**ELK 5 on Ubuntu: Pt. 2 – Installing and Configuring Elasticsearch, Logstash, Kibana & Nginx**

**Preparing the Server for the Installs**

1.) Add the Elastic repository to Ubuntu:

rob@LinELK01:~$ **sudo wget -qO – https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key** **add –**

rob@LinELK01:~$ **sudo echo “deb https://artifacts.elastic.co/packages/5.x/apt stable main” | sudo tee** **-a /etc/apt/sources.list.d/elastic-5.x.list**

Reference: <https://www.elastic.co/guide/en/beats/libbeat/current/setup-repositories.html>

2.) Add the Java 8 repository to Ubuntu:

rob@LinELK01:~$ **sudo add-apt-repository -y ppa:webupd8team/java**

3.) Run apt-get update:

rob@LinELK01:~$ **sudo apt-get update**

4.) Install Java:

rob@LinELK01:~$ **sudo apt-get -y install oracle-java8-installer**

5.) We will need an SSL certificate later for Nginx, so let’s create a self-signed certificate now. First create

the directories for the private key and certificate:

rob@LinELK01:~$ **sudo mkdir -p /etc/pki/tls/certs**

rob@LinELK01:~$ **sudo mkdir /etc/pki/tls/private**

6.) Open the openSSL configuration file:

rob@LinELK01:~$ **sudo nano /etc/ssl/openssl.cnf**

7.) Find the **[ v3\_ca ]** section and add the following line to enable the generation of a self-signed certificate tied to an IP address:

* subjectAltName = IP: 192.168.2.85

Change the IP: **<address>** to the IP of the ELK server.

8.) Generate the certificate and key:

rob@LinELK01:~$ **cd /etc/pki/tls**

rob@LinELK01:~$ **sudo openssl req -config /etc/ssl/openssl.cnf -x509 -days 3650 -batch -nodes -** **newkey rsa:4096 -keyout private/ELK-Stack.key -out certs/ELK-Stack.crt**

**Installing Elasticsearch**

1.) Install Elasticsearch:

echo "deb http://packages.elastic.co/kibana/4.5/debian stable main" | sudo tee -a /etc/apt/sources.list

rob@LinELK01:~$ **sudo apt-get -y install elasticsearch**

**sudo update-rc.d kibana defaults 95 10**

2.) Open the elasticsearch.yml configuration file:

rob@LinELK01:~$ **sudo nano /etc/elasticsearch/elasticsearch.yml**

3.) Uncomment/edit the following line to lock access down to the localhost:

* network.host: localhost

4.) Restart the service and enable it to start with the server:

rob@LinELK01:~$ **sudo service elasticsearch restart**

rob@LinELK01:~$ **sudo systemctl enable elasticsearch**

5.) Verify Elasticsearch is running:

rob@LinELK01:~$ **wget http://localhost:9200/**

1 http://localhost:9200/

2 Resolving localhost (localhost)... 127.0.0.1

3 Connecting to localhost (localhost)|127.0.0.1|:9200... connected. 4 HTTP request sent, awaiting response... 200 OK

5 Length: 327 [application/json]

6 Saving to: ‘index.html’

And then cat the output of the index.html to view the contents:

rob@LinELK01:~$ **cat index.html**

**Installing Kibana**

1.) Install Kibana:

sudo 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

rob@LinELK01:~$ **sudo apt-get -y install kibana**

**sudo /etc/init.d/kibana start sudo update-rc.d kibana defaults**

2.) Open the kibana.yml configuration file:

rob@LinELK01:~$ **sudo nano /etc/kibana/kibana.yml**

3.) Uncomment/edit the following line to lock access down to the localhost:

1 server.host: "localhost"

4.) Restart the service and enable it to start with the server:

rob@LinELK01:~$ **sudo service kibana restart**

rob@LinELK01:~$ **sudo systemctl enable kibana**

5.) Verify Kibana is running by browsing to **http://localhost:5601/**.

