And the beats go on!

David Pilato
Developer | Evangelist
@dadoonet

Contributors   Traffic   Commits   Code frequency   Punch card   Network   Members

Feb 7, 2010 – Sep 5, 2016

Contributions to master, excluding merge commits

Contributions: **Commits** ▾

kimchy   #1
4,737 commits / 1,401,534 ++ / 1,064,899 --

s1monw   #2
2,216 commits / 1,219,296 ++ / 1,217,559 --

dadoonet   #3
1,171 commits / 128,081 ++ / 84,741 --

javanna   #4
1,101 commits / 142,713 ++ / 119,744 --

martijnvg   #5
1,093 commits / 212,940 ++ / 142,674 --
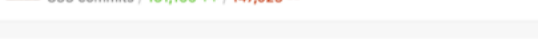
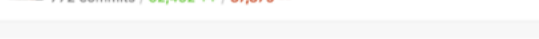### clintongormley   #6
854 commits / 232,442 ++ / 30,976 --

rmuir   #7
833 commits / 161,166 ++ / 147,023 --

jasontedor   #8
772 commits / 62,452 ++ / 37,875 --

jpountz   #9
737 commits / 173,140 ++ / 204,220 --

Elastic Stack

User Interface — Kibana

Store, Index, & Analyze — Elasticsearch

Ingest — Logstash, Beats

Elastic Cloud

X-Pack: Security, Alerting, Monitoring, Reporting, Graph

elastic

The only Elasticsearch as a Service offering powered by the creators of the Elastic Stack

- Always runs on the latest software

- One-click to scale/upgrade with no downtime

- Free Kibana and backups every 30 minutes

- Dedicated, SLA-based support

- Easily add X-Pack features: security (Shield), alerting (Watcher), and monitoring (Marvel)

- Pricing starts at $45 a month
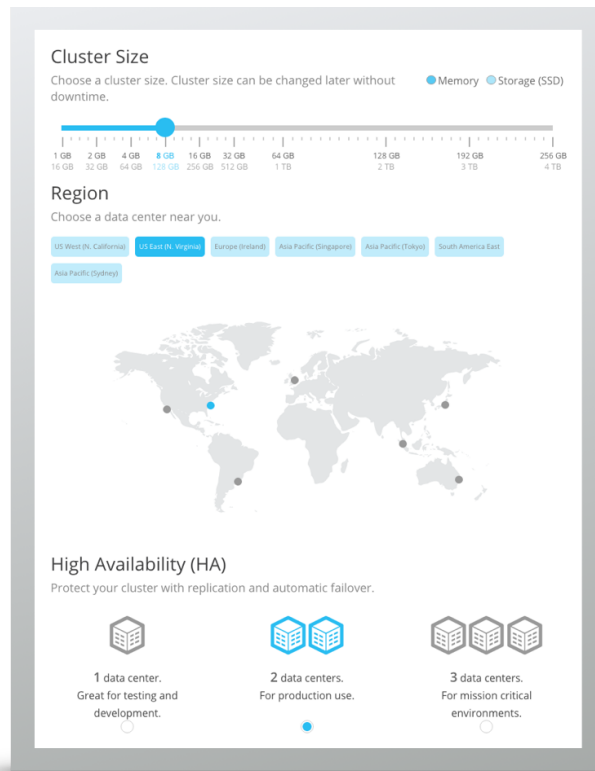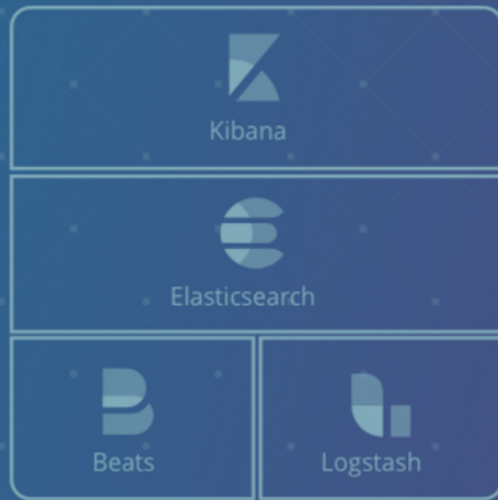
26 OCTOBER 2016   RELEASES   DE   FR   JP   KR   EN

Elastic Stack 5.0.0 Released

By Shay Banon

Share

In February of 2016, following Elastic{ON} 16, I wrote a post titled *Heya, Elastic Stack and X-Pack*. Today, after almost a year of substantial effort, including 5 Alphas, 1 Beta, and 1 Release Candidate we are pleased to announce the GA release of the Elastic Stack.

