

产品简介

日志系统是为解决分布式架构下日志收集、查询困难的一款日志产品，基于业界主流的全文检索引擎，通过蓝鲸智云的专属 agent 进行日志采集，提供多种场景化的采集、查询功能。

产品优势

- 功能强大的查询语法
- 实时日志和日志上下文
- 简单易用的日志采集
- 可视化的日志字段提取

术语解释

- **索引集**: 一个或多个符合一定条件的ES索引的集合,检索和监控的前提都要形成索引集. 具体的使用查看 [索引集管理](#)
- **数据分类**: 数据分类是基于采集对象来确定的,这个数据的分类基本同监控分类的相同.使用了这个数据上也将追加这个数据标签. 内置的数据维度也会有所区别.

主机 (1)

操作系统

服务 (1)

服务模块

应用 (1)

业务应用

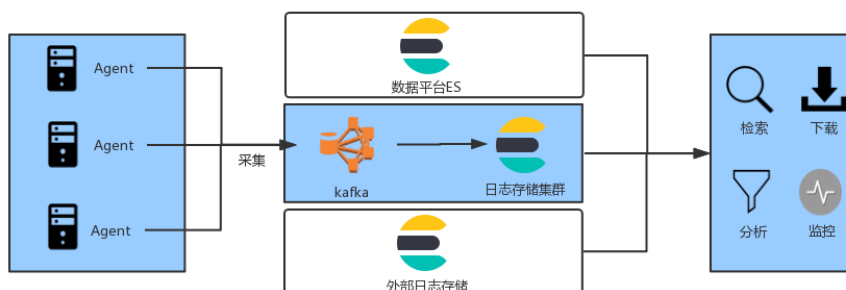
其他 (1)

1. **应用**: 指的是用户使用应用的情况,应用的运营数据. 如 移动端的使用情况, 业务应用的登陆

数等.

2. **服务**: 指的是运行在服务器操作系统之上的服务模块. 如 数据库, 进程等. 对应 CMDB-服务拓扑,对于多实例的数据采集时会有所区分.
3. **主机**: 指的是主机系统和硬件的层面. 如 CPU MEM 服务器硬件故障等. 对应 CMDB-主机拓扑
4. **数据中心**: 指的是和数据中心相关的网络和设备相关内容. 对应 CMDB-设备管理

产品架构图

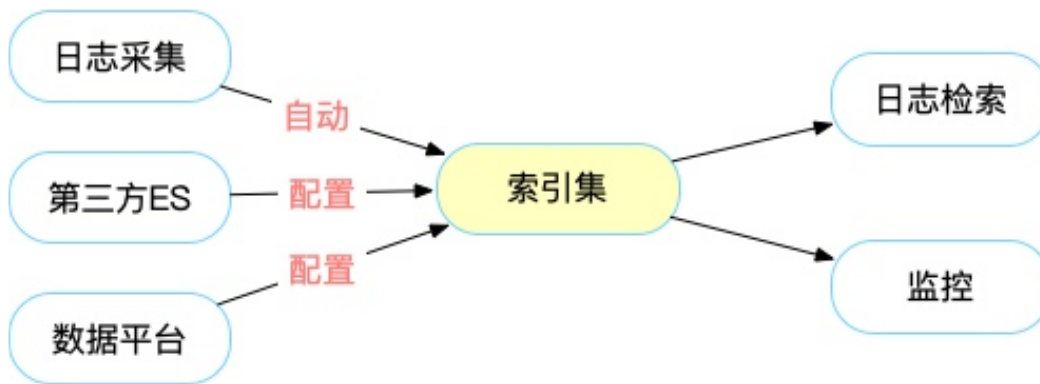


日志检索

日志检索主要是用来快速定位问题,避免在服务器端进行日志的查询,优点是性能高效和工具便捷.

前置步骤

能够进行日志的检索,需要已经有数据源的接入并且形成 **索引集**,才可以用来检索和监控.



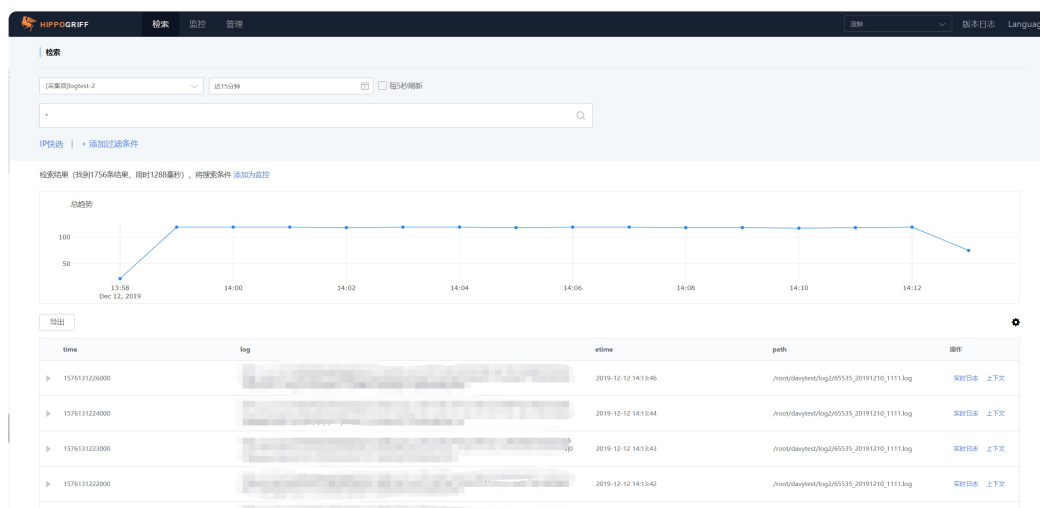
功能列表

主功能一览

- 日志检索和展示
- 过滤条件和 IP 快选
- 实时日志滚动
- 上下文
- 字段显示和排序

功能介绍

检索语句 支持 QueryString 语法和正则匹配。具体的查询语法查看[query string详解](#)



添加过滤条件 可以更精确的定位到日志内容

添加过滤条件

log_level

time

modulename

processid

serverip

lineno

message

is

请输入，注意空格符号

添加

取消

IP快选 通过关联 CMDB 的业务拓扑，控制日志检索范围。

IP快选

* 选择业务:

蓝鲸

* 选择方式:

拓扑选择

手动输入

已选择0个模块

搜索...

