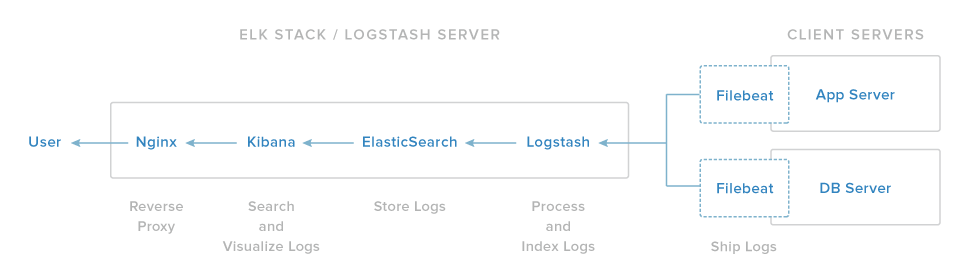
# **Elasticsearch, Logstash, and Kibana (ELK Stack) on CentOS 7**

**Server1: ELK**

**Our ELK stack setup has four main components:**

1. ****Logstash****: The server component of Logstash that processes incoming logs.
2. ****Elasticsearch****: Stores all of the logs.
3. ****Kibana****: Web interface for searching and visualizing logs, which will be proxied through Nginx.
4. ****Filebeat****: Installed on client servers that will send their logs to Logstash, Filebeat serves as a log shipping agent that utilizes the lumberjack networking protocol to communicate with Logstash.



We will install the first three components on a single server, which we will refer to as our ****ELK Server****. Filebeat will be installed on all of the client servers that we want to gather logs for, which we will refer to collectively as our ****Client Servers****.

**Step-1: Install Java 8**

**Download the Oracle Java 8**

wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u73-b02/jdk-8u73-linux-x64.rpm"

**Install Java**

yum -y localinstall jdk-8u73-linux-x64.rpm

**Delete the archive file that you downloaded earlier:**

rm ~/jdk-8u\*-linux-x64.rpm

**Step-2: Install Elasticsearch**

**Elasticsearch can be installed with a package manager by adding Elastic’s package repository.**

**Run the following command to import the Elasticsearch public GPG key into rpm:**

rpm --import http://packages.elastic.co/GPG-KEY-elasticsearch

**Create a new yum repository file for Elasticsearch. Note that this is a single command:**

echo '[elasticsearch-2.x]

name=Elasticsearch repository for 2.x packages

baseurl=http://packages.elastic.co/elasticsearch/2.x/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

' | sudo tee /etc/yum.repos.d/elasticsearch.repo

**Install Elasticsearch with this command:**

yum install elasticsearch -y

**Elasticsearch is now installed. Let’s edit the configuration:**

vim /etc/elasticsearch/elasticsearch.yml

Find the line that specifies network.host, uncomment it, and replace its value with “localhost”

network.host: localhost

systemctl start elasticsearch

systemctl enable elasticsearch

**Step-3: Install Kibana**

**Create and edit a new yum repository file for Kibana:**

vim /etc/yum.repos.d/kibana.repo

**Add the following repository configuration:**

[kibana-4.4]

name=Kibana repository for 4.4.x packages

baseurl=http://packages.elastic.co/kibana/4.4/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

**Install Kibana with this command:**

sudo yum -y install kibana

**Open the Kibana configuration file for editing:**

sudo vi /opt/kibana/config/kibana.yml

**In the Kibana configuration file, find the line that specifies server.host, and replace the IP address (“0.0.0.0” by default) with “localhost”:**

server.host: "localhost"

**Now start the Kibana service, and enable it:**

systemctl start kibana

chkconfig kibana on

**Step-4: Install Nginx**

**Add the EPEL repository to yum:**

yum install epel-release -y

**Now use yum to install Nginx and httpd-tools:**

yum install nginx httpd-tools -y

**Use htpasswd to create an admin user, called “kibanaadmin” (you should use another name), that can access the Kibana web interface:**

