elastic

Kibana Beats Logstash ECE Cloud Elasticsearch

logstash

Logstash is a Data Processing Pipeline

Ingests data

Data can come from a variety of sources

Filters

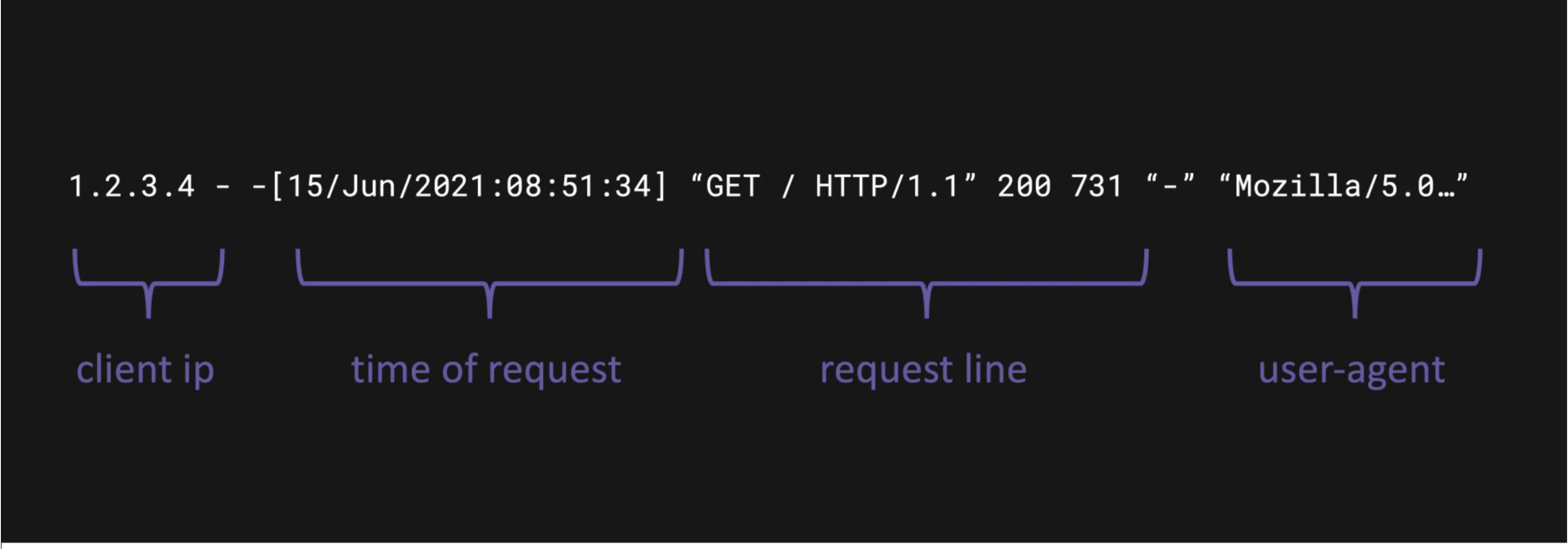
Allows you to normalize, enrich and even exclude data

Forwards

Finally, sends data to your favorite “stash”

Logstash Plugins

- There is already a collection of input, filter, output and codec plugins
  - Plugins help to ease the use of Logstash
- A popular set of input plugins is Beats
  - But there are a significant number of plugins for phases of the pipeline available
- Plugins are provided in self-contained Gems from RubyGems.org
  - Plugin manager script provides ability to add, update and remove plugins for your deployment



```
1.2.3.4 - -[15/Jun/2021:08:51:34] “GET / HTTP/1.1” 200 731 “-” “Mozilla/5.0…”
```

