

事件总线

事件规则

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

事件规则

- 事件规则概述

- 事件模式

- 创建事件规则

- 管理事件规则

- 配置数据转换

事件规则

事件规则概述

最近更新时间：2024-01-22 20:52:28

每个事件集包含若干条事件规则，事件规则是 EventBridge 的核心能力，主要承担事件触发，事件筛选，事件提取等能力。每条规则可指定若干事件目标，当规则被命中时，事件会被推送至所指定的事件目标。事件规则的相关操作如下：

事件规则中的过滤功能由事件模式提供。了解更多信息，请参见 [事件模式](#)。

事件规则中的目标触发功能由事件目标提供。

事件规则支持查看、编辑、删除。了解更多信息，请参见 [管理事件规则](#)。

事件模式

最近更新时间：2024-01-22 20:52:28

事件模式是事件总线 EventBridge 用来过滤相关事件的模式定义。事件总线 EventBridge 通过事件模式过滤事件并将事件路由到事件目标，事件模式必须和匹配的事件具有相同的结构。本文介绍事件模式的常用类型。

注意事项

事件模式匹配的原则如下：

事件必须包含事件模式中列出的所有字段名，且事件模式里的字段名必须和事件中的字段名具有相同嵌套结构。

事件模式是逐个字符精确匹配的，需注意大小写，匹配过程中不会对字符串进行任何标准化的操作。

要匹配的值遵循 JSON 规则：用引号引起来的字符串、数字以及不带引号的关键字 true、false 和 null。

指定值及 OR 和 AND 模式

您可以指定某个字段的值进行匹配，对比值在 JSON 阵列中，以 [] 包围。[] 内值为 OR，KEY 匹配为 AND。

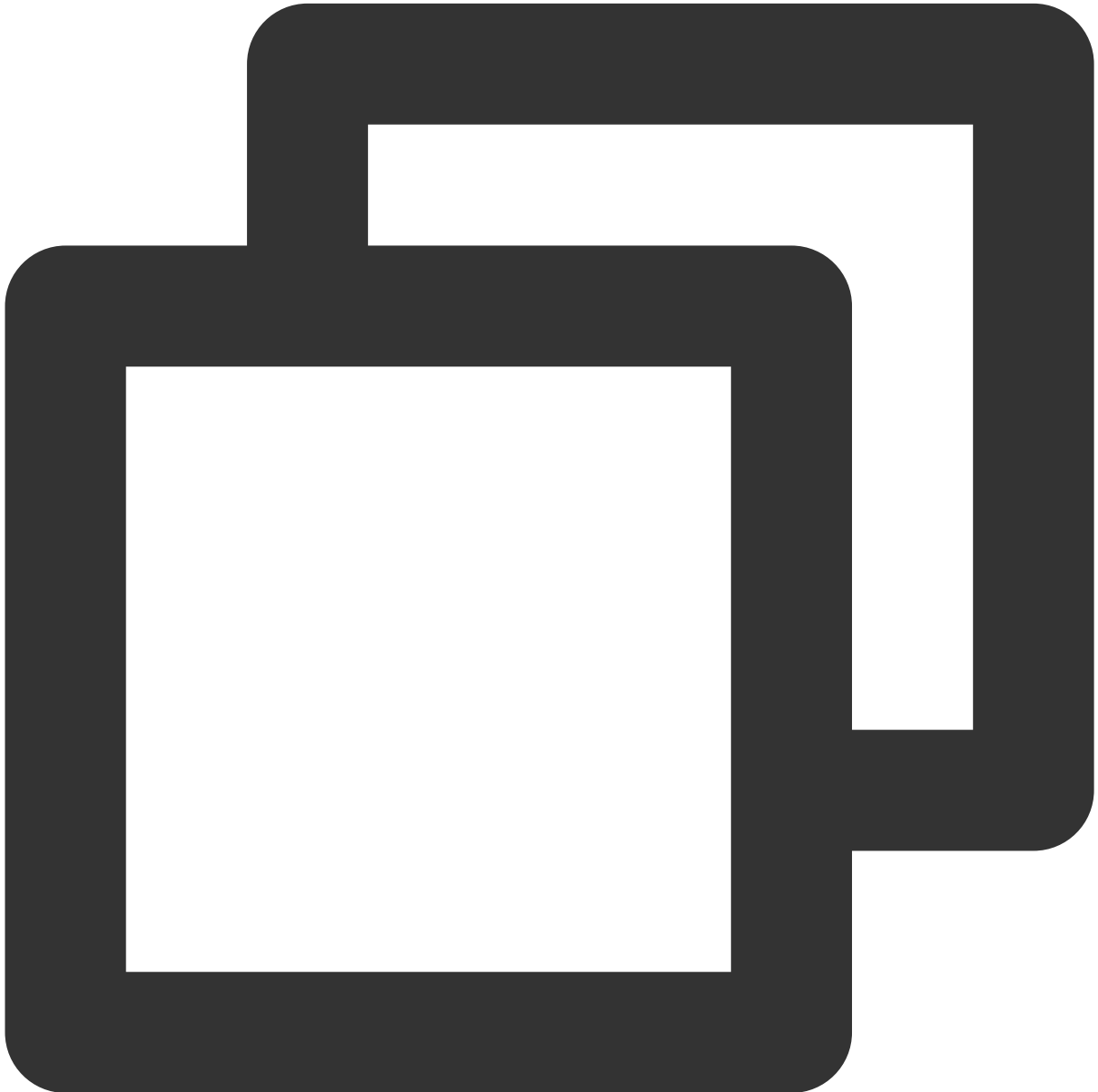
以 COS 数据为例，接收到的事件如下：



```
{
  "specversion": "1.0",
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
  "type": "cos:created:object",
  "source": "cos.cloud.tencent",
  "subject": "qcs::cos:ap-guangzhou:uid1250000000:bucketname",
  "time": "1615430559146",
  "region": "ap-guangzhou",
  "datacontenttype": "application/json;charset=utf-8",
  "resource": [
    "qcs::eb:ap-guangzhou:uid1250000000:eventbusid/eventruleid"
  ]
}
```

```
],  
  "data": {  
    "name": "testname",  
    "scope": 100  
  }  
}
```

对于如上事件，若指定 `data` 字段的 `name` 值进行指定值匹配，可以被正常触发的规则如下：

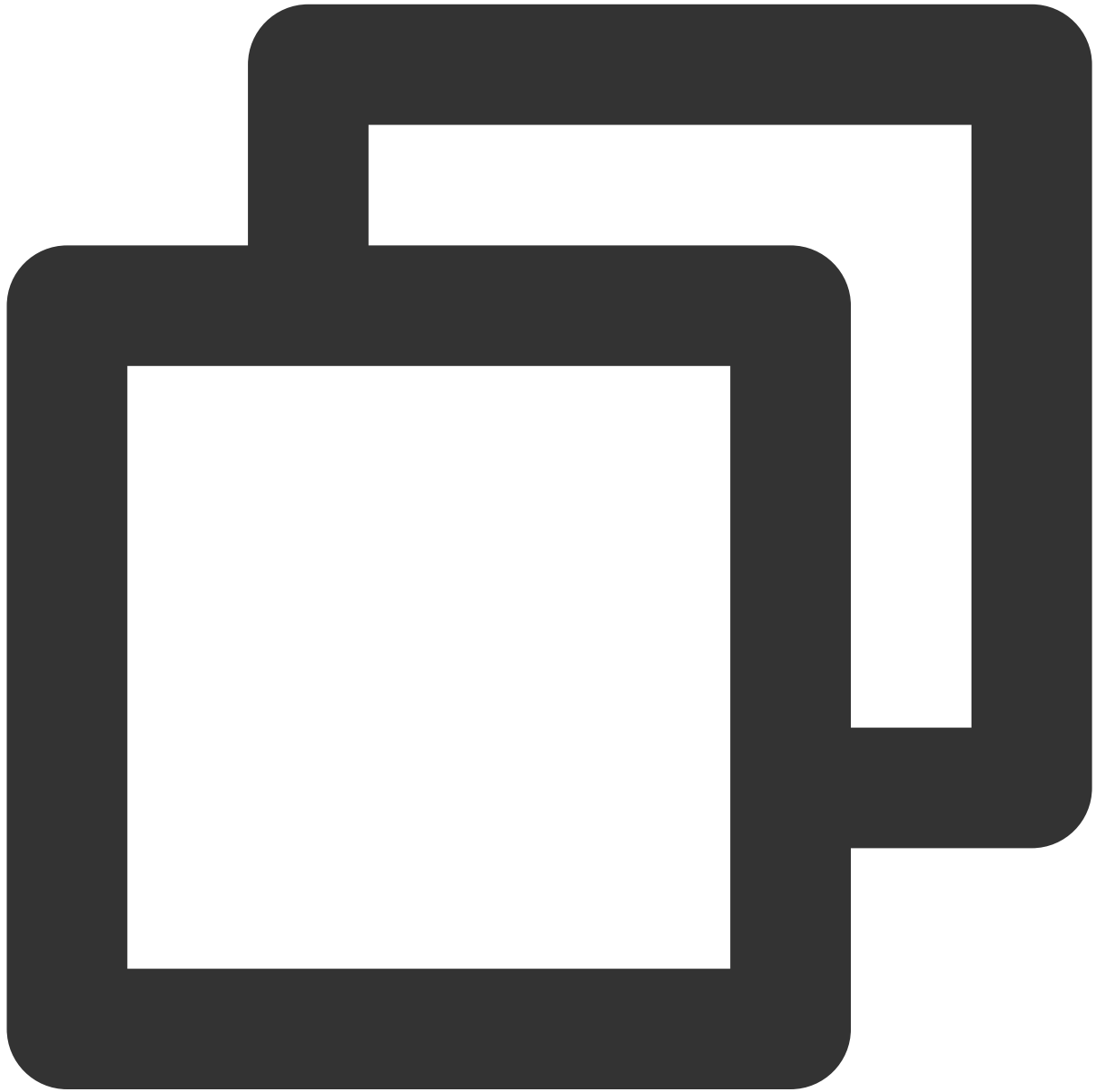


```
{  
  "data": {  
    "name": [  

```

```
        "testname"  
    ]  
}  
}
```

若指定 data 字段的 name 值进行 OR 匹配，可以被正常触发的规则如下：



```
{  
  "data": {  
    "name": [  
      "testname", "test"  
    ]  
  }  
}
```

