

Infosys - DGARM

Observability Cluster Setup Document

Last Updated:- 5th Feb, 2026

By

Prashant Kashyap

Document Control

Document No.	Doc_1
Document Version	1
Prepared By	Prashant Kashyap
Release Date	5 th Feb 2026

Table of Contents

Infosys - DGARM	1
Document Control	2
Installation of Elasticsearch	4
Kibana Installation	12
APM Installation on ELK setup	15
Fleet Installation in Elastic Cluster	18

Installation of Elasticsearch

1. Enable subscription-manager register

```
subscription-manager register --username actual_user --password  
actual_pass
```

- **Note :-** After installation please subscription-manager unregister

Command :- subscription-manager unregister

2 . Install java on node (server), check the java version and Installation it

```
yum install java*  
java -version
```

3. make Elasticsearch as user

```
useradd elasticsearch  
groupadd elasticsearch
```

4. create data directory as per elasticsearch node functionality

```
mkdir /data // for normal node master and data  
mkdir /ml-data // for ml node
```

5. create ELASTIC Directory for installation work

```
mkdir ELASTIC  
cd ELASTIC
```

Download Elasticsearch rpm package

```
wget
```

```
https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-  
9.1.5-x86_64.rpm
```

Install Elasticsearch

```
sudo rpm -ivh elasticsearch-9.1.5-x86_64.rpm
```

```
[root@localhost ELASTIC]# ls -ltr  
total 660728  
-rw-r--r--. 1 root root      161 Oct  6 16:44 elasticsearch-9.1.5-x86_64.rpm.sha512  
-rw-r--r--. 1 root root 676579945 Oct  6 16:44 elasticsearch-9.1.5-x86_64.rpm
```

6. take backup of original elasticsearch.yml file before make any changes

```
cp /etc/elasticsearch/elasticsearch.yml  
/etc/elasticsearch/elasticsearch.yml.backup
```

7. Change ownership of elk directories and owner

```
chown -R root:elasticsearch /etc/elasticsearch
chown -R root:elasticsearch /data
chown -R root:elasticsearch /var/log/elasticsearch
```

```
[root@localhost elasticsearch]# ls -ltr
total 56
-rw-rw----. 1 root      elasticsearch  197 Oct  3 03:41 roles.yml
-rw-rw----. 1 root      elasticsearch  473 Oct  3 03:41 role_mapping.yml
-rw-rw----. 1 root      elasticsearch 20073 Oct  3 03:41 log4j2.properties
-rw-rw----. 1 root      elasticsearch  3445 Oct  3 03:41 jvm.options
-rw-rw----. 1 root      elasticsearch  1042 Oct  3 03:41 elasticsearch-plugins.example.yml
drwxr-s---. 2 root      elasticsearch    6 Oct  3 03:43 jvm.options.d
-rw-rw----. 1 root      elasticsearch   536 Jan  6 17:21 elasticsearch.keystore
-rw-r-----. 1 root      elasticsearch  4054 Jan  6 17:22 elasticsearch.yml.backup
drwxr-x---. 2 elasticsearch elasticsearch 4096 Jan 13 17:07 certs
-rw-rw----. 1 root      elasticsearch  5627 Jan 13 17:31 elasticsearch.yml
-rw-rw----. 1 root      elasticsearch    0 Jan 13 19:39 users
-rw-rw----. 1 root      elasticsearch    0 Jan 13 19:39 users_roles
[root@localhost elasticsearch]#
```

8. Put necessary node certificates in elasticsearch certs folder

```
cd /etc/elasticsearch/certs
```

```
egit/ elasticsearch/
[root@localhost ~]# cd /etc/elasticsearch/certs/
[root@localhost ~]# ls -l
total 56
-rw-r--r--. 1 elasticsearch elasticsearch 4175 Oct 23 16:31 ca.crt
-rw-r--r--. 1 elasticsearch elasticsearch 229 Oct 23 17:13 [redacted].cnf
-rw-r--r--. 1 elasticsearch elasticsearch 2155 Oct 23 17:13 [redacted].crt
-rw-r--r--. 1 elasticsearch elasticsearch 6330 Oct 23 17:13 [redacted].fullchain.crt
-rw-r--r--. 1 elasticsearch elasticsearch 3272 Oct 23 17:13 [redacted].key
-rw-rw----. 1 elasticsearch elasticsearch 1935 Oct 22 13:19 http_ca.crt
-rw-rw----. 1 elasticsearch elasticsearch 10093 Oct 22 13:19 http.p12
-rw-r--r--. 1 elasticsearch elasticsearch 2175 Oct 23 13:56 intermediateCA.crt
-rw-rw----. 1 elasticsearch elasticsearch 5838 Oct 22 13:19 transport.p12
[root@localhost ~]#
```