And, importantly, it is available – today – on Elastic Cloud. If you want hosted Elasticsearch and Kibana there is no other place to start with the most recent code. We are committed to making Elastic Cloud the best place to run

Sign up for product updates!

Email Address

Beats are lightweight shippers that collect and ship all kinds of operational data to Elasticsearch

elastic

Beats are lightweight shippers that collect and ship all kinds of operational data to Elasticsearch

elastic

Beats are lightweight shippers that collect and ship all kinds of operational data to Elasticsearch

Examples of operational data

wire data

Packetbeat

system stats

Metricbeat

logs

Filebeat
Winlogbeat

elastic

Packetbeat

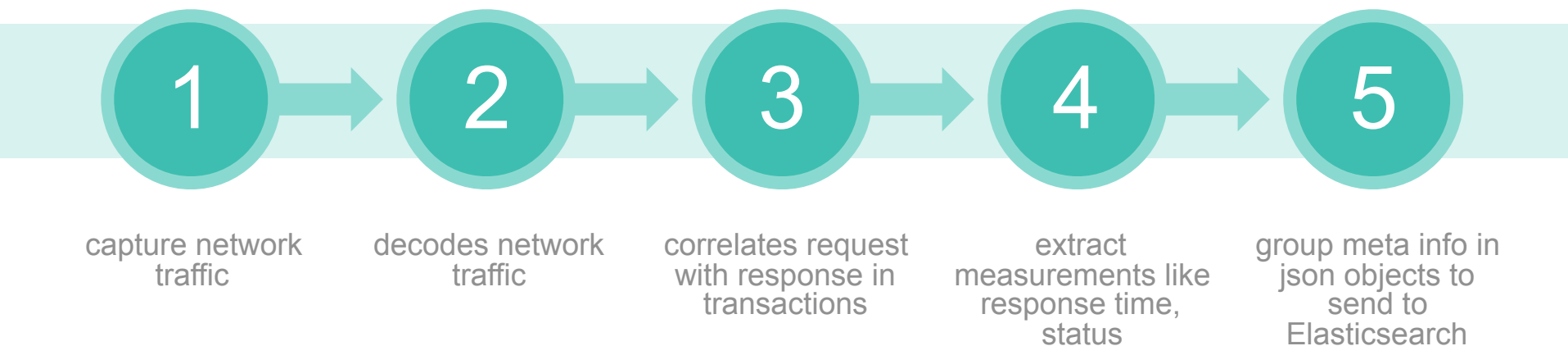Captures insights from network packets

58:31.720580 IP 192.168.0.8.61645 > 52.91.152.165.443: Flags [.], ack 266, win 4092, options [nop
58:31.763428 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], seq 3009:4427, ack 931, win 40
length 1418
58:31.763429 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [P.], seq 4427:4806, ack 931, win 40
length 379
58:31.812093 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [.], ack 4427, win 1644, options [n
58:31.812097 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [.], ack 4806, win 1642, options [n
58:31.968159 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 931:991, ack 4806, win 16
length 60
58:31.968204 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 991, win 4094, options [no
58:31.971541 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [P.], seq 4806:5768, ack 991, win 4
length 962
58:31.971619 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [P.], seq 5768:5987, ack 991, win 4
length 219
58:32.021961 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [.], ack 5768, win 1646, options [n
58:32.021964 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [.], ack 5987, win 1645, options [n
58:32.070031 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 991:1068, ack 5987, win 1
length 77
58:32.070037 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 1068:1246, ack 5987, win
length 178
58:32.070168 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1068, win 4093, options [n