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

**Now open the Nginx configuration file in your favorite editor:**

vim /etc/nginx/nginx.conf

Find the default server block (starts with server {), the last configuration block in the file, and delete it. When you are done, the last two lines in the file should look like this:

include /etc/nginx/conf.d/\*.conf;

}

**Now we will create an Nginx server block in a new file:**

vim /etc/nginx/conf.d/kibana.conf

**The following code block into the file. Be sure to update the server\_name to match your server’s 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;

}

}

**Now start and enable Nginx to put our changes into effect:**

systemctl start nginx

systemctl enable nginx

**Step-5: Install Logstash**

**The Logstash package shares the same GPG Key as Elasticsearch, and we already installed that public key, so let’s create and edit a new Yum repository file for Logstash:**

vim /etc/yum.repos.d/logstash.repo

**Add the following repository configuration:**

[logstash-2.2]

name=logstash repository for 2.2 packages

baseurl=http://packages.elasticsearch.org/logstash/2.2/centos

gpgcheck=1

gpgkey=http://packages.elasticsearch.org/GPG-KEY-elasticsearch

enabled=1

**Install Logstash with this command:**

yum install logstash -y

**Logstash is installed but it is not configured yet.**

**Step-6: Generate SSL Certificates**

**Open the OpenSSL configuration file:**

vim /etc/pki/tls/openssl.cnf

**Find the [ v3\_ca ] section in the file, and add this line under it (substituting in the ELK Server’s private IP address):**

subjectAltName = IP: ELK\_server\_private\_ip

**Now generate the SSL certificate and private key in the appropriate locations (/etc/pki/tls/), with the following commands:**

cd /etc/pki/tls

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

**Step-7: Configure Logstash**

**Let’s create a configuration file called 02-beats-input.conf and set up our “filebeat” input:**

vim /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"

}

}

**Now let’s create a configuration file called 10-syslog-filter.conf, where we will add a filter for syslog messages:**

vim /etc/logstash/conf.d/10-syslog-filter.conf

**Insert the following syslog filter configuration:**

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" ]

}

}

}

**Create a configuration file called 30-elasticsearch-output.conf:**

vim /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]}"

}

}

**Test your Logstash configuration with this command:**

service logstash configtest

**Restart and enable Logstash to put our configuration changes into effect:**

systemctl restart logstash

chkconfig logstash on

**Step-8: Load Kibana Dashboards**

**Download the sample dashboards archive to your home directory:**

cd ~

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

**Install the unzip package with this command:**

yum install unzip -y

**Extract the contents of the archive:**

unzip beats-dashboards-\*.zip

**Load the sample dashboards, visualizations and Beats index patterns into Elasticsearch with these commands:**

cd beats-dashboards-\*

./load.sh

**Step-9: Load Filebeat Index Template in Elasticsearch**

**Download the Filebeat index template to your home directory:**

cd ~

curl -O https://gist.githubusercontent.com/thisismitch/3429023e8438cc25b86c/raw/d8c479e2a1adcea8b1fe86570e42abab0f10f364/filebeat-index-template.json

**Load the template with this command:**

curl -XPUT 'http://localhost:9200/\_template/filebeat?pretty' -d@filebeat-index-template.json

**If the template loaded properly, you should see a message like this:**

**Output:**

**{**

**"acknowledged" : true**

**}**

**Now that our ELK Server is ready to receive Filebeat data, let’s move onto setting up Filebeat on each client server.**

**Client Server2: Filebeat**

**Step-1: Set Up Filebeat (Add Client Servers)**

**Copy SSL Certificate**

**On your ELK Server, copy the SSL certificate—created in the prerequisite tutorial—to your Client Server (substitute the client server’s address, and your own login):**

scp /etc/pki/tls/certs/logstash-forwarder.crt [root@192.168.81.145:/tmp](mailto:root@192.168.81.145:/tmp)