Note :- .cnf, .dgarm.gov.in.crt, fullchain.crt and key will differ as per node

9. Enter host entry in hosts file

Note : host entry will be different for DR and Prod as per cluster design or setup

```
nano /etc/hosts
```

All ELK Cluster node entry

10. go to elasticsearch folder and make changes in yml file

```
nano elasticsearch.yml
```

```
#
cluster.name: [REDACTED]
#
# ----- Node -----
#
# Use a descriptive name for the node:
#
node.name: [REDACTED]
node.roles: [ master, data_hot, data_content ]
#
# Add custom attributes to the node:
#
#node.attr.rack: r1
#
# ----- Paths -----
#
# Path to directory where to store the data (separate multiple locations by comma)
#
path.data: /data
#
# Path to log files:
#
path.logs: /var/log/elasticsearch
#
# ----- Memory -----
#
```

```
#network.host: [REDACTED]
network.host: [REDACTED]
http.port: 9200
#
# 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 -----
#
# 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"]
discovery.seed_hosts: ["[REDACTED]", "[REDACTED]", "[REDACTED]"]
#discovery.seed_hosts: ["[REDACTED]"]
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
#cluster.initial_master_nodes: ["node-1", "node-2"]
cluster.initial_master_nodes: ["[REDACTED]", "[REDACTED]", "[REDACTED]"]
#cluster.initial_master_nodes: ["[REDACTED]"]
```

```

xpack.security.enrollment.enabled: true

# Enable encryption for HTTP API client connections, such as Kibana, Logstash, and Agents
xpack.security.http.ssl:
#   enabled: true
#   keystore.path: certs/http.p12

# Enable encryption and mutual authentication between cluster nodes
xpack.security.transport.ssl:
#   enabled: true
#   verification_mode: certificate
#   keystore.path: certs/transport.p12
#   truststore.path: certs/transport.p12
xpack.security.transport.ssl.enabled: false
xpack.security.enabled: true
xpack.security.transport.ssl.enabled: true
xpack.security.transport.ssl.verification_mode: certificate
xpack.security.transport.ssl.certificate: /etc/elasticsearch/certs/[REDACTED].fullchain.crt
xpack.security.transport.ssl.key: /etc/elasticsearch/certs/[REDACTED].key
xpack.security.transport.ssl.certificate_authorities: ["/etc/elasticsearch/certs/ca.crt"]
#xpack.security.transport.ssl.keystore.path: /etc/elasticsearch/certs/transport.p12
#xpack.security.transport.ssl.truststore.path: /etc/elasticsearch/certs/transport.p12

xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.certificate: /etc/elasticsearch/certs/[REDACTED].fullchain.crt
xpack.security.http.ssl.key: /etc/elasticsearch/certs/[REDACTED].key
xpack.security.http.ssl.certificate_authorities: ["/etc/elasticsearch/certs/ca.crt"]
#xpack.security.http.ssl.keystore.path: /etc/elasticsearch/certs/transport.p12
#xpack.security.http.ssl.truststore.path: /etc/elasticsearch/certs/transport.p12

# Create a new cluster with the current node only
# Additional nodes can still join the cluster later
#cluster.initial_master_nodes: ["dcpl6kschmdh01"]

# Allow HTTP API connections from anywhere
# Connections are encrypted and require user authentication
http.host: 0.0.0.0

```

```

# Allow HTTP API connections from anywhere
# Connections are encrypted and require user authentication
http.host: 0.0.0.0

# Allow other nodes to join the cluster from anywhere
# Connections are encrypted and mutually authenticated
#transport.host: 0.0.0.0
#bootstrap.memory_lock: true

action.destructive_requires_name: true

```

11. start 9200 and 9300 port

```

systemctl start firewalld
firewall-cmd --permanent --add-port=9200/tcp
firewall-cmd --permanent --add-port=9300/tcp
systemctl reload firewalld

```

12. start the elasticsearch service

```
sudo systemctl daemon-reload
sudo systemctl enable elasticsearch
sudo systemctl start elasticsearch
sudo systemctl restart elasticsearch
sudo systemctl status elasticsearch
sudo systemctl stop elasticsearch
```

13. Commands for check the error of Elasticsearch

```
sudo journalctl -u elasticsearch.service
sudo journalctl -u elasticsearch.service -f
```

14. Elasticsearch Instance

```
ps aux | grep elasticsearch
sudo netstat -plnt | grep 9200
```

15. command to change password for Elasticsearch user i custom form

```
/usr/share/bin/elasticsearch-reset-password -u elastic -i // set
custom password
```

Note:- The same steps are followed for all other Elasticsearch nodes.

