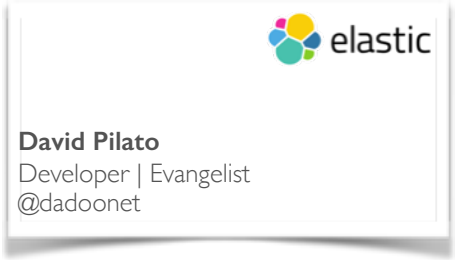
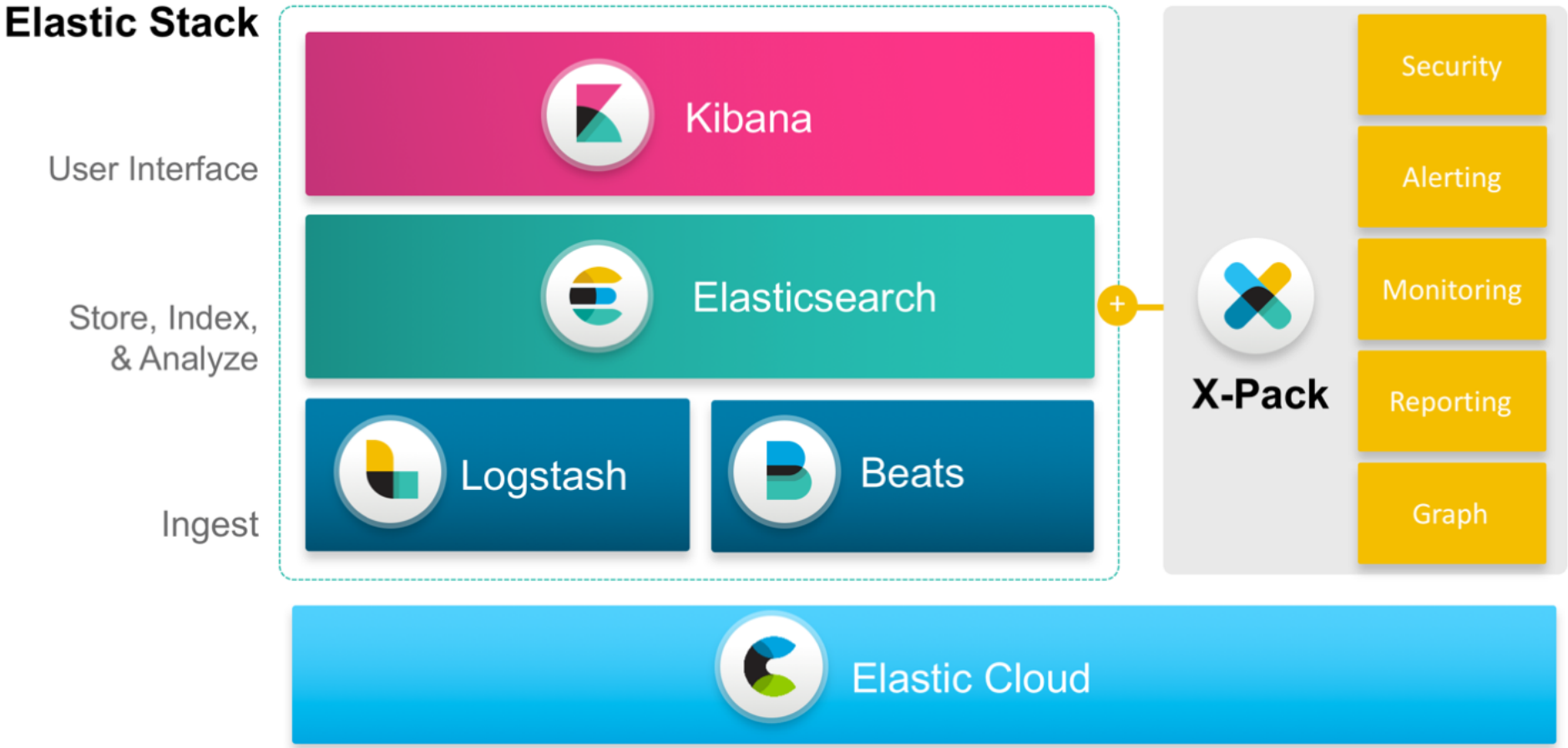
And the beats go on!

