

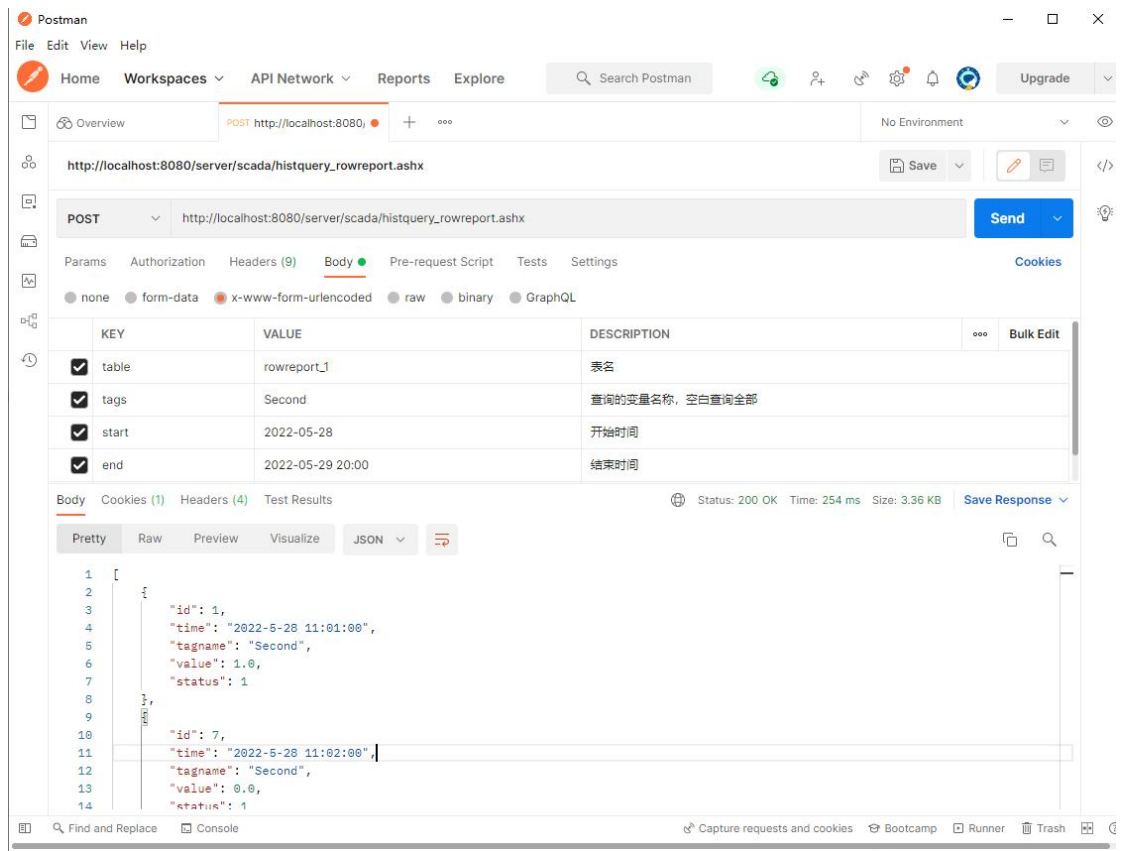
WTGateway 归档前端接口说明

WTGateway 提供了一组 Http 接口用于在前端方式行表、列表归档数据，可用于归档数据查询和分析，所有接口都是 POST 表单方式，返回 JSON 格式数据。

