Elk 2.0 on cloud

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.

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

```
root@gopal-VMware-Virtual-Platform:/usr/local/zeek/logs/current# sudo apt-get install cmake make gcc g++ flex bison libp
cap-dev libssl-dev python3-dev swig zlib1g-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
cmake is already the newest version (3.28.3-1build7).
make is already the newest version (4.3-4.1build2).
gcc is already the newest version (4:13.2.0-7ubuntu1).
g++ is already the newest version (4:13.2.0-7ubuntu1).
flex is already the newest version (2.6.4-8.2build1).
bison is already the newest version (2:3.8.2+dfsg-1build2).
libpcap-dev is already the newest version (1.10.4-4.1ubuntu3).
libssl-dev is already the newest version (3.0.13-0ubuntu3.5).
python3-dev is already the newest version (3.12.3-0ubuntu2).
swig is already the newest version (4.2.0-2ubuntu1).
zlib1g-dev is already the newest version (1:1.3.dfsg-3.1ubuntu2.1).
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
```

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

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

```
root@gopal-VMware-Virtual-Platform:/home/gopal/Downloads# ls
Snort.docx  splunk-9.1.2-b6b9c8185839-linux-2.6-amd64.deb  zeek-7.1.1  zeek-7.1.1.tar.gz
root@gopal-VMware-Virtual-Platform:/home/gopal/Downloads# cd zeek-7.1.1
root@gopal-VMware-Virtual-Platform:/home/gopal/Downloads/zeek-7.1.1# ./configure
Using cmake version 3.28.3

Build Directory : build
Source Directory: /home/gopal/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
-- Detecting C compiler ABI info - done
-- Check for working C compiler: /usr/bin/cc - skipped
-- Detecting C compile features
```

Next run **make** command

```
root@gopal-VMware-Virtual-Platform:/home/gopal/Downloads/zeek-7.1.1# make
make -C build all
make[1]: Entering directory '/home/gopal/Downloads/zeek-7.1.1/build'
make[2]: Entering directory '/home/gopal/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal/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/gopal/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal/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
[  0%] Building CXX object auxil/bifcl/CMakeFiles/bifcl.dir/module_util.cc.o
[  0%] Linking CXX executable bifcl
make[3]: Leaving directory '/home/gopal/Downloads/zeek-7.1.1/build'
[  0%] Built target bifcl
make[3]: Entering directory '/home/gopal/Downloads/zeek-7.1.1/build'
make[3]: Leaving directory '/home/gopal/Downloads/zeek-7.1.1/build'
make[3]: Entering directory '/home/gopal/Downloads/zeek-7.1.1/build'
```

Next run **make install** command

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

cd /usr/local/zeek/etc

ls

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

then **nano node.cfg**

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

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

```
root@gopal-VMware-Virtual-Platform:/usr/local/zeek/etc# 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 fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:d1:b7:18 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.155.213/24 brd 192.168.155.255 scope global dynamic noprefixroute ens33
       valid_lft 443sec preferred_lft 443sec
    inet6 2402:8100:3165:d81a:d718:db78:f730:e4b8/64 scope global temporary dynamic
       valid_lft 7178sec preferred_lft 7178sec
    inet6 2402:8100:3165:d81a:20c:29ff:fed1:b718/64 scope global dynamic mngtmpaddr
       valid_lft 7178sec preferred_lft 7178sec
    inet6 fe80::20c:29ff:fed1:b718/64 scope link
       valid_lft forever preferred_lft forever
```

