FreeBSD Administration - Support #437

Install an ElasticSearch, Fluentd, Kibana Stack on FreeBSD

08/13/2014 12:14 PM - Daniel Curtis

Status:	Closed	**Start date:**	07/10/2014
Priority:	Normal	**Due date:**	
Assignee:	Daniel Curtis	**% Done:**	100%
Category:	Logging Server	**Estimated time:**	6.00 hours
Target version:	FreeBSD 9	**Spent time:**	13.00 hours

Description

I've decided to centralize all the logs generated by a client's production systems to a syslog server and after assessing a bunch of products, I am now working with Fluentd.

The platform chosen to run Fluentd is FreeBSD inside a Jail (9.3-RELEASE at the time), a rock-solid and very well documented UNIX-like operating system. Besides, it also ships a production-ready ZFS implementation which always comes handy in the data center. FreeBSD-9.3 currently has the *sysutils/rubygem-fluentd* is available in the ports tree. This is a guide on how to install Fluentd with ElasticSearch and Kibana.

Setting up the Environment

- Start by making sure everything is up to date:

```
pkg update && pkg upgrade
portsnap fetch extract
```

- Set the default Ruby version to 2.2:

```
echo 'DEFAULT_VERSIONS= ruby=2.2' >> /etc/make.conf
```

- Install portmaster:

```
cd /usr/ports/ports-mgmt/portmaster
make install clean
pkg2ng
```

Install Fluentd

- Install rubygem-fluentd

```
portmaster sysutils/rubygem-fluentd
```

- Start and enable fluentd at boot:

```
echo 'fluentd_enable="YES"' >> /etc/rc.conf
service fluentd start
```

Installing Fluentd Plugins

We need a couple of plugins:

1. **out_elasticsearch**: this plugin lets Fluentd to stream data to Elasticsearch.
2. **outrecordreformer**: this plugin lets us process data into a more useful format.

- The following commands install both plugins:

```
fluent-gem install fluent-plugin-elasticsearch
fluent-gem install fluent-plugin-record-reformer
```

Add the Syslog configuration to Fluentd

Next, we configure Fluentd to listen to syslog messages and send them to Elasticsearch.

- Edit the fluentd config file:

```
vi /usr/local/etc/fluentd/fluent.conf
```

  - And add the following lines at the top of the file:

```
## Syslog input
<source>
 type syslog
 port 5140
 tag  system
</source>
<match system.*.*>
 type record_reformer
 tag elasticsearch
 facility ${tag_parts[1]}
 severity ${tag_parts[2]}
</match>
<match elasticsearch>
 type copy
 <store>
   type stdout
 </store>
 <store>
 type elasticsearch
 logstash_format true
 flush_interval 5s #debug
 </store>
</match>
```

- Restart fluentd:

```
service fluentd restart
```

  - (Optional) Start Fluentd with verbose debugging, run the following command:

```
fluentd -c /usr/local/fluentd/fluent.conf -vv &
```

Install ElasticSearch

- Install ElasticSearch:

```
portmaster textproc/elasticsearch
```

- Start and enable ElasticSearch at boot

```
echo 'elasticsearch_enable="YES"' >> /etc/rc.conf
service elasticsearch start
```

Securing Elasticsearch

- Up to version 1.2, Elasticsearch's dynamic scripting capability was enabled by default. Since this tutorial sets up the Kibana dashboard to be accessed from the public Internet, let's disable dynamic scripting by running the following line to append the parameter at the end of the ElasticSearch configuration file:

```
echo 'script.disable_dynamic: true' >> /usr/local/etc/elasticsearch/elasticsearch.yml
```

- Restart Elasticsearch:

```
service elasticsearch restart
```

Install Kibana

- Install kibana:

```
portmaster textproc/kibana
```

Configuring Kibana

Since Kibana will use port 80 to talk to Elasticsearch as opposed to the default port 9200, Kibana's config.js must be updated.

