



LOG  
ANALYSIS  
AND  
SYSTEMS  
INTEGRATION WITH  
**WAZUH**

**PREPARED BY  
MEHMOODULHASSAN**

FOLLOW ME ON  
[linkedin.com/in/mehmood.ulhassan](https://www.linkedin.com/in/mehmood.ulhassan)

# Linux log analysis

## Logs files:

A log file store events, process, messages and other data from applications, operation system or devices.

They provide information based on the actions perform by the user, playing an important role in monitoring.

So basically, Linux operating systems like kali and ubuntu their logs are stored in the following directory **/var/log**. So basically, there are some important logs file that we need to check and analyze them.

1. Boot
2. Cron
3. Secure
4. maillog
5. Httpd
6. Messages

So here, boot logs are created during the startup of the system. If there is an issue, every single event will be logged.

In the crontab, all logs related to scheduled tasks are created. For example, if we need to update the system, we can schedule the task in crontab. At the specified time, the system will update, and the corresponding event will be logged.

In secure log all the security information are stored like user login activity user logout authentication failure and password related etc.

In Linux, mail logs capture details about email processing, including sender/recipient addresses, delivery status, and errors, as well as connection and authentication attempts.

In Linux, httpd logs record details of web server activity, including client requests, URLs accessed, and server response statuses. They also log errors, warnings, and server issues during operation.

In Linux, messages logs store general system activity and diagnostic information, including kernel messages, boot processes, and system events. They help monitor overall system health and troubleshoot issues.

So, these are some short brief of those logs so now let's check and do some analysis.

```
fox@ubuntuvictim: /var/log
File Edit View Search Terminal Tabs Help
fox@ubuntuvictim: /var/log
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2024-08-22 10:25:38
fox@ubuntuvictim:~$ cd /var/log
fox@ubuntuvictim: /var/log$ ls
alternatives.log      bttmp      mail.err      sysstat
alternatives.log.1    bttmp.1    mail.err.1    tallylog
alternatives.log.2.gz cups        mail.log      ufw.log
apache2               dist-upgrade mail.log.1    ufw.log.1
appport.log           dpkg.log   mail.log.2.gz ufw.log.2.gz
appport.log.1         dpkg.log.1 mail.log.3.gz ufw.log.3.gz
appport.log.2.gz      dpkg.log.2.gz mysql         unattended-upgrades
appport.log.3.gz      faillog    postgresql    vboxadd-install.log
appport.log.4.gz      fontconfig.log samba         vboxadd-setup.log
appport.log.5.gz      gdm3       speech-dispatcher vboxadd-setup.log.1
appport.log.6.gz      gpu-manager.log suricata     vboxadd-setup.log.2
appport.log.7.gz      hp         syslog        vboxadd-setup.log.3
apt                  installer  syslog.1      vboxadd-setup.log.4
auth.log             journal    syslog.2.gz   vsftpd.log
auth.log.1           kern.log   syslog.3.gz   vsftpd.log.1
auth.log.2.gz        kern.log.1 syslog.4.gz    vsftpd.log.2
auth.log.3.gz        kern.log.2.gz syslog.5.gz    wtmp
boot.log             kern.log.3.gz syslog.6.gz    wtmp.1
bootstrap.log        lastlog    syslog.7.gz
fox@ubuntuvictim: /var/log$
```

First of all, as we can see, our Ubuntu logs are located in /var/log, where all the log files are stored. We will install a package and then check to see if the logs are being generated..

```
Activities Terminal
fox@ubuntuvictim: /var/log
File Edit View Search Terminal Tabs Help
fox@ubuntuvictim: /var/log$ sudo apt install stegosuite
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libcommons-cli-java liblogback-java libslf4j-java libswt-cairo-gtk-4-jni libswt-gtk-4-java
  libswt-gtk-4-jni libswt-gtk2-4-jni
Suggested packages:
  groovy libjanino-java libjansi-java libjetty9-java libmail-java libservet3.1-java libtomcat8-java
  libcommons-logging-java liblog4j1.2-java
The following NEW packages will be installed:
  libcommons-cli-java liblogback-java libslf4j-java libswt-cairo-gtk-4-jni libswt-gtk-4-java
  libswt-gtk-4-jni libswt-gtk2-4-jni stegosuite
0 upgraded, 8 newly installed, 0 to remove and 24 not upgraded.
Need to get 3,226 kB of archives.
After this operation, 5,273 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libcommons-cli-java all 1.4-1 [53.8 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libslf4j-java all 1.7.25-3 [141 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 liblogback-java all 1:1.2.3-2ubuntu1-18.04.1 [764 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 libswt-gtk-4-jni amd64 4.9.0-1-18.04.1 [166 kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 libswt-cairo-gtk-4-jni amd64 4.9.0-1-18.04.1 [30.2 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 libswt-gtk-4-java amd64 4.9.0-1-18.04.1 [1,783 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 libswt-gtk2-4-jni amd64 4.9.0-1-18.04.1 [105 kB]
Get:8 http://archive.ubuntu.com/ubuntu bionic/universe amd64 stegosuite all 0.8.0-1 [182 kB]
Fetched 3,226 kB in 4s (765 kB/s)
Selecting previously unselected package libcommons-cli-java.
(Reading database ... 182911 files and directories currently installed.)
Preparing to unpack .../0-libcommons-cli-java_1.4-1_all.deb ...
Unpacking libcommons-cli-java (1.4-1) ...
Selecting previously unselected package libslf4j-java.
```

Now here we can see I just run the apt command to install that utility just to check if their logs.

```

fox@ubuntu-victim: /var/log$ cd
fox@ubuntu-victim: ~$ cd /var/log
fox@ubuntu-victim: /var/log$ ls
alternatives.log  apport.log.6.gz  bttmp.1      hp          mail.log      syslog.1      ufw.log.1      vsftpd.log
alternatives.log.1  apport.log.7.gz  cups         installer   mail.log.1    syslog.2.gz  ufw.log.2.gz  vsftpd.log.1
alternatives.log.2.gz  apt             dist-upgrade  journal     mail.log.2.gz  syslog.3.gz  ufw.log.3.gz  vsftpd.log.2
apache2           auth.log        dpkg.log     kern.log    mail.log.3.gz  syslog.4.gz  unattended-upgrades  wtmp
apport.log        auth.log.1     dpkg.log.1   kern.log.1  mysql         syslog.5.gz  vboxadd-install.log  wtmp.1
apport.log.1      auth.log.2.gz  dpkg.log.2.gz  kern.log.2.gz  postgresql    syslog.6.gz  vboxadd-setup.log
apport.log.2.gz   auth.log.3.gz  faillog      kern.log.3.gz  samba        syslog.7.gz  vboxadd-setup.log.1
apport.log.3.gz   boot.log       fontconfig.log  lastlog     speech-dispatcher  sysstat     vboxadd-setup.log.2
apport.log.4.gz   bootstrap.log  gdm3          mail.err    suricata      tallylog     vboxadd-setup.log.3
apport.log.5.gz   bttmp          gpu-manager.log  mail.err.1  syslog        ufw.log      vboxadd-setup.log.4
fox@ubuntu-victim: /var/log$ cd apt/
fox@ubuntu-victim: /var/log/apt$
eipp.log.xz  history.log  history.log.1.gz  history.log.2.gz  term.log  term.log.1.gz  term.log.2.gz
fox@ubuntu-victim: /var/log/apt$

```

So, since we ran the apt command, we are now checking its logs. Navigate to the location /var/log/apt as mentioned in the figure, and you will find the history.log file. Open it using the nano command

```

fox@ubuntu-victim: /var/log/apt
File Edit View Search Terminal Tabs Help
fox@ubuntu-victim: /var/log/apt x  ubuntu-victim@ubuntu-victim: ~  ubuntu-victim@ubuntu-victim: ~
Install: crunch:amd64 (3.6-2)
End-Date: 2024-08-20 15:00:19

Start-Date: 2024-08-20 15:37:08
Commandline: apt install rdesktop
Requested-By: ubuntu-victim (1000)
Install: libgssglue1:amd64 (0.4-2ubuntu1, automatic), rdesktop:amd64 (1.8.3-2build1)
End-Date: 2024-08-20 15:37:12

Start-Date: 2024-08-21 11:54:18
Commandline: apt install nmap -y
Requested-By: ubuntu-victim (1000)
Install: liblinear3:amd64 (2.1.0+dfsg-2, automatic), nmap:amd64 (7.60-1ubuntu5), libblas3:amd64 (3.7.1-4ubuntu1, automatic)
End-Date: 2024-08-21 11:54:28

Start-Date: 2024-08-22 12:00:53
Commandline: apt install stegosuite
Requested-By: fox (1003)
Install: libswt-gtk2-4-jni:amd64 (4.9.0-1~18.04.1, automatic), libswt-cairo-gtk-4-jni:amd64 (4.9.0-1~18.04.1, automatic), libcommons-cli-java:amd64 (1.4-1, automatic), stegosuite:amd64 (0.8.0-1), libswt-gtk-4-jni:amd64 (4.9.0-1~18.04.1, automatic), libslf4j-java:amd64 (1.7.25-3, automatic), liblogback-java:amd64 (1:1.2.3-2ubuntu1~18.04.1, automatic), libswt-gtk-4-java:amd64 (4.9.0-1~18.04.1, automatic)
End-Date: 2024-08-22 12:01:01

```

Here the log of that installation is being created which means our system logs are monitoring fine.

```
ubuntu-victim@ubuntu-victim: /var/log
File Edit View Search Terminal Tabs Help
fox@ubuntu-victim: /var/log x ubuntu-victim@ubuntu-victim: ~ x ubuntu-victim@ubuntu-victim: /var/log x
ubuntu-victim@ubuntu-victim:~$ sudo apt update
[sudo] password for ubuntu-victim:
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.ubuntu.com/ubuntu bionic-backports InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
24 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu-victim@ubuntu-victim:~$ cat /var/log
cat: /var/log: Is a directory
ubuntu-victim@ubuntu-victim:~$ cd /var/log
ubuntu-victim@ubuntu-victim:/var/log$ ls
alternatives.log  appport.log.6.gz  bttmp.1  hp  mail.log  mail.log.1  syslog.1  ufw.log.1  vsftpd.log
alternatives.log.1  appport.log.7.gz  cups  installer  mail.log.2.gz  mail.log.3.gz  syslog.2.gz  ufw.log.2.gz  vsftpd.log.1
alternatives.log.2.gz  apt  dpkg.log  journal  mail.log.3.gz  mysql  syslog.3.gz  ufw.log.3.gz  vsftpd.log.2
apache2  appport.log  auth.log.1  kern.log  mail.log.3.gz  postgresql  syslog.4.gz  vboxadd-install.log  wtmp
appport.log.1  appport.log.2.gz  auth.log.2.gz  dpkg.log.1  kern.log.1  syslog.5.gz  vboxadd-setup.log  wtmp.1
appport.log.2.gz  appport.log.3.gz  auth.log.3.gz  dpkg.log.2.gz  kern.log.2.gz  syslog.6.gz  vboxadd-setup.log.1
appport.log.3.gz  appport.log.4.gz  boot.log  faillog  kern.log.3.gz  syslog.7.gz  vboxadd-setup.log.2
appport.log.4.gz  appport.log.5.gz  bootstrap.log  fontconfig.log  lastlog  sysstat  vboxadd-setup.log.3
appport.log.5.gz  bttmp  gdm3  mail.err  mail.err.1  syslog  ufw.log  vboxadd-setup.log.4
ubuntu-victim@ubuntu-victim:/var/log$ sudo cat mail.log
```