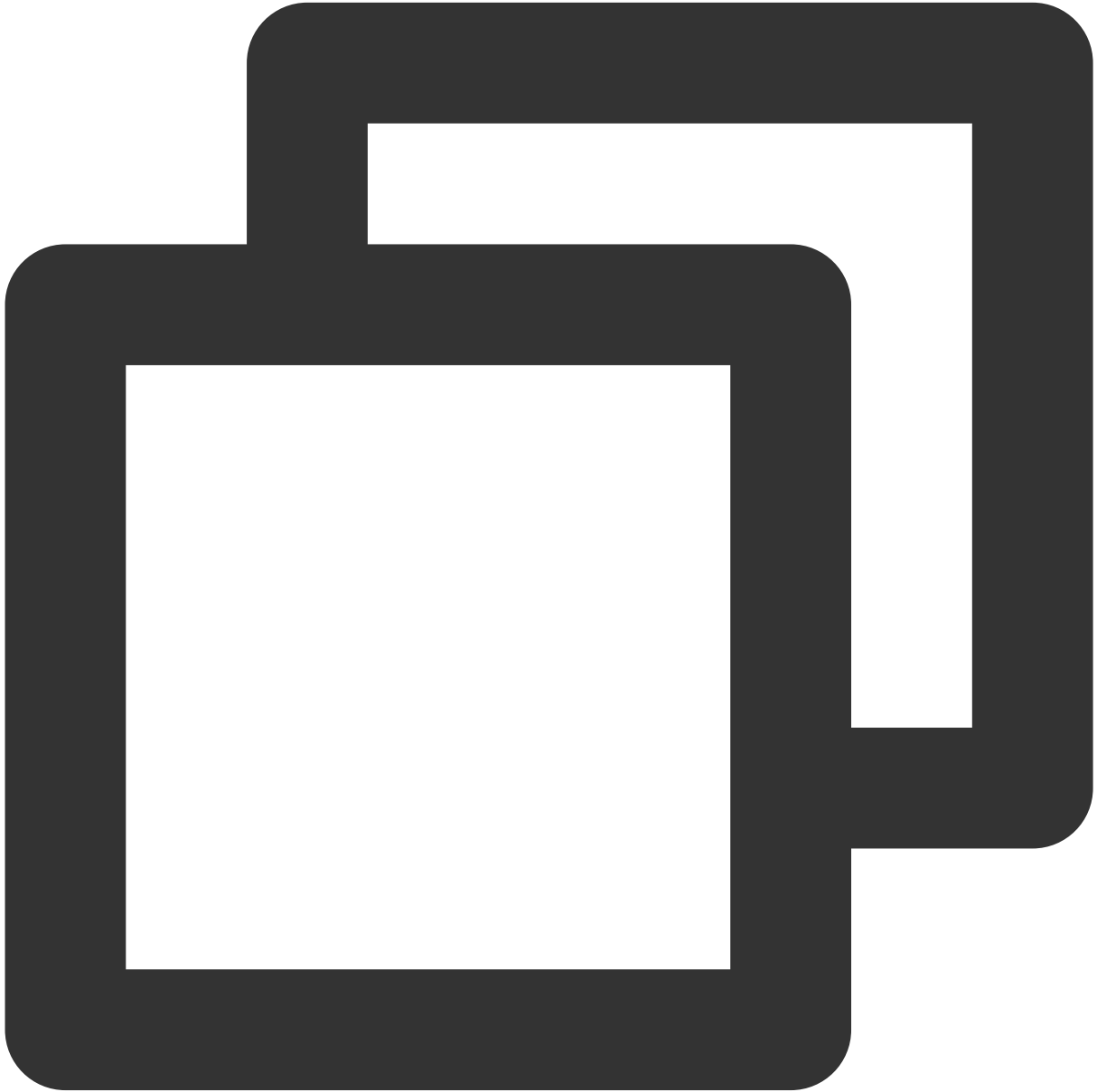


```
}  
}
```

前缀匹配

您可以对比事件来源中的前缀进行键值匹配，例如 { "prefix": "2021-10-02" }。

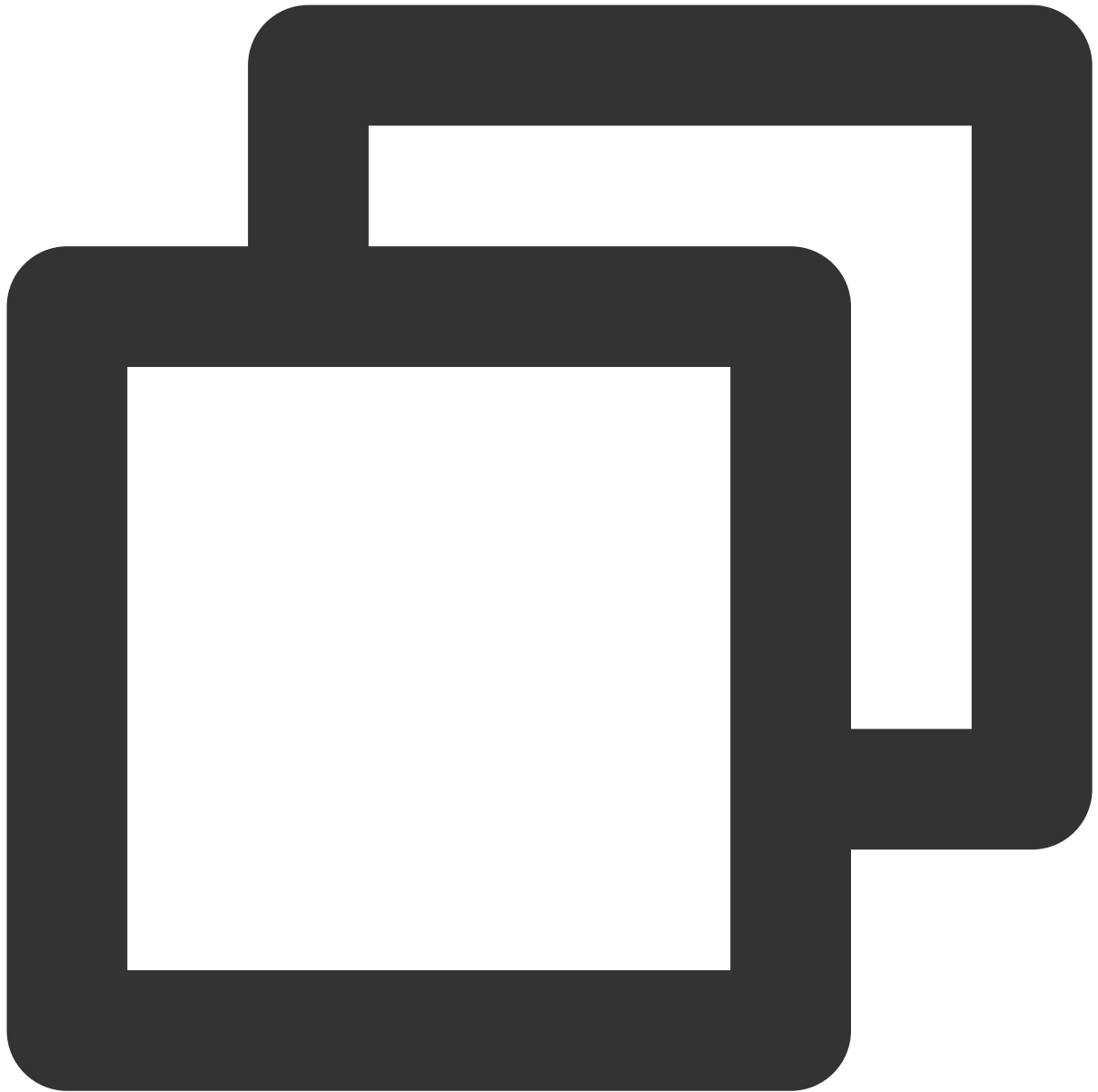
以 COS 数据为例，接收到的事件如下：



```
{  
  "specversion": "1.0",  
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
```

```
"type": "cos:created:object",
"source": "cos.cloud.tencent",
"subject": "qcs::cos:ap-guangzhou:uid1250000000:bucketname",
"time": "1615430559146",
"region": "ap-guangzhou",
"datacontenttype": "application/json;charset=utf-8",
"resource": [
  "qcs::eb:ap-guangzhou:uid1250000000:eventbusid/eventruleid"
],
"data": {
  "name": "testname",
  "scope": 100
}
}
```

指定 `data` 字段的 `name` 的前缀匹配值，可以被正常触发的规则如下：

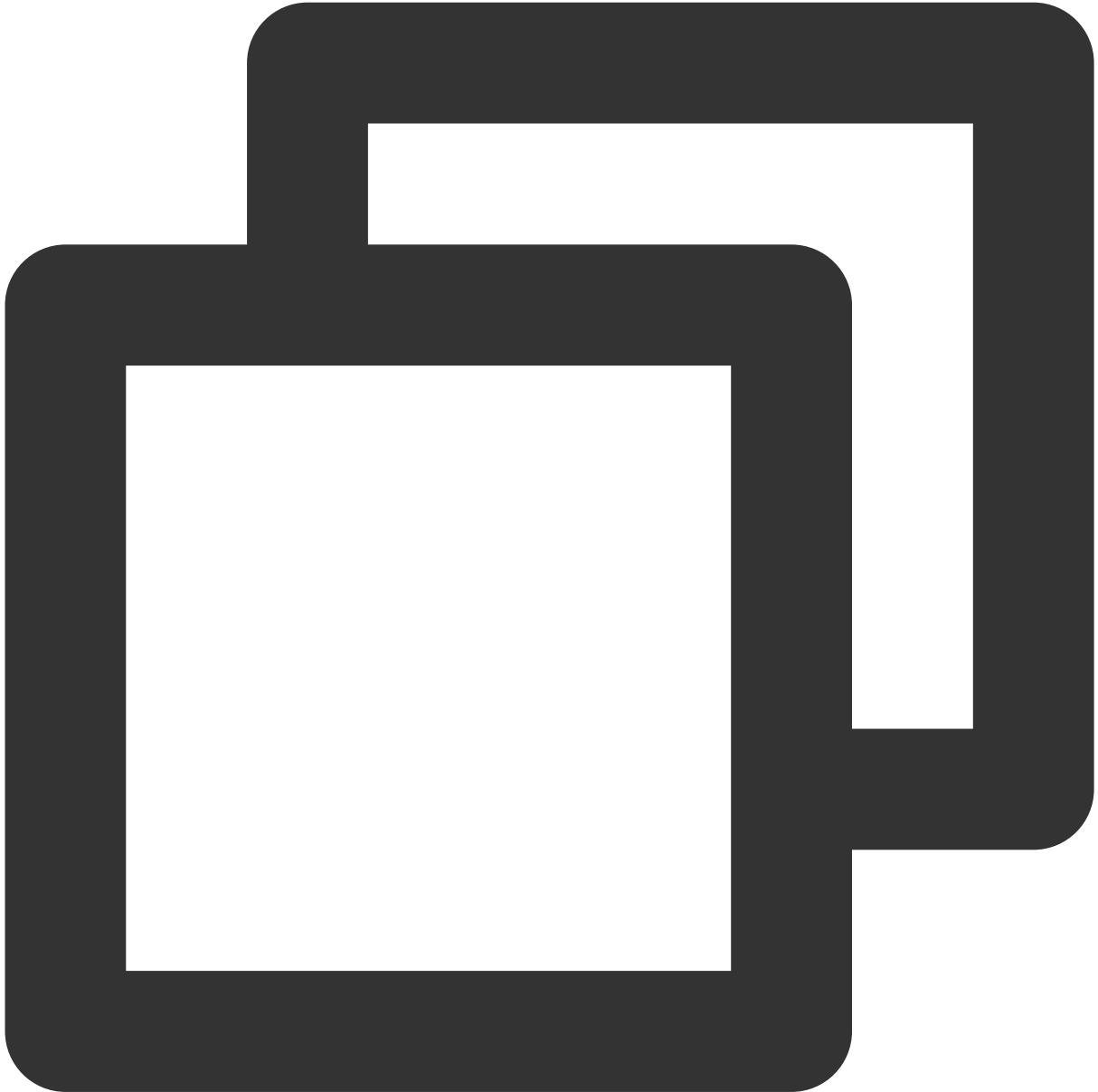


```
{
  "data": {
    "name": [
      {
        "prefix": "te"
      }
    ]
  }
}
```

后缀匹配

您可以对比事件来源中的后缀进行键值匹配，例如 { "suffix": ".txt" }。

以 TDMQ 数据为例，接收到的事件如下：



```
{
  "specversion": "1.0",
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
  "type": "connector:tdmq",
  "source": "tdmq.cloud.tencent",
  "subject": "qcs::tdmq:$region:$account:topicName/$topicSets.clusterId/$topicSet",
  "time": "1615430559146",
```

```
"region": "ap-guangzhou",
"datacontenttype": "application/json;charset=utf-8",
"data": {
    "topic": "persistent://appid/namespace/topic-1",
    "tags": "testtopic",
    "TopicType": "0",
    "subscriptionName": "xxxxxx",
    "toTimestamp": "1603352765001",
    "partitions": "0",
    "msgId": "123345346",
    "msgBody": "Hello from TDMQ!"
}
```

