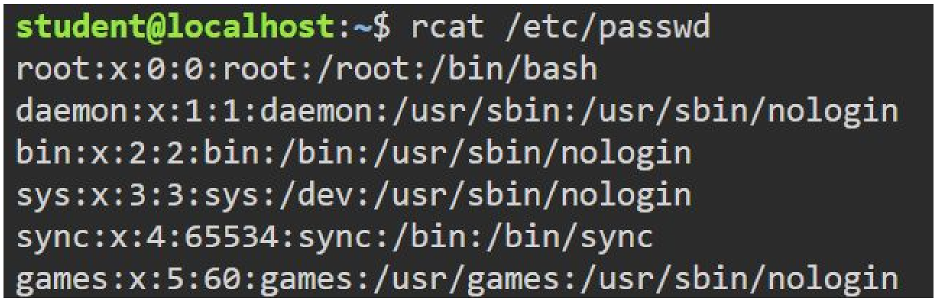
The corresponding logs can be checked in T3.



```
type=AVC msg=audit(1587626679.904:212): apparmor="STATUS" operation="profile_load" profile="unconfined" name="/bin/rcat" pid=615 comm="apparmo_
parser"
```

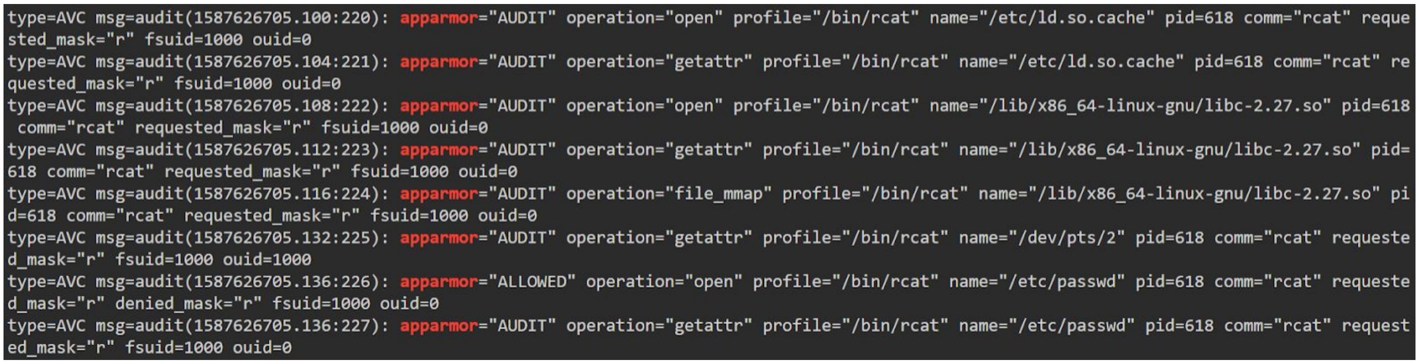
Step 21: Try to print the contents of the /etc/passwd file.

Command: rcat /etc/passwd



```
student@localhost:~$ rcat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
```

As the profile is in complain mode, the corresponding logs can be checked in T3.



```
type=AVC msg=audit(1587626705.100:220): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/etc/ld.so.cache" pid=618 comm="rcat" reque
sted_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.104:221): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/etc/ld.so.cache" pid=618 comm="rcat" re
quested_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.108:222): apparmor="AUDIT" operation="open" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=618
 comm="rcat" requested_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.112:223): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pid=
618 comm="rcat" requested_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.116:224): apparmor="AUDIT" operation="file_mmap" profile="/bin/rcat" name="/lib/x86_64-linux-gnu/libc-2.27.so" pi
d=618 comm="rcat" requested_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.132:225): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/dev/pts/2" pid=618 comm="rcat" requeste
d_mask="r" fsuid=1000 ouid=1000
type=AVC msg=audit(1587626705.136:226): apparmor="ALLOWED" operation="open" profile="/bin/rcat" name="/etc/passwd" pid=618 comm="rcat" requeste
d_mask="r" denied_mask="r" fsuid=1000 ouid=0
type=AVC msg=audit(1587626705.136:227): apparmor="AUDIT" operation="getattr" profile="/bin/rcat" name="/etc/passwd" pid=618 comm="rcat" request
ed_mask="r" fsuid=1000 ouid=0
```

In this manner, the profile can help preventing or auditing (monitoring) the restricted activities.

References:

- AppArmor man page (http://manpages.ubuntu.com/manpages/bionic/man7/apparmor.7.html) (http://manpages.ubuntu.com/manpages/bionic/man5/apparmor.d.5.html)
- Beginning AppArmor profile development (https://ubuntu.com/tutorials/beginning-apparmor-profile-development)
- Advanced Docker Security with AppArmor (https://appfleet.com/blog/advanced-docker-security-with-apparmor/)