**Installing Nginx**

1.) Install Nginx to be the proxy in front of Kibana and apache2-utils to help create the accounts used with the basic authentication:

rob@LinELK01:~$ **sudo apt-get install -y nginx apache2-utils**

2.) Create a user account for the basic authentication:

rob@LinELK01:~$ **sudo htpasswd -c /etc/nginx/htpasswd.users kibanaadmin**

|  |  |
| --- | --- |
| The kibanaadmin portion can be changed to whatever username is desired. After hitting enter, you should |  |
| then be prompted to create a password for the user. |  |

3.) Next wipe the nginx configuration file and then open it: rob@LinELK01:~$ **sudo truncate -s 0 /etc/nginx/sites-available/default** rob@LinELK01:~$ **sudo nano /etc/nginx/sites-available/default**

4.) Add the following to the configuration file:

server {

listen 80 default\_server; # Listen on port 80

server\_name 192.168.2.85; # Bind to the IP address of the server

return 301 https://$server\_name$request\_uri; # Redirect to 443/SSL

}

server {

listen 443 default ssl; # Listen on 443/SSL

# SSL Certificate, Key and Settings

ssl\_certificate /etc/pki/tls/certs/ELK-Stack.crt ;

ssl\_certificate\_key /etc/pki/tls/private/ELK-Stack.key;

ssl\_session\_cache shared:SSL:10m;

# Basic authentication using the account created with htpasswd

auth\_basic "Restricted Access";

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

location / {

# Proxy settings pointing to the Kibana instance

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;

}

}

5.) Restart Nginx and enable it to start with the server:

rob@LinELK01:~$ **sudo service nginx restart**

rob@LinELK01:~$ **sudo systemctl enable nginx**

6.) Verify Nginx is now working as a proxy for Kibana and the redirect to HTTPS and basic authentication work by browsing to **http://192.168.2.85/**.

**Installing Logstash**

1.) Install Logstash:

rob@LinELK01:~$ **sudo apt-get -y install logstash**

Manual Method –

Create and open the Beats input configuration file:

rob@LinELK01:~$ **sudo nano /etc/logstash/conf.d/02-beats-input.conf**

3.) Add the following to the configuration file:

input {

beats {

port => 5041

}

}

4.) Create and open the syslog filter configuration file:

rob@LinELK01:~$ **sudo nano /etc/logstash/conf.d/10-syslog-filter.conf**

5.) Add the following to the configuration file:

filter {

if [type] == "syslog" {

grok {

match => { "message" => "%{SYSLOGTIMESTAMP:syslog\_timestamp} %{SYSLOGHOST:sysl

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" ]

}

}

}

6.) Create and open the Elasticsearch output configuration file:

rob@LinELK01:~$ **sudo nano /etc/logstash/conf.d/30-elasticsearch-output.conf**

7.) Add the following to the configuration file:

output {

elasticsearch {

hosts => ["localhost:9200"]

sniffing => true

manage\_template => false

index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"

document\_type => "%{[@metadata][type]}"

}

}

Restart Logstash and enable it to start with the server:

rob@LinELK01:~$ **sudo service logstash restart**

rob@LinELK01:~$ **sudo systemctl enable logstash**

8.) Verify the service is running

rob@LinELK01:~$ systemctl status logstash

The Elk Stack should now be running and ready to receive data.

**Installing Filebeat on the ELK server**

1.) Install Filebeat:

rob@LinELK01:~$ **sudo apt-get install filebeat**

2.) Edit the filebeat configuration file:

rob@LinELK01:~$ **sudo nano /etc/filebeat/filebeat.yml**

3.) Edit or add additional paths to log files under the paths: section of the configuration file:

input\_type: log

# Paths that should be crawled and fetched. Glob based paths.

paths:

- /var/log/auth.log

- /var/log/syslog

#- /var/log/\*

#- c:\programdata\elasticsearch\logs\\*

You can add add more paths by adding an additional dash (-) followed by the path to the log. You can also use wildcards (\*) in the paths.