指定 `data` 字段的 `topic` 的后缀匹配值，可以被正常触发的规则如下：

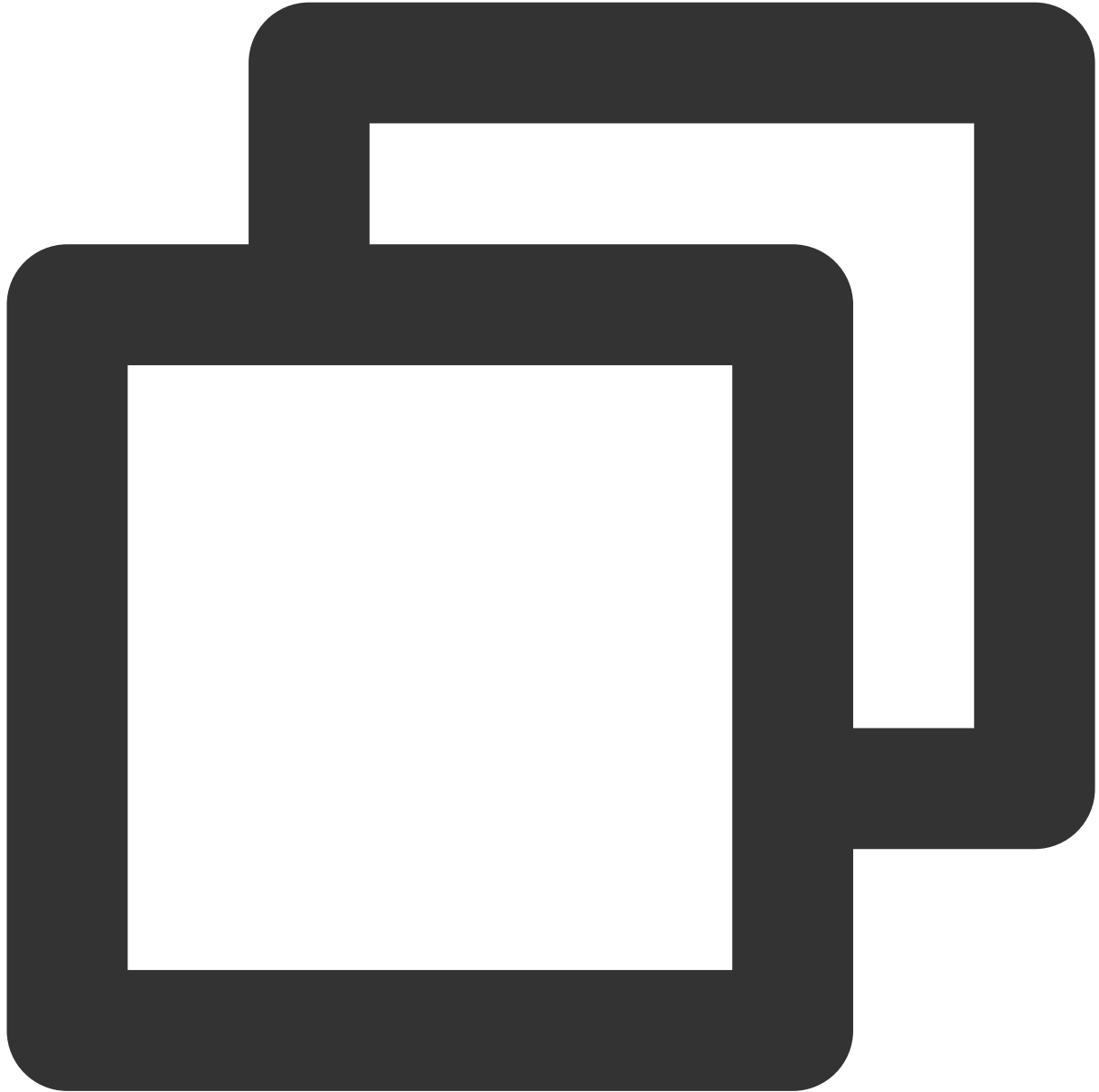


```
{
  "data": {
    "topic": [{
      "suffix": "/topic-1"
    }]
  }
}
```

除外匹配

您可以指定某个字段除了提供的值之外的任何值进行匹配，例如 { "anything-but": "initializing" }。

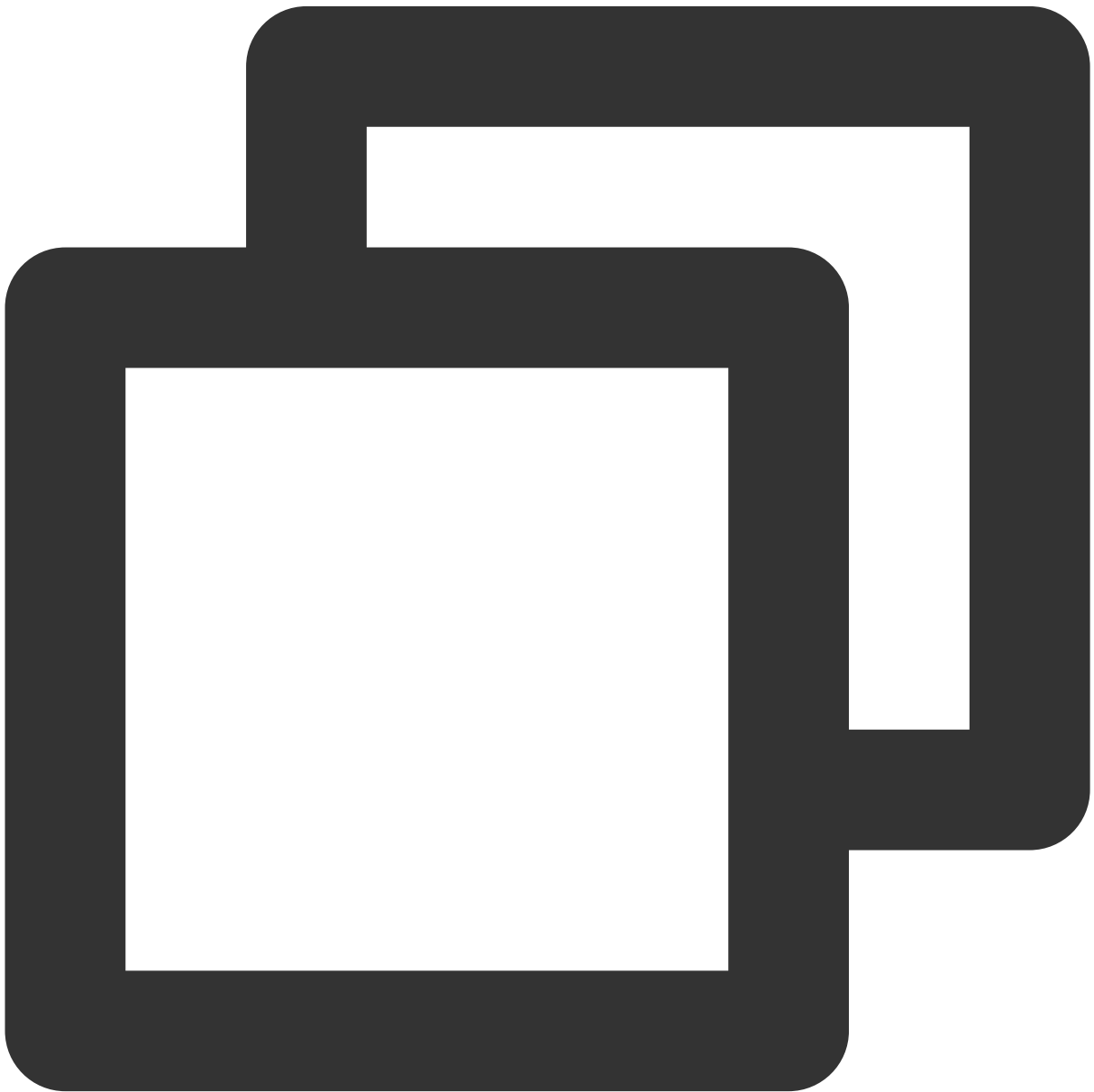
以 COS 数据为例，接收到的事件如下：



```
{
  "specversion": "1.0",
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
  "type": "cos:created:object",
  "source": "cos.cloud.tencent",
  "subject": "qcs::cos:ap-guangzhou:uid1250000000:bucketname",
  "time": "1615430559146",
  "region": "ap-guangzhou",
```

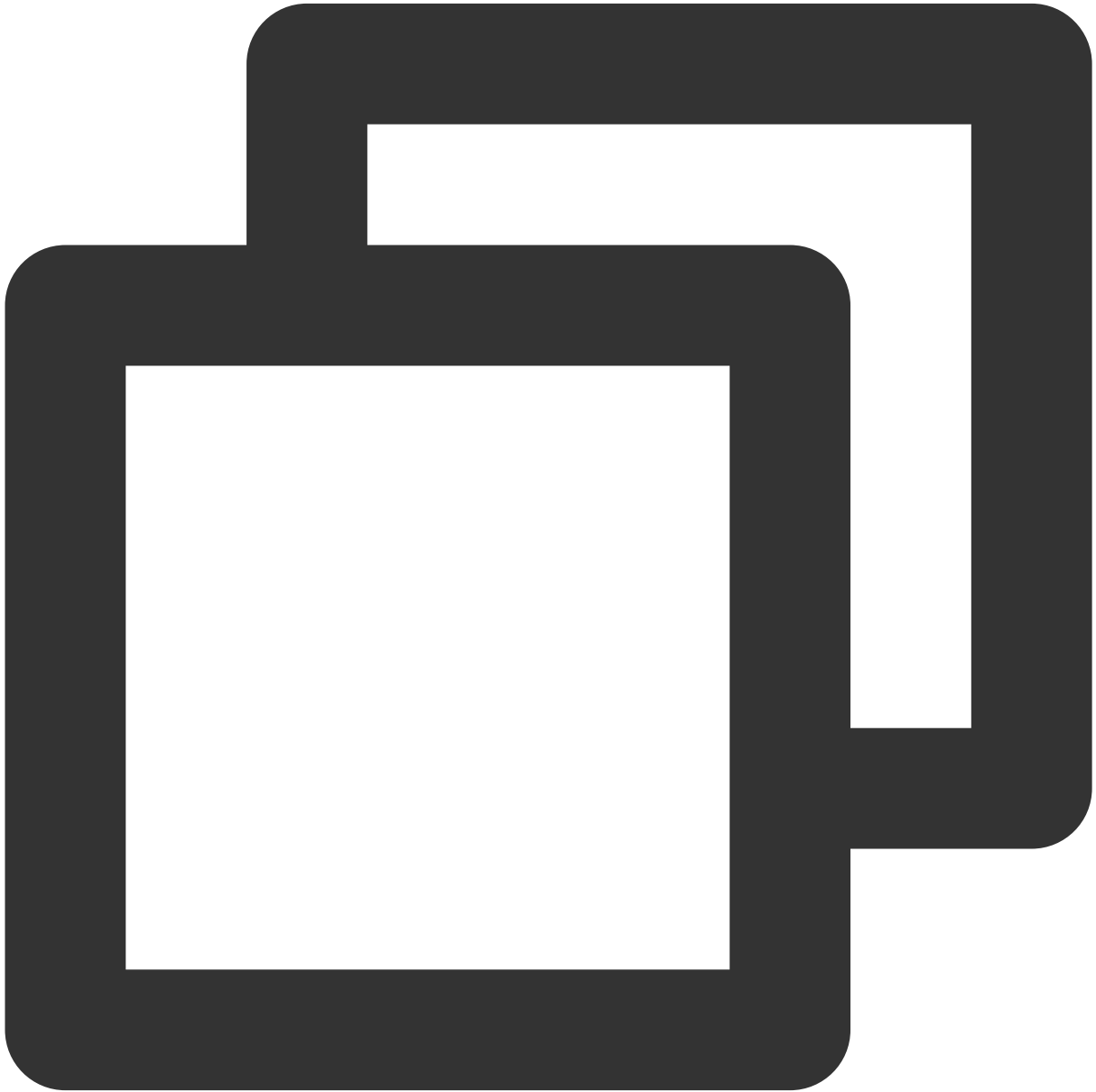
```
"datacontenttype": "application/json;charset=utf-8",  
"resource": [  
  "qcs::eb:ap-guangzhou:uid1250000000:eventbusid/eventruleid"  
],  
"data": {  
  "name": "testname",  
  "scope": 100  
}  
}
```

指定 data 字段的 name 的除外匹配值，可以被正常触发的规则如下：




```
{
  "data": {
    "name": [{
      "anything-but": "test1"
    }]
  }
}
```

