**Installing the ELK Stack on AWS: A Step-by-Step Guide**

Want to bring in the ELK stack for your AWS logging and monitoring needs? This guide will get you set up with the open source solution.

[**[](https://dzone.com/users/2819907/asafyigal.html)**](https://dzone.com/users/2819907/asafyigal.html)**by**

[**Asaf Yigal**](https://dzone.com/users/2819907/asafyigal.html)

**·**

**Dec. 22, 17 ·**[**Cloud Zone**](https://dzone.com/cloud-computing-tutorials-tools-news)**· Tutorial**

**Like**[**(8)**](https://dzone.com/articles/how-to-install-the-elk-stack-on-aws-a-step-by-step)

**Comment (1)**

**Save**

[**Tweet**](https://dzone.com/articles/how-to-install-the-elk-stack-on-aws-a-step-by-step)

**37.23K Views**

Join the DZone community and get the full member experience.

[**JOIN FOR FREE**](https://dzone.com/static/registration.html)

The [ELK Stack](https://logz.io/learn/complete-guide-elk-stack/) is a great open-source stack for log aggregation and analytics. It stands for Elasticsearch (a NoSQL database and search server), Logstash (a log shipping and parsing service), and Kibana (a web interface that connects users with the Elasticsearch database and enables visualization and search options for system operation users). With a large open-source community, ELK has become quite popular, and it is a pleasure to work with.

In this article, we will guide you through the simple installation process for installing the ELK Stack on Amazon Web Services.

The following instructions will lead you through the steps involved in creating a working sandbox environment. Due to the fact that a production setup is more comprehensive, we decided to elaborate on how each component configuration should be changed to prepare for use in a production environment.

We’ll start by describing the environment, then we’ll walk through how each component is installed, and finish by configuring our sandbox server to send its system logs to Logstash and view them via Kibana.

Note: All of the ELK components need Java to work, so we will have to install a Java Development Kit (JDK) first. Likewise, the instructions here were tested on version 6.x of the ELK Stack.

**The AWS Environment**

We ran this tutorial on a single AWS Ubuntu 16.04 instance on an m4.large instance using its local storage. We started an EC2 instance in the public subnet of a VPC, and then we set up the security group (firewall) to enable access from anywhere using SSH and TCP 5601 (Kibana). Finally, we added a new elastic IP address and associated it with our running instance in order to connect to the internet.

**Production tip:** A production installation needs at least three EC2 instances — one per component, each with an attached EBS SSD volume.

**Step-by-Step ELK Installation**

To start, connect to the running server via SSH: *ssh ubuntu@YOUR\_ELASTIC\_IP*

**Package Installations**

Prepare the system by running (this may take a few minutes):

1

sudo apt-get update

2

sudo apt-get upgrade

**Install Java**

The ELK Stack (Elasticsearch and Logstash specifically) require the installation of Java 8 and above.

1

sudo apt-get install default-jre

Verify that Java is installed:

1

java -version

If the output of the previous command is similar to this, then you’ll know that you’re heading in the right direction:

1

openjdk version "1.8.0\_151"

2

OpenJDK Runtime Environment (build 1.8.0\_151-8u151-b12-0ubuntu0.16.04.2-b12)

3

OpenJDK 64-Bit Server VM (build 25.151-b12, mixed mode)

**Elasticsearch Installation**

Elasticsearch is a widely used database and search server, and it’s the main component of the ELK setup.

Elasticsearch’s benefits include:

* Easy installation and use
* A powerful internal search technology (Lucene)
* A RESTful web interface
* The ability to work with data in schema-free JSON documents (noSQL)
* Open source

To begin the process of installing Elasticsearch, add the following repository key:

1

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

Add the following Elasticsearch list to the key:

1

echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-6.x.list

2

sudo apt-get update

**Install:**

1

sudo apt-get install elasticsearch

Open the Elasticsearch configuration file at: */etc/elasticsearch/elasticsearch.yml*, and apply the following configurations:

1

network.host: "localhost"