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

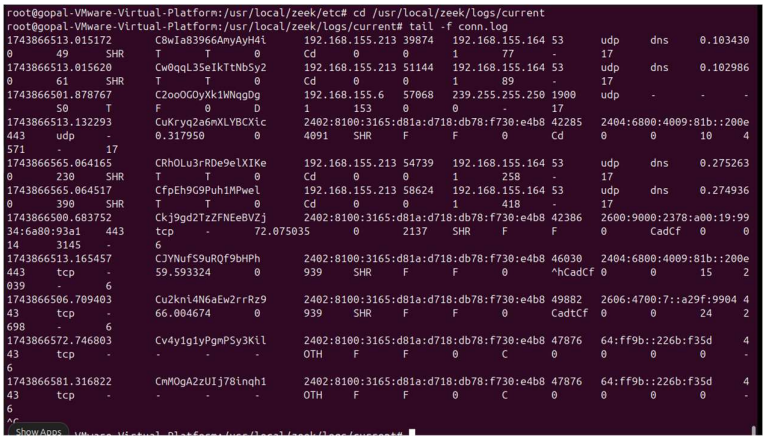
```
# Example ZeekControl node configuration.
#
# This example has a standalone node ready to go except for possibly changing
# the sniffing interface.

# 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.

#[
Ubuntu 24.04.1 LTS amd64
#host=localhost
#
#[manager]
#type=manager
#host=localhost
#
```

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

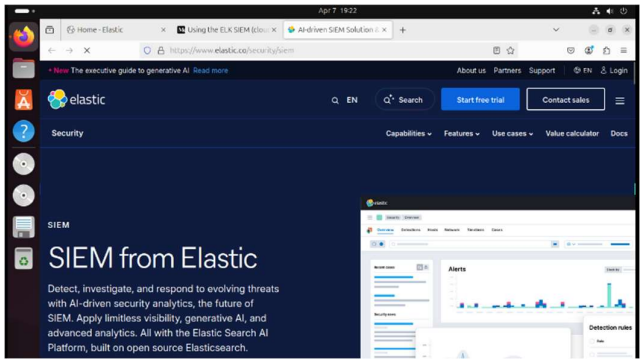
```
root@gopal-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl check
Hint: Run the zeekctl "deploy" command to get started.
zeek scripts are ok.
root@gopal-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl deploy
checking configurations ...
installing ...
creating policy directories ...
installing 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@gopal-VMware-Virtual-Platform:/usr/local/zeek/etc# zeekctl status
Name         Type       Host          Status    Pid    Started
zeek         standalone localhost     running   48799  05 Apr 20:51:34
```

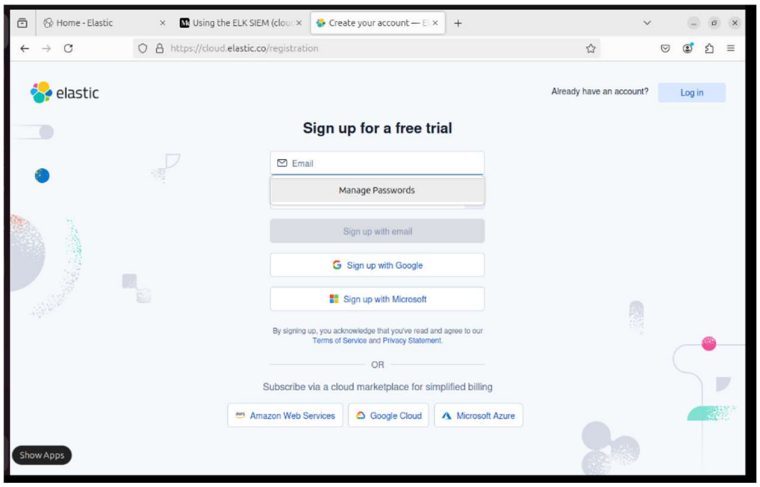
Next **cd /usr/local/zeek/logs/current** and check logs using this command **tail -f conn.log**



