ELK系統應用實務

資訊工業策進會
資安科技研究所 沈裕翔
2017.07.27

大綱

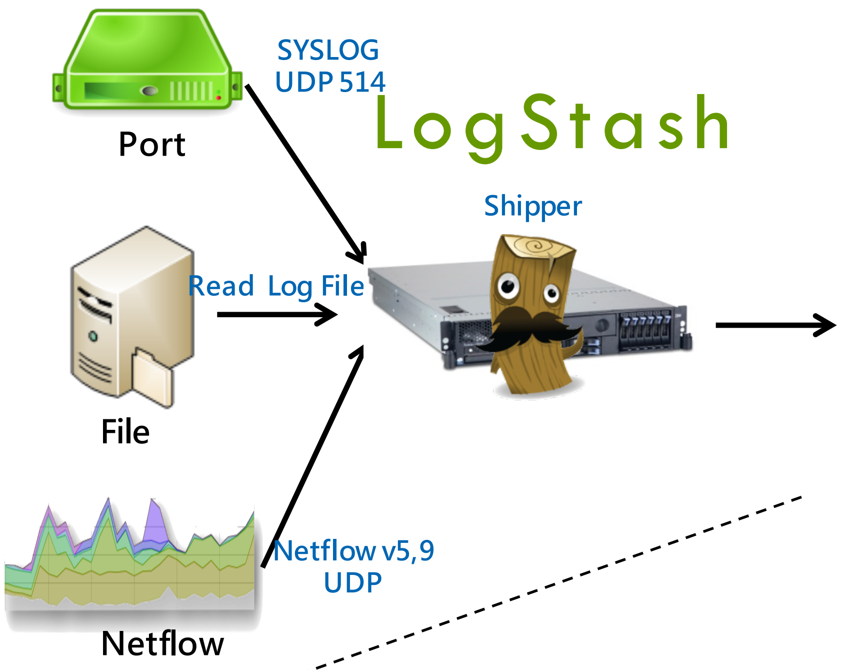
- ELK
- logstash I/O
- Filter log
- Grok log
- Multiline log
- Elasticsearch
- Kibana

- Basic Lab
- Web(nxlog)/FW log Parsing Lab
- Grok, Multiline Lab
- Flow Lab

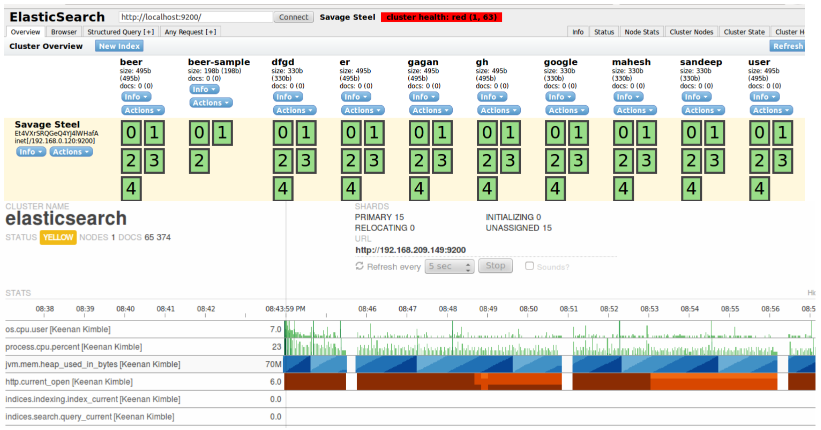
ELK簡介

- ELK
  - Elasticsearch, Logstash, and Kibana
    - Elasticsearch
      - The Amazing Log Search Tool
    - Logstash
      - Routing Your Log Data
    - Kibana
      - Visualizing Your Log Data
  - Real-time data and real-time analytics
  - Scalable, high-availability, multi-tenant
  - Full text search
  - Document orientation

ELK架構



elasticsearch.



ElasticSearch http://localhost:9200/ Connect Savage Steel cluster health: red (1, 63)

Overview | Browser | Structured Query [+] | Any Request [+] | Info | Status | Node Stats | Cluster Nodes | Cluster State | Cluster H

Cluster Overview New Index Refresh

	beer	beer-sample	dfgd	er	gagan	gh	google	mahesh	sandeep	user
	size: 495b (495b) docs: 0 (0) Info ▾ Actions ▾	size: 198b (198b) docs: 0 (0) Info ▾ Actions ▾	size: 330b (330b) docs: 0 (0) Info ▾ Actions ▾	size: 495b (495b) docs: 0 (0) Info ▾ Actions ▾	size: 495b (495b) docs: 0 (0) Info ▾ Actions ▾	size: 495b (495b) docs: 0 (0) Info ▾ Actions ▾	size: 330b (330b) docs: 0 (0) Info ▾ Actions ▾	size: 330b (330b) docs: 0 (0) Info ▾ Actions ▾	size: 330b (330b) docs: 0 (0) Info ▾ Actions ▾	size: 495b (495b) docs: 0 (0) Info ▾ Actions ▾
Savage Steel Et4VXrSRQGeQ4YJ4lWHafA inet[/192.168.0.120:9200] Info ▾ Actions ▾	0 1 2 3 4	0 1 2	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4

CLUSTER NAME
elasticsearch
STATUS YELLOW NODES 1 DOCS 65 374

SHARDS
PRIMARY 15 INITIALIZING 0
RELOCATING 0 UNASSIGNED 15
URL
http://192.168.209.149:9200
Refresh every 5 sec Stop ☐ Sounds?

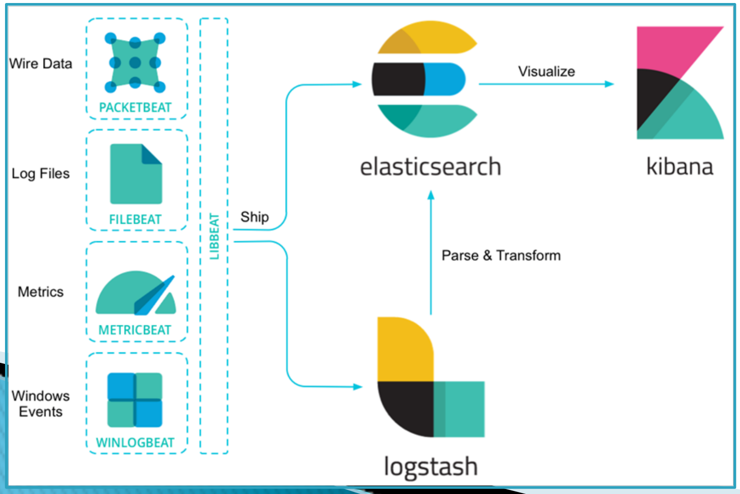
STATS

		08:38	08:39	08:40	08:41	08:42	08:43:59 PM	08:46	08:47	08:48	08:49	08:50	08:51	08:52	08:53	08:54	08:55	08:56	08:5
os.cpu.user [Keenan Kimble]	7.0																		
process.cpu.percent [Keenan Kimble]	23																		
jvm.mem.heap_used_in_bytes [Keenan Kimble]	70M																		
http.current_open [Keenan Kimble]	6.0																		
indices.indexing.index_current [Keenan Kimble]	0.0																		
indices.search.query_current [Keenan Kimble]	0.0																		

Store
Analysis

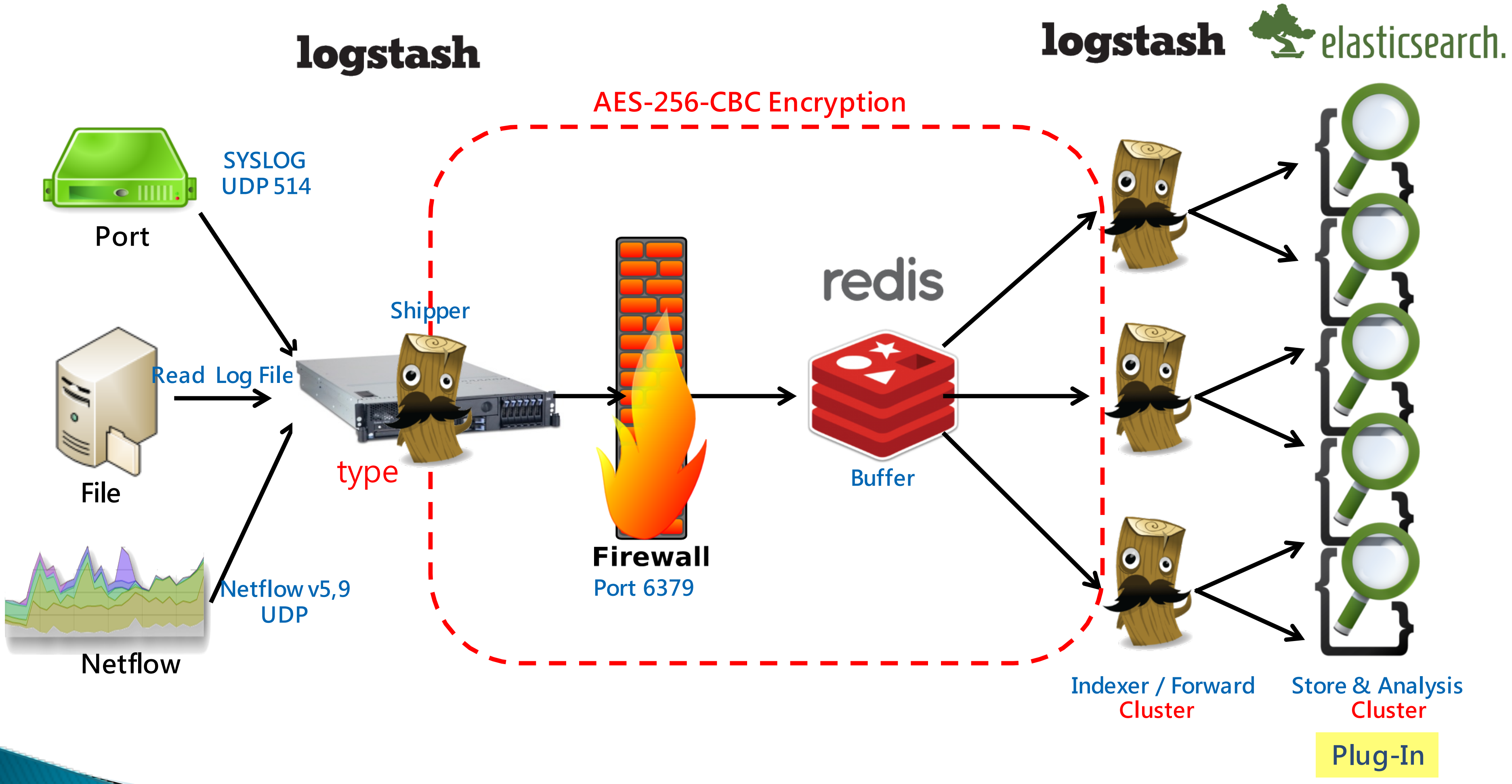
Kibana

Virtualization



WebUI

ELK 2.x



logstash

logstash elasticsearch.

