ELK

What is elk ?

The ELK Stack (Elasticsearch, Logstash, and Kibana) is an open-source log management and data analysis platform used for collecting, processing, storing, and visualizing large volumes of data in real-time. Elasticsearch is a powerful search and analytics engine, Logstash is a data processing pipeline that ingests logs from various sources, and Kibana provides a web-based interface for visualizing and analyzing data. Often used in cybersecurity, system monitoring, and business intelligence, ELK helps organizations gain insights from logs, detect anomalies, and enhance security through real-time monitoring and alerting.

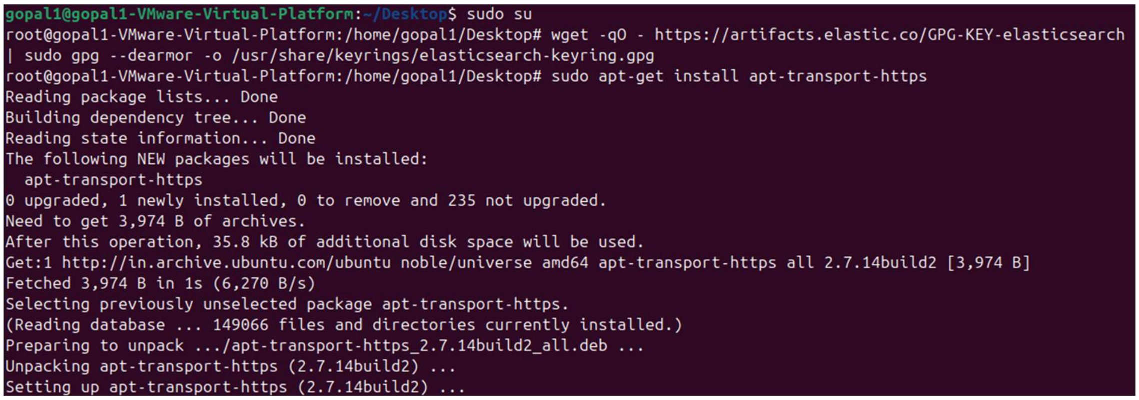
How to install elk/elk setup

For installation we have use root user account

First we have to run this command

wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg

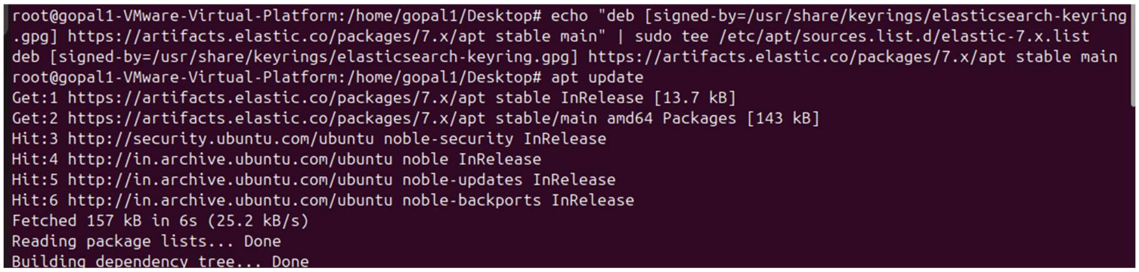
Then **sudo apt-get install apt-transport-https**



```
gopal1@gopal1-VMware-Virtual-Platform:~/Desktop$ sudo su
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch
| sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# sudo apt-get install apt-transport-https
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 235 not upgraded.
Need to get 3,974 B of archives.
After this operation, 35.8 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3,974 B]
Fetched 3,974 B in 1s (6,270 B/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 149066 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.7.14build2_all.deb ...
Unpacking apt-transport-https (2.7.14build2) ...
Setting up apt-transport-https (2.7.14build2) ...
```

After **echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list**

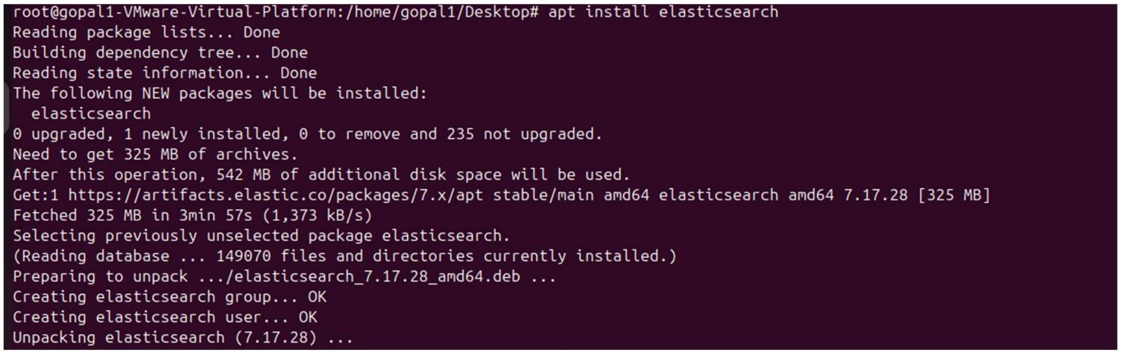
Now we have to update the system for installation we can use this command : **apt update**



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring
.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list
deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# apt update
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable InRelease [13.7 kB]
Get:2 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 Packages [143 kB]
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu noble InRelease
Hit:5 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:6 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Fetched 157 kB in 6s (25.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
```

After that **apt install elasticsearch**

After installation it looks like this



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# apt install elasticsearch
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  elasticsearch
0 upgraded, 1 newly installed, 0 to remove and 235 not upgraded.
Need to get 325 MB of archives.
After this operation, 542 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 elasticsearch amd64 7.17.28 [325 MB]
Fetched 325 MB in 3min 57s (1,373 kB/s)
Selecting previously unselected package elasticsearch.
(Reading database ... 149070 files and directories currently installed.)
Preparing to unpack .../elasticsearch_7.17.28_amd64.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (7.17.28) ...
```

We have configure elasticsearch file

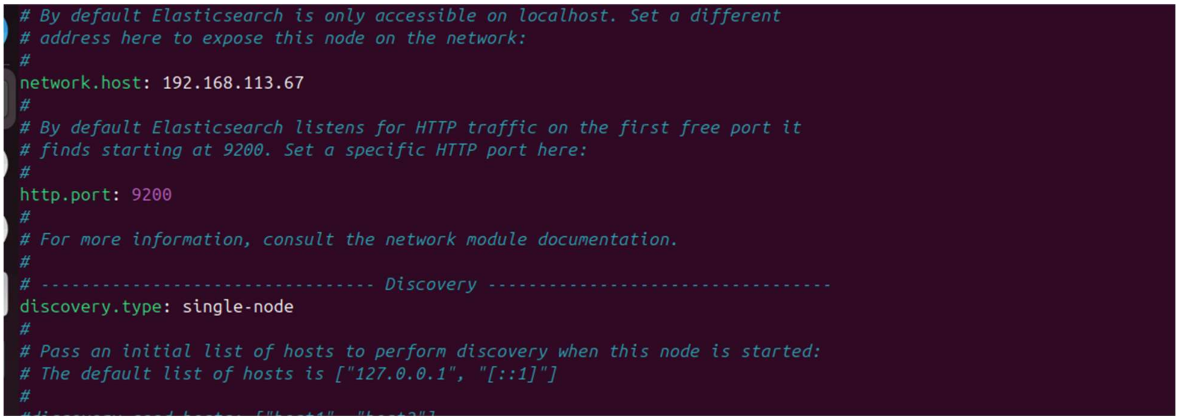
nano /etc/elasticsearch/elasticsearch.yml

We have to make changes in file as shown in image (In my case my ip address is 192.168.113.67)

Network.host: 192.168.113.67

http.port: 9200

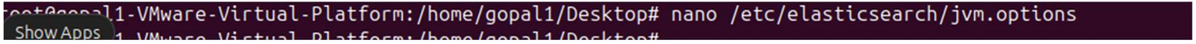
discovery.type: single-node



```
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: 192.168.113.67
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# --------------------------------- Discovery ----------------------------------
discovery.type: single-node
#
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.seed_hosts: ["host1", "host2"]
```

