Extended Detection Response

Ubuntu Installation –

Try to use mirror address as: http://mirror.sg.gs/ubuntu/

XDR monitors data in an enterprise's technology environment, from endpoint devices and firewalls to cloud and some third-party applications. XDR identifies incidents and threats across the environment and collates related occurrences, optimizing the number of security alerts and allowing security teams to understand a cyberattack more clearly.

Key benefits of XDR

- Increased visibility
- Alert Management
- Incident prioritization
- Automated tasks
- Increased efficiency
- Real-time threat detection
- An integrated response across multiple security tools

System Requirements

- Ubuntu server 20.04
- RAM 128GB
- Minimum 1TB SSD

Tools Requirement:

- Elastic Search
- Kibana

Elastic Installation Script:

Elasticsearch Installation (Script-make sure you install unzip first, and script tags can be ignored if wanted to install manually)

```
#Install Java#
read -p "Enter username:  " username
read -p "Enter Machine IP:  " machineIP
sudo apt update
sudo apt-get install openjdk-8-jdk -y

#Add Elastic Repo#
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee –a /etc/apt/sources.list.d/elastic-7.x.list

#Install apt-transport-https package#
sudo apt-get install apt-transport-https unzip
sudo apt-get update

#Install Elasticsearch
sudo apt-get install elasticsearch -y
sudo systemctl enable elasticsearch

#ask Elasticsearch cluster name#
read -p "Enter cluster name: " cluster_name

# Elasticsearch configuration#
#cat > /usr/share/elasticsearch/instances.yml
sudo cat >> /usr/share/elasticsearch/instances.yml <<EOF
instances:
  - name: "elasticsearch"
    ip:
      - "$machineIP"
  - name: "kibana"
    ip:
      - "$machineIP"
```

```
EOF

sudo echo "cluster.name: "$cluster_name"" >> /etc/elasticsearch/elasticsearch.yml
sudo echo "network.host: "$machineIP"" >> /etc/elasticsearch/elasticsearch.yml
sudo cat >> /etc/elasticsearch/elasticsearch.yml <<EOF
http.port: 9200
discovery.type: single-node
xpack.security.enabled: true
xpack.security.authc.api_key.enabled: true

#xpack.security.transport.ssl#
xpack.security.transport.ssl.enabled: true
xpack.security.transport.ssl.verification_mode: certificate
xpack.security.transport.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
xpack.security.transport.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
xpack.security.transport.ssl.certificate_authorities: ["/etc/elasticsearch/certs/ca/ca.crt"]

#xpack.security.http.ssl#

xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.verification_mode: certificate
xpack.security.http.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
xpack.security.http.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
xpack.security.http.ssl.certificate_authorities: ["/etc/elasticsearch/certs/ca/ca.crt"]

EOF

cat >> /etc/elasticsearch/jvm.options <<EOF
-Xms2g
-Xmx2g
EOF

#create certificates#
#cd /usr/share/elasticsearch/
#sudo mkdir -p /usr/share/elasticsearch/ca/
#sudo /usr/share/elasticsearch/bin/elasticsearch-certutil ca --pem --out ./elastic-stack-ca.zip
#sudo unzip /usr/share/elasticsearch/elastic-stack-ca.zip
#sudo /usr/share/elasticsearch/bin/elasticsearch-certutil cert --ca-cert ca/ca.crt --ca-key ca/ca.key --pem --in instances.yml  --out certs.zip

#unzip certs#
sudo unzip /usr/share/elasticsearch/certs.zip
sudo mkdir -p ./certs/
sudo mv /usr/share/elasticsearch/elasticsearch/* ./certs/
sudo mv /usr/share/elasticsearch/kibana/* ./certs/
sudo mkdir -p /etc/kibana/certs/ca/
sudo mkdir -p /etc/elasticsearch/certs/ca/

#Copy certificates#
sudo cp /usr/share/elasticsearch/ca/ca.* /etc/elasticsearch/certs/ca/
sudo cp /usr/share/elasticsearch/ca/ca.* /etc/kibana/certs/ca/
sudo cp /usr/share/elasticsearch/certs/elasticsearch.* /etc/elasticsearch/certs/
sudo cp /usr/share/elasticsearch/certs/kibana.* /etc/kibana/certs/
sudo cp /usr/share/elasticsearch/ca/ca.crt /
sudo rm -r /usr/share/elasticsearch/elasticsearch/ /usr/share/elasticsearch/kibana/

#Change ownership#
cd /usr/share/
sudo chown -R elasticsearch:elasticsearch elasticsearch/
sudo chown -R elasticsearch:elasticsearch /etc/elasticsearch/
cd /usr/share/elasticsearch/
sudo chown -R elasticsearch:elasticsearch certs/
sudo chown -R elasticsearch:elasticsearch ca/

#Elasticsearch service#
sudo systemctl start elasticsearch
sudo systemctl status elasticsearch
```