**Now, on your Client Server, copy the ELK Server’s SSL certificate into the appropriate location (/etc/pki/tls/certs):**

mkdir -p /etc/pki/tls/certs

cp /tmp/logstash-forwarder.crt /etc/pki/tls/certs/

**Install Filebeat Package:**

**On Client Server, create run the following command to import the Elasticsearch public GPG key into rpm:**

rpm --import http://packages.elastic.co/GPG-KEY-elasticsearch

Create and edit a new yum repository file for Filebeat:

vim /etc/yum.repos.d/elastic-beats.repo

**Add the following repository configuration:**

[beats]

name=Elastic Beats Repository

baseurl=https://packages.elastic.co/beats/yum/el/$basearch

enabled=1

gpgkey=https://packages.elastic.co/GPG-KEY-elasticsearch

gpgcheck=1

**Install Filebeat with this command:**

yum install filebeat -your

Filebeat is installed but it is not configured yet.

**Configure Filebeat:**

**Now we will configure Filebeat to connect to Logstash on our ELK Server.**

**On Client Server, create and edit Filebeat configuration file:**

vim /etc/filebeat/filebeat.yml

We’ll modify the existing prospector to send secure and messages logs to Logstash. Under paths, comment out the - /var/log/\*.log file. This will prevent Filebeat from sending every .log in that directory to Logstash. Then add new entries for syslog and auth.log. It should look something like this when you’re done:

...

paths:

- /var/log/secure

- /var/log/messages

# - /var/log/\*.log

...

Then find the line that specifies document\_type:, uncomment it and change its value to “syslog”. It should look like this after the modification:

...

document\_type: syslog

...

Next, under the output section, find the line that says elasticsearch:, which indicates the Elasticsearch output section (which we are not going to use). Delete or comment out the entire Elasticsearch output section (up to the line that says logstash:).

Find the commented out Logstash output section, indicated by the line that says #logstash:, and uncomment it by deleting the preceding #. In this section, uncomment the hosts: ["localhost:5044"] line. Change localhost to the private IP address (or hostname, if you went with that option) of your ELK server:

### Logstash as output

logstash:

# The Logstash hosts

hosts: ["192.168.81.129:5044"]

ssl.certificate\_authorities: ["/etc/pki/tls/certs/logstash-forwarder.crt"]

Directly under the hosts entry, and with the same indentation, add this line:

bulk\_max\_size: 1024

Next, find the tls section, and uncomment it. Then uncomment the line that specifies certificate\_authorities, and change its value to ["/etc/pki/tls/certs/logstash-forwarder.crt"]. It should look something like this:

...

tls:

# List of root certificates for HTTPS server verifications

certificate\_authorities: ["/etc/pki/tls/certs/logstash-forwarder.crt"]

This configures Filebeat to use the SSL certificate that we created on the ELK Server.

Now start and enable Filebeat to put our changes into place:

systemctl start filebeat

systemctl enable filebeat

**Step-2: Test Filebeat Installation**

**On your ELK Server, verify that Elasticsearch is indeed receiving the data by querying for the Filebeat index with this command:**

curl -XGET 'http://localhost:9200/filebeat-\*/\_search?pretty'

You should see a bunch of output that looks like this:

Sample Output:

...

{

"\_index" : "filebeat-2016.01.29",

"\_type" : "log",

"\_id" : "AVKO98yuaHvsHQLa53HE",

"\_score" : 1.0,

"\_source":{"message":"Feb 3 14:34:00 rails sshd[963]: Server listening on :: port 22.","@version":"1","@timestamp":"2016-01-29T19:59:09.145Z","beat":{"hostname":"topbeat-u-03","name":"topbeat-u-03"},"count":1,"fields":null,"input\_type":"log","offset":70,"source":"/var/log/auth.log","type":"log","host":"topbeat-u-03"}

}

...