1. 业务

2. 模块

3. 平台

4. 应用

5. 组件

6. 服务

7. 实例

8. 节点

9. 设备

10. 网络

11. 存储

12. 安全

13. 运维

14. 监控

15. 日志

16. 告警

17. 报表

18. 其他

保存

取消

实时日志滚动

实时滚动日志



IP: 日志路径:

```
(0),duplicate(0),none_point_counts(0),time range(2019-12-27 03:39:00+0000 - 2019-12-27 04:09:00+0000)
492 2019-12-27 12:10:25 INFO      26102  access.event      processor.py[270] pull AccessCustomEventProcess event_record from kafka
finished(0)
493 2019-12-27 12:10:24 INFO      32572  access.data      processor.py[185] strategy(204),item(238),total_records(0),access records
(0),duplicate(0),none_point_counts(0),time range(2019-12-27 03:39:00+0000 - 2019-12-27 04:09:00+0000)
494 2019-12-27 12:10:29 INFO      27280  self_monitor    processor.py[037] metric record[access_custom_event.status] create: Fals
e
495 2019-12-27 12:10:29 INFO      27280  self_monitor    processor.py[037] metric record[access_data.status] create: False
496 2019-12-27 12:10:29 INFO      27280  self_monitor    processor.py[037] metric record[access_gse_event.status] create: False
497 2019-12-27 12:10:29 INFO      27280  self_monitor    processor.py[037] metric record[access_real_time_data.status] create: Fa
lse
498 2019-12-27 12:10:31 INFO      26070  access.event    processor.py[225] topic(0bkmonitor_10000) poll alarm list(0)
499 2019-12-27 12:10:31 INFO      26070  access.event    processor.py[138] push AccessGseEventProcess event_record to match queue
finished(0)
```

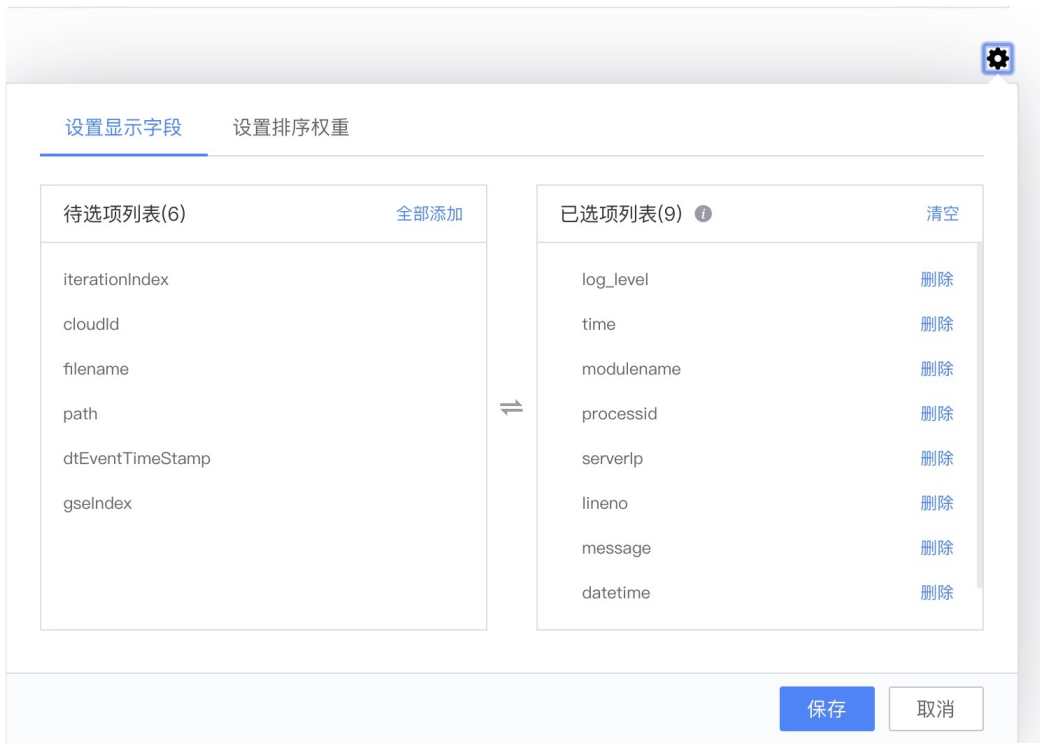
上下文查看



上下文

```
(0),duplicate(2130),none_point_counts(71),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
-3 2019-12-27 12:08:51 INFO      26070  access.event    processor.py[225] topic(0bkmonitor_10000) poll alarm list(0)
-2 2019-12-27 12:08:51 INFO      26070  access.event    processor.py[138] push AccessGseEventProcess event_record to match queue fin
shed(0)
-1 2019-12-27 12:08:52 INFO      31858  access.data    processor.py[185] strategy(138),item(170),total_records(3906),access records
(0),duplicate(3606),none_point_counts(300),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
0 2019-12-27 12:08:53 INFO      30735  access.data    processor.py[185] strategy(276),item(310),total_records(0),access records(0),
duplicate(0),none_point_counts(0),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
1 2019-12-27 12:08:53 INFO      31859  access.data    processor.py[185] strategy(280),item(314),total_records(2),access records(0),
duplicate(2),none_point_counts(0),time range(2019-12-27 04:06:00+0000 - 2019-12-27 04:07:00+0000)
2 2019-12-27 12:08:54 INFO      31859  access.data    processor.py[185] strategy(205),item(239),total_records(0),access records(0),
duplicate(0),none_point_counts(0),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
3 2019-12-27 12:08:54 INFO      31862  access.data    processor.py[185] strategy(183),item(217),total_records(0),access records(0),
duplicate(0),none_point_counts(0),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
4 2019-12-27 12:08:54 INFO      31861  access.data    processor.py[185] strategy(233),item(267),total_records(0),access records(0),
duplicate(0),none_point_counts(0),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
5 2019-12-27 12:08:54 INFO      30733  access.data    processor.py[185] strategy(266),item(300),total_records(0),access records(0),
duplicate(0),none_point_counts(0),time range(2019-12-27 03:37:00+0000 - 2019-12-27 04:07:00+0000)
6 2019-12-27 12:08:54 INFO      31858  access.data    processor.py[185] strategy(182),item(216),total_records(0),access records(0)
```

字段显示和顺序,还有多列排序功能.