1) 行表归档接口

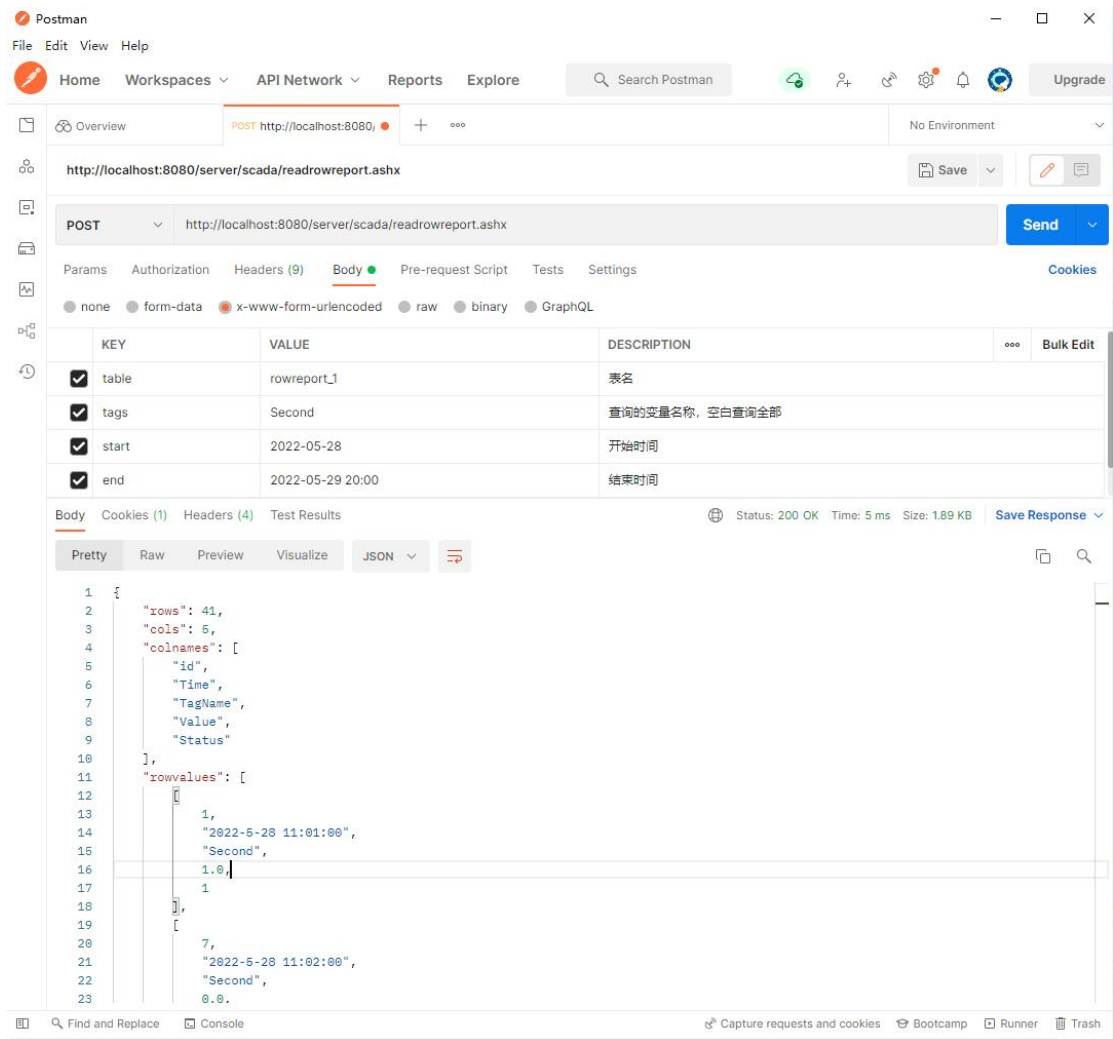
/server/scada/histquery_rowreport.ashx

查询行表归档，JSON 字段名全部为小写



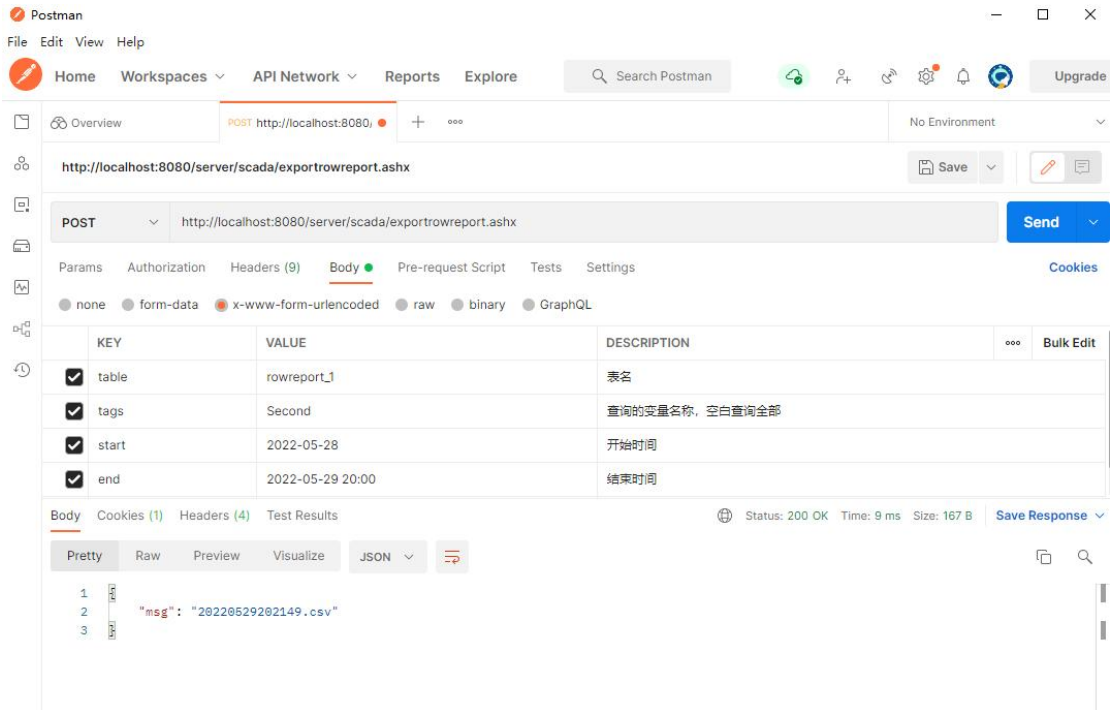
</server/scada/readrowreport.ashx> (2.11 后提供)

查询行表归档，JSON 数据包括字段列表和数据列表，适合直接拼接产生 html 表格。



[/server/scada/exportrowreport.ashx](#) (2.11 后提供)

导出 CSV 格式到 exports 目录下



文件路径在 exports 目录下，可以从 web 下载。

</server/scada/queryrowreport.ashx> (2.14.2 后提供)

查询行表归档数据，直接转换为 JSON 格式

POST http://localhost:8080/server/scada/queryrowreport.ashx

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> table	rowreport_1	表名
<input checked="" type="checkbox"/> tags	Second	查询的变量名称, 空白查询全部
<input checked="" type="checkbox"/> start	2022-05-28	开始时间
<input checked="" type="checkbox"/> end	2022-05-29 20:00	结束时间

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 5 ms Size: 3.4 KB Save Response