58:32.070268 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1246, win 4090, options [n
58:32.070948 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 1246:1444, ack 5987, win
length 198
58:32.070955 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 1444:1490, ack 5987, win
length 46
58:32.071061 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1444, win 4089, options [n
58:32.071061 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1490, win 4088, options [n
58:32.072967 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [P.], seq 5987:6033, ack 1490, win
length 46
58:32.120485 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [.], ack 6033, win 1653, options [n
58:32.183536 IP 192.168.0.8.61645 > 52.91.152.165.443: Flags [P.], seq 102:203, ack 266, win 4096
gth 101
58:32.457241 IP 52.91.152.165.443 > 192.168.0.8.61645: Flags [.], ack 203, win 122, options [nop,
58:32.457247 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 1490:1540, ack 6033, win
length 50
58:32.457247 IP 64.233.166.189.443 > 192.168.0.8.61563: Flags [P.], seq 1540:1600, ack 6033, win
length 60
58:32.457385 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1540, win 4094, options [n
58:32.457385 IP 192.168.0.8.61563 > 64.233.166.189.443: Flags [.], ack 1600, win 4092, options [n
58:34.349331 IP 192.168.0.8.51759 > 52.22.148.39.443: Flags [P.], seq 1:38, ack 325, win 4096, op
n 37
58:34.518786 IP 52.22.148.39.443 > 192.168.0.8.51759: Flags [.], ack 38, win 136, options [nop,no
58:34.812485 IP 52.91.152.165.443 > 192.168.0.8.61645: Flags [P.], seq 266:415, ack 203, win 122,
gth 149

Sniffing the network traffic

Client

Server

sniff sniff
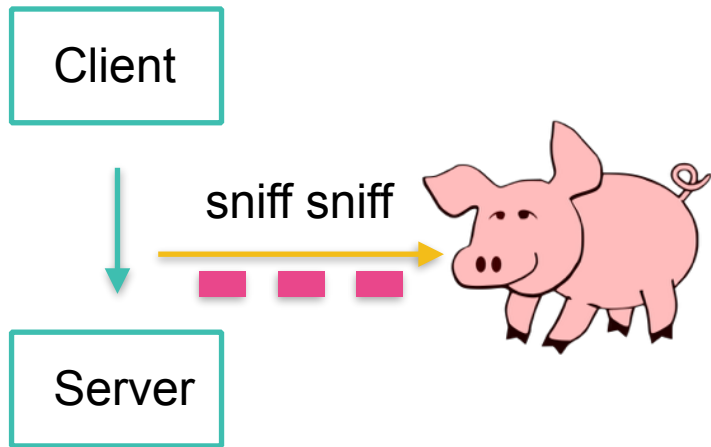
- Copy traffic at OS or hardware level

- Is completely passive

- ZERO latency overhead

- Not in the request/response path, cannot break your application

elastic

Packetbeat: Real-time application monitoring

1 capture network traffic

2 decodes network traffic

3 correlates request with response in transactions

4 extract measurements like response time, status

5 group meta info in json objects to send to Elasticsearch

It does all of these in real-time directly on the target servers.