第三方ES接入

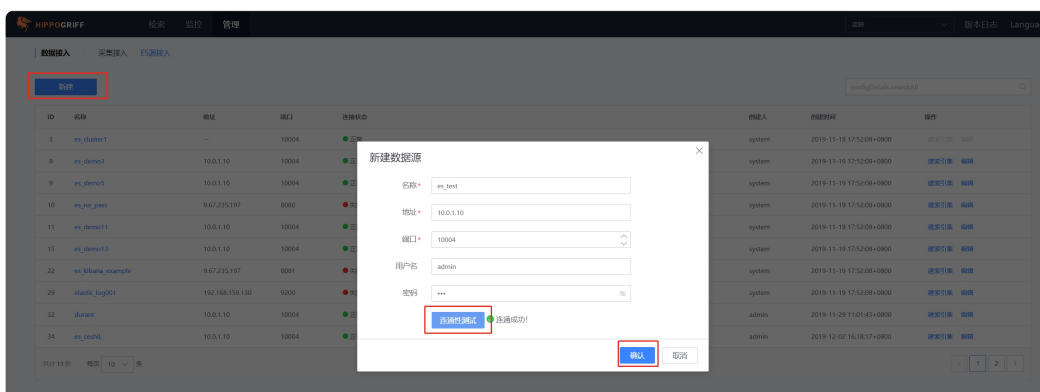
第三方 ES 接入的场景主要是解决,已存在的 ES 集群接入日志服务使用,或者是从采集入到第三方的集群中。

前置步骤

功能介绍

- 先接入 ES 集群

导航路径: 管理 → 数据接入 → ES源接入 → 新建



输入完整的 ES 集群地址后, 进行连通性测试, 测试通过方可保存。

“

- 创建索引集

接入第三方 ES 的集群后,要能够真正的使用起来,还需要创建索引集才可以使用.

导航路径: 管理 → 索引集管理 → 新建

新增索引

*索引名称: log_from_es

*数据分类: 服务-服务模块

*数据源: ☐ 采集接入 ☐ 数据平台 ☒ 第三方ES

es_demo5

新增索引

索引: 索引名称

索引: 2_bklog_test1224_20191211

支持 "*" 匹配, 不支持其他特殊符号

成功匹配索引设置

索引

2_bklog_test1224_20191211_0

2_bklog_test1224_20191211_write

共计 2 条

*时间字段: 请选择

确认 取消

支持 "*" 匹配多个索引，要求所有匹配到的索引的字段一致，并且索引必须包含一个时间字段，否则无法完成接入。

- 采集接入

想把采集数据接入到第三方ES的集群,在采集的时候选择第三方的ES集群即可. 具体查看[采集接入](#)

索引集管理

索引集管理为管理员提供索引集的增、删、查、改功能

存储在 ES 中的日志要先通过建立索引集，才能进行查询。查询的对象为索引集。

资源管理

资源

资源名称

资源类型

资源来源

资源名称

创建时间

创建人

资源权限

操作

资源ID	资源	资源分类	资源源	资源名	创建时间	创建人	资源权限	操作
test3	2_bkdata_search; 2_bkdata_test	测试	数据平台	--	2019-12-12 11:02:21+0800	admin	运维人员; 开发人员; 产品人员; 日常普通用户	编辑 删除
test1	2_bkdata_search	操作系统	数据平台	--	2019-12-07 21:52:53+0800	admin	运维人员	编辑 删除
测试日志1	2_bklog.monitor	操作系统	采集器输入	es_cluster1	2019-12-07 21:52:52+0800	admin	运维人员	编辑 删除
transfer101	2_bklog.transfer101	操作系统	采集器输入	es_cluster1	2019-11-30 14:33:35+0800	admin	运维人员; 开发人员; 产品人员; 日常普通用户	编辑 删除
transfer_search_2	2_bkdata_search	数据平台	--	--	2019-11-12 11:49:27+0800	admin	运维人员; 开发人员; 产品人员; 日常普通用户	编辑 删除

共15条

每页 10 条

1 2

索引集列表字段解读

索引集： 用户自定义索引集合的名称，支持中英文

索引： 保存在 ES 中的 index

数据分类： 日志类型标签

数据源： 数据平台、采集接入、第三方 ES。

“

数据平台 指日志来自数据平台ES； 采集接入 指日志来自日志检索内部采集； 第三方 ES 指接入用户独立的第三方ES集群。

”

集群名： 用户独立的第三方ES集群的名字

新建索引集

在日志检索内采集接入的日志，会自动创建对应的索引集。 其他情况需要管理员手动创建索引集。

← 新建索引集

索引集名称

新建索引集测试

数据分类

服务-服务模块

数据源

采集接入

数据平台

第三方ES

新增索引

索引	业务	权限	操作
2_bkdata_test	蓝鲸	--	删除

查看权限

开发人员,产品人员

提交

取消

新增索引

* 业务

蓝鲸

* 索引

2_bkdata_search

找不到需要的日志? 到 [数据平台](#) 采集接入

字段	类型
log	text
ip	string
report_time	string
gseindex	long
path	string

确认

取消

新增索引

* 索引

*_bklog_regexp121103_20190924_write

支持 “*” 匹配, 不支持其他特殊符号

成功匹配索引1条

索引
*_bklog_regexp121103_20190924_write2_bklog_regexp121103_20190924_write

共计 1 条

<

1

>

* 时间字段

datetime

确认

取消

新增索引

*索引

2_bklog_bennynginx_*

支持 "*" 匹配, 不支持其他特殊符号

成功匹配索引28条

索引
2_bklog_bennynginx_20191121_write
2_bklog_bennynginx_20191203_write
2_bklog_bennynginx_20191127_write
2_bklog_bennynginx_20191111_write

*时间字段

请选择

确认

取消

编辑索引集

编辑索引集可修改索引集名称、数据分类、增删索引、修改授权, 不能修改数据源类型。

编辑索引集

*索引集名称

test3

*数据分类

其他-其他

*数据源

采集接入

数据平台

第三方ES

新增索引

索引	业务	权限	操作
2_bkdata_search	蓝鲸	正常	删除
2_bkdata_test	蓝鲸	正常	删除

*查看权限

运维人员,开发人员,产品人员,日志查询组A

提交

取消

用户管理

用户管理主要是解决用户的权限问题,在采集,索引集等设置时需要设置权限。

权限如何申请

1. 先申请配置平台某业务的权限(至少需要查看权限)
2. 加入某个角色. (支持的角色: 运维人员 开发人员 产品人员)
 - 如果自己有配置平台业务权限,在业务里面编辑角色加入到某个角色中.
 - 如果自己没有配置平台业务权限,找管理员进行添加.
3. 如果要排查错误,查看作业执行历史,需要申请作业平台的执行历史所有查看权限.
4. 如果要创建新的用户组,使用 **用户组** 创建功能.