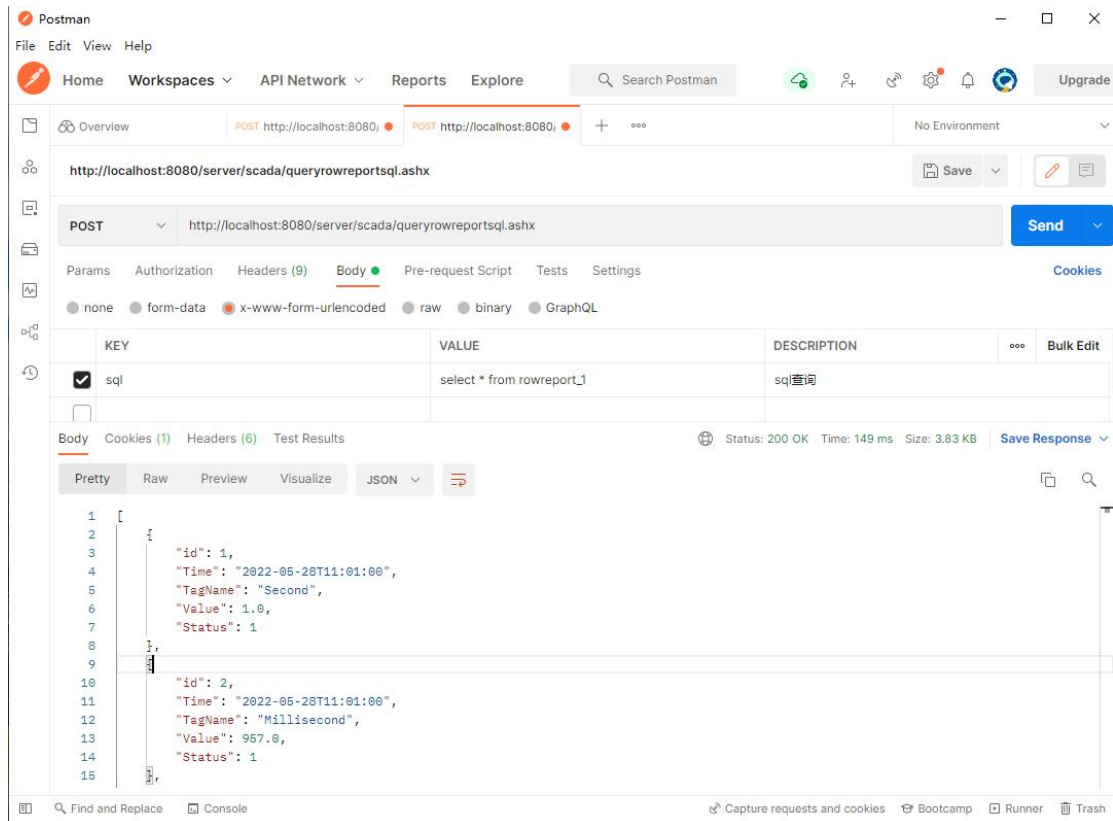
Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "Time": "2022-05-28T11:01:00",
5     "TagName": "Second",
6     "Value": 1.0,
7     "Status": 1
8   },
9   {
10    "id": 7,
11    "Time": "2022-05-28T11:02:00",
12    "TagName": "Second",
13    "Value": 0.0,
14    "Status": 1
15  },
16  {
17    "id": 13,
18    "Time": "2022-05-28T11:03:00",
19    "TagName": "Second",
20    "Value": 0.0,
21    "Status": 1
22  },
23 ]
```

这个接口跟/server/scada/histquery_rowreport.ashx 接口返回的数据很相似，区别在于字段是数据库的原始字段，直接从 DataTable 序列化而来，/server/scada/histquery_rowreport.ashx 字段全部为小写，时间日期格式作了处理。

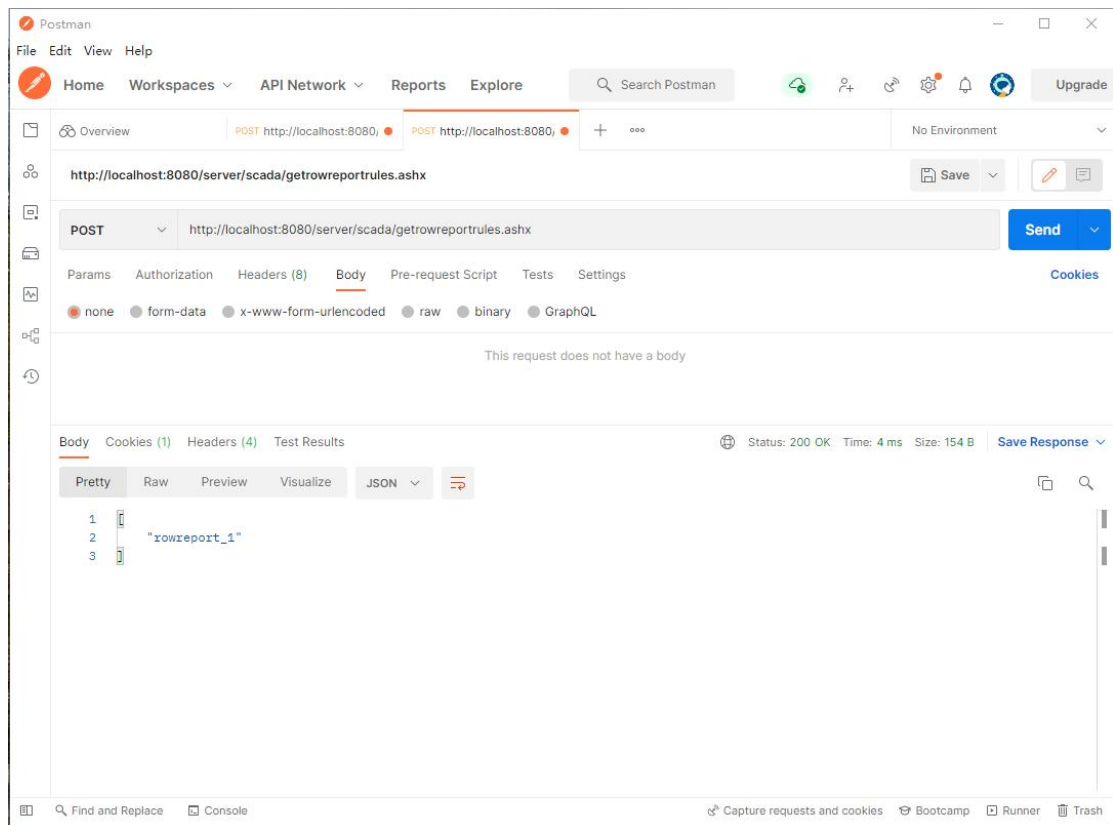
[/server/scada/queryrowreportsql.ashx](#) (2.14.2 后提供)

在行表归档数据库执行 SQL 查询，仅支持 select 开头的查询



[/server/scada/getrowreportrules.ashx](#) (2.14.2 后提供)

获取行表归档配置的表列表



[/server/scada/getrowreporttags.ashx](#) (2.14.2 后提供)

获取指定行表归档的变量名称列表

Postman interface showing a POST request to `http://localhost:8080/server/scada/getrowreporttags.ashx`. The request body is `x-www-form-urlencoded` with a parameter `name` set to `rowreport_1`. The response is a JSON array of strings: `[\"Second\", \"Millisecond\", \"TCPLinkCount\", \"IOLinkCount\", \"BlinkSlow\", \"Sim1#电源\"]`.