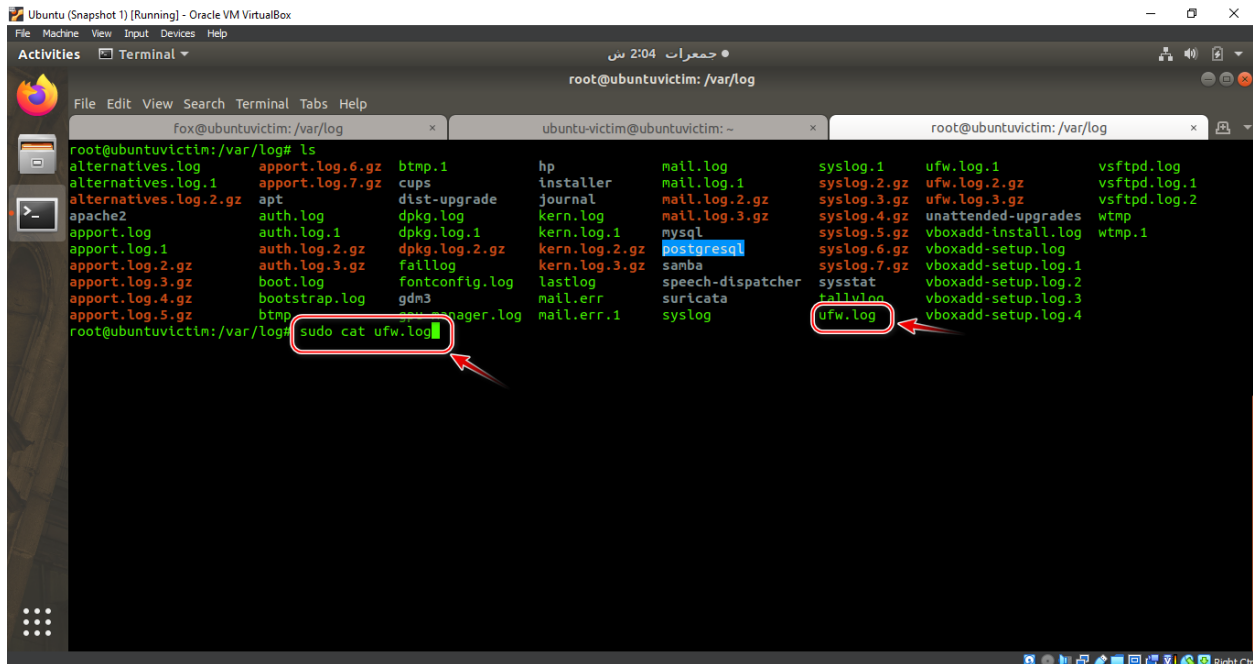
Now, here is another log file named mail.log. This file contains all logs related to mail, including information about sent and received emails, as well as any failures on the network

```
File Edit View Search Terminal Tabs Help
fox@ubuntu-victim: /var/log x ubuntu-victim@ubuntu-victim: ~ x ubuntu-victim@ubuntu-victim: /var/log x
Aug 20 22:30:41 mail postfix/master[1712]: warning: unix_trigger_event: read timeout for service public/pickup
Aug 21 00:06:22 mail postfix/pickup[5135]: 5F7172804F0: uid=0 from=<root>
Aug 21 00:06:22 mail postfix/cleanup[5357]: 5F7172804F0: message-id=<20240820190622.5F7172804F0@mail.ubuntu-victim.lab>
Aug 21 00:06:22 mail postfix/qmgr[1726]: 5F7172804F0: from=<root@mail.ubuntu-victim.lab>, size=1907, nrcpt=1 (queue active)
Aug 21 00:06:22 mail postfix/local[5359]: 5F7172804F0: to=<root@mail.ubuntu-victim.lab>, orig_to=<root>, relay=local, delay=0.2, delays=0.1/0.0
9/0/0.01, dsn=2.0.0, status=sent (delivered to maildir)
Aug 21 00:06:22 mail postfix/qmgr[1726]: 5F7172804F0: removed
Aug 21 11:50:48 mail dovecot: master: Dovecot v2.2.33.2 (d6601f4ec) starting up for imap, pop3 (core dumps disabled)
Aug 21 11:50:54 mail postfix/postfix-script[1764]: starting the Postfix mail system
Aug 21 11:50:54 mail postfix/master[1768]: daemon started -- version 3.3.0, configuration /etc/postfix
Aug 21 13:58:15 mail dovecot: master: Dovecot v2.2.33.2 (d6601f4ec) starting up for imap, pop3 (core dumps disabled)
Aug 21 13:58:19 mail postfix/postfix-script[1691]: starting the Postfix mail system
Aug 21 13:58:19 mail postfix/master[1693]: daemon started -- version 3.3.0, configuration /etc/postfix
Aug 21 18:44:58 mail postfix/postfix-script[4557]: refreshing the Postfix mail system
Aug 21 18:44:58 mail postfix/master[1693]: reload -- version 3.3.0, configuration /etc/postfix
Aug 21 18:45:03 mail postfix/postfix-script[4613]: refreshing the Postfix mail system
Aug 21 18:45:03 mail postfix/master[1693]: reload -- version 3.3.0, configuration /etc/postfix
Aug 21 22:44:56 mail dovecot: master: Dovecot v2.2.33.2 (d6601f4ec) starting up for imap, pop3 (core dumps disabled)
Aug 21 22:45:07 mail postfix/postfix-script[1709]: starting the Postfix mail system
Aug 21 22:45:08 mail postfix/master[1718]: daemon started -- version 3.3.0, configuration /etc/postfix
Aug 22 01:04:41 mail postfix/pickup[3935]: 8403B2802A9: uid=0 from=<root>
Aug 22 01:04:41 mail postfix/cleanup[4189]: 8403B2802A9: message-id=<20240821200441.8403B2802A9@mail.ubuntu-victim.lab>
Aug 22 01:04:41 mail postfix/qmgr[1720]: 8403B2802A9: from=<root@mail.ubuntu-victim.lab>, size=1907, nrcpt=1 (queue active)
Aug 22 01:04:41 mail postfix/local[4193]: 8403B2802A9: to=<root@mail.ubuntu-victim.lab>, orig_to=<root>, relay=local, delay=0.19, delays=0.09/0
.1/0/0.01, dsn=2.0.0, status=sent (delivered to maildir)
Aug 22 01:04:41 mail postfix/qmgr[1720]: 8403B2802A9: removed
Aug 22 10:08:17 mail dovecot: master: Dovecot v2.2.33.2 (d6601f4ec) starting up for imap, pop3 (core dumps disabled)
Aug 22 10:08:25 mail postfix/postfix-script[1834]: starting the Postfix mail system
Aug 22 10:08:26 mail postfix/master[1838]: daemon started -- version 3.3.0, configuration /etc/postfix
ubuntu-victim@ubuntu-victim:/var/log$
```

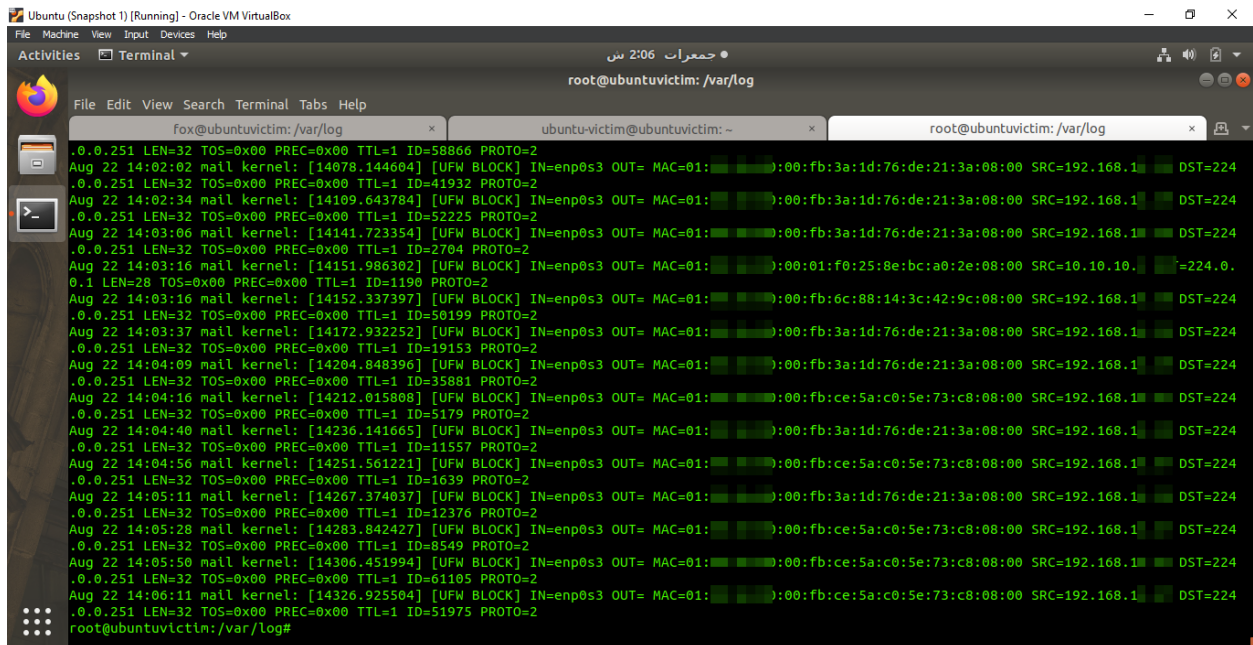
Here are the mails all of the mail logs are showing in the figure.





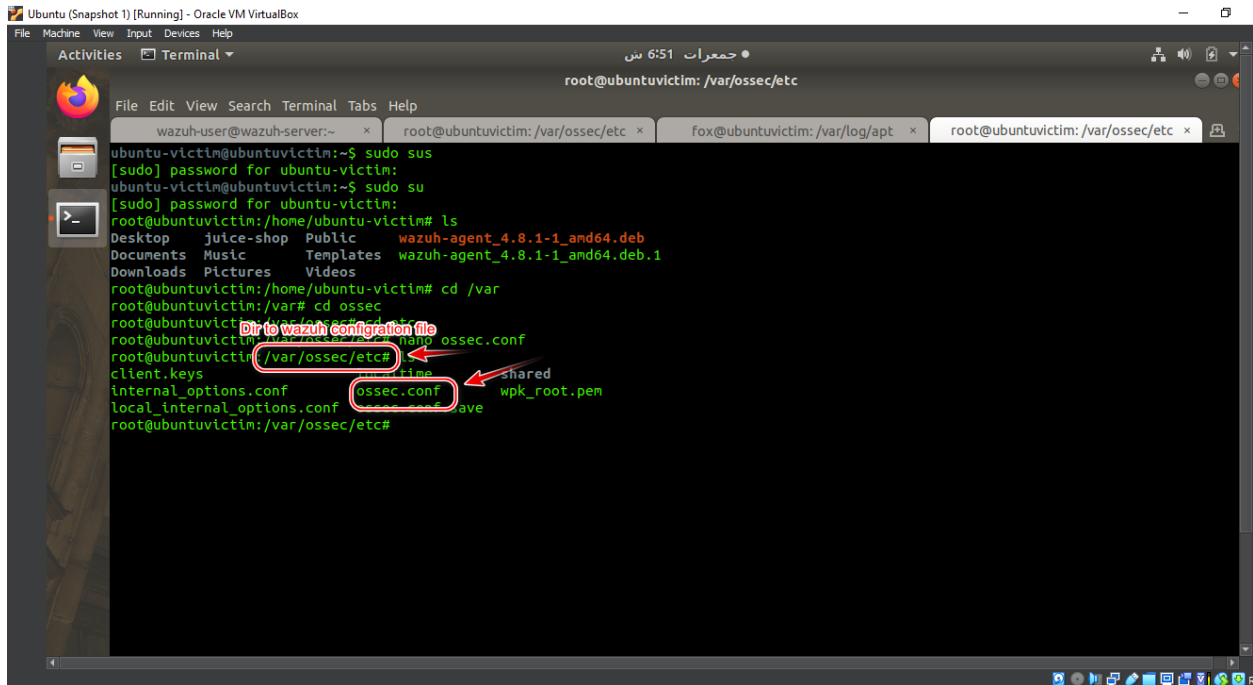


As shown in the figure, this is the Ubuntu firewall log file. It stores all the information related to UFW, such as which services are enabled or blocked.

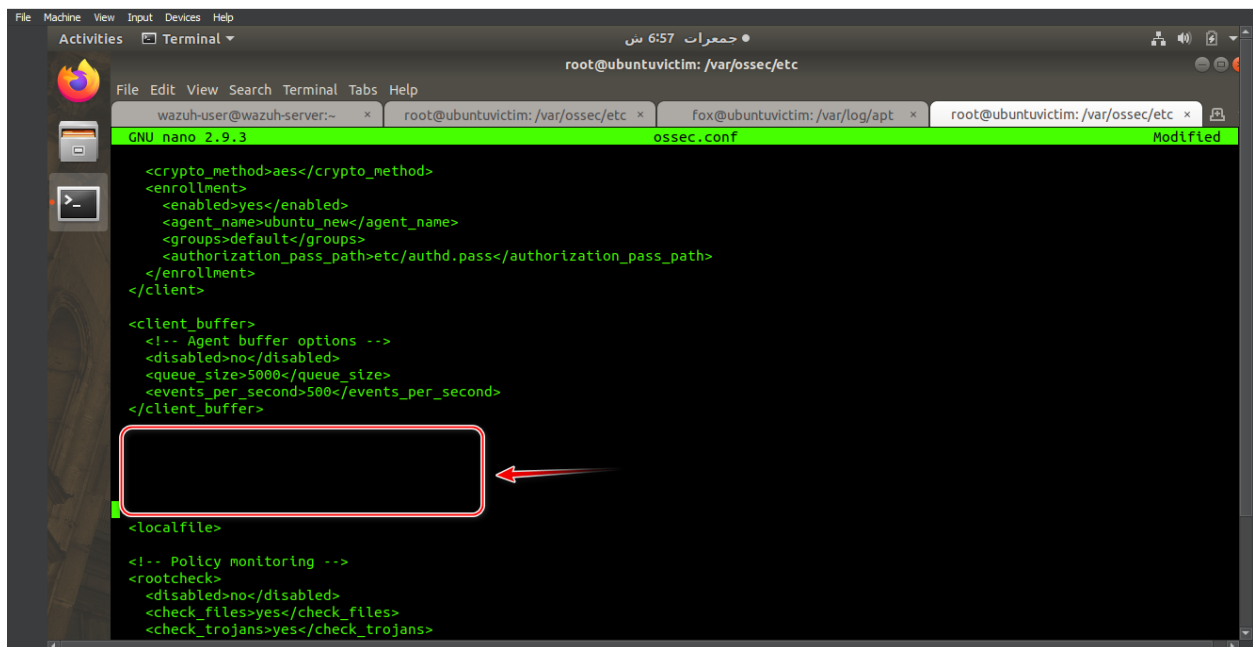


Now as we open the UFW file so here our all details we can see that some service are blocked by UFW and some are allowed according to the rules.

Now after all that let's do integration mean to integrate our ubuntu with Wazuh to monitor all the logs and traffic there on Wazuh(siem).

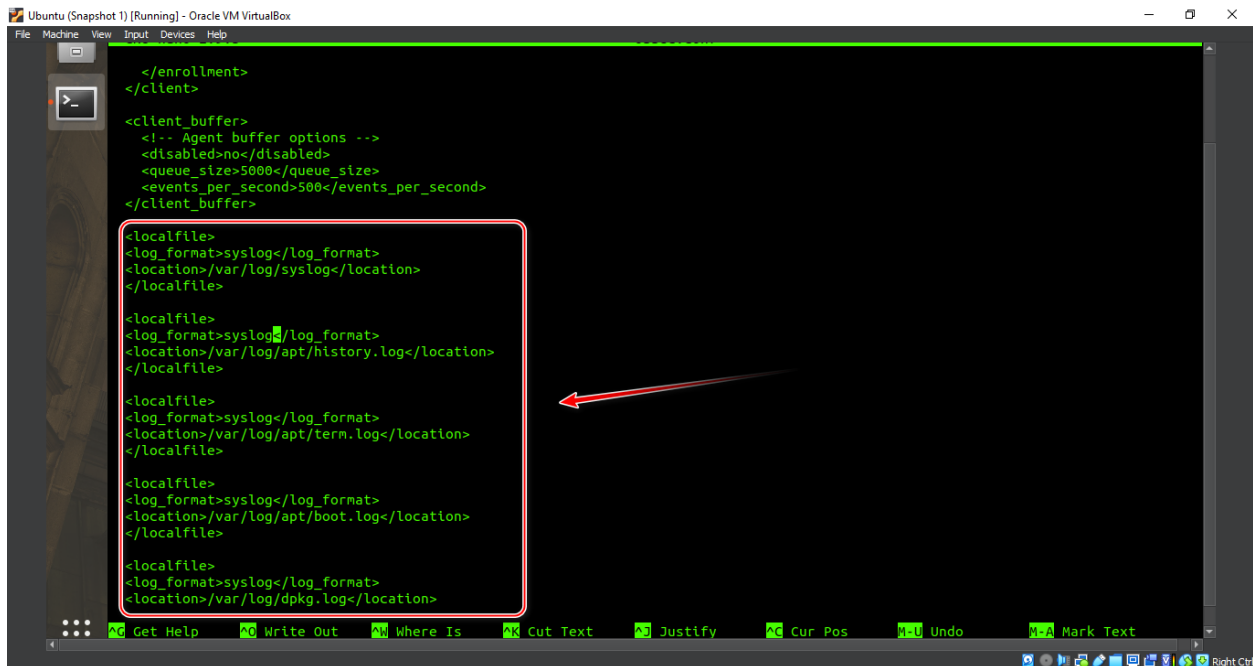


To perform this integration, first navigate to the `/var/ossec/etc` directory. There, you will find the **ossec.conf** file, which is the configuration file for Wazuh.