#Create passwords
cd /usr/share/elasticsearch/
#sudo touch /home/$username/passwords.txt
#read -p "Press y to create password" password
#echo $password
#sudo ./bin/elasticsearch-setup-passwords auto 1> /home/$username/passwords.txt
#kibana_password = `cat /home/$username/passwords.txt | grep "PASSWORD kibana_system" | cut -d '=' -f 2`

#Install Kibana
sudo apt-get install kibana -y

cat >> /etc/kibana/kibana.yml <<EOF
server.host: $machineIP
server.port: 5601
server.publicBaseUrl: https://$machineIP
elasticsearch.hosts: ["https://$machineIP:9200"]

elasticsearch.username: "kibana_system"
elasticsearch.password: "$kibana_password"

server.ssl.enabled: true
server.ssl.certificate: "/etc/kibana/certs/kibana.crt"
server.ssl.key: "/etc/kibana/certs/kibana.key"

xpack.encryptedSavedObjects.encryptionKey: 3c7cd13abcc677fff24c49755a3883ce
xpack.reporting.encryptionKey: a74e79eb9b8b3ac83acc4ae6091b1689
xpack.security.encryptionKey: 18856156b26c268d3800a60961e10817
xpack.security.session.idleTimeout: "30m"

elasticsearch.ssl.certificateAuthorities: ["/etc/kibana/certs/ca/ca.crt"]
elasticsearch.ssl.certificate: "/etc/kibana/certs/kibana.crt"
elasticsearch.ssl.key: "/etc/kibana/certs/kibana.key"

EOF

#Kibana service#
sudo systemctl enable kibana
sudo systemctl start kibana
sudo systemctl status kibana

script ends here….
**********Now commands related to password generation, needs to be executed manually**********
systemctl stop kibana
sudo ./bin/elasticsearch-setup-passwords auto 1> /home/passwords.txt
press “q” and hit enter. Password file will be generated. And add kibana_system password in kibana.yml file and restart services.

Once the Elasticsearch & kibana is installed we must change the Fleet setting on Kibana console on select the endpoint integrations policy:

Step 1: Select the Endpoint Integration Policy
Login into Kibana>Menu Option >Management>Integration>Endpoint Security>Add Integration > Enable all settings here >save
Integration>Endpoint Security>Assets>Hosts>Select Integration>agent>redirect on fleet page – deploy fleet server and then add host.

Step 2: Change the fleet setting
Login into Kibana>Menu Option > Fleet > fleet setting >
Select - quickstart
fleet server - http://machine IP:8220
elastic https://machine IP:9200

Step 3: Download the agent - linux 64 - to create the server , below are the download address:
wget https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-7.17.6-linux-x86_64.tar.gz

tar xvf elastic (unzip the file)
cd elastic

Step 4: Token generate & need to run from master machine
generate token from kibana> to create server on default fleet server policy--> generate new token and replace in below command along with policy number

sudo ./elastic-agent install --fleet-server-es=https://machine ip:9200 --fleet-server-service-token=<token> --fleet-server-policy=<policy> --fleet-server-insecure-http --fleet-server-es-ca=/etc/elasticsearch/certs/ca/ca.crt

yes

run this command in the folder only. Go to Elastic > fleet - server must be added and click continue. let it update and be healthy.

Agent Installation for windows machine:

Now go to kibana > fleet > add agent > window > download and store in one folder on desktop, also copy ca.crt in same folder. then generate token while adding agent and policy number in below command

Step 1: Download the agent file from Kibana

Step 2: Copy the elasticsearch ca.crt file to the windows machine

copy /etc/elasticsearch/ca/ca.crt file in windows.

Step 3: Import the certificate

Import-Certificate -FilePath .\certificate path\ca.crt -CertStoreLocation 'Cert:\LocalMachine\Root' -Verbose

Get-ChildItem Cert:\LocalMachine\Root\ | Where-Object { $_.Subject -like '*Elastic*'}

Step 4: Run the elastic agent on windows machine

.\elastic-agent.exe install --fleet-server-es=https://elastic IP:port --fleet-server-service-token=<token> --fleet-server-policy=<policy> --fleet-server-insecure-http --fleet-server-es-ca=C:\certification path\ca.crt

Now go to kibana>discover – all the logs collected by agent is displayed here in raw format.

Kibana > dashboard – can be created manually or choose one

Kibana>security > alerts – any rule match and alert triggered shown here

Kibana > security > rule – can be activated here or can create here