Overview for Elasticsearch nodes:- 3 master nodes, 2 hot data nodes, 4 warm data nodes, 2 ml nodes.

Logstash Installation

1. Enable subscription-manager register

```
subscription-manager register --username ABC --password  
abc@123
```

Note :- After installation please subscription-manager unregister

Command :- subscription-manager unregister

2. Install java on node (server), check the java version and Installation it

```
yum install java*  
java -version
```

3. make logstash as user and group

```
useradd logstash  
Groupadd logstash
```

4. create logstash Directory for installation work

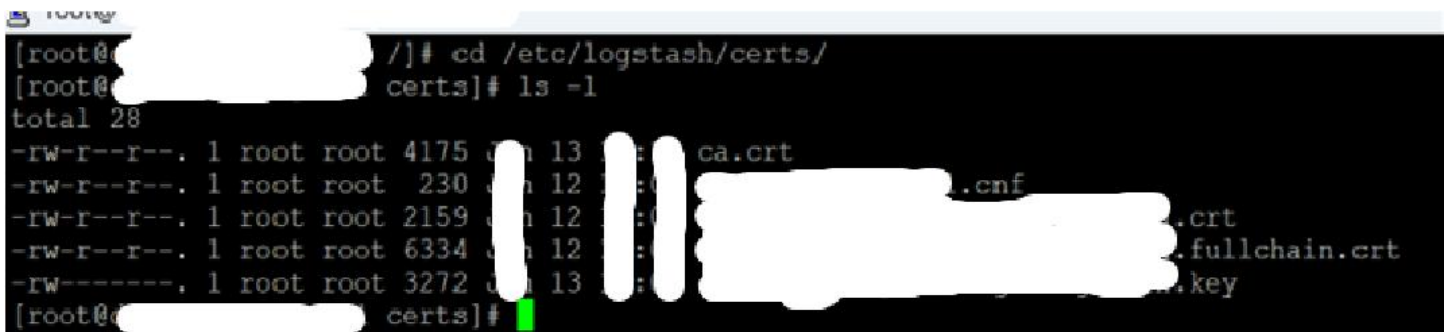
```
mkdir /logstash  
cd /logstash/  
wget https://artifacts.elastic.co/downloads/logstash/logstash-9.1.5-x86\_64.rpm  
rpm -ivh logstash-9.1.5-x86_64.rpm
```



```
[root@redhat ~]# cd /logstash  
[root@redhat logstash]# ls -l  
total 418676  
-rw-r--r--. 1 root root 428720333 1 logstash-9.1.5-x86_64.rpm  
[root@redhat logstash]#
```

5. make certs folder and copy all important certificate

```
/etc/logstash  
mkdir certs
```



```
[root@redhat ~]# cd /etc/logstash/certs/  
[root@redhat certs]# ls -l  
total 28  
-rw-r--r--. 1 root root 4175 Jan 13 12:00 ca.crt  
-rw-r--r--. 1 root root 230 Jan 12 12:00 .cnf  
-rw-r--r--. 1 root root 2159 Jan 12 12:00 .crt  
-rw-r--r--. 1 root root 6334 Jan 12 12:00 fullchain.crt  
-rw-r--r--. 1 root root 3272 Jan 13 12:00 .key  
[root@redhat certs]#
```

```
cd /etc/logstash/certs/
```

6. changer owner and directory permission

```
chown -R logstash:logstash /etc/logstash /usr/share/logstash  
/var/log/logstash /var/lib/logstash
```

7. Go to config directory and create conf file as per requirement

```
/etc/logstash/conf.d  
nano output.conf
```

```
[root@redhat certs]# cd /etc/logstash/conf.d  
[root@redhat conf.d]# ls -l  
total 4  
-rw-r--r--. 1 root root 565 [redacted] output.conf  
[root@redhat conf.d]# nano output.conf  
GNU nano 5.6.1 output.conf  
input {  
  beats {  
    port => 5044  
  }  
}  
  
filter {  
  # empty or transformations can go here  
}  
  
output {  
  elasticsearch {  
    hosts => [  
      "https://[redacted]:9200",  
      "https://[redacted]:9200",  
      "https://[redacted]:9200"  
    ]  
    ssl_enabled => true  
    ssl_certificate_authorities => ["/etc/logstash/certs/ca.crt"]  
    ssl_certificate => "/etc/logstash/certs/[redacted] fullchain.crt"  
    ssl_key => "/etc/logstash/certs/[redacted].key"  
    user => "elastic"  
    password => "elastic@123"  
  }  
}
```

8. Start the port

```
firewall-cmd --permanent --add-port=5044/tcp  
systemctl reload firewalld  
sudo firewall-cmd --list-ports
```