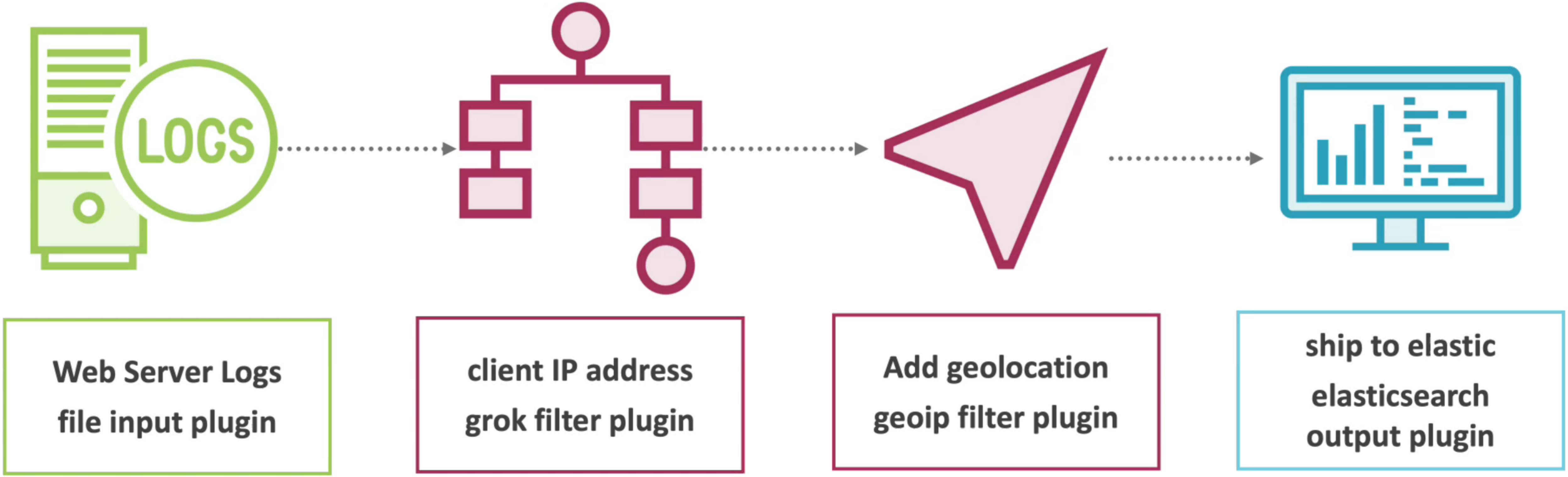
Filter Example - Grok

The grok filter provides the ability to provide structure to arbitrary text

This helps to make the data queryable

Grok works well with log data that is written to be human-readable, such as Apache logs