指定 `data` 字段的 `name` 的除外匹配值，不可以被正常触发的规则如下：



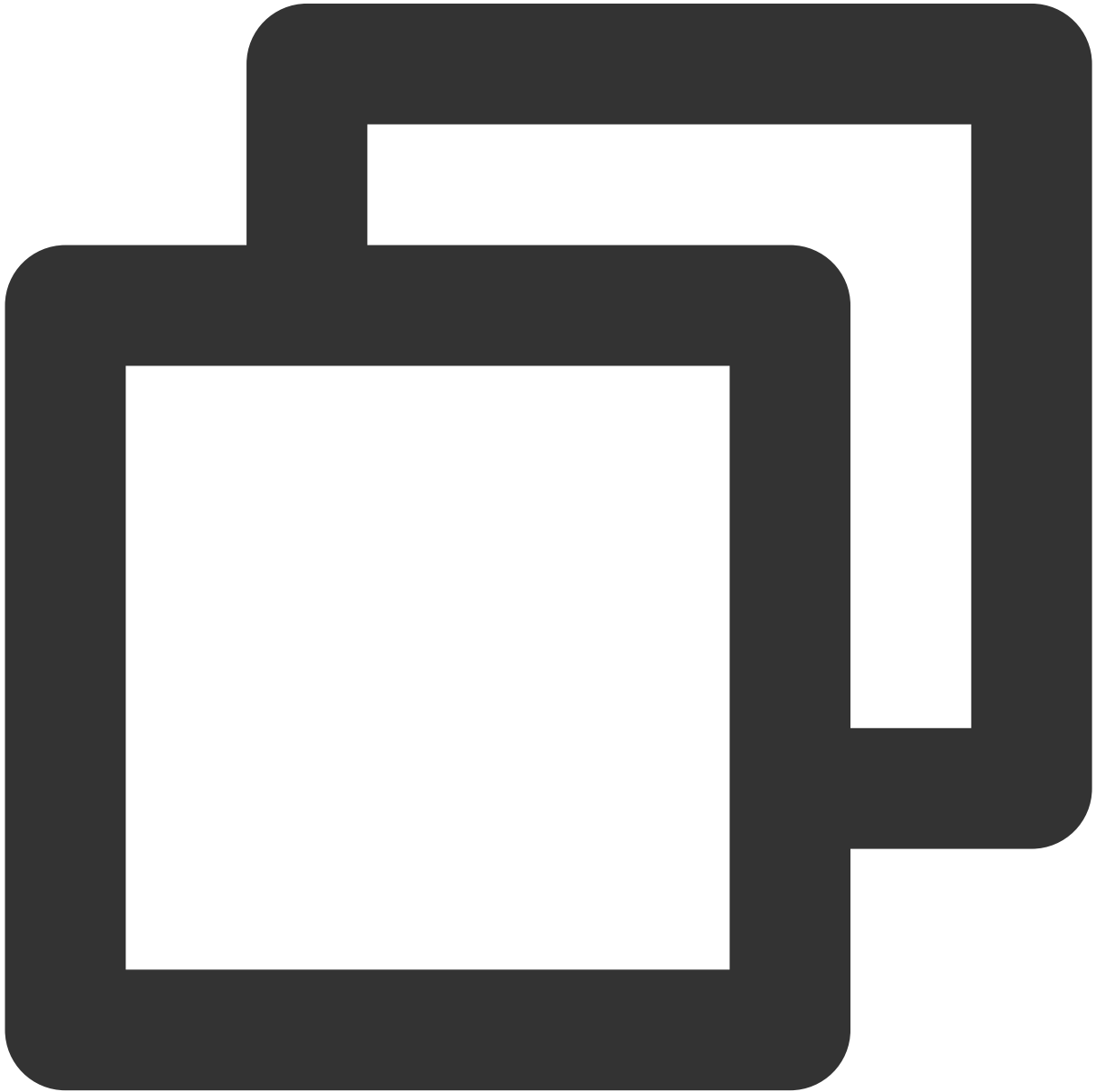
```
{
```

```
"data": {  
  "name": [{  
    "anything-but": "testname"  
  }]  
}
```

包含匹配

您可以指定 data 中存在的某个字段进行匹配，例如 { "contain": ".txt" }。

以 TDMQ 数据为例，接收到的事件如下：



```
{
  "specversion": "1.0",
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
  "type": "connector:tdmq",
  "source": "tdmq.cloud.tencent",
  "subject": "qcs::tdmq:$region:$account:topicName/$topicSets.clusterId/$topicSet",
  "time": "1615430559146",
  "region": "ap-guangzhou",
  "datacontenttype": "application/json;charset=utf-8",
  "data": {
    "topic": "persistent://appid/namespace/topic-1",
    "tags": "testtopic",
    "TopicType": "0",
    "subscriptionName": "xxxxxx",
    "toTimestamp": "1603352765001",
    "partitions": "0",
    "msgId": "123345346",
    "msgBody": "Hello from TDMQ!"
  }
}
```

指定 data 字段的 topic 的包含匹配值，可以被正常触发的规则如下：



```
{
  "data": {
    "topic": [{
      "contain": "topic-1"
    }]
  }
}
```

指定 `data` 字段的 `topic` 同时包含多个匹配值，可以被正常触发的规则如下：

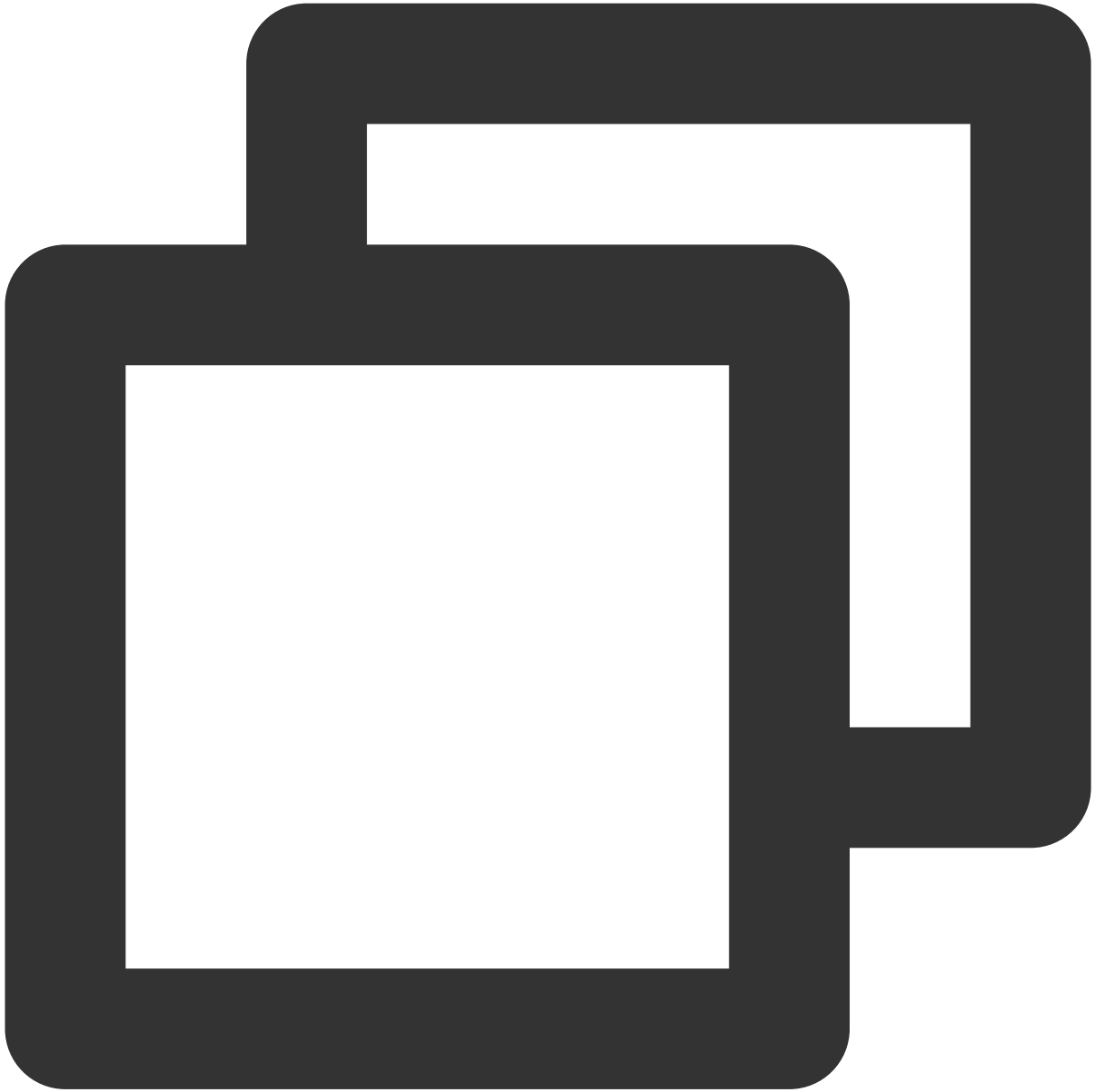


```
{
  "data": {
    "topic": [{
      "contain": ["topic-1", "appid"]
    }]
  }
}
```

数组匹配

您可以通过语法过滤数组类型的字段，例如 `{"array": "{\\"key1\\":\\"value1\\"}"}`。

典型场景如要根据产品属性 `data` 结构中的某个字段生成事件规则，以数据订阅 DTS 数据为例，接收到的事件如下：



```
{
  "id": "13a3f42d-7258-4ada-da6d-023a33*****",
  "type": "dts:mysql:update",
  "specversion": "1.0",
  "source": "dts.cloud.tencent",
  "subject": "cdb-xxx",
  "time": 1660013278609,
```

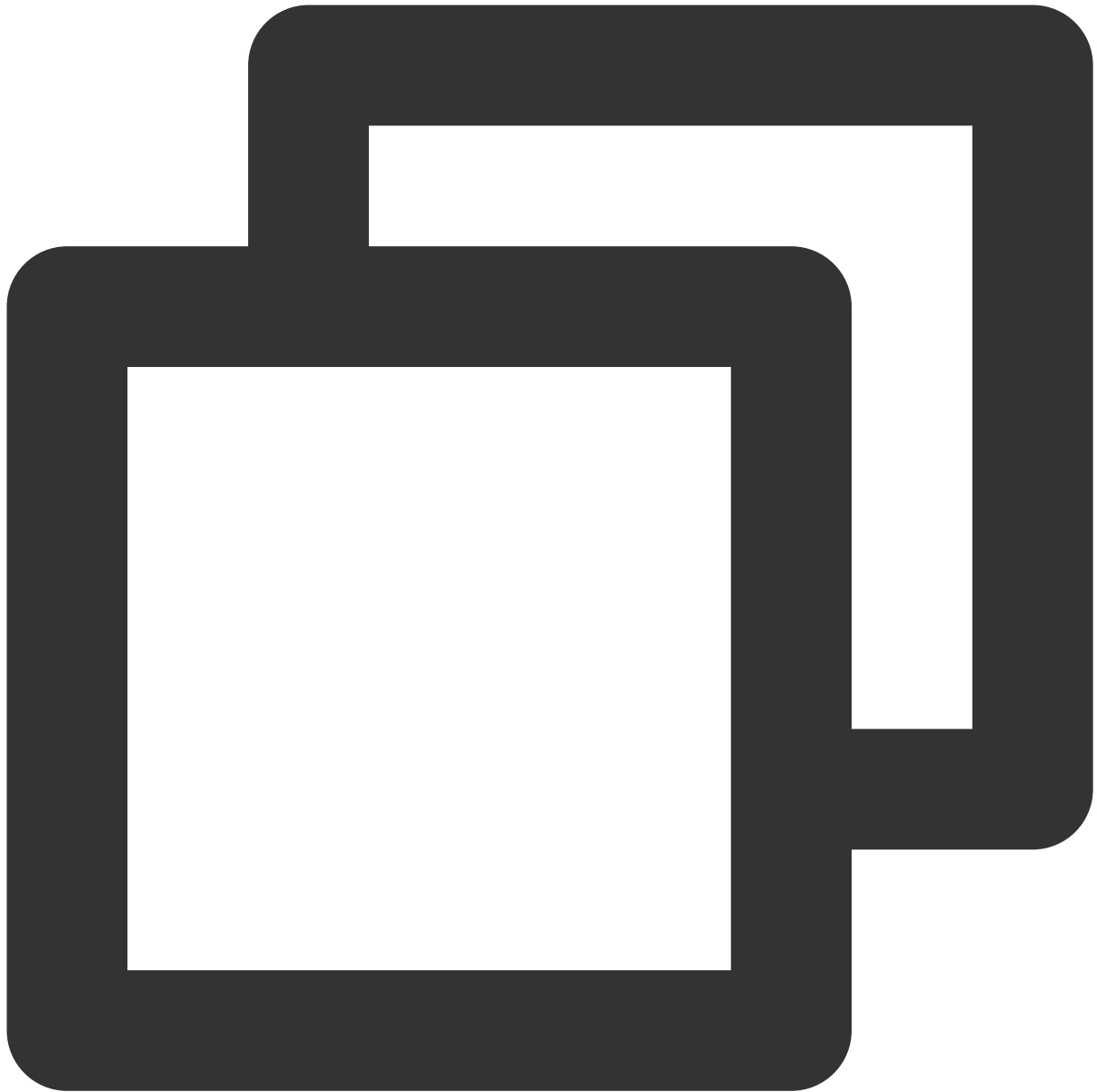
```

"region": "ap-guangzhou",
"dataContentType": "application/json;charset=utf-8",
"tags": {
  "key1": "value1",
  "key2": "value2"
},
"data": {
  "topic": "topic-sub-xxx-cdb-xxx",
  "partition": 0,
  "offset": 72235,
  "partition_seq": 72236,
  "event": {
    "dmlEvent": {
      "dmlEventType": 1,
      "columns": [
        {
          "name": "time",
          "originalType": "time"
        },
        {
          "name": "id",
          "originalType": "int(11)",
          "isKey": true
        }
      ],
      "rows": [
        {
          "oldColumns": [
            {
              "dataType": 13,
              "charset": "utf8",
              "bv": "c3NzYWZhcnVWZmMTEe"
            }
          ],
          "newColumns": [
            {
              "dataType": 13,
              "charset": "utf8",
              "bv": "MjA6MTI6MjI="
            }
          ]
        }
      ]
    }
  },
  "header": {
    "sourceType": 1,