2

http.port: 9200

**Start service:**

1

sudo service elasticsearch start

**Test:**

1

sudo curl http://localhost:9200

If the output is similar to this, then you will know that Elasticsearch is running properly:

1

{

2

"name" : "J-Cm4Eg",

3

"cluster\_name" : "elasticsearch",

4

"cluster\_uuid" : "-Hqx5vMgSbaZdM4-hjzMEQ",

5

"version" : {

6

"number" : "6.0.1",

7

"build\_hash" : "601be4a",

8

"build\_date" : "2017-12-04T09:29:09.525Z",

9

"build\_snapshot" : false,

10

"lucene\_version" : "7.0.1",

11

"minimum\_wire\_compatibility\_version" : "5.6.0",

12

"minimum\_index\_compatibility\_version" : "5.0.0"

13

},

14

"tagline" : "You Know, for Search"

15

}

In order to make the service start on boot run:

1

sudo update-rc.d elasticsearch defaults 95 10

**Production tip:** DO NOT open any other ports, like 9200, to the world! There are many bots that search for 9200 and execute groovy scripts to overtake machines. DO NOT bind Elasticsearch to a public IP.

**Logstash Installation**

Logstash is an open-source tool that collects, parses, and stores logs for future use and makes rapid log analysis possible. Logstash is useful for both aggregating logs from multiple sources, like a cluster of Docker instances, and parsing them from text lines into a structured format such as JSON. In the ELK Stack, Logstash uses Elasticsearch to store and index logs.

Install Logstash with:

1

sudo apt-get install logstash

**Collect System Logs with Logstash**

Create a Logstash configuration file:

1

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

Enter the following configuration:

1

input {

2

file {

3

type => "syslog"

4

path => [ "/var/log/messages", "/var/log/\*.log" ]

5

}

6

}

7

output {

8

stdout {

9

codec => rubydebug

10

}

11

​

12

elasticsearch {

13

hosts => "localhost" # Use the internal IP of your Elasticsearch server

14

}

15

}

This file tells Logstash to store the local syslog ‘*/var/log/syslog*’ and all the files under ‘*/var/log\*.log*’ inside the Elasticsearch database in a structured way.

The input section specifies which files to collect (path) and what format to expect (syslog). The output section uses two outputs – stdout and elasticsearch. The stdout output is used to debug Logstash – you should find nicely-formatted log messages under ‘*/var/log/logstash/logstash.stdout*’. The elasticsearch output is what actually stores the logs in Elasticsearch.

In this example, we are using localhost for the Elasticsearch hostname. In a real production setup, however, the Elasticsearch hostname would be different because Logstash and Elasticsearch should be hosted on different machines.

**Production tip:** Running Logstash and Elasticsearch is a very common pitfall of the ELK stack and often causes servers to fail in production. You can read some more tip on [how to install ELK in production](https://logz.io/blog/deploy-elk-production/).

Finally, start Logstash to read the configuration:

1

sudo service logstash restart

To make sure the data is being indexed, use:

1

sudo curl -XGET 'localhost:9200/\_cat/indices?v&pretty'

You should see your new Logstash index created:

1

health status index uuid pri rep docs.count docs.deleted store.size pri.store.size

2

yellow open logstash-2017.12.12 sCf9FxPETImc6pyZMFYacw 5 1 31 0 55.5kb 55.5kb

**Kibana Installation**

Kibana is an open-source data visualization plugin for Elasticsearch. It provides visualization capabilities on top of the content indexed on an Elasticsearch cluster. Users can create bar, line, and scatter plots; pie charts; and maps on top of large volumes of data.

Among other uses, Kibana makes working with logs easy. Its graphical web interface even lets beginning users execute powerful log searches.