AES-256-CBC Encryption

SYSLOG
UDP 514

Port

Shipper

Read Log File

File

type

Netflow v5,9
UDP

Netflow

Firewall
Port 6379

redis

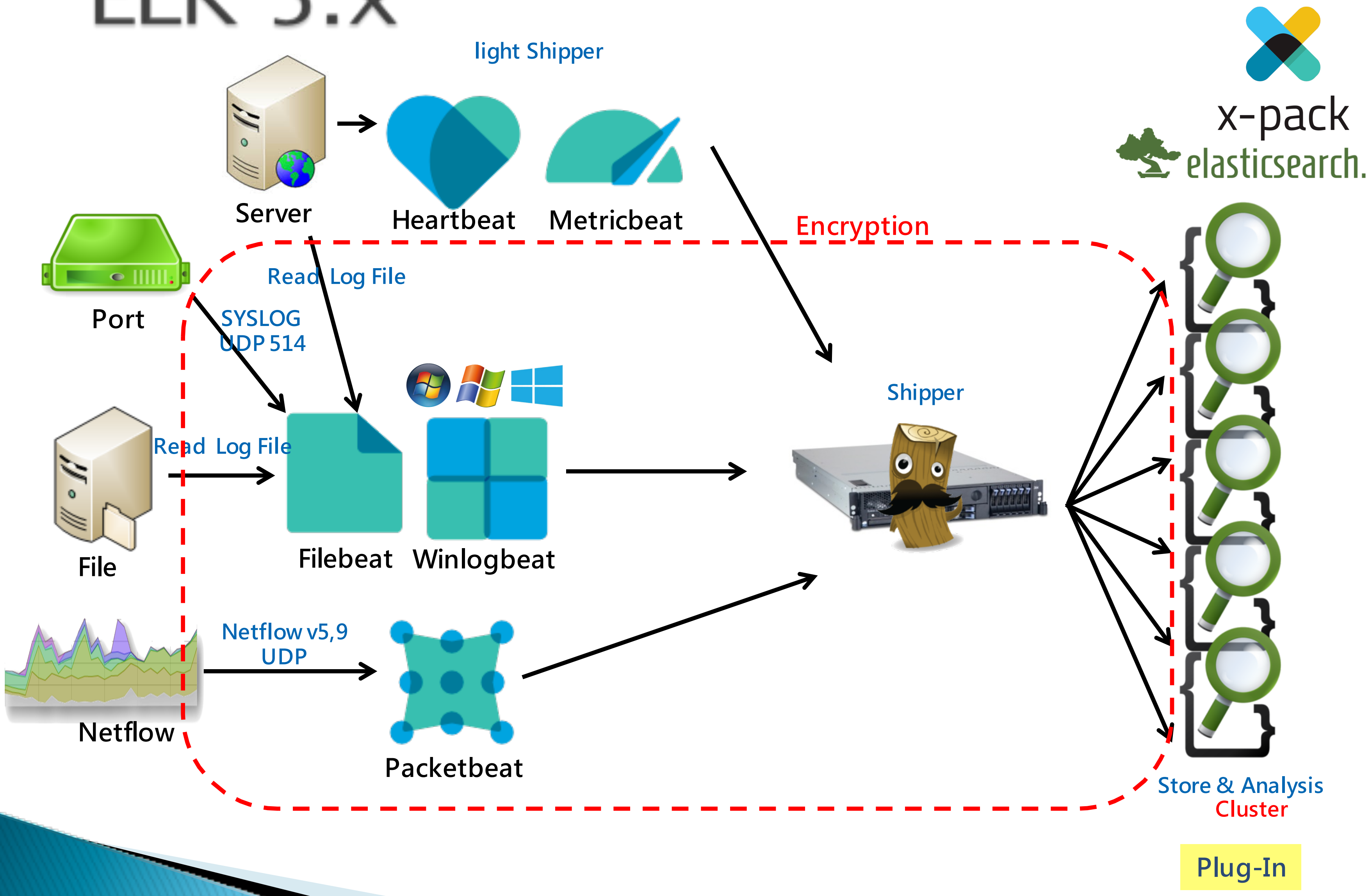
Buffer

Indexer / Forward
Cluster

Store & Analysis
Cluster

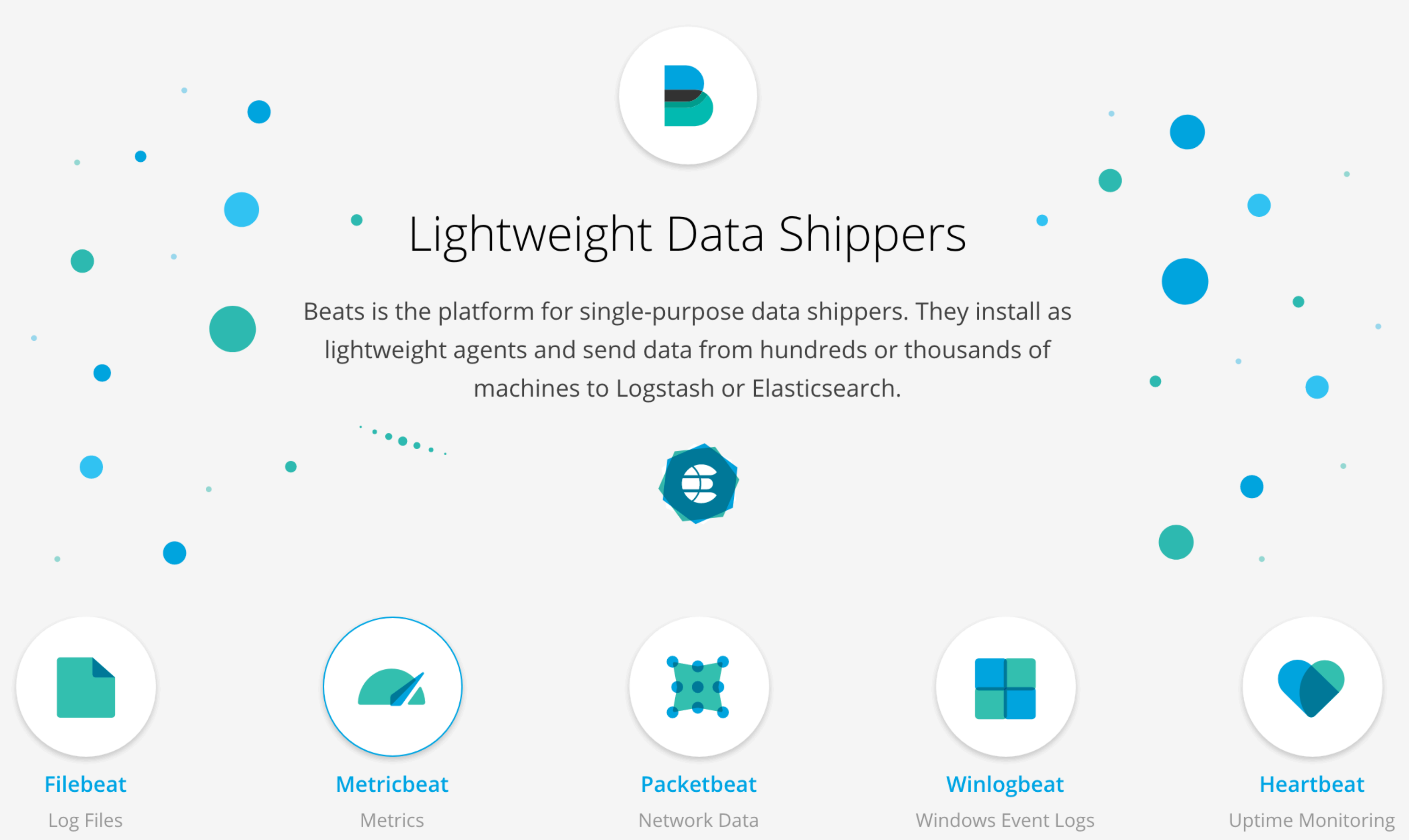
Plug-In

ELK 5.x


light Shipper
x-pack
elasticsearch.
Server
Heartbeat
Metricbeat
Encryption
Port
Read Log File
SYSLOG
UDP 514
Shipper
File
Read Log File
Filebeat
Winlogbeat
Netflow v5,9
UDP
Netflow
Packetbeat
Store & Analysis
Cluster
Plug-In

Beats 相關資訊

- https://www.elastic.co/products/beats



Lightweight Data Shippers

Beats is the platform for single-purpose data shippers. They install as lightweight agents and send data from hundreds or thousands of machines to Logstash or Elasticsearch.

Filebeat
Log Files

Metricbeat
Metrics

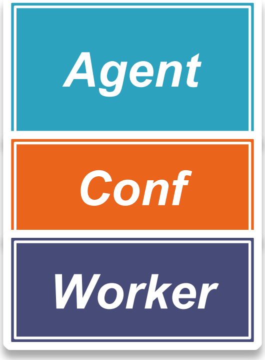
Packetbeat
Network Data

Winlogbeat
Windows Event Logs

Heartbeat
Uptime Monitoring

How to Use LogStash

- Download
  - https://www.elastic.co/downloads/logstash
- Docoument
  - https://www.elastic.co/guide/en/logstash/current/index.html
- Execute(ubuntu)
  - $sudo apt-get install default-jdk (Before Execute)
  - $sudo apt-get install dpkg
  - $dpkg -i logstash-5.x.x.deb
    - /usr/share/logstash
    - /etc/logstash
  - $ cd /usr/share/logstash
  - $ ./bin/logstash -e 'input { stdin { } } output { stdout {} }’



First LogStash Config

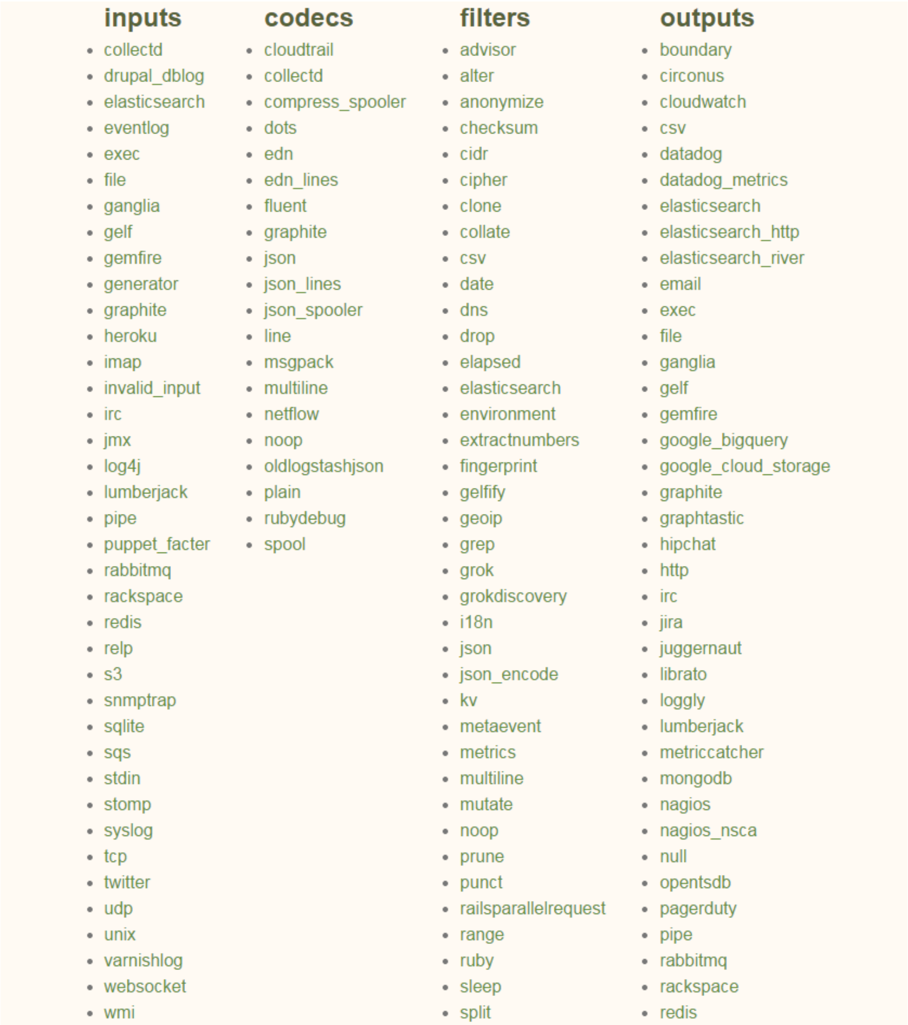
- $touch sample.conf
- $./bin/logstash -f sample.conf -w 2
  - (conf. path) (worker)

sample.conf

```
input {
    stdin { }
}
output {
    stdout {
            codec => rubydebug
    }
}
```

https://www.elastic.co/guide/en/logstash/current/running-logstash-command-line.html

LogStash Config



inputs

- collectd
- drupal_dblog
- elasticsearch
- eventlog
- exec
- file
- ganglia
- gelf
- gemfire
- generator
- graphite
- heroku
- imap
- invalid_input
- irc
- jmx
- log4j
- lumberjack
- pipe
- puppet_facter
- rabbitmq
- rackspace
- redis
- relp
- s3
- snmptrap
- sqlite
- sqs
- stdin
- stomp
- syslog
- tcp
- twitter
- udp
- unix
- varnishlog
- websocket
- wmi

codecs

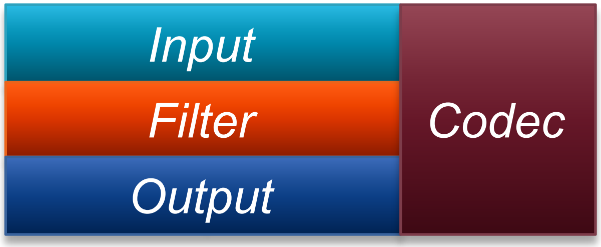
- cloudtrail
- collectd
- compress_spooler
- dots
- edn
- edn_lines
- fluent
- graphite
- json
- json_lines
- json_spooler
- line
- msgpack
- multiline
- netflow
- noop
- oldlogstashjson
- plain
- rubydebug
- spool

filters

- advisor
- alter
- anonymize
- checksum
- cidr
- cipher
- clone
- collate
- csv
- date
- dns
- drop
- elapsed
- elasticsearch
- environment
- extractnumbers
- fingerprint
- gelfify
- geoip
- grep
- grok
- grokdiscovery
- i18n
- json
- json_encode
- kv
- metaevent
- metrics
- multiline
- mutate
- noop
- prune
- punct
- railsparallelrequest
- range
- ruby
- sleep
- split

outputs

- boundary
- circonus
- cloudwatch
- csv
- datadog
- datadog_metrics
- elasticsearch
- elasticsearch_http
- elasticsearch_river
- email
- exec
- file
- ganglia
- gelf
- gemfire
- google_bigquery
- google_cloud_storage
- graphite
- graphtastic
- hipchat
- http
- irc
- jira
- juggernaut
- librato
- loggly
- lumberjack
- metriccatcher
- mongodb
- nagios
- nagios_nsca
- null
- opentsdb
- pagerduty
- pipe
- rabbitmq
- rackspace
- redis



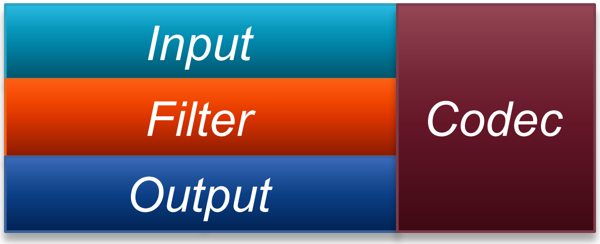
https://www.elastic.co/guide/en/logstash/current/input-plugins.html

https://www.elastic.co/guide/en/logstash/current/output-plugins.html

LogStash Config

- Sample_Strc.conf

```
input {
            file {
                 path => "/tmp/access_log"
                 start_position => beginning
           }
}
filter  {
           grok { match => ["message” , "%{COMMONAPACHELOG}" ]
           }
           date { match => [ "timestamp" , "dd/MMM/yyyy:HH:mm:ss  Z" ]
           }
}
output {
      stdout {
           codec => rubydebug
           }
}
```