9.Restart logstash service and check the status

```
sudo systemctl daemon-reload  
sudo systemctl enable logstash
```

```
sudo systemctl start logstash  
sudo systemctl restart logstash  
sudo systemctl status logstash
```

**Note :- Please put Certificate and other details as per host and server.
Same process will be followed for the other logstash node also.**

Kibana Installation

1. Enable subscription-manager register

```
subscription-manager register --username ABC --password abc@123
```

Note :- After installation please subscription-manager unregister

Command :- subscription-manager unregister

2. Install java on node (server), check the java version and Install it

```
yum install java*  
java -version
```

3. Make Kibana as user and group

```
useradd elasticsearch  
groupadd elasticsearch
```

4. Create Kibana Directory for installation work

```
mkdir /kibana  
cd /kibana/  
wget https://artifacts.elastic.co/downloads/kibana/kibana-9.1.5-x86_64.rpm  
yum install net-tools  
rpm -ivh kibana-9.1.5-x86_64.rpm
```

```
-rw-----, 1 root root 1408 [redacted] anaconda-ks.cfg  
[root@[redacted] ~]# cd /kibana/  
[root@[redacted] kibana]# ls -ltr  
total 359852  
-rw-r--r--, 1 root root 368484602 [redacted] kibana-9.1.5-x86_64.rpm  
[root@[redacted] kibana]#
```

5. put all certificates in certs directory

```
[root@[redacted] kibana]# cd /etc/kibana/certs/  
[root@[redacted] certs]# ls -l  
total 28  
-rwxrwxr-x, 1 root kibana 4175 [redacted] ca.crt  
-rwxrwxr-x, 1 root kibana 229 [redacted] cnf  
-rwxrwxr-x, 1 root kibana 2155 [redacted] .crt  
-rwxrwxr-x, 1 root kibana 6330 [redacted] .fullchain.crt  
-rwxrwxr-x, 1 root kibana 3272 [redacted] key  
[root@[redacted] certs]# pwd  
/etc/kibana/certs
```

```
cd /etc/kibana/certs/
```

6. make changes in kibana.yml file

```
# Enables you to specify a file where Kibana stores log output.
logging:
  appenders:
    file:
      type: file
      fileName: /var/log/kibana/kibana.log
      layout:
        type: json
  root:
    appenders:
      - default
      - file
# policy:
```

```
server.host: "[redacted]"
server.name: "[redacted]"
server.publicBaseUrl: "http://[redacted]:5601"

server.ssl.enabled: true
server.ssl.certificate: /etc/kibana/certs/[redacted].fullchain.crt
server.ssl.key: /etc/kibana/certs/[redacted].key

elasticsearch.hosts: ["https://[redacted]:9200","https://[redacted]:9200","https://[redacted]:9200"]
elasticsearch.ssl.certificateAuthorities: ["/etc/kibana/certs/ca.crt"]
elasticsearch.ssl.verificationMode: full
elasticsearch.username: "kibana system"
elasticsearch.password: "[redacted]"
# xpack.encryptedSavedObjects.encryptionKey: "[redacted]"
```

Note : As per node, cluster and server host name will be different

7. open port and reload

```
firewall-cmd --permanent --add-port=5601/tcp
systemctl reload firewalld
netstat -an | grep 5601
```

8. Give permission to kibana respect folders

```
chown -R kibana:kibana /etc/kibana/certs
chown -R kibana:kibana /opt/kibana
chown -R kibana:kibana /var/log/kibana
```