To install Kibana, use this command:

1

sudo apt-get install kibana

Open the Kibana configuration file and enter the following configurations:

1

sudo vim /etc/kibana/kibana.yml

1

server.port: 5601

2

server.host: "localhost"

3

elasticsearch.url: "http://localhost:9200"

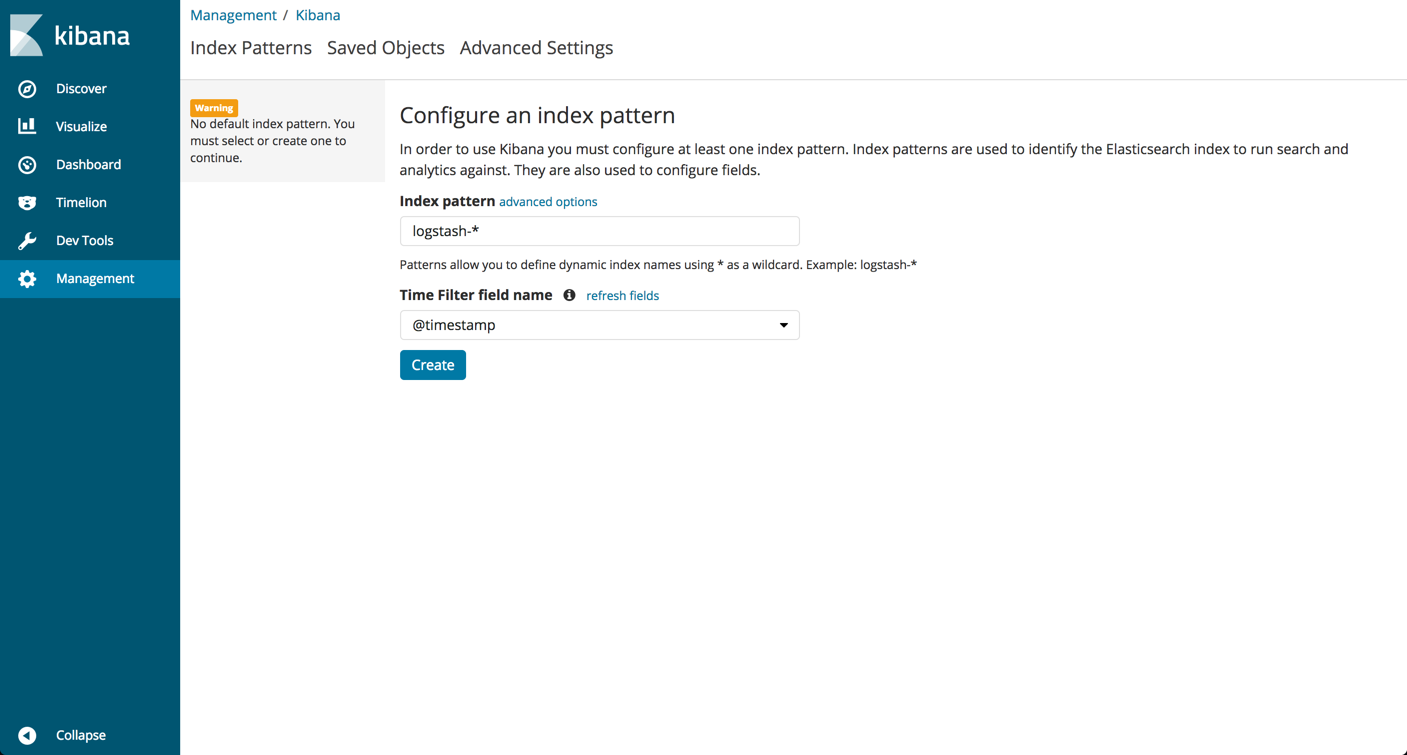
Start Kibana:

1

sudo service kibana start

Test: Point your browser to ‘http://YOUR\_ELASTIC\_IP:5601’ after Kibana is started.

You should see a page similar to this:



Before continuing with the Kibana setup, you must define an Elasticsearch index pattern.