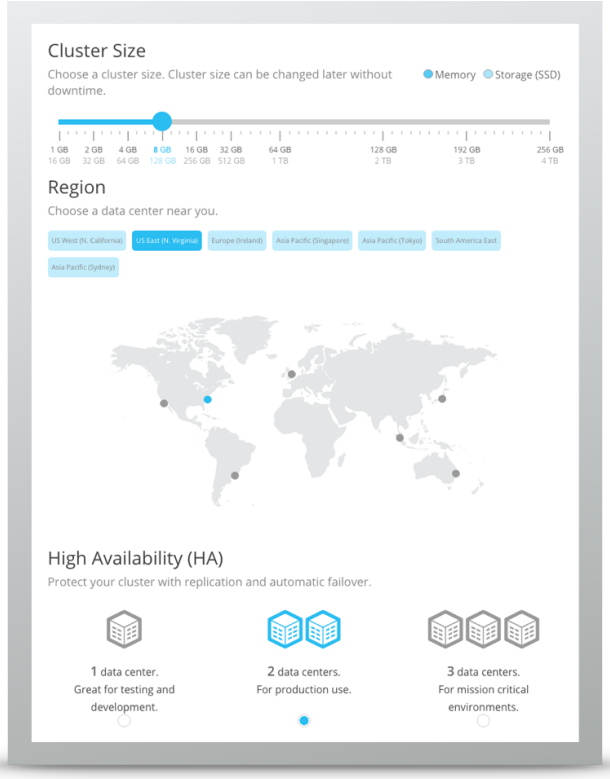



Elastic Stack
User Interface
Store, Index, & Analyze
Ingest
Kibana
Elasticsearch
Logstash
Beats
+
X-Pack
Security
Alerting
Monitoring
Reporting
Graph
Elastic Cloud