[/server/scada/exportexcelrowreport.ashx](#) (2.14.2 后提供)

导出 Excel 格式到 `exports` 目录下

Postman interface showing a POST request to `http://localhost:8080/server/scada/exportexcelrowreport.ashx`. The request body is `x-www-form-urlencoded` with parameters: `table` (rowreport_1), `tags`, `start` (2022-05-28), and `end` (2022-05-29 20:00). The response is a JSON object: `{\"msg\": \"20220530201311.xlsx\"}`.

2) 列表归档接口

[/server/scada/histquery_report.ashx](#)

查询列表归档，JSON 字段名称全部为小写

Postman interface showing a successful POST request to `http://localhost:8080/server/scada/histquery_report.ashx`. The request body is `x-www-form-urlencoded` with the following parameters:

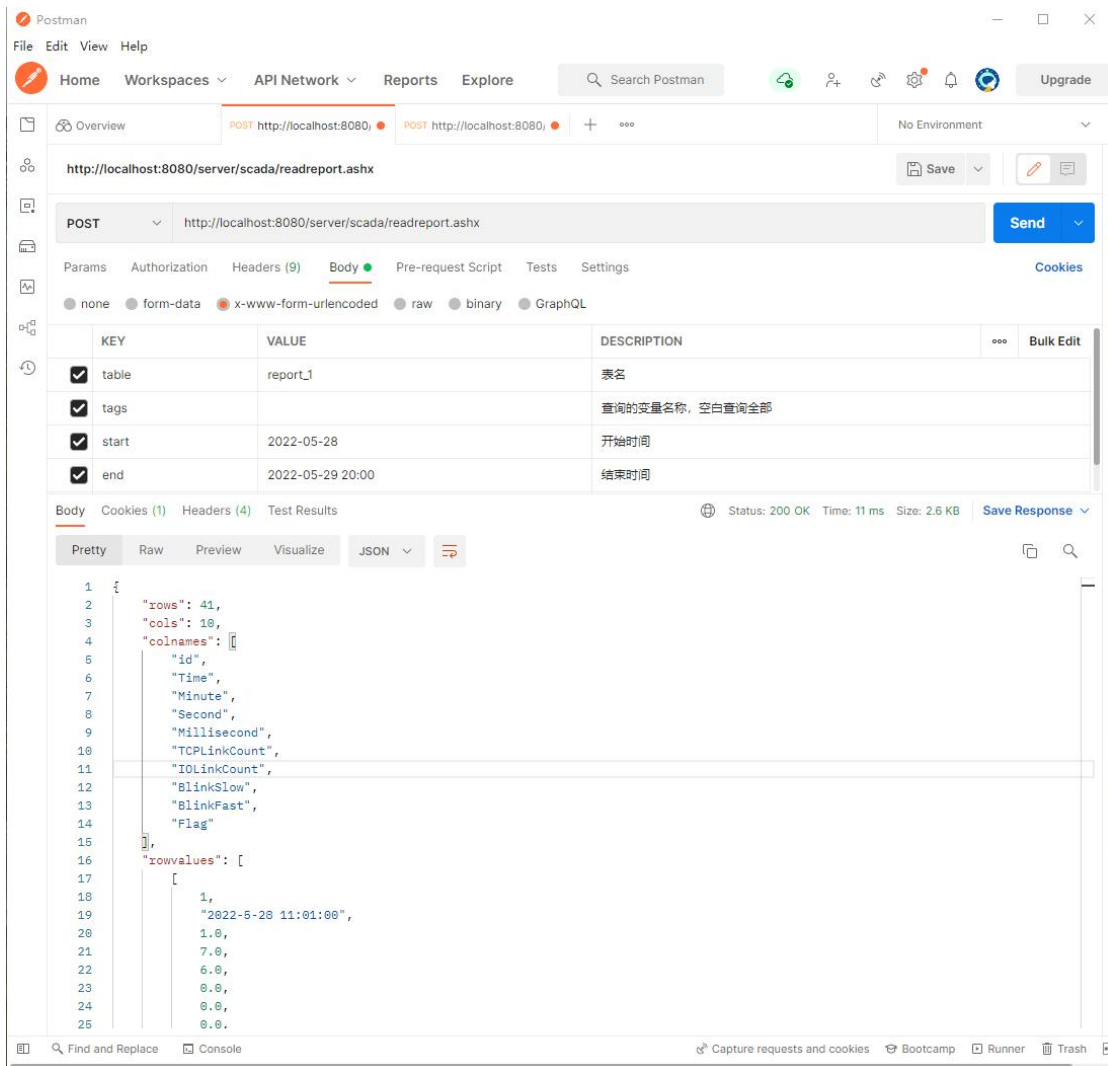
KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> table	report_1	表名
<input checked="" type="checkbox"/> tags		查询的变量名称, 空白查询全部
<input checked="" type="checkbox"/> start	2022-05-28	开始时间
<input checked="" type="checkbox"/> end	2022-05-29 20:00	结束时间

The response is a JSON array of two objects, each containing the following fields:

```
1 {
2   "id": 1,
3   "time": "2022-5-28 11:01:00",
4   "minute": 1.0,
5   "second": 7.0,
6   "millisecond": 6.0,
7   "toplinkcount": 0.0,
8   "iolinkcount": 0.0,
9   "blinkslow": 0.0,
10  "blinkfast": 0.0,
11  "flag": 1
12 }
13 ,
14 {
15   "id": 2,
16   "time": "2022-5-28 11:02:00",
17   "minute": 2.0,
18   "second": 0.0,
19   "millisecond": 54.0,
20   "toplinkcount": 0.0,
21   "iolinkcount": 0.0,
22   "blinkslow": 0.0,
23   "blinkfast": 1.0,
24   "flag": 0
25 }
```

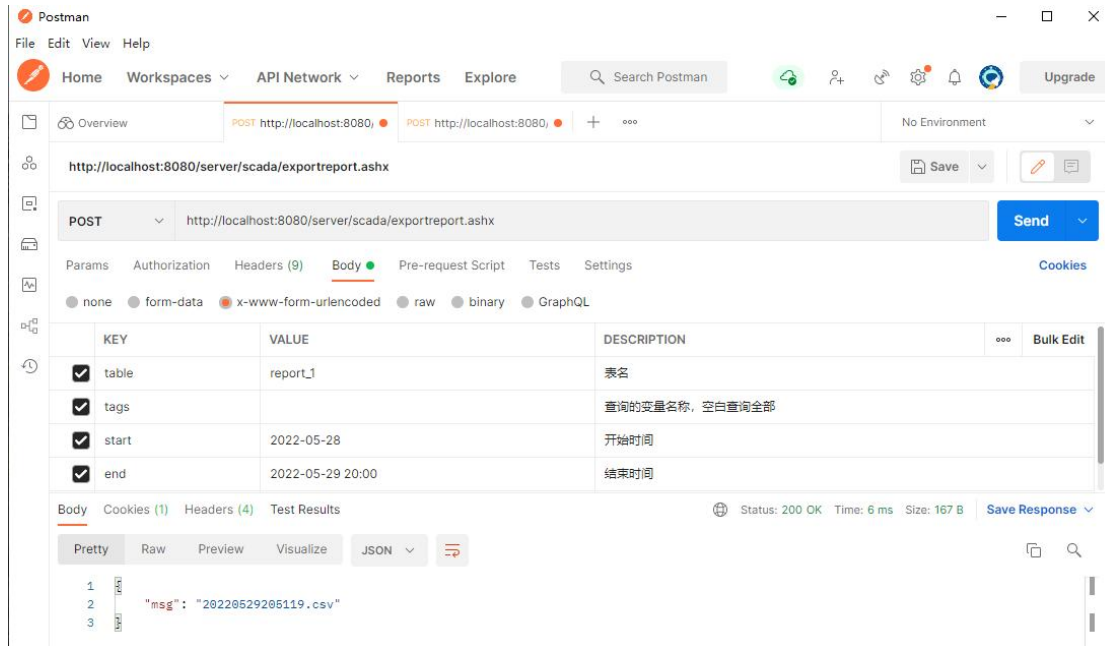
[/server/scada/readreport.ashx](#) (2.11 后提供)

查询列表归档, 字段和数据分 2 个数组返回



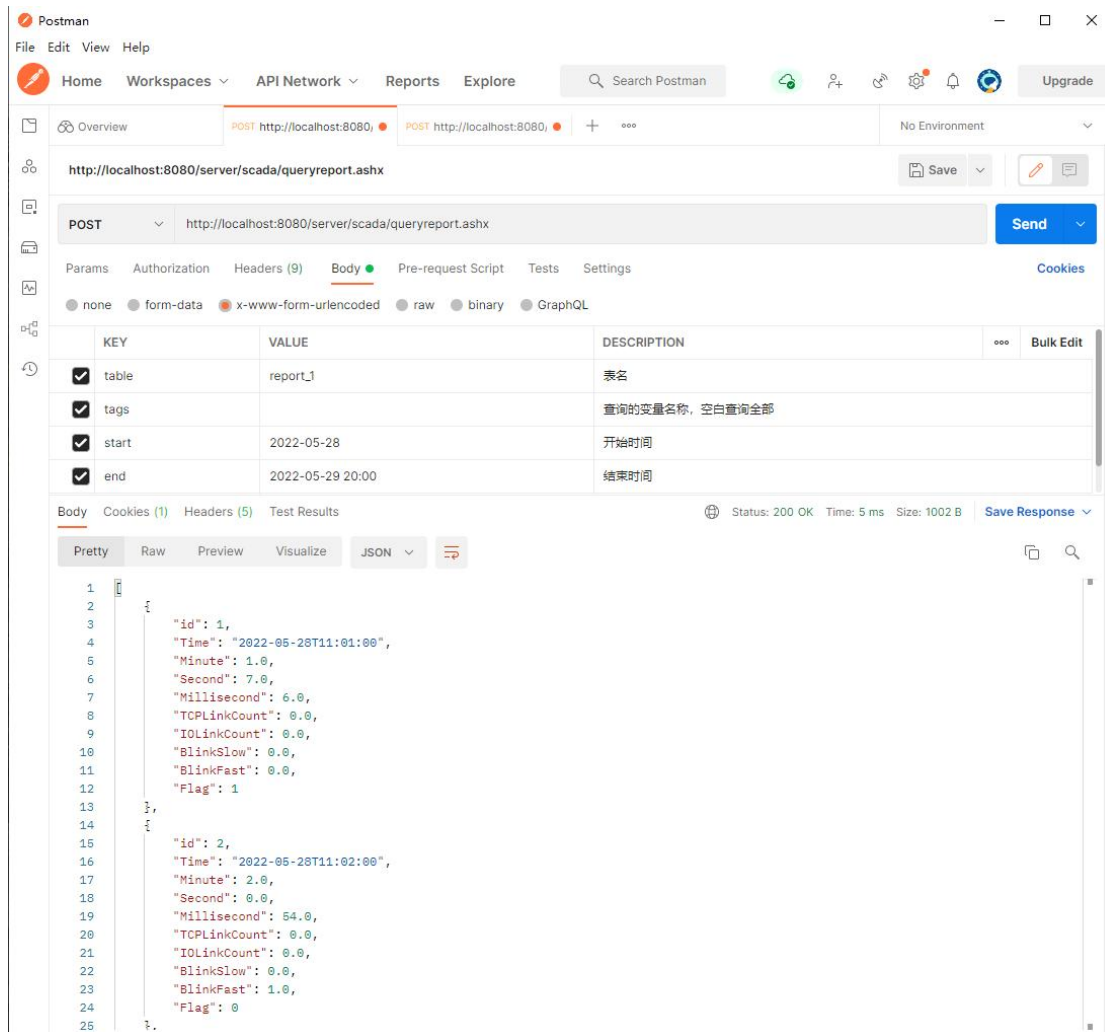
[/server/scada/expoortreport.ashx](#) (2.11 后提供)

导出列表归档到 csv 文件, 文件存在在 exports 目录下, 可以通过 web 下载



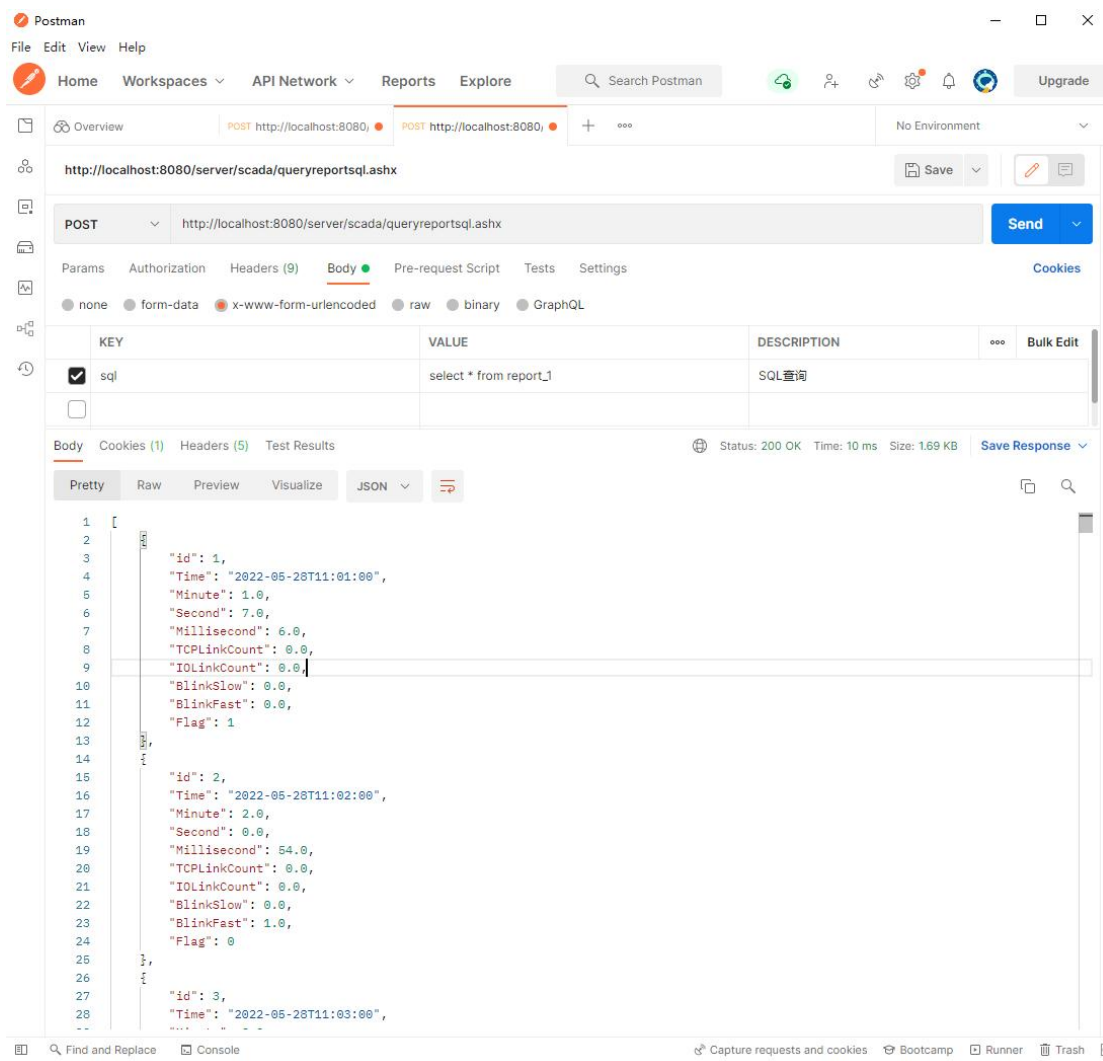