用户组创建

运维人员、开发人员、产品人员 三个用户组为默认组，默认组内成员从 CMDB 同步，不支持修改。

管理员可以新建用户组，以支持更多样的使用场景。

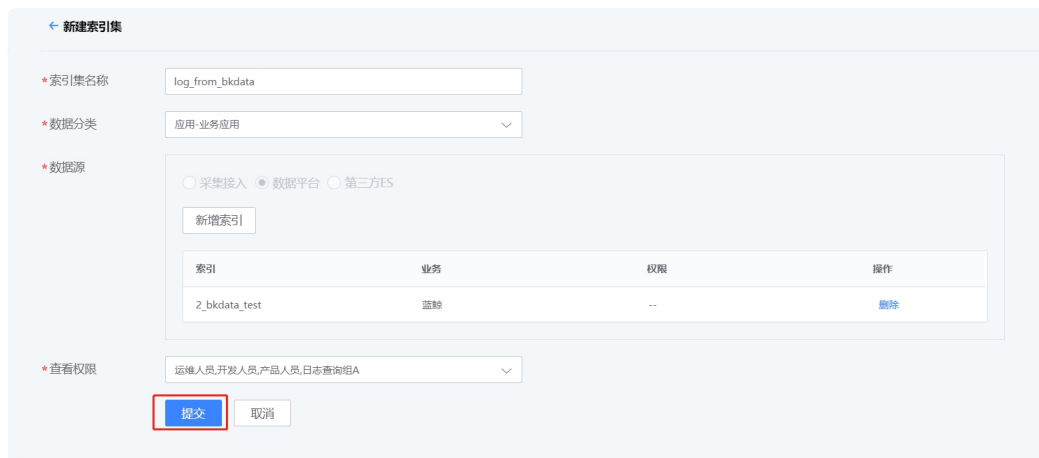
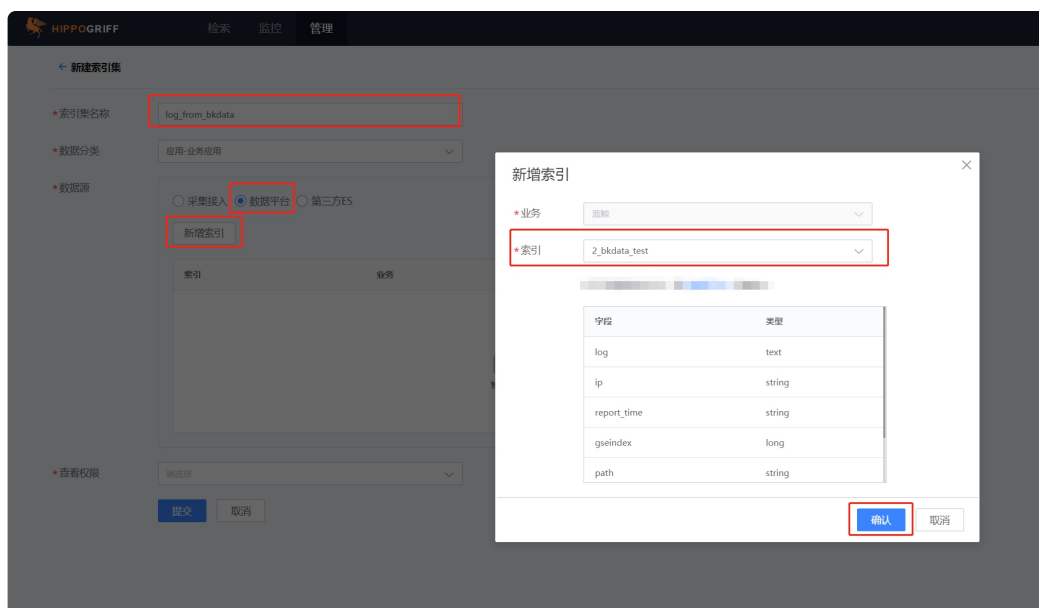
用户组内新增用户，新用户自动获得所在用户组对应的所有权限。

用户管理				
新建				
组名	成员	创建时间	创建人	操作
运维人员	admin; hallelujah; jay; jic; jianshang; logman; sun_test; cctest; edwin;	2019-11-12 11:47:12+0800		编辑 删除
开发人员		2019-11-12 11:47:12+0800		编辑 删除
产品人员	hengchen;	2019-11-12 11:47:12+0800		编辑 删除
日志索引组	admin;	2019-12-07 21:53:49+0800	admin	编辑 删除

接入数据平台的日志

1. 在数据平台完成日志的采集、清洗和存储，具体操作可查看数据平台相关指引

导航路径: 管理 → 索引集管理 → 新建



“

注意：一个索引集下面添加多个索引时，各索引的字段要保持一致。提交后数据源类型不可变更。

”

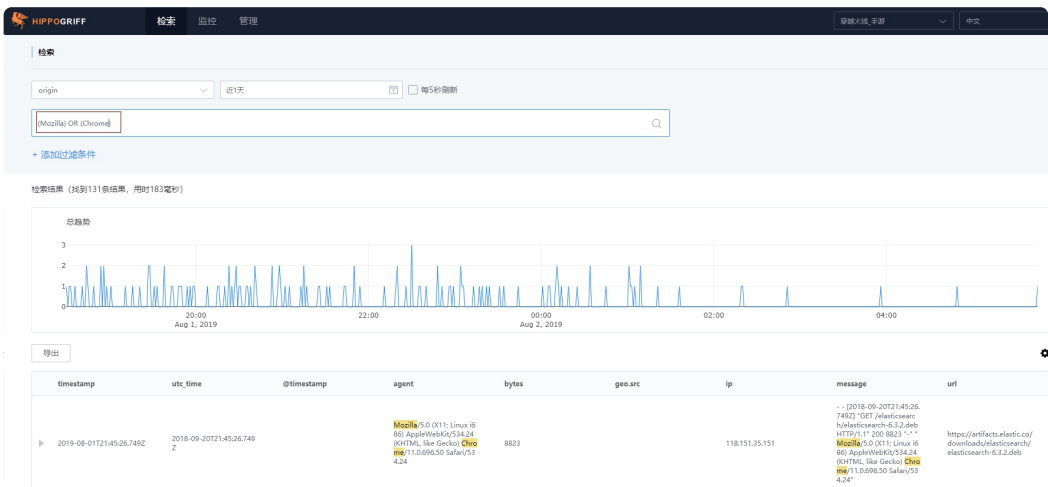
检索语法指南

在日志检索当中，使用了标准的 Elasticsearch QueryString，基本支持所有语法。下面是基础的语法介绍。

QueryString简介

查询后台会根据 QueryString 传入的文本信息进行解析，解析的语法原则是根据具体的操作符进行分割判断，并对分割后的每一段进行独立分析，然后进行查询分析。

如图：



查询语句 (Mozilla) OR (Chrome)