```

```
"messageType": 2,
"timestamp": 1648555949,
"serverId": 109741,
"fileName": "mysql-bin.000005",
"position": 11172920,
"gtid": "38cecd93-a9c2-11ec-b952-043f72d8da53:55",
"schemaName": "dts",
"tableName": "dts_mysql",
"seqId": 72286,
"isLast": true
},
"eb_consumer_time": "2022-03-29T20:12:29+08:00",
"eb_connector": "cdb-xxx"
}
}
```

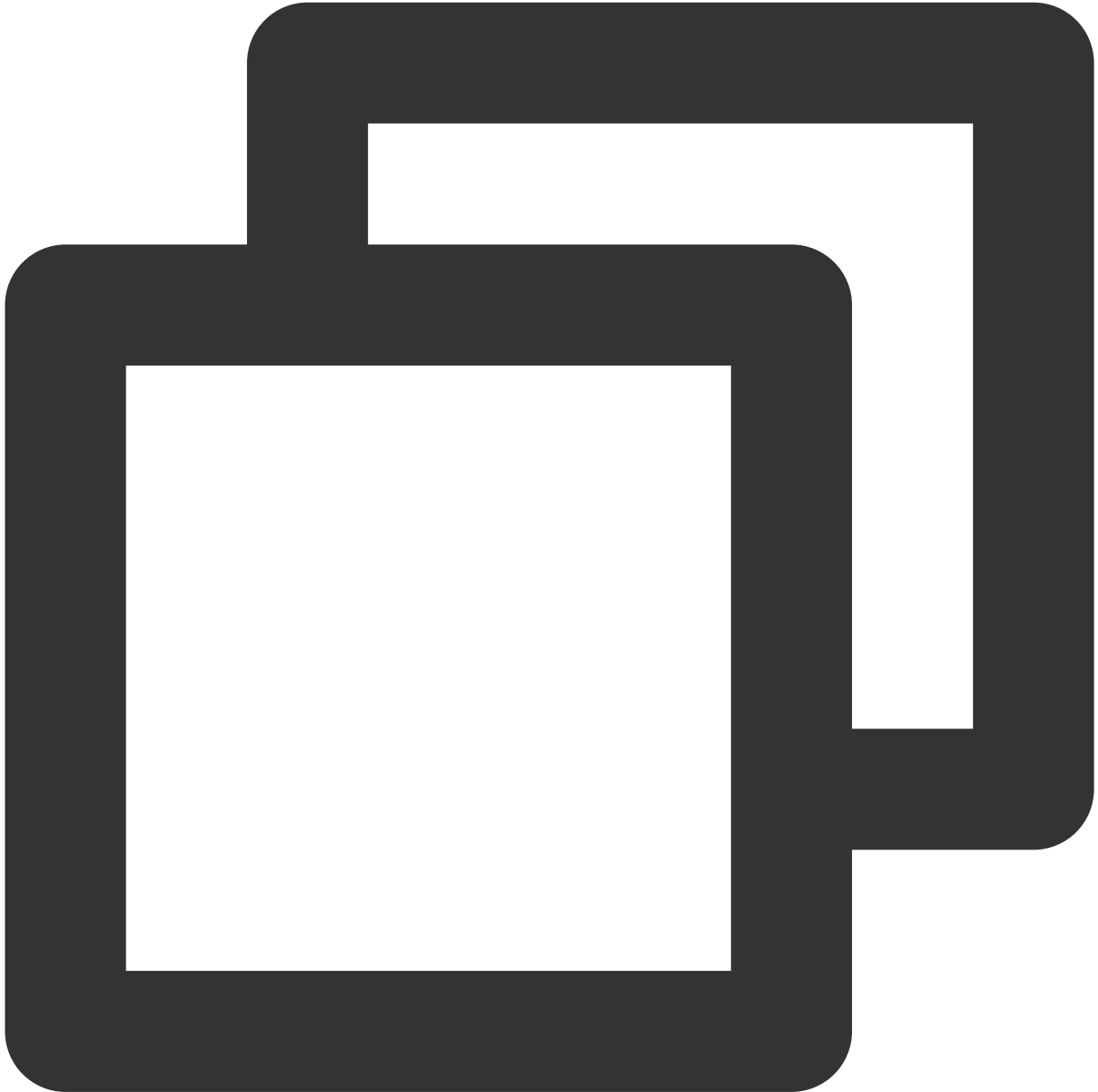
对于如上事件，若需要通过 `columns` 字段进行规则匹配，可以被正常触发的规则如下：



```
{
  "source": "dts.cloud.tencent",
  "type": "dts:mysql:update",
  "data": {
    "event": {
      "dmlEvent": {
        "columns": [{
          "array": "{\"name\":\"time\"}"
        }]
      }
    }
  }
}
```

```
}  
}
```

一个字段的多个数据过滤规则时，多条数据之间是“与”的关系：



```
{  
  "source": "dts.cloud.tencent",  
  "type": "dts:mysql:update",  
  "data": {  
    "event": {  
      "dmlEvent": {  
        "columns": [{
```

```
        "array": [{"name": "id", "originalType": "int(11)"}]
    }
  }
}
```

IP 地址匹配

您可以指定 data 中字段的 IP 地址。例如，以下示例事件模式中只匹配 a 为 10.0.0.0/24 的事件：`{"cidr": "10.0.0.0/24"}`。

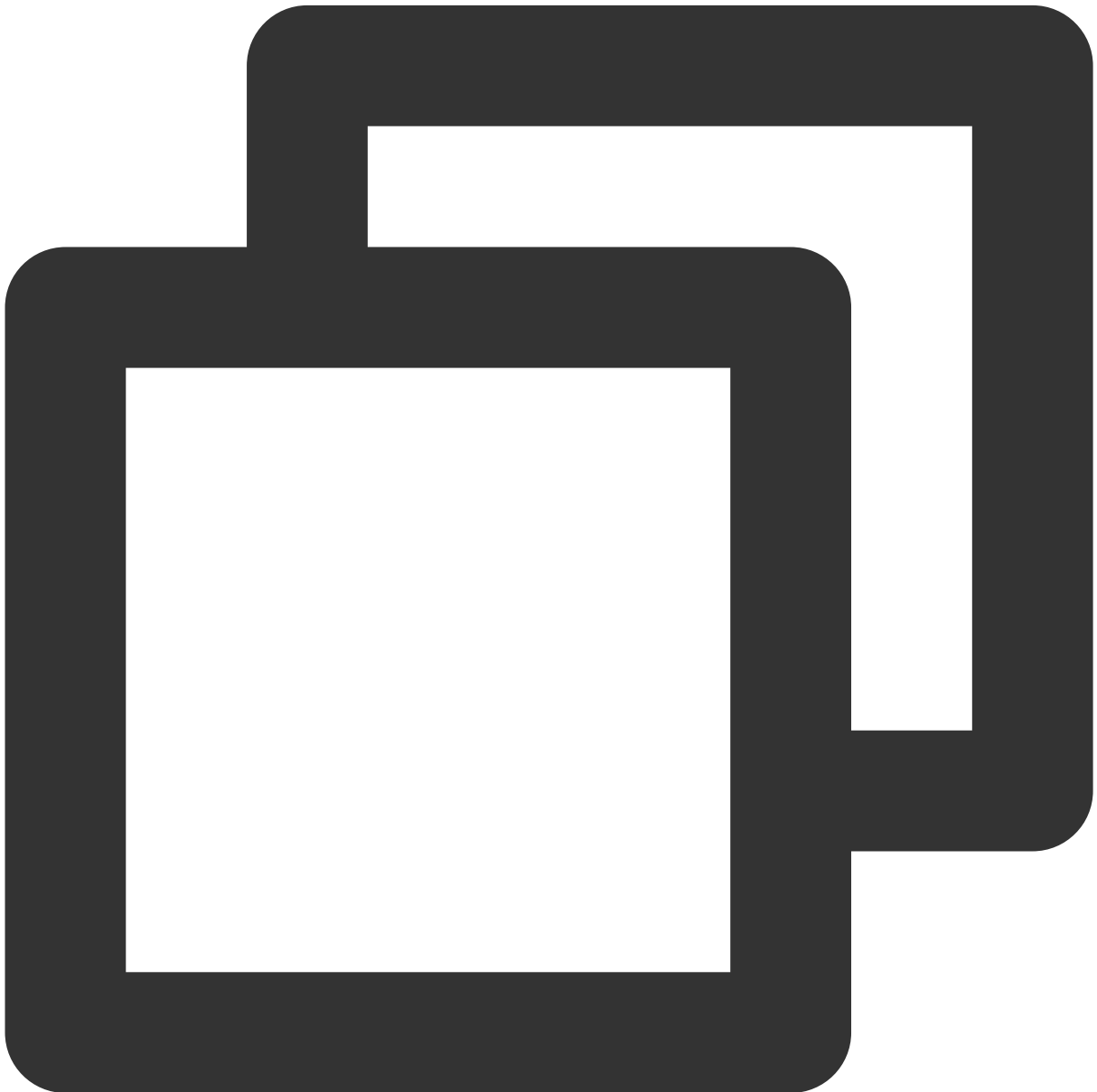
以 COS 数据为例，接收到的事件如下：



```
{
  "specversion": "1.0",
  "id": "13a3f42d-7258-4ada-da6d-023a333b4662",
  "type": "cos:created:object",
  "source": "cos.cloud.tencent",
  "subject": "qcs::cos:ap-guangzhou:uid1250000000:bucketname",
  "time": "1615430559146",
  "region": "ap-guangzhou",
  "datacontenttype": "application/json;charset=utf-8",
  "resource": [
    "qcs::eb:ap-guangzhou:uid1250000000:eventbusid/eventruleid"
  ]
}
```

```
],  
  "data": {  
    "name": "testname",  
    "scope": 100,  
    "source-ip": "10.0.0.123"  
  }  
}
```

指定 data 字段的 source-ip 包含匹配值，可以被正常触发的规则如下：



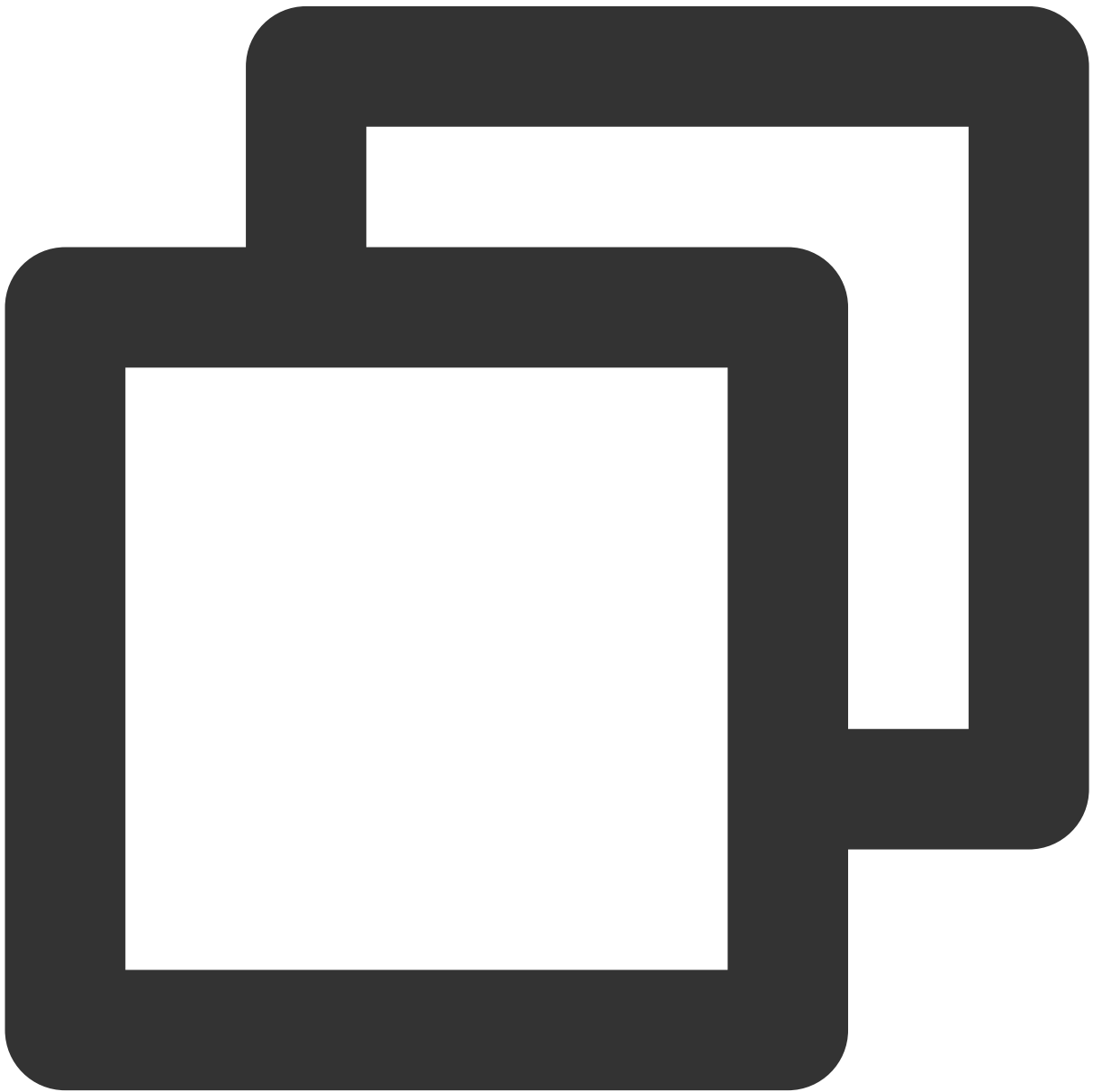
```
{  
  "data": {
```

```
"source-ip": [{  
  "cidr": "10.0.0.0/24"  
}]  
}  
}
```