Then another changes in jvm.options

nano /etc/elasticsearch/jvm.options



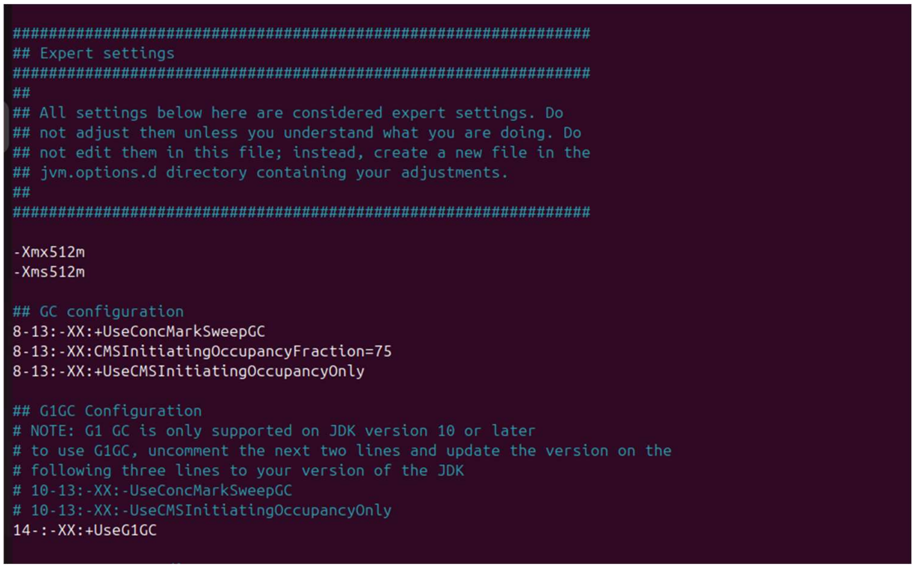
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# nano /etc/elasticsearch/jvm.options
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop#
```

We have to add this in file

-Xmx512m

-Xms512m

After that **ctrl + x** for save **y** and click **Enter** button

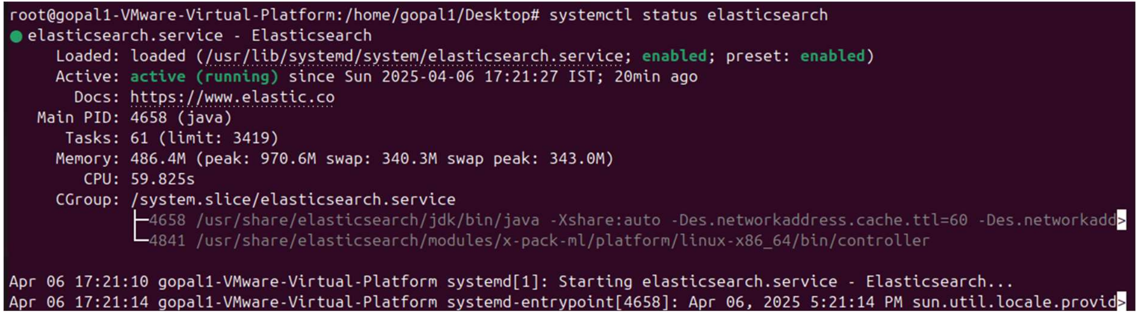


```
################################################################
## Expert settings
################################################################
##
## All settings below here are considered expert settings. Do
## not adjust them unless you understand what you are doing. Do
## not edit them in this file; instead, create a new file in the
## jvm.options.d directory containing your adjustments.
##
################################################################

-Xmx512m
-Xms512m

## GC configuration
8-13:-XX:+UseConcMarkSweepGC
8-13:-XX:CMSInitiatingOccupancyFraction=75
8-13:-XX:+UseCMSInitiatingOccupancyOnly

## G1GC Configuration
# NOTE: G1 GC is only supported on JDK version 10 or later
# to use G1GC, uncomment the next two lines and update the version on the
# following three lines to your version of the JDK
# 10-13:-XX:-UseConcMarkSweepGC
# 10-13:-XX:-UseCMSInitiatingOccupancyOnly
14-:-XX:+UseG1GC
```

Then we have to restart elasticsearch services and check services are active for this we can use this commands :

systemctl restart elasticsearch

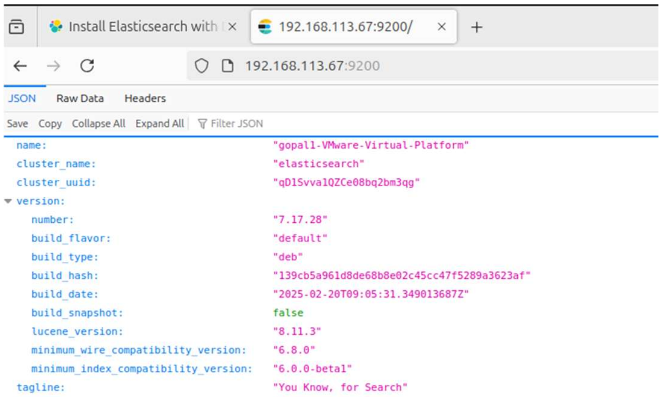
systemctl status elasticsearch



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
     Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-04-06 17:21:27 IST; 20min ago
       Docs: https://www.elastic.co
   Main PID: 4658 (java)
      Tasks: 61 (limit: 3419)
     Memory: 486.4M (peak: 970.6M swap: 340.3M swap peak: 343.0M)
        CPU: 59.825s
     CGroup: /system.slice/elasticsearch.service
             ├─4658 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkaddress.cache.ttl=60 -Des.networkadd>
             └─4841 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/controller

Apr 06 17:21:10 gopal1-VMware-Virtual-Platform systemd[1]: Starting elasticsearch.service - Elasticsearch...
Apr 06 17:21:14 gopal1-VMware-Virtual-Platform systemd-entrypoint[4658]: Apr 06, 2025 5:21:14 PM sun.util.locale.provid>
```

After this we can check on webbrowser about our elasticsearch setup by adding our ip address and portnumber

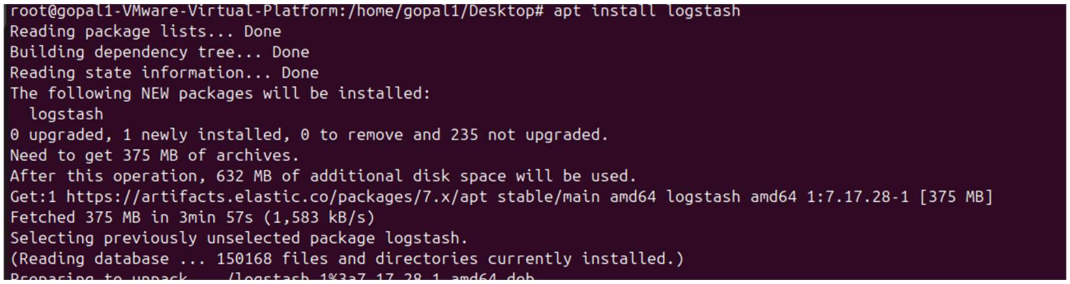
The interface shows like this



```
name:                                    "gopal1-VMware-Virtual-Platform"
cluster_name:                            "elasticsearch"
cluster_uuid:                            "qD1Svva1QZCe08bq2bm3qg"
▾ version:
    number:                              "7.17.28"
    build_flavor:                        "default"
    build_type:                          "deb"
    build_hash:                          "139cb5a961d8de68b8e02c45cc47f5289a3623af"
    build_date:                          "2025-02-20T09:05:31.349013687Z"
    build_snapshot:                      false
    lucene_version:                      "8.11.3"
    minimum_wire_compatibility_version:  "6.8.0"
    minimum_index_compatibility_version: "6.0.0-beta1"
tagline:                                 "You Know, for Search"
```

After this we have to install logstash we can use this command

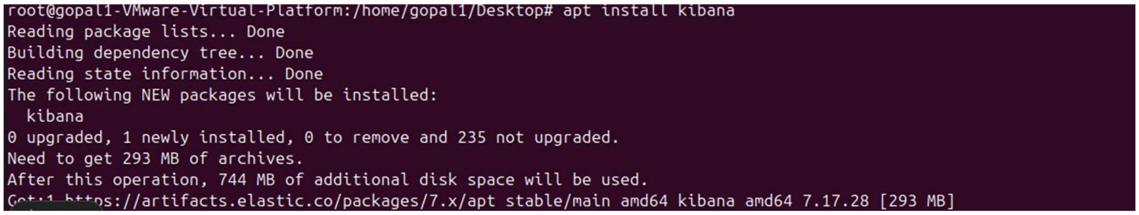
apt install logstash



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# apt install logstash
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  logstash
0 upgraded, 1 newly installed, 0 to remove and 235 not upgraded.
Need to get 375 MB of archives.
After this operation, 632 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 logstash amd64 1:7.17.28-1 [375 MB]
Fetched 375 MB in 3min 57s (1,583 kB/s)
Selecting previously unselected package logstash.
(Reading database ... 150168 files and directories currently installed.)
Preparing to unpack .../logstash_1%3a7.17.28-1_amd64.deb
```

Also we have to install kibana we can use this command

apt install kibana



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# apt install kibana
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  kibana
0 upgraded, 1 newly installed, 0 to remove and 235 not upgraded.
Need to get 293 MB of archives.
After this operation, 744 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 kibana amd64 7.17.28 [293 MB]
```