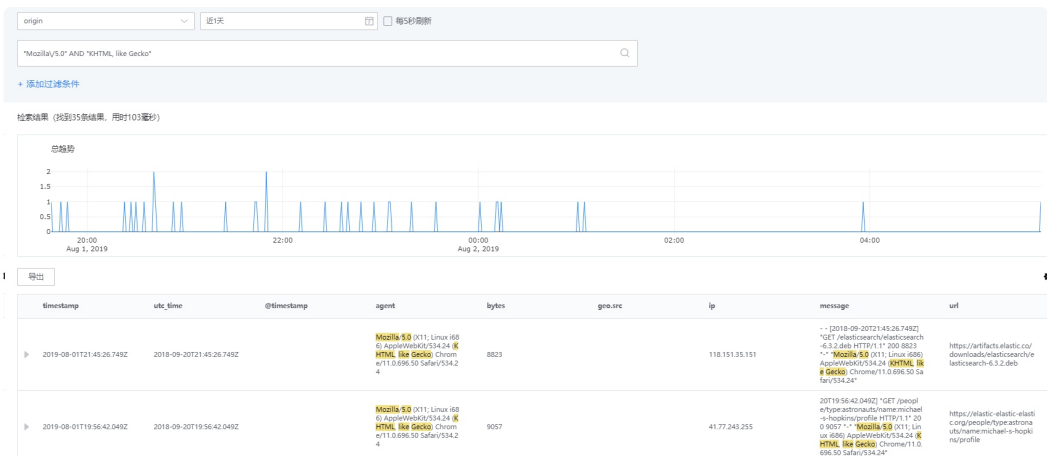
查询语句会根据操作分成 Mozilla 和 Chrome 两个部分，每个部分都会被查询后台独立进行分析。

“ 注意：“空格”不会被作为是操作符，如查询(KHTML, like Gecko) OR (200 8823), KHTML, like Gecko 会被完整的传入到后台去进行查找分析。能够搜索出只有 Gecko 或者只有200的字段，因为具体的字段在ES后天进行了分词配置。如果想独立搜索200和8823可通过语句200 AND 8823查询。 ”

QueryString语法

基础语法

上面提到 QueryString 的内容会被解析成词语或者操作符，词语可以是单词-如 MozillaORChrome，也可以是个短语语句，被双引号包围 "Mozilla\5.0" AND "KHTML, like Gecko"。

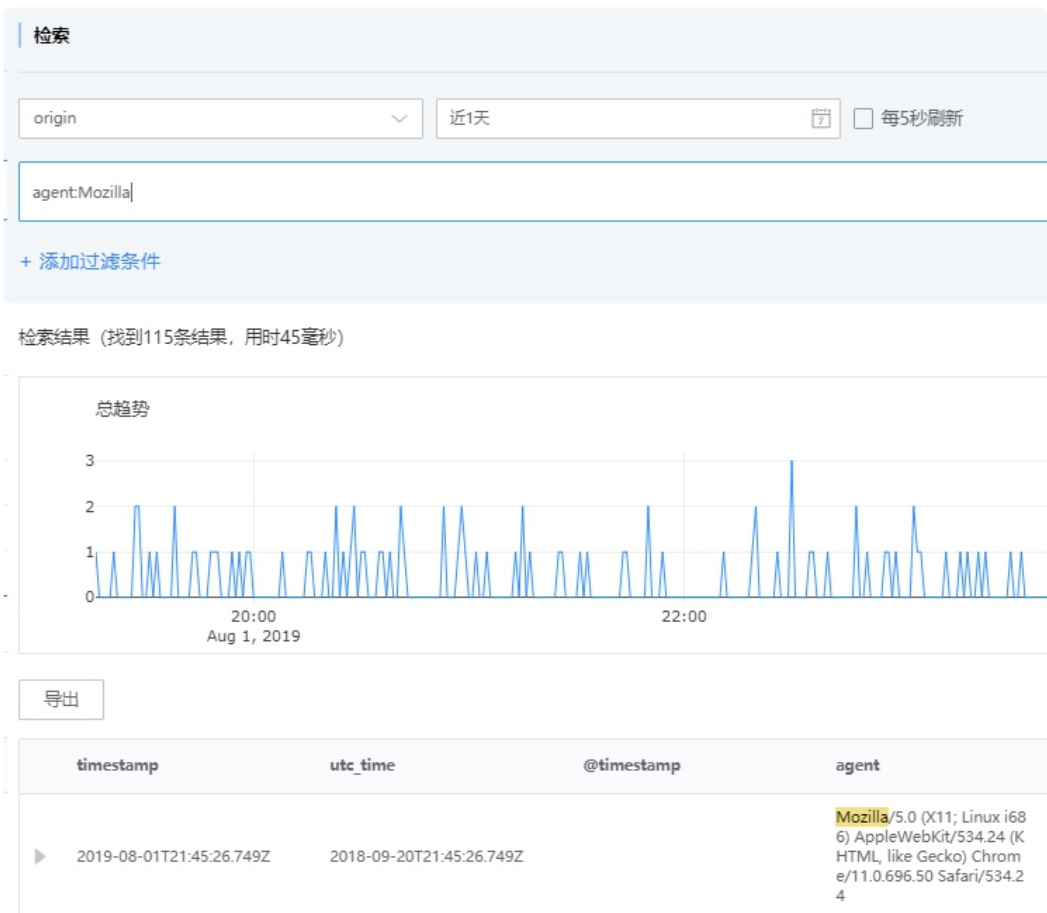


查询指定字段

默认情况下 QueryString 会将查询内容解析到所有的字段(_all)，可以通过字段设定固定查询具体的字段。

如： 查询 agent 里面包含 Mozilla

agent:Mozilla



查询agent里面包含Mozilla或者X11或者4.0

agent: Mozilla OR X11 OR 4.0

▶	2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	8823
▶	2019-08-01T20:49:29.440Z	2018-09-20T20:49:29.440Z	Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1	2950
▶	2019-08-01T19:56:42.049Z	2018-09-20T19:56:42.049Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	9057
▶	2019-08-01T18:51:16.933Z	2018-09-20T18:51:16.933Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	2189
▶	2019-08-01T18:20:52.083Z	2018-09-20T18:20:52.083Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	209
▶	2019-08-01T18:19:02.490Z	2018-09-20T18:19:02.490Z	Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1	5207
▶	2019-08-01T17:36:12.827Z	2018-09-20T17:36:12.827Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	2188

