Elasticsearch 在奇安信的大规模实践

张超
2019/11/24, 资深研发工程师,奇安信
《Elasticsearch 源码解析与优化实战》作者

主要内容

- Elasticsearch 在奇安信的应用情况
- 内核改进
- 生产环境建议

应用场景

集群规模

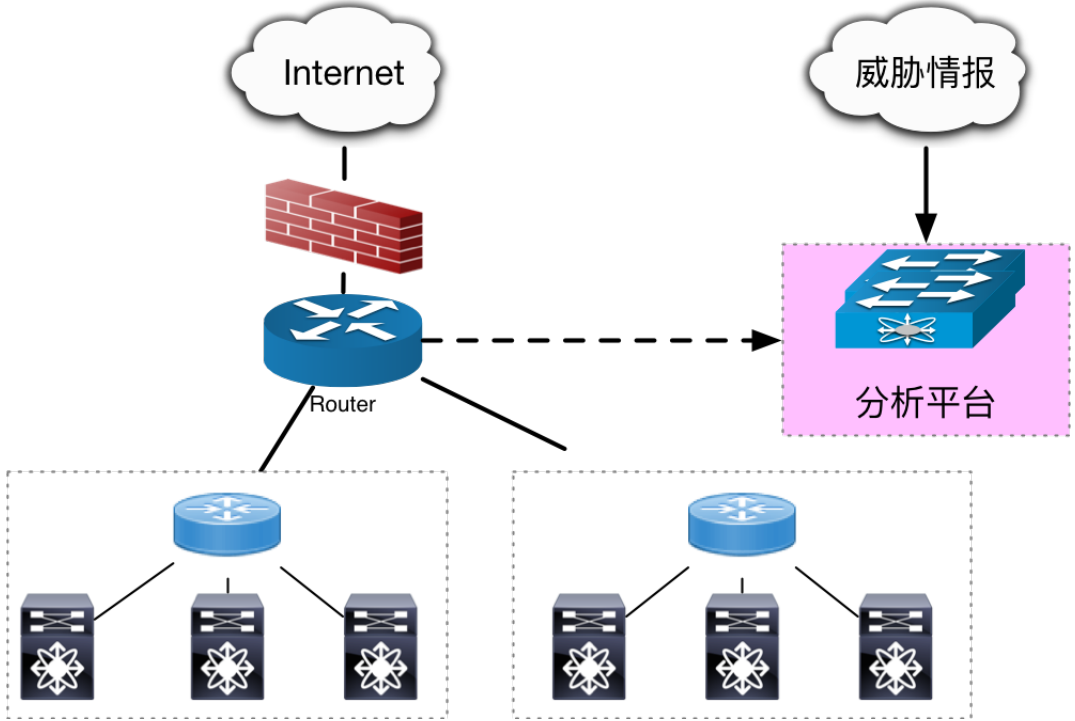
总体规模
节点数：6000+
数据总量：50PB+

单集群
最大100+节点
千亿条文档
百TB数据总量

典型场景
流量分析
告警信息
应用程序日志

应用场景-流量分析

从流量中发现威胁，包括 TCP，UDP，HTTP，DNS，MAIL，LDAP...

