Сбор и анализ логов и метрик распределенного приложения с помощью Elasticsearch/Logstash/Kibana (ELK)

Разработка ПО/CEE-SEC(R) 23.10.2014

Passion to Perform

Содержание

Цели анализа событий распределенной системы

- Способы решения
- Рассмотренные методы решения

Почему именно Elasticsearch/Logstash/Kibana?

- Elasticsearch достаточно популярен
- Как ELK работает у нас: компоненты
- Веб интерфейс Kibana
- Пример JSON записей логов в Elasticsearch
- Фильтры в Kibana и график
- JVM GC в кластере
- Информация о времени операций в кластере
- Статистика
- Метрики, специфичные для приложения

Содержание

- Конфигурация Logstash: секция input
- Конфигурация Logstash: секция filter
- Конфигурация Logstash: секция output
- Удаление индексов старше заданной даты
- Поиск в Elasticsearch регулярными выражениями
- Поиск в Elasticsearch регулярными выражениями
- Как еще можно использовать информацию из ES
- Мониторинг и администрирование ES: Elastic HQ
- Мониторинг и администрирование ES: Elasticsearch head
- Мониторинг и администрирование ES: Marvel

Результаты

- Ресурсы

Q&A

Цели анализа событий распределенной системы

- Поиск причины неисправности в лог журналах системы;
- Поиск причины ухудшения производительности системы;
- Анализ метрик, специфичных для приложения;
- Прогнозирование на основе исторических данных;

Способы решения

— Покупка системы или сервиса анализа логов/метрик;
— Разработка своего «велосипеда»;
— Использование Open Source решений;

Классификация решения по типу хранилища

— Централизованное хранилище;
— Распределенное хранилище;
— Без дополнительного хранилища;

Рассмотренные методы решения

- grep + awk/perl + distributed shell;
- Копирование файлов в распределенную файловую систему и последующая распределенная обработка и поиск;
- Свой «велосипед». Централизованное хранение логов, веб интерфейс
- Apache Chainsaw v2+log4j socket appender
- hawtio + insight-log4j

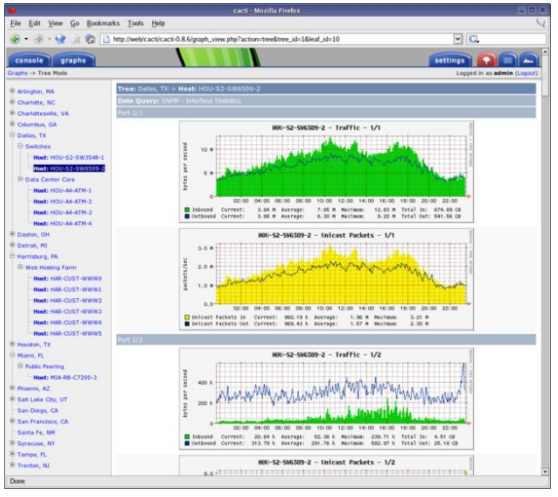
Рассмотренные методы решения

— Splunk

— Syslog/Syslog-NG

— CollectD

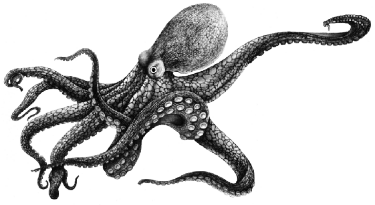
— RRDtool + cacti



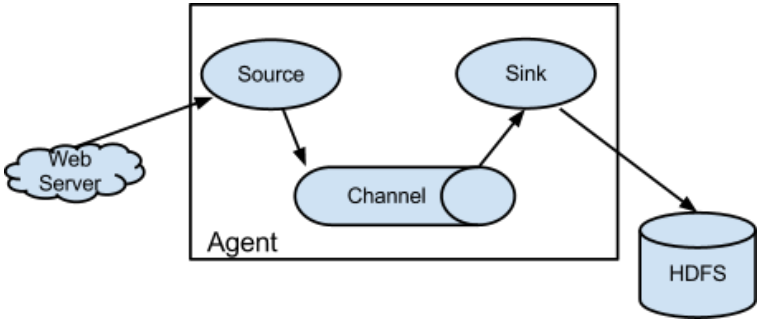
Рассмотренные методы решения

— Enterprise log search and archive (ELSA)

— Octopussy

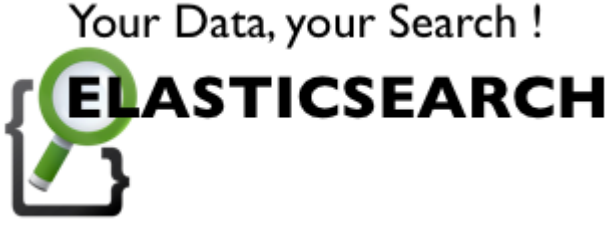


— Apache Flume



— Fluentd

— Elasticsearch/Logstash/Kibana(ELK)



Почему именно Elasticsearch/Logstash/Kibana?

- **Elasticsearch** - распределенная система хранения индексов и данных, выполнение запросов и действий с индексами;
- Функциональные возможности Apache Lucene и стабильность;
- Не надо менять формат логов. Конфигурация **Logstash**;
- HTML5 фронтэнд **Kibana**, визуальное создание фильтров;
- Быстрый старт. ELK интегрированное решение. Можно запустить и попробовать все три компонента системы запуском одного процесса. При том что возможно горизонтальное масштабирование/автоматическое шардирование индексов;
- JVM/Java совместимая технология и для нас это достоинство. Модули для ES на java. Logstash работает в jruby;
- Apache license;

Elasticsearch достаточно популярен

Bloomberg

Bloomberg crunches 1.5B log lines per day for better operational visibility.

theguardian

The Guardian analyzes how 5M users interact with news — all in real-time.

GitHub

Developers search 8M code repositories on GitHub — the world's largest code host.

