SQLAlert:
基于 ES 的 SQL 报警引擎

南京云利来软件科技有限公司

张立丹

目录

概述

- **使用 ES？**

  强聚合能力（Metrics、Buckets）、低延迟

- **使用 SQL？**

  较高抽象层次（相对于DSL）、“人”尽皆知

- **使用 RDL（Rule Description Language）？**

  “规则描述语言”，提供辅助计算功能

目录

SQL 及扩展：示例

```
SELECT
    sum(inbyte + outbyte) AS bytes,
    count(dip)            AS count
FROM
    'tcp-*'
WHERE
    last(5m) AND
    ip_range(sip, ['192.168.0.0', '192.168.0.255'], '192.168.0.0/24')
GROUP BY dip, sip
LIMIT 10, 10
ORDER BY bytes
```

SQL 及扩展：过滤

- **过滤表达式**

  inbyte + outbyte > 10000 **AND** len(domain) > 64 **AND** sip = '192.168.0.55'

- **过滤函数**

  last(5m)、**last_days**(3, 5m)、**last_weeks**(4, 5m)、**last_workdays**(3, 10m)

  range(flows, [100,1000])、**ip_range**(sip,'192.168.0.0/24')、**date_range**(...)

  query_string('sip:[192.168.0.100 **TO** 192.168.0.200]')

SQL 及扩展：聚合

- **聚合函数**

  count(*)、**count**(sip)、**count**(UNIQUE url)、**count**(inbyte + outbyte)

  sum()、**min**()、**max**()、**avg**()

  stdev()、**squares**()、**variance**()

SQL 及扩展：桶

- **桶字段及桶函数**

  sip、dip

  range()、ip_range()、 date_range()

  histogram()、 date_histogram()

  filters()

- **示例：根据 dip 区分“东西走向”与“南北走向”**

```
filters(
        ip_range(dip, '192.168.0.0/24') AS lan,
    NOT ip_range(dip, '192.168.0.0/24') AS wan
) AS type
```

目录

RDL 脚本：示例

```
# This is test script of RDL.
# Author: ZHANG Li Dan.

__es_host__ = "192.168.0.101:9299";
__es_index_alert__ = { "index": "alert-%Y-%M", "type": "alert" };
alert_mail = false;

# 100 MBytes.
BYTES_THRESHOLD = 100 * 1024 * 1024;

sql = "SELECT inbyte + outbyte AS bytes FROM ... ";
result = query_filter(sql, "bytes > BYTES_THRESHOLD");
if alert_mail {
    alert_mail(result);
} else {
    alert_es(result);
}

```

RDL 脚本：语法

- **数据类型**

  int、float、string、 list、 dict、

- **操作符**

  =、+、-、*、\、%、<、<=、>、>=、

  ==、!=、 &&、||、!、… ? ... : ...

- **条件语句**

  if <expr> { ... } else if <expr> { ... } else { ... }

- **循环语句**

  for (cc=0; cc<100; cc++) { ... }、for key, value in object { ... }

  break、continue、return

- **函数**

  do_something(arg1, arg2, arg3);

  def my_func (arg1, arg2) { <statement>; <statement>; ... }

RDL 脚本：功能

- **内建函数**

  print(), **len**(), **type**(), **keys**(), **values**(), **append**(), **is_empty**(), ...

  query(), **query_filter**(), **alert**(), **alert_mail**(), **alert_es**(), ...

  max(), **min**(), **avg**(), **median**(), **agg_max**(), **agg_min**(), ...

  fmt_bytes(), **fmt_time**(), **fmt_percentage**(), ...

  exit(), **check_datetime**(), ...