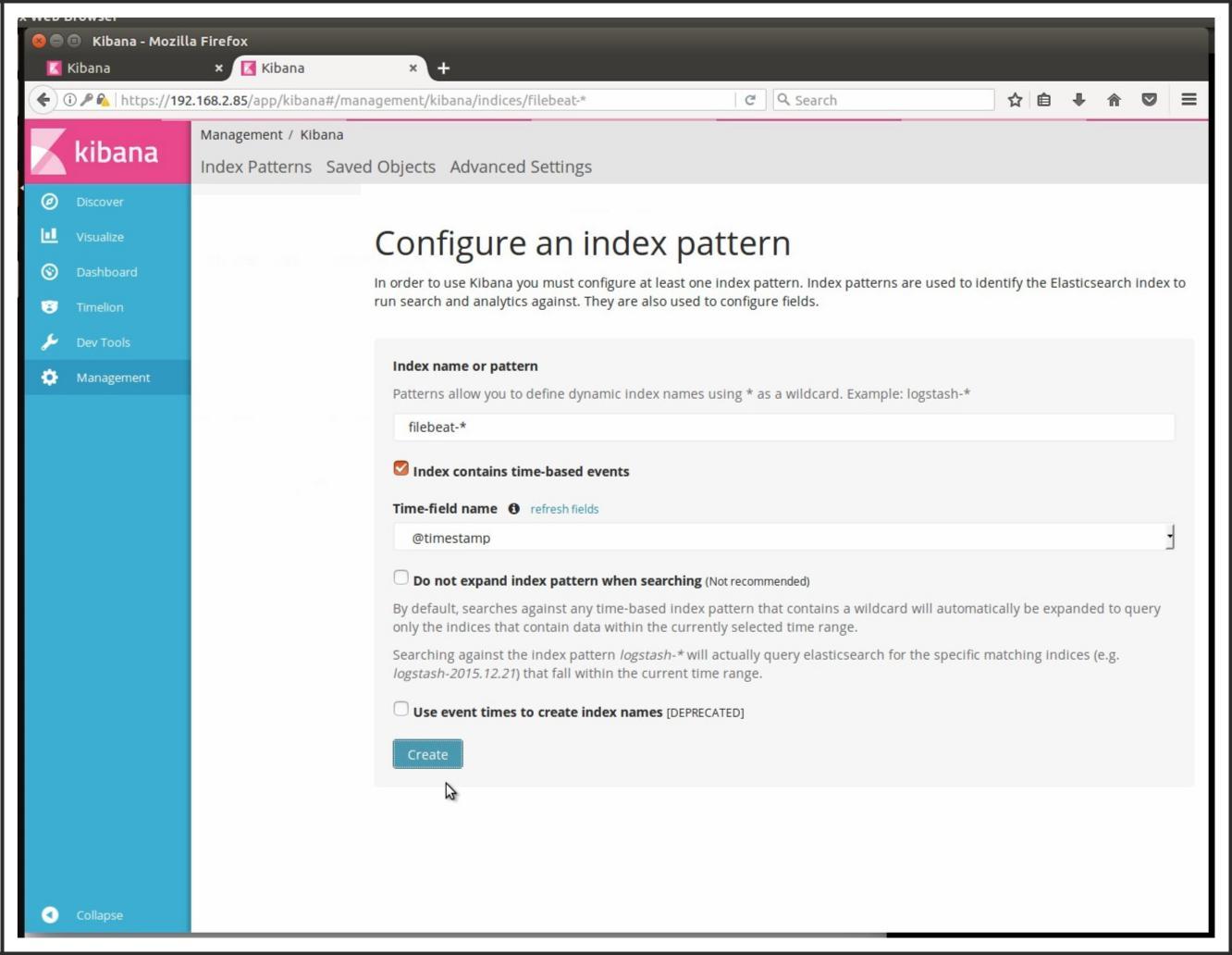
4.) Restart Logstash and enable it to start with the server:

rob@LinELK01:~$ **sudo service filebeat restart**

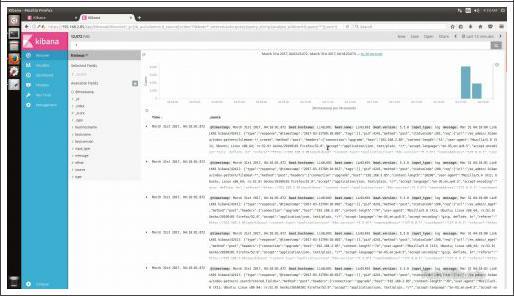
rob@LinELK01:~$ **sudo systemctl enable filebeat**

5.) Go back to Kibana and configure the index pattern for Filebeat:

**filebeat-\***



And now we are able to search against the filebeat index pattern and view the logs that filebeat is feeding the stack:



<http://robwillis.info/2017/04/elk-5-on-ubuntu-pt-2-installing-and-configuring-elasticsearch-logstash-kibana-nginx/>

<https://www.e2enetworks.com/help/knowledge-base/installing-elk-stack-on-ubuntu-14-04/>

<https://hostpresto.com/community/tutorials/install-and-configure-elk-stack-on-ubuntu-14-04/>



Comments are closed.

http://robwillis.info/2017/04/elk-5-on-ubuntu-pt-2-installing-and-configuring-elasticsearch-logstash-kibana-nginx/ 9/9