So, here at this underlined place here we have to write our ubuntu logs file integration details.





```
</enrollment>
</client>

<client_buffer>
  <!-- Agent buffer options -->
  <disabled>no</disabled>
  <queue_size>5000</queue_size>
  <events_per_second>500</events_per_second>
</client_buffer>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/syslog</location>
</localfile>

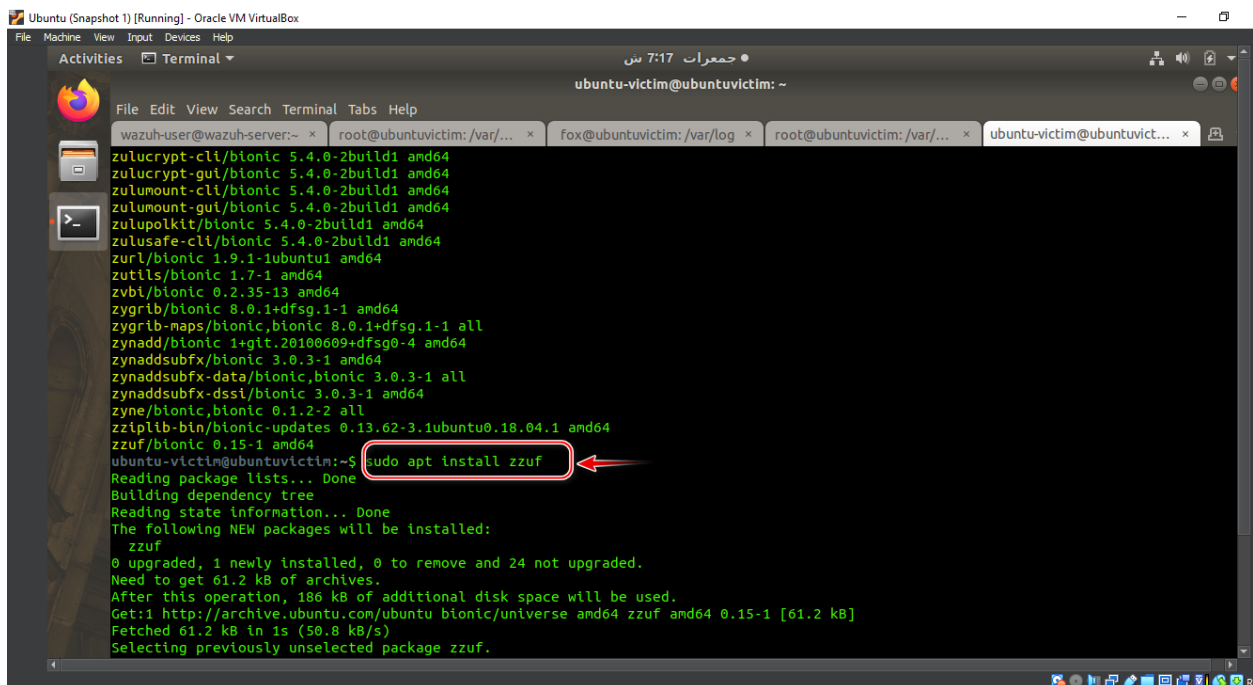
<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/apt/history.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/apt/term.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/apt/boot.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/dpkg.log</location>
</localfile>
```

I have described how to integrate Ubuntu log files into the Wazuh server for monitoring.

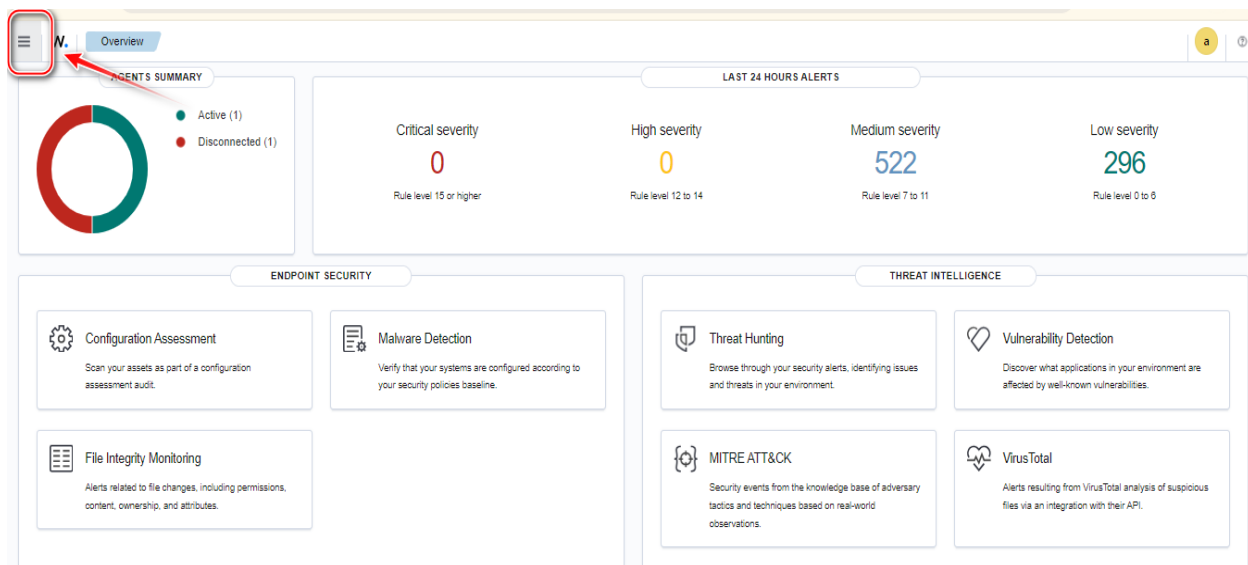


```
ubuntu-victim@ubuntu-victim: ~
File Edit View Search Terminal Tabs Help
wazuh-user@wazuh-server:~ x root@ubuntu-victim: /var/... x fox@ubuntu-victim: /var/log x root@ubuntu-victim: /var/... x ubuntu-victim@ubuntu-victim: ~

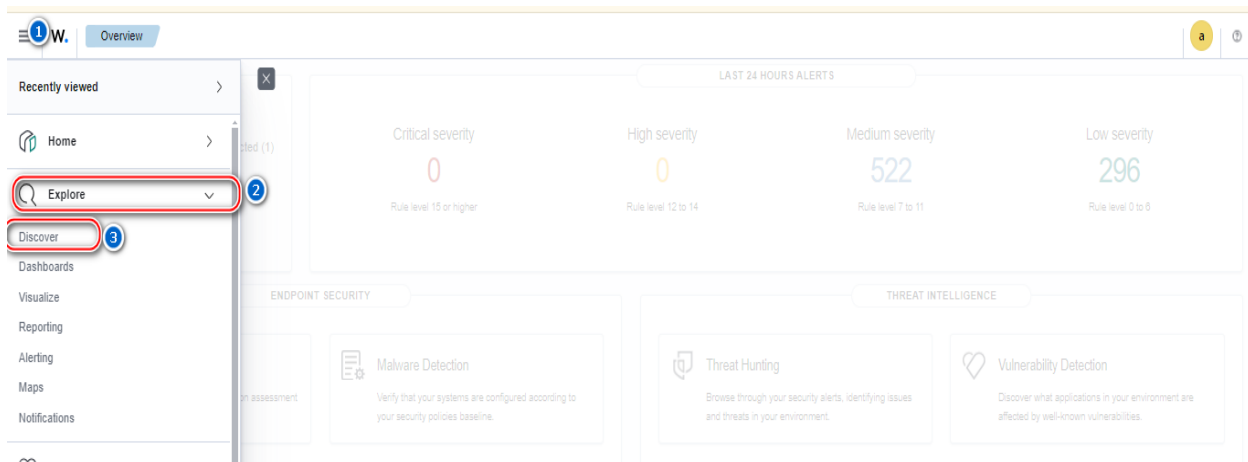
zulucript-cli/bionic 5.4.0-2build1 amd64
zulucript-gui/bionic 5.4.0-2build1 amd64
zulumount-cli/bionic 5.4.0-2build1 amd64
zulumount-gui/bionic 5.4.0-2build1 amd64
zulupolkit/bionic 5.4.0-2build1 amd64
zulunsafe-cli/bionic 5.4.0-2build1 amd64
zurl/bionic 1.9.1-1ubuntu1 amd64
zutils/bionic 1.7-1 amd64
zvbi/bionic 0.2.35-13 amd64
zygrib/bionic 8.0.1+dfsg.1-1 amd64
zygrib-maps/bionic,bionic 8.0.1+dfsg.1-1 all
zynadd/bionic 1+git.20100609+dfsg0-4 amd64
zynaddsubfx/bionic 3.0.3-1 amd64
zynaddsubfx-data/bionic,bionic 3.0.3-1 all
zynaddsubfx-dssi/bionic 3.0.3-1 amd64
zyne/bionic,bionic 0.1.2-2 all
zziplib-bin/bionic-updates 0.13.62-3.1ubuntu0.18.04.1 amd64
zzuf/bionic 0.15-1 amd64
ubuntu-victim@ubuntu-victim:~$ sudo apt install zzuf
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  zzuf
0 upgraded, 1 newly installed, 0 to remove and 24 not upgraded.
Need to get 61.2 kB of archives.
After this operation, 186 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic/universe amd64 zzuf amd64 0.15-1 [61.2 kB]
Fetched 61.2 kB in 1s (50.8 kB/s)
Selecting previously unselected package zzuf.
```

Now for checking it we have here just for checkup installed a zzuf package with apt to monitor that log on the Wazuh server.

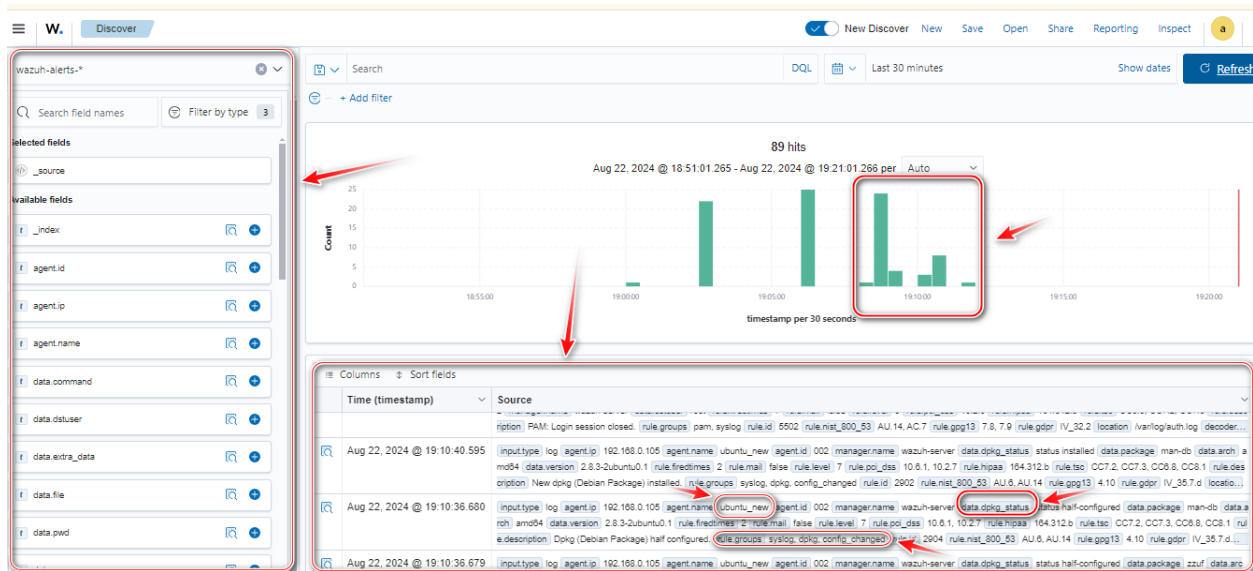
So, after running the command to monitor the logs just open your Wazuh server and make sure your agent is active there.



So here in the above figure our Wazuh dashboard is opened and agent is active so now just click on the three dots on the right-hand side.



So, after clicking the three dots, a new tab will open, as shown in the above figure. In this tab, you will see several options, including one labeled **Explore**. Within the **Explore** section, click on the **Discover** option to proceed.