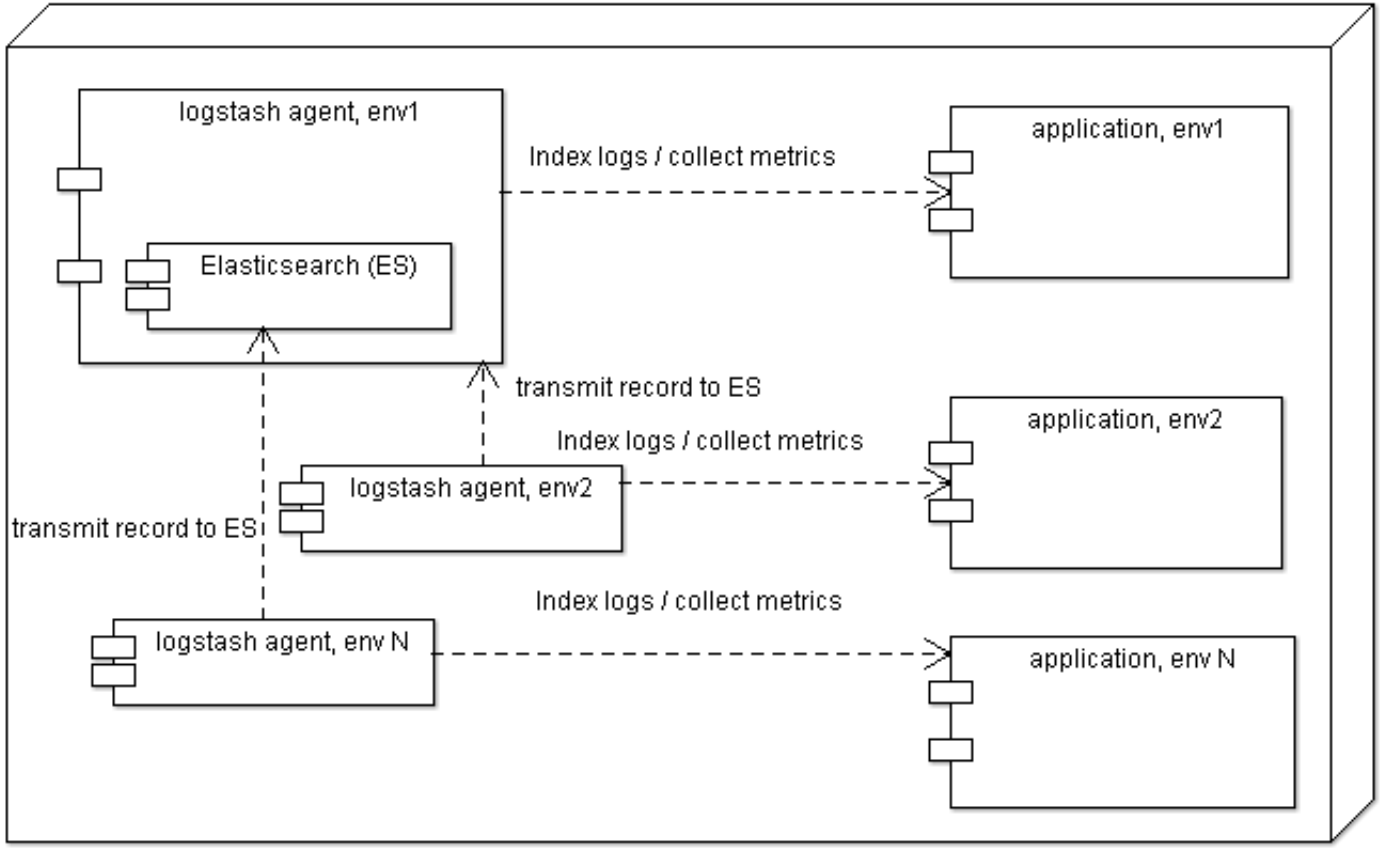
Path · github · foursquare · SONY

SOUNDCLOUD · bazaarvoice · XING · salesforce desk · IGN · mozilla

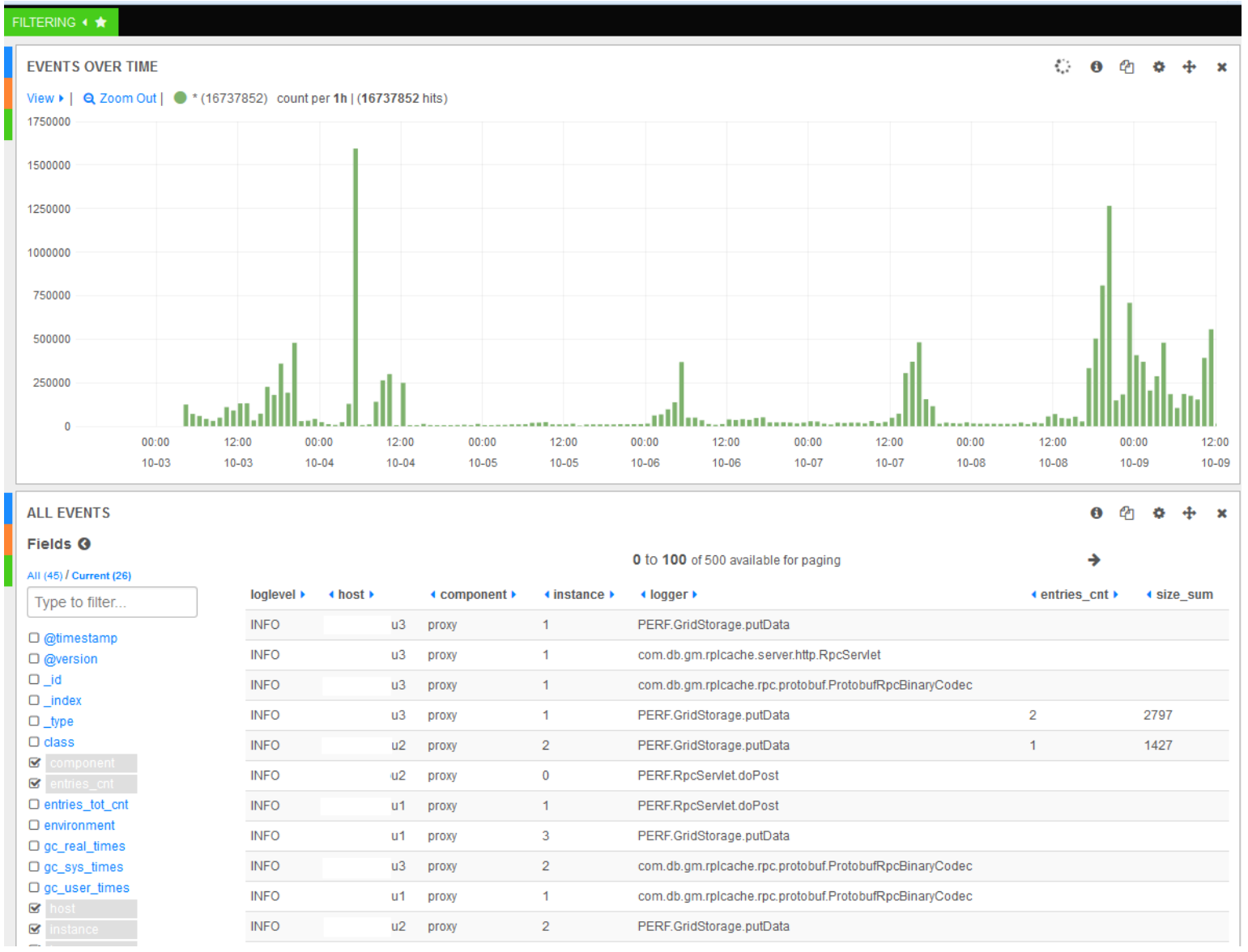
stackoverflow · KLOUT · infochimps · BREWSTER · SCOUT 24