cloud

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



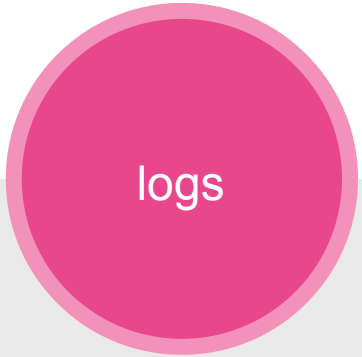
beats

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

Examples of operational data

Packetbeat

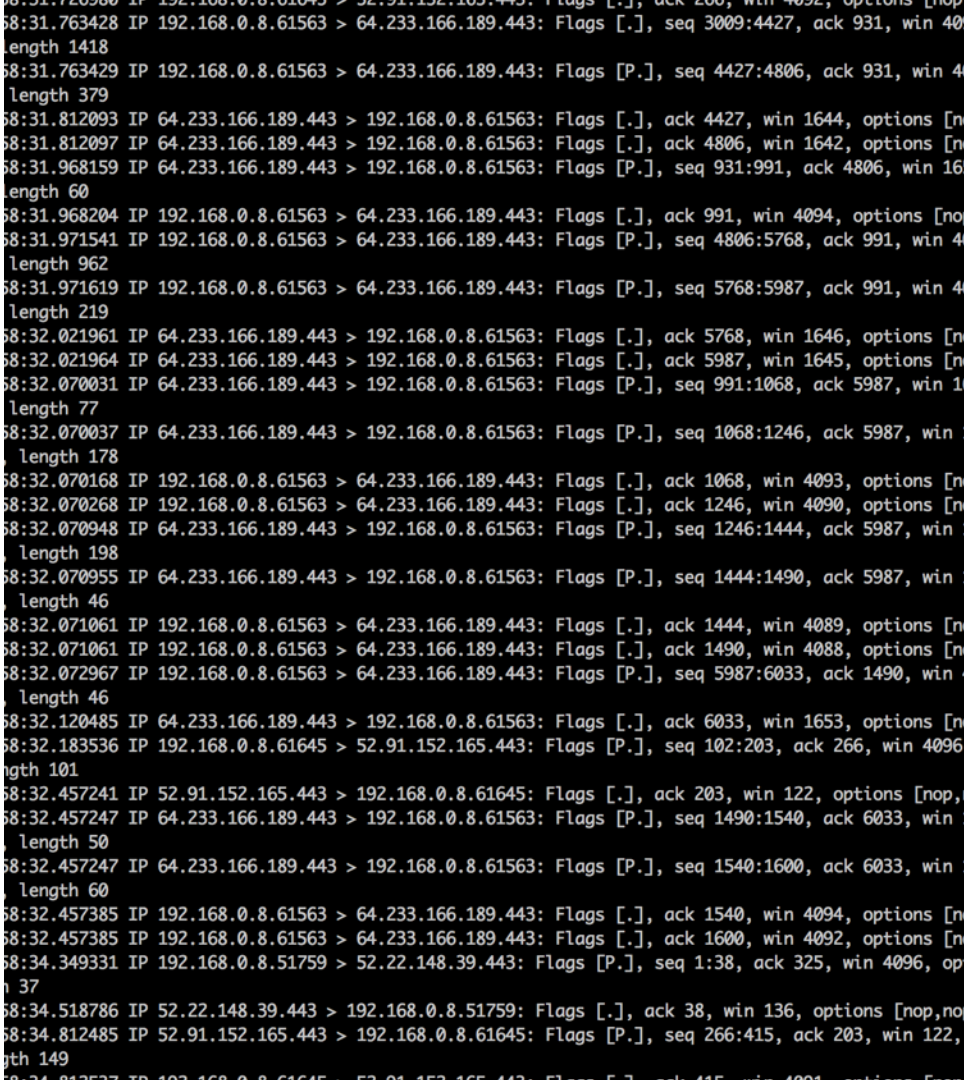
Metricbeat



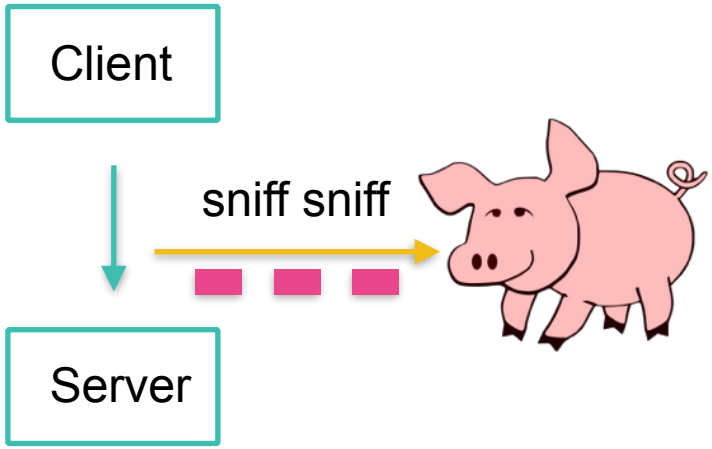
Filebeat
Winlogbeat

Packetbeat

Captures insights from network packets



Sniffing the network traffic



- Copy traffic at OS or hardware level
- Is completely passive
- ZERO latency overhead
- Not in the request/response path, cannot break your application

Packetbeat: Real-time application monitoring

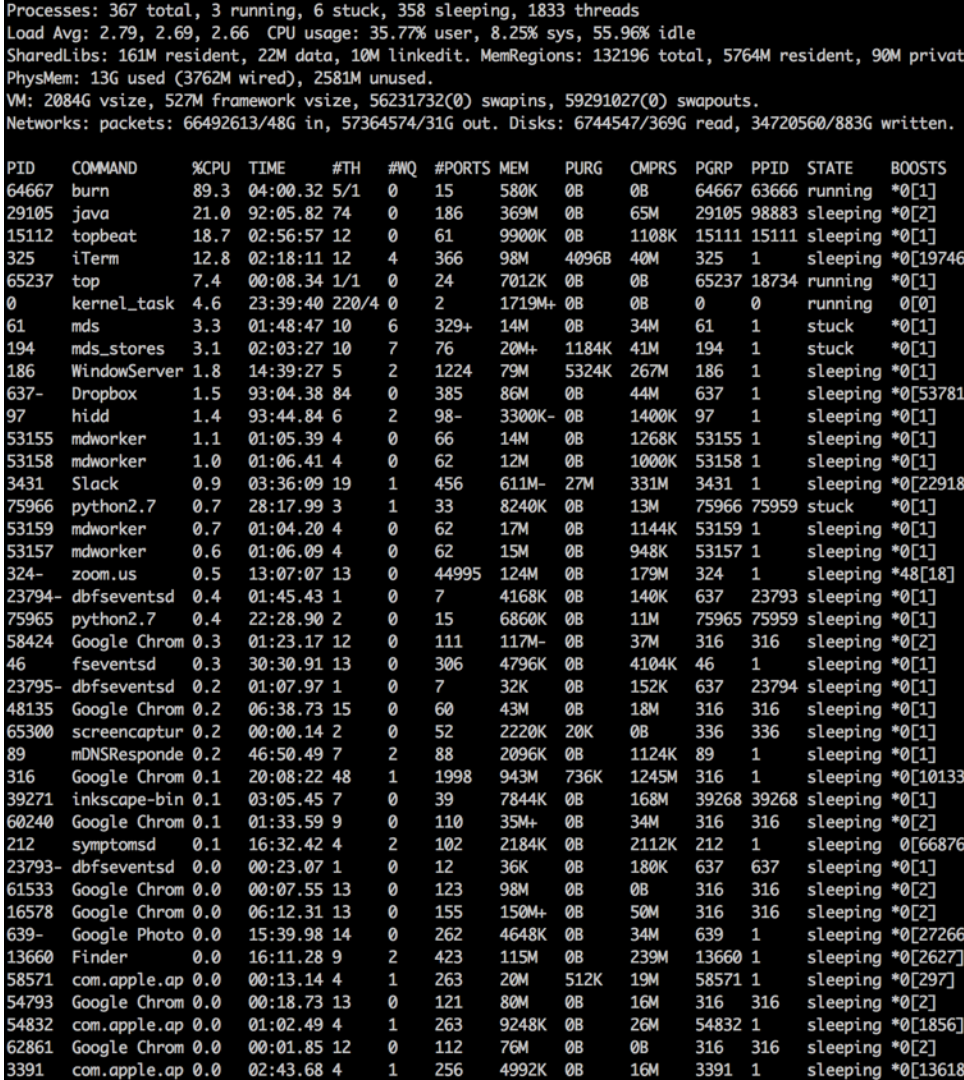
It does all of these in real-time directly on the target servers.