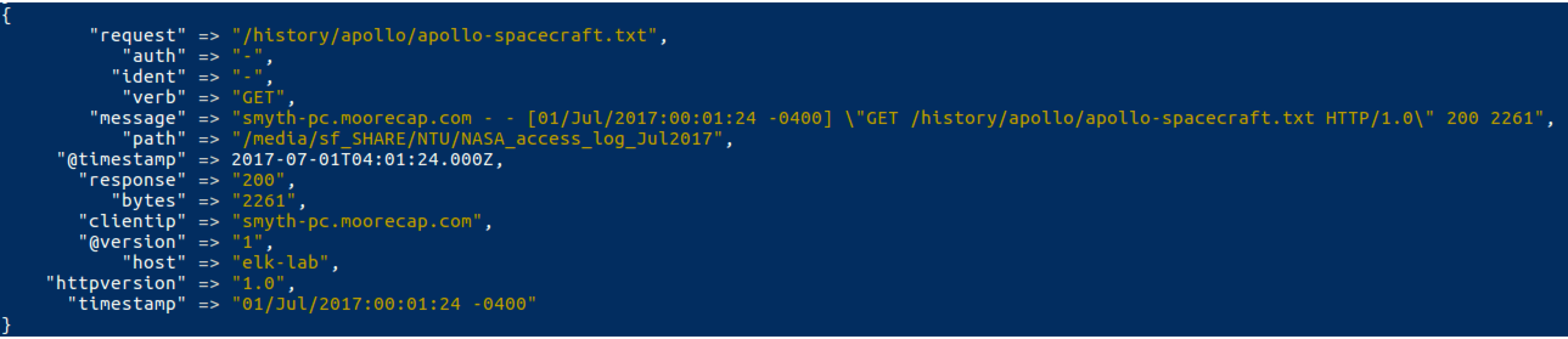

Input
Filter
Output
Codec

LogStash Exec

- Running

./bin/logstash –f Sample_Strc.conf

```
smyth-pc.moorecap.com - - [01/Jul/2017:00:01:24 -0400] "GET
/history/apollo/apollo-spacecraft.txt HTTP/1.0" 200 2261
```



```
{
        "request" => "/history/apollo/apollo-spacecraft.txt",
           "auth" => "-",
          "ident" => "-",
           "verb" => "GET",
        "message" => "smyth-pc.moorecap.com - - [01/Jul/2017:00:01:24 -0400] \"GET /history/apollo/apollo-spacecraft.txt HTTP/1.0\" 200 2261",
           "path" => "/media/sf_SHARE/NTU/NASA_access_log_Jul2017",
     "@timestamp" => 2017-07-01T04:01:24.000Z,
       "response" => "200",
          "bytes" => "2261",
       "clientip" => "smyth-pc.moorecap.com",
       "@version" => "1",
           "host" => "elk-lab",
    "httpversion" => "1.0",
      "timestamp" => "01/Jul/2017:00:01:24 -0400"
}
```

INPUT

stdin { }

File

Read Log File

```
input {
            file {
            path => "/log/access_log"
            start_position => beginning   Default: 1s
            sincedb_path => "/log/access_log_postision.db"
            }
}
```

Port

TCP 1234 SSL

```
input {
            syslog {
            port => 514
            }
            tcp {
            port  => 1234
            mode => "server"
            ssl_enable => false
            }
}
```

$nc 127.0.0.1 1234 < file.log

Netflow

Netflow
v5,9
UDP

```
input {
            udp {
            port  => 8888
            codec => netflow
            }
}
```

OUTPUT

Stdout

```
output {
            stdout {
            codec => rubydebug
            worker => 2
            }
}
```

Store Log File

```
output {
            file {
            path => \log\sample.log
            message_format => “%{message}”
            }
}
```

ElasticSearch

```
output {
            elasticsearch{
            host => "localhost"
            index => “sample"
            index_type => “sample_event"
            cluster => “sample"
            protocol => "http"
            workers => 1
            }
}
```

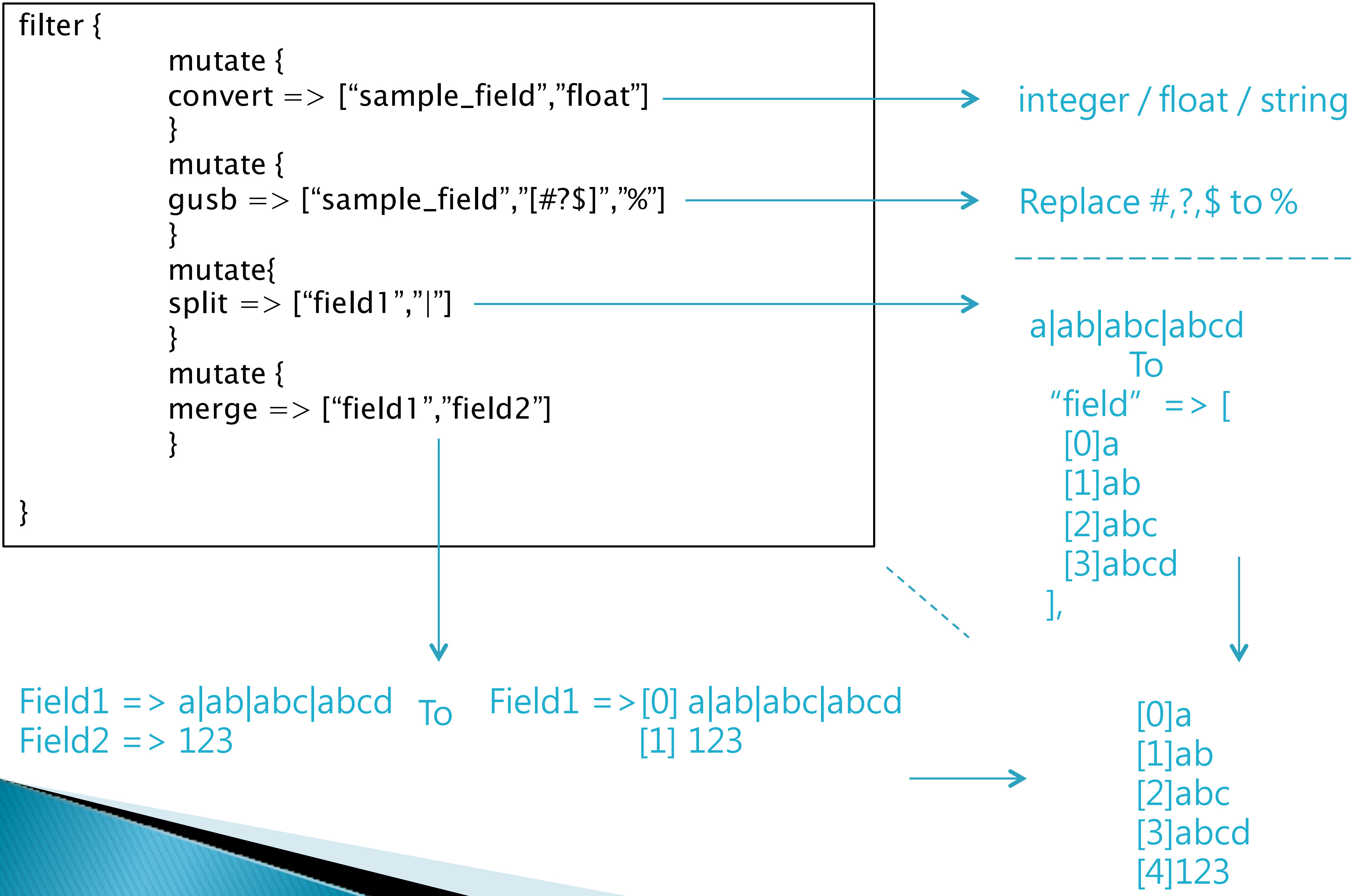
Lab 1

- A. Input stdin/ output stdout
- B. Input file / output file
  - 1. set “start_position”
  - 2. set “sincedb_path”
  - 3. set output file “path”
- C. Input file / output file
  - 1. output file (message field only)
    - Hint: “message_format”
- D. Output to ElasticSearch

FILTER

- In common use
  - mutate
    - The mutate filter allows you to do general mutations to fields. You can rename, remove, replace, and modify fields in your events.
  - grep
    - Useful for dropping events you don’t want to pass, or adding tags or fields to events that match.
  - date
    - parsing dates from fields "timestamp" , "dd/MMM/yyyy:HH:mm:ss Z“
  - geoip
    - adds information about geographical location of IP addresses
  - grok
    - parses arbitrary text and structure it.

Mutate



```
filter {
                mutate {
                convert => ["sample_field","float"]
                }
                mutate {
                gusb => ["sample_field","[#?$]","%"]
                }
                mutate{
                split => ["field1","|"]
                }
                mutate {
                merge => ["field1","field2"]
                }

}
```

integer / float / string

Replace #,?,$ to %

a|ab|abc|abcd
To
"field" => [
[0]a
[1]ab
[2]abc
[3]abcd
],

Field1 => a|ab|abc|abcd
Field2 => 123

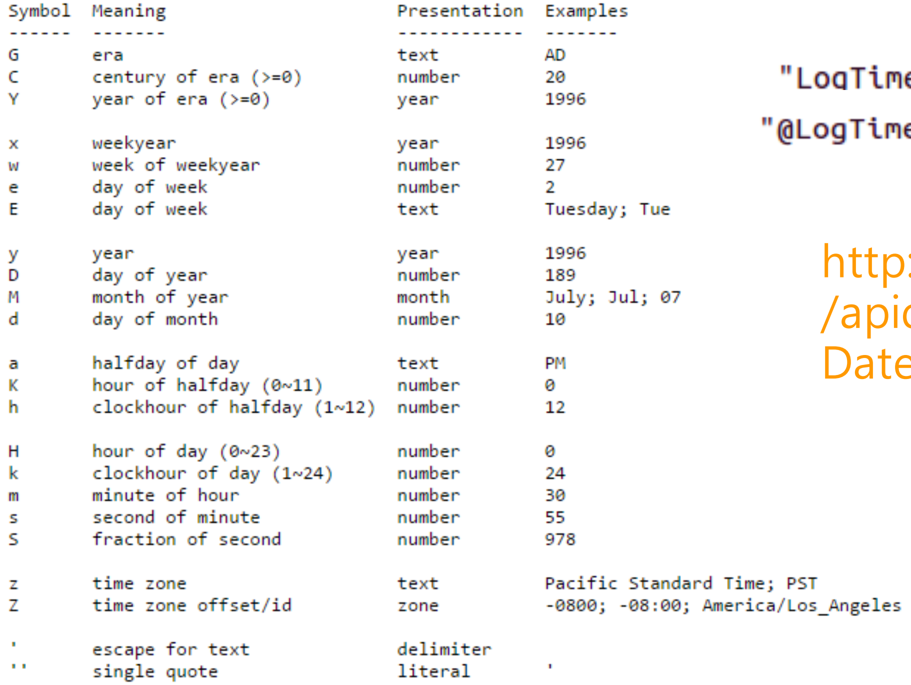
To

Field1 => [0] a|ab|abc|abcd
[1] 123

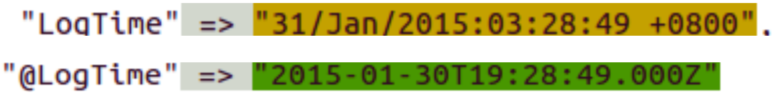
[0]a
[1]ab
[2]abc
[3]abcd
[4]123

Date

```
filter {
            date {
            match => [“LogTime", dd/MMM/yyyy:HH:mm:ss  Z]
            target => "@LogTime"
            }
}
```



Symbol	Meaning	Presentation	Examples
G	era	text	AD
C	century of era (>=0)	number	20
Y	year of era (>=0)	year	1996
x	weekyear	year	1996
w	week of weekyear	number	27
e	day of week	number	2
E	day of week	text	Tuesday; Tue
y	year	year	1996
D	day of year	number	189
M	month of year	month	July; Jul; 07
d	day of month	number	10
a	halfday of day	text	PM
K	hour of halfday (0~11)	number	0
h	clockhour of halfday (1~12)	number	12
H	hour of day (0~23)	number	0
k	clockhour of day (1~24)	number	24
m	minute of hour	number	30
s	second of minute	number	55
S	fraction of second	number	978
z	time zone	text	Pacific Standard Time; PST
Z	time zone offset/id	zone	-0800; -08:00; America/Los_Angeles
'	escape for text	delimiter	
''	single quote	literal	'



```
"LogTime"  => "31/Jan/2015:03:28:49 +0800".
"@LogTime" => "2015-01-30T19:28:49.000Z"
```

http://joda-time.sourceforge.net/apidocs/org/joda/time/format/DateTimeFormat.html

Geoip

```
filter {
        geoip {
        source=> “SourceIP”
         target => "geoip"
         database =>"/opt/logstash/vendor/geoip/GeoLiteCity.dat"
         add_field => [ "[geoip][coordinates]", "%{[geoip][longitude]}" ]
         add_field => [ "[geoip][coordinates]", "%{[geoip][latitude]}"  ]
}
```

加拿大
北大西洋
北太平洋
南太平洋
南大西洋
印度洋
俄羅斯
中國
日本
巴西
820
695
775
556
1.60k
1.02k
5.06k

```
"geoip" =>
  {
   "ip" => "140.128.0.1",
   "country_code2" =>   “TW",
   "country_code3" =>   “TWN",
   "country_name" =>   “Taiwan",
   "continent_code" =>   “TW",
   "region_name" =>   “29",
   "city_name" =>   “Taipei",
   "latitude" => 23.97387500000001,
   "longitude" => 120.982024,
   "timezone" => "Asia/Taipei",
   "real_region_name" =>   “Taipei",
   "location" => [
        [0] 120.982024,
        [1] 23.97387500000001
   }
```

Before Use Grok/Grep

We Must to Know >

- What is the “Regular expression”?