更多说明

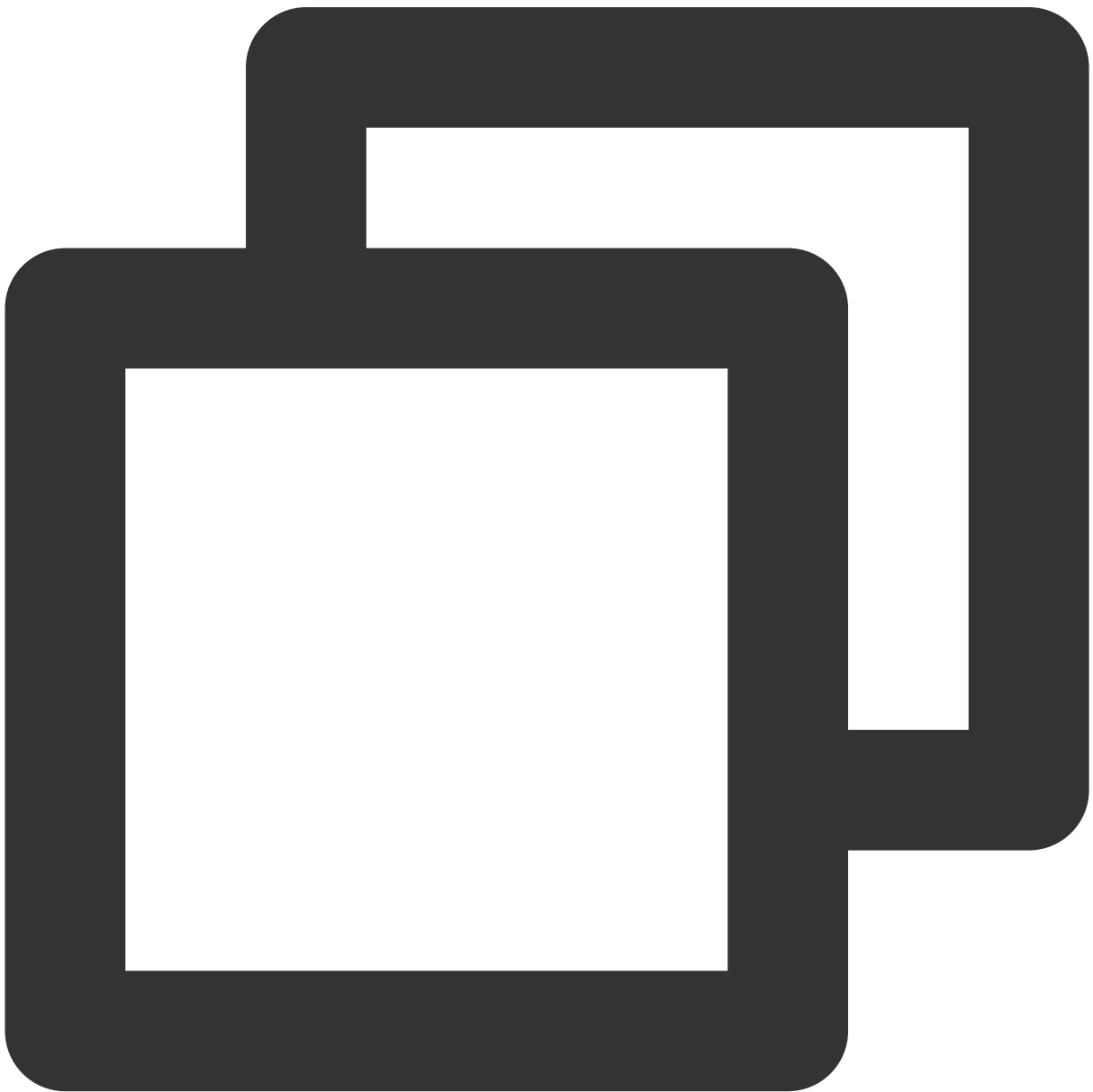
进行模式匹配时，`null` 值和空字符串不等同。用于匹配空字符串的模式不会匹配到 `null` 值。

所有匹配模式可被嵌套使用，如下示例，同时嵌套除外匹配与前缀匹配。



```
{
  "data": {
    "name": [{
      "anything-but": {
        "prefix": "init"
      }
    }]
  }
}
```

所有匹配模式支持 OR 模式规则，如下示例，指定前缀匹配或指定后缀匹配。



```
{
  "data": {
    "topic": [
      {
        "prefix": "pre"
      },
      {
        "suffix": "suf"
      }
    ]
  }
}
```


创建事件规则

最近更新时间：2024-01-22 20:52:28

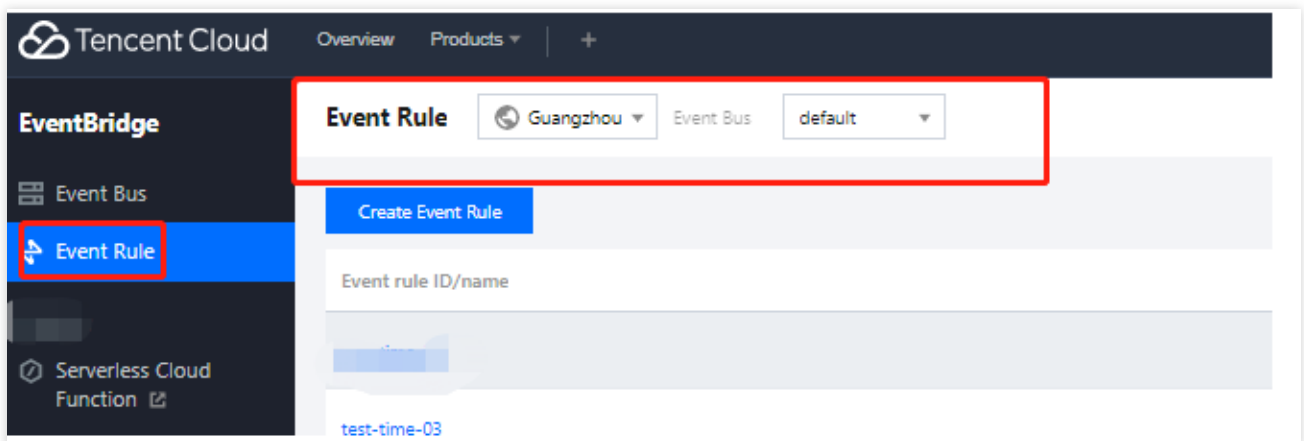
事件规则负责事件总线的最基础资源单位，可以通过配置事件集来完成对事件源的接收。本文为您介绍如何在事件总线控制台创建事件规则。