[/server/scada/queryreport.ashx](#) (2.14.2 后提供)

查询列表归档, DataTable 直接序列化产生 JSON



[/server/scada/queryreportsql.ashx](#) (2.14.2 后提供)

在列表归档数据库上执行 SQL 查询，DataTable 直接序列化产生 JSON



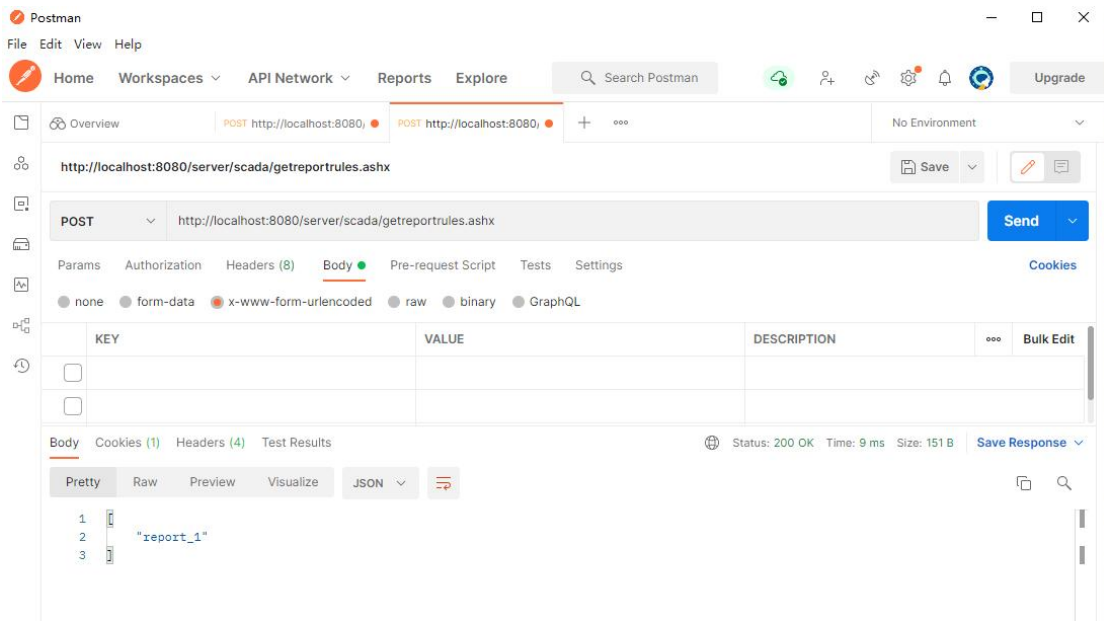
The screenshot shows the Postman interface for a POST request to `http://localhost:8080/server/scada/queryreportsql.ashx`. The request body is a JSON object with a `sql` key containing the query `select * from report_1`. The response status is 200 OK, and the response body is a JSON array of three objects, each representing a report entry with fields like `id`, `Time`, `Minute`, `Second`, `Millisecond`, `TCPLinkCount`, `IOLinkCount`, `BlinkSlow`, `BlinkFast`, and `Flag`.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> sql	select * from report_1	SQL查询