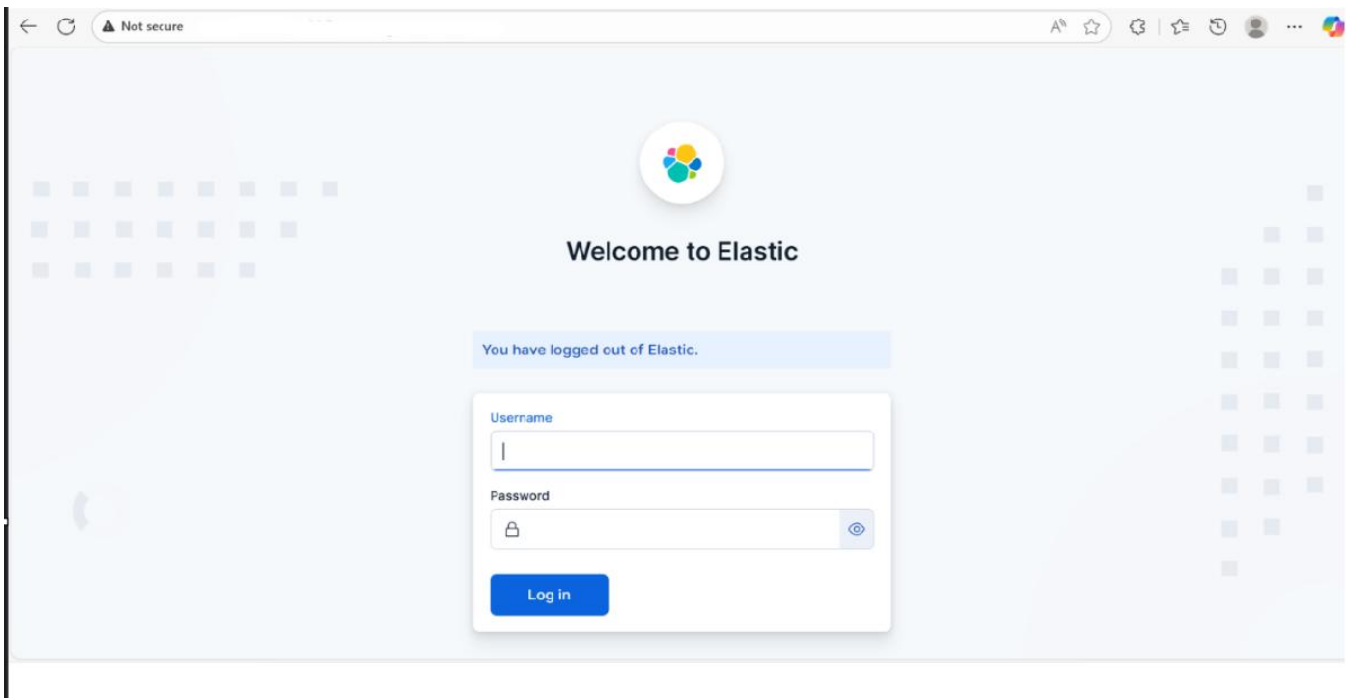
9. restart kibana service

```
sudo systemctl daemon-reload
sudo systemctl enable kibana
sudo systemctl start kibana
sudo systemctl restart kibana
sudo systemctl status kibana
```

10. Command to reset kibana_system user password from elasticsearch master node.

```
Cd /usr/share/elasticsearch
Sudo ./bin/elasticsearch-reset-password -u kibana_system
```

11. Now go to search engine and access kibana dashboard or url



<https://kibanalocalhost:5601>

Enter elastic and its password

Username :- elastic

Password :- custom or auto generated.

Note: The same process will be followed for the other Kibana node.

APM Installation on ELK setup

1. Enable subscription-manager register

```
subscription-manager register --username actual_user --password
actual_pass
```

Note :- After installation please subscription-manager unregister

Command :- subscription-manager unregister

2. Install java on node (server), check the java version and Installation it

```
yum install java*
java -version
```

3. make apm-server as user

```
useradd apm-server
groupadd apm-server
```

4. Create APM directory and install apm-server

```
mkdir /apm
cd /apm/
wget https://artifacts.elastic.co/downloads/apm-server/apm-server-9.1.5-
x86_64.rpm
rpm -ivh apm-server-9.1.5-x86_64.rpm
```

5. Take backup of yml file and make changes

```
cd /etc/apm-server/  
cp apm-server.yml apm-server.yml.backup  
nano apm-server.yml
```

6. Put certificate on certs location under apm directory

```
cd /etc/apm-server/certs/
```

```
[root@redhat ~]# cd /etc/apm-server/certs/
[root@redhat certs]# ls -l
total 28
-rw-r--r--. 1 apm-server apm-server 4175 [redacted].cnf
-rw-r--r--. 1 apm-server apm-server 229 [redacted].key
-rw-r--r--. 1 apm-server apm-server 2155 [redacted].crt
-rw-r--r--. 1 apm-server apm-server 6330 [redacted].crt
-rw-----. 1 apm-server apm-server 3268 [redacted].key
[root@redhat certs]#
```

7. Set permission for apm folders

```
chown -R apm-server:apm-server /etc/apm-server/
chown -R apm-server:apm-server /var/log/apm-server/
```