NETWORKED INSIGHTS · Fog Creek SOFTWARE · Revinate · picturesafe

Как ELK работает у нас: компоненты



Веб интерфейс Kibana

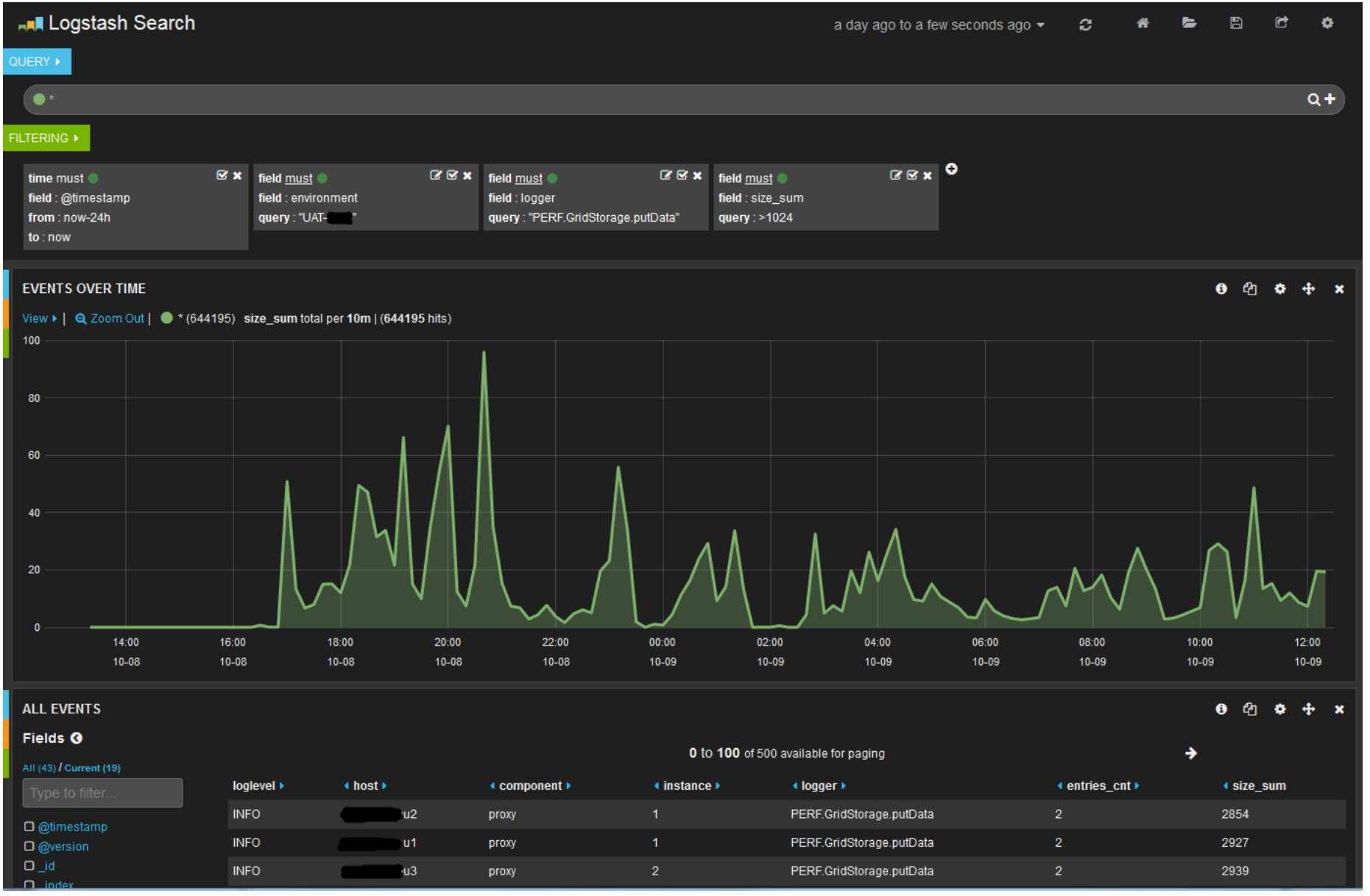


Пример JSON записей логов в Elasticsearch


```
{
    _index: "logstash-2014.10.09",
    _type: "logs",
    _id: "9eZQOAvXSxSf2fgWGQV4bA",
    _score: 1,
    ▼ _source: {
        message: "Request completed, tracer=6740236904141455 EXEC_TIME: 2.19ms",
        @version: "1",
        @timestamp: "2014-10-09T00:01:18.433Z",
        environment: "UAT-[redacted]",
        host: "[redacted]2",
        class: "qtp920138597-13740",
        loglevel: "INFO",
        component: "proxy",
        instance: 3,
        logger: "PERF.RpcServlet.doPost",
        times: 2.19
    }
},
{
    _index: "logstash-2014.10.09",
    _type: "logs",
    _id: "Na8vsq1nTEeY93XmPEJSRg",
    _score: 1,
    ▼ _source: {
        message: "2014-10-09T01:01:07.084+0100: [GC [ParNew: 943744K->73933K(943744K), 0.1091770 secs]
        @version: "1",     2100836K->1308588K(8283776K), 0.1092600 secs] [Times: user=0.27 sys=0.09, real=0.11 secs] ",
        @timestamp: "2014-10-09T00:01:07.000Z",
        wrapper: "true",
        environment: "DEV-[redacted]",
        host: "[redacted]3",
        loglevel: "INFO",
        source: "jvm 1 ",
        component: "server",
        instance: 1,
        gc_user_times: 0.27,
        gc_sys_times: 0.09,
        gc_real_times: 0.11
    }
},
```