Packetbeat: Available decoders

- HTTP
- MySQL
- PostgreSQL
- Redis
- Thrift-RPC
- Memcache
- MongoDB
- ICMP
- DNS
- AMQP
- + Add your own

Metricsbeat

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



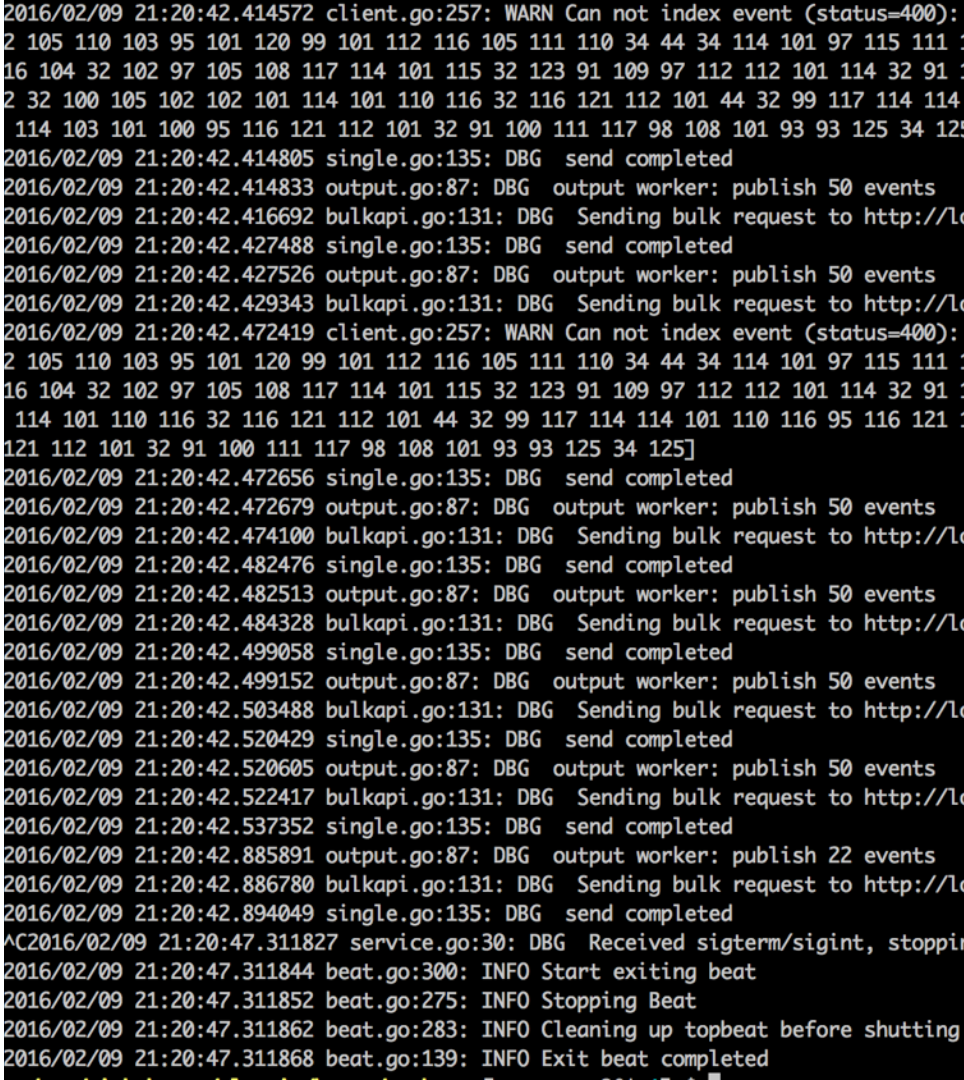
```
Processes: 367 total, 3 running, 6 stuck, 358 sleeping, 1833 threads
Load Avg: 2.79, 2.69, 2.66  CPU usage: 35.77% user, 8.25% sys, 55.96% idle
SharedLibs: 161M resident, 22M data, 10M linkedit. MemRegions: 132196 total, 5764M resident, 90M privat
PhysMem: 13G used (3762M wired), 2581M unused.
VM: 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[2]
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[2]
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[2]
3391   com.apple.ap 0.0   02:43.68 4     1    256    4992K  0B     16M    3391  1     sleeping *0[13618
```

