How to Control Outlet and Get Status using HTTP API with PWR-RMT-RBT2-515R-LCV2

This document is for firmware version PWR-RMT-RBT2-515R-LCV2 S01/S02 v3 and above only.

1. Generate a Cookie-Token for User Login:

HTTP Packet Request

“POST /goform/login HTTP/1.1”
“Accept: */*”
“Accept-Encoding: gzip, deflate”
“Content-Type: application/x-www-form-urlencoded”
CRLF
“user=” <Web Account> “&password=” <Web Password>

NOTE:
- <Web Password>: The pass word of the PWR-RMT-RBT2-515R-LCV2’s web interface.

If you don’t know the information above, please refer to the user manual, or use a web browser to set up PWR-RMT-RBT2-515R-LCV2.

HTTP Packet Response

“Pragma: no-cache”
“Set-Cookie: WQJhuEcnAVA3t7WE+ug6A=<token>; HttpOnly; Path=/”
“Content-Type: text/html”
“Server: GoAhead-Webs”
“Location: /index.asp”
“Date: Thu Jul 1 01:01:01 2020”
“Cache-Control: no-cache”
CRLF

NOTE:
- <token>: The generated access token. It is valid for only 24 hours.

Example of cURL Request:
curl -v http://192.168.3.44/goform/login -X POST --data "user=admin&password=password"

Example of cURL Response:
* About to connect() to 192.168.3.44 port 80 (#0)
* Trying 192.168.3.44... connected
* Connected to 192.168.3.44 (192.168.3.44) port 80 (#0)
  > POST /goform/login HTTP/1.1
  > User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
    NSS/3.14.0.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
  > Host: 192.168.3.44
  > Accept: */*
  > Content-Length: 28
  > Content-Type: application/x-www-form-urlencoded
  >
  * HTTP 1.0, assume close after body
  < HTTP/1.0 200 Data follows
  < Server: GoAhead-Webs
  < Date: Fri Mar 25 10:10:14 2022
  < Set-Cookie: WQKJhuEcnsAVA3t7WE+ug6A=xbPdDGyK0T5OBNjWW; HttpOnly; Path=/
  < Pragma: no-cache
  < Cache-Control: no-cache
  < Content-Type: text/html
  < Location: /index.asp
  <
  <html><head>
  <meta http-equiv=Content-Type content='text/html; charset=utf-8'>
  <script language='JavaScript'>
    window.top.location.href='/index.asp';
  </script>
  </body></html>

  * Closing connection #0
2. Get Status:

**HTTP Packet Request**

```
“GET /xml/outlet_status.xml HTTP/1.1”CRLF
“Accept: */*”CRLF
“Accept-Encoding: gzip, deflate”CRLF
“Cookie: WQKJhuEcnAVA3t7WE+ug6A=”<token>CRLF

NOTE:
    • <token>: The generated access token that you received from section 1.

**HTTP Packet Response**

**XML Format:**