8. Make changes in apm conf yml file

```
cd /etc/apm-server/
```

nano apm-server.yml

```
GNU nano 2.8.1 apm-server.yml
# =====
# APM Server Configuration
# =====

apm-server:
  host: "0.0.0.0:8200"
  ssl.enabled: true
  ssl.certificate: /etc/apm-server/certs/ cert
  ssl.key: /etc/apm-server/certs/ key

# =====
# Elasticsearch Output
# =====

output.elasticsearch:
  hosts: ["https://", "https://", "https://"]
  username: "elastic"
  password: "elastic@123"
  ssl.certificate_authorities: ["/etc/apm-server/certs/"]
  ssl.verification_mode: full

# =====
# Kibana Integration
# =====

setup.kibana:
  hosts: ["https://", "https://"]
  ssl.enabled: true
  ssl.certificate_authorities: ["/etc/apm-server/certs/"]

# =====
# Security / Monitoring
# =====

monitoring.enabled: true
monitoring.cluster_uid: auto
monitoring.elasticsearch:
  hosts: ["https://", "https://"]
  username:
  password:

monitoring.enabled: true
monitoring.cluster_uid: auto
monitoring.elasticsearch:
  hosts: ["https://", "https://"]
  username:
  password:
  ssl.certificate_authorities: ["/etc/apm-server/certs/"]

# =====
# Logging
# =====

logging.to_files: true
logging.files:
  path: /var/log/apm-server
  name: apm-server
  keepfiles: 7
  permissions: 0644
logging.level: debug
```

Save yml file

9. Open port 8200 for apm

```
sudo firewall-cmd --zone=public --add-port=8200/tcp --permanent
sudo firewall-cmd --reload
sudo firewall-cmd --list-ports
firewall-cmd --list-all
```

Note:- The same process will be followed for other apm node.

10. Put ca certificate

```
cp ca.crt /etc/pki/ca-trust/source/anchors/
```

Note :- copy ca certificate from elastic master location and copy it on apm server given location

11. Start apm service and check the status

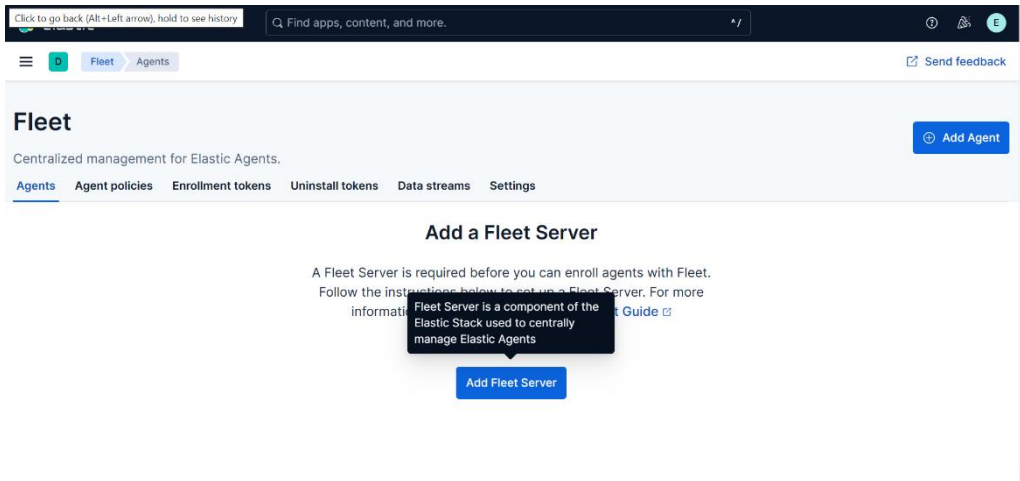
```
systemctl daemon-reload
systemctl status apm-server.service
systemctl enable apm-server.service --now
journalctl -xe -f -u apm-server.service --no-pager
journalctl -xe -f apm-server.service --no-pager
journalctl -xe -f -u apm-server.service --no-pager
```

Fleet Installation in Elastic Cluster

1. Go to Fleet Server and Enter All ELK Host entry in host file

```
nano /etc/hosts
```

2. Copy ca.crt certificate in certificate directory



```
Cd /etc/pki/ca-trust/source/anchors/
```

```
Copy certificate here and run update-ca-trust command.
```

3. Make fleet directory and go inside fleet and install elastic-agent rpm here

Add a Fleet Server

A Fleet Server is required before you can enroll agents with Fleet. Follow the instructions below to set up a Fleet Server. For more information, see the [Fleet and Elastic Agent Guide](#)

Quick Start

Advanced

1 Create a policy for Fleet Server

Fleet Server runs on Elastic Agent, and agents are enrolled in agent policies which represent hosts. We'll need to create a dedicated agent policy for Fleet Server to run on dedicated hosts.