So, you will notice that, as we have integrated Wazuh with Ubuntu and ran the command **sudo apt install zzuf** to install the package, all related logs are being monitored and displayed here, as shown.

```

root@ubuntuvmictim: /var/ossec/etc
File Edit View Search Terminal Tabs Help
wazuh-user@wazuh-... x root@ubuntuvmictim: ~ x fox@ubuntuvmictim: /v... x root@ubuntuvmictim: /... x root@ubuntuvmictim: /... x
GNU nano 2.9.3 ossec.conf

<location>/var/log/apt/boot.log</location>
</localfile>

<localfile>
<log_format>syslog</log_format>
<location>/var/log/dpkg.log</location>
</localfile>

<localfile>
<log_format>syslog</log_format>
<location>/var/log/ufw.log</location>
</localfile>

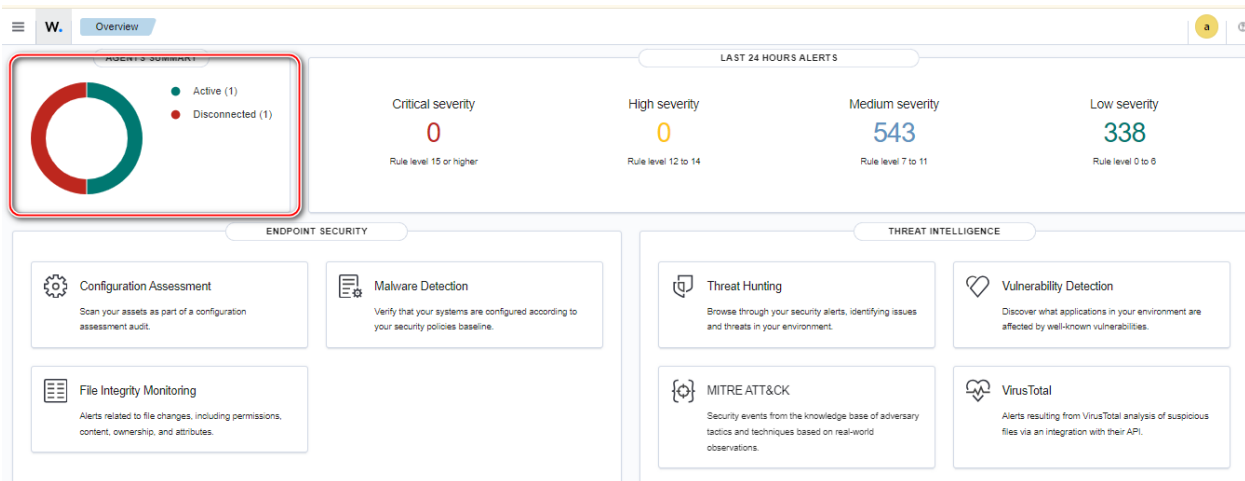
<!-- Policy Monitoring -->
<rootcheck>
<disabled>no</disabled>
<check_files>yes</check_files>

<M> Get Help <M> Write Out <M> Where Is <M> Cut Text <M> Justify <M> Cur Pos <M> Undo
<X> Exit <R> Read File <A> Replace <U> Uncut Text <T> To Spell <G> Go To Line <B> Redo
  
```

Now here as I added one another **ufw** log file in ossec.conf, so now we have to check their logs.

```
File Edit View Search Terminal Tabs Help
wazuh-user@wazuh-s... x root@ubuntuvictim: ~ x fox@ubuntuvictim: /v... x root@ubuntuvictim: /... x root@ubuntuvictim: /... x
-bash: cd: /etc/ossec/etc: No such file or directory
root@ubuntuvictim:~# cd /var/etc
-bash: cd: /var/etc: No such file or directory
root@ubuntuvictim:~# cd /var
root@ubuntuvictim:/var# ls
agentx  cache  lib  lock  mail  opt  ossec.conf  snap  tmp
backups crash local log metrics ossec run spool www
root@ubuntuvictim:/var# sudo nano ossec.conf
root@ubuntuvictim:/var# cd
root@ubuntuvictim:~# cd /var
root@ubuntuvictim:/var# ls
agentx  cache  lib  lock  mail  opt  ossec.conf  snap  tmp
backups crash local log metrics ossec run spool www
root@ubuntuvictim:/var# cd ossec/
root@ubuntuvictim:/var/ossec# ls
active-response backup etc logs ruleset var
agentless bin lib queue tmp wodles
root@ubuntuvictim:/var/ossec# cd etc
root@ubuntuvictim:/var/ossec/etc# ls
client.keys localtime shared
internal_options.conf ossec.conf wpk_root.pem
local_internal_options.conf ossec.conf.save
root@ubuntuvictim:/var/ossec/etc# nano ossec.conf
root@ubuntuvictim:/var/ossec/etc# sudo systemctl restart wazuh-agent
```

Now let's restart Wazuh agent.



So then after restarting as we can see our agent is active.

```
wazuh-user@wazuh-s... x root@ubuntuvictim: ~ x fox@ubuntuvictim: /v... x root@ubuntuvictim: /... x root@ubuntuvictim: /... x
Commands:
enable          enables the firewall
disable         disables the firewall
default ARG     set default policy
logging LEVEL   set logging to LEVEL
allow ARGS     add allow rule
deny ARGS      add deny rule
reject ARGS     add reject rule
limit ARGS     add limit rule
delete RULE|NUM delete RULE
insert NUM RULE insert RULE at NUM
route RULE     add route RULE
route delete RULE|NUM delete route RULE
route insert NUM RULE insert route RULE at NUM
reload         reload firewall
reset          reset firewall
status         show firewall status
status numbered show firewall status as numbered list of RULES
status verbose show verbose firewall status
show ARG       show firewall report
version        display version information

Application profile commands:
```

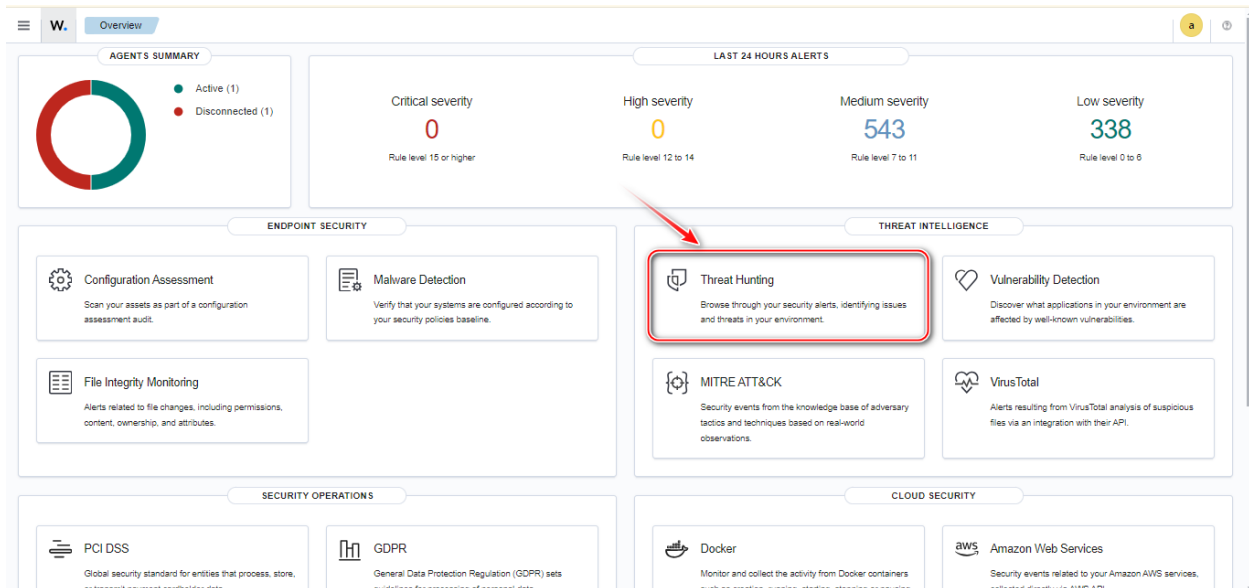
So our agent is active so let's do something to monitor their logs there on ubuntu. So all these commands are used in ufw which I opened with `sudo ufw --help`.

```
root@ubuntuvictim: /var/ossec/etc
File Edit View Search Terminal Tabs Help
wazuh-user@wazuh-s... x root@ubuntuvictim: ~ x fox@ubuntuvictim: /v... x root@ubuntuvictim: /... x root@ubuntuvictim: /... x
route insert NUM RULE insert route RULE at NUM
reload             reload firewall
reset             reset firewall
status            show firewall status
status numbered   show firewall status as numbered list of RULES
status verbose    show verbose firewall status
show ARG          show firewall report
version           display version information

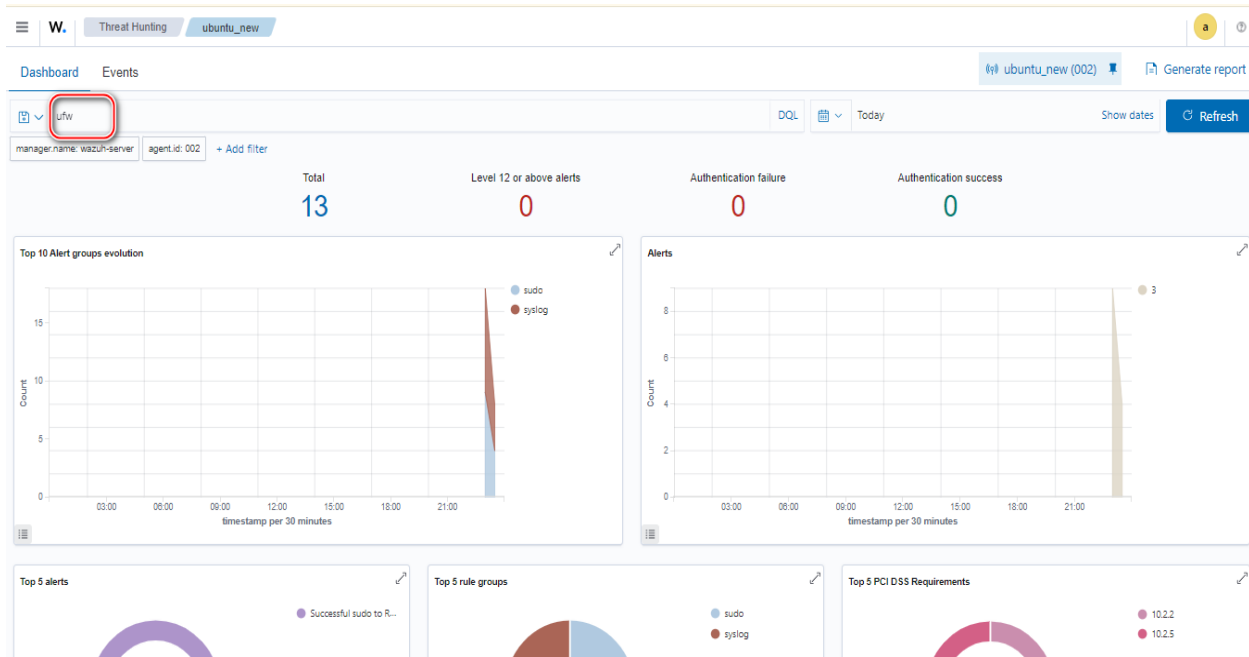
Application profile commands:
app list          list application profiles
app info PROFILE  show information on PROFILE
app update PROFILE update PROFILE
app default ARG   set default application policy

root@ubuntuvictim:/var/ossec/etc# sudo ufw disable
firewall stopped and disabled on system startup
root@ubuntuvictim:/var/ossec/etc# sudo ufw enable
firewall is active and enabled on system startup
root@ubuntuvictim:/var/ossec/etc# sudo ufw deny from 192.168.23.99
rule added
root@ubuntuvictim:/var/ossec/etc# sudo ufw allow from 191.168.0.0/24
rule added
```

Then here I used some basic commands like disabled enable status allow the specific network deny the specific ip etc. so now let's check their logs.



So for logs go to your dashboard and check your agent is active or not then click on threat hunting tab.



Then here in search bar I typed **ufw** to monitor ufw logs.

@timestamp	2024-08-22T18:35:57.511Z
_id	oeVfe5EBhnZPwExY2Bp7
agent.id	002
agent.ip	192.168.0.105
agent.name	ubuntu_new
data.command	/usr/sbin/ufw allow from 191.168.0.0/24
data.dstuser	root
data.pwd	/var/ossec/etc
data.srcuser	root
data.tty	pts/4
decoder.ftscoment	First time user executed the sudo command
decoder.name	sudo
decoder.parent	sudo
full_log	Aug 22 23:35:58 mail sudo: root : TTY=pts/4 ; PWD=/var/ossec/etc ; USER=root ; COMMAND=/usr/sbin/ufw allow from 191.168.0.0/24
id	1724351757.1332930
input.type	log
location	

Then here the logs are monitored as I allow the network.

Table	JSON	Rule
@timestamp	2024-08-22T18:34:27.365Z	
_id	meVee5EBhnZPwExYcBrq	
agent.id	002	
agent.ip	192.168.0.105	
agent.name	ubuntu_new	
data.command	/usr/sbin/ufw disable	
data.dstuser	root	
data.pwd	/var/ossec/etc	
data.srcuser	root	
data.tty	pts/4	
decoder.ftscoment	First time user executed the sudo command	

Then here is another log of disabling the ufw.

Security Alerts					
Time ↓	Technique(s)	Tactic(s)	Description	Level	Rule ID
Aug 22, 2024 @ 23:35:57.511	T1548.003	Privilege Escalation, Defense Evasion	Successful sudo to ROOT executed.	3	5402
Aug 22, 2024 @ 23:34:35.382	T1548.003	Privilege Escalation, Defense Evasion	Successful sudo to ROOT executed.	3	5402

Table	JSON	Rule
@timestamp	2024-08-22T18:34:35.382Z	
_id	nOVee5EBhnZPwExYjBpE	
agent.id	002	
agent.ip	192.168.0.105	
agent.name	ubuntu_new	
data.command	/usr/sbin/ufw enable	
data.dstuser	root	
data.pwd	/var/ossec/etc	
data.srcuser	root	
data.tty	pts/4	
decoder.ftscoment	First time user executed the sudo command	

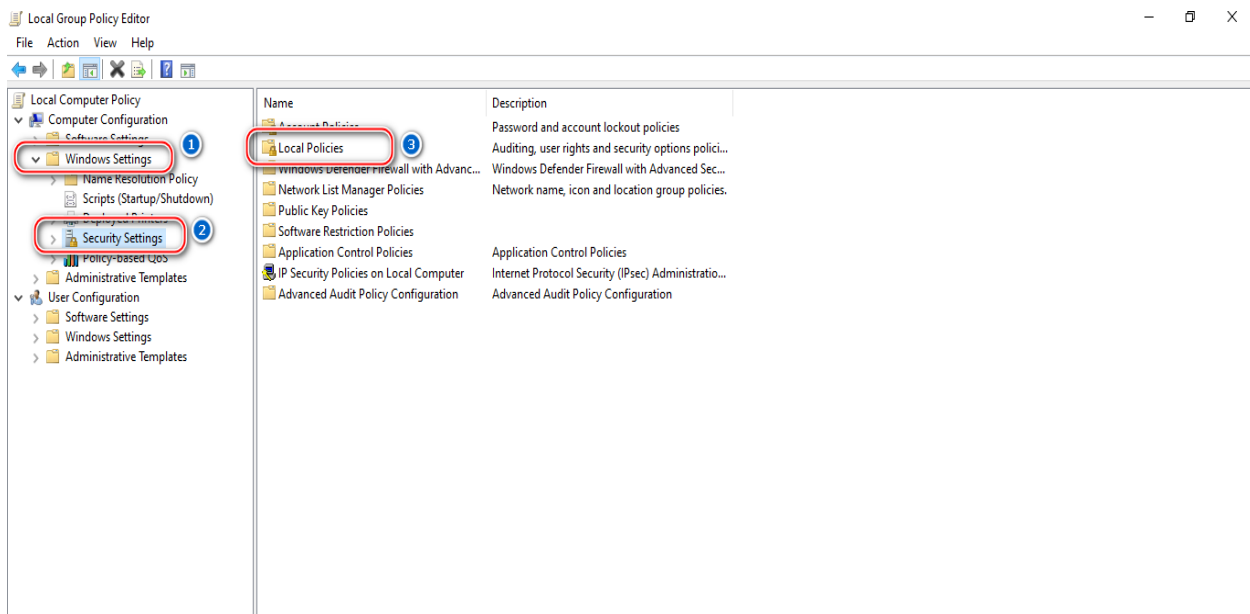
The here is another log of enabling the ufw.

