Installing ELK

(ElasticSearch , Logstash and Kibana)

On Ubuntu 14.04

This Documentation is completely based on the following links:

https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-logstash-and-kibana-elk-stack-on-ubuntu-14-04#install-elasticsearch

https://www.elastic.co/guide/en/elasticsearch/reference/1.7/\_installation.html

Step 1: Install Java 8

Add the Oracle Java PPA to apt

sudo add-apt-repository -y ppa:webupd8team/java

Update Package

sudo apt-get update

Install the Latest version of Java 8

sudo apt-get -y install oracle-java8-installer

Step 2: Install Elasticsearch

Install Elasticsearch with a package manager by adding Elastic’s package source list

wget -qO - https://packages.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

Create the Elasticsearch source list

echo "deb http://packages.elastic.co/elasticsearch/2.x/debian stable main" | sudo tee -a /etc/apt/sources.list.d/elasticsearch-2.x.list

Update Package

sudo apt-get update

Install Elasticsearch

sudo apt-get -y install elasticsearch

Edit the config file for Elasticsearch

sudo nano /etc/elasticsearch/elasticsearch.yml

NOTE: Edited nothing. However, as per the link referenced, we must use the following:

In order to restrict outside access to our Elasticsearch instance on port 9200 through the HTTP API,

We must find the line that specifies ‘network.host’, uncomment it and replace it’s value with ‘localhost’, So that it looks like:

network.host: localhost

Save and Exit the file.

Restart elasticsearch service

sudo service elasticsearch restart

To run elasticsearch service on bootup, use

sudo update-rc.d elasticsearch defaults 95 10

Step 3: Install Kibana

Install Kibana using a package manager by adding Elastic’s package source list

echo "deb http://packages.elastic.co/kibana/4.4/debian stable main" | sudo tee -a /etc/apt/sources.list.d/kibana-4.4.x.list

Update package

sudo apt-get update

Install Kibana

sudo apt-get -y install kibana

Edit the Kibana configuration file

sudo nano /opt/kibana/config/kibana.yml

Find the Line that specifies ‘server.host’, replace the IP Address with ‘localhost’

It must look like: server.host: "localhost"

Enable Kibana service and restart it

sudo update-rc.d kibana defaults 96 9

sudo service kibana start

Step 4: Install Nginx

Use apt to install Nginx and Apache2-utils

sudo apt-get install nginx apache2-utils

Create an adminuser called ‘kibanaadmin’ that can access the web interface

sudo htpasswd -c /etc/nginx/htpasswd.users kibanaadmin

Edit the Default block of Nginx server

sudo nano /etc/nginx/sites-available/default

Cut everything and paste the following: (Update Server’s name to your Server Name)

server {

listen 80;

server\_name example.com;

auth\_basic "Restricted Access";

auth\_basic\_user\_file /etc/nginx/htpasswd.users;

location / {

proxy\_pass http://localhost:5601;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection 'upgrade';

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

}

}

This enables Nginx to directly listen to your server’s HTTP Traffic to the Kibana application, which is listening on port localhost: 5601

Restart Nginx

sudo service nginx restart

Step 5: Install Logstash

Create Logstash source list

echo 'deb http://packages.elastic.co/logstash/2.2/debian stable main' | sudo tee /etc/apt/sources.list.d/logstash-2.2.x.list

Update Package

sudo apt-get update

Install Logstash

sudo apt-get install logstash

Generate SSL Certificates

sudo mkdir -p /etc/pki/tls/certs

sudo mkdir /etc/pki/tls/private

One Option for Generating SSL Certificate is:

Using URL of the Server over which ELK resides

Edit the configuration file:

sudo nano /etc/ssl/openssl.cnf

Look for ‘[ v3\_ca ]‘ and add this line under it:

subjectAltName = IP: ELK\_server\_private\_IP

Save and Exit

Generate SSL Certificates and private key now:

cd /etc/pki/tls

sudo openssl req -config /etc/ssl/openssl.cnf -x509 -days 3650 -batch -nodes -newkey rsa:2048 -keyout private/logstash-forwarder.key -out certs/logstash-forwarder.crt

Option 2: Via FQDN (This is not used by me)

Cnfigure Logstash

sudo nano /etc/logstash/conf.d/02-beats-input.conf

Insert the following input configuration:

* input {
* beats {
* port => 5044
* ssl => true
* ssl\_certificate => "/etc/pki/tls/certs/logstash-forwarder.crt"
* ssl\_key => "/etc/pki/tls/private/logstash-forwarder.key"
* }
* }

Save and Quit

Create another configuration file to add a filter for syslog messages

sudo nano /etc/logstash/conf.d/10-syslog-filter.conf

Insert the following filter cofiguration:

* filter {
* if [type] == "syslog" {
* grok {
* match => { "message" => "%{SYSLOGTIMESTAMP:syslog\_timestamp} %{SYSLOGHOST:syslog\_hostname} %{DATA:syslog\_program}(?:\[%{POSINT:syslog\_pid}\])?: %{GREEDYDATA:syslog\_message}" }
* add\_field => [ "received\_at", "%{@timestamp}" ]
* add\_field => [ "received\_from", "%{host}" ]
* }
* syslog\_pri { }
* date {
* match => [ "syslog\_timestamp", "MMM d HH:mm:ss", "MMM dd HH:mm:ss" ]
* }
* }
* }

Save and Quit

Create another configuration file to parse incoming syslogs to make it structured and query table:

sudo nano /etc/logstash/conf.d/30-elasticsearch-output.conf

Insert the following output configuration:

* output {
* elasticsearch {
* hosts => ["localhost:9200"]
* sniffing => true
* manage\_template => false
* index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"
* document\_type => "%{[@metadata][type]}"
* }
* }

Save and Exit

Test Logstash configuration

sudo service logstash configtest

It must return: Configuration OK

Restart Logstash and enable it to load on startup

sudo service logstash restart

sudo update-rc.d logstash defaults 96 9

Step 6: Load Kibana Dashboard

cd ~

curl -L -O <https://download.elastic.co/beats/dashboards/beats-dashboards-1.1.0.zip>

Install unzip to unzip the File contents

sudo apt-get -y install unzip

Extract the contents of the archive

unzip beats-dashboards-\*.zip

Load a sample Dashboard

cd beats-dashboards-\*

./load.sh

You can also use:

cd /usr/share/elasticsearch/bin

./elasticsearch

If the above command doesn’t work out, run the command as a root.

Use:

cd /usr/share/elasticsearch

bin/elasticsearch -Des.insecure.allow.root=true

This will display all errors if present.

For Visualizing Kibana Dashboard:

Visit: http://localhost:5601/app/kibana

For accessing the sample template:

Visit: http://127.0.0.1:9200/

One Error Obtained was resolved using this link: <https://github.com/elastic/ansible-elasticsearch/issues/58>