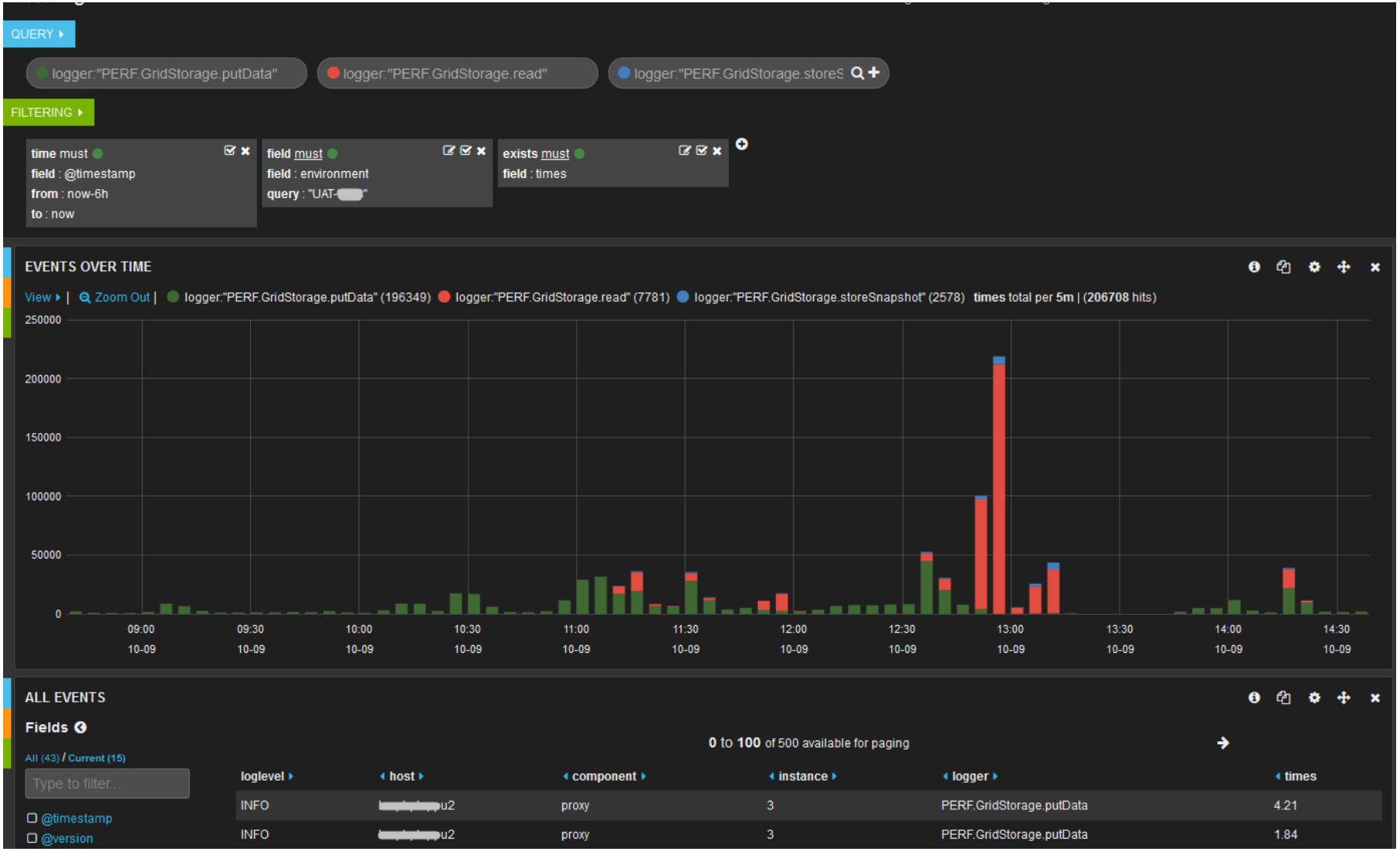

@timestamp is the ISO8601 high-precision timestamp for the event.
@version is the version number of this json schema