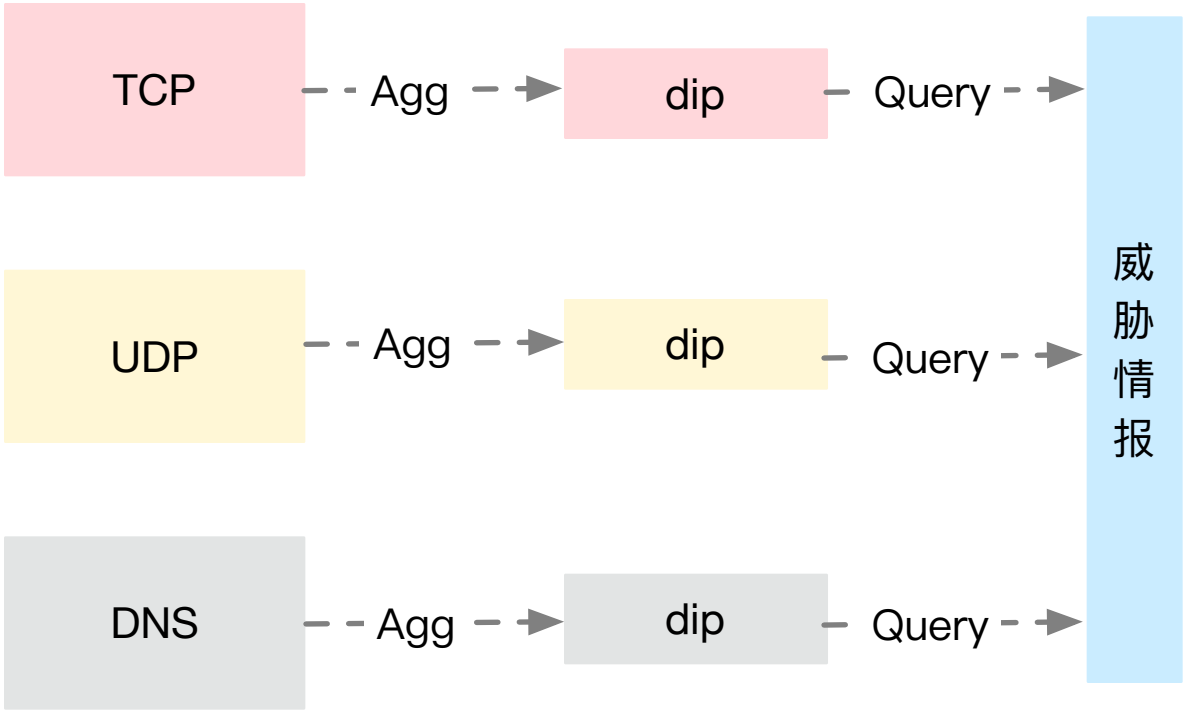


elastic

流量解析后的结果

```
"sport": { "type": "integer"},
"dport": { "type": "integer"},
"sip": { "type": "keyword"},
"dip": { "type": "keyword"},
"stime": { "type": "date", "format": "YYYY-MM-dd HH:mm:ss.SSSSSS"},
"dtime": { "type": "date", "format": "YYYY-MM-dd HH:mm:ss.SSSSSS"},
"proto": { "type": "keyword"},
"uplink_length": { "type": "integer"},
"downlink_length": { "type": "integer"},
"client_os": { "type": "text" },
"server_os": { "type": "text" },
"src_mac": { "type": "keyword"},
"dst_mac": { "type": "keyword"},
"down_payload": { "type": "text" },
"up_payload": { "type": "text" },
"summary": { "type": "keyword"}
```