(REGEX)

How REGEX work?

- REGEX
  - **regular expression**
    - A sequence of characters that forms a search pattern, mainly for use in pattern matching with strings, or string matching, i.e. "find and replace"-like operations
  - https://msdn.microsoft.com/zh-tw/library/az24scfc.aspx#character_classes
- Learn
  - https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html
- Debug
  - https://github.com/elastic/logstash/tree/v1.4.2/patterns
  - http://grokdebug.herokuapp.com/
  - https://regex101.com/

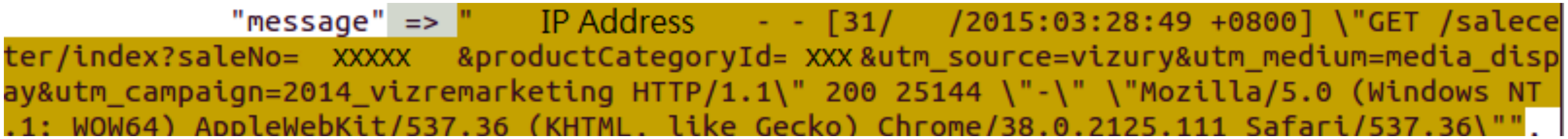
Grep/Grok

```
filter {
            grep {
            match => ["message”,“keyword“]
            }
}
```

Default: Drop Not Match

```
filter {
            grok {
            match => ["message" , “%{Field_Name:REGEX}“]
            }
}
```

%{DATA:IP}\s.*?\[%{DATA:LogTime}]\s\"%{DATA:Action}\s%{DATA:URI}\s%{DATA:Protocol}\"\s%{DATA:Status}\s%{DATA:Size}\s\"%{DATA:URL}\"\s\"%{DATA:Browser}\"

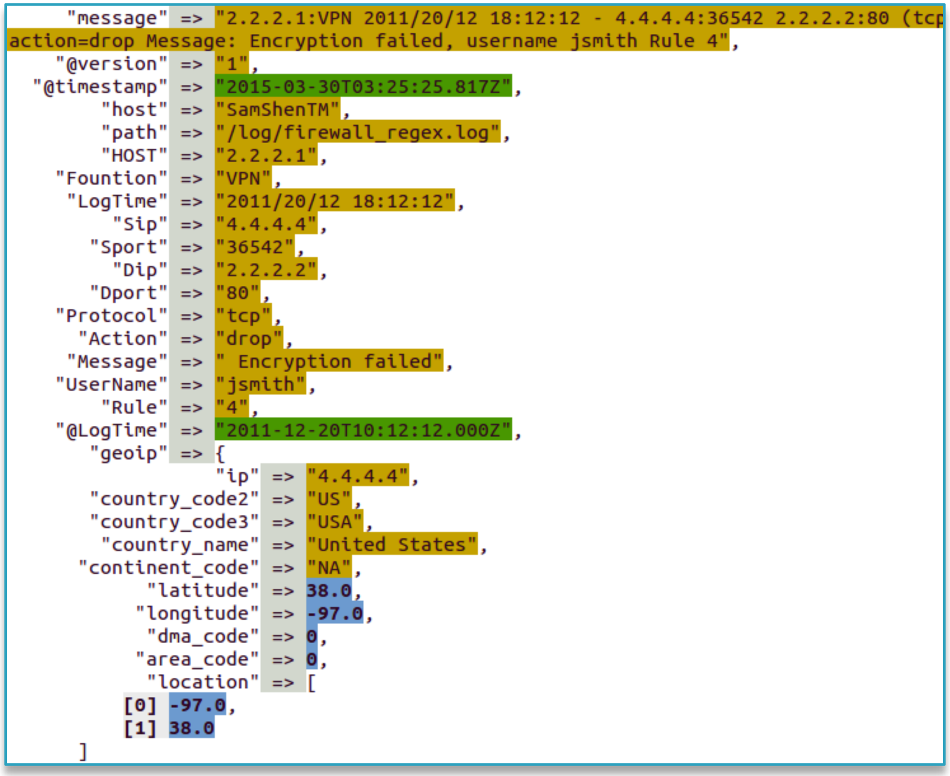


```
              "message" => "     IP Address     - - [31/    /2015:03:28:49 +0800] \"GET /salece
ter/index?saleNo=  XXXXX   &productCategoryId= XXX &utm_source=vizury&utm_medium=media_disp
ay&utm_campaign=2014_vizremarketing HTTP/1.1\" 200 25144 \"-\" \"Mozilla/5.0 (Windows NT
.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.111 Safari/537.36\"",
```

http://grokdebug.herokuapp.com/

Lab 2

- A. Parsing Firewall Sample log
  - 1. Grep “VPN”
  - 2. Grok VPN log “All fields”
  - 3. Create log time date
  - 4. Geoip SourceIP



```
       "message" => "2.2.2.1:VPN 2011/20/12 18:12:12 - 4.4.4.4:36542 2.2.2.2:80 (tcp
action=drop Message: Encryption failed, username jsmith Rule 4",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:25:25.817Z",
          "host" => "SamShenTM",
          "path" => "/log/firewall_regex.log",
          "HOST" => "2.2.2.1",
      "Fountion" => "VPN",
       "LogTime" => "2011/20/12 18:12:12",
           "Sip" => "4.4.4.4",
         "Sport" => "36542",
           "Dip" => "2.2.2.2",
         "Dport" => "80",
      "Protocol" => "tcp",
        "Action" => "drop",
       "Message" => "  Encryption failed",
      "UserName" => "jsmith",
          "Rule" => "4",
      "@LogTime" => "2011-12-20T10:12:12.000Z",
         "geoip" => {
                    "ip" => "4.4.4.4",
         "country_code2" => "US",
         "country_code3" => "USA",
          "country_name" => "United States",
        "continent_code" => "NA",
              "latitude" => 38.0,
             "longitude" => -97.0,
              "dma_code" => 0,
             "area_code" => 0,
              "location" => [
            [0] -97.0,
            [1] 38.0
        ]
```

Lab 2 Answer

- A. Parsing Firewall Sample log
  - 1. Grep “VPN”
    ```
    grep { match => ["message",".*:VPN”] }
    ```
  - 2. Grok VPN log “All fields”
    ```
    grok { match => ["message","%{DATA:HOST}\:%{DATA:Fountion}\s%{DATA:LogTime}\s-\s%{DATA:Sip}\:%{DATA:Sport}\s%{DATA:Dip}\:%{DATA:Dport}\s\(%{DATA:Protocol}\)\saction=%{DATA:Action}\sMessage:%{DATA:Message},\susername\s%{DATA:UserName}\sRule\s%{GREEDYDATA:Rule}"] }
    ```
  - 3. Create log time date
    ```
    Date { match => ["LogTime", "yyyy/dd/MM HH:mm:ss”] target => "@LogTime" }
    ```
  - 4. Geoip SourceIP
    ```
    Geoip { source => "Sip" }
    ```

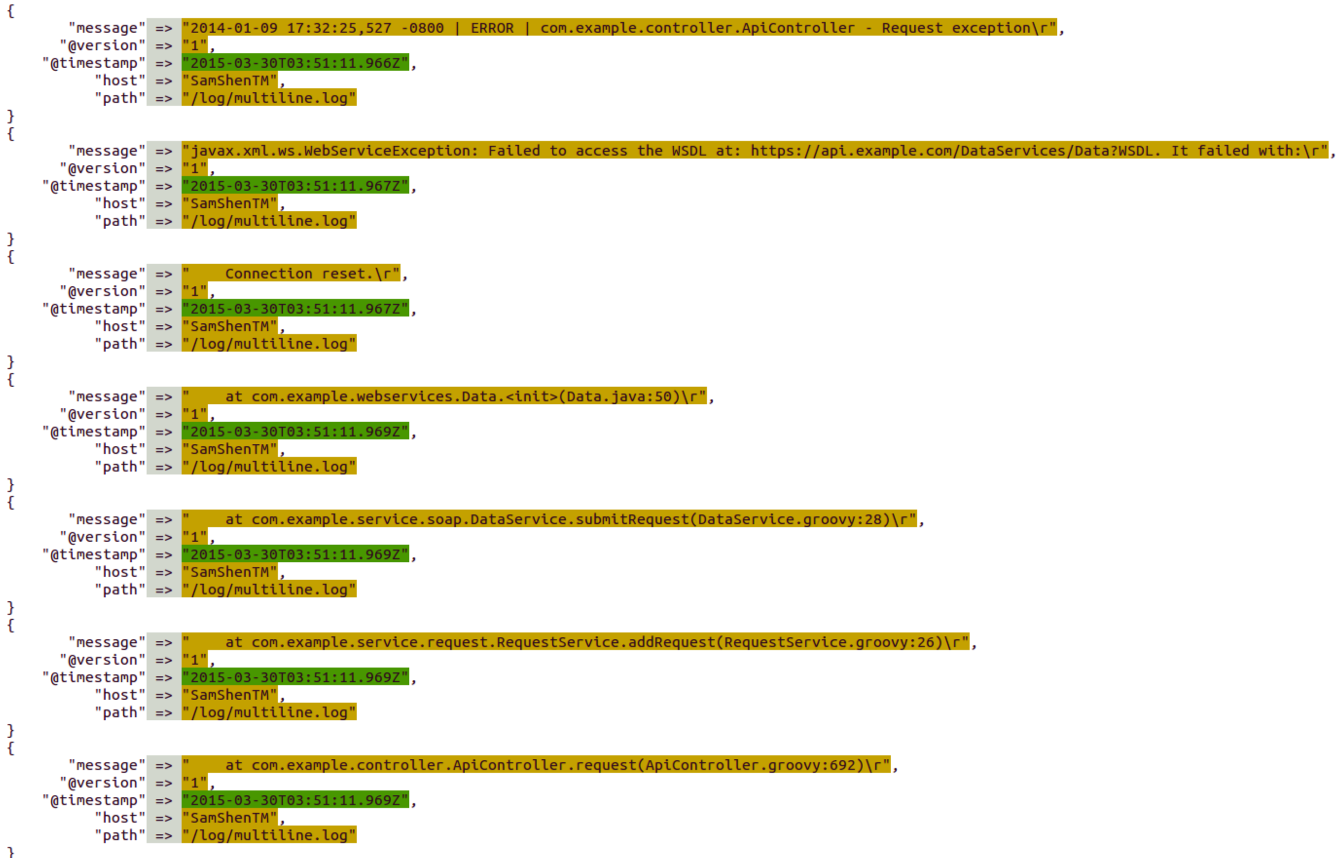
Multi-Line Prob.

- Do you ever think….log like this:

```
2014-01-09 17:32:25,527 -0800 | ERROR | com.example.controller.ApiController - Request exception
javax.xml.ws.WebServiceException: Failed to access the WSDL at:
https://api.example.com/DataServices/Data?WSDL. It failed with:
    Connection reset.
    at com.example.webservices.Data.<init>(Data.java:50)
    at com.example.service.soap.DataService.submitRequest(DataService.groovy:28)
    at com.example.service.request.RequestService.addRequest(RequestService.groovy:26)
    at com.example.controller.ApiController.request(ApiController.groovy:692)
    at grails.plugin.cache.web.filter.PageFragmentCachingFilter.doFilter(PageFragmentCachingFilter.java:200)
    at grails.plugin.cache.web.filter.AbstractFilter.doFilter(AbstractFilter.java:63)
    at org.apache.jk.server.JkCoyoteHandler.invoke(JkCoyoteHandler.java:190)
    at org.apache.jk.common.HandlerRequest.invoke(HandlerRequest.java:311)
    at org.apache.jk.common.ChannelSocket.invoke(ChannelSocket.java:776)
    at org.apache.jk.common.ChannelSocket.processConnection(ChannelSocket.java:705)
    at org.apache.jk.common.ChannelSocket$SocketConnection.runIt(ChannelSocket.java:898)
Caused by: java.net.SocketException: Connection reset
    ... 17 more
```

Multi–Line Prob.