So in short words we can monitor here all the logs here in wazuh as we integrated.

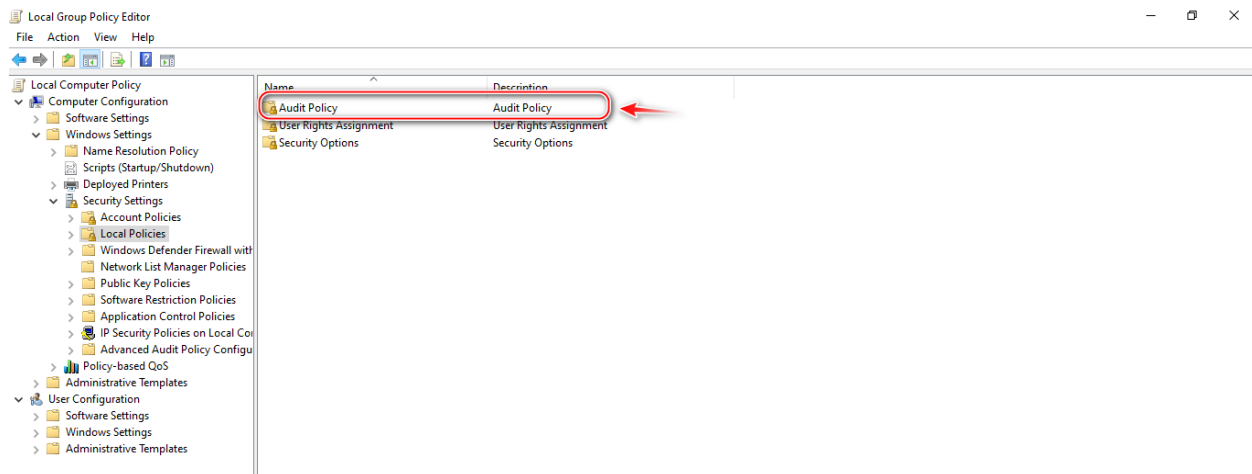


# Windows logs analysis

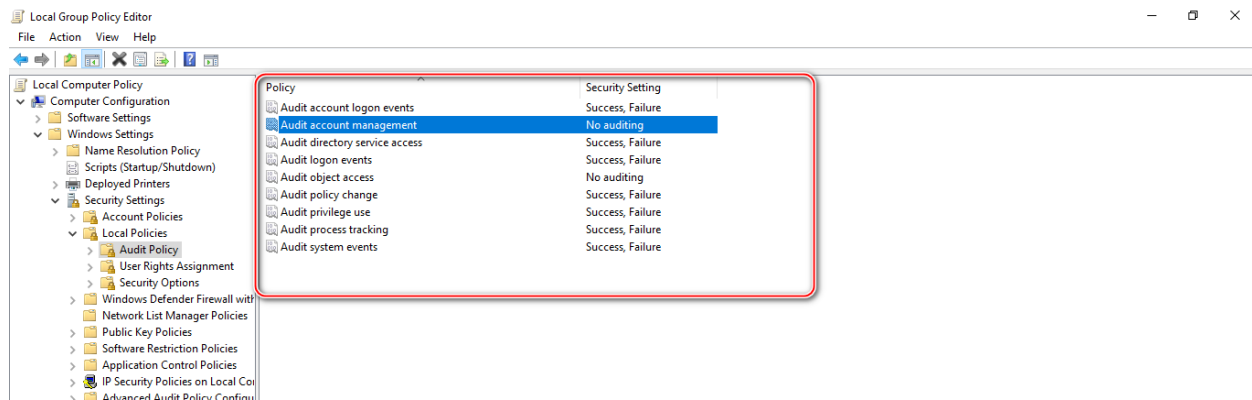
Windows log analysis involves examining various logs generated by the operating system and applications to monitor system activity, identify security events, and troubleshoot issues. Key logs include the Security, Application, and System logs found in the Event Viewer, which record events like user logins, application errors, and system warnings. By analyzing these logs, administrators can detect unauthorized access, track changes, and ensure system integrity. Tools like Wazuh can centralize and automate this analysis, making it easier to identify patterns, anomalies, and potential threats across multiple systems.



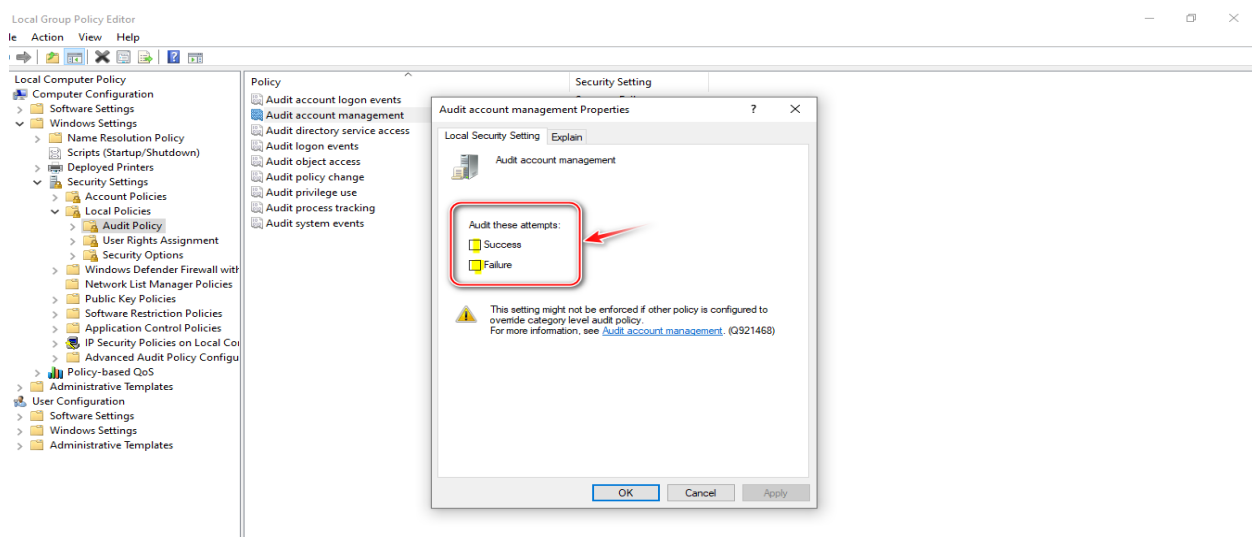
So for first of all in windows type local group policy and then in windows settings click on security settings as we want to monitor first security logs then click on local policies.



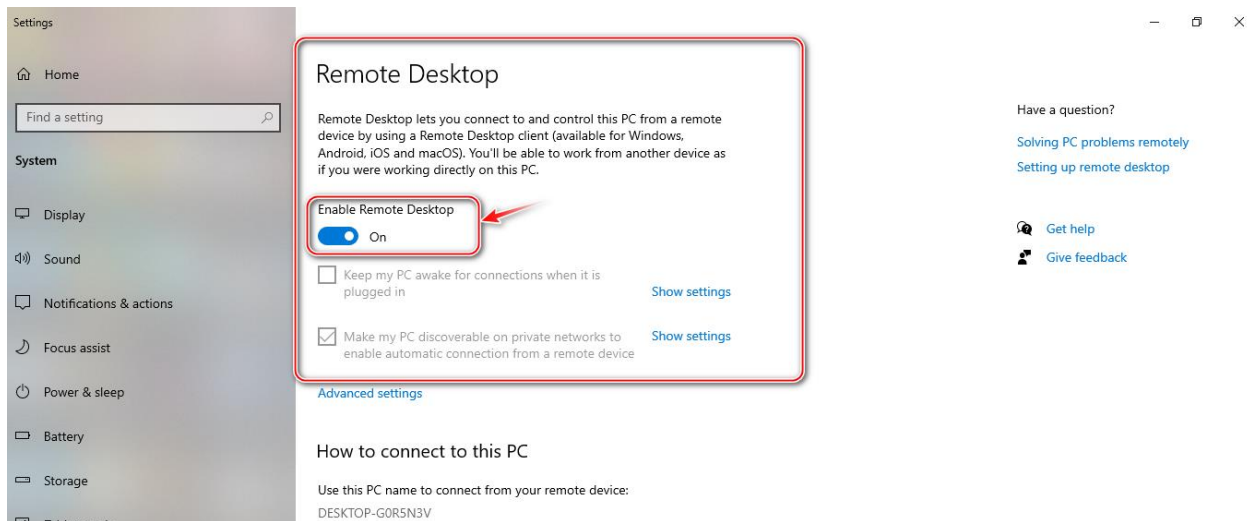
Then in local policy click on audit policy.



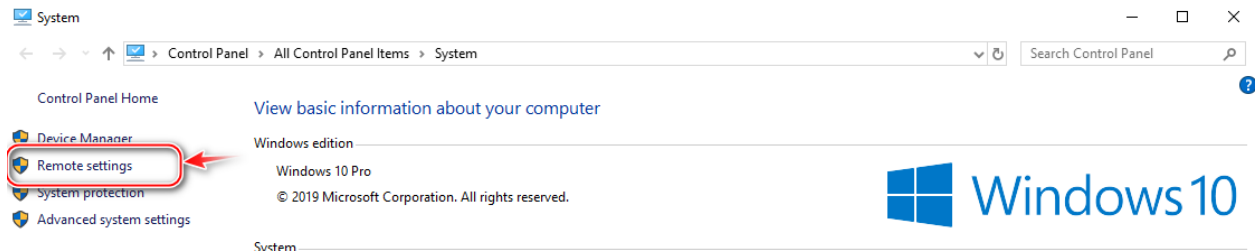
Now here enable this event policy which you want to monitor.



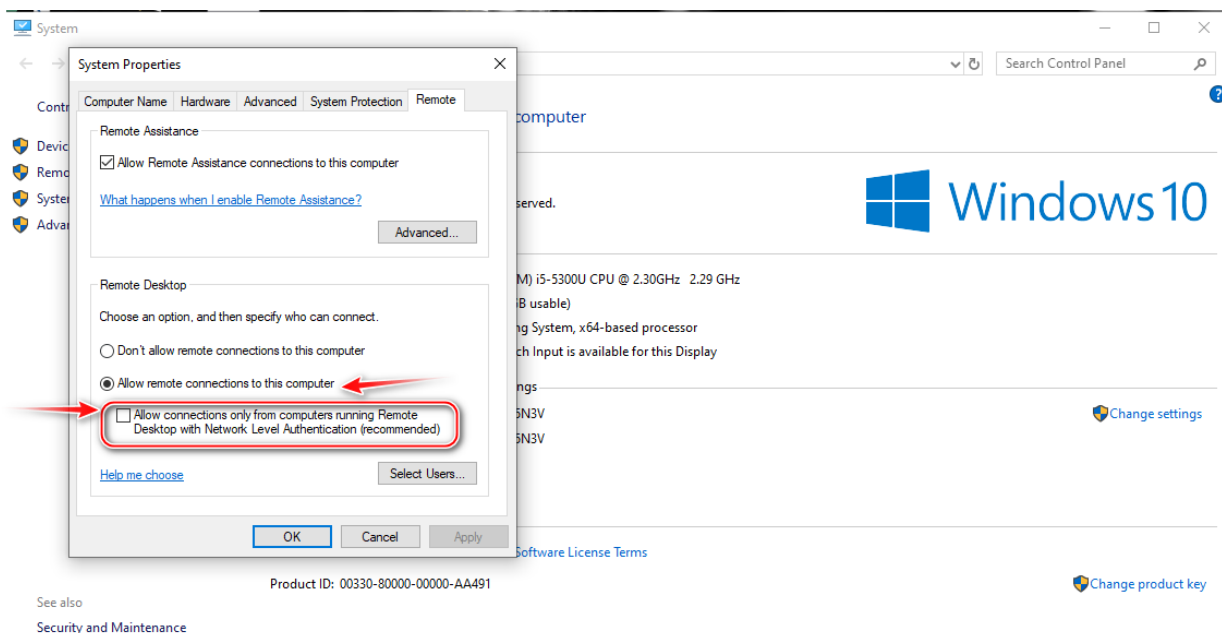
For enabling right click on the policy you want to enable and click on these dialog boxes.



Now all monitoring environment is set so we first on our remote desktop for practice demo.



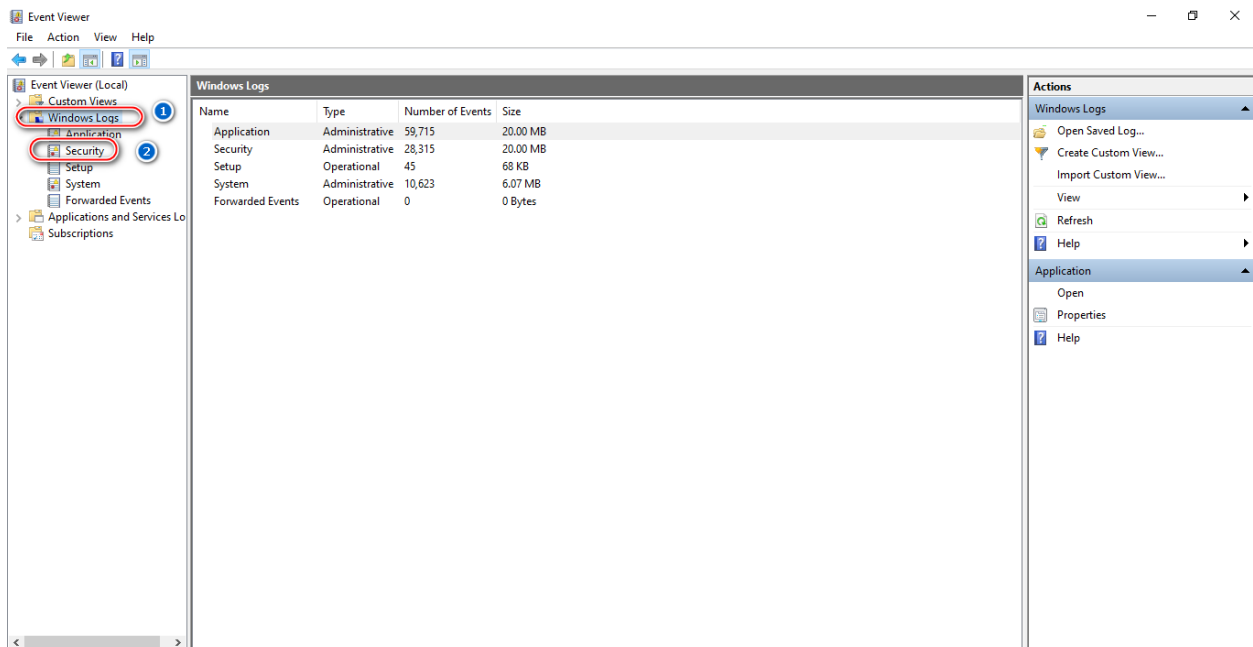
Now one more setting for enabling rdp is right click on this pc and then click properties so you will see this window interface. Then click on remote settings.