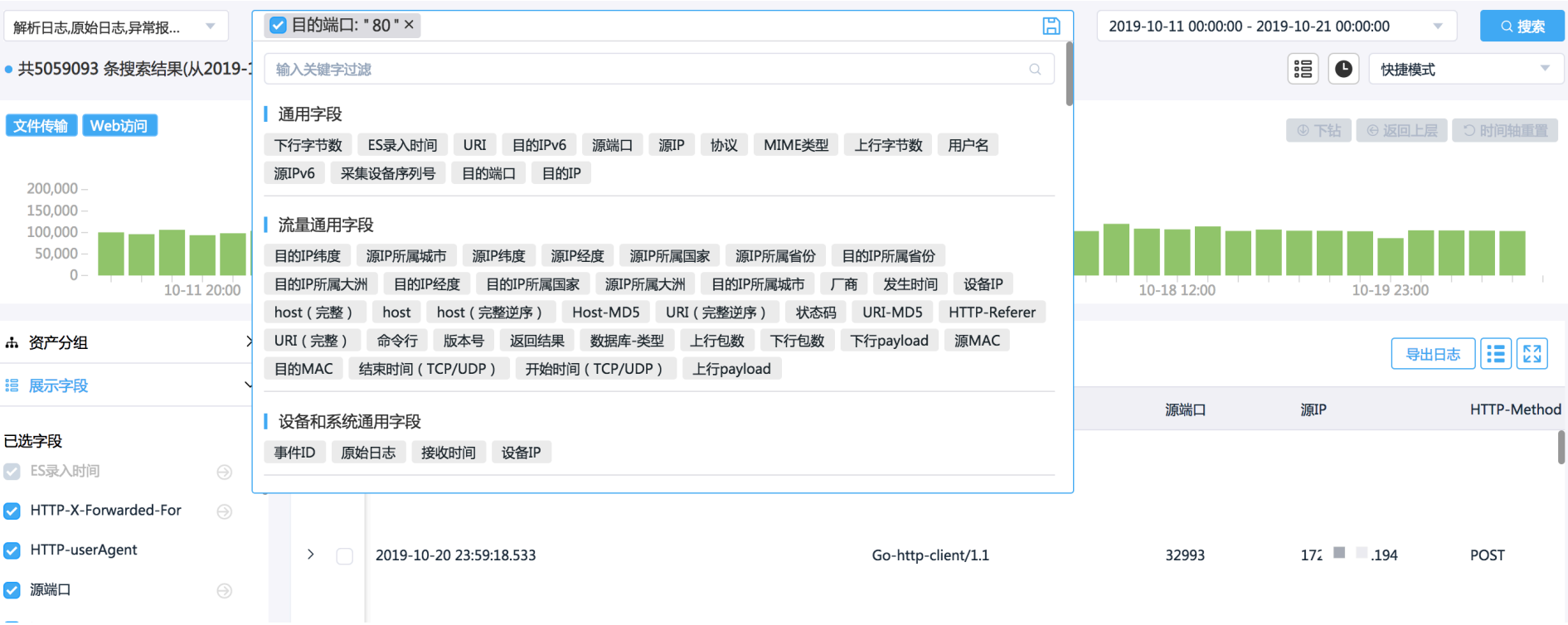
查询过程实现



流量分析-产生告警

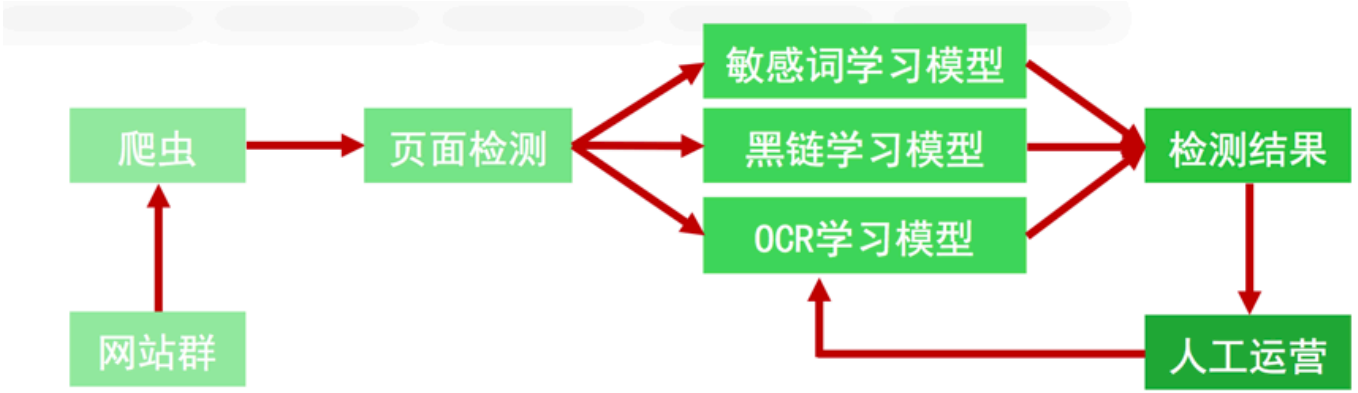
存储检测结果，提供查询和聚合

应用场景-用户手工查询



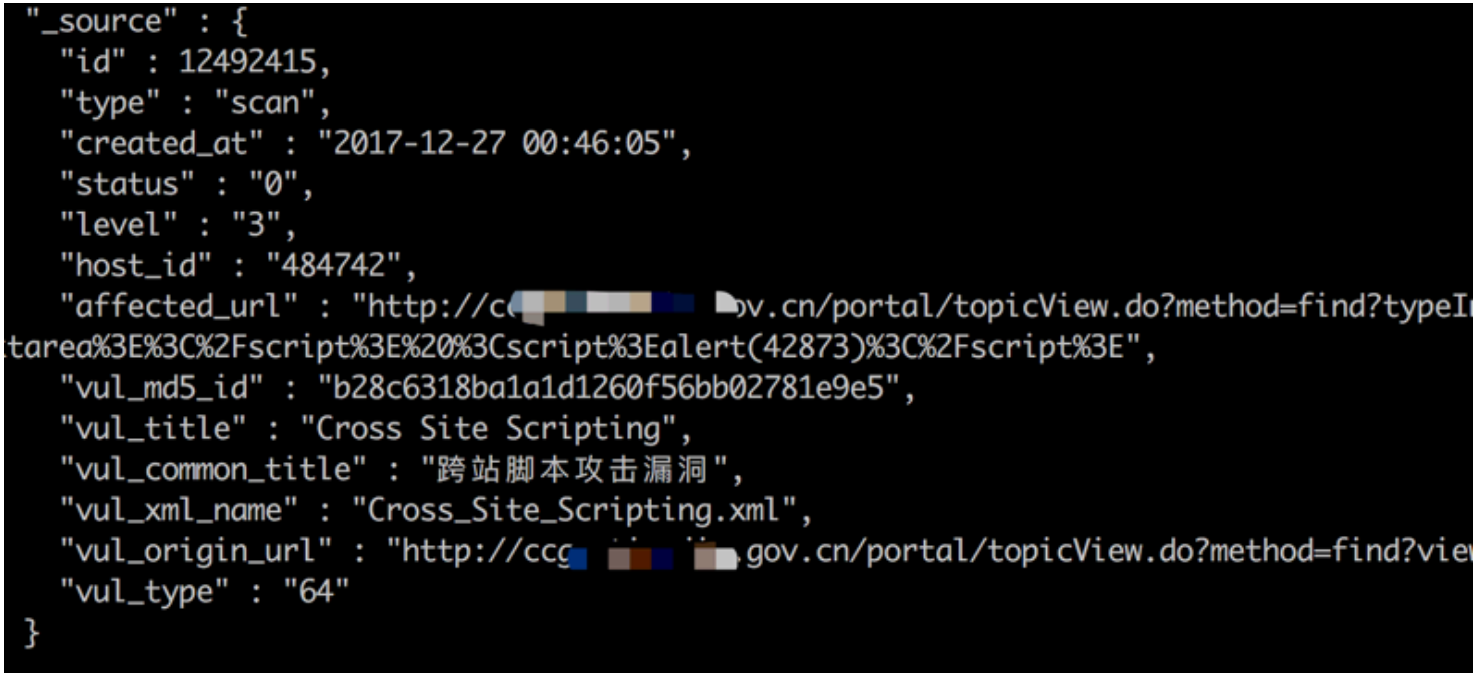
elastic

应用场景-网站云监测



elastic

结果样本



elastic

告警明细

需要密集查询最近结果

告警总数：5155　　去重后网站个数：95　　是否开启自动刷新

处理已选　确认已选　误报已选　处理全部　确认全部　误报全部

时间	基本信息	威胁等级	告警类型	告警信息	运维	操作
未读 2019-11-24 17:02:22	网站：http://w[redacted].com 网站名：none 账号：test-yunji[redacted].cn	中危	违规内容	告警状态：已确认 词语数量：1 敏感词语：狐媚 URL数量：1 相关资产页面：查看相关资产页面	误报	通知　处理　详情 调整策略
未读 2019-11-24 17:02:02	网站：http://ww[redacted]om 网站名：none 账号：test-yunji[redacted]0.cn	中危	违规内容	告警状态：已确认 词语数量：1 敏感词语：色情 URL数量：4 相关资产页面：查看相关资产页面	误报	通知　处理　详情 调整策略
2019-11-24 17:00:37	网站：http://1[redacted]om 网站名：可视化站47 账号：test-yunjian[redacted].cn	高危	可用性	告警状态：已处理 状态：已恢复 原因：总响应时间超过阈值 相关资产页面：查看相关资产页面		详情　调整策略