```
1 [
2   {
3     "id": 1,
4     "Time": "2022-05-28T11:01:00",
5     "Minute": 1.0,
6     "Second": 7.0,
7     "Millisecond": 6.0,
8     "TCPLinkCount": 0.0,
9     "IOLinkCount": 0.0,
10    "BlinkSlow": 0.0,
11    "BlinkFast": 0.0,
12    "Flag": 1
13  },
14  {
15    "id": 2,
16    "Time": "2022-05-28T11:02:00",
17    "Minute": 2.0,
18    "Second": 0.0,
19    "Millisecond": 54.0,
20    "TCPLinkCount": 0.0,
21    "IOLinkCount": 0.0,
22    "BlinkSlow": 0.0,
23    "BlinkFast": 1.0,
24    "Flag": 0
25  },
26  {
27    "id": 3,
28    "Time": "2022-05-28T11:03:00",
29    ...
30  }
31 ]
```

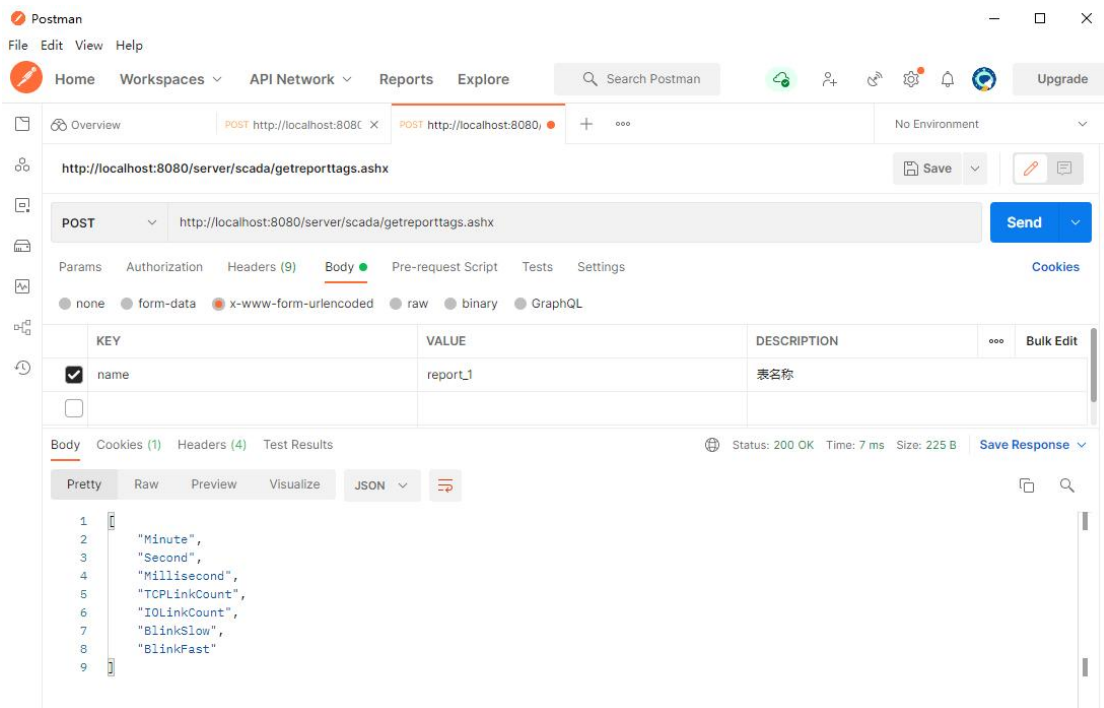
[/server/scada/getreportrules.ashx](#) (2.14.2 后提供)

获取列表归档的表名列表



[/server/scada/getreporttags.ashx](#) (2.14.2 后提供)

获取指定列表归档的变量名称



[/server/scada/exportexcelreport.ashx](#) (2.14.2 后提供)

导出 Excel 格式到 exports 目录下

Overview POST http://localhost:8080/

http://localhost:8080/server/scada/exportexcelreport.ashx

POST http://localhost:8080/server/scada/exportexcelreport.ashx

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> table	report_1	表名	
<input checked="" type="checkbox"/> tags		查询的变量名称, 空白查询全部	
<input checked="" type="checkbox"/> start	2022-05-28	开始时间	
<input checked="" type="checkbox"/> end	2022-05-29 20:00	结束时间	
Key	Value	Description	

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 15 ms Size: 168 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "msg": "20220530201551.xlsx"
3 }

