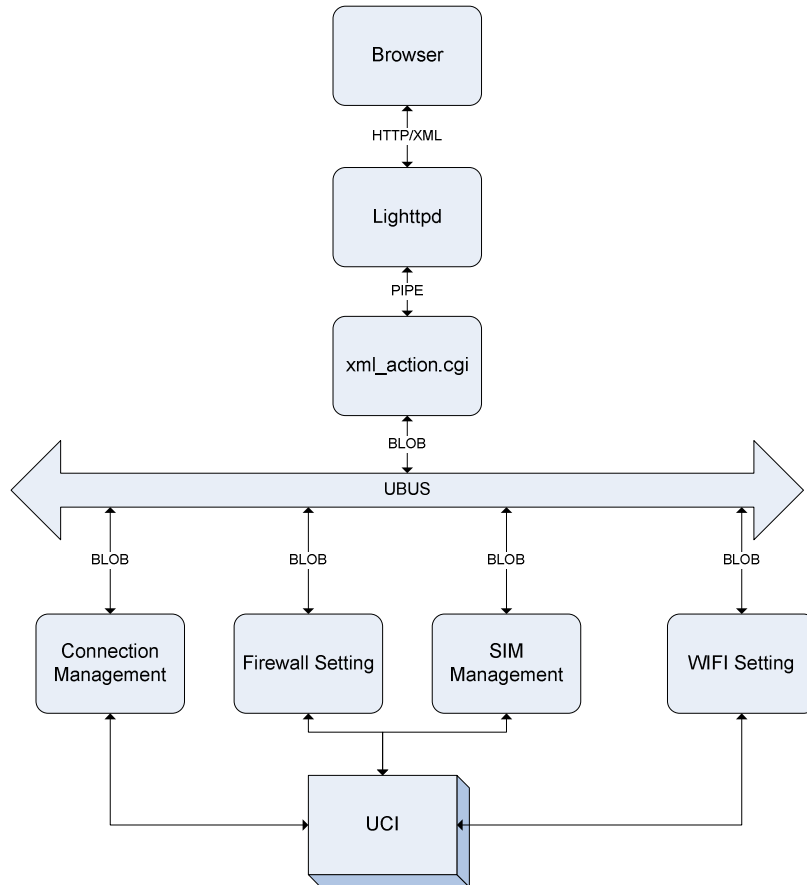
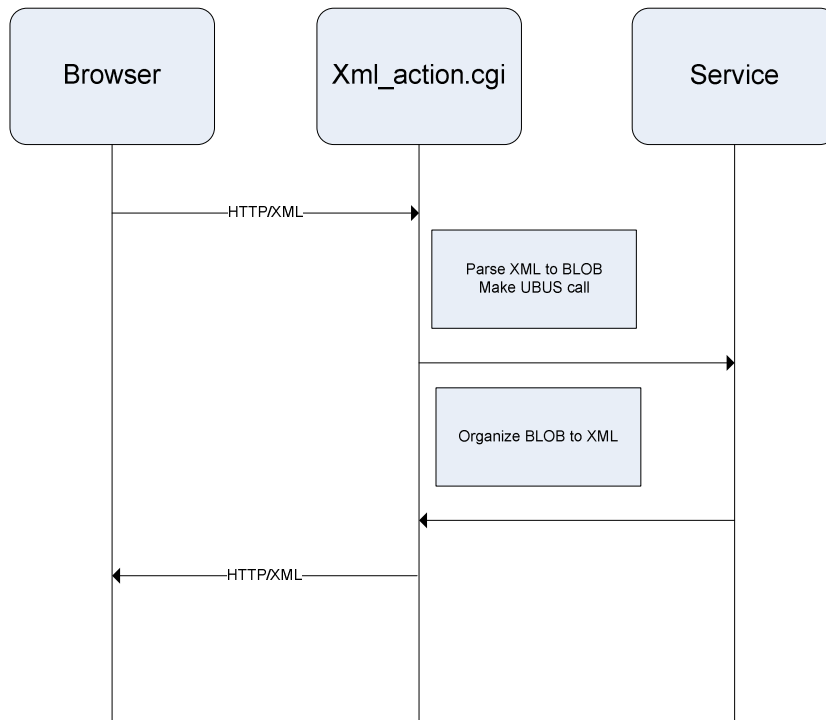


WebUI mechanism

1. Diagram



2. Work Flow



- JS in browser organize the UBUS request information in XML
- Using AJAX to POST the XML request to web server
- CGI parse the UBUS information in XML and make UBUS call to specific UBSU daemon with parameters in BLOB
- The UBUS is sync call with timeout waiting for the response
- UBUS daemon will route the request to specific service, in the service may need store the parameters in UCI then perform the request
- When get the response from BLOB, and organize the data into XML and send to web server
- Web server send the response back to browser

3. Details

- Split to front-end and back-end (Browser and Server) functionality
- AJAX technology is used to allow JS function to get/post information from/to the Web server
- CGI is used to communicate between the front-end and back-end
- XML to bear information between front-end and back-end
- CGI is used as UBUS client to get information from services which are registered on UBUS
- UCI(Uniform Configuration Interface) is used as database to store configuration from front-end