In next step we have to configure kibana.yml file for this we can use this command :

nano /etc/kibana/kibana.yml

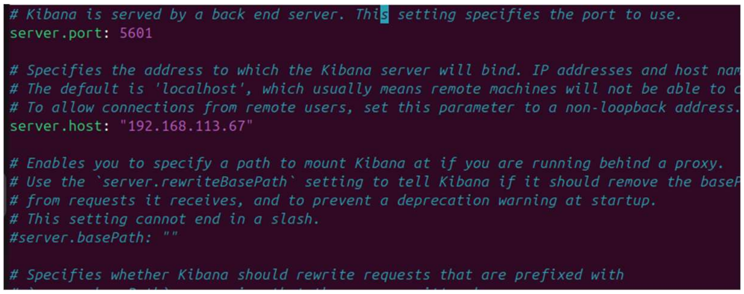
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# nano /etc/kibana/kibana.yml
```

In this file we have make changes shown in image (In my case my ip address is 192.168.155.12)

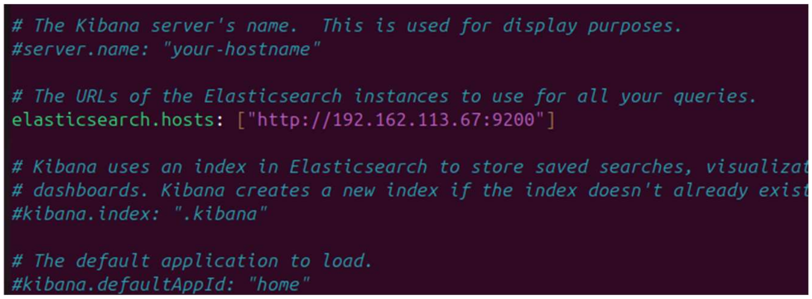
server.port: 5601

server.host: “192.168.155.12”

elasticsearch.hosts:[“http://192.168.155.12:9200”]



```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# Specifies the address to which the Kibana server will bind. IP addresses and host nam
# The default is 'localhost', which usually means remote machines will not be able to c
# To allow connections from remote users, set this parameter to a non-loopback address.
server.host: "192.168.113.67"

# Enables you to specify a path to mount Kibana at if you are running behind a proxy.
# Use the `server.rewriteBasePath` setting to tell Kibana if it should remove the baseP
# from requests it receives, and to prevent a deprecation warning at startup.
# This setting cannot end in a slash.
#server.basePath: ""

# Specifies whether Kibana should rewrite requests that are prefixed with
```

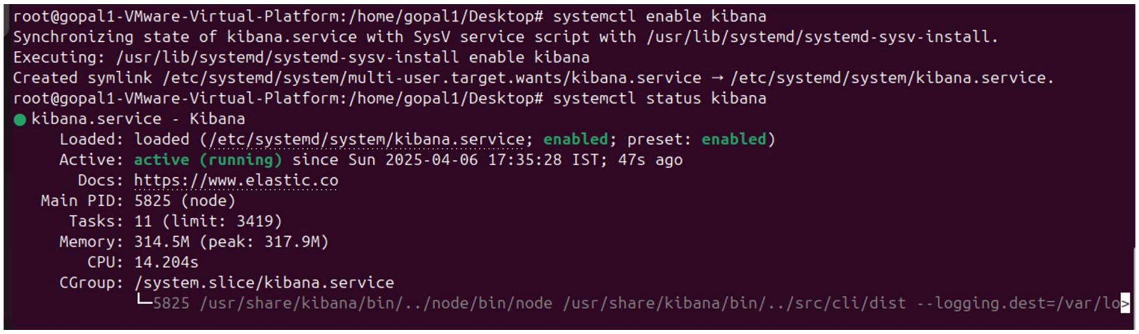


```
# The Kibana server's name.  This is used for display purposes.
#server.name: "your-hostname"

# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://192.162.113.67:9200"]

# Kibana uses an index in Elasticsearch to store saved searches, visualizat
# dashboards. Kibana creates a new index if the index doesn't already exist

#kibana.index: ".kibana"

# The default application to load.
#kibana.defaultAppId: "home"
```

Also here we have to restart kibana services and check status

systemctl restart kibana

systemctl status kibana

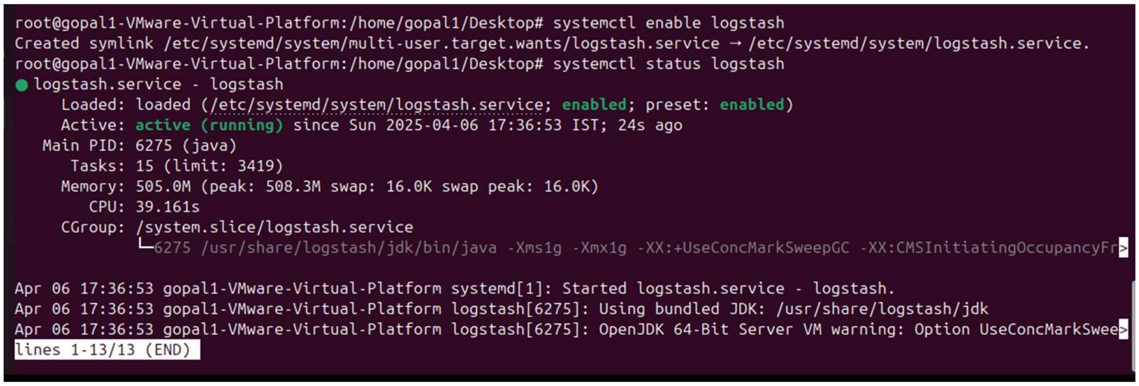


```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# systemctl enable kibana
Synchronizing state of kibana.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable kibana
Created symlink /etc/systemd/system/multi-user.target.wants/kibana.service → /etc/systemd/system/kibana.service.
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# systemctl status kibana
● kibana.service - Kibana
     Loaded: loaded (/etc/systemd/system/kibana.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-04-06 17:35:28 IST; 47s ago
       Docs: https://www.elastic.co
   Main PID: 5825 (node)
      Tasks: 11 (limit: 3419)
     Memory: 314.5M (peak: 317.9M)
        CPU: 14.204s
     CGroup: /system.slice/kibana.service
             └─5825 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/../src/cli/dist --logging.dest=/var/lo>
```

Also here we have to restart logstash services and check status

systemctl restart logstash

systemctl status logstash



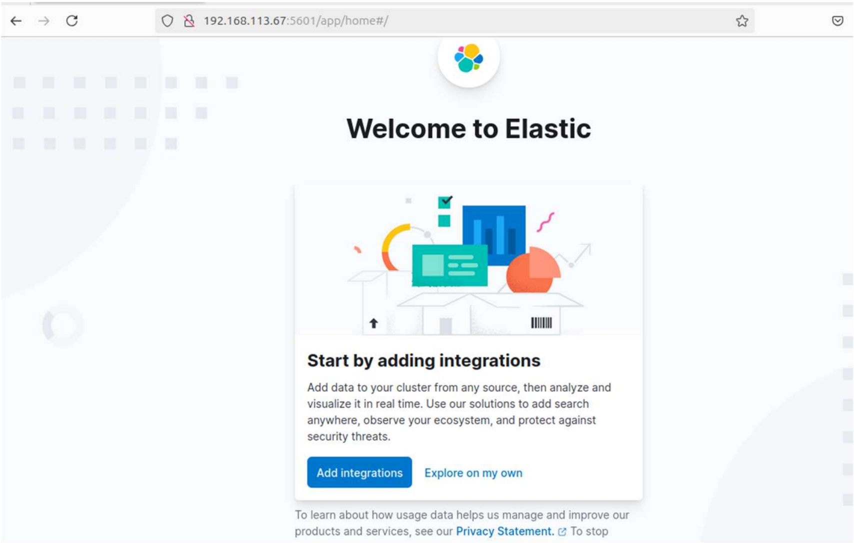
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# systemctl enable logstash
Created symlink /etc/systemd/system/multi-user.target.wants/logstash.service → /etc/systemd/system/logstash.service.
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# systemctl status logstash
● logstash.service - logstash
     Loaded: loaded (/etc/systemd/system/logstash.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-04-06 17:36:53 IST; 24s ago
   Main PID: 6275 (java)
      Tasks: 15 (limit: 3419)
     Memory: 505.0M (peak: 508.3M swap: 16.0K swap peak: 16.0K)
        CPU: 39.161s
     CGroup: /system.slice/logstash.service
             └─6275 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFr>

Apr 06 17:36:53 gopal1-VMware-Virtual-Platform systemd[1]: Started logstash.service - logstash.
Apr 06 17:36:53 gopal1-VMware-Virtual-Platform logstash[6275]: Using bundled JDK: /usr/share/logstash/jdk
Apr 06 17:36:53 gopal1-VMware-Virtual-Platform logstash[6275]: OpenJDK 64-Bit Server VM warning: Option UseConcMarkSwee>
lines 1-13/13 (END)
```