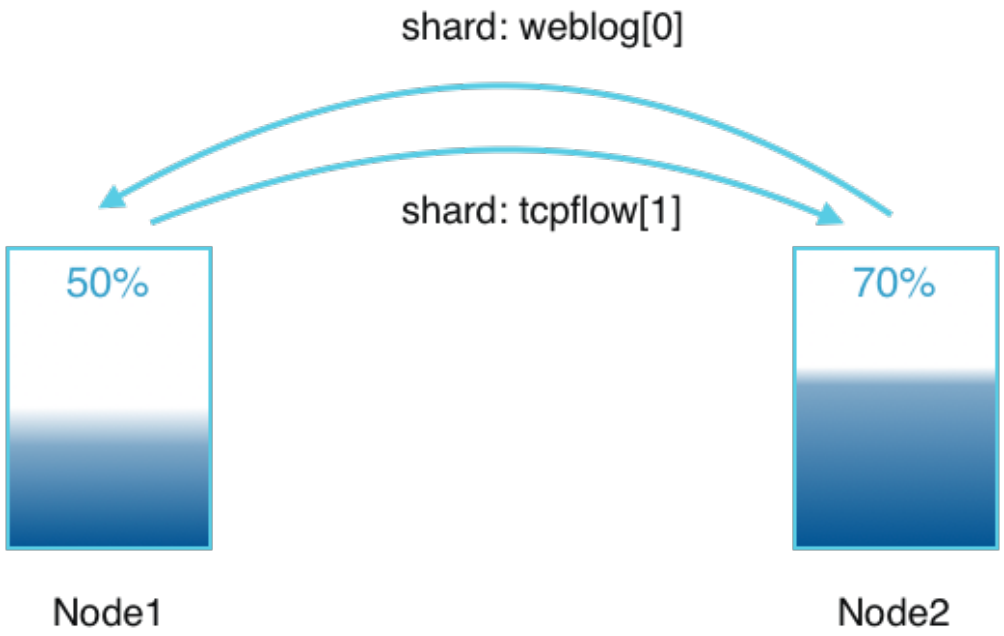
elastic

应用场景-态势感知

多维度安全事件监测、预测、回溯等。存储攻击情况，漏洞信息

内核改进

节点间的磁盘平衡



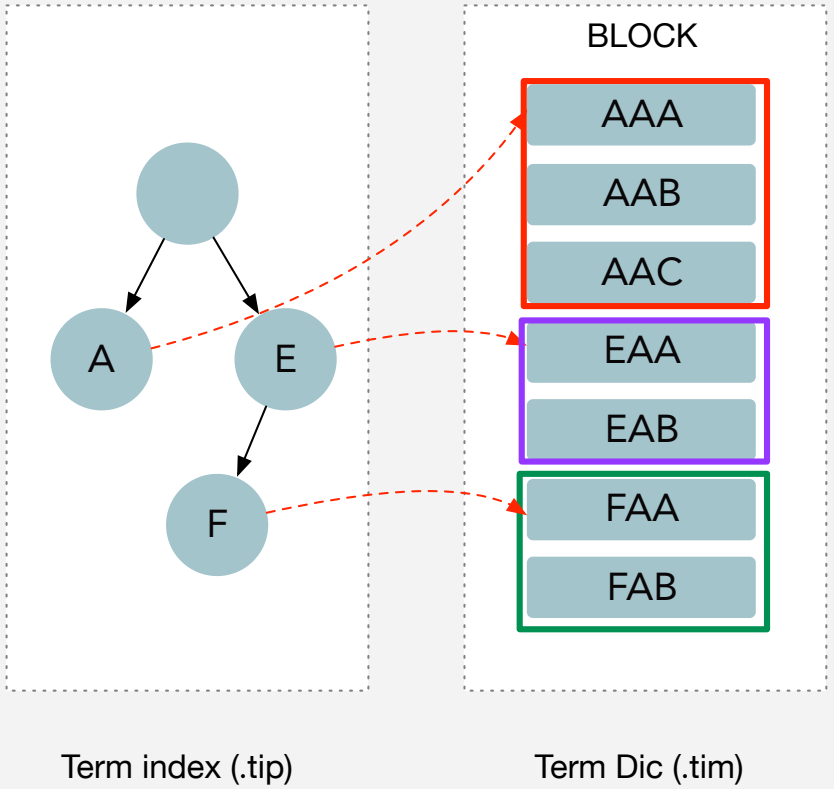
elastic

内存优化

减小 FST 体积

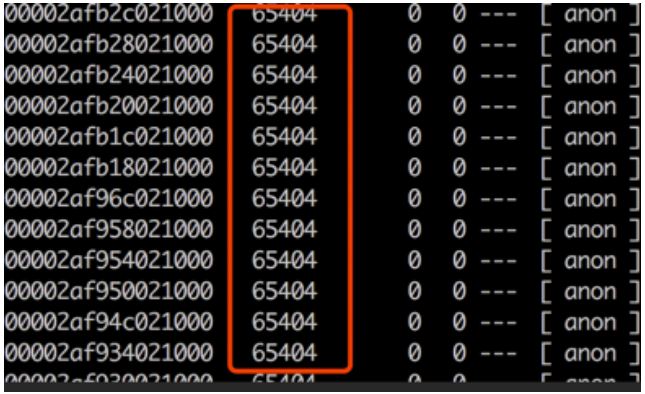
调整 block size

默认值：25-48
修改为：100-198
内存占用降低20%+

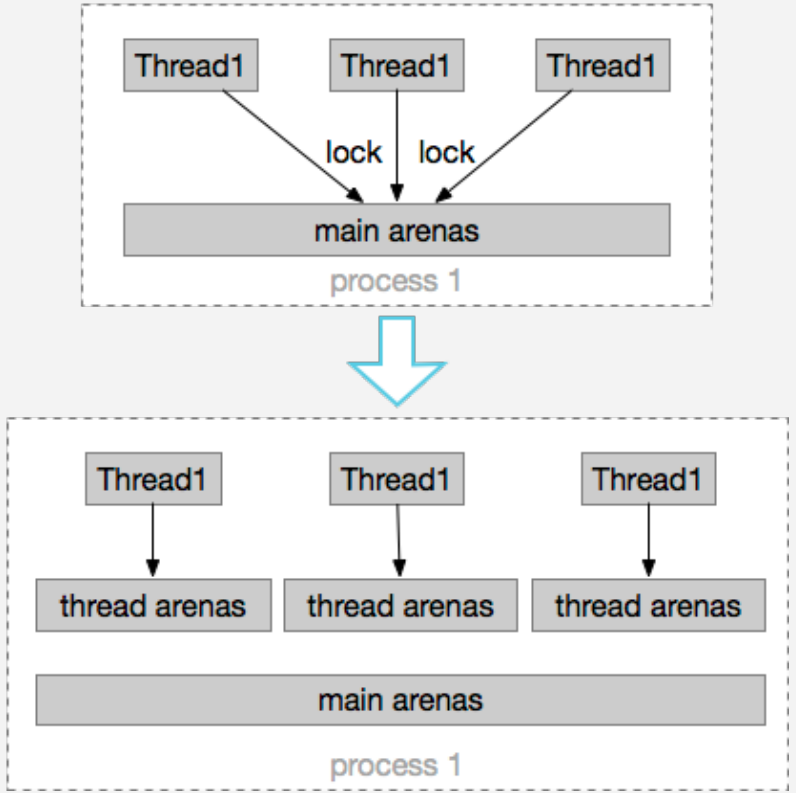


内存优化

降低堆外内存使用量



glibc >= 2.10
64位系统: cpu cores*8*64M ~ 12GB