- After “Normal Parsing” like this…



```
{
       "message" => "2014-01-09 17:32:25,527 -0800 | ERROR | com.example.controller.ApiController - Request exception\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.966Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "javax.xml.ws.WebServiceException: Failed to access the WSDL at: https://api.example.com/DataServices/Data?WSDL. It failed with:\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.967Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "    Connection reset.\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.967Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "    at com.example.webservices.Data.<init>(Data.java:50)\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.969Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "    at com.example.service.soap.DataService.submitRequest(DataService.groovy:28)\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.969Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "    at com.example.service.request.RequestService.addRequest(RequestService.groovy:26)\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.969Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
{
       "message" => "    at com.example.controller.ApiController.request(ApiController.groovy:692)\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T03:51:11.969Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log"
}
```

Multiline

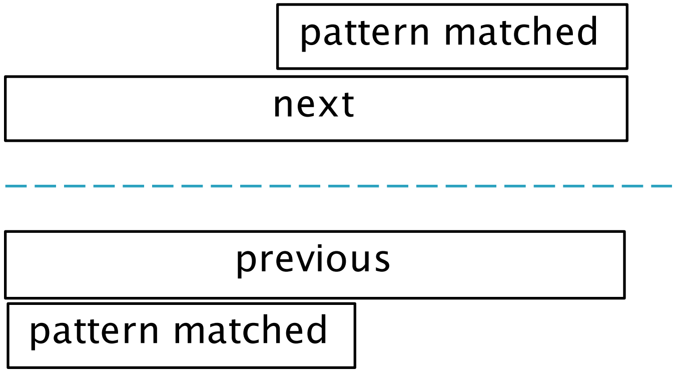
```
filter {
        multiline {
                negate => "true”

                what => "previous”

                pattern => ”REGEX”
                }
}
```

Default: false
Negate the regexp pattern
(if not matched, stop/normal conti.)

If the pattern matched, does event belong to the next or previous event?

position

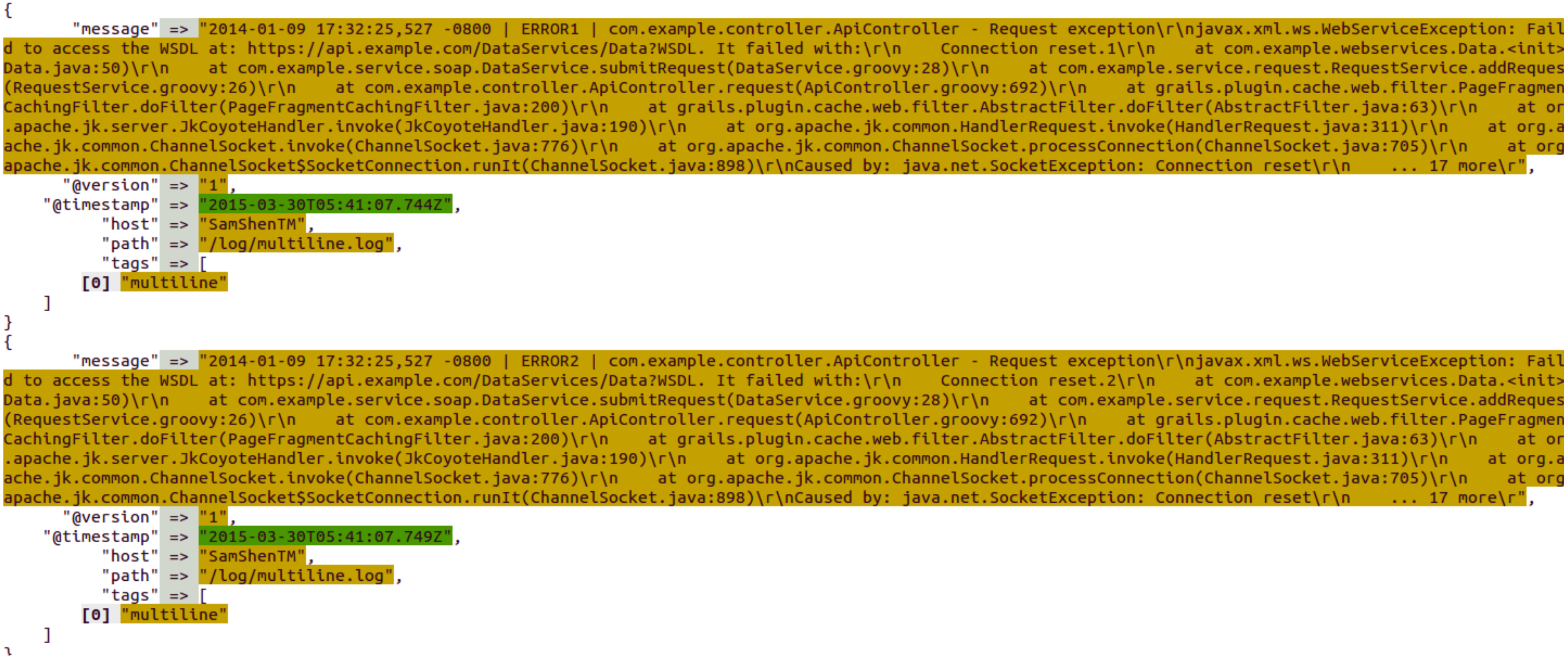
\d+-\d+-\d+\s\d+:\d+:\d+,



```
2014-01-09 17:32:25,527 -0800 | ERROR1 | com.example.controller.ApiController  - Request exception
javax.xml.ws.WebServiceException:  Failed  to access  the WSDL at: https://api.example.com/DataServices/
Data?WSDL.  It failed  with:
    Connection reset.1
    at com.example.webservices.Data.<init>(Data.java:50)
    at com.example.service.soap.DataService.submitRequest(DataService.groovy:28)
    at com.example.service.request.RequestService.addRequest(RequestService.groovy:26)
    at com.example.controller.ApiController.request(ApiController.groovy:692)
    at grails.plugin.cache.web.filter.PageFragmentCachingFilter.doFilter(PageFragmentCachingFilter.java:200)
    at grails.plugin.cache.web.filter.AbstractFilter.doFilter(AbstractFilter.java:63)
    at org.apache.jk.server.JkCoyoteHandler.invoke(JkCoyoteHandler.java:190)
    at org.apache.jk.common.HandlerRequest.invoke(HandlerRequest.java:311)
    at org.apache.jk.common.ChannelSocket.invoke(ChannelSocket.java:776)
    at org.apache.jk.common.ChannelSocket.processConnection(ChannelSocket.java:705)
    at org.apache.jk.common.ChannelSocket$SocketConnection.runIt(ChannelSocket.java:898)
Caused by: java.net.SocketException:  Connection  reset
    … 17 more
```

Multiline

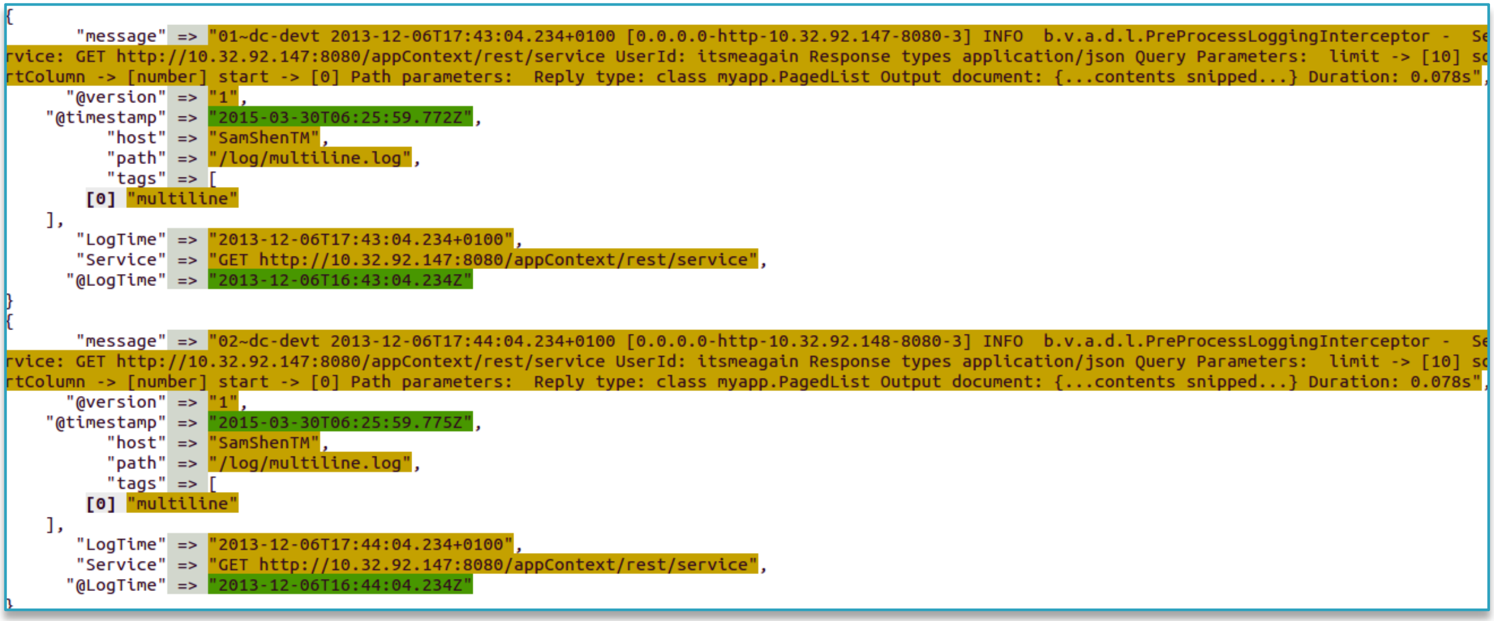
- After multiline filter



```
{
       "message" => "2014-01-09 17:32:25,527 -0800 | ERROR1 | com.example.controller.ApiController - Request exception\r\njavax.xml.ws.WebServiceException: Fail
d to access the WSDL at: https://api.example.com/DataServices/Data?WSDL. It failed with:\r\n    Connection reset.1\r\n    at com.example.webservices.Data.<init>
Data.java:50)\r\n    at com.example.service.soap.DataService.submitRequest(DataService.groovy:28)\r\n    at com.example.service.request.RequestService.addReques
(RequestService.groovy:26)\r\n    at com.example.controller.ApiController.request(ApiController.groovy:692)\r\n    at grails.plugin.cache.web.filter.PageFragmen
CachingFilter.doFilter(PageFragmentCachingFilter.java:200)\r\n    at grails.plugin.cache.web.filter.AbstractFilter.doFilter(AbstractFilter.java:63)\r\n    at or
.apache.jk.server.JkCoyoteHandler.invoke(JkCoyoteHandler.java:190)\r\n    at org.apache.jk.common.HandlerRequest.invoke(HandlerRequest.java:311)\r\n    at org.a
ache.jk.common.ChannelSocket.invoke(ChannelSocket.java:776)\r\n    at org.apache.jk.common.ChannelSocket.processConnection(ChannelSocket.java:705)\r\n    at org
apache.jk.common.ChannelSocket$SocketConnection.runIt(ChannelSocket.java:898)\r\nCaused by: java.net.SocketException: Connection reset\r\n    ... 17 more\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T05:41:07.744Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log",
          "tags" => [
        [0] "multiline"
    ]
}
{
       "message" => "2014-01-09 17:32:25,527 -0800 | ERROR2 | com.example.controller.ApiController - Request exception\r\njavax.xml.ws.WebServiceException: Fail
d to access the WSDL at: https://api.example.com/DataServices/Data?WSDL. It failed with:\r\n    Connection reset.2\r\n    at com.example.webservices.Data.<init>
Data.java:50)\r\n    at com.example.service.soap.DataService.submitRequest(DataService.groovy:28)\r\n    at com.example.service.request.RequestService.addReques
(RequestService.groovy:26)\r\n    at com.example.controller.ApiController.request(ApiController.groovy:692)\r\n    at grails.plugin.cache.web.filter.PageFragmen
CachingFilter.doFilter(PageFragmentCachingFilter.java:200)\r\n    at grails.plugin.cache.web.filter.AbstractFilter.doFilter(AbstractFilter.java:63)\r\n    at or
.apache.jk.server.JkCoyoteHandler.invoke(JkCoyoteHandler.java:190)\r\n    at org.apache.jk.common.HandlerRequest.invoke(HandlerRequest.java:311)\r\n    at org.a
ache.jk.common.ChannelSocket.invoke(ChannelSocket.java:776)\r\n    at org.apache.jk.common.ChannelSocket.processConnection(ChannelSocket.java:705)\r\n    at org
apache.jk.common.ChannelSocket$SocketConnection.runIt(ChannelSocket.java:898)\r\nCaused by: java.net.SocketException: Connection reset\r\n    ... 17 more\r",
      "@version" => "1",
    "@timestamp" => "2015-03-30T05:41:07.749Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log",
          "tags" => [
        [0] "multiline"
    ]
}
```

Lab 3

- Use multiline of filter
  - Grok => LogTime, Service (limit 1 grok)
  - Create log time date



```
{
       "message" => "01~dc-devt 2013-12-06T17:43:04.234+0100 [0.0.0.0-http-10.32.92.147-8080-3] INFO  b.v.a.d.l.PreProcessLoggingInterceptor -  Se
rvice: GET http://10.32.92.147:8080/appContext/rest/service UserId: itsmeagain Response types application/json Query Parameters:  limit -> [10] so
rtColumn -> [number] start -> [0] Path parameters:  Reply type: class myapp.PagedList Output document: {...contents snipped...} Duration: 0.078s",
      "@version" => "1",
    "@timestamp" => "2015-03-30T06:25:59.772Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log",
          "tags" => [
        [0] "multiline"
    ],
       "LogTime" => "2013-12-06T17:43:04.234+0100",
       "Service" => "GET http://10.32.92.147:8080/appContext/rest/service",
      "@LogTime" => "2013-12-06T16:43:04.234Z"
}
{
       "message" => "02~dc-devt 2013-12-06T17:44:04.234+0100 [0.0.0.0-http-10.32.92.148-8080-3] INFO  b.v.a.d.l.PreProcessLoggingInterceptor -  Se
rvice: GET http://10.32.92.147:8080/appContext/rest/service UserId: itsmeagain Response types application/json Query Parameters:  limit -> [10] so
rtColumn -> [number] start -> [0] Path parameters:  Reply type: class myapp.PagedList Output document: {...contents snipped...} Duration: 0.078s",
      "@version" => "1",
    "@timestamp" => "2015-03-30T06:25:59.775Z",
          "host" => "SamShenTM",
          "path" => "/log/multiline.log",
          "tags" => [
        [0] "multiline"
    ],
       "LogTime" => "2013-12-06T17:44:04.234+0100",
       "Service" => "GET http://10.32.92.147:8080/appContext/rest/service",
      "@LogTime" => "2013-12-06T16:44:04.234Z"
}
```