Then we can check our elk setup from webbrowser

192.168.155.12:5601

The interface looks like this

Next we have to select **add integration** option



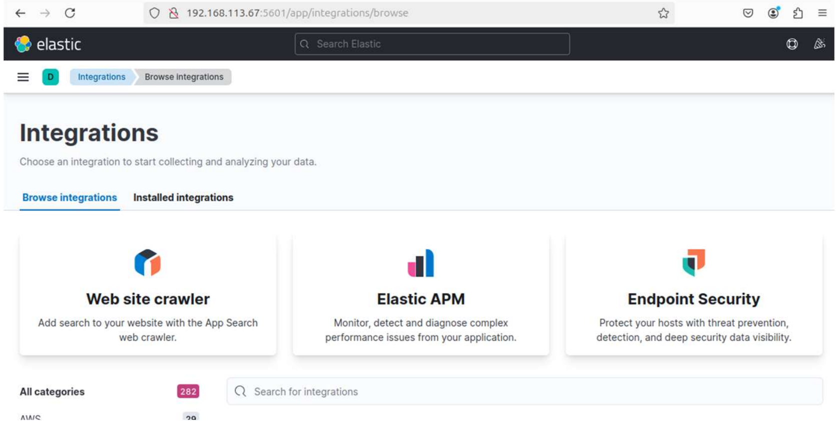
Welcome to Elastic

Start by adding integrations

Add data to your cluster from any source, then analyze and visualize it in real time. Use our solutions to add search anywhere, observe your ecosystem, and protect against security threats.

To learn about how usage data helps us manage and improve our products and services, see our Privacy Statement. To stop

Now our **ELK** setup is done.



Integrations

Choose an integration to start collecting and analyzing your data.

Web site crawler

Add search to your website with the App Search web crawler.

Elastic APM

Monitor, detect and diagnose complex performance issues from your application.

Endpoint Security

Protect your hosts with threat prevention, detection, and deep security data visibility.

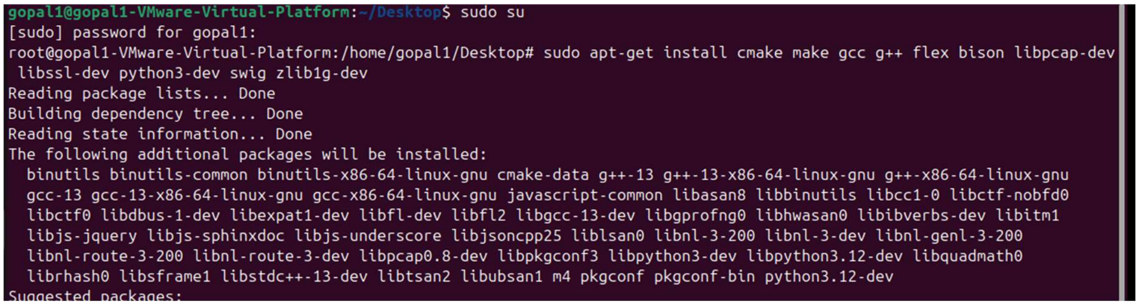
Now we are adding zeek in same ubantu using this commands as shown below:

Download source code from this website

https://zeek.org/get-zeek/

First we are using this command:

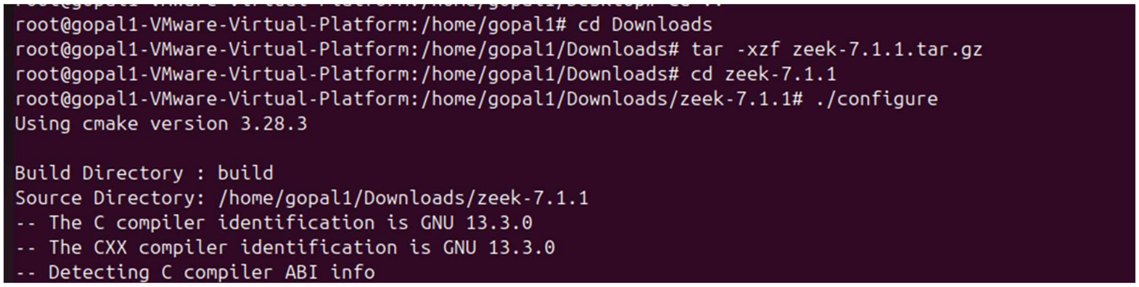
sudo apt-get install cmake make gcc g++ flex bison libpcap-dev libssl-dev python3-dev swig zlib1g-dev



```
gopal1@gopal1-VMware-Virtual-Platform:~/Desktop$ sudo su
[sudo] password for gopal1:
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# sudo apt-get install cmake make gcc g++ flex bison libpcap-dev
 libssl-dev python3-dev swig zlib1g-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cmake-data g++-13 g++-13-x86-64-linux-gnu g++-x86-64-linux-gnu
  gcc-13 gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu javascript-common libasan8 libbinutils libcc1-0 libctf-nobfd0
  libctf0 libdbus-1-dev libexpat1-dev libfl-dev libfl2 libgcc-13-dev libgprofng0 libhwasan0 libibverbs-dev libitm1
  libjs-jquery libjs-sphinxdoc libjs-underscore libjsoncpp25 liblsan0 libnl-3-200 libnl-3-dev libnl-genl-3-200
  libnl-route-3-200 libnl-route-3-dev libpcap0.8-dev libpkgconf3 libpython3-dev libpython3.12-dev libquadmath0
  librhash0 libsframe1 libstdc++-13-dev libtsan2 libubsan1 m4 pkgconf pkgconf-bin python3.12-dev
Suggested packages:
```

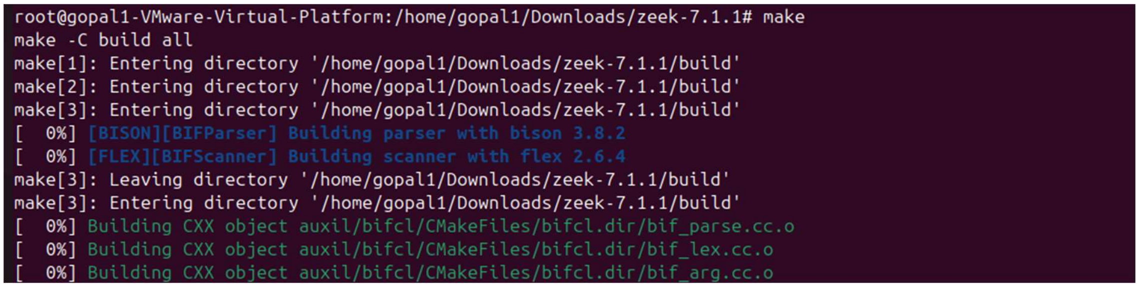
Next **cd Downloads** and **tar -xzf zeek(your version).tar.gz also**

