FREE eBook

LEARNING logstash

Free unaffiliated eBook created from
Stack Overflow contributors.

#logstash

Table of Contents

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: logstash

It is an unofficial and free logstash ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official logstash.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with logstash

Remarks

This section provides an overview of what logstash is, and why a developer might want to use it.

It should also mention any large subjects within logstash, and link out to the related topics. Since the Documentation for logstash is new, you may need to create initial versions of those related topics.

Examples

Installation or Setup

Detailed instructions on getting logstash set up or installed.

A basic, complete Syslog example

Going to its roots, Logstash has the ability to parse and store syslog data. This example shows a basic configuration that gets you to that.

```
input {
  file {
    path => [
      "/var/log/syslog",
      "/var/log/auth.log"
    ]
    type => "syslog"
  }
}

filter {
  if [type] == "syslog" {
    # Uses built-in Grok patterns to parse this standard format
    grok {
      match => {
        "message" => "%{SYSLOGBASE}%{SPACE}%{GREEDYDATA:SYSLOGMESSAGE}"
      }
    }
    # Sets the timestamp of the event to the timestamp of recorded in the log-data
    # By default, logstash sets the timestamp to the time it was ingested.
    date {
      match => [ "timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
    }
  }
}

output {
  # Outputs processed events to an elasticsearch instance local to the box.
  elasticsearch {
    hosts => [
      "localhost"
```

```
    ]
  }
}
```

Outputting to ElasticSearch: multiple indices and mappings

Sometimes, you need to output to more than one index in ElasticSearch, or have a custom mapping you want to apply to new indices as they roll in.

There are two ways to apply a custom mapping. One way, is to upload an ElasticSearch template. See the ElasticSearch documentation for that. The other way is to specify a mapping in the `elasticsearch {}` output itself. That is what is shown here.

```
output {
  if [type] == 'metrics' {
    # The 'metrics' index rotates weekly.
    # The 'metrics-mapping.json' file defines the custom mappings.
    elasticsearch {
      hosts              => [ 'localhost' ]
      index              => "metrics-%{xxxx.ww}"
      manage_template    => true
      template           => "/etc/logstash/metrics-mapping.json"
      template_overwrite => true
    }
  }
}
```

This will output `metrics` events to `metrics-` indexes on ElasticSearch, which will rotate weekly using the ISO week. The template used for new indexes is defined as part of this configuration. Defining a template has the advantage of forcing the types of fields to a uniform type. This is useful in larger configurations where multiple types may attempt to define a field as a slightly different data-type.

This method is useful in staging and QA environments, as the ElasticSearch templates are defined by the LogStash code and don't have to be configured separately as part of the ElasticSearch cluster setup.

Read Getting started with logstash online: https://riptutorial.com/logstash/topic/6903/getting-started-with-logstash

Credits

S. No	Chapters	Contributors
1	Getting started with logstash	Community, sysadmin1138