Lab 3 Answer

- Use multiline of filter
  - multiline { negate => "true” pattern => "\d+\~\w+-devt\s” what => "previous” }
  - Grok => LogTime, Service (limit 1 grok)
    - grok { match => ["message","\d+\~\w+\-\w+\s%{DATA:LogTime}\s.*?Service:\s%{DATA:Service}\sUserId:”] }
  - Create log time date
    - date { match =>["LogTime", "yyyy-MM-dd'T'HH:mm:ss.SSSZ"] target => "@LogTime” }

Elastic Search

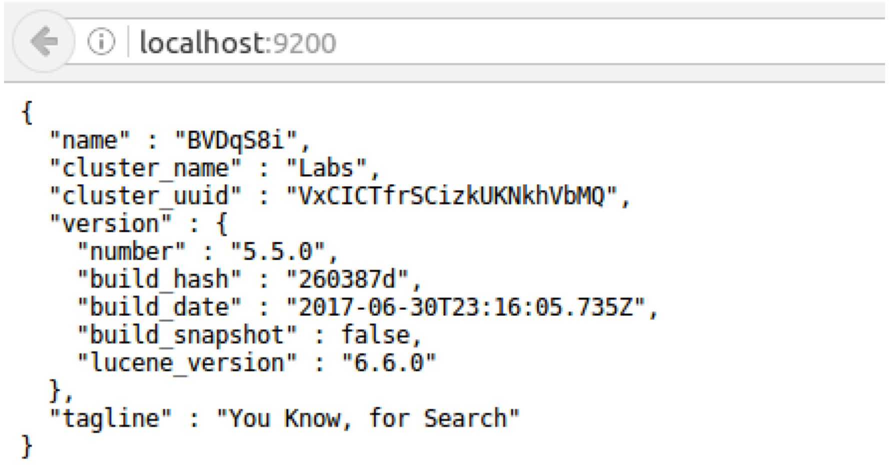
- Download
  - https://www.elastic.co/downloads/elasticsearch
- Docoument
  - https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html
- Execute(ubuntu)
  - $dpkg –i elasticearch-5.x.x.deb
    - /usr/share/elasticsearch
    - /etc/elasticsearch
      - elasticsearch.yml
      - jvm.options
      - log4j2.properties
  - $./bin/elasticsearch
      - localhost:9200

ES Config

- elasticsearch.yml
  - vi /etc/elasticsearch/elasticsearch.yml
    - path.data
    - path.logs
    - http.port

vi /etc/elasticsearch/jvm.options

- jvm.options
  - vi /etc/elasticsearch/jvm.options
    - heap_size



```
{
  "name" : "BVDqS8i",
  "cluster_name" : "Labs",
  "cluster_uuid" : "VxCICTfrSCizkUKNkhVbMQ",
  "version" : {
    "number" : "5.5.0",
    "build_hash" : "260387d",
    "build_date" : "2017-06-30T23:16:05.735Z",
    "build_snapshot" : false,
    "lucene_version" : "6.6.0"
  },
  "tagline" : "You Know, for Search"
}
```

Elastic Basic Command

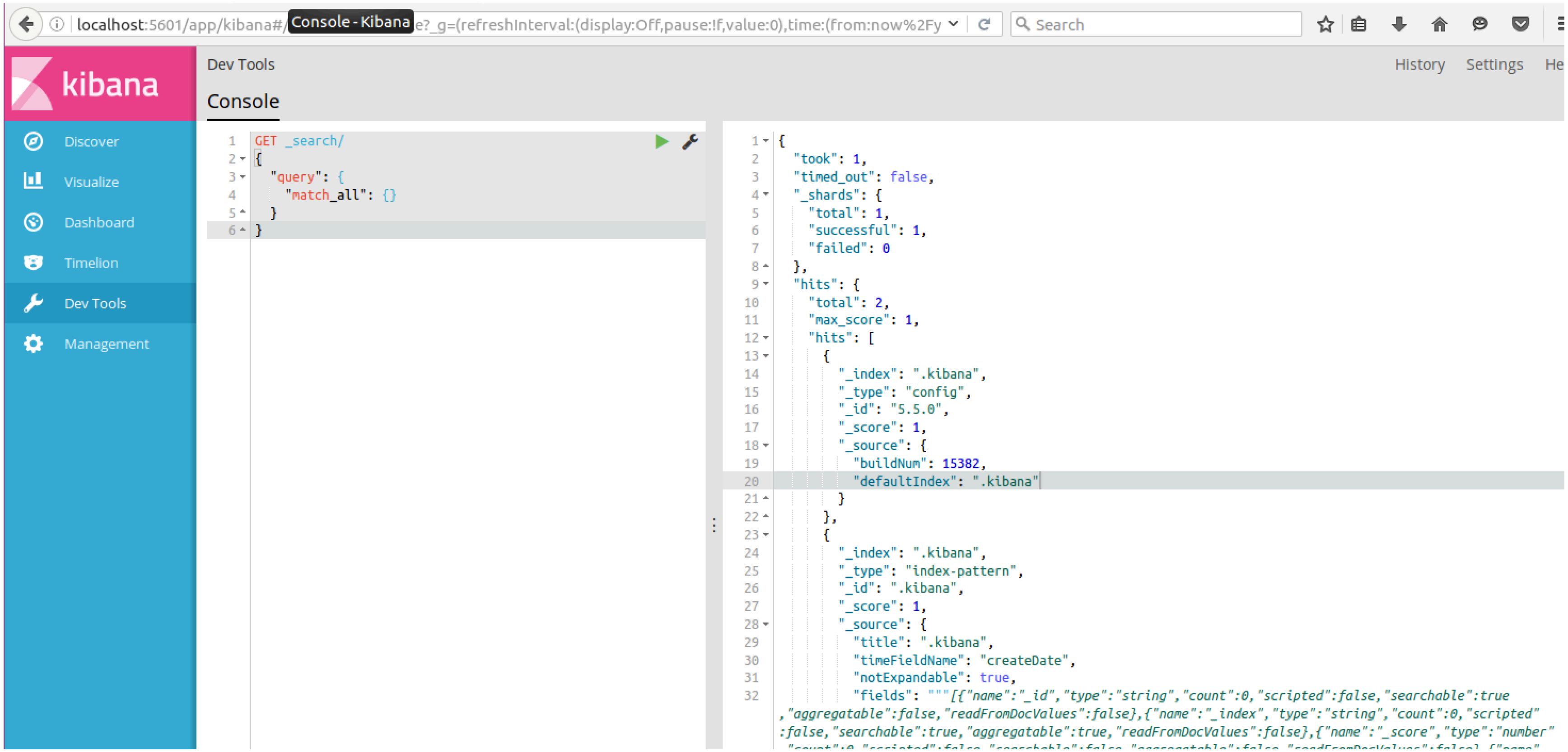
- 啟動狀態

  -
    ```
    curl –i –XGET ‘localhost:9200’
    ```
- 索引列表

  -
    ```
    curl http://localhost:9200/_cat/indices?v
    ```
- 刪除索引

  -
    ```
    curl –XDELETE http://localhost:9200/index.name
    ```

ES API Console



```
GET _search/
{
  "query": {
    "match_all": {}
  }
}
```

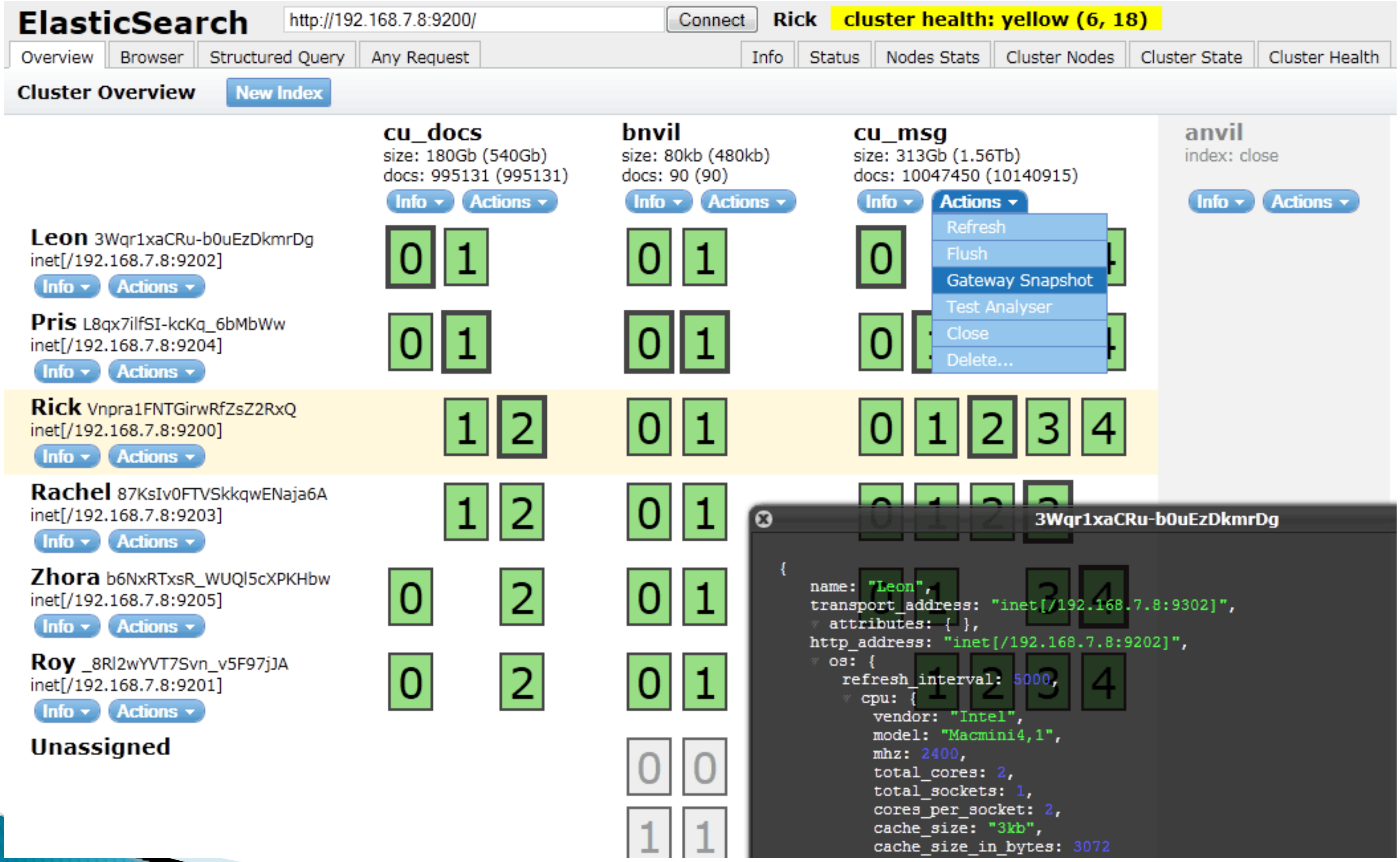
```
{
  "took": 1,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "failed": 0
  },
  "hits": {
    "total": 2,
    "max_score": 1,
    "hits": [
      {
        "_index": ".kibana",
        "_type": "config",
        "_id": "5.5.0",
        "_score": 1,
        "_source": {
          "buildNum": 15382,
          "defaultIndex": ".kibana"
        }
      },
      {
        "_index": ".kibana",
        "_type": "index-pattern",
        "_id": ".kibana",
        "_score": 1,
        "_source": {
          "title": ".kibana",
          "timeFieldName": "createDate",
          "notExpandable": true,
          "fields": """[{"name":"_id","type":"string","count":0,"scripted":false,"searchable":true
,"aggregatable":false,"readFromDocValues":false},{"name":"_index","type":"string","count":0,"scripted"
:false,"searchable":true,"aggregatable":true,"readFromDocValues":false},{"name":"_score","type":"number"
,"count":0,"scripted":false,"searchable":false,"aggregatable":false,"readFromDocValues":false},{"name"
```

https://www.elastic.co/guide/en/elasticsearch/reference/current/docs.html

ES Plugin – head

Running with built in server

- `git clone git://github.com/mobz/elasticsearch-head.git`
- `cd elasticsearch-head`
- `npm install`
- `npm run start`
- `open` http://localhost:9100/



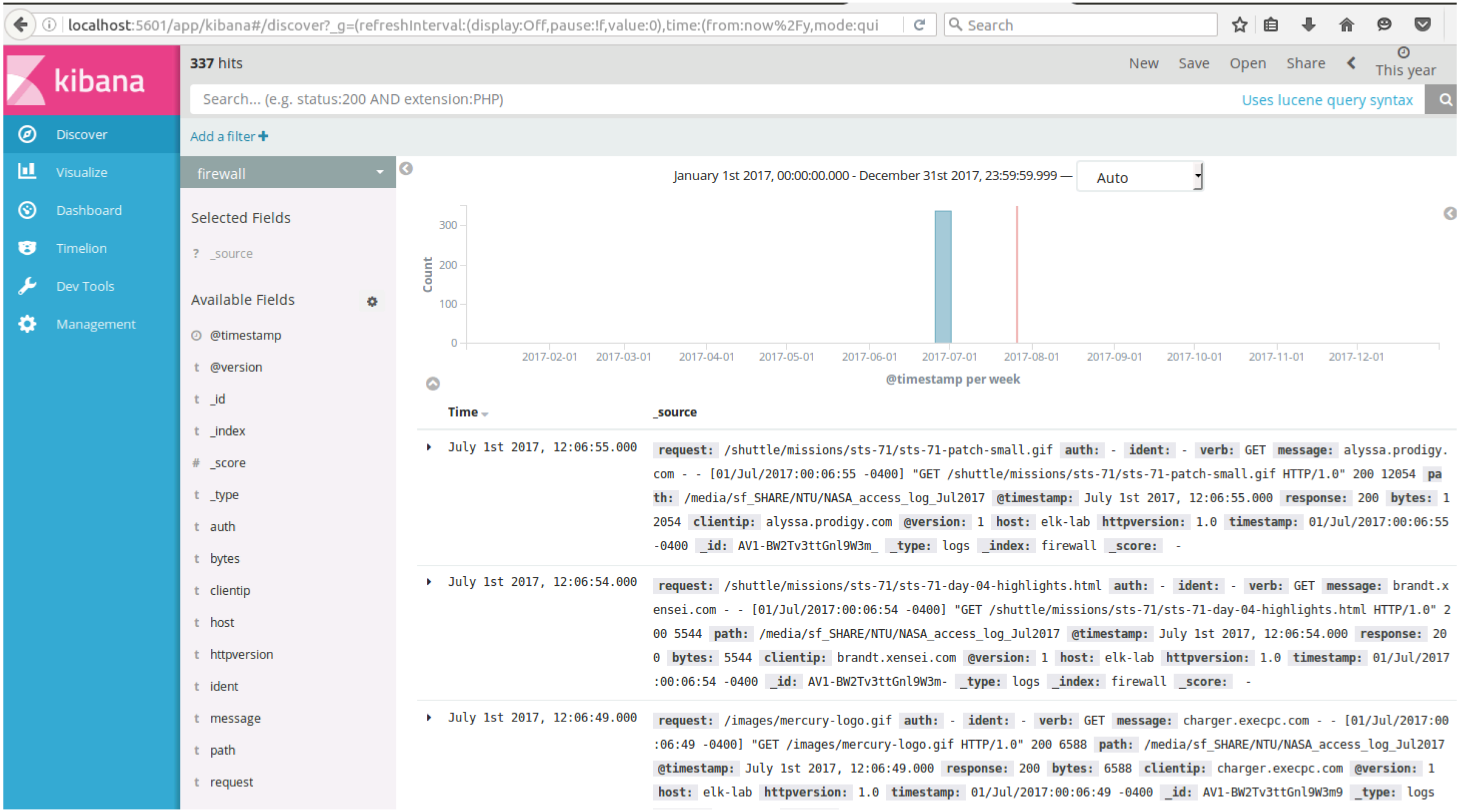
	cu_docs size: 180Gb (540Gb) docs: 995131 (995131)	bnvil size: 80kb (480kb) docs: 90 (90)	cu_msg size: 313Gb (1.56Tb) docs: 10047450 (10140915)	anvil index: close
Leon 3Wqr1xaCRu-b0uEzDkmrDg inet[/192.168.7.8:9202]	0 1	0 1	0 4	
Pris L8qx7ilfSI-kcKq_6bMbWw inet[/192.168.7.8:9204]	0 1	0 1	0 4	
Rick Vnpra1FNTGirwRfZsZ2RxQ inet[/192.168.7.8:9200]	1 2	0 1	0 1 2 3 4	
Rachel 87KsIv0FTVSkkqwENaja6A inet[/192.168.7.8:9203]	1 2	0 1	0 1 2 3	
Zhora b6NxRTxsR_WUQl5cXPKHbw inet[/192.168.7.8:9205]	0 2	0 1	0 1 3 4	
Roy _8Rl2wYVT7Svn_v5F97jJA inet[/192.168.7.8:9201]	0 2	0 1	1 2 3 4	
Unassigned		0 0 1 1		

3Wqr1xaCRu-b0uEzDkmrDg