查询agent里面具体的短语(使用双引号包裹)

agent: "KHTML, like Gecko"

timestamp	utc_time	@timestamp	agent
▶ 2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T19:56:42.049Z	2018-09-20T19:56:42.049Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T17:05:26.359Z	2018-09-20T17:05:26.359Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T17:01:38.841Z	2018-09-20T17:01:38.841Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T16:14:35.212Z	2018-09-20T16:14:35.212Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T16:12:51.873Z	2018-09-20T16:12:51.873Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24

查询 geo.通配符含有CN的内容

```
geo.\*:CN
```

查询 geo.src 非空

```
_exists_: geo.src
```

通配符

通配符支持 ? 和 *

? 替换一个单独的字符

* 替换一个0个或者多个字符

这个与正则表达式类似

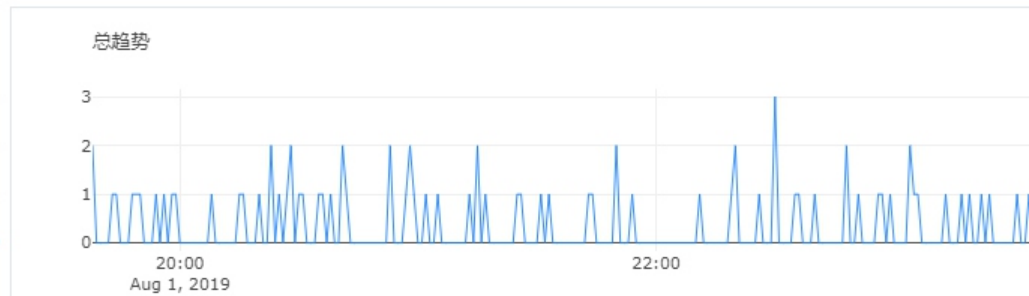
如查询

```
agent: M?zi*a
```


agent: M?zi*a

+ 添加过滤条件

检索结果 (找到107条结果, 用时71毫秒)



导出

timestamp	utc_time	@timestamp	agent
▶ 2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24
▶ 2019-08-01T20:49:29.440Z	2018-09-20T20:49:29.440Z		Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1
▶ 2019-08-01T19:56:42.040Z	2018-09-20T19:56:42.040Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko)

“

注意：通配符查询是会消耗大量内存的，在一个短语或者词语中存在多个通配符，会导致大量的词语列入搜索范围。

”

如：agent: zia

能够查询出 agent 中包含的结果，但是对于 agent 字段的所有短语都会进行检索，消耗大量的时间和后台内存。考虑必要的查询场景，日志检索没有对该功能进行禁用。使用通配符的时候需要详细考虑一下具体的查询语句。

正则表达式

正则表达式模式嵌套可以再 Query String中使用，使用时需要将查询内容包裹在两个正斜杠中(“/”)。

如：agent: /[L-N].*z*1{2}a/

origin

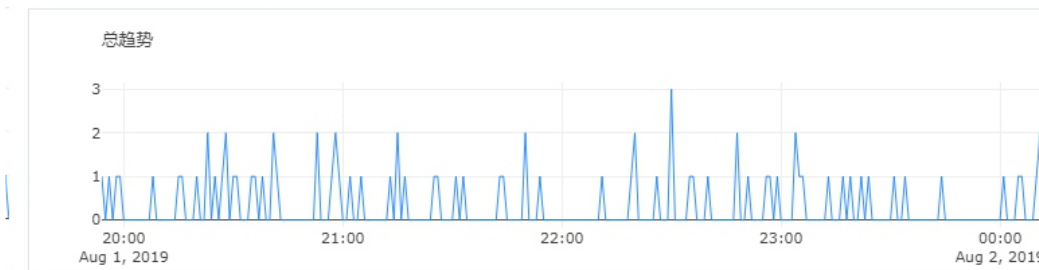
近1天

☐ 每5秒刷新

agent: /[L-N].*z{l(2)a/

+ 添加过滤条件

检索结果 (找到100条结果, 用时39毫秒)



导出

timestamp	utc_time	@timestamp	agent	bytes
▶ 2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	8823
▶ 2019-08-01T20:49:29.440Z	2018-09-20T20:49:29.440Z		Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1	2950
▶ 2019-08-01T19:56:42.040Z	2018-09-20T19:56:42.040Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko)	9057

模糊查询

可以带波浪号对末尾模糊查询

agent: Mozill~

可以针对短语进行模糊查询

agent:"KHTML Gecko"~2


origin
近1天
每5秒刷新

agent:"KHTML Gecko"

+ 添加过滤条件

检索结果 (找到0条结果, 用时1毫秒)

导出

timestamp	utc_time	@timestamp	agent	bytes	geo.src	ip	ma
 未查询到数据							

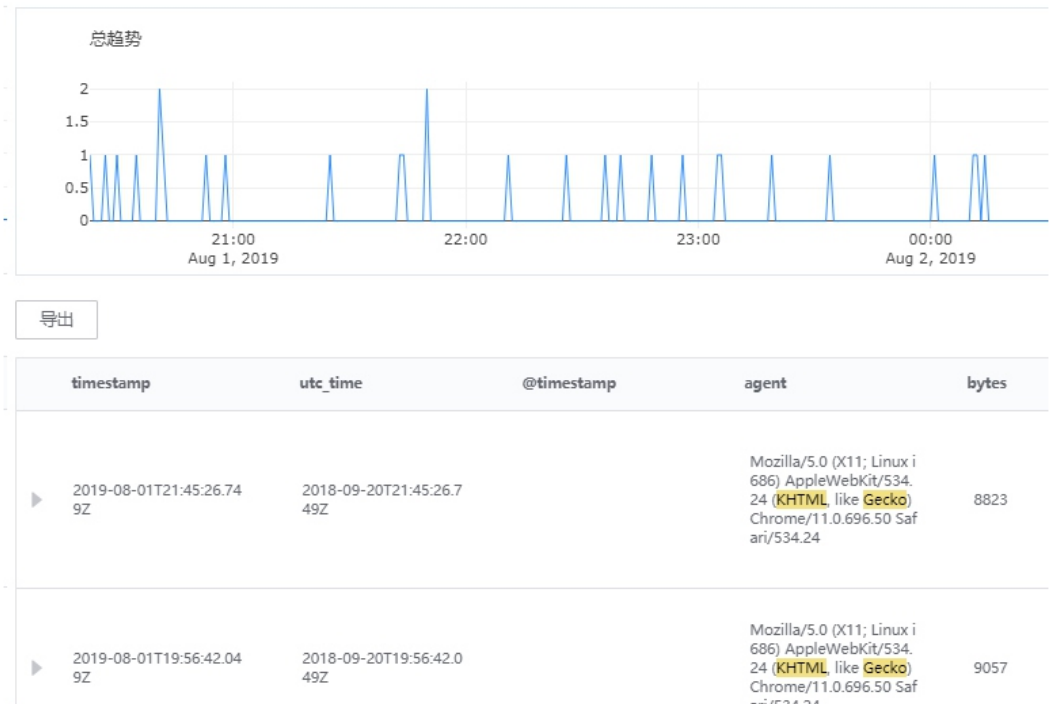
检索

origin
近1天
每5秒刷新

agent:"KHTML Gecko"~2

+ 添加过滤条件

检索结果 (找到32条结果, 用时37毫秒)