export MALLOC_ARENA_MAX=1



其他小改进

节点内磁盘均衡

- 以剩余可用空间作为权重，加权轮询决定分片放置到哪个磁盘

分批查询

- search.shard_count.limit
- max_concurrent_shard_requests

提升主节点效率

- 并发写索引和分片 state
- 达到 recovery并发上限时跳过分片分配

生产环境建议

生产环境建议

使用方式要合理

- 尽量分离角色部署
- 尽量避免使用通配查询
- 尽量避免业务直接读写 ES
- scroll 并发别太大
- ……

调整默认配置

- node_left.delayed_timeout
- routing.allocation.enable
- total_shards_per_node
- destructive_requires_name

版本选择

- 坚持持续升级
- 生产环境延迟升级

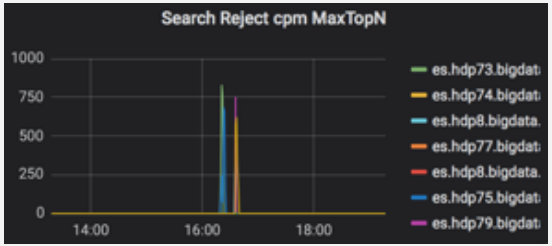
重点监控

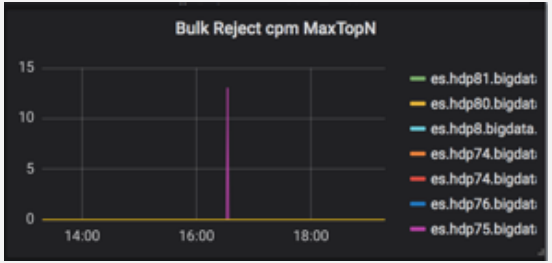
Request QPS
indexing/search

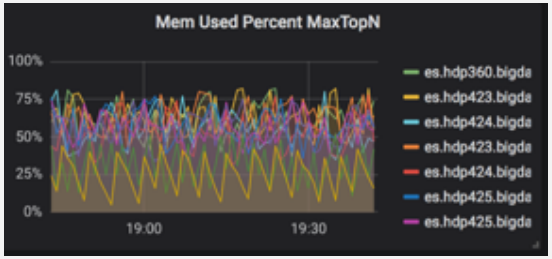
ThreadPool
active/queue/reject

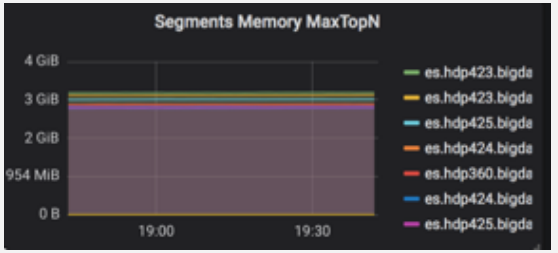
SlowLog
indexing/query/fetch

Memory
used percent/segments memory









Thank you!

elastic