Фильтры в Kibana и график

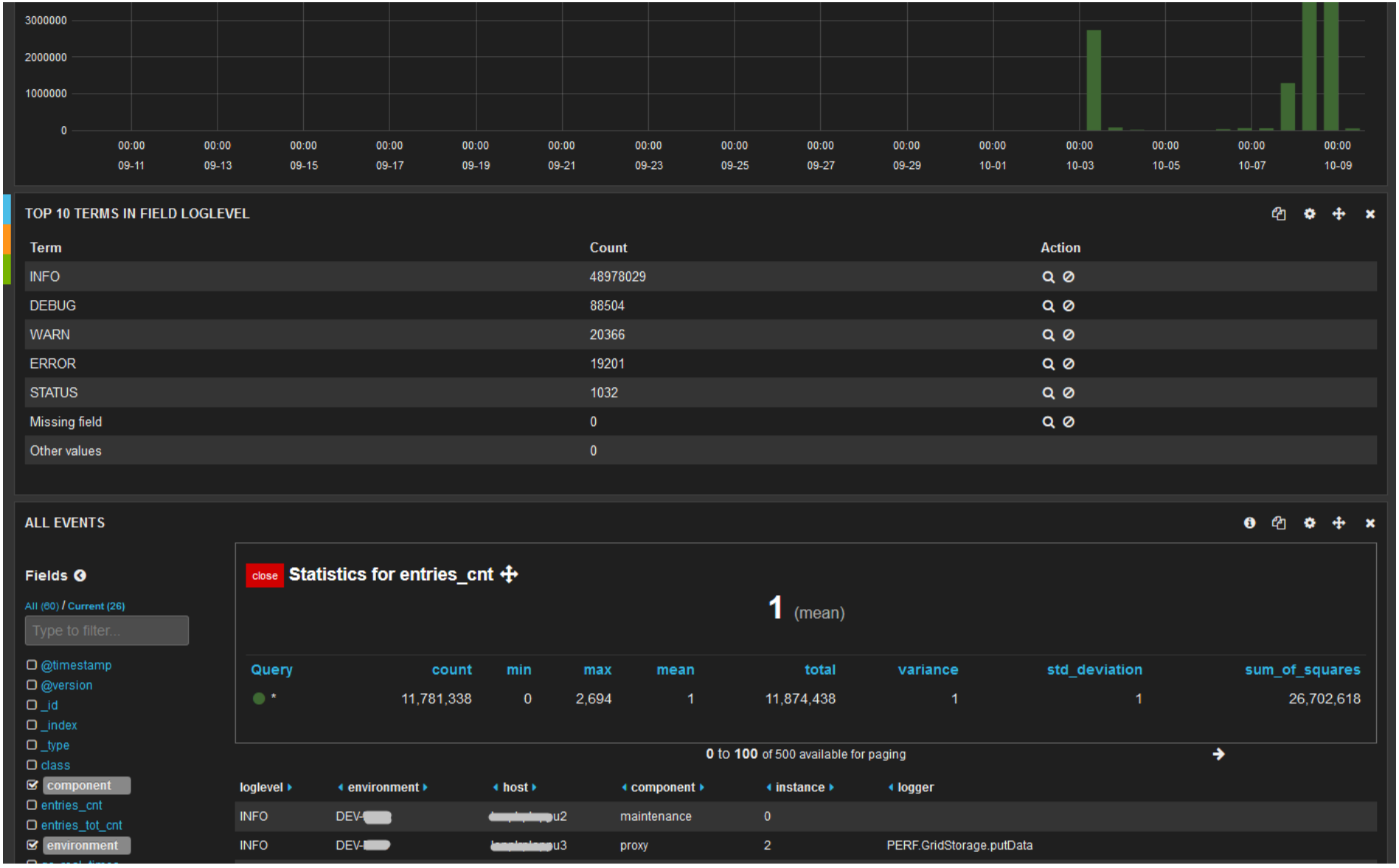


JVM GC в кластере

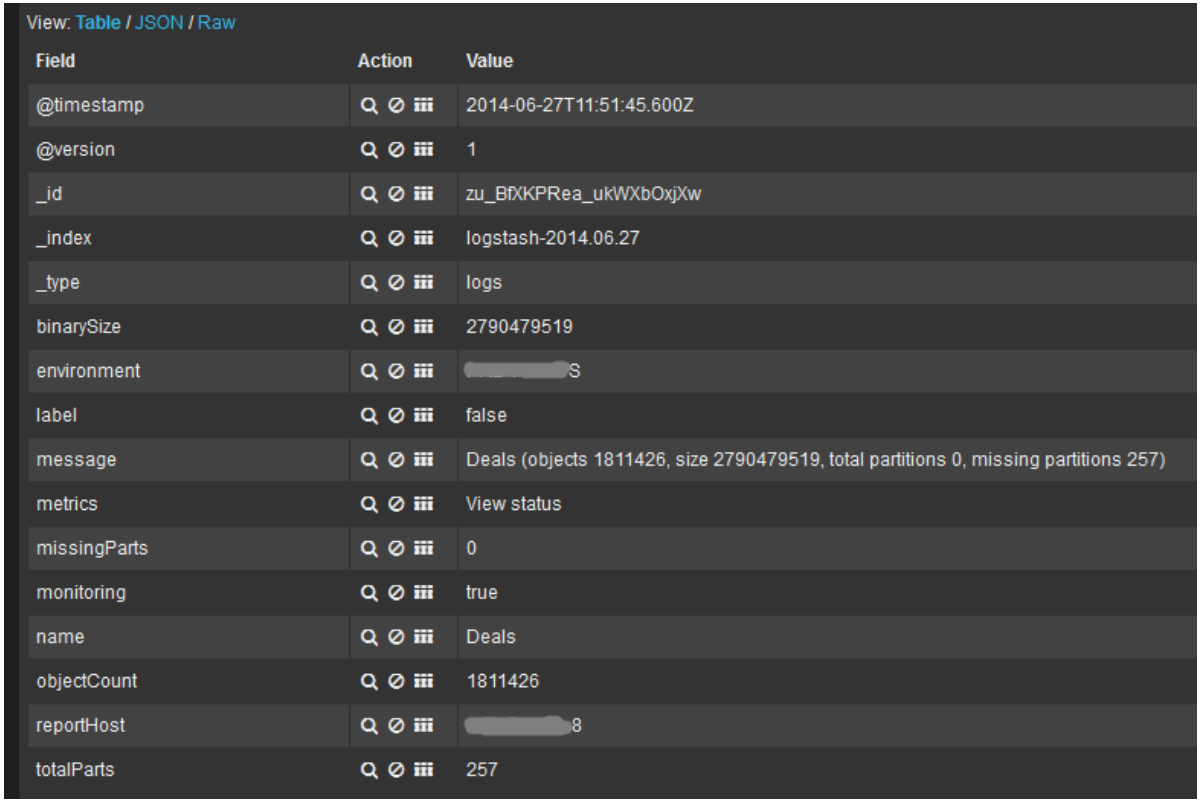
Информация о времени операций в кластере



Статистика

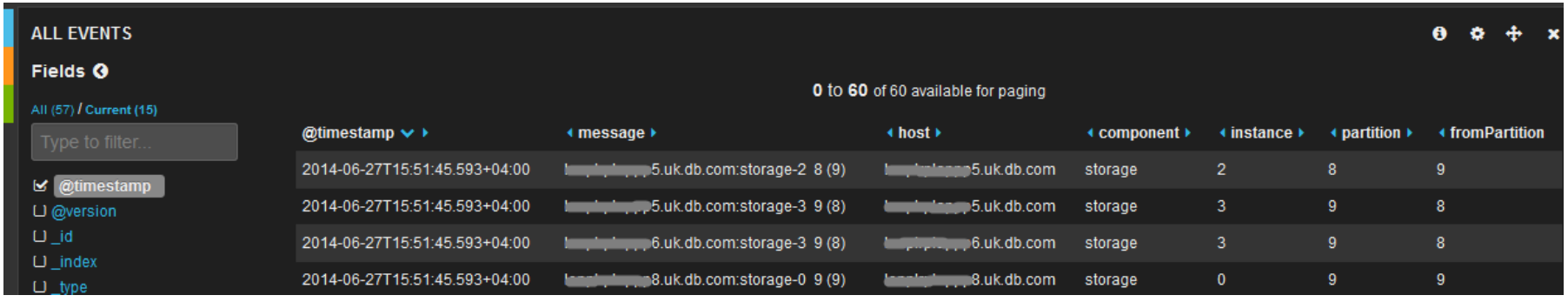

3000000
2000000
1000000
0
00:00 09-11
00:00 09-13
00:00 09-15
00:00 09-17
00:00 09-19
00:00 09-21
00:00 09-23
00:00 09-25
00:00 09-27
00:00 09-29
00:00 10-01
00:00 10-03
00:00 10-05
00:00 10-07
00:00 10-09
TOP 10 TERMS IN FIELD LOGLEVEL
Term
Count
Action
INFO 48978029
DEBUG 88504
WARN 20366
ERROR 19201
STATUS 1032
Missing field 0
Other values 0
ALL EVENTS
Fields
All (60) / Current (26)
Type to filter...
@timestamp
@version
_id
_index
_type
class
component
entries_cnt
entries_tot_cnt
environment
close
Statistics for entries_cnt
1 (mean)
Query count min max mean total variance std_deviation sum_of_squares
* 11,781,338 0 2,694 1 11,874,438 1 1 26,702,618
0 to 100 of 500 available for paging
loglevel environment host component instance logger
INFO DEV- u2 maintenance 0
INFO DEV- u3 proxy 2 PERF.GridStorage.putData