**Step-3: Firewall Update**

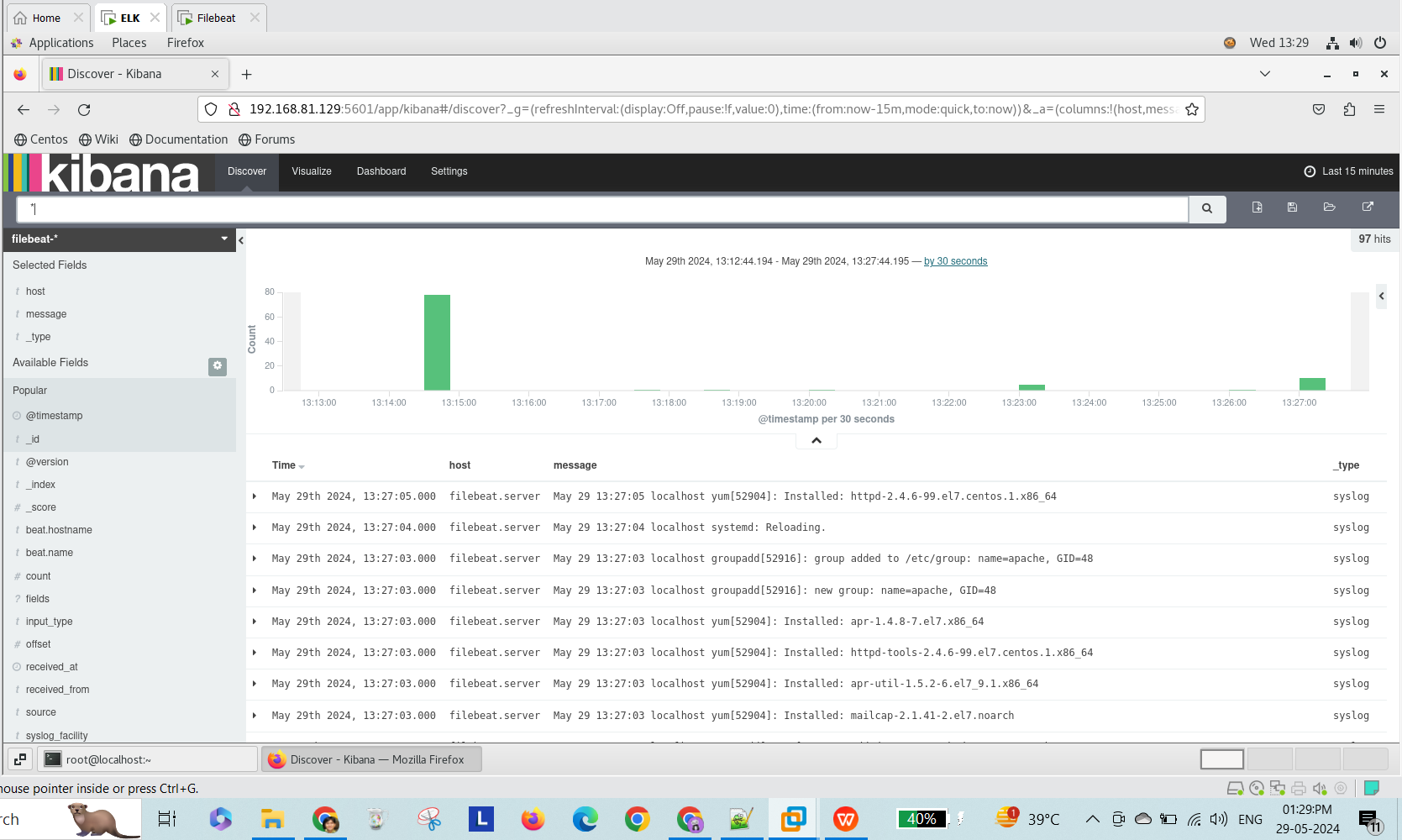
sudo firewall-cmd --add-port=9200/tcp --permanent

sudo firewall-cmd --add-port=5601/tcp --permanent

sudo firewall-cmd --add-port=5044/tcp --permanent

sudo firewall-cmd --reload

**Connect to Kibana:**



**Filebeat:**