Example Web Server Pipeline



Web Server Logs
file input plugin

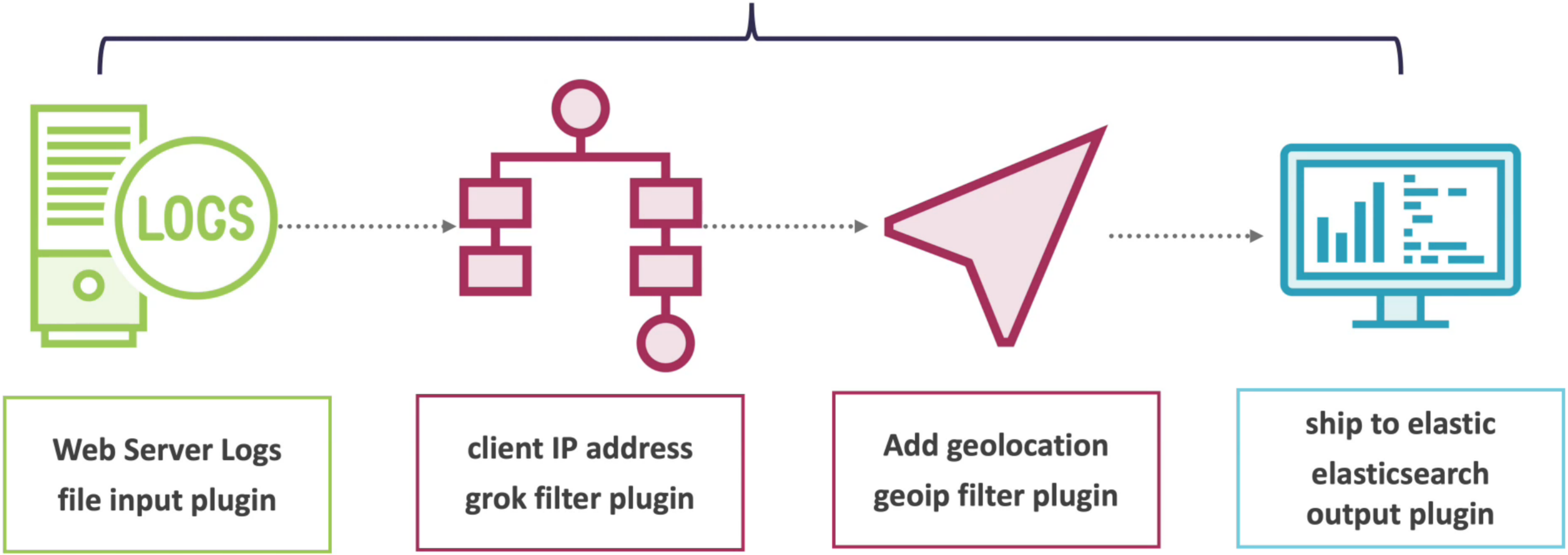
client IP address
grok filter plugin

Add geolocation
geoip filter plugin

ship to elastic
elasticsearch
output plugin

Example Web Server Pipeline

config files

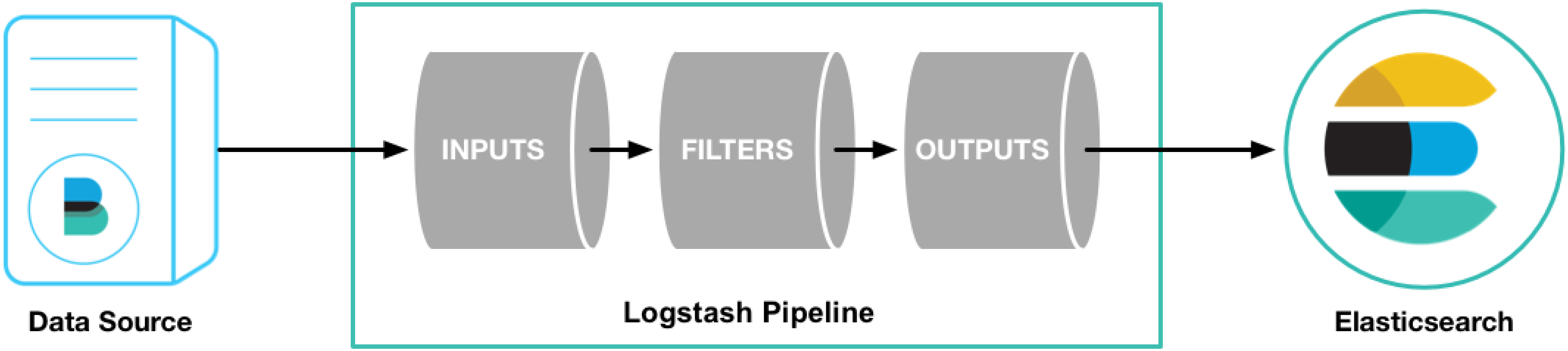


Web Server Logs
file input plugin

client IP address
grok filter plugin

Add geolocation
geoip filter plugin

ship to elastic
elasticsearch
output plugin


Data Source
INPUTS
FILTERS
OUTPUTS
Logstash Pipeline
Elasticsearch

Environment Setup

Download Logstash:

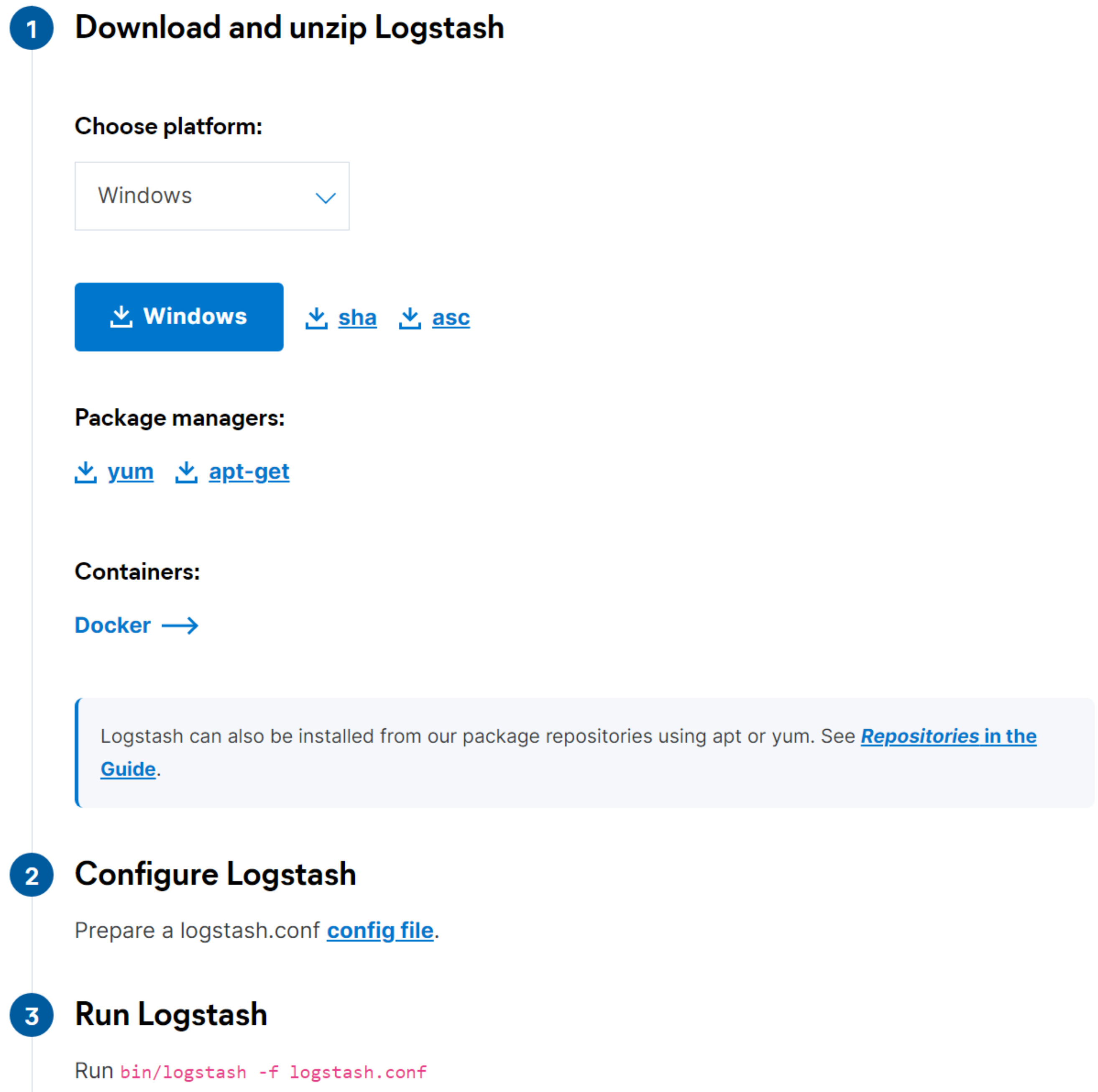
https://www.elastic.co/downloads/past-releases/logstash-7-11-2

Latest Version:

https://www.elastic.co/downloads/logstash

JVM Settings:

https://www.elastic.co/guide/en/logstash/current/jvm-settings.html



1 Download and unzip Logstash

Choose platform:

Windows

Windows **sha** **asc**

Package managers:

yum **apt-get**

Containers:

Docker →

> Logstash can also be installed from our package repositories using apt or yum. See ***Repositories* in the Guide**.

2 Configure Logstash

Prepare a logstash.conf **config file**.

3 Run Logstash

Run `bin/logstash -f logstash.conf`

Download Logstash:

https://www.elastic.co/downloads/past-releases/logstash-7-11-2

Latest Version:

https://www.elastic.co/downloads/logstash

Starting First Event (using CLI from terminal)

In windows:

```
.\bin\logstash.bat -e "input { stdin { } } output { stdout {} }"
```