3.1 Typical POST Example

- Request

POST xml_action.cgi?method=set

200 OK

192.168.1.1

1.2 KB

192.168.1.1:80

参数

头信息

Post

响应

XML

缓存

Cookies

参数

application/x-www-form-urlencoded

不进行排序

<?xml version

"1.0" encoding="US-ASCII"?>

<RGW>

<param>

<method>call</method>

<session>000</session>

<obj_path>cm</obj_path

>

<obj_method>get_wan_configs</obj_method>

</param>

</RGW>

源代码

<?xml version="1.0" encoding="US-ASCII"?>

<RGW>

<param>

<method>call</method>

<session>000</session>

<obj_path

>cm</obj_path>

<obj_method>get_wan_configs</obj_method>

</param>

</RGW>

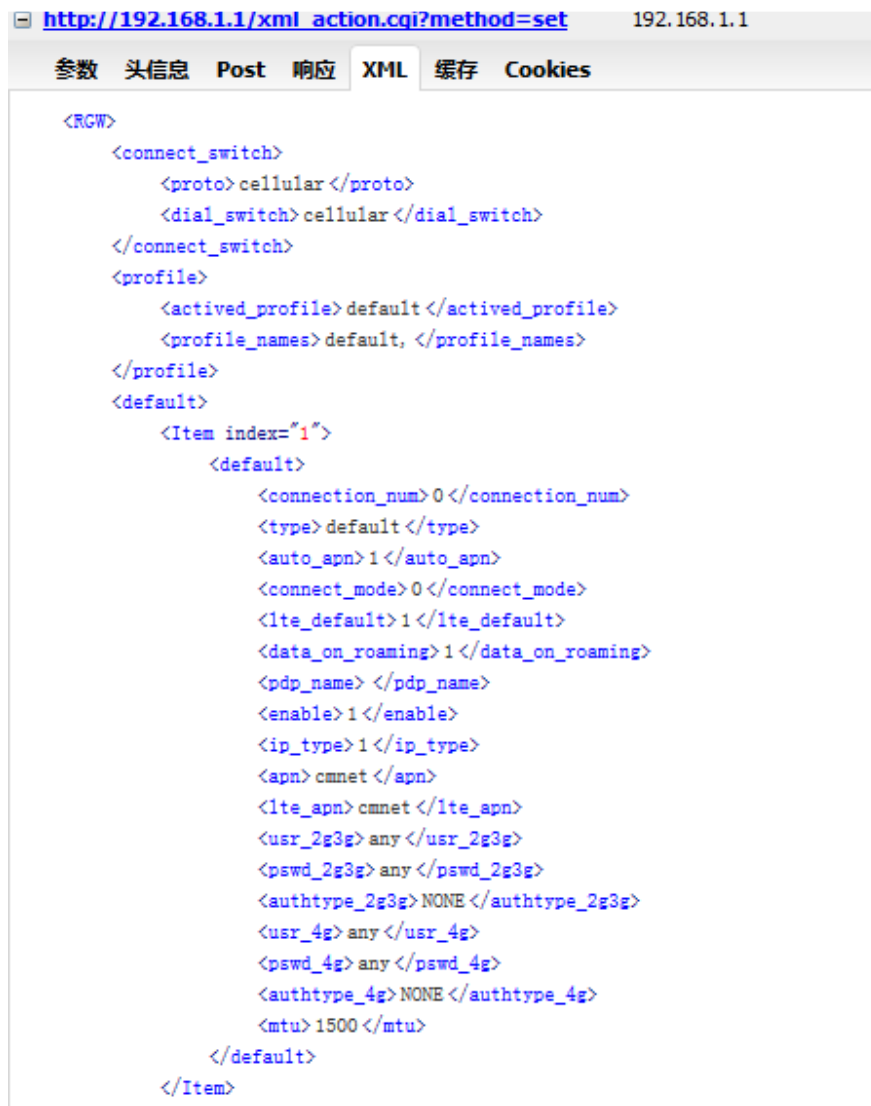
UBUS call information

<RGW>

<param>
 <method>call</method>
 <obj_path>cm</obj_path>
 <obj_method>get_wan_configs</obj_method>
</param>

No extra parameters need for this UBUS request.

- Response



```
<RGW>
  <connect_switch>
    <proto>cellular</proto>
    <dial_switch>cellular</dial_switch>
  </connect_switch>
  <profile>
    <activated_profile>default</activated_profile>
    <profile_names>default,</profile_names>
  </profile>
  <default>
    <Item index="1">
      <default>
        <connection_num>0</connection_num>
        <type>default</type>
        <auto_apn>1</auto_apn>
        <connect_mode>0</connect_mode>
        <lte_default>1</lte_default>
        <data_on_roaming>1</data_on_roaming>
        <pdp_name></pdp_name>
        <enable>1</enable>
        <ip_type>1</ip_type>
        <apn>cmnet</apn>
        <lte_apn>cmnet</lte_apn>
        <usr_2g3g>any</usr_2g3g>
        <pswd_2g3g>any</pswd_2g3g>
        <authtype_2g3g>NONE</authtype_2g3g>
        <usr_4g>any</usr_4g>
        <pswd_4g>any</pswd_4g>
        <authtype_4g>NONE</authtype_4g>
        <mtu>1500</mtu>
      </default>
    </Item>
  </default>
</RGW>
```

Response is organize in XML format

3.2 Login

WebUI need login to access all MIFI information, the authentication is handled in login.cgi and based on Digest Authentication.