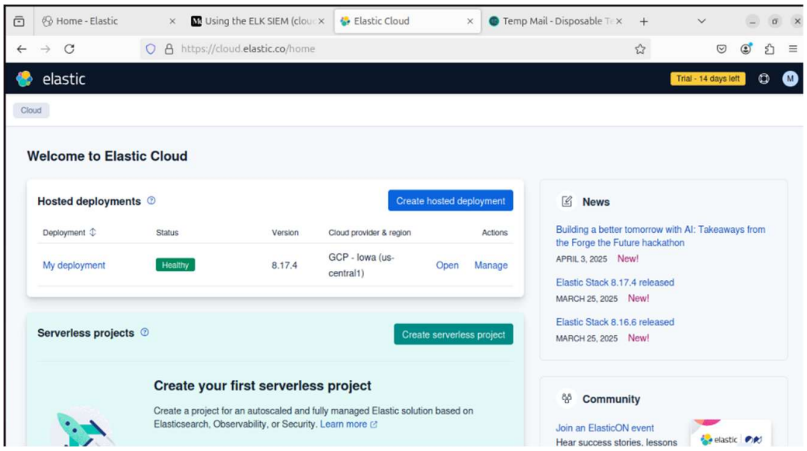
Now setup elasticsearch go to elasticsearch website



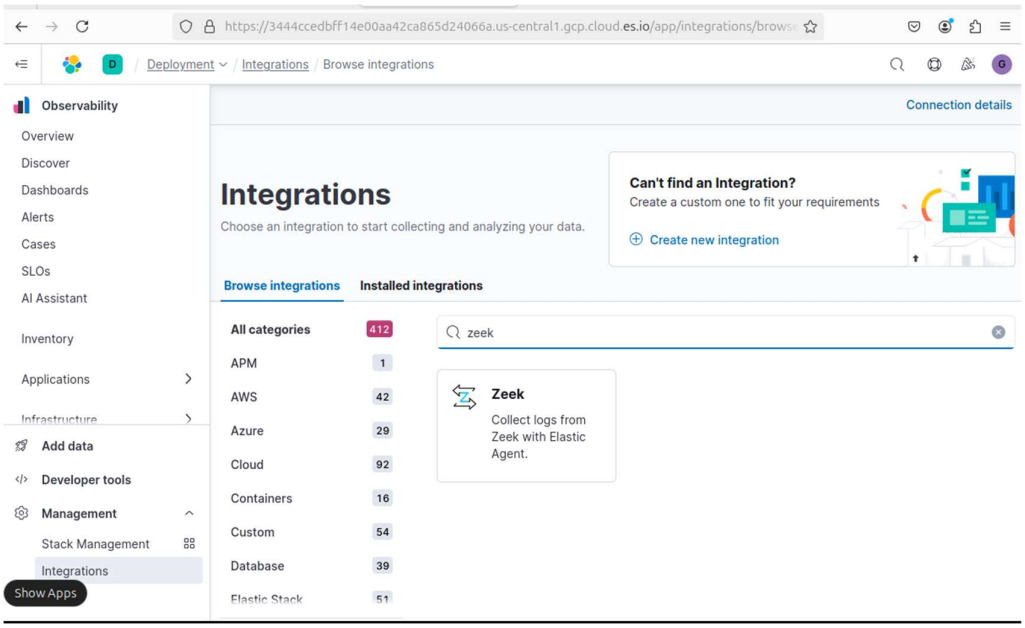
Next create your account (I am using free trial account)



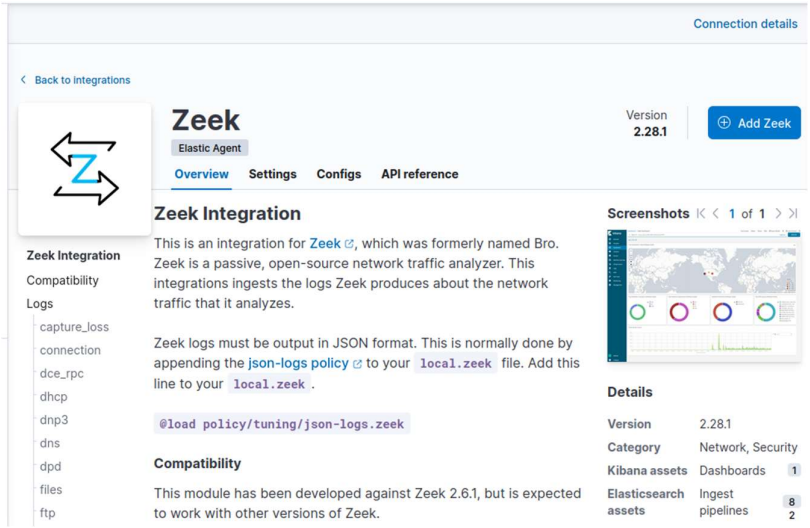
After some configuration your interface looks like this