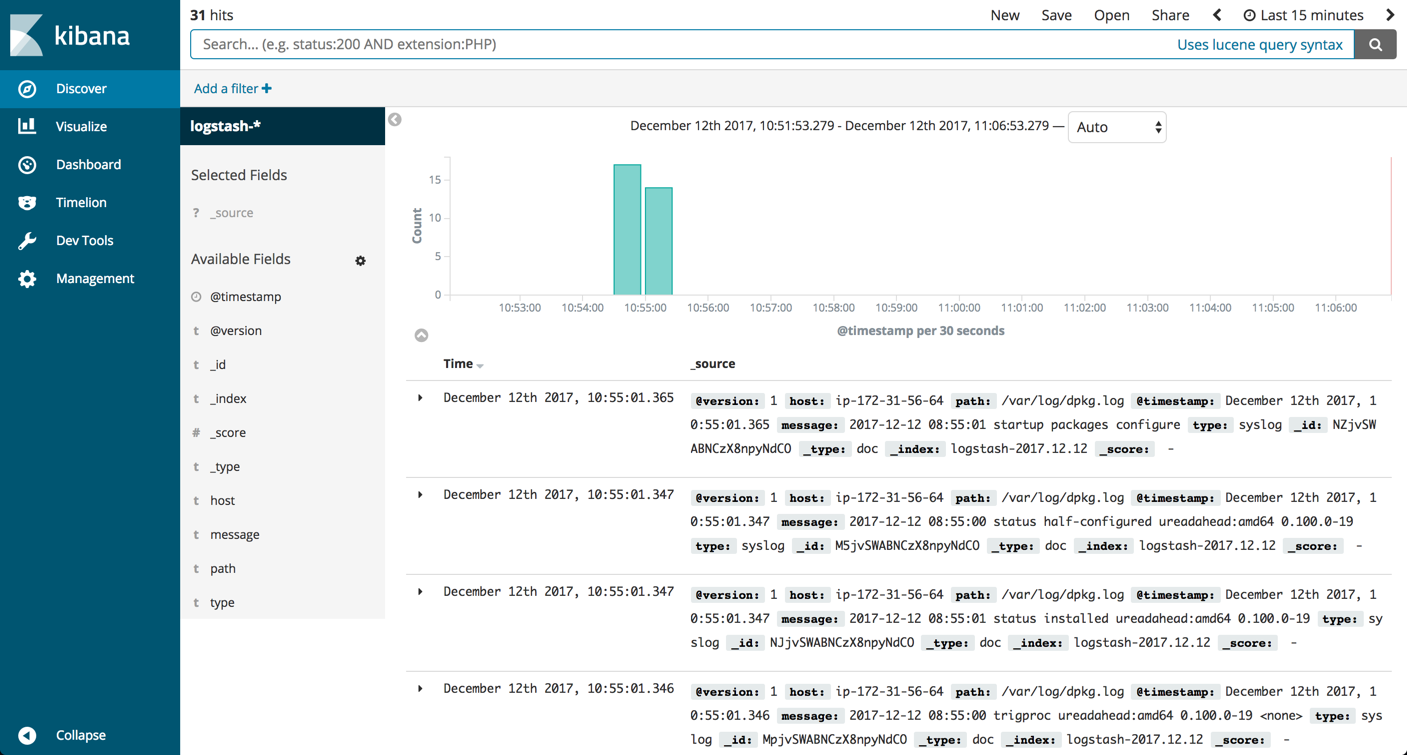
What does an “index pattern” mean, and why do we have to configure it? Logstash creates a new Elasticsearch index (database) every day. The names of the indices look like this: logstash-YYYY.MM.DD — for example, “logstash-2017.12.10” for the index that was created on December 10, 2017.

Kibana works with these Elasticsearch indices, so it needs to know which ones to use. The setup screen provides a default pattern, ‘logstash-\*’, that basically means “Show the logs from all of the dates.”

Since we created a new Logstash index in the previous section, all we have to do us click the “Create” button to define the pattern in Kibana.

**Production tip:** In this tutorial, we are accessing Kibana directly through its application server on port 5601, but in a production environment you might want to put a reverse proxy server, like Nginx, in front of it.

To see your logs, go to the Discover page in Kibana:



As you can see, creating a whole pipeline of log shipping, storing, and viewing is not such a tough task. In the past, storing, and analyzing logs was an arcane art that required the manipulation of huge, unstructured text files. But the future looks much brighter and simpler.