elastic

Packetbeat: Available decoders

HTTP

MySQL

PostgreSQL

Redis

Thrift-RPC

Memcache

MongoDB

ICMP

DNS

AMQP

+ Add your own

elastic

Metricsbeat

Like the Unix **top** command but sends the output periodically to Elasticsearch. Also works on Windows.

```
Processes: 367 total, 3 running, 6 stuck, 358 sleeping, 1833 threads
Load Avg: 2.79, 2.69, 2.66  CPU usage: 35.77% user, 8.25% sys, 55.96% idle
SharedLibs: 161M resident, 22M data, 10M linkedit. MemRegions: 132196 total, 5764M resident, 90M privat
PhysMem: 13G used (3762M wired), 2581M unused.
VM: 2084G vsize, 527M framework vsize, 56231732(0) swapins, 59291027(0) swapouts.
Networks: packets: 66492613/48G in, 57364574/31G out. Disks: 6744547/369G read, 34720560/883G written.

PID    COMMAND       %CPU  TIME      #TH    #WQ  #PORTS MEM     PURG   CMPRS   PGRP   PPID   STATE    BOOSTS
64667  burn          89.3  04:00.32  5/1    0    15     580K    0B     0B      64667  63666  running  *0[1]
29105  java          21.0  92:05.82  74     0    186    369M    0B     65M     29105  98883  sleeping *0[2]
15112  topbeat       18.7  02:56:57  12     0    61     9900K   0B     1108K   15111  15111  sleeping *0[1]
325    iTerm         12.8  02:18:11  12     4    366    98M     4096B  40M     325    1      sleeping *0[19746
65237  top           7.4   00:08.34  1/1    0    24     7012K   0B     0B      65237  18734  running  *0[1]
0      kernel_task   4.6   23:39:40  220/4  0    2      1719M+  0B     0B      0      0      running  0[0]
61     mds           3.3   01:48:47  10     6    329+   14M     0B     34M     61     1      stuck    *0[1]
194    mds_stores    3.1   02:03:27  10     7    76     20M+    1184K  41M     194    1      stuck    *0[1]
186    WindowServer  1.8   14:39:27  5      2    1224   79M     5324K  267M    186    1      sleeping *0[1]
637-   Dropbox       1.5   93:04.38  84     0    385    86M     0B     44M     637    1      sleeping *0[53781
97     hidd          1.4   93:44.84  6      2    98-    3300K-  0B     1400K   97     1      sleeping *0[1]
53155  mdworker      1.1   01:05.39  4      0    66     14M     0B     1268K   53155  1      sleeping *0[1]
53158  mdworker      1.0   01:06.41  4      0    62     12M     0B     1000K   53158  1      sleeping *0[1]
3431   Slack         0.9   03:36:09  19     1    456    611M-   27M    331M    3431   1      sleeping *0[22918
75966  python2.7     0.7   28:17.99  3      1    33     8240K   0B     13M     75966  75959  stuck    *0[1]
53159  mdworker      0.7   01:04.20  4      0    62     17M     0B     1144K   53159  1      sleeping *0[1]
53157  mdworker      0.6   01:06.09  4      0    62     15M     0B     948K    53157  1      sleeping *0[1]
324-   zoom.us       0.5   13:07:07  13     0    44995  124M    0B     179M    324    1      sleeping *48[18]
23794- dbfseventsd   0.4   01:45.43  1      0    7      4168K   0B     140K    637    23793  sleeping *0[1]
75965  python2.7     0.4   22:28.90  2      0    15     6860K   0B     11M     75965  75959  sleeping *0[1]
58424  Google Chrom  0.3   01:23.17  12     0    111    117M-   0B     37M     316    316    sleeping *0[2]
46     fseventsd     0.3   30:30.91  13     0    306    4796K   0B     4104K   46     1      sleeping *0[1]
23795- dbfseventsd   0.2   01:07.97  1      0    7      32K     0B     152K    637    23794  sleeping *0[1]
48135  Google Chrom  0.2   06:38.73  15     0    60     43M     0B     18M     316    316    sleeping *0[1]
65300  screencaptur  0.2   00:00.14  2      0    52     2220K   20K    0B      336    336    sleeping *0[1]
89     mDNSResponde  0.2   46:50.49  7      2    88     2096K   0B     1124K   89     1      sleeping *0[1]
316    Google Chrom  0.1   20:08:22  48     1    1998   943M    736K   1245M   316    1      sleeping *0[10133
39271  inkscape-bin  0.1   03:05.45  7      0    39     7844K   0B     168M    39268  39268  sleeping *0[1]
60240  Google Chrom  0.1   01:33.59  9      0    110    35M+    0B     34M     316    316    sleeping *0[2]
212    symptomsd     0.1   16:32.42  4      2    102    2184K   0B     2112K   212    1      sleeping 0[66876
23793- dbfseventsd   0.0   00:23.07  1      0    12     36K     0B     180K    637    637    sleeping *0[1]
61533  Google Chrom  0.0   00:07.55  13     0    123    98M     0B     0B      316    316    sleeping *0[1]
16578  Google Chrom  0.0   06:12.31  13     0    155    150M+   0B     50M     316    316    sleeping *0[2]
639-   Google Photo  0.0   15:39.98  14     0    262    4648K   0B     34M     639    1      sleeping *0[27266
13660  Finder        0.0   16:11.28  9      2    423    115M    0B     239M    13660  1      sleeping *0[2627]
58571  com.apple.ap  0.0   00:13.14  4      1    263    20M     512K   19M     58571  1      sleeping *0[297]
54793  Google Chrom  0.0   00:18.73  13     0    121    80M     0B     16M     316    316    sleeping *0[1]
54832  com.apple.ap  0.0   01:02.49  4      1    263    9248K   0B     26M     54832  1      sleeping *0[1856]
62861  Google Chrom  0.0   00:01.85  12     0    112    76M     0B     0B      316    316    sleeping *0[1]
3391   com.apple.ap  0.0   02:43.68  4      1    256    4992K   0B     16M     3391   1      sleeping *0[13618
```