Next click on **management -> integrations** and **search for zeek**



Then its look like this

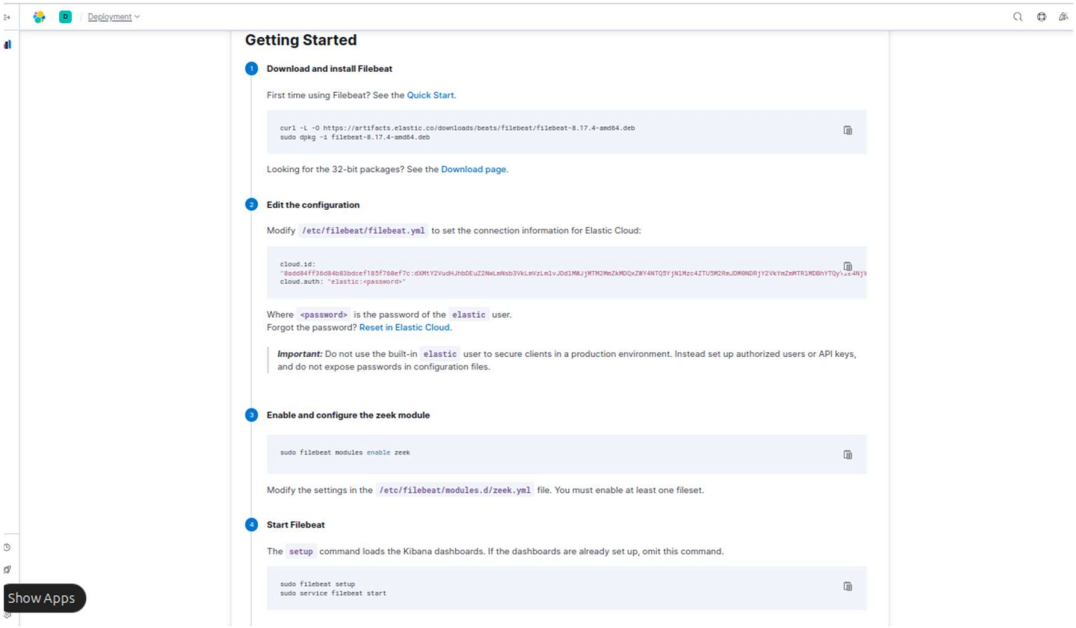


Then run nthis command **nano /usr/local/zeek/share/zeek/site/local.zeek** and add this line

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

```
# 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
@load policy/tuning/json-logs.zeek
```

For showing logs we have to run this commands and make changes in some files



Then run this commands one by one

```
gopal@gopal-VMware-Virtual-Platform:~/Desktop$ sudo su
[sudo] password for gopal:
root@gopal-VMware-Virtual-Platform:/home/gopal/Desktop# curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat
/filebeat-8.17.4-amd64.deb
sudo dpkg -i filebeat-8.17.4-amd64.deb
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 53.3M  100 53.3M    0     0  2125k      0  0:00:25  0:00:25 --:--:-- 2615k
(Reading database ... 183553 files and directories currently installed.)
Preparing to unpack filebeat-8.17.4-amd64.deb ...
Unpacking filebeat (8.17.4) over (8.17.4) ...
Setting up filebeat (8.17.4) ...
root@gopal-VMware-Virtual-Platform:/home/gopal/Desktop#
```