Метрики, специфичные для приложения



View: Table / JSON / Raw

Field	Action	Value
@timestamp		2014-06-27T11:51:45.600Z
@version		1
_id		zu_BfXKPRea_ukWXbOxjXw
_index		logstash-2014.06.27
_type		logs
binarySize		2790479519
environment		[redacted]S
label		false
message		Deals (objects 1811426, size 2790479519, total partitions 0, missing partitions 257)
metrics		View status
missingParts		0
monitoring		true
name		Deals
objectCount		1811426
reportHost		[redacted]8
totalParts		257



ALL EVENTS

Fields

All (57) / Current (15)

Type to filter...

- @timestamp
- @version
- _id
- _index
- _type

0 to 60 of 60 available for paging

@timestamp	message	host	component	instance	partition	fromPartition
2014-06-27T15:51:45.593+04:00	[redacted]5.uk.db.com:storage-2 8 (9)	[redacted]5.uk.db.com	storage	2	8	9
2014-06-27T15:51:45.593+04:00	[redacted]5.uk.db.com:storage-3 9 (8)	[redacted]5.uk.db.com	storage	3	9	8
2014-06-27T15:51:45.593+04:00	[redacted]6.uk.db.com:storage-3 9 (8)	[redacted]6.uk.db.com	storage	3	9	8
2014-06-27T15:51:45.593+04:00	[redacted]8.uk.db.com:storage-0 9 (9)	[redacted]8.uk.db.com	storage	0	9	9

Конфигурация Logstash: секция **input**

```
input {
        file {
                start_position => beginning
                path => [ "/local/...../cache/logs/*-wrapper.log.*" ]
                sincedb_path => "/...../cache/df-cluster/.sincedb_path_UAT-1"
                exclude => ["logserver-*"]
                add_field => {
                        "wrapper" => "true"
                        "environment" => "UAT-1"
                }
        }
        file {
                start_position => beginning
                path => [ "/local/applogs/cache/df-cluster/UAT-1/cache/logs/*.log.*" ]
                sincedb_path => "/local/applogs/cache/df-cluster/.sincedb_path_UAT-1"
                exclude => ["*-wrapper*", "logserver-*"]
                add_field => {
                        "environment" => "UAT-1"
                }
        }
   .....
}
```

Конфигурация Logstash: секция **filter**

```
filter{
    if[monitoring] != "true"{
        multiline {
            patterns_dir => "./patterns“  negate => true                    what => previous
            pattern => "^%{LOG4J_DATESTAMP:timestamp} "
     }
        grok{
            patterns_dir => "./patterns"
            match => [ "message", "%{LOG4J_DATESTAMP:logtimestamp} -
    \[%{GREEDYDATA:class}\] - %{LOGLEVEL:loglevel} –
          %{GREEDYDATA:logmessage}"]
            named_captures_only => true
        }
        grok {
            match => [ "path", "%{UNIXPATH}/%{WORD:component}-
    %{INSTANCEID:instance:int}" ]
            named_captures_only => true
        }
        date {
            match => [ "logtimestamp", "YYYY/MM/dd HH:mm:ss,SSS" ]
        }
  .............
    }
}
```

Конфигурация Logstash: секция **output**

```
output {
        elasticsearch {
                embedded => true
                cluster => "logserver"
                embedded_multicast_enabled => "false"
                embedded_unicast_hosts => ".....u4-pri.uk.db.com:9300,.....u5-pri.uk.db.com:9300,....."
            embedded_file_lock => "/local/...../logserver.lock"
                embedded_path_data => "/local/...../logserver/storage"
                embedded_path_logs => "/local/...../logserver/logs"
        }
}
```

Удаление индексов старше заданной даты

```
public class CleanupRiver extends BaseScheduledRiver {
    @Override
    protected Runnable getRiverTask() {
        return new Runnable() {
            @Override
            public void run() {
                LOGGER.info("Cleanup indexes older than {} days from task start time.", getDays());
                SimpleDateFormat indexDateFormat = new SimpleDateFormat("yyyy.MM.dd");
                ClusterStateResponse clusterStateResponse = getClient().admin().cluster().prepareState().execute().actionGet();
                ImmutableOpenMap<String, IndexMetaData> indices = clusterStateResponse.getState().getMetaData().getIndices();
                Iterator<ObjectObjectCursor<String, IndexMetaData>> iterator = indices.iterator();
                while (iterator.hasNext()) {
                    ObjectObjectCursor<String, IndexMetaData> next = iterator.next();
                    String index = next.value.index();
                    try {
                        if(!index.startsWith("logstash")) continue;
                        String source = index.split("-")[1];
                        Date date = indexDateFormat.parse(source);
                        long intervalInDays = (System.currentTimeMillis() - date.getTime()) / TimeUnit.DAYS.toMillis(1);
                        if (intervalInDays > getDays()) {
                            LOGGER.info("delete index '"+index+"', which was created "+intervalInDays+" days ago");
                            getClient().admin().indices().delete(new DeleteIndexRequest(index));
                        }
                    } catch (Exception e) {
                        LOGGER.error("Skip index '"+index+"'", e);
                    }
                }
                LOGGER.info("finish cleanup");
            }
        };
    }
```