Topbeat: Exported data

| System wide	Per process	Disk usage
• system load • total CPU usage • CPU usage per core • Swap, memory usage	• state • name • command line • pid • CPU usage • memory usage	• available disks • used, free space • mounted points

elastic

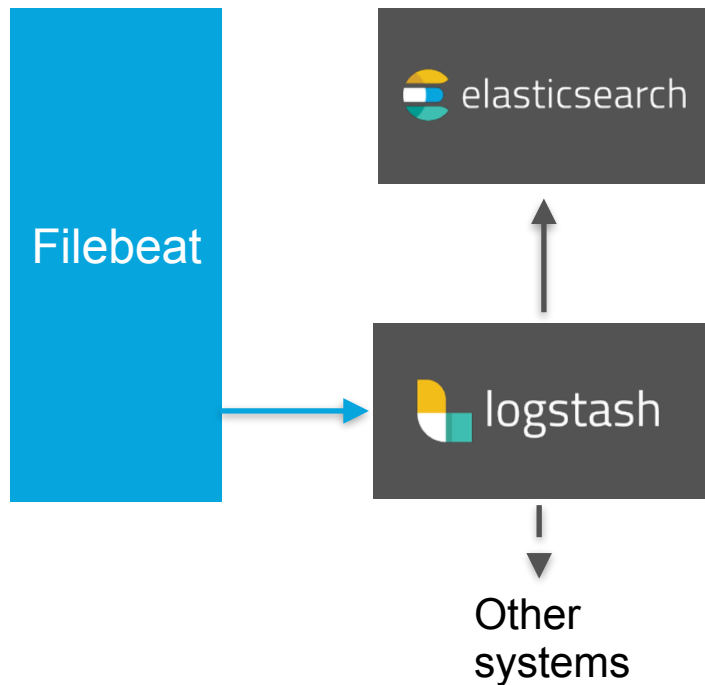# Filebeat

Forwards log lines to Elasticsearch

2016/02/09 21:20:42.414572 client.go:257: WARN Can not index event (status=400):
2 105 110 103 95 101 120 99 101 112 116 105 111 110 34 44 34 114 101 97 115 111
16 104 32 102 97 105 108 117 114 101 115 32 123 91 109 97 112 112 101 114 32 91
2 32 100 105 102 102 101 114 101 110 116 32 116 121 112 101 44 32 99 117 114 114
114 103 101 100 95 116 121 112 101 32 91 100 111 117 98 108 101 101 93 93 125 34 125
2016/02/09 21:20:42.414805 single.go:135: DBG send completed
2016/02/09 21:20:42.414833 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.416692 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.427488 single.go:135: DBG send completed
2016/02/09 21:20:42.427526 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.429343 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.472419 client.go:257: WARN Can not index event (status=400):
2 105 110 103 95 101 120 99 101 112 116 105 111 110 34 44 34 114 101 97 115 111
16 104 32 102 97 105 108 117 114 101 115 32 123 91 109 97 112 112 101 114 32 91
114 101 110 116 32 116 121 112 101 44 32 99 117 114 114 101 110 116 95 116 121 121
121 112 101 32 91 100 111 117 98 108 101 101 93 93 125 34 125]