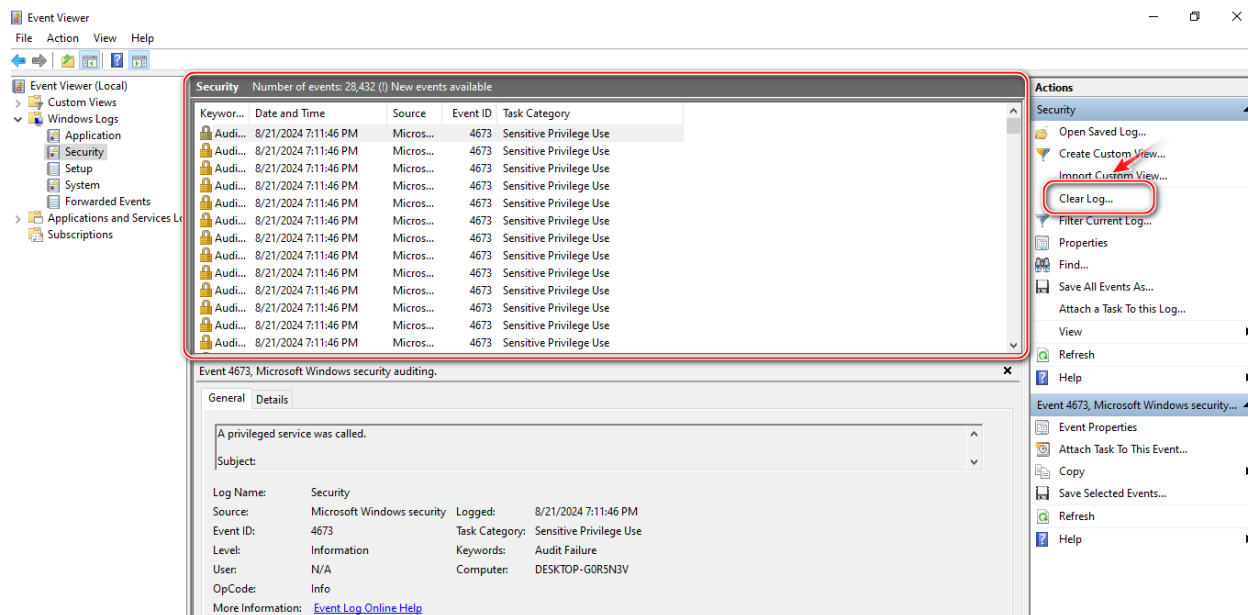
Then make sure that your setting look like same it is uncheck because this option only allow same level authentication like this is windows10 so the other one will same it is.



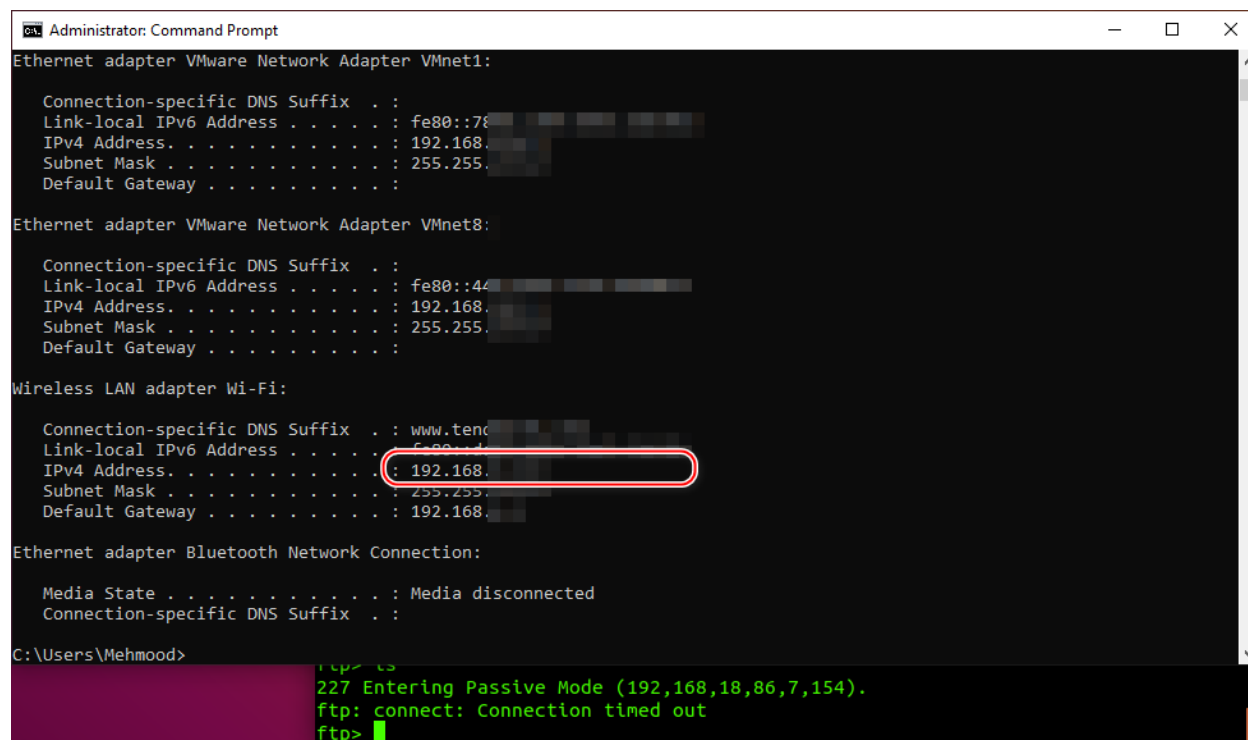
Then after that in search bar search the event viewer and open it.



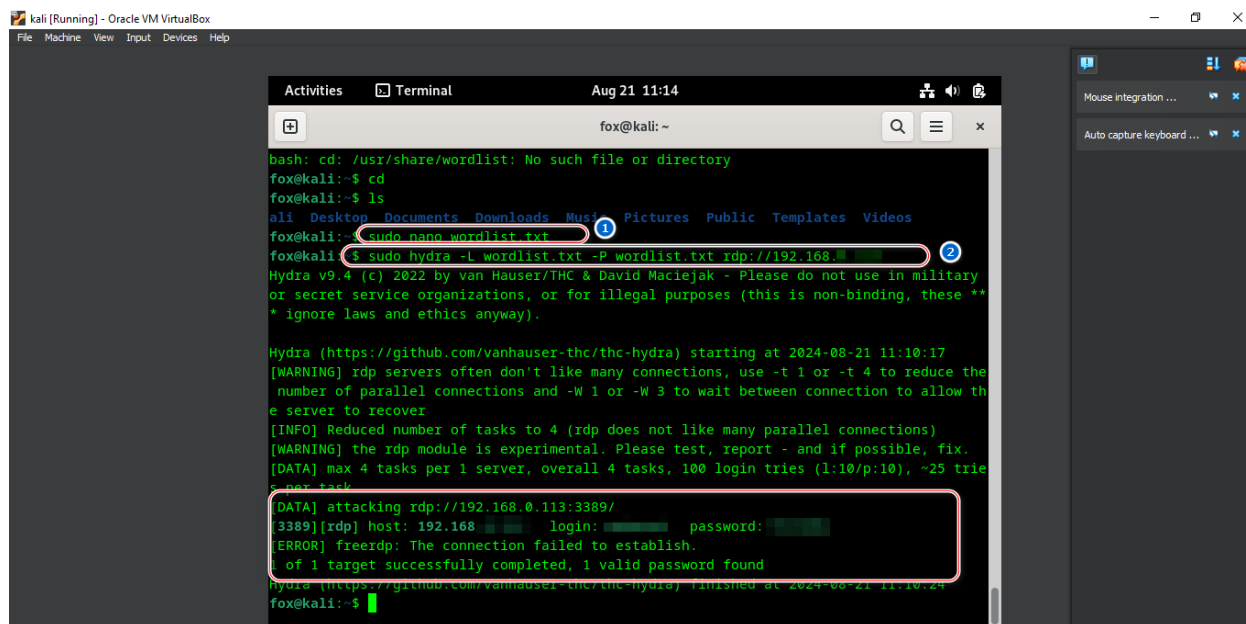
After opening that click on windows logs and then security to check the security logs.



So here as we can see there are some normal default logs of the system so let's clear the logs.

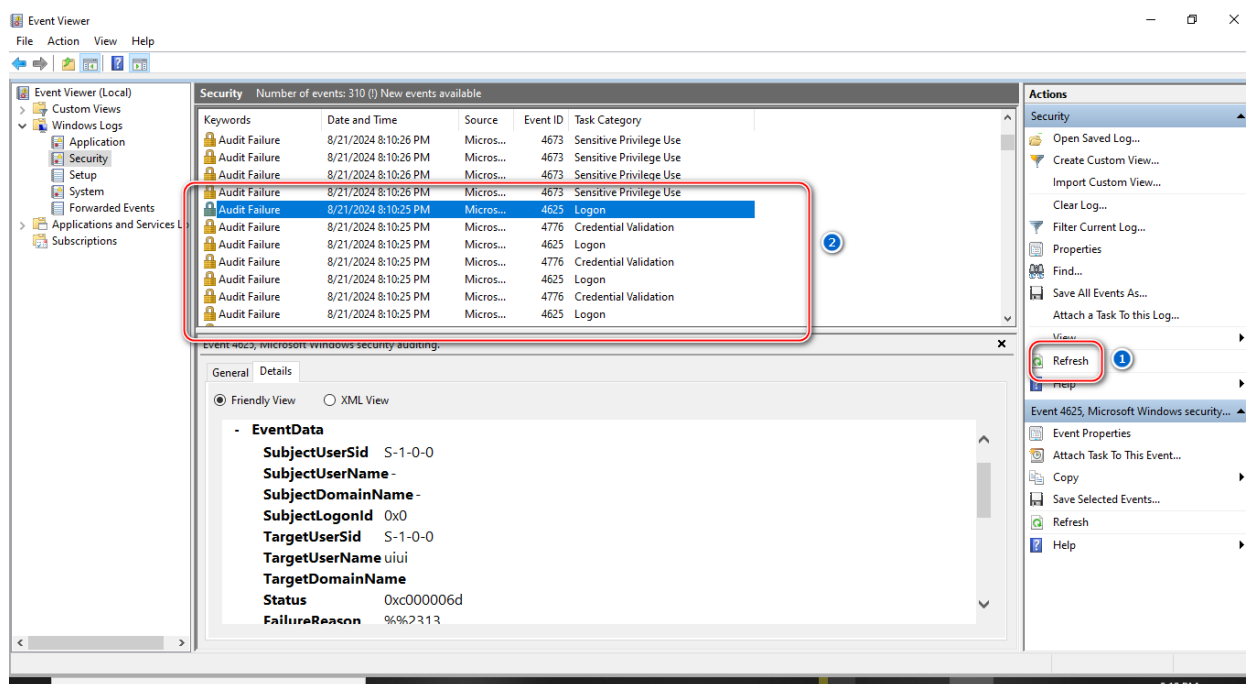


Now to monitor security logs we are just trying brute force attack so for now this window is our victim machine so check windows ip in cmd.

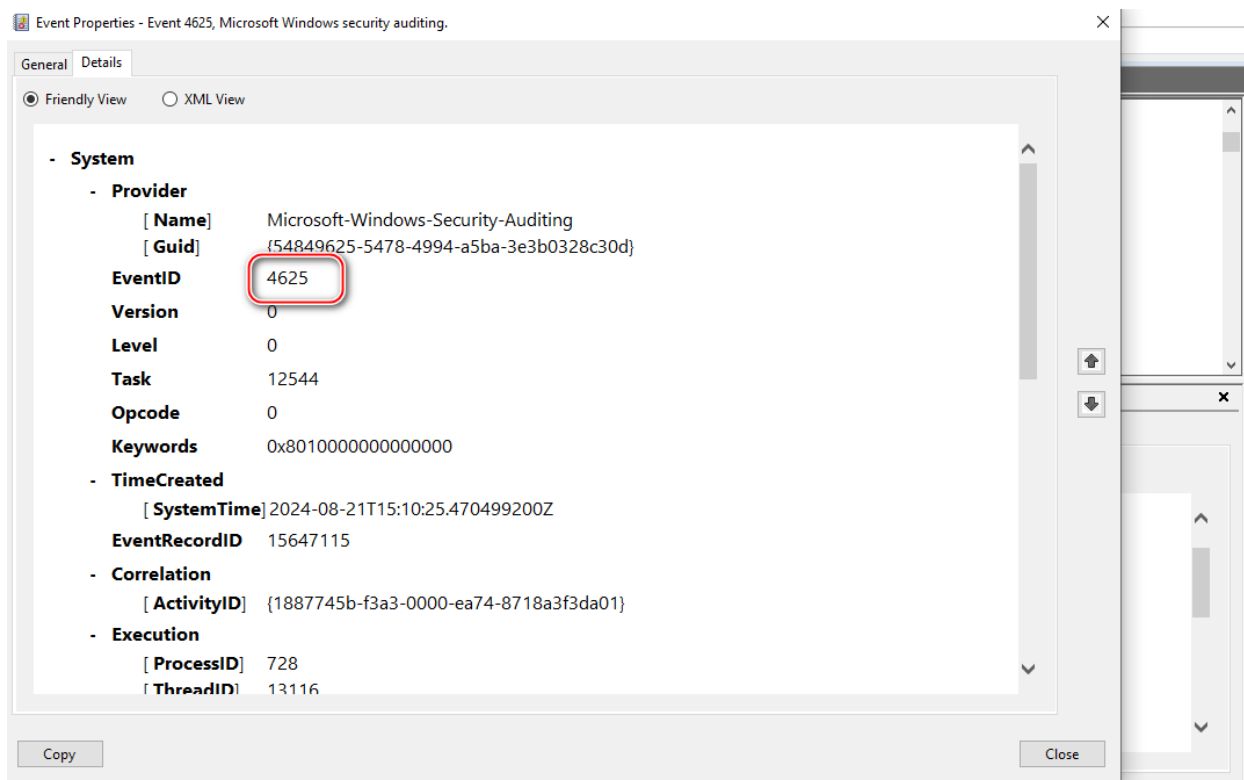


```
fox@kali: ~  
bash: cd: /usr/share/wordlist: No such file or directory  
fox@kali:~$ cd  
fox@kali:~$ ls  
ali Desktop Documents Downloads Music Pictures Public Templates Videos  
fox@kali:~$ sudo nano wordlist.txt  
fox@kali:~$ sudo hydra -L wordlist.txt -P wordlist.txt rdp://192.168.0.13389  
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military  
or secret service organizations, or for illegal purposes (this is non-binding, these **  
* ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-21 11:10:17  
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to reduce the  
number of parallel connections and -W 1 or -W 3 to wait between connection to allow the  
server to recover  
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connections)  
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.  
[DATA] max 4 tasks per 1 server, overall 4 tasks, 100 login tries (1:10/p:10), ~25 tries  
per task  
[DATA] attacking rdp://192.168.0.113:3389/  
3389][rdp] host: 192.168.0.113 login: password:  
[ERROR] freerdp: The connection failed to establish.  
of 1 target successfully completed, 1 valid password found  
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-21 11:10:24  
fox@kali:~$
```