Next **cd zeek(your version)** and **run ./configure for configuration**



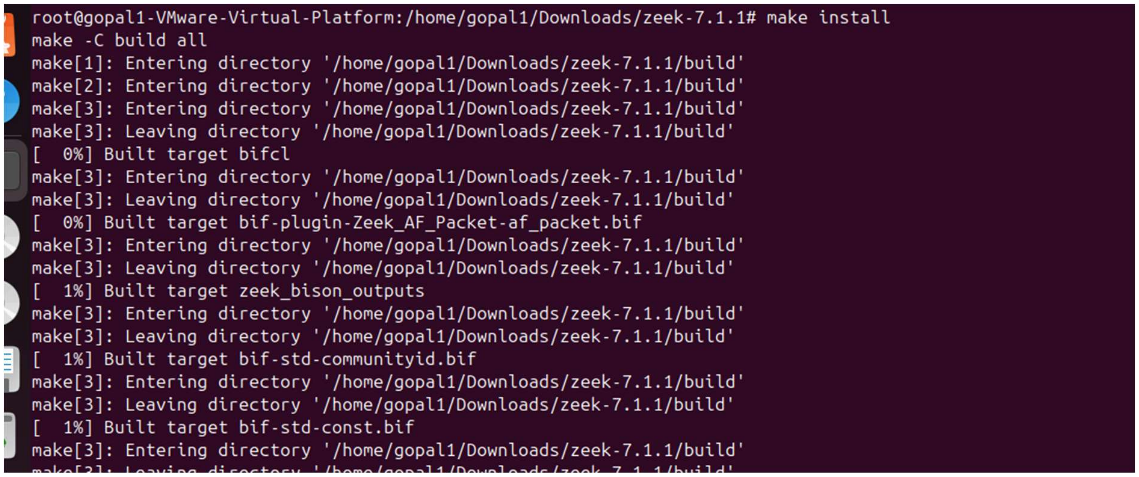
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1# cd Downloads
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# tar -xzf zeek-7.1.1.tar.gz
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# cd zeek-7.1.1
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# ./configure
Using cmake version 3.28.3

Build Directory : build
Source Directory: /home/gopal1/Downloads/zeek-7.1.1
-- The C compiler identification is GNU 13.3.0
-- The CXX compiler identification is GNU 13.3.0
-- Detecting C compiler ABI info
```

Next run **make** command



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# make
make -C build all
make[1]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[2]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  0%] [BISON][BIFParser] Building parser with bison 3.8.2
[  0%] [FLEX][BIFScanner] Building scanner with flex 2.6.4
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  0%] Building CXX object auxil/bifcl/CMakeFiles/bifcl.dir/bif_parse.cc.o
[  0%] Building CXX object auxil/bifcl/CMakeFiles/bifcl.dir/bif_lex.cc.o
[  0%] Building CXX object auxil/bifcl/CMakeFiles/bifcl.dir/bif_arg.cc.o
```

Next run **make install** command



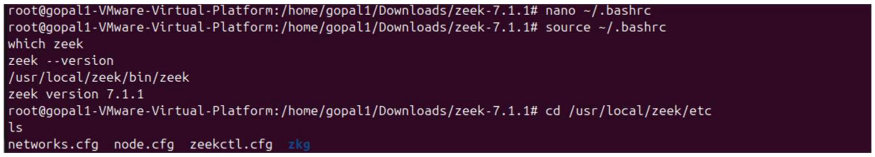
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# make install
make -C build all
make[1]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[2]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  0%] Built target bifcl
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  0%] Built target bif-plugin-Zeek_AF_Packet-af_packet.bif
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  1%] Built target zeek_bison_outputs
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  1%] Built target bif-std-communityid.bif
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
[  1%] Built target bif-std-const.bif
make[3]: Entering directory '/home/gopal1/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal1/Downloads/zeek-7.1.1/build'
```

To use zeek as a service we need to add the zeek home directory to the bashrc file.

export PATH=/usr/local/zeek/bin:$PATH add this in last line of **bashrc** file.

to apply changes made run source command and check zeek version and directory run this command :

source ~/.bashrc
which zeek
zeek --version

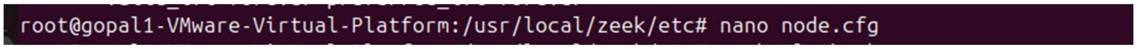


```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# nano ~/.bashrc
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# source ~/.bashrc
which zeek
zeek --version
/usr/local/zeek/bin/zeek
zeek version 7.1.1
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads/zeek-7.1.1# cd /usr/local/zeek/etc
ls
networks.cfg  node.cfg  zeekctl.cfg  zkg
```

Now change the directory to /usr/local/zeek/etc check the what files are there in the directory.

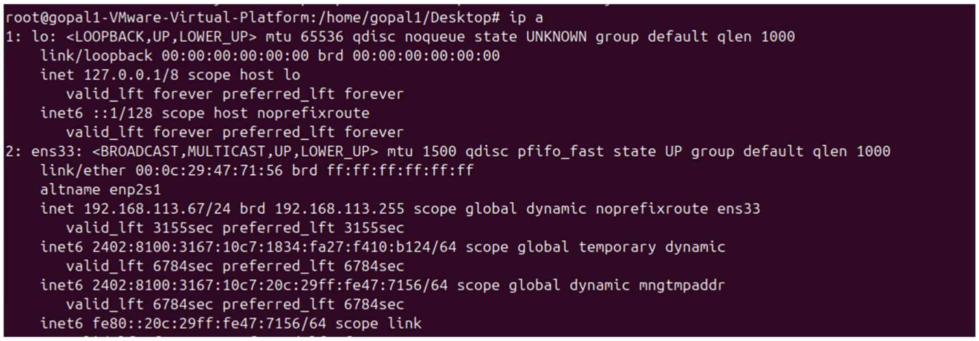
cd /usr/local/zeek/etc
ls

then **nano node.cfg**



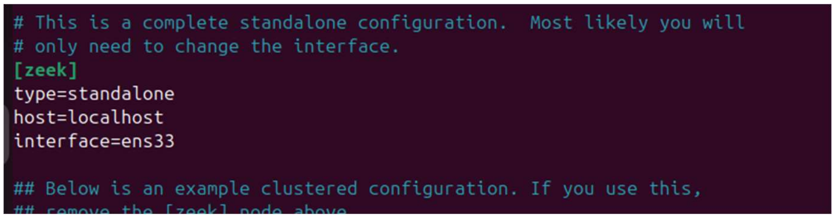
```
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/etc# nano node.cfg
```

First we have to check interface using **ip a** command



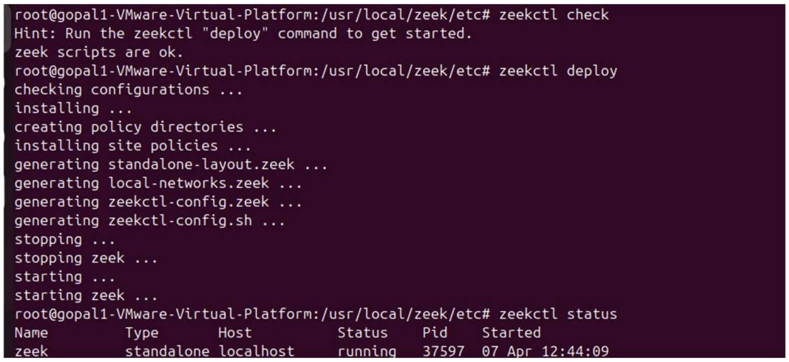
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Desktop# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:47:71:56 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.113.67/24 brd 192.168.113.255 scope global dynamic noprefixroute ens33
       valid_lft 3155sec preferred_lft 3155sec
    inet6 2402:8100:3167:10c7:1834:fa27:f410:b124/64 scope global temporary dynamic
       valid_lft 6784sec preferred_lft 6784sec
    inet6 2402:8100:3167:10c7:20c:29ff:fe47:7156/64 scope global dynamic mngtmpaddr
       valid_lft 6784sec preferred_lft 6784sec
    inet6 fe80::20c:29ff:fe47:7156/64 scope link
```

Then add interface in it. (in my case my interface name is ens33)