2016/02/09 21:20:42.472656 single.go:135: DBG send completed
2016/02/09 21:20:42.472679 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.474100 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.482476 single.go:135: DBG send completed
2016/02/09 21:20:42.482513 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.484328 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.499058 single.go:135: DBG send completed
2016/02/09 21:20:42.499152 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.503488 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.520429 single.go:135: DBG send completed
2016/02/09 21:20:42.520605 output.go:87: DBG output worker: publish 50 events
2016/02/09 21:20:42.522417 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.537352 single.go:135: DBG send completed
2016/02/09 21:20:42.885891 output.go:87: DBG output worker: publish 22 events
2016/02/09 21:20:42.886780 bulkapi.go:131: DBG Sending bulk request to http://lo
2016/02/09 21:20:42.894049 single.go:135: DBG send completed
^C2016/02/09 21:20:47.311827 service.go:30: DBG Received sigterm/sigint, stoppi
2016/02/09 21:20:47.311844 beat.go:300: INFO Start exiting beat
2016/02/09 21:20:47.311852 beat.go:275: INFO Stopping Beat
2016/02/09 21:20:47.311862 beat.go:283: INFO Cleaning up topbeat before shutting
2016/02/09 21:20:47.311868 beat.go:139: INFO Exit beat completed

Filebeat: Never lose a log line
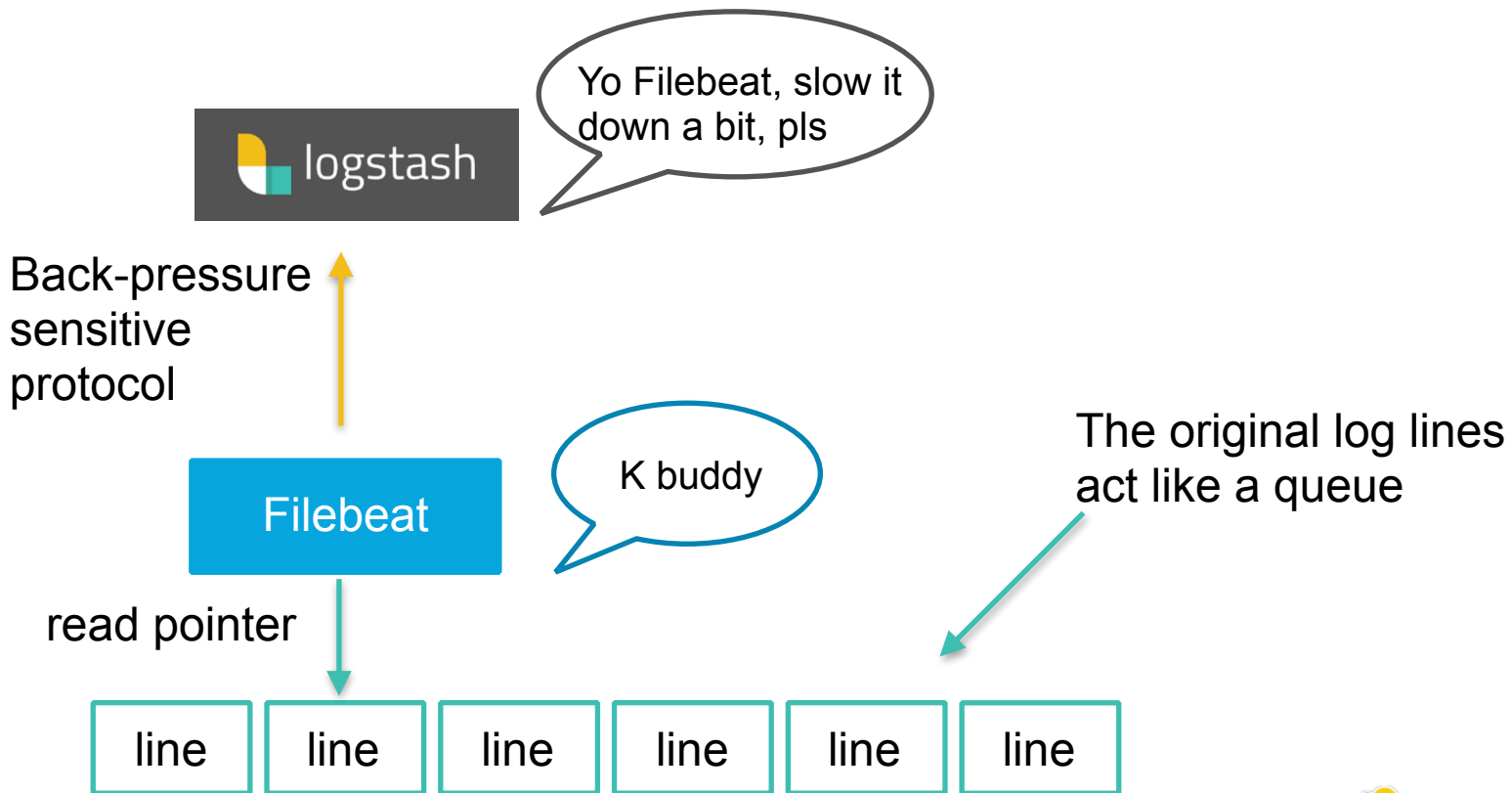
Filebeat: Parse logs with Logstash