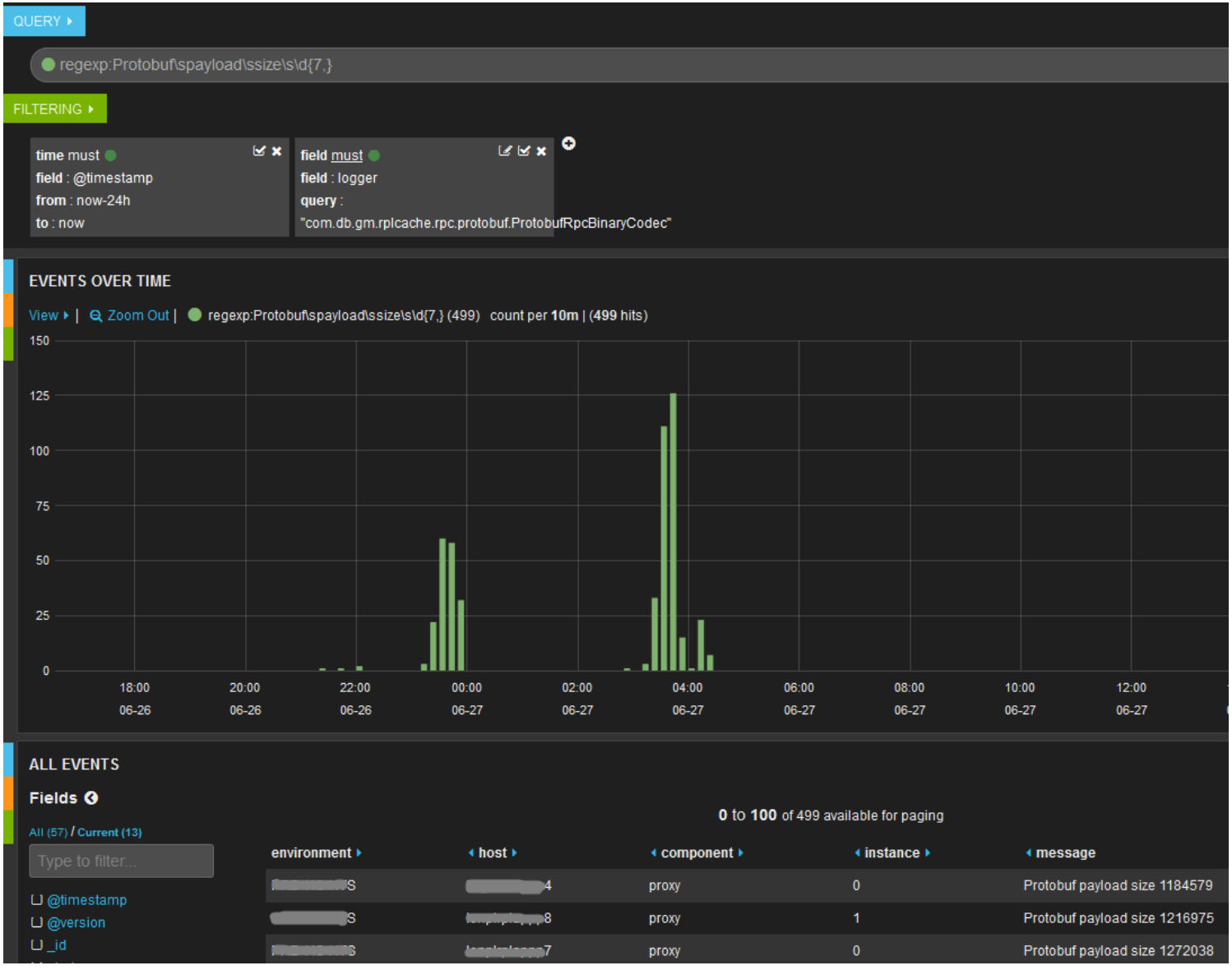
Поиск в Elasticsearch регулярными выражениями

— The regexp query allows you to use regular expression *term queries*. ... The "term queries" in that first sentence means that Elasticsearch **will apply the regexp to the terms produced by the tokenizer for that field**, and **not to the original text of the field**.

— Выход: elasticsearch scripting и перезапись запроса

The scripting module **allows to use scripts in order to evaluate custom expressions**. For example, scripts can be used to return "script fields" as part of a search request, or can be used to evaluate a custom score for a query and so on. The scripting module uses by default mvel as the scripting language with some extensions. mvel is used since it is extremely fast and very simple to use, and in most cases, simple expressions are needed (for example, mathematical equations).

Поиск в Elasticsearch регулярными выражениями



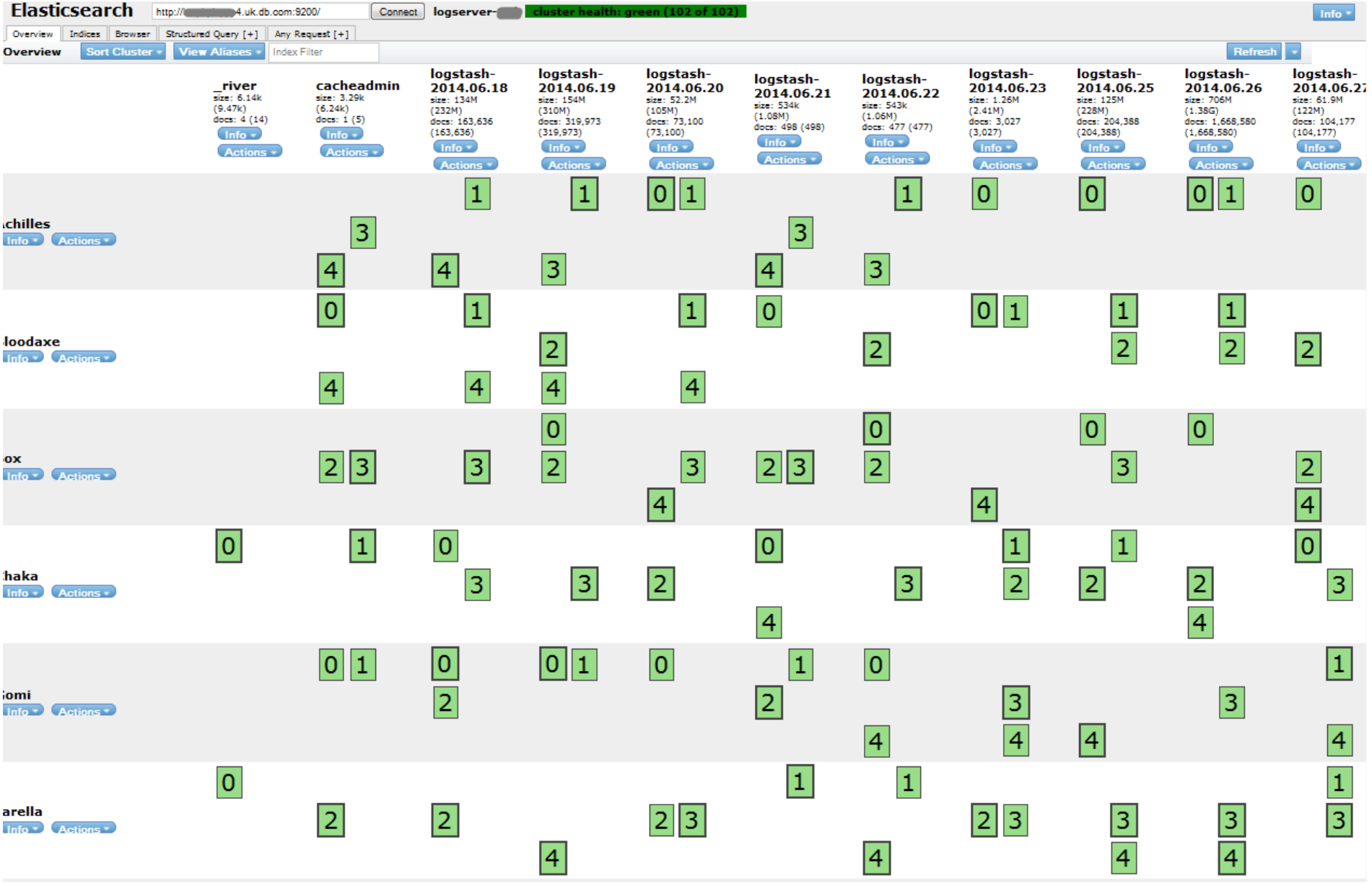
Как еще можно использовать информацию из ES