范围查询

范围查询是针对时间、数字和字符串类型的字段使用的。

范围查询的操作符主要是 `[]` 和 `{}`，其中 `[]` 是闭合查询，`{}` 非闭合查询。

如： 查询 Bytes 字段从8023到9057区间内的数据， 包含8023和9057

bytes: [8823 TO 9057]

timestamp	utc_time	@timestamp	agent	bytes
▶ 2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	8823
▶ 2019-08-01T19:56:42.049Z	2018-09-20T19:56:42.049Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	9057

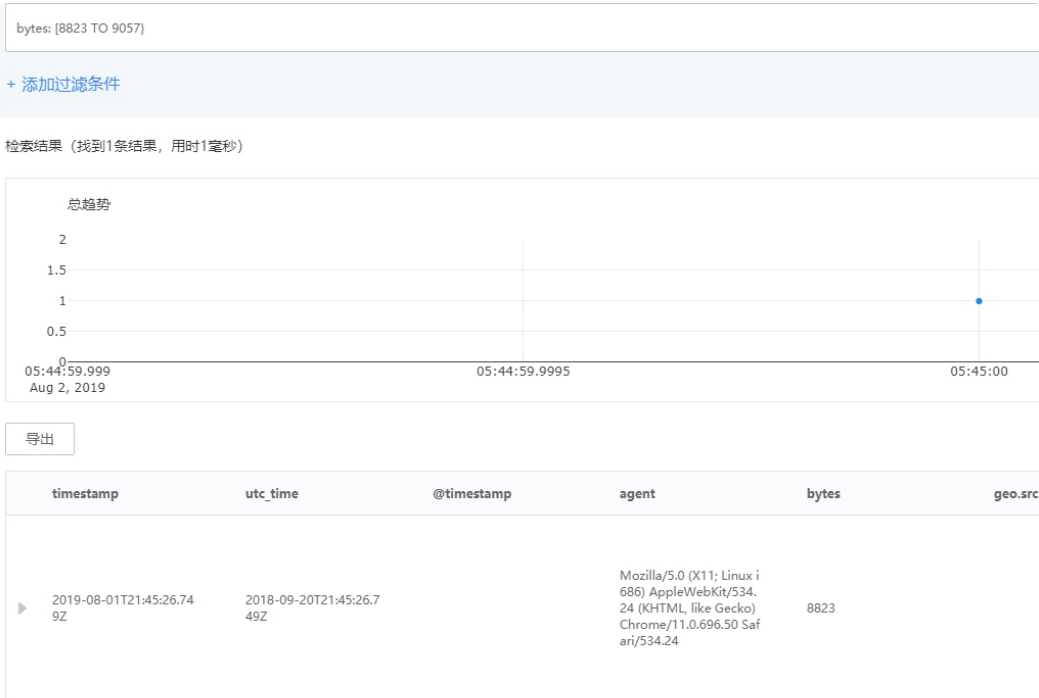
查询时间字段是时间类型的区间

timestamp:[2019-08-01T19:56:00 TO 2019-08-01T22:00:00]

timestamp	utc_time	@timestamp	agent	bytes
▶ 2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	8823
▶ 2019-08-01T20:49:29.440Z	2018-09-20T20:49:29.440Z		Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1	2950
▶ 2019-08-01T19:56:42.049Z	2018-09-20T19:56:42.049Z		Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24	9057

查询Bytes字段从8023到9057区间内的数据， 不包含9057

bytes: [8823 TO 9057}



Bool操作符

回顾一个最基础的查询

agent: Mozilla X11 4.0 5.0

在这个查询当中, 所有的短语都是可选的, 也就是会得到如下的结果

2019-08-01T21:45:26.749Z	2018-09-20T21:45:26.749Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	8823	118.151.35.151	6.74602 "GET /elasticsear ch/elasticsearch-6.3.2. deb HTTP/1.1" 200 8823 - - Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	https://artifacts.elastic.co/downloads/elasticsear ch/elasticsearch-6.3.2. deb
2019-08-01T20:49:29.440Z	2018-09-20T20:49:29.440Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	2950	99.76.103.49	- [2018-09-20T20:49:29.440Z] "GET /elasticsear ch/elasticsearch-6.3.2. deb HTTP/1.1" 200 2950 - - Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	https://artifacts.elastic.co/downloads/elasticsear ch/elasticsearch-6.3.2. deb
2019-08-01T19:56:42.049Z	2018-09-20T19:56:42.049Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	9057	41.77.243.255	2019-06-02.0492Z "GET /people/typeahead/v1/people/profile HTTP/1.1" 200 9057 - - Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	https://elastic-elasticsearch.org/people/typeahead/v1/people/profile
2019-08-01T18:51:16.933Z	2018-09-20T18:51:16.933Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	2189	74.184.0.64	[2018-09-20T18:51:16.933Z] "GET /beat/metricbeat/metricbeat-6.3.2-1-088.rpm HTTP/1.1" 200 2189 - - Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)"	https://artifacts.elastic.co/downloads/beat/metricbeat/metricbeat-6.3.2-1-088.rpm
2019-08-01T18:20:52.089Z	2018-09-20T18:20:52.089Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	209	213.116.126.196	213.116.126.196 - [2018-09-20T18:20:52.089Z] "GET /site-search HTTP/1.1" 200 209 - - Mozilla/4.0	https://www.elastic.co/solutions/site-search
2019-08-01T18:19:02.490Z	2018-09-20T18:19:02.490Z	Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/71.0.696.50 Safari/534.24	5207	232.20.97.5	2018-09-20T18:19:02.490Z "GET /beat/metricbeat/metricbeat-6.3.2-a-088.rpm HTTP/1.1" 200 5207 - - Mozilla/5.0	https://artifacts.elastic.co/downloads/beat/metricbeat/metricbeat-6.3.2-a-088.rpm