Parse logs with Logstash

Filebeat

elasticsearch

logstash

Other
systems

- Filebeat sends out unparsed log lines

- Use filters from Logstash to parse the log lines

- Flexible, with conditionals & custom filters

- Forward data to other systems using the Logstash output plugins

elastic

Filebeat: Parse logs with Ingest Node

Parse logs with Ingest node in Elasticsearch
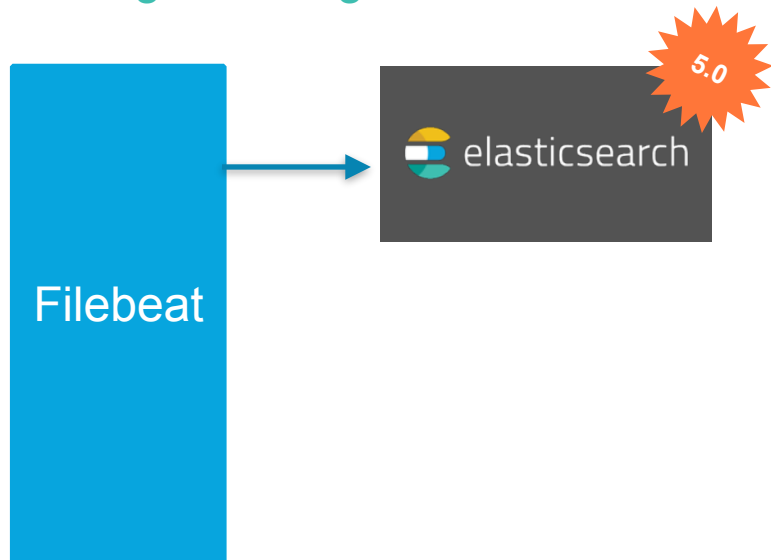
- Filebeat sends out unparsed log lines directly to Elasticsearch