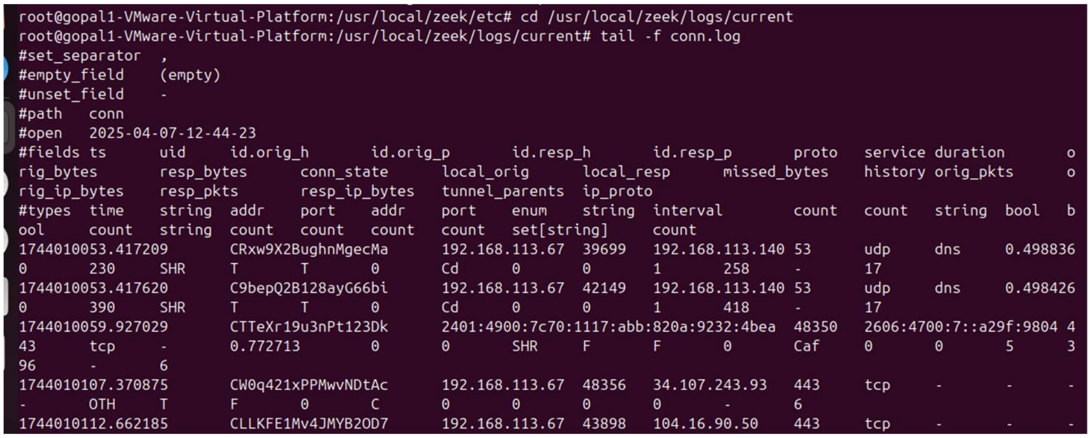
```
# This is a complete standalone configuration.  Most likely you will
# only need to change the interface.
[zeek]
type=standalone
host=localhost
interface=ens33

## Below is an example clustered configuration. If you use this,
## remove the [zeek] node above.
```

Now check zeek using **zeekctl check** command and next run **zeekctl deploy** for deployment also check for status using **zeekctl status**



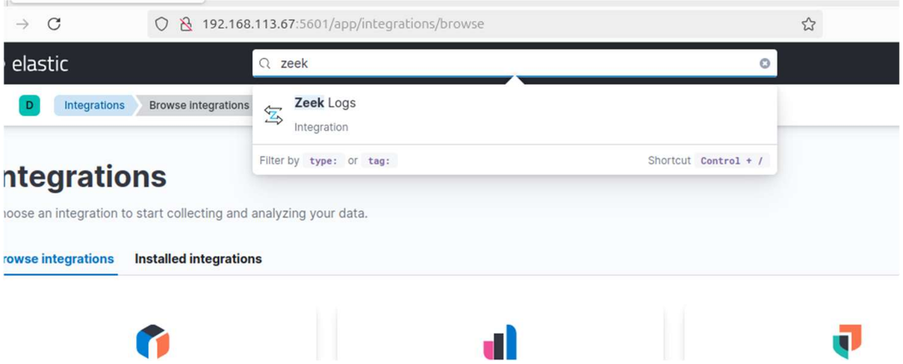
```
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl check
Hint: Run the zeekctl "deploy" command to get started.
zeek scripts are ok.
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl deploy
checking configurations ...
installing ...
creating policy directories ...
installing site policies ...
generating standalone-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
stopping ...
stopping zeek ...
starting ...
starting zeek ...
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl status
Name         Type       Host          Status    Pid    Started
zeek         standalone localhost     running   37597  07 Apr 12:44:09
```

Next **cd /usr/local/zeek/logs/current** and check logs using this command **tail -f conn.log**

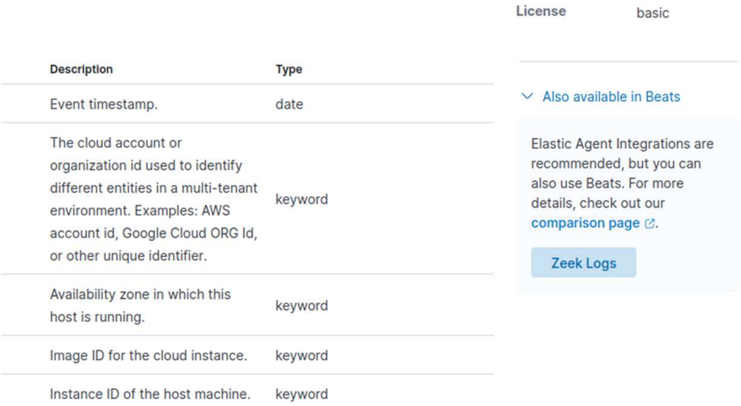


```
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/etc# cd /usr/local/zeek/logs/current
root@gopal1-VMware-Virtual-Platform:/usr/local/zeek/logs/current# tail -f conn.log
#set_separator  ,
#empty_field    (empty)
#unset_field    -
#path   conn
#open   2025-04-07-12-44-23
#fields ts      uid     id.orig_h       id.orig_p       id.resp_h       id.resp_p       proto   service duration       o
rig_bytes       resp_bytes      conn_state      local_orig      local_resp      missed_bytes    history orig_pkts      o
rig_ip_bytes    resp_pkts       resp_ip_bytes   tunnel_parents  ip_proto
#types  time    string  addr    port    addr    port    enum    string  interval        count   count   string  bool   b
ool     count   string  count   count   count   count   set[string]     count
1744010053.417209       CRxw9X2BughnMgecMa      192.168.113.67  39699   192.168.113.140 53      udp     dns     0.498836
0       230     SHR     T       T       0       Cd      0       0       1       258     -       17
1744010053.417620       C9bepQ2B128ayG66bi      192.168.113.67  42149   192.168.113.140 53      udp     dns     0.498426
0       390     SHR     T       T       0       Cd      0       0       1       418     -       17
1744010059.927029       CTTeXr19u3nPt123Dk      2401:4900:7c70:1117:abb:820a:9232:4bea  48350   2606:4700:7::a29f:9804  4
43      tcp     -       0.772713        0       0       SHR     F       F       0       Caf     0       0       5      3
96      -       6
1744010107.370875       CW0q421xPPMwvNDtAc      192.168.113.67  48356   34.107.243.93   443     tcp     -       -      -
-       OTH     T       F       0       C       0       0       0       0       -       6
1744010112.662185       CLLKFE1Mv4JMYB2OD7      192.168.113.67  43898   104.16.90.50    443     tcp     -       -      -
```

Then return on elk website and search **zeek logs**



Then click on **also available in beats** and click on **zeek logs.**



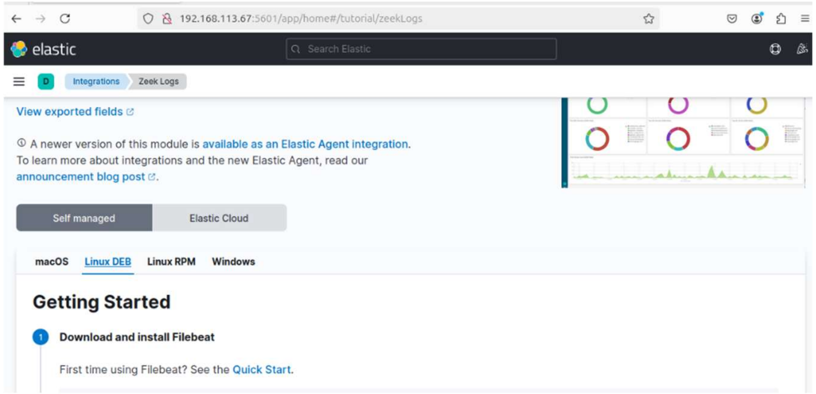
License basic

Description	Type
Event timestamp.	date
The cloud account or organization id used to identify different entities in a multi-tenant environment. Examples: AWS account id, Google Cloud ORG Id, or other unique identifier.	keyword
Availability zone in which this host is running.	keyword
Image ID for the cloud instance.	keyword
Instance ID of the host machine.	keyword

Also available in Beats

Elastic Agent Integrations are recommended, but you can also use Beats. For more details, check out our comparison page.

Next click on linux deb and this shows some command and go through this.



View exported fields

A newer version of this module is available as an Elastic Agent integration.
To learn more about integrations and the new Elastic Agent, read our announcement blog post.

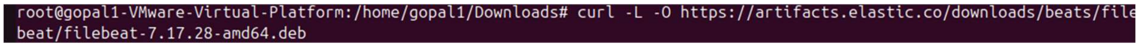
Getting Started

1 Download and install Filebeat

First time using Filebeat? See the Quick Start.