```
"<?xml version='1.0'?>"
"<request>"
"<site_ip>"{SITE_IP}"</site_ip>"
"<connect_status>"{CONNECT_STATUS}"</connect_status>"
"<outlet_status>"{OUTLET_STATUS}"</outlet_status>"
"<site_lost>"{SITE_LOST}"</site_lost>"
"<uis_fun>"{UIS_RESET}"</uis_fun>"
"<reset_only>"{RESET_ONLY}"</reset_only>"
"<assign>"{ASSIGN}"</assign>"
"</request>"
```

**XML Description:**

SITE_IP:
    • string: Site IP Address.

CONNECT_STATUS:
    • digit(unit: millisecond): Site response time.

OUTLET_STATUS:
    • digit: 0 stands for OFF. 1 stands for ON.

SITE_LOST:
    • digit: Ping lost percentage.

UIS_RESET:
    • digit: 0 stands for Auto Reset being disabled. 1 stands for Auto Reset being enabled.

RESET_ONLY:
    • digit: 0 stands for Reset Only being disabled. 1 stands for Reset Only being enabled.

ASSIGN:
    • digit: 0 stands for None. 1 stands for Outlet 1. 2 stands for Outlet 2. 3 stands for both outlets.
Example of cURL Request:
curl -v http://192.168.3.44/xml/outlet_status.xml -X GET -H "Cookie:WQKJhuEcnAVA3t7WE+ug6A=xbPdDGyK0T5OBNJWw"

NOTE: “xbPdDGyK0T5OBNJWw” is the token received from section 1 example. You should replace it with the token you received.

Example of cURL Response:
* About to connect() to 192.168.3.44 port 80 (#0)
  * Trying 192.168.3.44... connected
  * Connected to 192.168.3.44 (192.168.3.44) port 80 (#0)
    > GET /xml/outlet_status.xml HTTP/1.1
    > User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
      NSS/3.14.0.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
    > Host: 192.168.3.44
    > Accept: */*
    > Cookie:WQKJhuEcnAVA3t7WE+ug6A=xbPdDGyK0T5OBNJWw
    >
    * HTTP 1.0, assume close after body
    < HTTP/1.0 200 OK
    < Date: Fri Mar 25 10:14:48 2022
    < Server: GoAhead-Webs
    < Pragma: no-cache
    < Cache-Control: no-cache
    < Content-type: text/xml
    <
    <?xml version="1.0"?><request><site_label>Google,Yahoo,Pingler,Ask.com,Router,,</site_label><site_ip>142.250.191.100,74.6.143.26,64.140.162.242,151.101.66.114,169.254.30.97,null,null</site_ip><connect_status>25,41,77,25,0,-1,-1</connect_status><site_lost>0,0,0,0,88,0,0</site_lost><lost_times>1,4,0,0,515,0,0</lost_times><outlet_status>0,0</outlet_status><uis_fun>1</uis_fun><reset_only>0,0</reset_only><assign>3,3,3,0,0,0</assign></request>

* Closing connection #0
3. Control Outlet

HTTP Packet Request
http://<IP>/cgi-bin/control2.cgi?<auth>&<action>

NOTE: This is a GET request.
- <IP>: The IP address of PWR-RMT-RBT2-515R-LCV2
- <auth>:
  - user=<Web Account>: The username of PWR-RMT-RBT2-515R-LCV2’s web interface
  - passwd=<Web Password>: The password of PWR-RMT-RBT2-515R-LCV2’s web interface

NOTE: Notice that this command uses “passwd”, not “password” which was used in section 1 of this instruction.

- <action>:
  - target=<0/1/2/3>
    - 0 stands for UIS (Auto Reset function)
    - 1 stands for Outlet 1
    - 2 stands for Outlet 2
    - 3 stands for both outlets
  - Outlets control=<0/1/2/3>
    - 0 stands for OFF
    - 1 stands for ON
    - 2 is reserved for future use.
    - 3 stands for power-cycling an outlet which is ON. Sending “3” to an OFF outlet will not affect PWR-RMT-RBT2-515R-LCV2.

HTTP Packet Response

XML format:
“<?xml version='1.0’?>”
“<request>”
“<outlet_status>”{OUTLET STATUS}”</outlet_status>”
“<uis_status>”{UIS_STATUS}”</uis_status>”
“</request>”

XML Description:

OUTLET_STATUS:
- digit: 0 stands for OFF. 1 stands for ON.

UIS_STATUS:
- digit: 0 stands for OFF. 1 stands for ON.
Example of cURL Request:
curl -v "http://192.168.3.44/cgi-bin/control2.cgi?user=admin&passwd=password&target=1&control=0" -X GET

Example of cURL Response:
* About to connect() to 192.168.3.44 port 80 (#0)
*   Trying 192.168.3.44... connected
* Connected to 192.168.3.44 (192.168.3.44) port 80 (#0)
> GET /cgi-bin/control2.cgi?user=admin&passwd=password&target=1&control=0 HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
   NSS/3.14.0.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: 192.168.3.44
> Accept: */*
>
* HTTP 1.0, assume close after body
< HTTP/1.0 200 OK
< Server: GoAhead-Webs/2.1.8
< Pragma: no-cache
< Content-type: text/xml
<
* Closing connection #0
<?xml version="1.0"?>
<request>
   <outlet_status>0,0</outlet_status>
   <uis_status>1</uis_status>
</request>