Metricbeat: Exported data

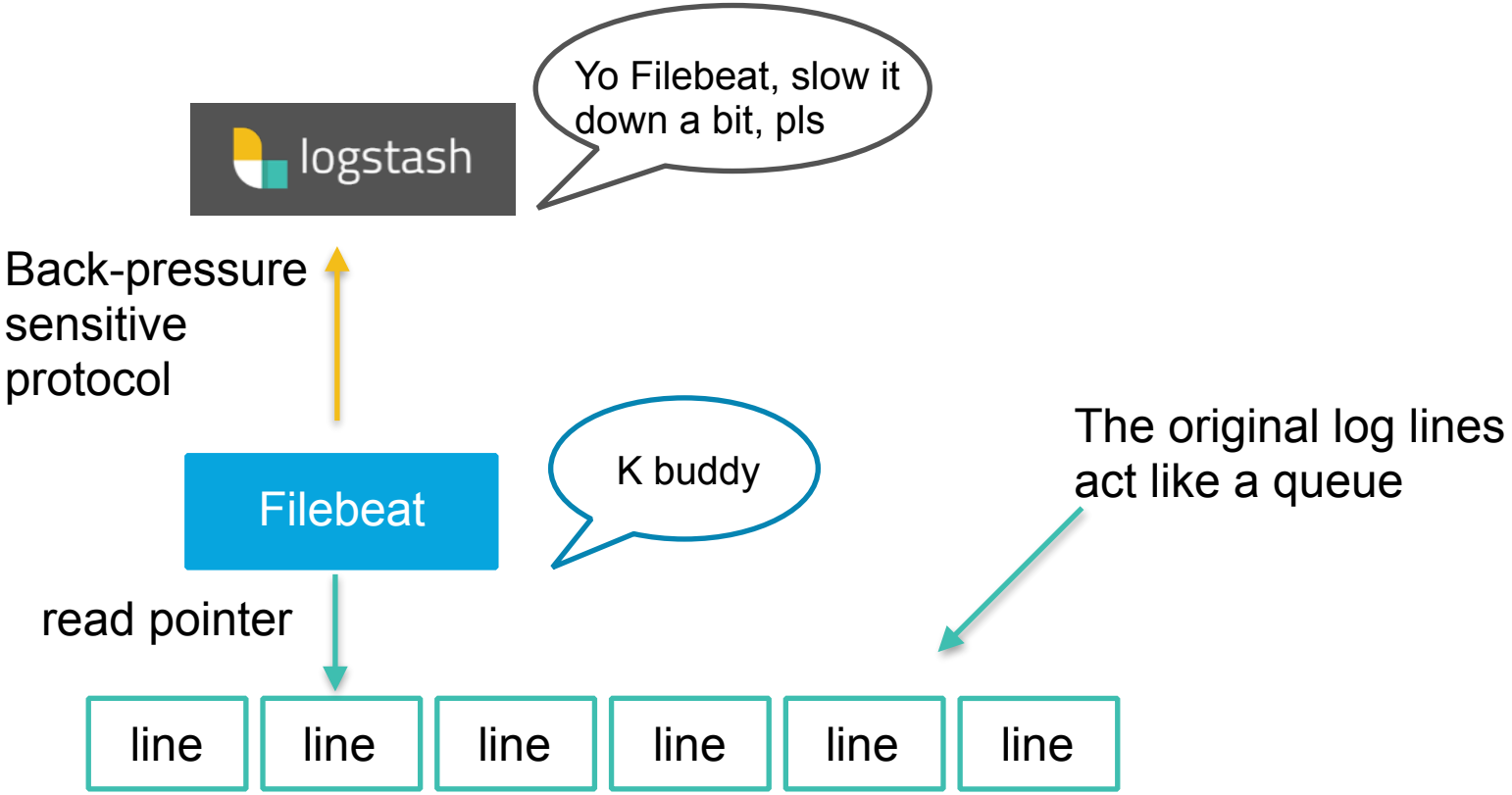
System wide	Per process	Disk usage
• system load • total CPU usage • CPU usage per core • Swap, memory usage	• state • name • command line • pid • CPU usage • memory usage	• available disks • used, free space • mounted points