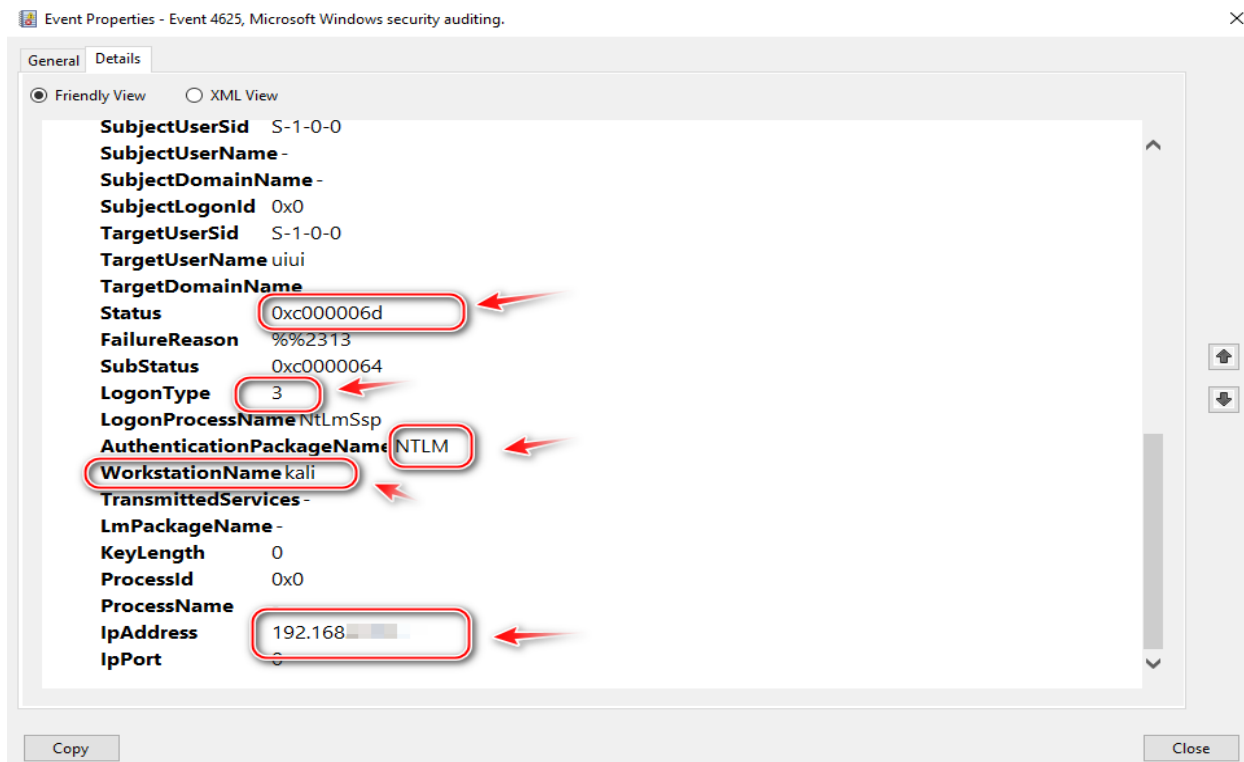
So here as we can see first, I had created a wordlist that include some usernames and passwords. And then I had tried brute force attack on my victim machine with hydra tool so in the end their username and are matched.



So as in that wordlist there are some random usernames and password so here we can see logs are monitored in event viewer so double click on it to check what info is there.



As we can see this is our event id monitored there.



And these are some attacker info and status etc is include in this event.



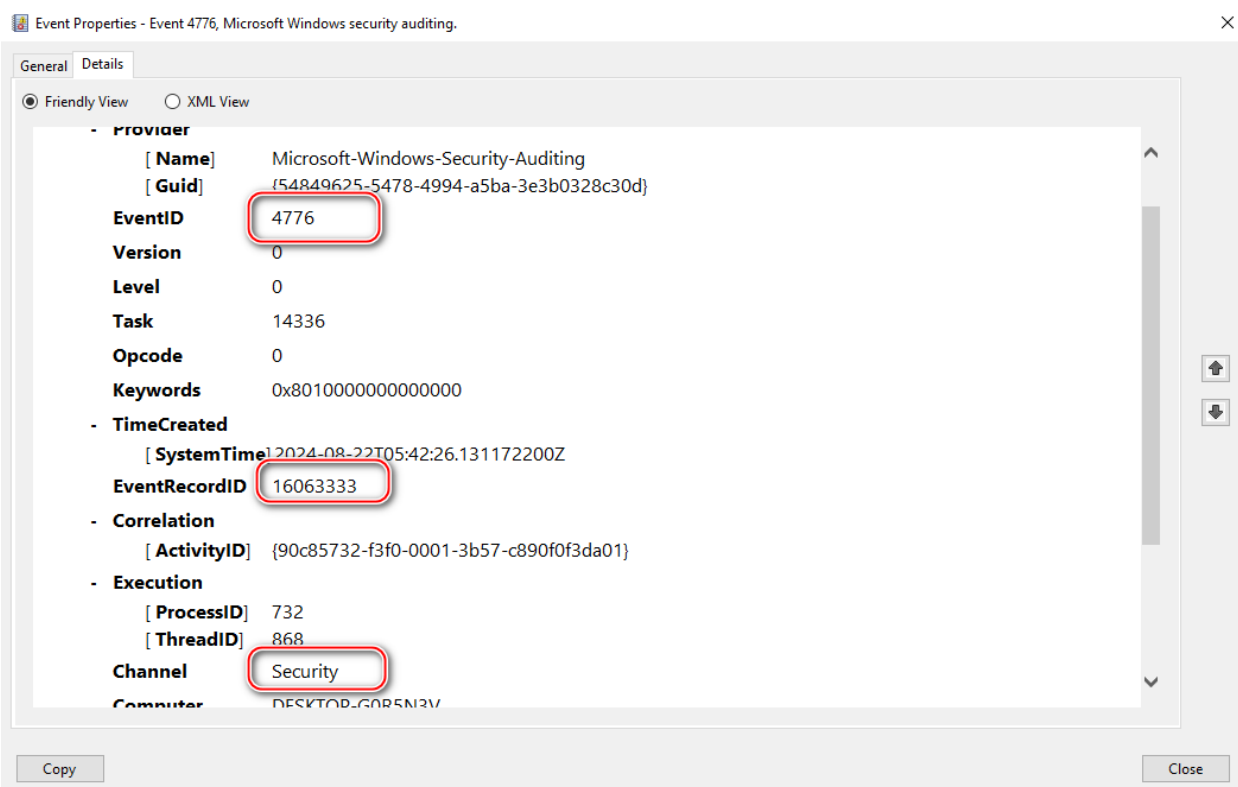
Event ID	Description
4625	<p>A failed logon attempt. Large numbers of these throughout a network may be indicative of password guessing or password spraying attacks. Again, the Network Information section of the event description can provide valuable information about a remote host attempting to log on to the system. Note that failed logons over RDP may log as Type 3 rather than Type 10, depending on the systems involved.</p> <p>You can determine more about the reason for the failure by consulting the Failure Information section of the event description.</p>

So here event id 4625 had some description that user had tried some random password as I tried wordlist and trying to guess the password and username.

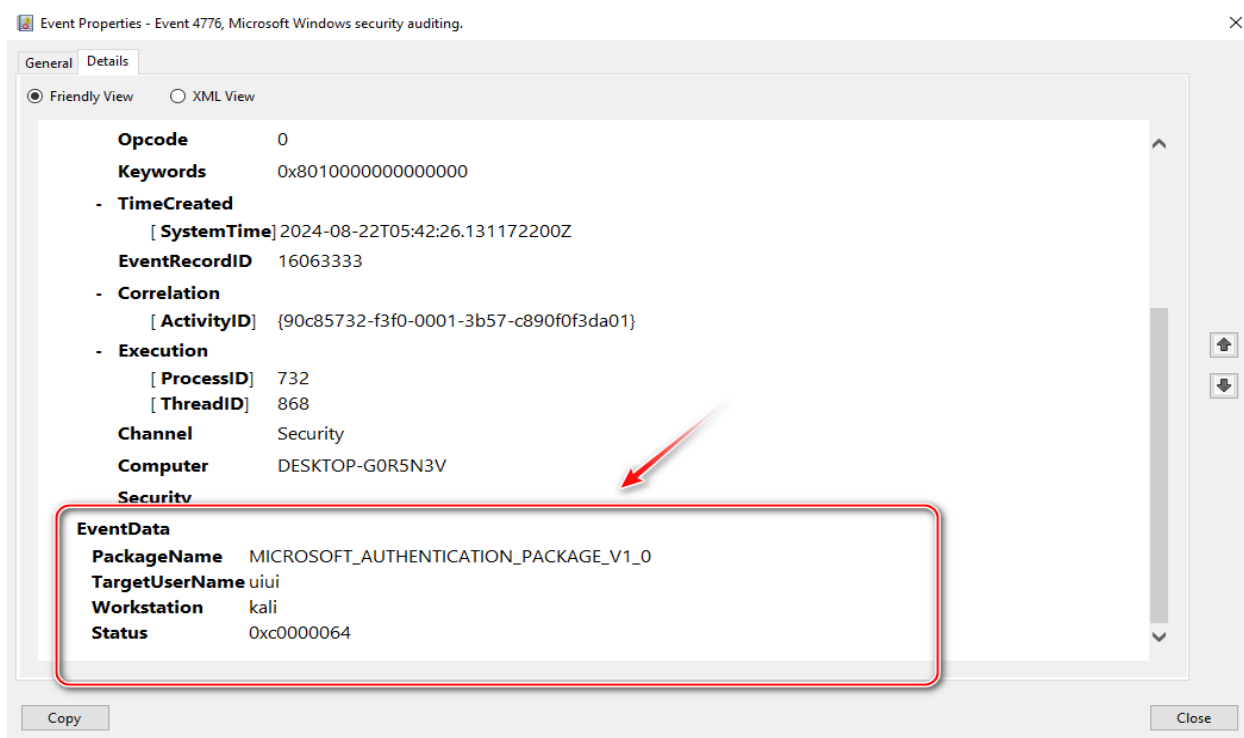
#### Common logon failure status codes

Status code	Description
0XC000005E	Currently no logon servers are available to service the logon request.
0xC0000064	User logon with misspelled or bad user account.
0xC000006A	User logon with misspelled or bad password.
0XC000006D	This is either due to a bad username or incorrect authentication information.
0XC000006E	Unknown username or bad password.
0xC000006F	User logon outside authorized hours.
0xC0000070	User logon from unauthorized workstation.
0xC0000071	User logon with expired password.
0xC0000072	User logon to account disabled by administrator.
0XC00000DC	Indicates the Server was in the wrong state to perform the desired operation.
0XC0000133	Clocks between domain controller and other computer too far out of sync.
0XC000015B	The user has not been granted the requested logon type (also known as logon right) at this machine.

The event failure status code is that mean the bad user is trying wrong passwords.



Now I have another event there occurred with event id 4776 and event record is that etc.



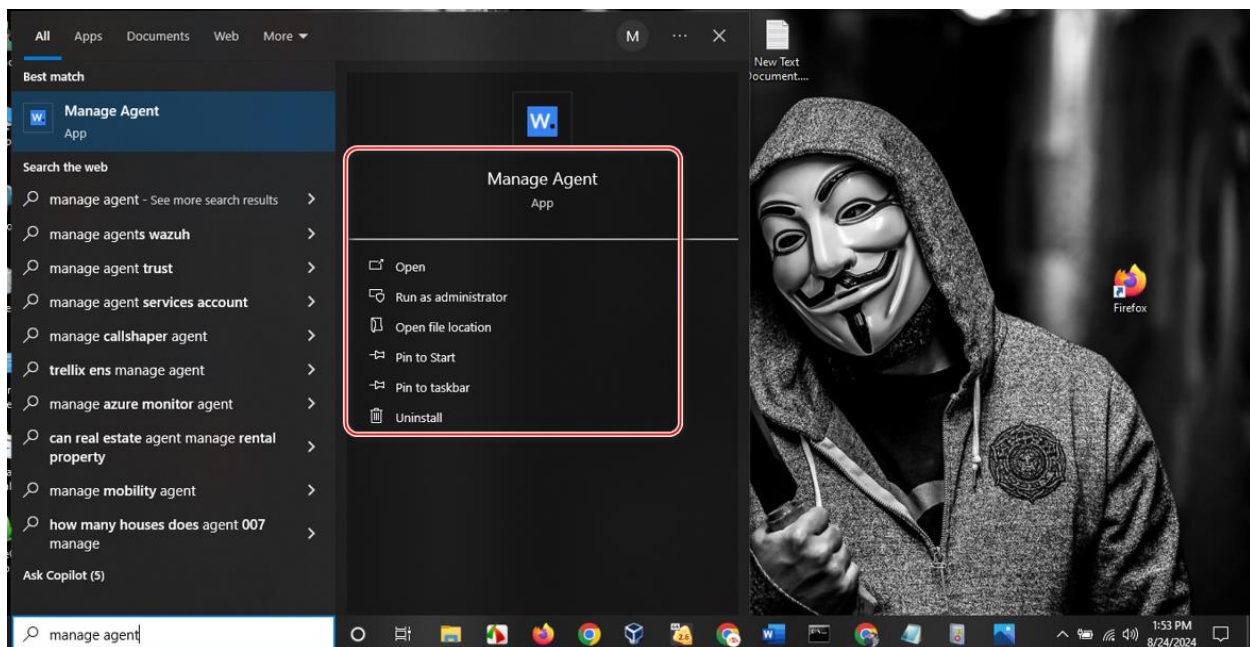
Here is the attacker information and status code.