```
{
    name: "Leon",
    transport_address: "inet[/192.168.7.8:9302]",
    attributes: { },
    http_address: "inet[/192.168.7.8:9202]",
    os: {
      refresh_interval: 5000,
      cpu: {
        vendor: "Intel",
        model: "Macmini4,1",
        mhz: 2400,
        total_cores: 2,
        total_sockets: 1,
        cores_per_socket: 2,
        cache_size: "3kb",
        cache_size_in_bytes: 3072
```

npm, nodejs, nodejs-legacy

ES Index Mgt.



localhost:5601/app/kibana#/discover?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now%2Fy,mode:qui

337 hits

January 1st 2017, 00:00:00.000 - December 31st 2017, 23:59:59.999 —

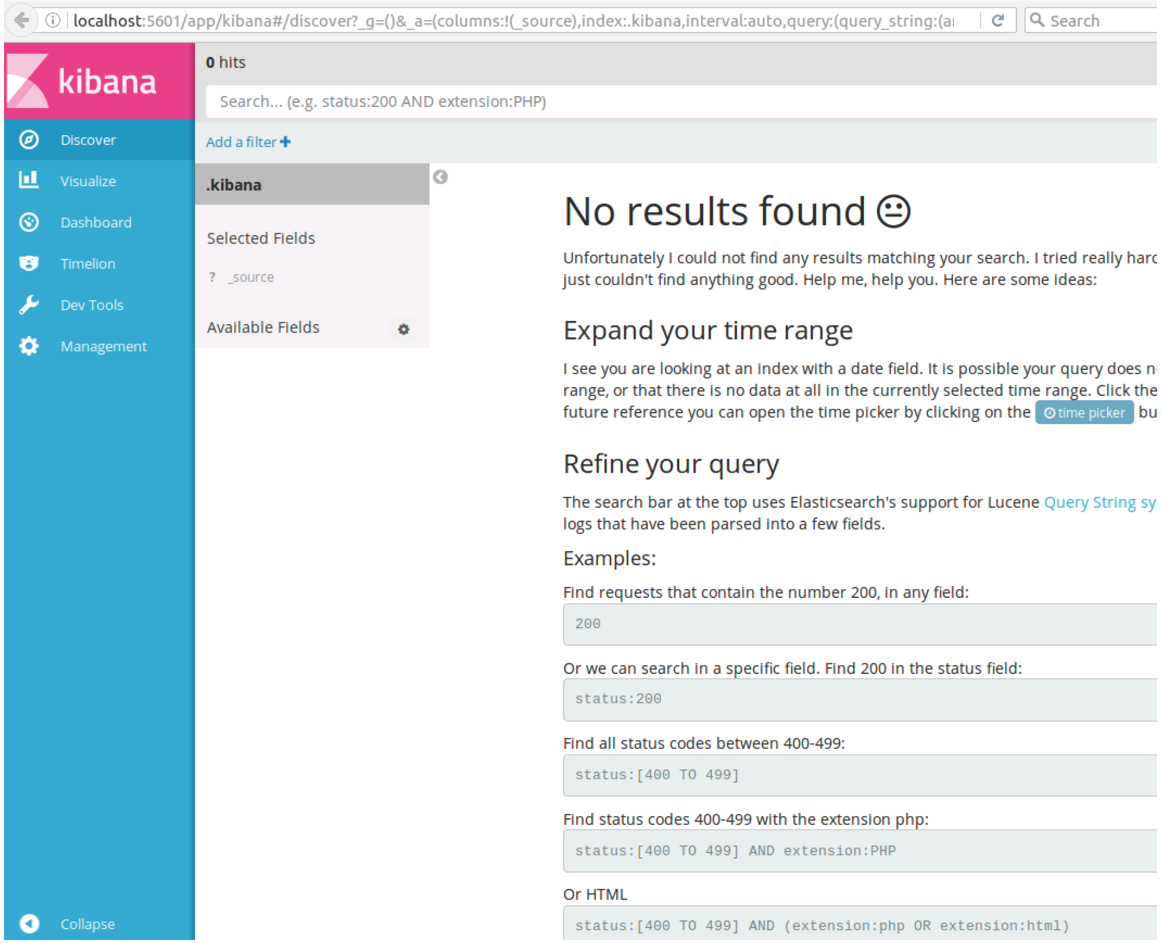
```
Time                              _source
July 1st 2017, 12:06:55.000       request: /shuttle/missions/sts-71/sts-71-patch-small.gif auth: - ident: - verb: GET message: alyssa.prodigy.com - - [01/Jul/2017:00:06:55 -0400] "GET /shuttle/missions/sts-71/sts-71-patch-small.gif HTTP/1.0" 200 12054 path: /media/sf_SHARE/NTU/NASA_access_log_Jul2017 @timestamp: July 1st 2017, 12:06:55.000 response: 200 bytes: 12054 clientip: alyssa.prodigy.com @version: 1 host: elk-lab httpversion: 1.0 timestamp: 01/Jul/2017:00:06:55 -0400 _id: AV1-BW2Tv3ttGnl9W3m_ _type: logs _index: firewall _score: -
July 1st 2017, 12:06:54.000       request: /shuttle/missions/sts-71/sts-71-day-04-highlights.html auth: - ident: - verb: GET message: brandt.xensei.com - - [01/Jul/2017:00:06:54 -0400] "GET /shuttle/missions/sts-71/sts-71-day-04-highlights.html HTTP/1.0" 200 5544 path: /media/sf_SHARE/NTU/NASA_access_log_Jul2017 @timestamp: July 1st 2017, 12:06:54.000 response: 200 bytes: 5544 clientip: brandt.xensei.com @version: 1 host: elk-lab httpversion: 1.0 timestamp: 01/Jul/2017:00:06:54 -0400 _id: AV1-BW2Tv3ttGnl9W3m- _type: logs _index: firewall _score: -
July 1st 2017, 12:06:49.000       request: /images/mercury-logo.gif auth: - ident: - verb: GET message: charger.execpc.com - - [01/Jul/2017:00:06:49 -0400] "GET /images/mercury-logo.gif HTTP/1.0" 200 6588 path: /media/sf_SHARE/NTU/NASA_access_log_Jul2017 @timestamp: July 1st 2017, 12:06:49.000 response: 200 bytes: 6588 clientip: charger.execpc.com @version: 1 host: elk-lab httpversion: 1.0 timestamp: 01/Jul/2017:00:06:49 -0400 _id: AV1-BW2Tv3ttGnl9W3m9 _type: logs
```

Kibana

- Download
  - https://www.elastic.co/downloads/kibana
- Docoument
  - https://www.elastic.co/guide/en/kibana/current/index.html
- Execute(ubuntu)
  - $dpkg –i kibana-5.x.x.deb
    - /usr/share/kibana
    - /etc/kibana
      - kibana.yml
  - $./bin/kibana
      - localhost:5601

Kibana Config

- kibana.yml
  - server.host
  - server.port



localhost:5601/app/kibana#/discover?_g=()&_a=(columns:!(_source),index:.kibana,interval:auto,query:(query_string:(a

No results found 😐

Unfortunately I could not find any results matching your search. I tried really harc
just couldn't find anything good. Help me, help you. Here are some ideas:

Expand your time range

I see you are looking at an index with a date field. It is possible your query does n
range, or that there is no data at all in the currently selected time range. Click the
future reference you can open the time picker by clicking on the time picker bu

Refine your query

The search bar at the top uses Elasticsearch's support for Lucene Query String sy
logs that have been parsed into a few fields.

Examples:

Find requests that contain the number 200, in any field:

```
200
```

Or we can search in a specific field. Find 200 in the status field:

```
status:200
```

Find all status codes between 400-499:

```
status:[400 TO 499]
```

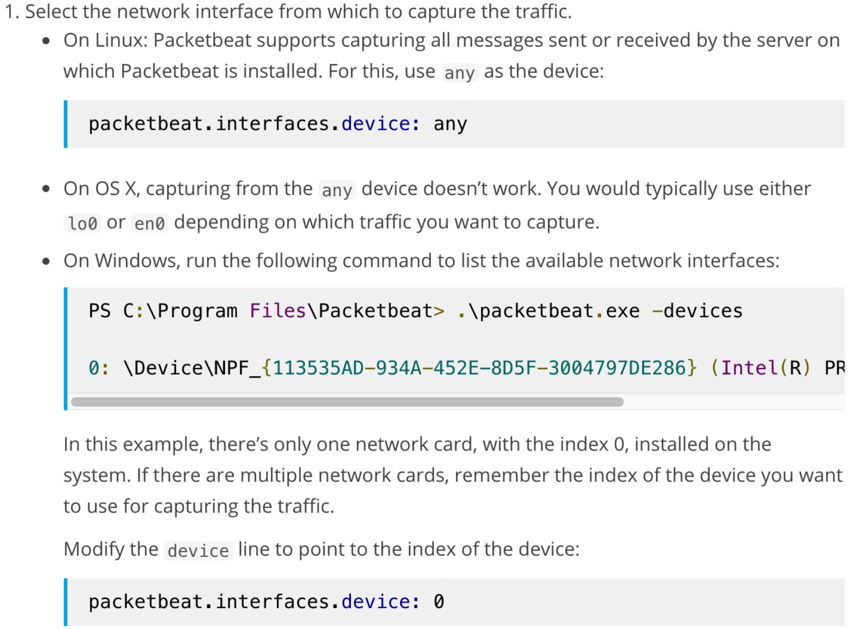
Find status codes 400-499 with the extension php:

```
status:[400 TO 499] AND extension:PHP
```

Or HTML

```
status:[400 TO 499] AND (extension:php OR extension:html)
```

Lab 4



1. Select the network interface from which to capture the traffic.
   - On Linux: Packetbeat supports capturing all messages sent or received by the server on which Packetbeat is installed. For this, use `any` as the device:

     ```
     packetbeat.interfaces.device: any
     ```

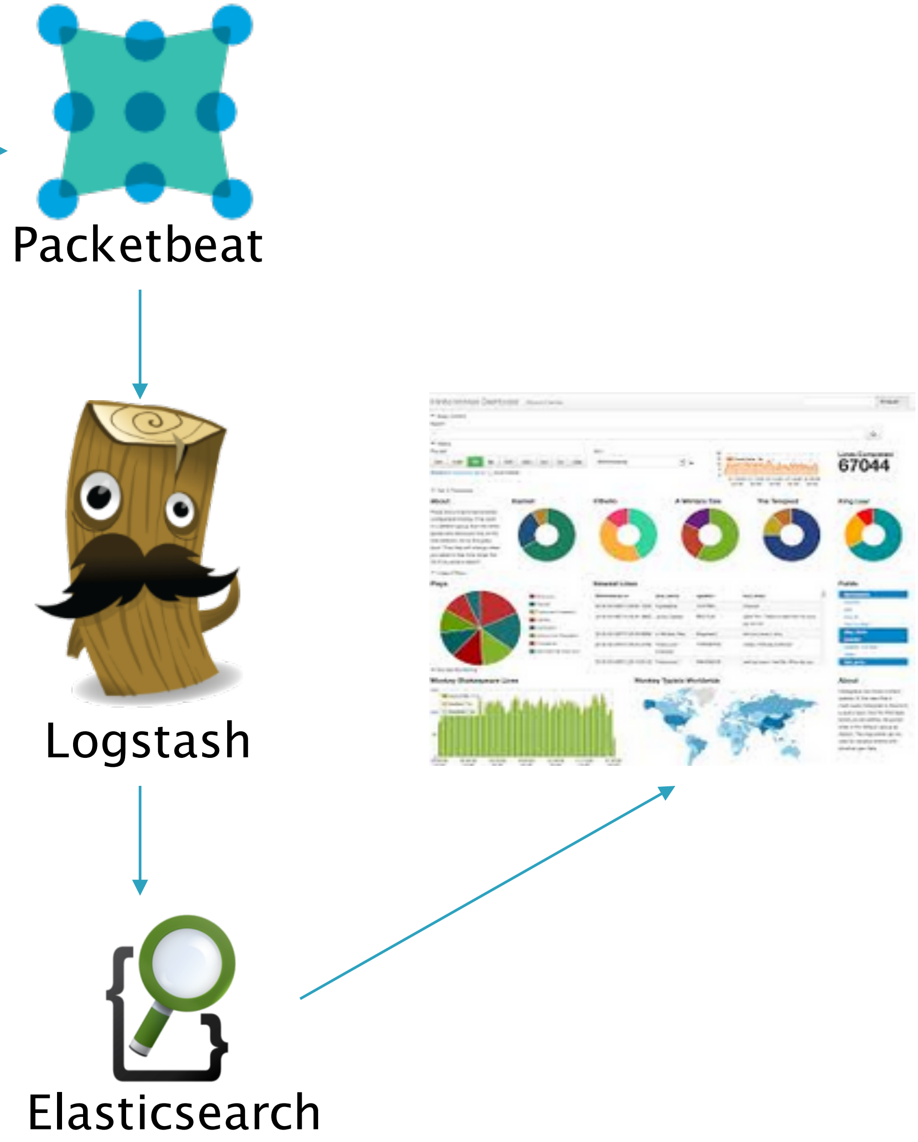
   - On OS X, capturing from the `any` device doesn't work. You would typically use either `lo0` or `en0` depending on which traffic you want to capture.
   - On Windows, run the following command to list the available network interfaces:

     ```
     PS C:\Program Files\Packetbeat> .\packetbeat.exe -devices

     0: \Device\NPF_{113535AD-934A-452E-8D5F-3004797DE286} (Intel(R) PR
     ```

     In this example, there's only one network card, with the index 0, installed on the system. If there are multiple network cards, remember the index of the device you want to use for capturing the traffic.

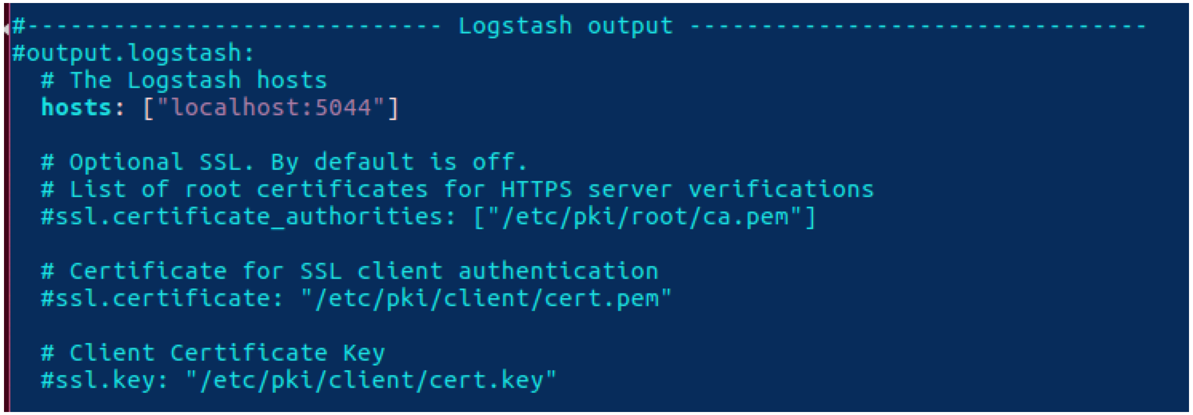
     Modify the `device` line to point to the index of the device:

     ```
     packetbeat.interfaces.device: 0
     ```



Lab 4

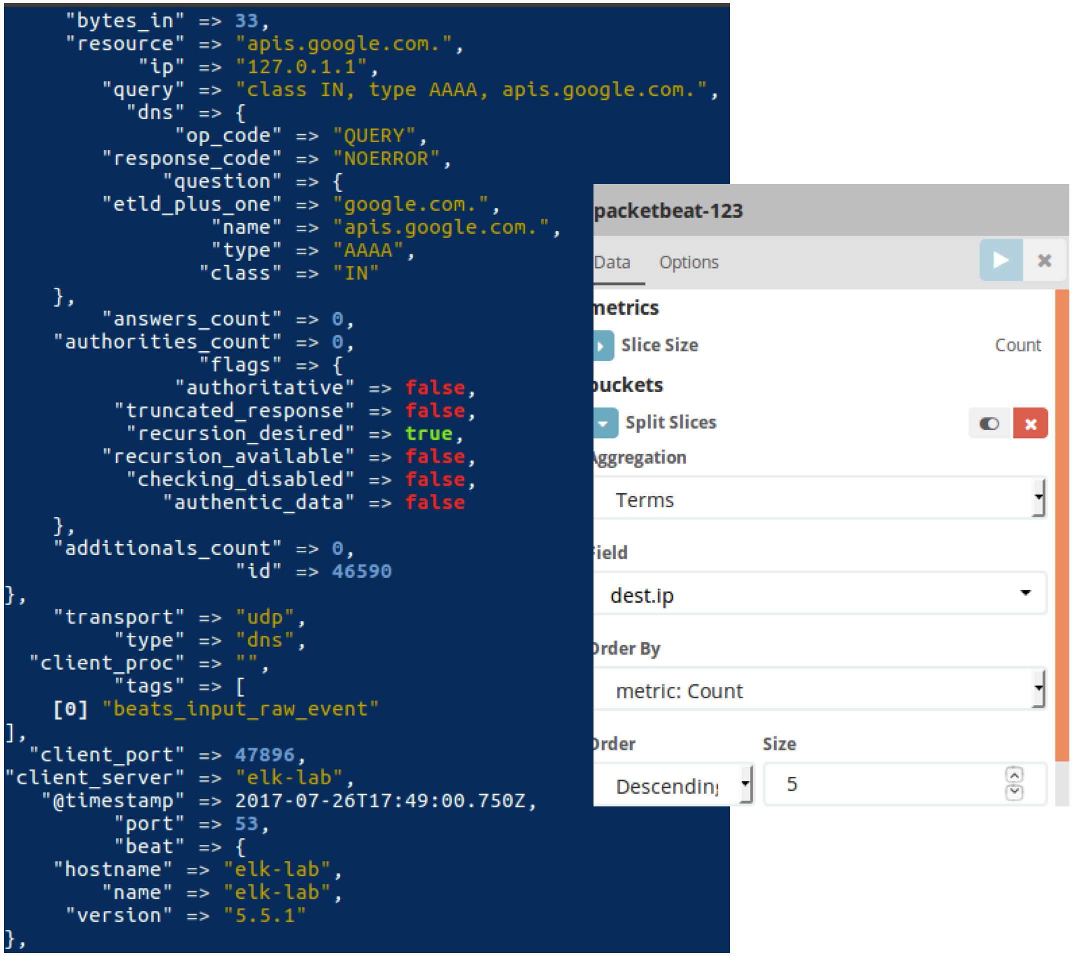
- download packetbeat
- config packetbeat
  - /etc/packetbeat/packetbeat.yml



```
#------------------------------ Logstash output --------------------------------
#output.logstash:
  # The Logstash hosts
  hosts: ["localhost:5044"]

  # Optional SSL. By default is off.
  # List of root certificates for HTTPS server verifications
  #ssl.certificate_authorities: ["/etc/pki/root/ca.pem"]

  # Certificate for SSL client authentication
  #ssl.certificate: "/etc/pki/client/cert.pem"

  # Client Certificate Key
  #ssl.key: "/etc/pki/client/cert.key"
```

- apt-get install libpcap-dev
- service packetbeat start

Lab 4



```
     "bytes_in" => 33,
     "resource" => "apis.google.com.",
           "ip" => "127.0.1.1",
        "query" => "class IN, type AAAA, apis.google.com.",
          "dns" => {
              "op_code" => "QUERY",
        "response_code" => "NOERROR",
             "question" => {
        "etld_plus_one" => "google.com.",
                 "name" => "apis.google.com.",
                 "type" => "AAAA",
                "class" => "IN"
    },
        "answers_count" => 0,
    "authorities_count" => 0,
                "flags" => {
              "authoritative" => false,
         "truncated_response" => false,
          "recursion_desired" => true,
        "recursion_available" => false,
          "checking_disabled" => false,
             "authentic_data" => false
    },
    "additional_count" => 0,
                  "id" => 46590
},
    "transport" => "udp",
         "type" => "dns",
  "client_proc" => "",
         "tags" => [
    [0] "beats_input_raw_event"
],
  "client_port" => 47896,
"client_server" => "elk-lab",
   "@timestamp" => 2017-07-26T17:49:00.750Z,
         "port" => 53,
         "beat" => {
    "hostname" => "elk-lab",
        "name" => "elk-lab",
     "version" => "5.5.1"
},
```

packetbeat-123

Data Options

metrics

Slice Size Count

buckets

Split Slices

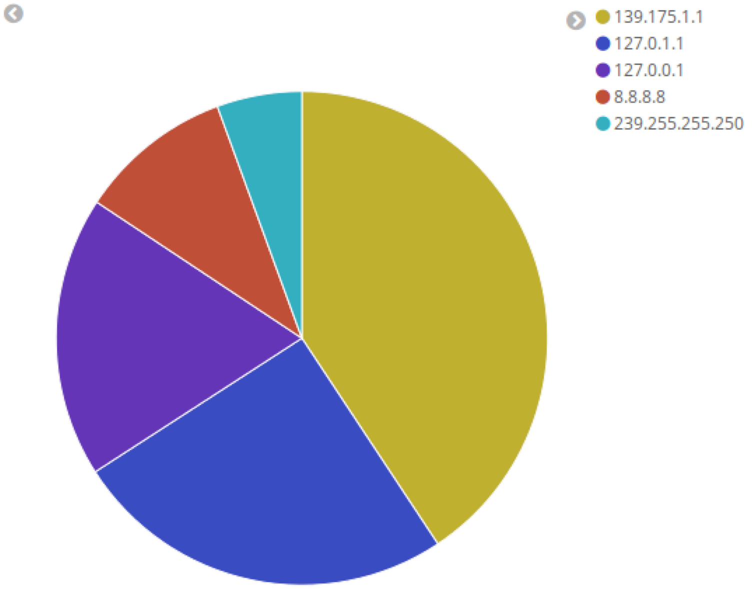
ggregation: Terms

ield: dest.ip

rder By: metric: Count

rder: Descendin;

Size: 5



Thank You

samyshen@iii.org.tw