Filebeat

Forwards log lines to Elasticsearch

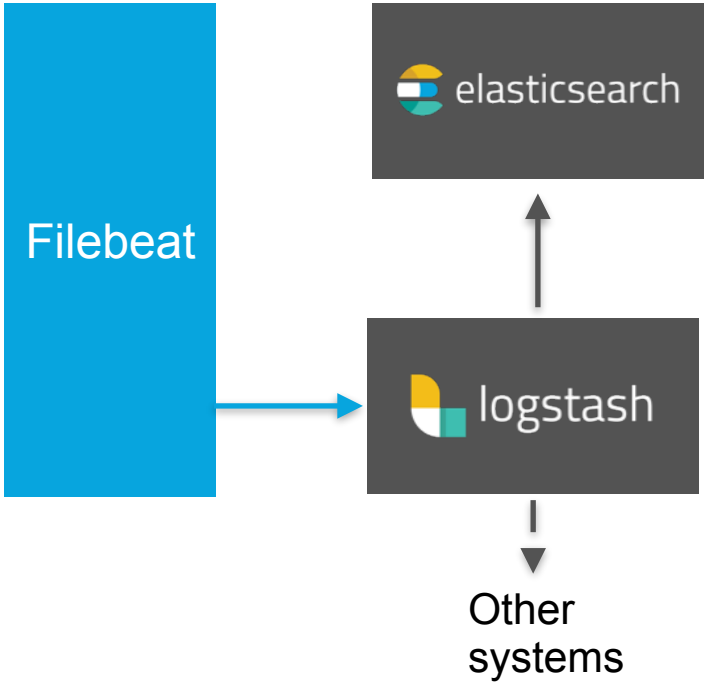


Filebeat: Never lose a log line



Filebeat: Parse logs with Logstash

Parse logs with Logstash



- 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

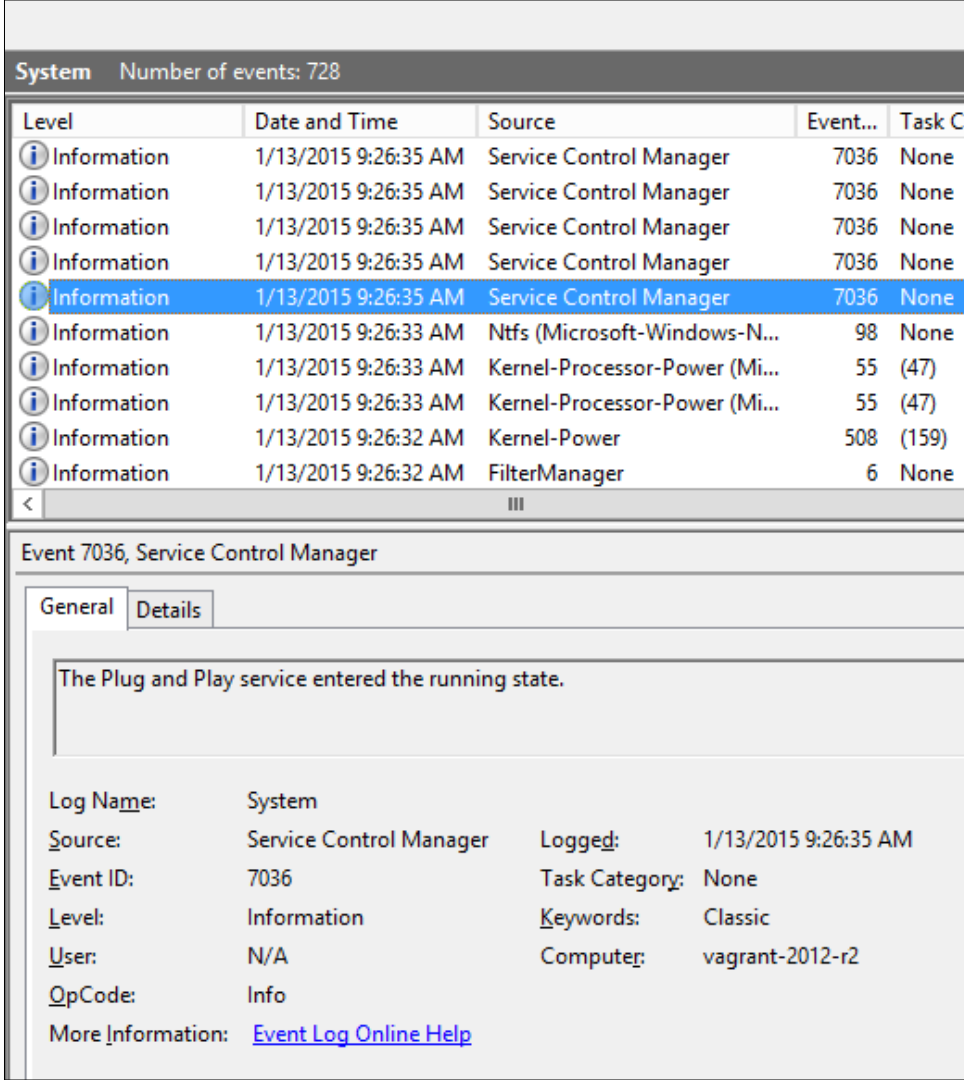