— JavaScripts/scripts
  - REST API;
  - jqPlot;
  - gnuplot JS

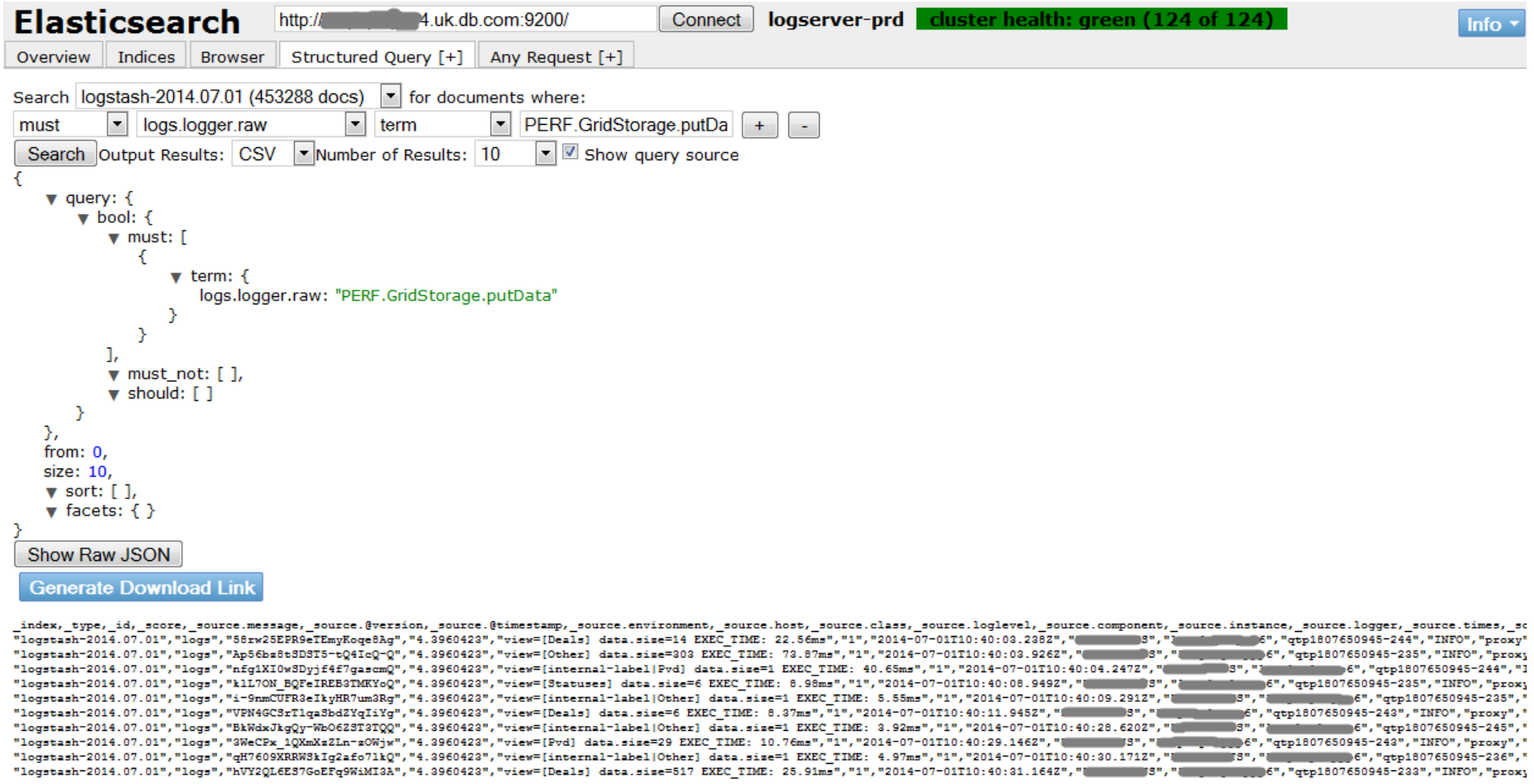
— Shell script
  - es2unix
  - cURL + jq

— Java
  - Elasticsearch java API;
  - JasperReport/JasperServer;
  - Sencha GXT plot.

Мониторинг и администрирование ES: Elastic HQ



Мониторинг и администрирование ES: Elasticsearch head



Мониторинг и администрирование ES: Marvel

Результаты

- Система, агригирующая информацию кластера из лог файлов и специфичных для приложения агентов сбора метрик;
- Полнотекстовый поиск и поиск по регулярным выражениям;
- Визуальное построение фильтров поиска;
- Возможность визуализации запросов в Kibana, подсчет агрегатов численных значений, а также частоты появления строки в результатах запроса;
- Автоматическое удаление индексов старше N дней;
- Возможность автоматизации действий над данными с помощью REST интерфейса Elasticsearch;

Ресурсы

- http://www.elasticsearch.org
- http://www.elasticsearch.org/overview/logstash
- http://www.elasticsearch.org/overview/kibana
- http://www.elastichq.org
- http://mobz.github.io/elasticsearch-head
- http://www.elasticsearch.org/overview/marvel


?
Q
&
A
!