4776	<p>This event ID is recorded for NTLM authentication attempts. The Network Information section of the event description contains additional information about the remote host in the event of a remote logon attempt. The Keywords field indicates whether the authentication attempt succeeded or failed. In the event of authentication failure, the error code in the event description provides additional details about the failure, as described in Table 8.3.</p> <p>A series of failed 4776 events with Error Code C000006A (the password is invalid) followed by an Error Code C0000234 (the account is locked out) may be indicative of a failed password guessing attack (or a user who has simply forgotten the account password). Similarly, a series of failed 4776 events followed by a successful 4776 event may show a successful password guessing attack. The presence of Event ID 4776 on a member server or client is indicative of a user attempting to authenticate to a local account on that system and may in and of itself be cause for further investigation.</p>
------	---

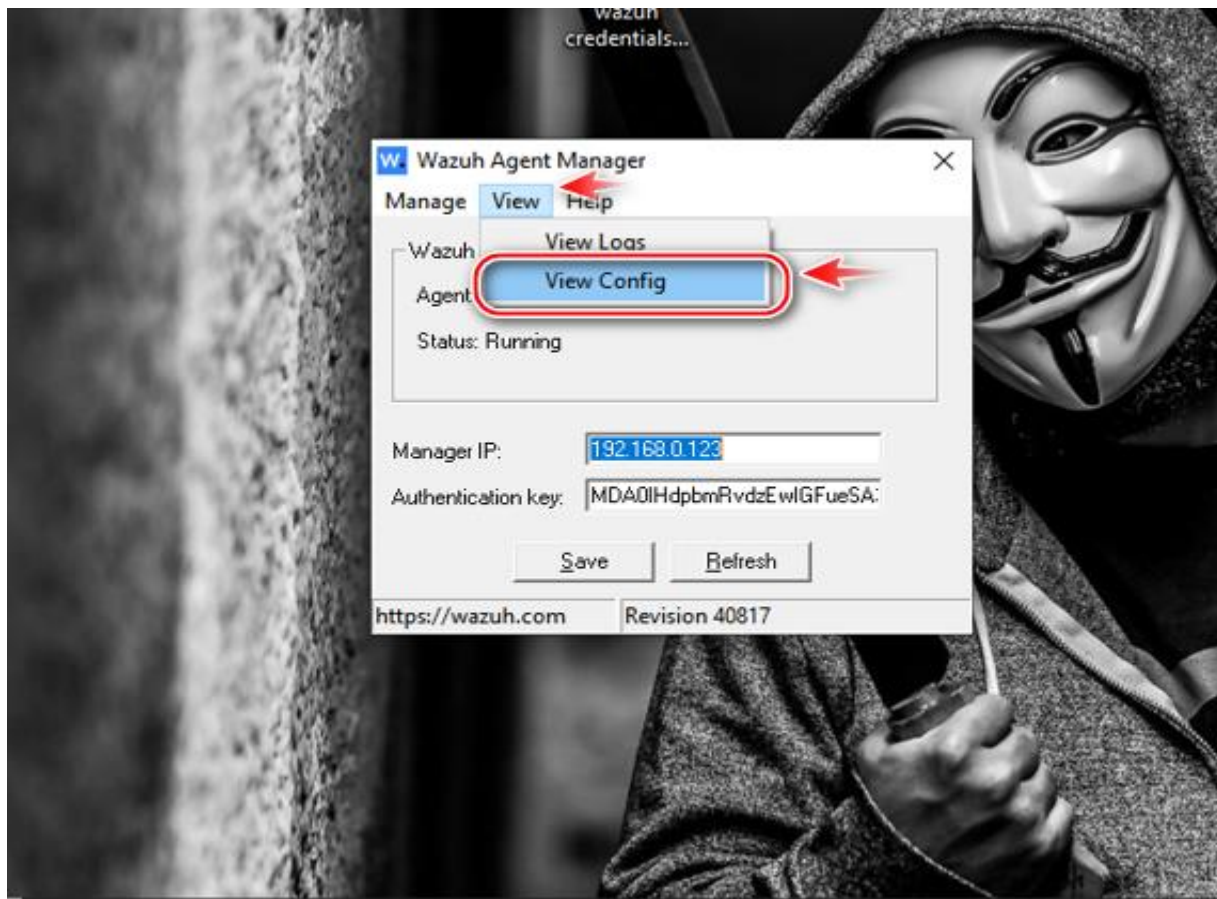
So according that information like event id the user had tried many attempts remotely. But attacker failed.

Now this is all the information about the windows log analysis in event viewer so we have to integrate these events of window to Wazuh to monitor there.

so first of all make sure that you Wazuh agent is installed in your windows machine. As I installed.



In search bar of Window you can find your Wazuh agent manager.



Then in view tab click view config file to open the Wazuh configuration file.

A screenshot of the ossec.conf file in Notepad. The configuration file content is displayed, and the <localfile> section is highlighted with a red rectangle and a red arrow. The configuration file content is as follows:

```
<!-- Active response -->
<active-response>
  <disabled>no</disabled>
  <ca_store>wpk_root.pem</ca_store>
  <ca_verification>yes</ca_verification>
</active-response>

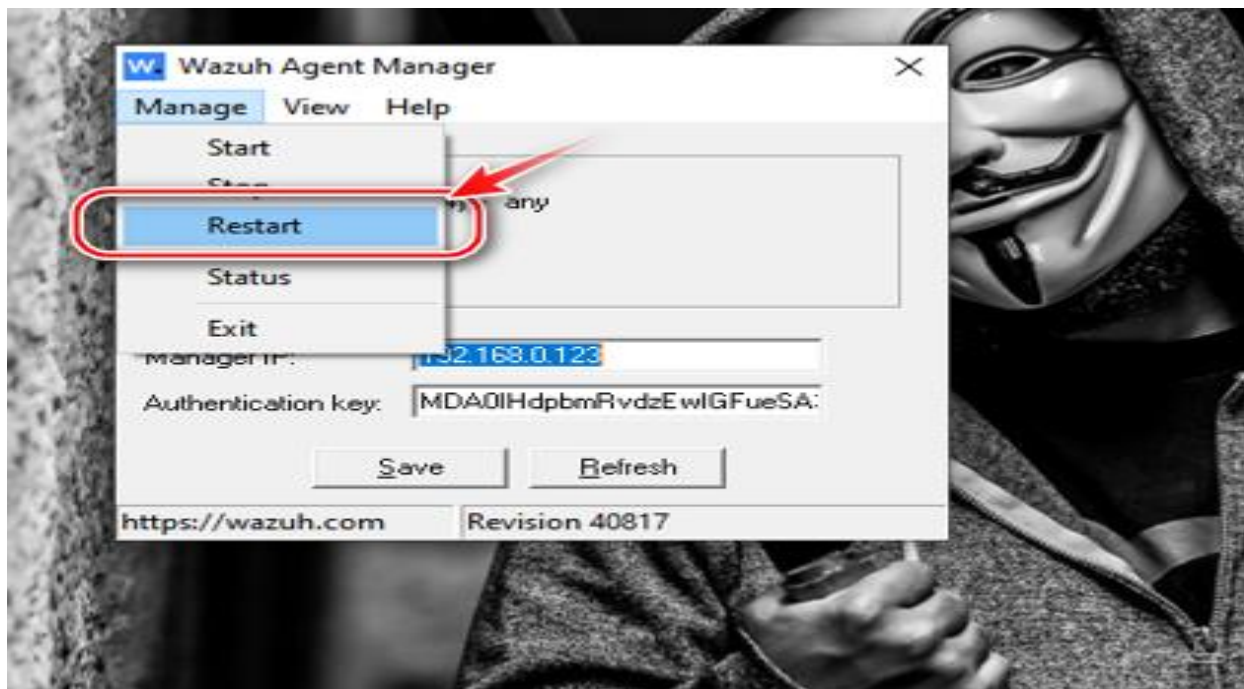
<!-- Choose between plain or json format (or both) for internal logs -->
<logging>
  <log_format>plain</log_format>
</logging>

<localfile>
  <log_format>eventchannel</log_format>
  <location>Security</location>
</localfile>

</ossec_config>

<!-- END of Default Configuration. -->
```

In configuration as I added this file to monitor the security events.

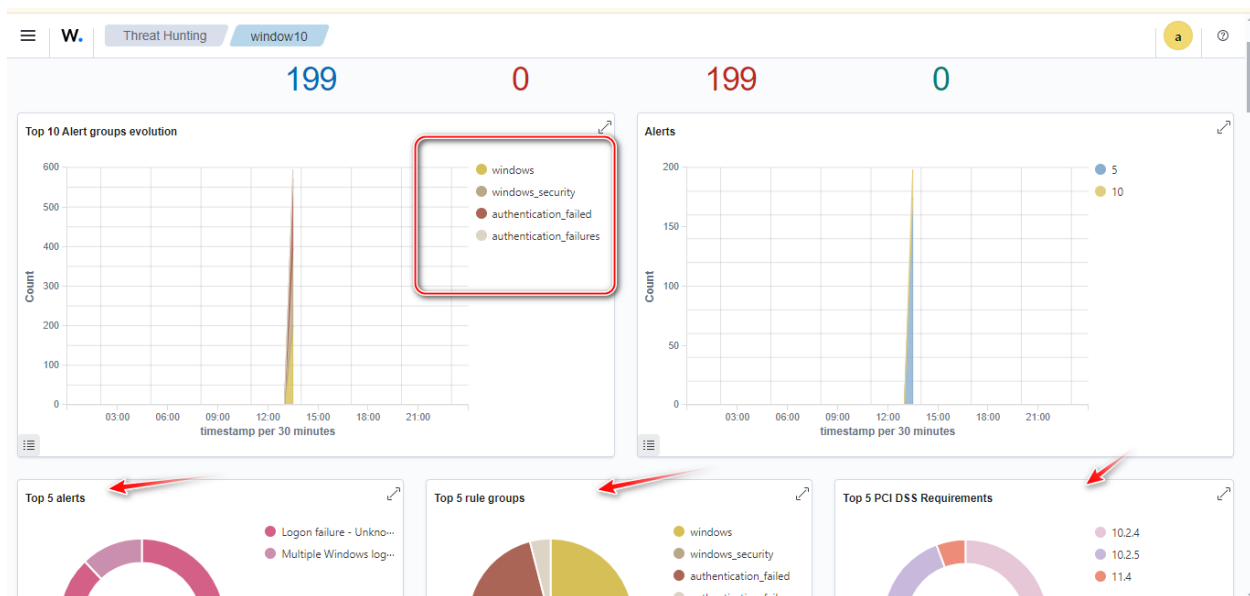


After that you will need just to restart your Wazuh agent.

```
fox@kali:~$ sudo hydra -L wordlist.txt -P wordlist.txt rdp://192.168.0.1
hydra v9.4 (c) 2022 by vanhauser/thc & david maticjak - Please do not use in military
or secret service organizations, or for illegal purposes (this is non-binding, these **
ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-24 04:44:53
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to reduce the
number of parallel connections and -W 1 or -W 3 to wait between connection to allow th
e server to recover
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connections)
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 100 login tries (1:10/p:10), ~25 trie
s per task
[DATA] attacking rdp://192.168.0.1 3389/
3389[rdp] host: 192.168.0.1 login: [REDACTED] password: kha[REDACTED]
[ERROR] freerdp: The connection failed to establish.
1 of 1 target successfully completed, 1 valid password found
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-24 04:45:00
fox@kali:~$
```

Then after all setting up, I just open my kali which is my attacker machine this time and tried the brute force attack.



Here now open your Wazuh dashboard and select your machine and go to threat hunting section. So here you can see all events.

The screenshot displays the details of a security event in the Wazuh dashboard. The event details are organized into sections:

- Subject:**
  - Security ID: S-1-0-0
  - Account Name: -
  - Account Domain: -
  - Logon ID: 0x0
- Logon Type:** 3
- Account For Which Logon Failed:**
  - Security ID: S-1-0-0
  - Account Name: uiui
  - Account Domain: -
- Failure Information:** (Highlighted with a red box)
  - Failure Reason: Unknown user name or bad password.
  - Status: 0xC000006D
  - Sub Status: 0xC0000064
- Process Information:**
  - Caller Process ID: 0x0
  - Caller Process Name: -
- Network Information:** (Highlighted with a red box)
  - Workstation Name: kali
  - Source Network Address: 192.168.0.120
  - Source Port: 0

Here are some details of the attacker machine and status codes which I explained earlier in this lecture.

data.win.eventdata.substatus	0x00000004
data.win.eventdata.subjectLogonId	0x0
data.win.eventdata.subjectUserSid	S-1-0-0
data.win.eventdata.targetUserName	uiui
data.win.eventdata.targetUserSid	S-1-0-0
data.win.eventdata.workstationName	kali
data.win.system.channel	Security
data.win.system.computer	DESKTOP-G0R5N3V
data.win.system.eventID	4625
data.win.system.eventRecordID	18717319
data.win.system.keywords	0x8010000000000000
data.win.system.level	0
data.win.system.message	"An account failed to log on

Here is how some more information of this event like event id attacker system name event type security etc.



## **Linux and Windows Log Analysis Summary**

### **Linux Logs:**

Logs in Linux systems, such as boot.log, cron.log, secure.log, mail.log, httpd.log, and messages.log, are stored in the /var/log directory and are essential for monitoring system activity and diagnosing issues. Each log serves a specific purpose: boot.log records startup events, cron.log logs scheduled tasks, secure.log tracks security-related information, mail.log covers email processing, httpd.log details web server activity, and messages.log captures general system events. To analyze these logs, utilities can be used to review their contents and integrate them with Wazuh for centralized monitoring.

### **Windows Logs:**

In Windows, logs such as Security, Application, and System logs are accessed through the Event Viewer and are crucial for monitoring system activity, detecting security events, and troubleshooting issues. Monitoring these logs helps identify unauthorized access and track system changes. Integrating Windows logs with Wazuh involves configuring the Wazuh agent to capture and send logs to a central server, where they can be analyzed for potential threats using the Wazuh dashboard.