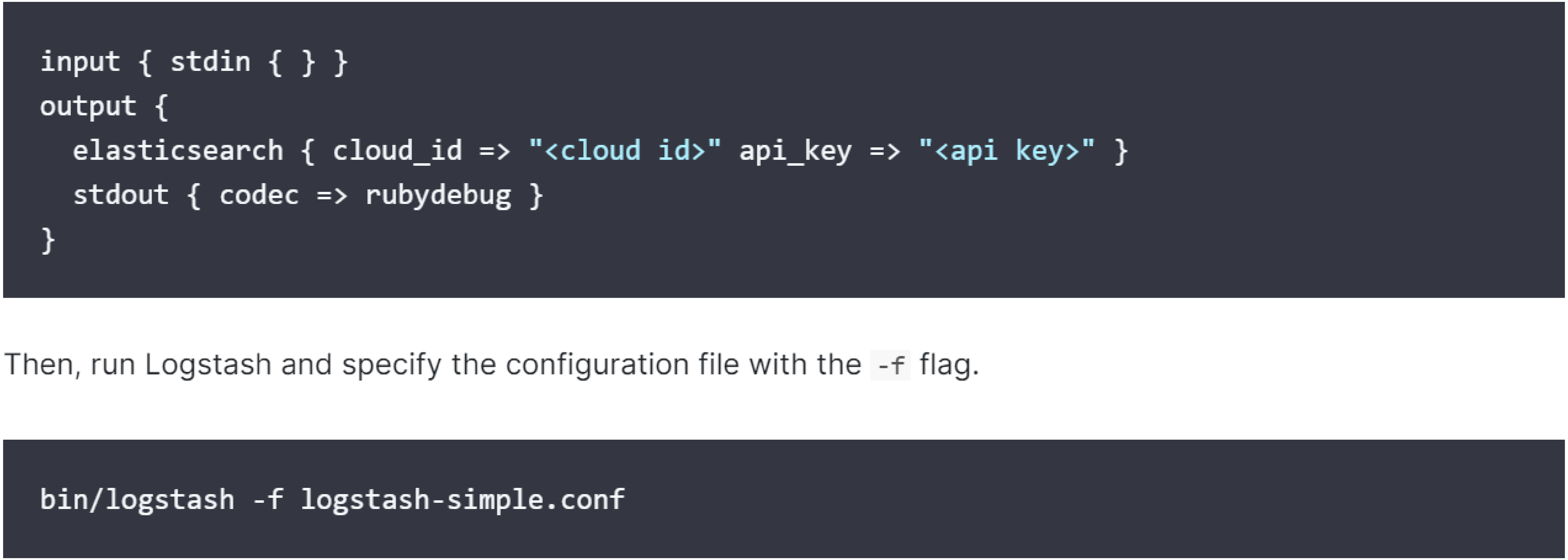
In Linux:

```
bin/logstash   -e 'input {stdin {}} output {stdout {}}'
```

```
bin/logstash  -e 'input {stdin {}} output {elasticsearch {hosts => [192.168.127.200]}}'
```

Starting First Event (using conf file)

1. Create conf file in pipelines directory
2. Update configuration file
3. Run logstash using configuration file



```
input { stdin { } }
output {
  elasticsearch { cloud_id => "<cloud id>" api_key => "<api key>" }
  stdout { codec => rubydebug }
}
```

Then, run Logstash and specify the configuration file with the `-f` flag.

```
bin/logstash -f logstash-simple.conf
```

Run bin/logstash -f logstash.conf

https://www.elastic.co/guide/en/logstash/current/configuration.html

```
.\bin\logstash.bat  -e "input { stdin { } } output { stdout {} }“

.\bin\logstash.bat  -e 'input {stdin {}} output {elasticsearch {hosts => [192.168.127.200]}}'
```

There are two ways to config Logstash

From command line “–e”
Editing configuration file which is actual file “-f”

On Linux:

Check status:

```
systemctl status logstash.service
```

From Command Line:-

```
/usr/share/logstash/bin/logstash -e 'input {stdin {}} output {elasticsearch {hosts => [192.168.127.200]}}'
```

```
curl https://192.168.127.200:9200/logstash-*/_search
```