- Open Kibana configuration file and look for the following line:

```
vi /usr/local/www/kibana/config.js
```

    - And change the elasticsearch: "http://"+window.location.hostname+":9200", parameter to the following:

```
  elasticsearch: "http://"+window.location.hostname+":80",
```

Setting Up Kibana Dashboard Panels

Kibana's default panels are very generic, so it's recommended to customize them. Here, we show two methods.

Method 1: Using a Template

- The Fluentd team offers an alternative Kibana configuration that works with this setup better than the default one. To use this alternative configuration, run the following command:

```
sudo cp default.json /usr/local/kibana/app/dashboards/default.json
```

Note: The original configuration file is from the author's GitHub gist.

If you refresh your Kibana dashboard home page at your server's URL, Kibana should now be configured to show histograms by syslog severity and facility, as well as recent log lines in a table.

Method 2: Manually Configuring

Go to your server's IP address or domain to view the Kibana dashboard.

There are a couple of starter templates, but let's choose the blank one called **Blank Dashboard**.

Next, click on the **+ ADD A ROW** button on the right side of the dashboard. A configuration screen for a new row (a row consists of one or more panels) should show up. Enter a title, press the **Create Row** button, followed by **Save**. This creates a row.

When an empty row is created, Kibana shows the prompt Add panel to empty row on the left. Click this button. It takes you to the configuration screen to add a new panel. Choose histogram from the dropdown menu.

There are many parameters to configure for a new histogram, but you can just scroll down and press the Save button. This creates a new panel.

Install Nginx

We will use Nginx as a proxy server to allow access to the dashboard from the Public Internet (with basic authentication).

- Install Nginx as follows:

```
portmaster www/nginx security/py-htpasswd
```

- Enable nginx to start at boot

```
echo 'nginx_enable="YES"' >> /etc/rc.conf
```

- Start nginx

```
service nginx start
```

- Edit /usr/local/etc/nginx/nginx.conf

```
vi /usr/local/etc/nginx/nginx.conf
```

  - And change the primary server block as follows:

```
#
# Nginx proxy for Elasticsearch + Kibana
#
# In this setup, we are password protecting the saving of dashboards. You may
# wish to extend the password protection to all paths.
#
# Even though these paths are being called as the result of an ajax request, the
# browser will prompt for a username/password on the first request
#
# If you use this, you'll want to point config.js at http://FQDN:80/ instead of
# http://FQDN:9200
#
server {
 listen                *:80 ;
 server_name           localhost;
 access_log            /var/log/nginx-kibana.log;

 location / {
   root  /usr/local/www/kibana;
   index  index.html  index.htm;
 }
```

```
 location ~ ^/_aliases$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
 }

 location ~ ^/.*/_aliases$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
 }

 location ~ ^/_nodes$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
 }

 location ~ ^/.*/_search$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
 }

 location ~ ^/.*/_mapping {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
 }

 # Password protected end points
 location ~ ^/kibana-int/dashboard/.*$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
  limit_except GET {
   proxy_pass http://127.0.0.1:9200;
   auth_basic "Restricted";
   auth_basic_user_file /usr/local/etc/nginx/log.example.com.htpasswd;
  }
 }

 location ~ ^/kibana-int/temp.*$ {
  proxy_pass http://127.0.0.1:9200;
  proxy_read_timeout 90;
  limit_except GET {
   proxy_pass http://127.0.0.1:9200;
   auth_basic "Restricted";
   auth_basic_user_file /usr/local/etc/nginx/log.example.com.htpasswd;
  }
 }
}
```

- And generate a htpasswd file:

```
python2.7 /usr/local/bin/htpasswd.py -c -b /usr/local/etc/nginx/log.example.com.htpasswd usern
ame SuperSecretPassword
```

NOTE: Make sure to change the username and SuperSecretPassword to your needs

- Start and enable nginx at boot:

```
echo 'nginx_enable="YES"' >> /etc/rc.conf
service nginx start
```

Now, you should be able to see the generic Kibana dashboard at your server's IP address or domain, using your favorite browser.