前提条件

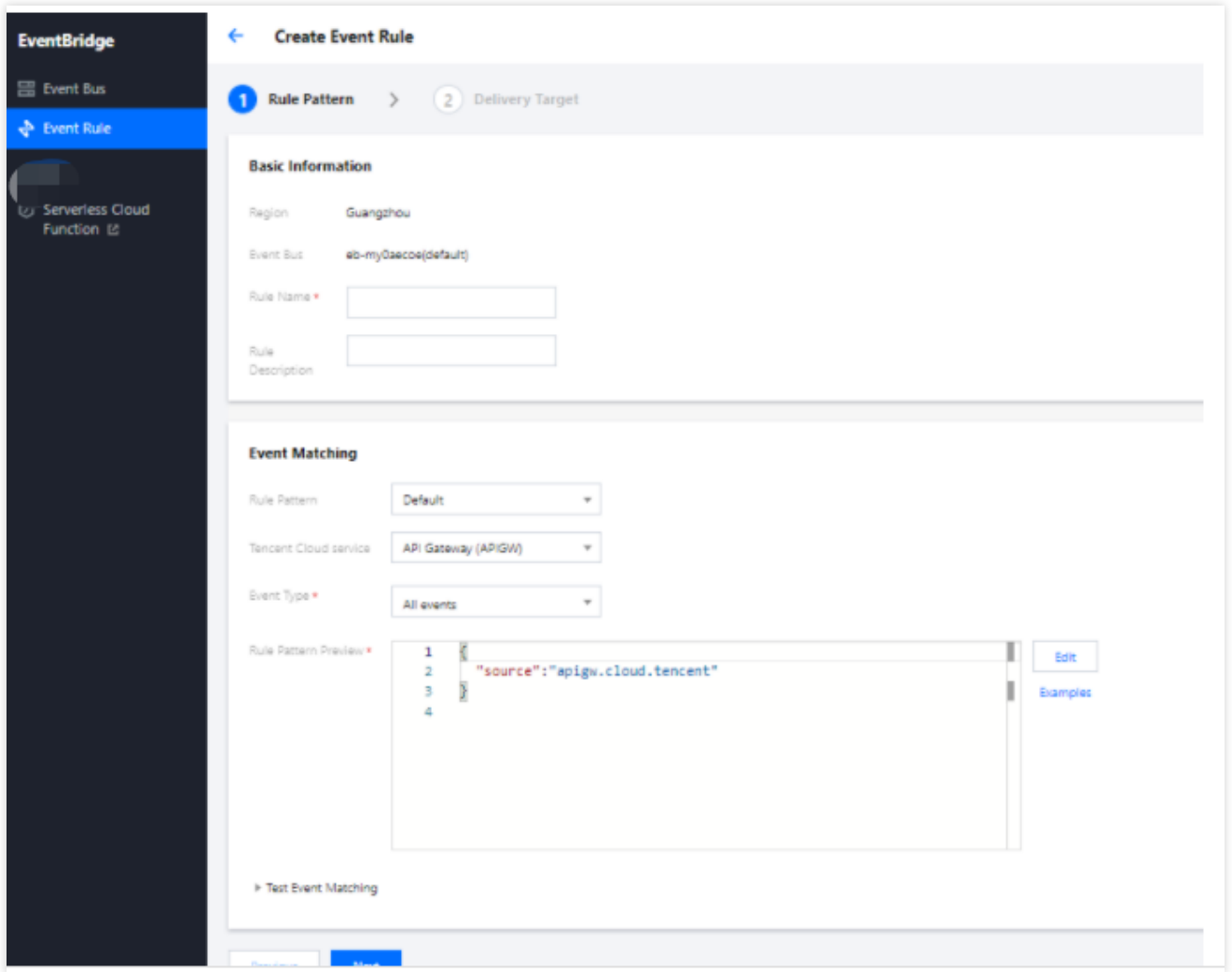
已 [创建事件集](#)。

操作步骤

1. 登录事件总线控制台，选择左侧导航栏中的 [事件规则](#)。
2. 在“事件规则”列表页面上方，选择期望创建事件规则的事件集归属及所在的地域。



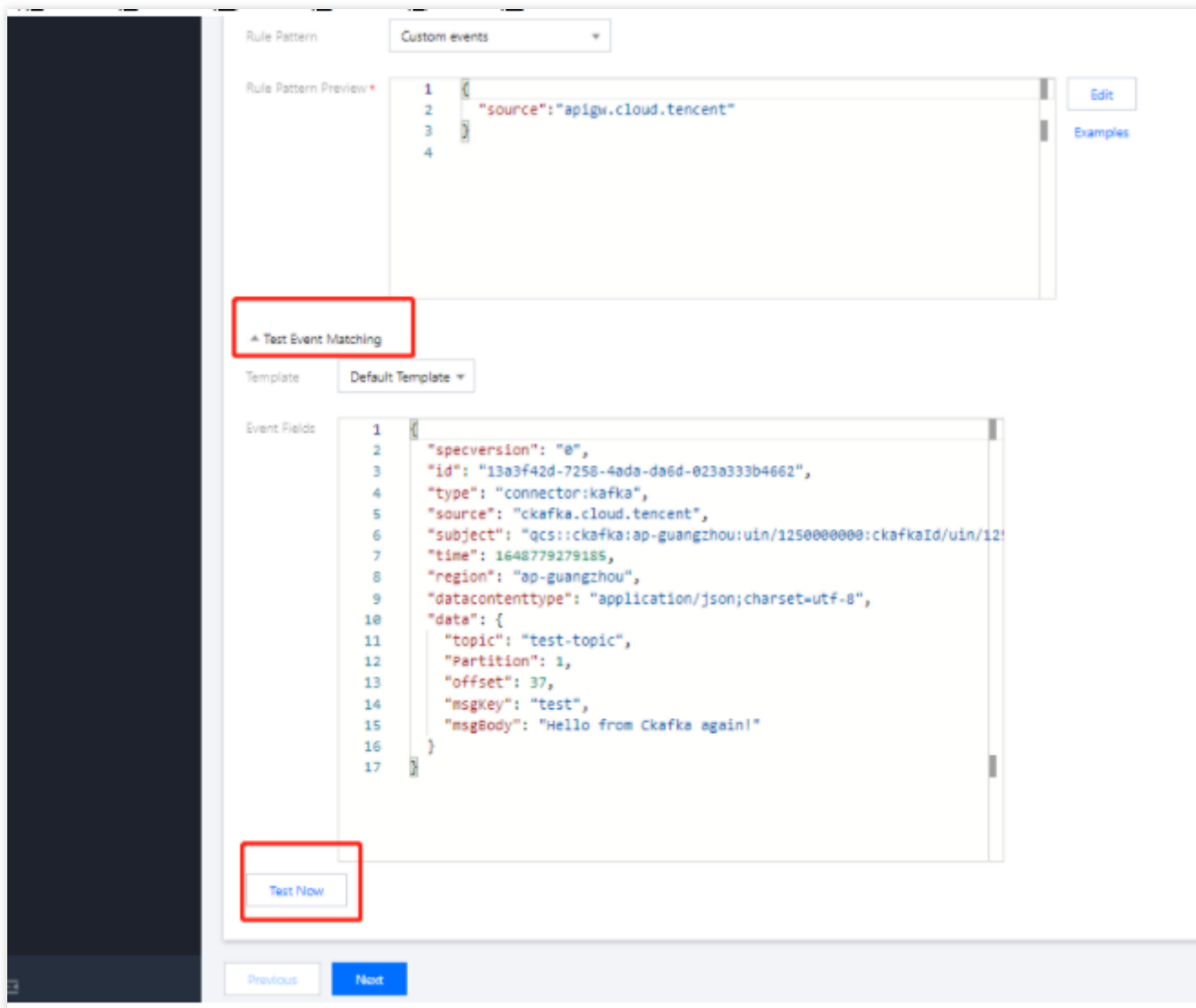
3. 单击 **新建事件规则**，根据页面提示填写相关信息，如下图所示：



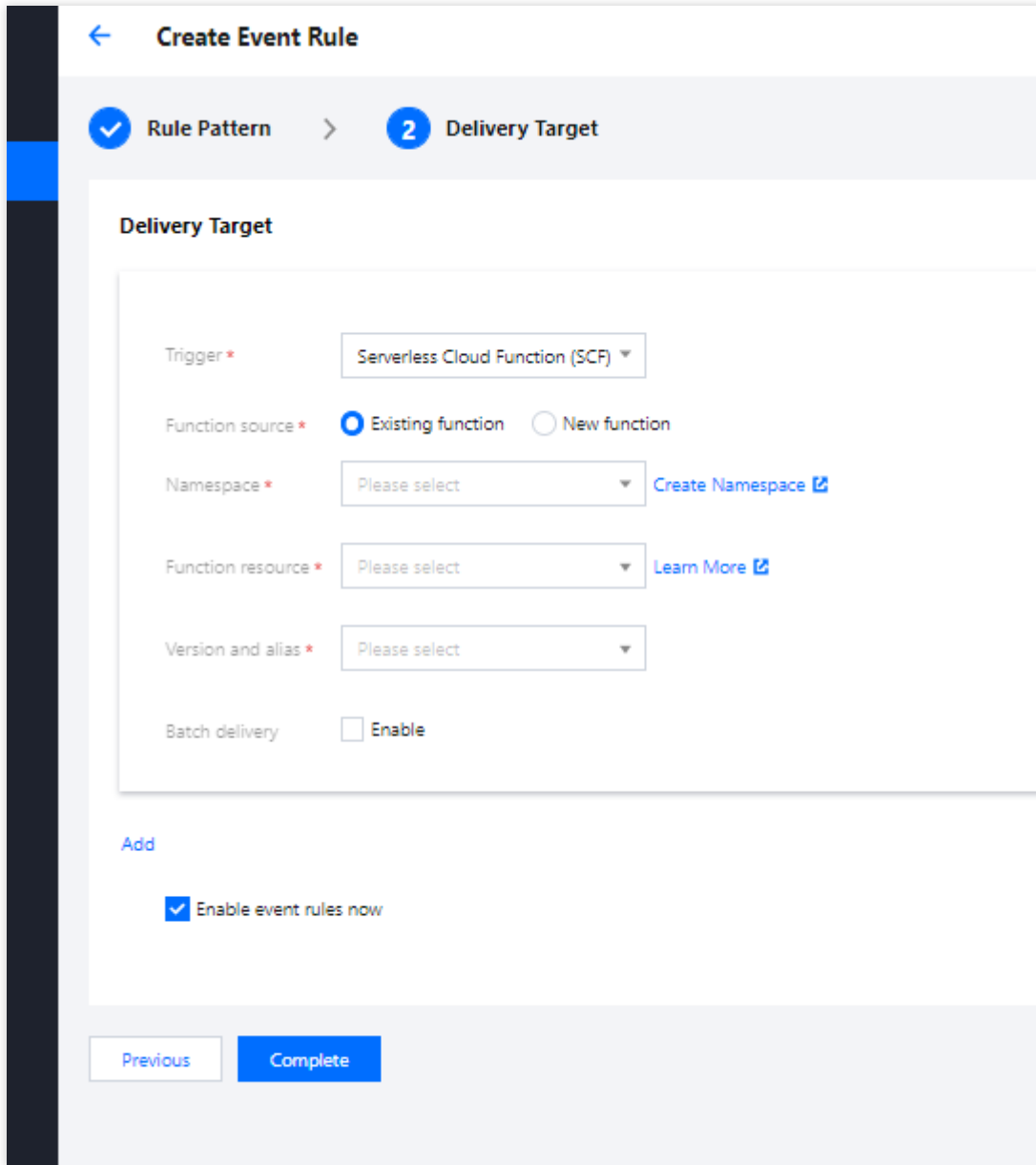
事件匹配：用于事件的过滤筛选。在此处定义何种事件可以被匹配触发，可以配置自定义事件匹配模式，或者选择已有的模板规则。如上图中，所有来自 TDMQ 消息队列的事件都可以通过匹配。更多事件模式编写规则，请参见 [事件模式](#) 文档。

事件目标：事件最终触发的目标。

4. 单击展开**测试事件匹配**后，可以对已定义的事件模式进行测试。**发送模板**中预置了当前支持的所有官方云服务事件模板与连接器生成的事件模板。



5. 单击**下一步**后选择该条规则绑定的事件目标，一条规则可以有多个事件目标。



事件总线当前支持以下事件目标，具体配置方法请参见：

[SCF 云函数](#)

[Ckafka](#)