the result we get unformatted

```
sudo apt install jq
```

Then run the same above command with jq and redirection operator

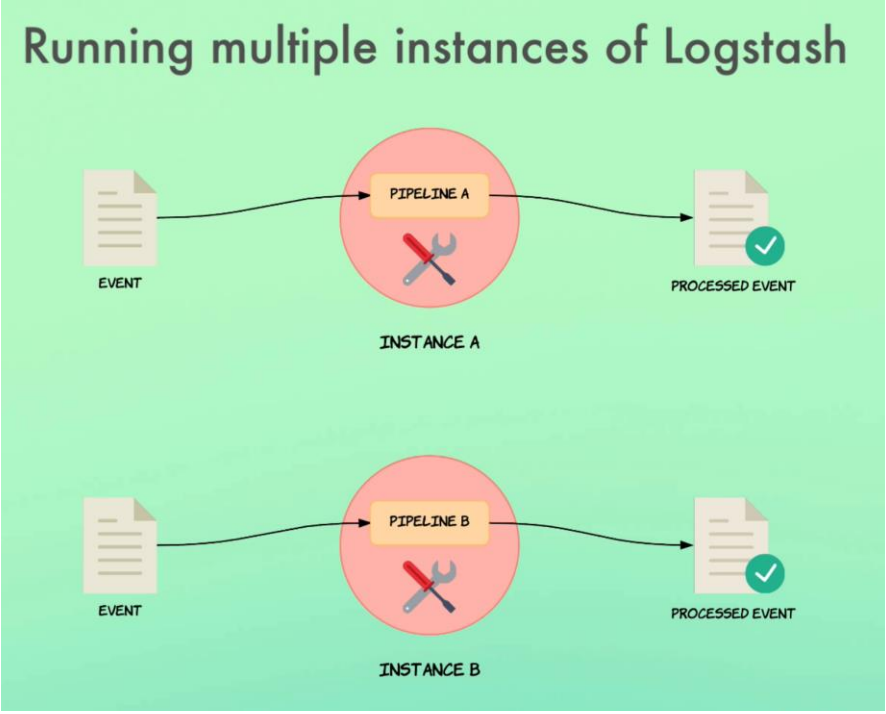
```
curl https://192.168.127.200:9200/logstash-*/_search | jq .
```

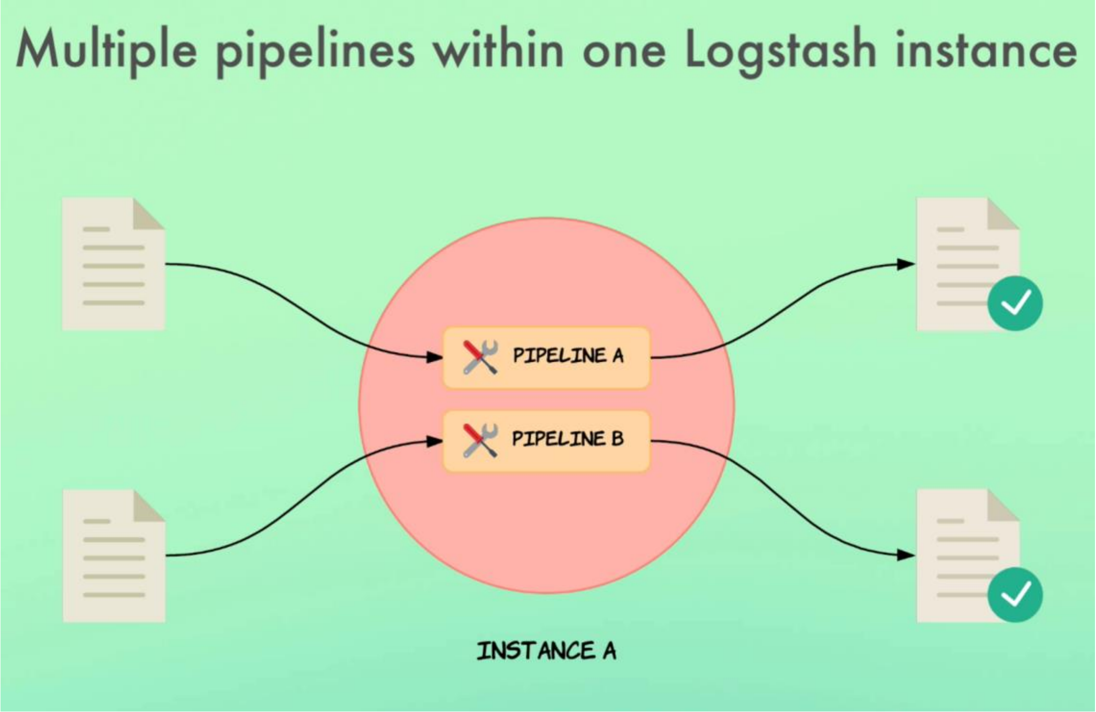
```
sudo systemctl enable logstash.service
sudo systemctl enable kibana.service
sudo systemctl enable elasticsearch.service
```

```
sudo systemctl enable kibana.service
sudo systemctl demon-reload
sudo systemctl enable kibana.service
sudo systemctl start kibana.service
sudo systemctl start elasticsearch.service
sudo systemctl start logstash.service
```