```

/logquery 日志查询接口

Home Workspaces API Network Reports Explore Search Postman Upgrade

Overview POST http://localhost:8080/

http://localhost:8080/logquery

POST http://localhost:8080/logquery

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> p1	0	系统日志查询	
<input checked="" type="checkbox"/> p2	2022-7-9	开始时间	
<input checked="" type="checkbox"/> p3	2022-7-10	结束时间	
<input type="checkbox"/>			

Body Cookies (1) Headers (5) Test Results Status: 200 OK Time: 12 ms Size: 1.5 KB Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "zet": 0,
3   "msgtype": "",
4   "message": "",
5   "rows": [
6     {
7       "machine": "xENYUAN2",
8       "severity": 0,
9       "user": "admin",
10      "time": "2022-07-09 19:54:13.623",
11      "source": "System",
12      "data": "删除日志记录",
13      "desc": "",
14      "status": ""
15    }
16  ]
17 }

```

Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

系统日志查询

Home Workspaces API Network Reports Explore Search Postman Upgrade

Overview POST http://localhost:8080/ No Environment

http://localhost:8080/logquery Save Send

POST http://localhost:8080/logquery Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> p1	3	报警历史查询	
<input checked="" type="checkbox"/> p2	2022-7-9	开始时间	
<input checked="" type="checkbox"/> p3	2022-7-10	结束时间	
Key	Value	Description	

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 8 ms Size: 972 B Save Response

Pretty Raw Preview Visualize JSON

```

31  {
32    "source": "P",
33    "data": "10.00",
34    "desc": "放大倍数",
35    "status": "低低报警"
36  },
37  {
38    "machine": "WENYUAN2",
39    "severity": 3,
40    "user": "admin",
41    "time": "2022-07-09 20:23:28.612",
42    "source": "I",
43    "data": "50.00",
44    "desc": "积分时间",
45    "status": "高报警"
46  }

```

Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

查询指定时间范围的历史报警记录

Home Workspaces API Network Reports Explore Search Postman Upgrade

Overview POST http://localhost:8080/ No Environment

http://localhost:8080/logquery Save Send

POST http://localhost:8080/logquery Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> p1	3	报警历史查询	
<input checked="" type="checkbox"/> p2	2022-7-9	开始时间	
<input checked="" type="checkbox"/> p3	2022-7-10	结束时间	
<input checked="" type="checkbox"/> p4	I	变量名称	
Key	Value	Description	

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 5 ms Size: 498 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "ret": 0,
3    "msgtype": "",
4    "message": "",
5    "rows": [
6      {
7        "machine": "WENYUAN2",
8        "severity": 3,
9        "user": "admin",
10       "time": "2022-07-09 20:23:28.612",
11       "source": "I",
12       "data": "50.00",
13       "desc": "积分时间",
14       "status": "高报警"
15     }
16   ]

```

Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

查询指定时间范围的某个变量的历史报警记录（2.14.3 后提供）

The screenshot shows a Postman interface for a POST request to `http://localhost:8080/logquery`. The parameters are:

KEY	VALUE	DESCRIPTION
p1	4	操作记录查询
p2	2022-7-9	开始时间
p3	2022-7-10	结束时间

The response body is JSON, showing a log entry for '放大倍数':

```
1 {
2   "zet": 0,
3   "msgtype": "",
4   "message": "",
5   "rows": [
6     {
7       "machine": "WENYUAN2",
8       "severity": 4,
9       "user": "admin",
10      "time": "2022-07-09 20:22:25.195",
11      "source": "P",
12      "data": "200",
13      "desc": "放大倍数",
14      "status": ""
15    },
16  ]
17 }
```

查询指定时间范围内的操作记录

The screenshot shows a Postman interface for a POST request to `http://localhost:8080/logquery`. The parameters are:

KEY	VALUE	DESCRIPTION
p1	4	操作记录查询
p2	2022-7-9	开始时间
p3	2022-7-10	结束时间
p4	I	变量名称

The response body is JSON, showing a log entry for '积分时间':

```
1 {
2   "zet": 0,
3   "msgtype": "",
4   "message": "",
5   "rows": [
6     {
7       "machine": "WENYUAN2",
8       "severity": 4,
9       "user": "admin",
10      "time": "2022-07-09 20:22:59.729",
11      "source": "I",
12      "data": "0",
13      "desc": "积分时间",
14      "status": ""
15    },
16  ]
17 }
```

查询指定时间范围内的某个变量的操作记录（2.14.3 后提供）

POST http://localhost:8080/logquery

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> p1	5	变量日志记录查询	
<input checked="" type="checkbox"/> p2	2022-7-9	开始时间	
<input checked="" type="checkbox"/> p3	2022-7-10	结束时间	

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 8 ms Size: 490 B Save Response

```

1 {
2   "ret": 0,
3   "msgtype": "",
4   "message": "",
5   "rows": [
6     {
7       "machine": "WENYUAN2",
8       "severity": 5,
9       "user": "admin",
10      "time": "2022-07-09 20:29:54.281",
11      "source": "autoMode",
12      "data": "1",
13      "desc": "手自动状态",
14      "status": "1"
15    }
16  ]
17 }

```

查询指定时间范围内的变量变化日志

POST http://localhost:8080/logquery

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> p1	5	变量日志记录查询	
<input checked="" type="checkbox"/> p2	2022-7-9	开始时间	
<input checked="" type="checkbox"/> p3	2022-7-10	结束时间	
<input checked="" type="checkbox"/> p4	autoMode	变量名称	

Body Cookies (1) Headers (4) Test Results Status: 200 OK Time: 4 ms Size: 490 B Save Response

```

1 {
2   "ret": 0,
3   "msgtype": "",
4   "message": "",
5   "rows": [
6     {
7       "machine": "WENYUAN2",
8       "severity": 5,
9       "user": "admin",
10      "time": "2022-07-09 20:29:54.281",
11      "source": "autoMode",
12      "data": "1",
13      "desc": "手自动状态",
14      "status": "1"
15    }
16  ]
17 }

```

查询指定时间范围内的某个变量的变化日志记录 (2.14.3 后提供)