Filebeat: Parse logs with Ingest Node

Parse logs with Ingest node in Elasticsearch

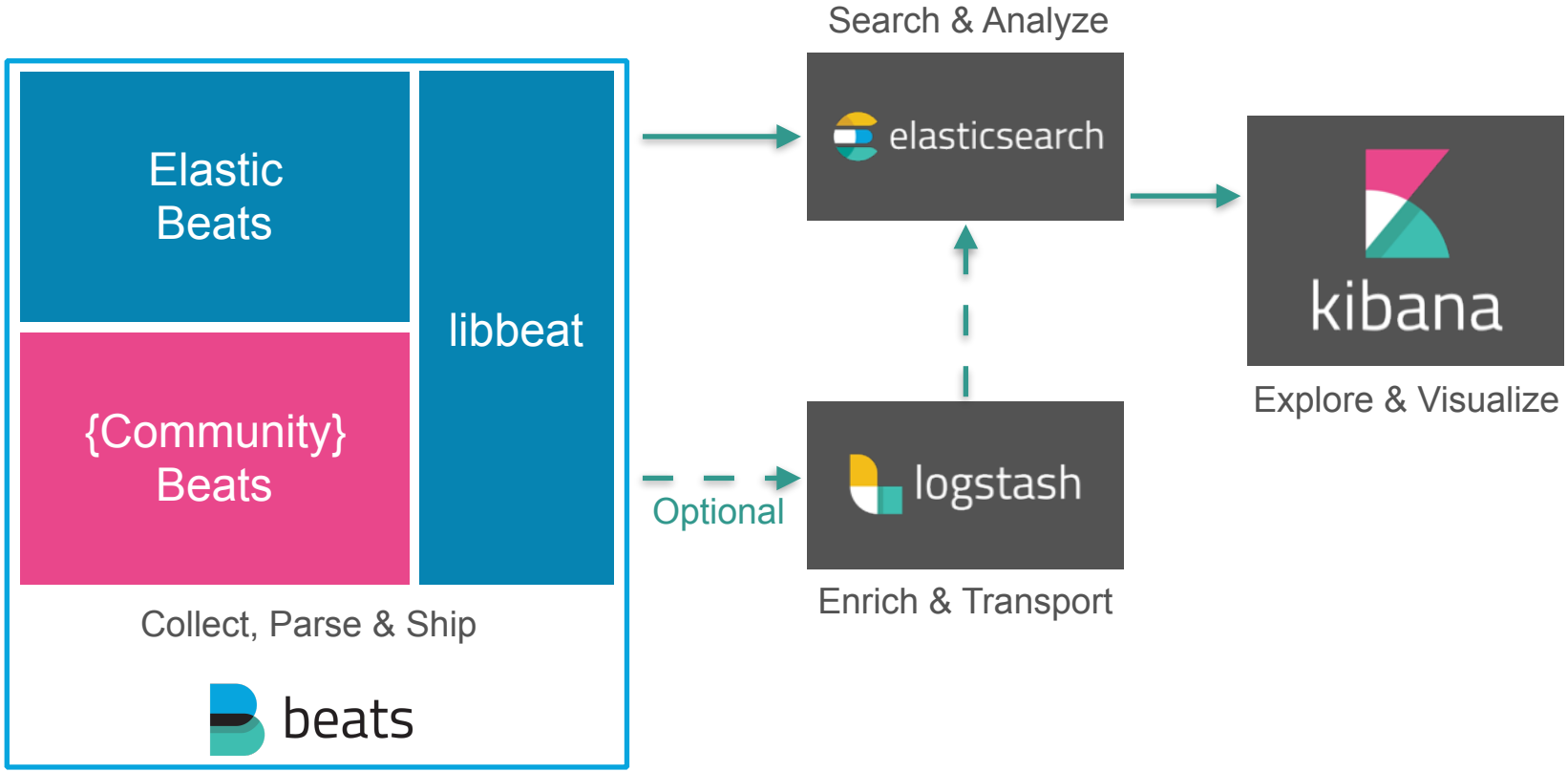
- Filebeat sends out unparsed log lines directly to Elasticsearch
- Use Ingest Node processors (grok, geoip…) to parse the log lines
- Easier to setup