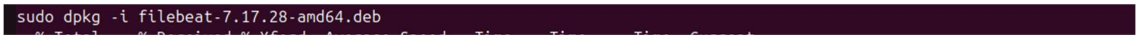
Now we are trying zeek logs capture on elk

Run this command:



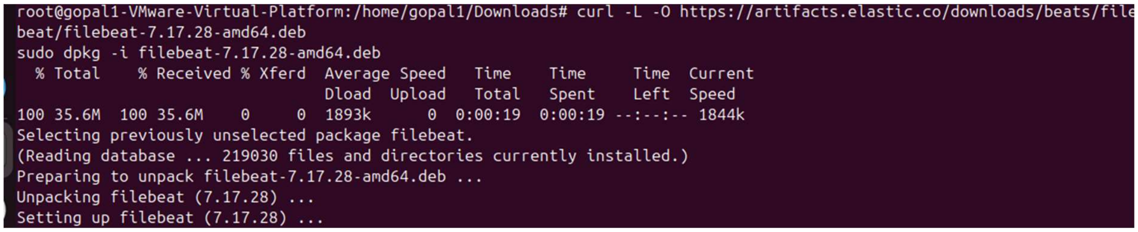
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# curl -L -O https://artifacts.elastic.co/downloads/beats/file
beat/filebeat-7.17.28-amd64.deb
```

After that this command



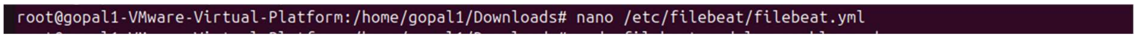
```
sudo dpkg -i filebeat-7.17.28-amd64.deb
```

After that it shows like this:

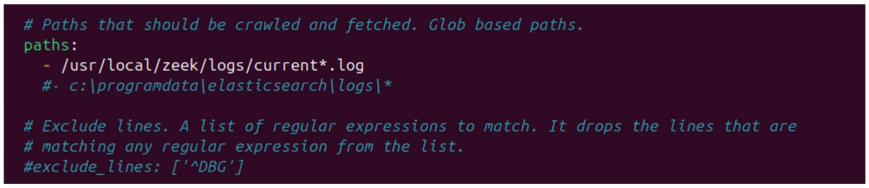


```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# curl -L -O https://artifacts.elastic.co/downloads/beats/file
beat/filebeat-7.17.28-amd64.deb
sudo dpkg -i filebeat-7.17.28-amd64.deb
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 35.6M  100 35.6M    0     0  1893k      0  0:00:19  0:00:19 --:--:-- 1844k
Selecting previously unselected package filebeat.
(Reading database ... 219030 files and directories currently installed.)
Preparing to unpack filebeat-7.17.28-amd64.deb ...
Unpacking filebeat (7.17.28) ...
Setting up filebeat (7.17.28) ...
```

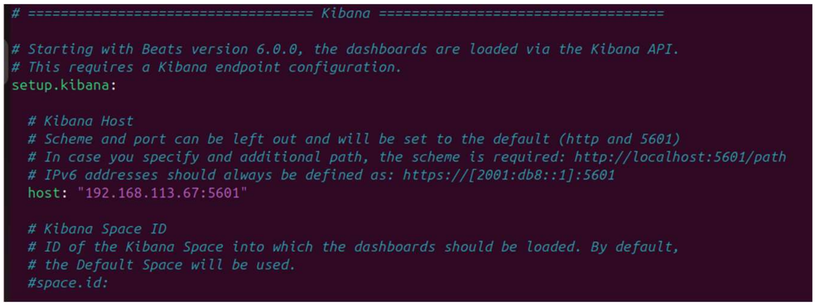
Then run this command make changes in file as shown below:



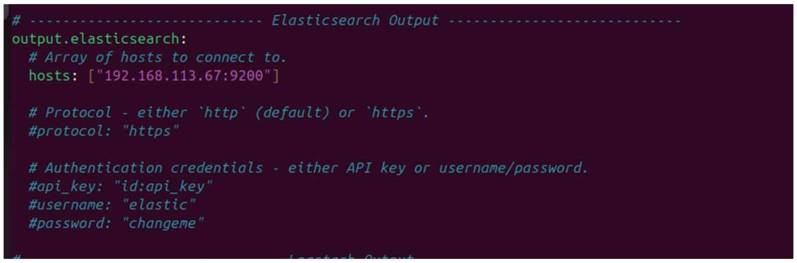
```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# nano /etc/filebeat/filebeat.yml
```



```
# Paths that should be crawled and fetched. Glob based paths.
paths:
  - /usr/local/zeek/logs/current*.log
  #- c:\programdata\elasticsearch\logs\*

# Exclude lines. A list of regular expressions to match. It drops the lines that are
# matching any regular expression from the list.
#exclude_lines: ['^DBG']
```



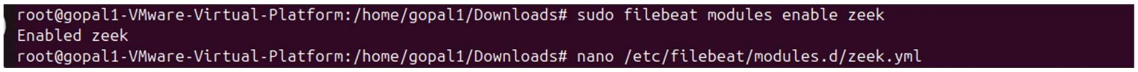
```
# ================================== Kibana ==================================

# Starting with Beats version 6.0.0, the dashboards are loaded via the Kibana API.
# This requires a Kibana endpoint configuration.
setup.kibana:

  # Kibana Host
  # Scheme and port can be left out and will be set to the default (http and 5601)
  # In case you specify and additional path, the scheme is required: http://localhost:5601/path
  # IPv6 addresses should always be defined as: https://[2001:db8::1]:5601
  host: "192.168.113.67:5601"

  # Kibana Space ID
  # ID of the Kibana Space into which the dashboards should be loaded. By default,
  # the Default Space will be used.
  #space.id:
```



```
# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["192.168.113.67:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"

  # Authentication credentials - either API key or username/password.
  #api_key: "id:api_key"
  #username: "elastic"
  #password: "changeme"
```

Enable zeek module using this command **filebeat modules enable zeek**

Then **nano /etc/filebeat/modules.d/zeek.yml** run this command



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# sudo filebeat modules enable zeek
Enabled zeek
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# nano /etc/filebeat/modules.d/zeek.yml
```

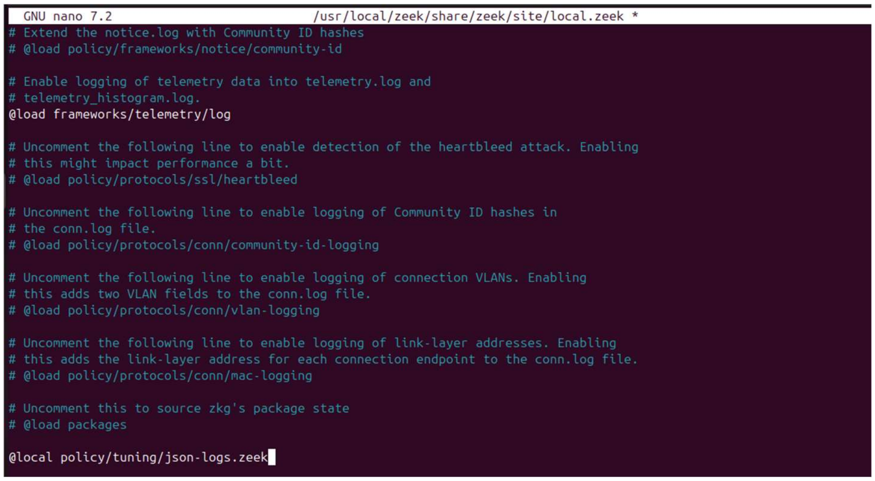
And add this in it. As it is

```
# Module: zeek
# Docs: https://www.elastic.co/guide/en/beats/filebeat/7.x/filebeat-module-zeek.html
- module: zeek
  capture_loss:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/capture_loss.log"]
  connection:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/conn.log"]
  dce_rpc:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/dce_rpc.log"]
  dhcp:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/dhcp.log"]
  dnp3:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/dnp3.log"]
  dns:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/dns.log"]
  dpd:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/dpd.log"]
  files:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/files.log"]
  ftp:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/ftp.log"]
  http:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/http.log"]
  intel:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/intel.log"]
  irc:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/irc.log"]
  kerberos:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/kerberos.log"]
  modbus:
    enabled: true
    var.paths: ["/usr/local/zeek/logs/current/modbus.log"]
  mysql:
```

```
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/mysql.log"]
notice:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/notice.log"]
ntlm:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/ntlm.log"]
ntp:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/ntp.log"]
ocsp:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/oscp.log"]
pe:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/pe.log"]
radius:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/radius.log"]
rdp:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/rdp.log"]
rfb:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/rfb.log"]
signature:
  enabled: false
  var.paths: ["/usr/local/zeek/logs/current/signature.log"]
sip:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/sip.log"]
smb_cmd:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/smb_cmd.log"]
smb_files:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/smb_files.log"]
smb_mapping:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/smb_mapping.log"]
smtp:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/smtp.log"]
snmp:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/snmp.log"]
socks:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/socks.log"]
ssh:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/ssh.log"]
ssl:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/ssl.log"]
stats:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/stats.log"]
syslog:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/syslog.log"]
traceroute:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/traceroute.log"]
tunnel:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/tunnel.log"]
weird:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/weird.log"]
x509:
  enabled: true
  var.paths: ["/usr/local/zeek/logs/current/x509.log"]
  # Set custom paths for the log files. If left empty,
  # Filebeat will choose the paths depending on your OS.
  #var.paths:
```