如果希望对查询操作有更多的控制, 可以通过Bool操作符如:

agent: Mozilla X11 +4.0 -5.0

对于这个查询的理解

Mozilla X11是可选的，主要满足其中之一，记录就会找出

4.0是必须的

5.0是不能存在的

得到如下的结果

▶	2019-08-01T18:51:16.933Z	2018-09-20T18:51:16.933Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
▶	2019-08-01T18:20:52.083Z	2018-09-20T18:20:52.083Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
▶	2019-08-01T17:36:12.827Z	2018-09-20T17:36:12.827Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
▶	2019-08-01T17:21:47.719Z	2018-09-20T17:21:47.719Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
▶	2019-08-01T17:09:57.561Z	2018-09-20T17:09:57.561Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
▶	2019-08-01T17:01:49.801Z	2018-09-20T17:01:49.801Z	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)

以上的查询可以被理解成

```
agent: ((Mozilla AND 4.0) OR (X11 AND 4.0) OR 4.0) AND NOT 5.0
```

相比而言使用 Bool 操作符就能够简单获得需要的结果。

条件分组

```
agent: ((Mozilla AND 4.0) OR (X11 AND 4.0) OR 4.0) AND NOT 5.0
```

上面的这种查询就是分组的示例，还可以针对不同的字段进行条件分组

bytes: "2189" AND agent: ((Mozilla AND 4.0) OR (X11 AND 4.0) OR 4.0) AND NOT 5.0

	timestamp	utc_time	@timestamp	agent	bytes
▶	2019-08-01T18:51:16.933Z	2018-09-20T18:51:16.933Z		Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)	2189

特殊字符的转义

如果希望查询的内容当中是包含日志检索的操作符，那么需要在 Query String 当中进行转义，通过反斜杠进行转义，需要转义的字符如下：

+ - = & | | > < ! () { } [] ^ " ~ * ? : \ /

如果忘记了转义会导致查询报错

如希望查询 agent 是 Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1

+ 添加过滤条件

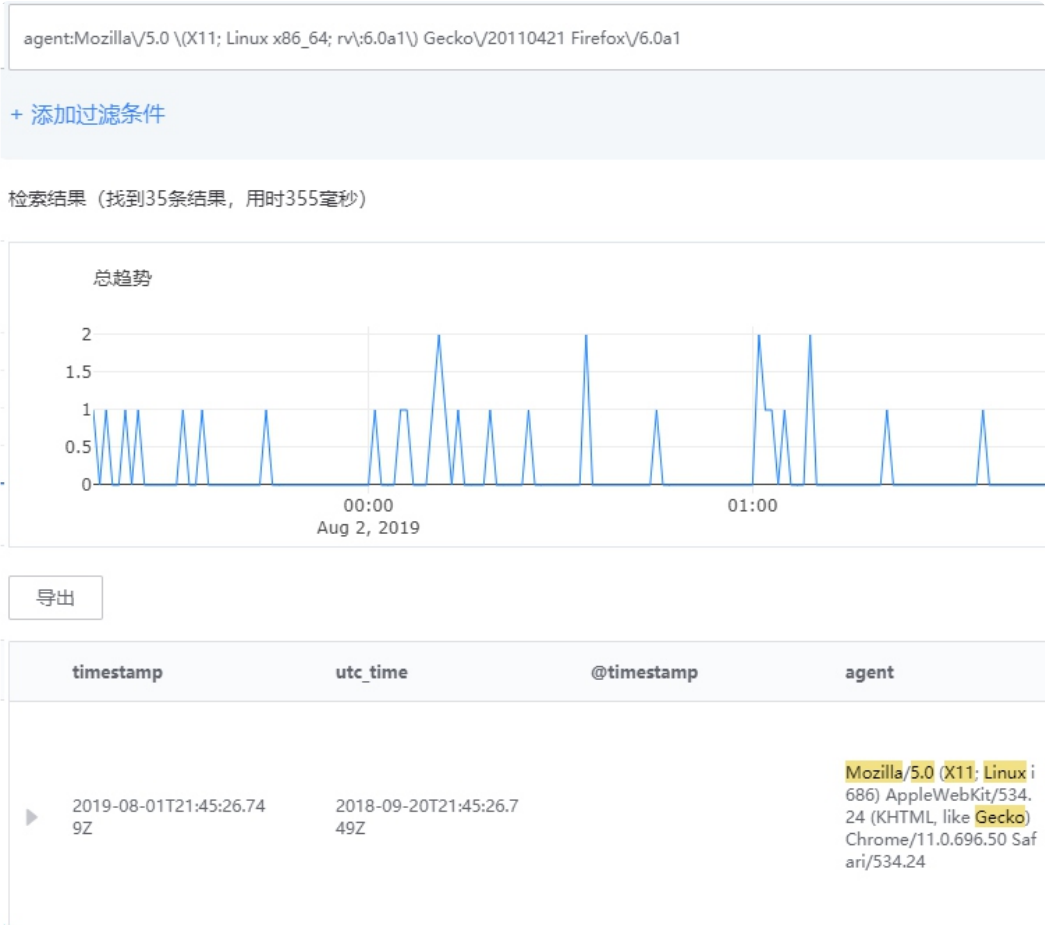
检索结果 (找到0条结果, 用时0毫秒)

导出

timestamp	utc_time	@timestamp	agent	bytes	geo.src	ip
<div>未查询到数据</div>						

通过转义

agent:Mozilla\\5.0 \\(X11; Linux x86_64; rv\\:6.0a1\\) Gecko\\20110421 Firefox\\6.0a1



转义后会转成分词的模式进行查询

常见通配符：

字符	含义	举例
*	匹配 0 或多个字符	a*b , a与b之间可以有任意长度的任意字符, 也可以一个也没有, 如aabc b , axyz b , a012 b , a b 。
?	匹配任意一个字符	a?b , a与b之间必须也只能有一个字符, 可以是任意字符, 如aab, abb, acb, a0b。
[list]	匹配 list 中的任意单一字符	a[xyz]b , a与b之间必须也只能有一个字符, 但只能是 x 或 y 或 z, 如: axb, ayb, azb。
[!list]	匹配 除list 中的任意单一字符	a[!0-9]b , a与b之间必须也只能有一个字符, 但不能是阿拉伯数字, 如axb, aab, a-b。
[c1-c2]	匹配 c1-c2 中的任意单一字符	[0-9] [a-z], a[0-9]b , 0与9之间必须也只能有一个字符 如a0b, a1b... a9b。
	匹配 sring1 或 string2 (或更多)其一字符串	a