Winlogbeat

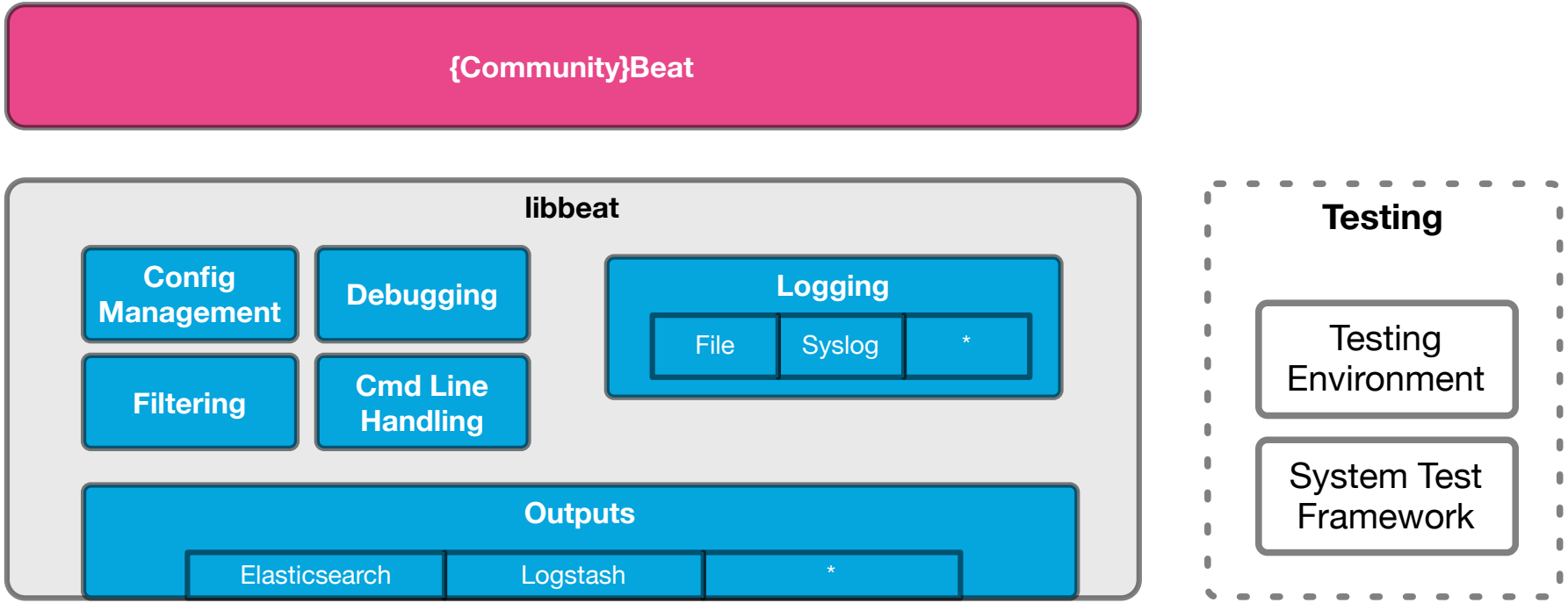
Forwards Windows Event logs to Elasticsearch



Beats Platform



Architecture Overview - libbeat



Beats Packer

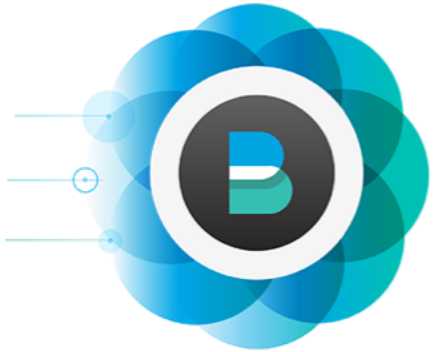
Produces RPMs, DEBs, …

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


Super-Sound-Single
45 rpm
Disco-Remix
THE WHISPERS
RCA
And The Beat Goes On
7:30
Side 1
AF
Can You Do The Boogie
6:07
Side 2
FC 1895
Top Hit
USA

metricbeat, packetbeat and soundbeat

https://github.com/dadoonet/soundbeat



Thanks!

https://github.com/dadoonet/soundbeat