Create policy

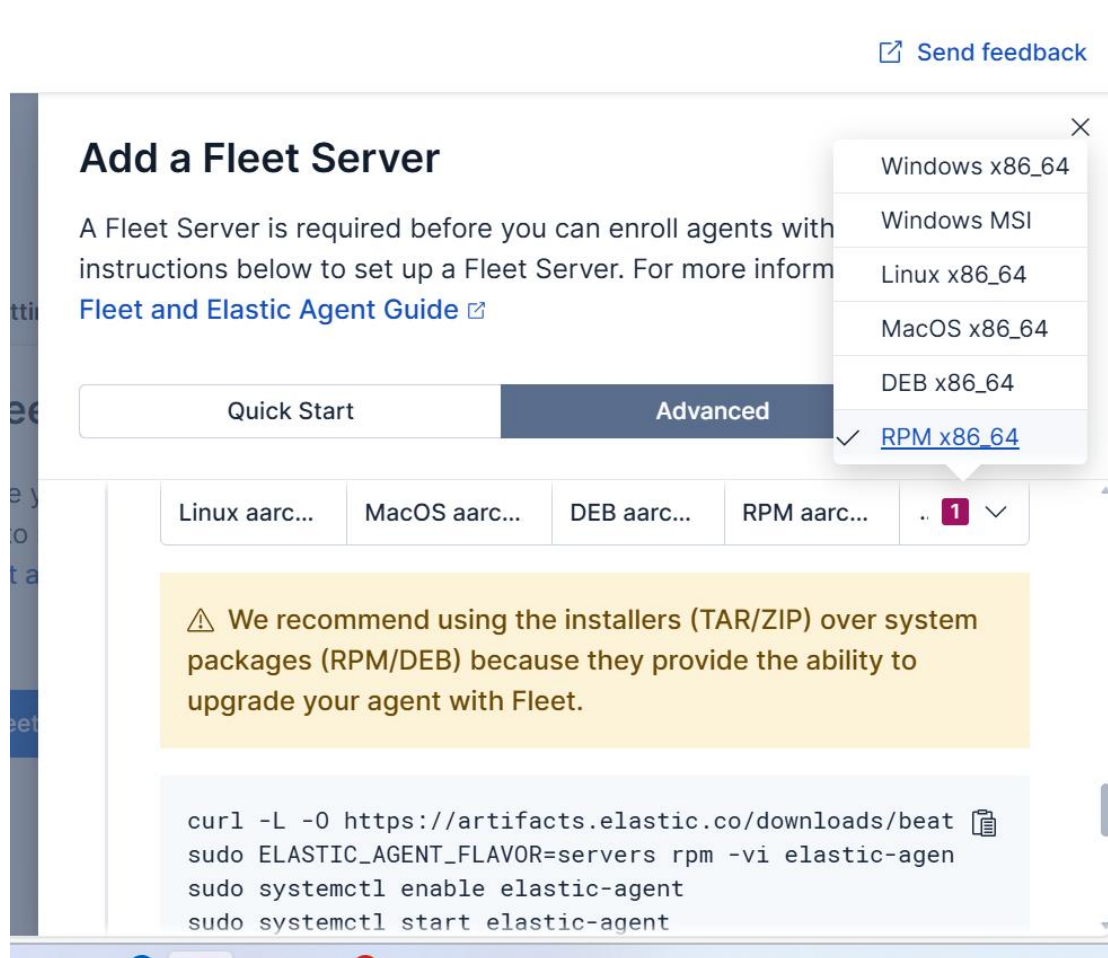
☒ Collect system logs and metrics ⓘ

> Advanced options

3. Go to Kibana and go to fleet section via 3 line in left side and add fleet.

5. select or click to add fleet server and under advanced section give Policy name as per our requirements can give custom any name

NOTE :- Follow the step which shows after creating policy to install fleet. To install fleet we need to install elastic-agent on server it automatically installs fleet on that particular server.



1. Select rpm or tar option for Installation of elastic-agent on our server and copy that installation command and run in our /fleet directory of server

```
[root@elastic-agent fleet]# curl -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-9.1.5-linux-x86_64.tar.gz
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
Dload  Upload   Total    Spent    Left    Speed
100 445M 100 445M    0     0 11.6M      0  0:00:38  0:00:38 --:--:-- 12.9M
[root@elastic-agent fleet]# tar xzvf elastic-agent-9.1.5-linux-x86_64.tar.gz
elastic-agent-9.1.5-linux-x86_64/.elastic-agent.active.commit
elastic-agent-9.1.5-linux-x86_64/data/elastic-agent-7bee01/otelcol
elastic-agent-9.1.5-linux-x86_64/data/elastic-agent-7bee01/elastic-agent
```