Forwarding Syslog to Fluentd

Forwarding Debian rsyslog Traffic to Fluentd

I use Debian an many production systems, and one one the packages is rsyslogd. It needs to be reconfigured to forward syslog events to the port Fluentd listens to (port 5140 in this example).

- Open the rsyslog configuration file and add the following line at the top

```
sudo vi/etc/rsyslog.conf
```

  - And add the following line

```
*.* @127.0.0.1:5140
```

- After saving and exiting the editor, restart rsyslogd as follows:

```
sudo service rsyslog restart
```

Forwarding FreeBSD syslogd Traffic to Fluentd

I also have been switching many of my production systems to FreeBSD, and the default logging mechanism is syslogd. It needs to be reconfigured to forward syslog events to the port Fluentd listens to (port 5140 in this example).

- Open the rsyslog configuration file and add the following line at the top

```
sudo vi/etc/syslog.conf
```

  - And add the following line

```
*.* @127.0.0.1:5140
```

Resources

- http://docs.fluentd.org/articles/install-from-source
- https://www.digitalocean.com/community/tutorials/elasticsearch-fluentd-and-kibana-open-source-log-search-and-visualization
- https://raw.githubusercontent.com/elasticsearch/kibana/kibana3/sample/nginx.conf

Related issues:

Copied from FreeBSD Administration - Support #414: Install an ElasticSearch, ...	**Closed**	**07/10/2014**

History

#1 - 08/13/2014 12:14 PM - Daniel Curtis

- Copied from Support #414: Install an ElasticSearch, Logstash, Kibana (ELK) Stack on FreeBSD added

#2 - 08/13/2014 01:51 PM - Daniel Curtis

- Description updated

#3 - 08/13/2014 05:38 PM - Daniel Curtis

- Description updated

#4 - 08/13/2014 06:31 PM - Daniel Curtis

- Status changed from New to Resolved

- % Done changed from 10 to 100

#5 - 08/14/2014 09:21 AM - Daniel Curtis

- Status changed from Resolved to Feedback

- % Done changed from 100 to 90

#6 - 08/14/2014 10:46 AM - Daniel Curtis

- Subject changed from Installing A ElasticSearch, Fluentd, Kibana Stack on FreeBSD to Installing an ElasticSearch, Fluentd, Kibana Stack on FreeBSD

- Description updated

- Status changed from Feedback to Resolved

- % Done changed from 90 to 100

#7 - 12/26/2014 09:02 PM - Daniel Curtis

- Description updated

#8 - 12/26/2014 09:34 PM - Daniel Curtis

- Description updated

#9 - 12/26/2014 09:34 PM - Daniel Curtis

- Project changed from 90 to FreeBSD Administration

#10 - 01/08/2015 03:41 PM - Daniel Curtis

- Description updated

#11 - 01/08/2015 03:41 PM - Daniel Curtis

- Description updated

#12 - 01/14/2015 02:06 PM - Daniel Curtis

- Description updated

#13 - 01/15/2015 02:36 PM - Daniel Curtis

- Description updated

#14 - 01/15/2015 02:39 PM - Daniel Curtis

- Status changed from Resolved to Closed

#15 - 02/14/2015 10:37 AM - Daniel Curtis

- Target version set to FreeBSD 9

#16 - 02/14/2015 11:45 AM - Daniel Curtis

- Category set to Logging Server

#17 - 04/04/2015 02:44 PM - Daniel Curtis

- Description updated

#18 - 04/04/2015 08:50 PM - Daniel Curtis

- Subject changed from Installing an ElasticSearch, Fluentd, Kibana Stack on FreeBSD to Install an ElasticSearch, Fluentd, Kibana Stack on FreeBSD

- Description updated

#19 - 04/05/2015 10:55 AM - Daniel Curtis

- Description updated

#20 - 04/05/2015 11:18 AM - Daniel Curtis

- Description updated