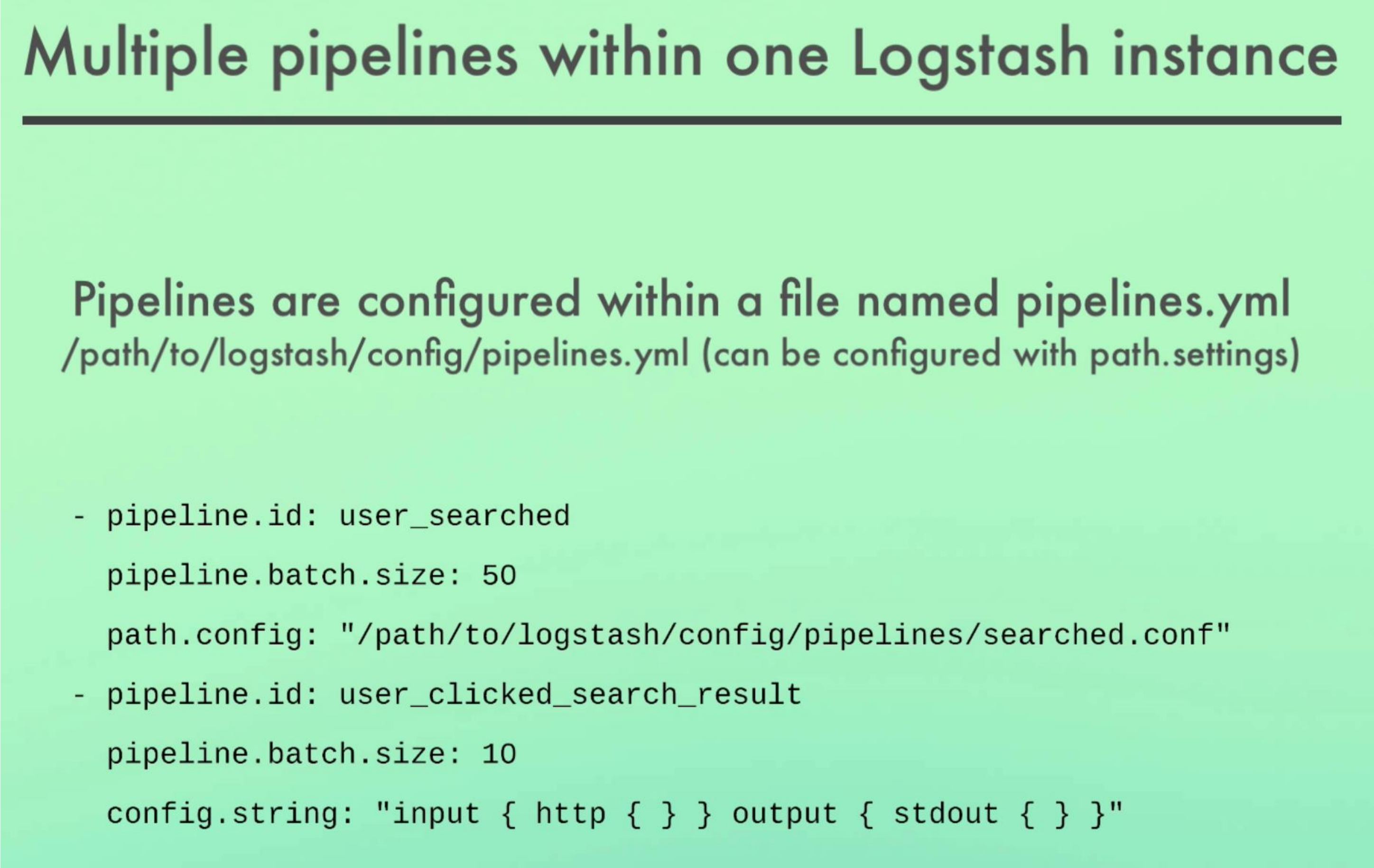
Running Multiple Pipelines

We use pipeline.yml file to configure multiple pipelines





https://www.elastic.co/guide/en/logstash/current/logstash-settings-file.html



Multiple pipelines within one Logstash instance

Pipelines are configured within a file named pipelines.yml

/path/to/logstash/config/pipelines.yml (can be configured with path.settings)

```
- pipeline.id: user_searched
  pipeline.batch.size: 50
  path.config: "/path/to/logstash/config/pipelines/searched.conf"
- pipeline.id: user_clicked_search_result
  pipeline.batch.size: 10
  config.string: "input { http { } } output { stdout { } }"
```