2. After installing elastic-agent bring or copy all respect certificate on fleet server under elastic-agent directory

```
curl -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-9.1.5-x86_64.rpm
sudo ELASTIC_AGENT_FLAVOR=servers rpm -vi elastic-agent-9.1.5-x86_64.rpm
sudo systemctl enable elastic-agent
sudo systemctl start elastic-agent

sudo elastic-agent enroll --url=https://elastic-agent-fleet-server:8220 \
--fleet-server-es=https://elastic-agent-fleet-server:9200 \

--fleet-server-service-token=elastic-agent-fleet-server-token \
RnZDlicEFydW \
--fleet-server-policy=fleet-server-policy \
--certificate-authorities=/etc/elastic-agent/certs/elastic-agent-fleet-server-ca.pem \
--fleet-server-es-ca=/etc/elastic-agent/certs/elastic-agent-fleet-server-ca.pem \
--fleet-server-cert=/etc/elastic-agent/certs/elastic-agent-fleet-server-cert.pem \
--fleet-server-cert-key=/etc/elastic-agent/certs/elastic-agent-fleet-server-cert-key.pem \
--fleet-server-port=8220
```

```
[root@elastic-agent fleet]# cd elastic-agent-9.1.5-linux-x86_64
[root@elastic-agent elastic-agent-9.1.5-linux-x86_64]#
```

cd /etc/elastic-agent

mkdir certs

cd /etc/elastic-agent/certs

Give proper permission to certificate and certs directory to.

```

[certs]# scp root@[redacted] /opt/CA/elastic/[redacted].* /etc/elastic-agent/certs
password:
please try again.
password:

100% 3268 49.8KB/s 00:00
100% 229 3.3KB/s 00:00
100% 2155 33.3KB/s 00:00
100% 6330 97.6KB/s 00:00

[certs]# scp root@[redacted] /opt/CA/ca.crt* /etc/elastic-agent/certs
ca.crt
100% 4175 66.4KB/s 00:00

[certs]# ls -l
total 28
-rw-r--r-- 1 root root 4175 [redacted] crt
-rw-r--r-- 1 root root 229 [redacted] cnf
-rw-r--r-- 1 root root 2155 [redacted] crt
-rw-r--r-- 1 root root 6330 [redacted] crt
-rw-r--r-- 1 root root 3268 [redacted] y

[certs]# chmod -R 775 /etc/elastic-agent/certs
bash: chmod: command not found...
Similar command is: 'chmod'

[certs]# chmod -R 775 /etc/elastic-agent/certs
[certs]# ls -l
total 28
-rwxrwxr-x 1 root root 4175 [redacted] crt
-rwxrwxr-x 1 root root 229 [redacted] cnf
-rwxrwxr-x 1 root root 2155 [redacted] crt
-rwxrwxr-x 1 root root 6330 [redacted] crt
-rwxrwxr-x 1 root root 3268 [redacted] y

```

3. Open port 8220 for elastic-agent and fleet

4. Run installation or enroll command which is given in kibana UI for elastic-agent

```

[redacted fleet]# firewall-cmd --permanent --add-port=8220/tcp
firewall-cmd --permanent --add-port=8220/udp
firewall-cmd --reload
success
success
success

```

5. Check status on both UI and CLI. It should be **successfully enroll elastic-agent**

```

Unknown command verb daemon-re.
[redacted certs]# systemctl daemon-reload
[redacted certs]# sudo elastic-agent enroll --url=https://[redacted] 8220 \
--fleet-server-es=https://[redacted] 9200 \
--fleet-server-service-token=[redacted] \
--fleet-server-policy=fleet-server-policy \
--certificate-authorities=/etc/elastic-agent/certs/ca.crt \
--fleet-server-es-ca=/etc/elastic-agent/certs/ca.crt \
--fleet-server-cert=/etc/elastic-agent/certs/[redacted].crt \
--fleet-server-cert-key=/etc/elastic-agent/certs/[redacted].key \
--fleet-server-port=8220

```

Note :- same steps we need to follow for another 2nd node of fleet instead of default fleet we need to give respect fleet url.

Add new fleet server host and **Rest of the steps remaining same.**



Find apps, content, and more.



FleetAgents

Send feedback

Fleet

Centralized management for Elastic Agents.

AgentsAgent policiesEnrollment tokensUninstall tokensData streamsSettings

Agent activity

Add Fleet Server

Add agent

fleet-agents.policy_id : fleet-server-policy

Status 6Tags 1Agent policy 1Upgrade available

Showing 2 agents

Clear filters

Healthy 2Unhealthy 0Orphaned 0Updating 0Offline 0Inactive 0Unenrolled 0Uninstalled 0

☐	Status	Host	Agent policy	CPU	Memory	Last activity	Version	Actions
☐	Healthy	10.10.10.10	Fleet_Server_Policy_1 rev. 1	N/A	480 MB	22 seconds ago	9.1.5	...
☐	Healthy	10.10.10.11	Fleet_Server_Policy_1 rev. 1	1.52 %	570 MB	39 seconds ago	9.1.5	...

Note:- The same process will be followed for the other fleet node also.