- **自定义函数**

  def **max3**(x, y, z) { return x > y : max(x, z) : max(y, z); }

目录

规则：调度

- **SQLAlert 本地调度**

  配置调度项

```
"test":  {
    "enable": true,
    "runOnStart": true,
    "interval": 300,
    "file": [
        "alert/test.rule",
        "alert/test2.rule"
    ]
}
```

- **SQLAlert 作为代理**

  接收 HTTP POST 请求

规则：示例一

- **固定阀值报警**

时间范围：过去 5 分钟内；

报警条件：总字节数超过 200M 或者总包数超过 10000 个；

报警输出：总字节数、总包数、SIP、DIP；

报警方式：写回 ES。

规则：示例一（实现）

```
__es_host__ = "192.168.0.101:9299";
__es_index_alert__ = { "index": "alert-%Y-%M", "type": "alert" };

K_BYTES   = 200 * 1024 * 1024;
K_PACKETS = 10000;

sql = "
    SELECT
        sum(inbyte + outbyte)     AS bytes,
        sum(inpacket + outpacket) AS packets
    FROM 'tcp-*'
    WHERE last(5m)
    GROUP BY sip, dip
    LIMIT 20, 20
";

result = query_filter(sql, 'bytes > K_BYTES || packets > K_PACKETS');
alert_es(result);
```

规则：示例二

- **历史数据作为报警阀值**

时间范围：过去 5 分钟内；

参考时间：昨天当前时间段；

报警条件：总字节数超过 200M 或者总包数超过 10000 个，且超过历史数据的 50%；

报警输出：总字节数、总包数、SIP；

报警方式：写回 ES。

规则：示例二（实现）

```
__es_host__ = "192.168.0.101:9299";
__es_index_alert__ = { "index": "alert-%Y-%M", "type": "alert" };

K_BYTES   = 200 * 1024 * 1024;
K_PACKETS = 10000;
K_SCALE   = 1.5;

result1 = query_filter("
    SELECT
        sum(inbyte + outbyte)     AS bytes,
        sum(inpacket + outpacket) AS packets
    FROM 'tcp-*'
    WHERE last(5m)
    GROUP BY sip
    LIMIT 20
", 'bytes > K_BYTES || packets > K_PACKETS');

sip_list = item_values(result1, 'sip');
```

```
result2 = query("
    SELECT
        sum(inbyte + outbyte)     AS bytes,
        sum(inpacket + outpacket) AS packets
    FROM 'tcp-*'
    WHERE last_days(1, 5m) AND sip IN $(sip_list)
    GROUP BY sip
");

alert_result = [];
for item in result1 {
    if check_result_item(item, result2) {
        alert_result = append(alert_result, item);
    }
}

alert_es(alert_result);
```

规则：示例三

- **变化率报警**

  时间范围：过去 5 分钟内；

  参考时间：过去四周内相同时刻的 30 分钟时间段（例如，当前为周一，则参考时间为过去四周每个周一的当前 30分钟）；

  参考数据：历史数据的 90% 的百分位数；

  报警输出：总字节数；

  报警条件：当前变化率超过参考数据的 2 倍时报警;

  报警方式：写回 ES。

规则：示例三（实现）

```
__es_host__ = "192.168.0.101:9299";
__es_index_alert__ = { "index": "alert-%Y-%M", "type": "alert" };

K_PERCENT = 0.9;
K_SCALE   = 2;

result1 = query("
    SELECT
        sum(inbyte + outbyte) AS bytes
    FROM 'tcp-*'
    WHERE last_weeks(4, 30m)
    GROUP BY
        date_histogram(5m) AS time_5m,
        date_histogram(1m) AS time_1m
");

rates = values(agg_stdev(result1, 'time_5m', 'bytes'));
k_bytes = percentile(rates, K_PERCENT) * K_SCALE;
```

```
result2 = query("
    SELECT
        sum(inbyte + outbyte) AS bytes
    FROM 'tcp-*'
    WHERE last(5m)
    GROUP BY
        date_histogram(1m) AS time
");

rate = stdev(item_values(result2, 'bytes'));
if rate > k_bytes {
    alert_es(result2);
}
```

目录

- 概述
- SQL 及扩展
- RDL 脚本
- 规则
- 总结

总结：SQLAlert

- 使用 SQL 查询 ES；
- RDL 提供辅助的计算；
- 查询多个索引（甚至多个ES）；
- 规则调度及报警输出；

谢谢!

- Q&A?