管理事件规则

最近更新时间：2024-01-22 20:52:28

本文为您介绍如何查看、编辑和删除事件规则。

前提条件

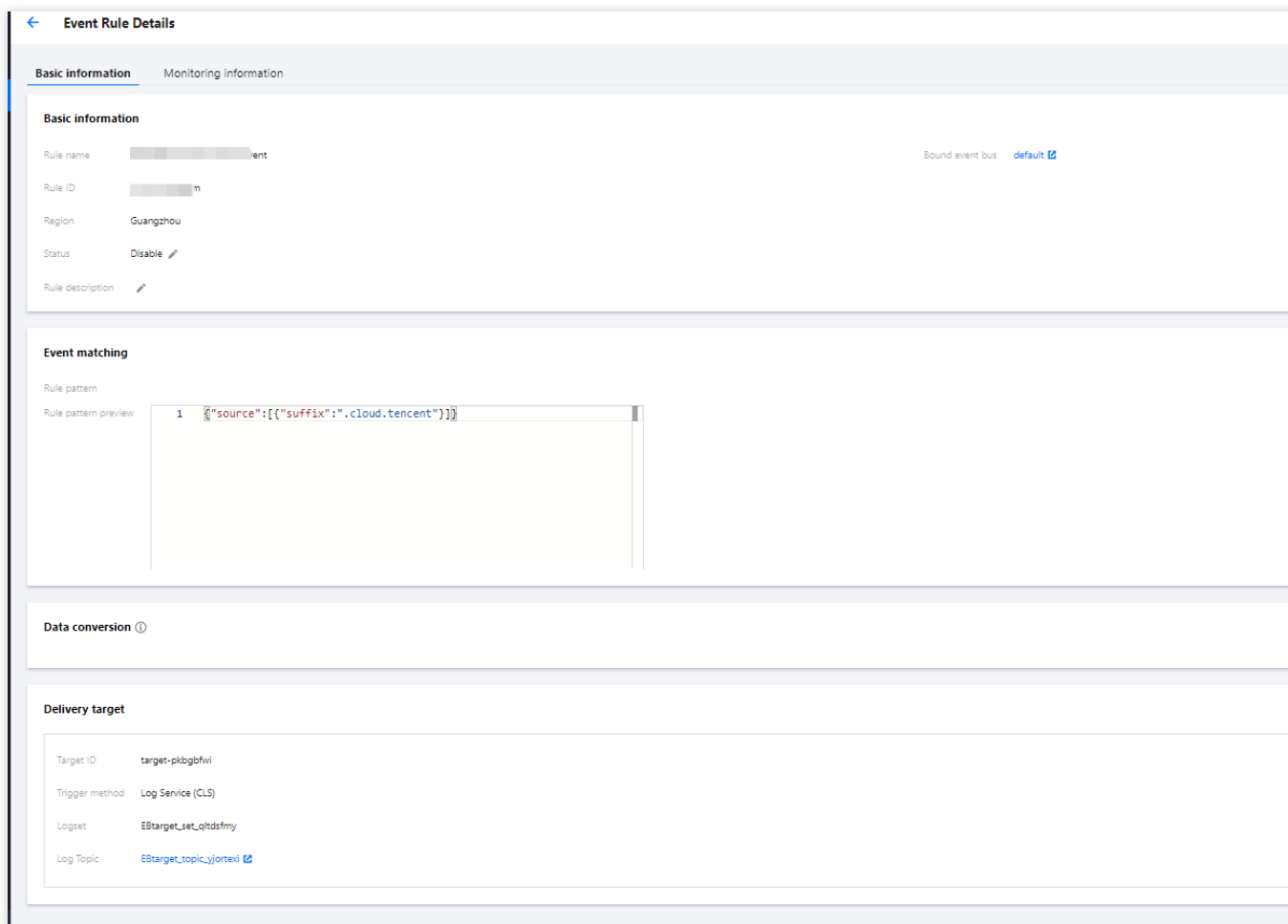
已 [创建事件规则](#)。

操作步骤

查看事件规则

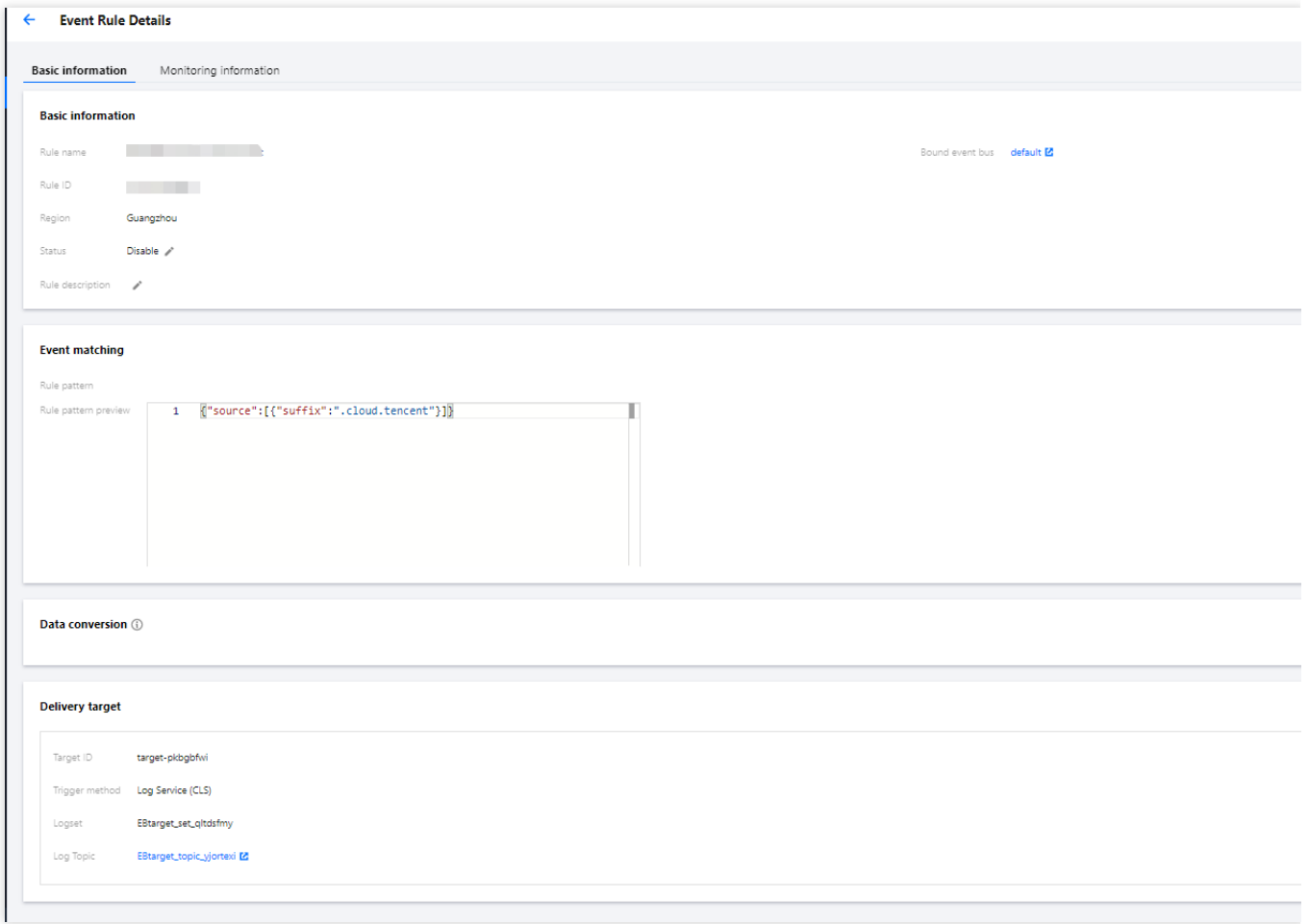
您可以通过以下操作查看事件集的详细信息。

1. 登录 [事件规则控制台](#)。
2. 在顶部菜单栏，选择地域与事件集。
3. 在事件规则列表单击需要查看的事件规则。
4. 事件规则页面显示事件规则的基础信息、事件匹配信息、事件目标等信息。如下图所示：



编辑事件规则

1. 登录 [事件规则控制台](#)。
2. 在顶部菜单栏，选择地域与事件集。
3. 在事件规则列表单击需要编辑的事件规则。
4. 事件规则页面可编辑基础信息、事件匹配、删除/新增事件目标等内容。如下图所示：

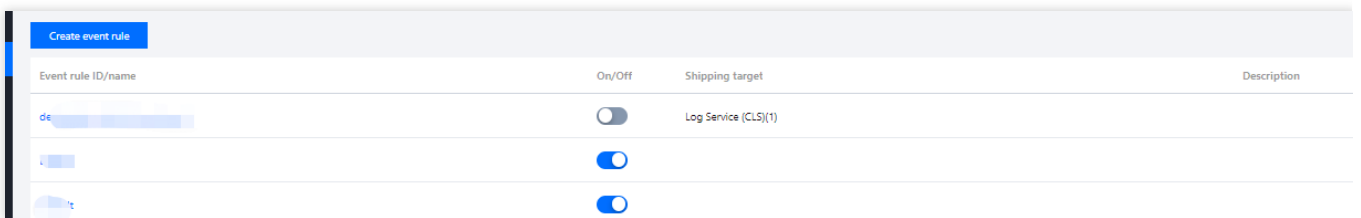


删除事件规则

注意：

在删除事件规则之前，您需要确认已删除该规则下的所有事件目标。

1. 登录 [事件规则控制台](#)。
2. 在顶部菜单栏，选择地域与事件集。
3. 在事件规则列表找到您需要删除的事件规则，在其右侧操作栏下单击**删除**。如下图所示：



4. 在弹出的确认窗口中确认删除即可。

配置数据转换

最近更新时间：2024-01-22 20:52:28

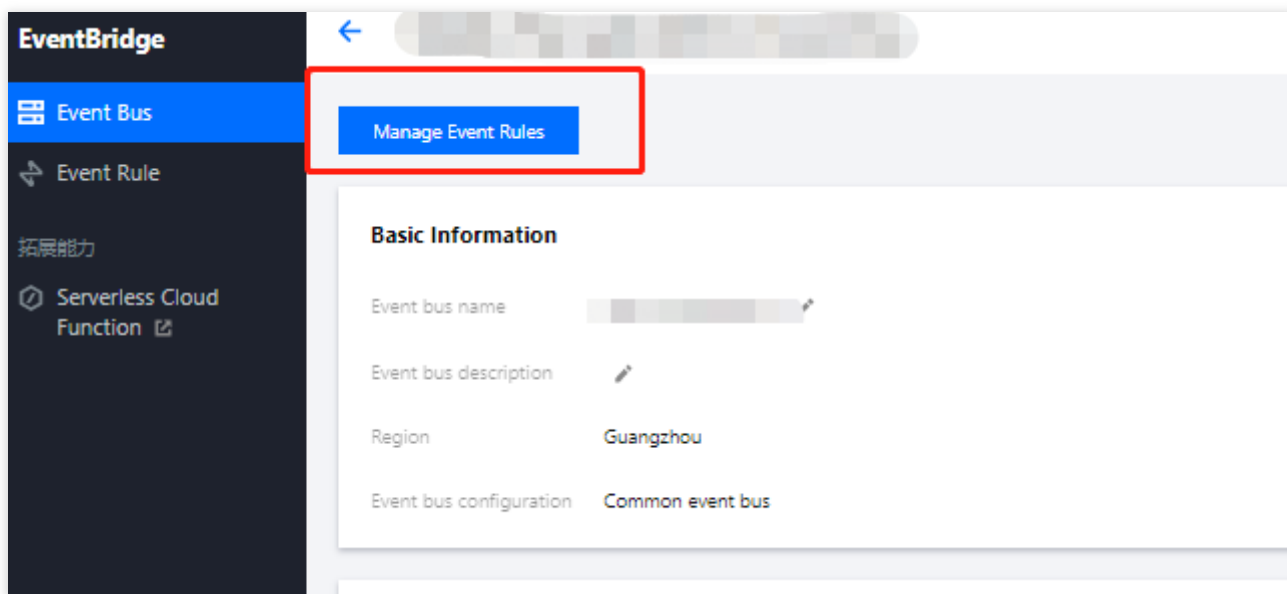
操作场景

除了基础事件筛选之外，EventBridge 还提供简单的数据处理功能，通过传入数据和配置项，可以实现对数据格式化处理，然后返回处理完成的结构化数据，分发下游目标，构建数据源和数据处理系统间的桥梁。

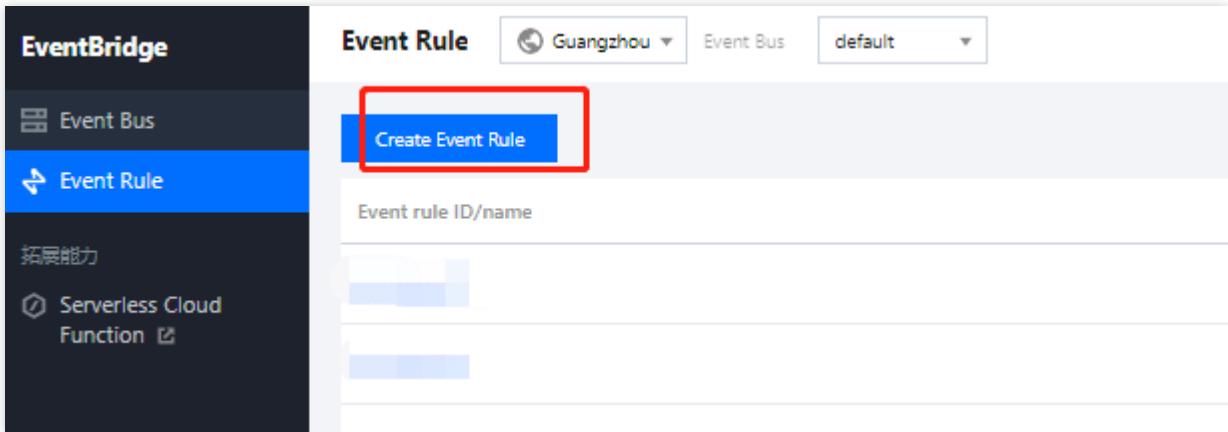
操作步骤

创建规则

1. 登录 [事件总线控制台](#)，选择指定事件集。
2. 在事件集详情页，单击**管理事件规则**，进行新增规则配置。如下图所示：



3. 进入**事件规则**页面，单击**新建事件规则**。如下图所示：



4. 根据提示，填写任务基本信息，勾选**立即启用数据转换**。

5. 单击**下一步**，设置数据转换规则。

事件模式：支持**模板数据**或者**自定义**。

解析格式：支持 **JSON** 解析方式。

6. 选择解析模式后，单击**确认**按钮，开始解析数据。

7. 解析完成后，设置过滤器规则和数据处理方式。

说明：

输出格式目前支持 JSON。

过滤器：仅输出符合过滤器规则的数据。过滤器的匹配模式支持**前缀匹配**、**后缀匹配**、**包含匹配 (contains)**、**除外匹配 (except)**、**数值匹配**和**IP匹配**。

数据处理：TYPE 有**默认**、**系统预设**、**映射**和**自定义** 四种。

TYPE = 默认：VALUE 从解析结果中映射，不可编辑。

TYPE = 系统预设：可以选择系统预设的 VALUE ，目前支持 DATE（时间戳）。

TYPE = 映射：可以选择已有的 KEY，最终输出的 VALUE 值由指定的 KEY 映射而来。

TYPE = 自定义：可以输入自定义 VALUE。

8. 单击**测试**，查看测试结果。

9. 单击**下一步**，完成数据目标绑定。

编辑规则

在**规则详情**页面，单击**数据转换**模块右上角的**编辑**，可修改数据处理规则。您也可以在详情页面进行规则的删除或添加。

过滤器规则说明

过滤器可以对数据进行规则过滤，例如过滤字段的大小等，符合规则的数据才会被保留。

注意事项

过滤器匹配是逐字符精确匹配的，需注意大小写，匹配过程中不会对字符串进行任何标准化的操作。要匹配的值遵循 JSON 规则：用引号引起来的字符串、数字以及不带引号的关键字 true、false 和 null。

前缀匹配

您可以对比数据中的前缀进行键值匹配。

例如数据 `{"password":"topicname"}`，指定 `password` 的前缀匹配值为 `top`，则 `{"password":"topicname"}` 可以被正常匹配。

后缀匹配

您可以对比数据中的后缀进行键值匹配。

例如数据 `{"password":"topicname"}`，指定 `password` 的后缀匹配值为 `name`，则 `{"password":"topicname"}` 可以被正常匹配。

包含匹配

您可以指定数据中存在的某个字段进行匹配。

例如数据 `{"password":"topicname"}`，指定 `password` 的包含匹配值为 `na`，则 `{"password":"topicname"}` 可以被正常匹配。

除外匹配

您可以指定某个字段除了提供的值之外的任何值进行匹配。

例如数据 `{"password":"topicname"}`，指定 `password` 的除外匹配值为 `topicname`，则只有 `{"password":"topicname"}` 不可以被正常匹配。

数值匹配

您可以指定某个字段的数值或范围。

例如数据 `{"numeric": 10}`，指定 `numeric` 的数值匹配为 `<15`，则 `{"numeric": 10}` 可以被正常匹配。

数值匹配规则举例：

大于10，输入`>10`

大于等于10，输入`>=10`

大于等于10且小于等于20，输入`>=10&<=20`

大于等于10或小于等于5，输入`>=10|<=5`

IP 地址匹配

您可以指定数据中字段的 IP 地址。输入内容为 CIDR 地址，例如输入`1.2.3.4/24`即匹配前24位所有1.2.3开头的 IP 地址。