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

```
root@gopal-VMware-Virtual-Platform:~# nano /etc/filebeat/filebeat.yml
```

```
# Unique ID among all inputs, an ID is required.
id: my-filestream-id

# Change to true to enable this input configuration.
enabled: false

# 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']

# Include lines. A list of regular expressions to match. It exports the lines that are
# matching any regular expression from the list.
#include_lines: ['^ERR', '^WARN']
```

```
# ============================== Elastic Cloud ==============================

# These settings simplify using Filebeat with the Elastic Cloud (https://cloud.elastic.co/).

# The cloud.id setting overwrites the `output.elasticsearch.hosts` and
# `setup.kibana.host` options.
# You can find the `cloud.id` in the Elastic Cloud web UI.
#cloud.id:
cloud.id: "8add84ff36d84b83bdcef185f760ef7c:dXMtY2VudHJhbDEuZ2NwLmNsb3VkLmVzLmlvJDdlMWJjMTM2MmZkMDQxZWY4NTQ5YjNlMzc4ZTU>
cloud.auth: "elastic:kIFdzqVbwj1kocOyldAPGV7I"
# The cloud.auth setting overwrites the `output.elasticsearch.username` and
# `output.elasticsearch.password` settings. The format is `<user>:<pass>`.
#cloud.auth:
```

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

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

```
root@gopal-VMware-Virtual-Platform:~# sudo filebeat modules enable zeek
Enabled zeek
root@gopal-VMware-Virtual-Platform:~# nano /etc/filebeat/modules.d/z
zeek.yml                zookeeper.yml.disabled  zoom.yml.disabled       zscaler.yml.disabled
root@gopal-VMware-Virtual-Platform:~# nano /etc/filebeat/modules.d/z
zeek.yml                zookeeper.yml.disabled  zoom.yml.disabled       zscaler.yml.disabled
root@gopal-VMware-Virtual-Platform:~# 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:
```

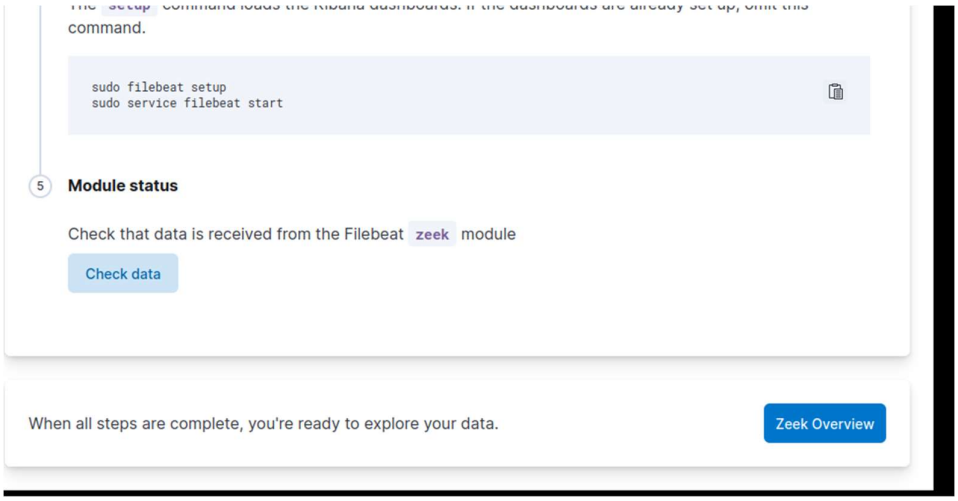
Then run this two commands and your setup is done

filebeat setup

service filebeat start

```
root@gopal-VMware-Virtual-Platform:~# sudo filebeat setup
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite: true` for enabling.

Index setup finished.
Loading dashboards (Kibana must be running and reachable)
```

After this go to elastic search website and click on **check data** next click **zeek overview**



Now your dashboard is ready you can check