

Question

[Rochdi Badis](#) · Oct 9, 2022

HTTP request with open stream

Hi Guys,

This link (<https://github.com/CassiaNetworks/CassiaSDKGuide/blob/master/nodeexamples...>) uses Javascript request to open a a steam and collect data in an array (screenshot below) and I'm trying to replicate this with Cache script with HTTP request which is connecting Ok but don't know how can I collect data because the code is hanging at `Set tSc=Httprequest.Get(Url)` and cannot execute `S BLEConnect=Httprequest.HttpResponse.StatusLine` or any line after it.

I think the url call

(ac1.mqtt.sx3ac.com/api/gap/nodes?mac=CC:1B:E0:E1:A6:44&accessToken=8f7f44d842cbcadd53713cca2ea36f256b2ec6581658fee4e246d93e28c6feb3&active=1&event=1

) should do that by keeping an open stream because if I run the same url through the browser or Postman it does the same, but how can I get the data the same way this JavaScript code is doing?

```
104 const query = {
105   /*
106    * filter devices whose rssi is below -75, and name begins with 'Cassia',
107    * there are many other filters, you can find them in document
108    * use proper filters can significantly reduce traffic between Router and AC
109    */
110   filter_rssi: -75,
111   filter_name: 'Cassia*',
112   /*
113    * use active scan, default is passive scan
114    * active scan makes devices response with data packet which usually contains device's name
115    */
116   active: 1,
117   mac: routerMac, // which router you want to start scan
118   access_token: token // you can put token in query 'access_token=token' or in header 'Bearer <token>'
119 };
120 const url = `${AC_HOST}/gap/nodes?event=1&${qs.encode(query)}`;
121 const sse = new EventSource(url);
122
123 sse.on('error', function(error) {
124   console.error('open scan sse failed:', error);
125 });
126
127 /*
128 * if scan open successful, it will return like follow:
129 * data: { "bdaaurs": [{" bdaaur": "E0:47:00:03:A9:18", "bdaddrType": "public"}, {"scanData": "0C09536C656570616365205A32", "name": "Sleepace Z2", "rssi": -37, "evt_type": 4}
130 */
131 sse.on('message', function(message) {
132   let data = JSON.parse(message.data);
133   let deviceMac = data.bdaaurs[0].bdaddr;
134   let addrType = data.bdaaurs[0].bdaddrType;
135   /*
136    * enqueue device data to connect it lately
137    * the scanning will get multiple scan data of same device in short time,
138    * so we need filter same device
139    */
140   connectQ.enqueue({mac: deviceMac, addrType});
141 });
142 }
```

```
s Token=##class(MSDS.SpaceSense.API).GetAuth()
Set Httprequest=##class(%Net.HttpRequest).%New()
Set Httprequest.Server="ac1.mqtt.sx3ac.com"
Set Httprequest.Timeout=30
Set Httprequest.Https=0
```

HTTP request with open stream

Published on InterSystems Developer Community (<https://community.intersystems.com>)

```
set Httprequest.ContentType="application/json"
Do Httprequest.SetHeader("Accept","application/json")
Do Httprequest.SetHeader("Accept-Language","enUS" )
Do Httprequest.SetHeader("Authorization","Bearer "Token)
Set HttpURL="/api/gap/nodes?filterrssi=-75&chip=1&mac=CC:1B:E0:E2:3C:10&active=1&event=1"
Set tSc=Httprequest.Get(HttpURL)
S BLEConnect=Httprequest.HttpResponse.StatusLine
```

Thanks

[#Ensemble](#)

Product version: Ensemble 2014.1

Source URL: <https://community.intersystems.com/post/http-request-open-stream>