- Use Ingest Node processors (grok, geoip…) to parse the log lines

- Easier to setup

Winlogbeat

Forwards Windows Event logs
to Elasticsearch

Beats Platform

Elastic
Beats

{Community}
Beats

libbeat

Collect, Parse & Ship

beats

Search & Analyze

elasticsearch

Optional

logstash

Enrich & Transport

kibana

Explore & Visualize

22

elastic

Architecture Overview - libbeat

{Community}Beat

libbeat

Config Management

Debugging

Logging

| File | Syslog | * |

Filtering

Cmd Line Handling

Outputs

| Elasticsearch | Logstash | * |

Testing

Testing Environment

System Test Framework

elastic
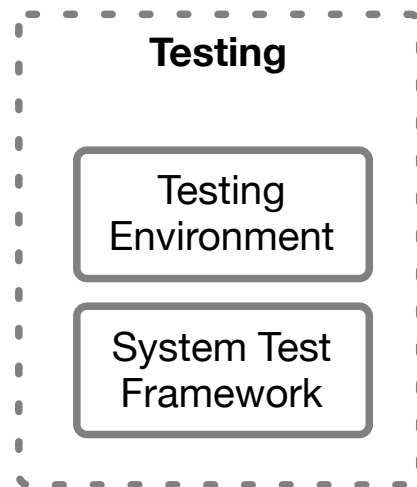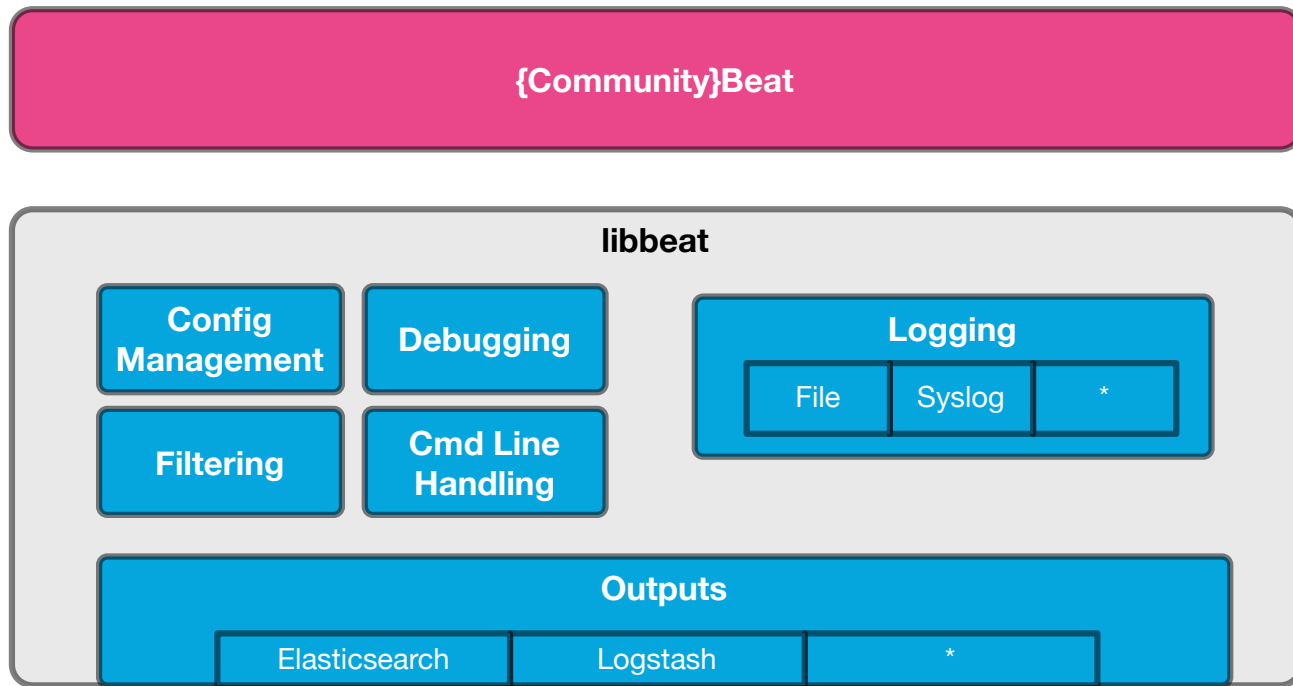
Beats Packer

Produces RPMs, DEBs, …

https://github.com/elastic/beats-packer

metricbeat, packetbeat and soundbeat

https://github.com/dadoonet/soundbeat

{{ softshake'16 }}

elastic

thanks!

https://github.com/dadoonet/soundbeat