Then run nthis command **nano /usr/local/zeek/share/zeek/site/local.zeek** and add this line

@load policy/tuning/json-logs.zeek at bottom

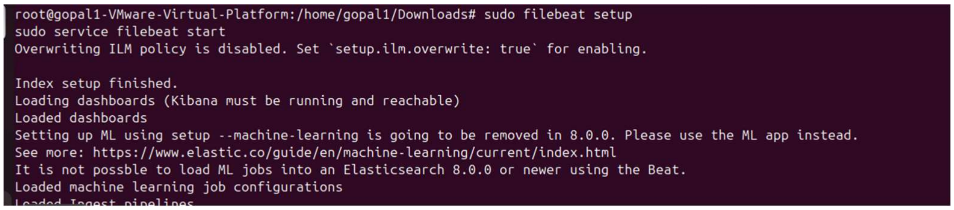


```
  GNU nano 7.2                           /usr/local/zeek/share/zeek/site/local.zeek *
# Extend the notice.log with Community ID hashes
# @load policy/frameworks/notice/community-id

# Enable logging of telemetry data into telemetry.log and
# telemetry_histogram.log.
@load frameworks/telemetry/log

# Uncomment the following line to enable detection of the heartbleed attack. Enabling
# this might impact performance a bit.
# @load policy/protocols/ssl/heartbleed

# Uncomment the following line to enable logging of Community ID hashes in
# the conn.log file.
# @load policy/protocols/conn/community-id-logging

# Uncomment the following line to enable logging of connection VLANs. Enabling
# this adds two VLAN fields to the conn.log file.
# @load policy/protocols/conn/vlan-logging

# Uncomment the following line to enable logging of link-layer addresses. Enabling
# this adds the link-layer address for each connection endpoint to the conn.log file.
# @load policy/protocols/conn/mac-logging

# Uncomment this to source zkg's package state
# @load packages

@local policy/tuning/json-logs.zeek
```

Then run this two commands and your setup is done

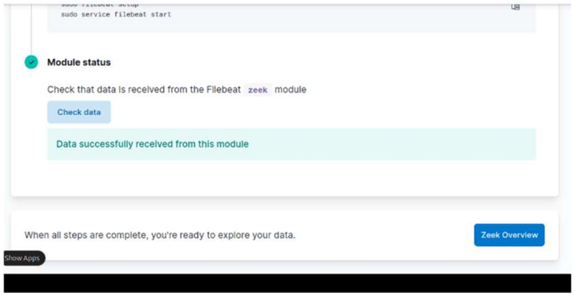
filebeat setup

service filebeat start



```
root@gopal1-VMware-Virtual-Platform:/home/gopal1/Downloads# sudo filebeat setup
sudo service filebeat start
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite: true` for enabling.

Index setup finished.
Loading dashboards (Kibana must be running and reachable)
Loaded dashboards
Setting up ML using setup --machine-learning is going to be removed in 8.0.0. Please use the ML app instead.
See more: https://www.elastic.co/guide/en/machine-learning/current/index.html
It is not possble to load ML jobs into an Elasticsearch 8.0.0 or newer using the Beat.
Loaded machine learning job configurations
Loaded Ingest pipelines
```

Next click on check data and click on zeek overview



```
sudo filebeat setup
sudo service filebeat start
```

Module status

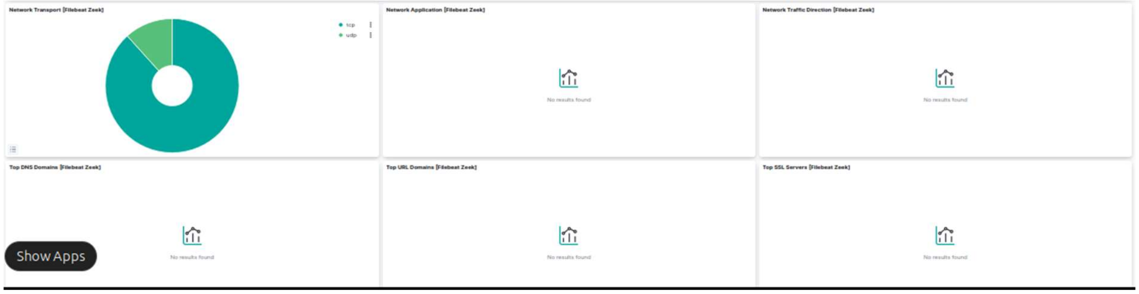
Check that data is received from the Filebeat `zeek` module

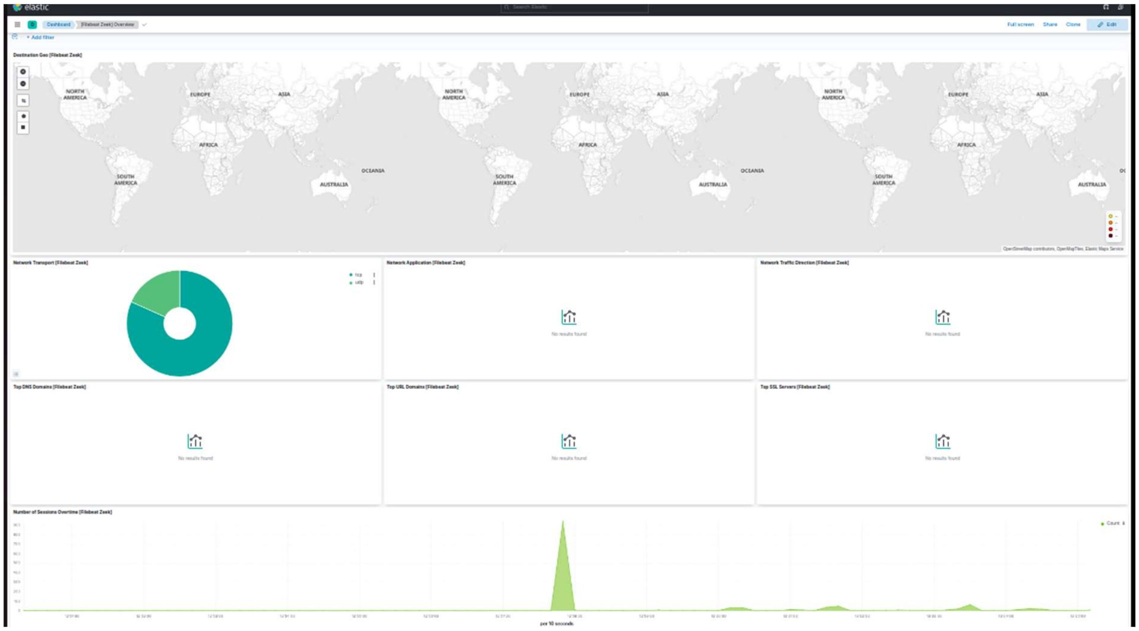
Check data

Data successfully received from this module

When all steps are complete, you're ready to explore your data.

You interface look like this


Network Transport [Filebeat Zeek]
tcp
udp
Network Application [Filebeat Zeek]
No results found
Network Traffic Direction [Filebeat Zeek]
No results found
Top DNS Domains [Filebeat Zeek]
Show Apps
No results found
Top URL Domains [Filebeat Zeek]
No results found
Top SSL Servers [Filebeat Zeek]
No results found


elastic
Search Elastic
Dashboard
[Filebeat Zeek] Overview
Full screen
Share
Clone
Edit
+ Add filter
Destination Geo [Filebeat Zeek]
NORTH AMERICA
EUROPE
ASIA
AFRICA
SOUTH AMERICA
AUSTRALIA
OCEANIA
OpenStreetMap contributors, OpenMapTiles, Elastic Maps Service
Network Transport [Filebeat Zeek]
tcp
udp
Network Application [Filebeat Zeek]
No results found
Network Traffic Direction [Filebeat Zeek]
No results found
Top DNS Domains [Filebeat Zeek]
No results found
Top URL Domains [Filebeat Zeek]
No results found
Top SSL Servers [Filebeat Zeek]
No results found
Number of Sessions Overtime [Filebeat Zeek]
Count
per 10 seconds