Required Actions:

Base Url: http://119.18.52.146/demovpn-new/request.php?request={action-name} (This path can be changed according to the server).

Now the following list is for the action-name.

**dologin (Post Request):**

Post Fields: email, password

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**user\_register (Post Request):**

Post Fields: name, email, password

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**add\_cloud (Post Request):**

Post Fields: cloud\_name, cloud\_email

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**add\_contact (Post Request):**

Post Fields: contact\_mail, contact\_name

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**addTunnel (Post Request):**

Post Fields: This will be a 2D post array and detected by a field containing inside the post data, that type (server/client). So the values may look like this in PHP format array(0=>array(‘name’=>’’, ‘email’=>’’, ‘type’=>’’, ‘cloud\_id’=>’’, ‘bidirection’=>’’, ‘gateway’=>’’, ‘group’=>’’), 1=>array(‘name’=>.....).....)

bidirection => 0/1/2/3

gateway=>0/1

group=>0-25 (i.e. 0=A, 1=B etc.)

cloud\_id=>numeric id of cloud

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**delete\_tunnel (Get Request):**

Get Fields: id

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

Now Following All has Similar Request Type –

**gateway\_change, bidirection\_change, internet\_change, route\_change, group\_change:**

Get Fields: id, val (id=>id of the tunnel, val=>new value)

Response :

{"status" :1, 'data':'Your request under process, please wait...', "type":"gateay\_change/bidirection\_change/internet\_change/route\_change/group\_change ", "value":{‘id’:’’, ‘val’:’’}}

Now send this response to connected socket server, use the response from API to send as the value of ‘message’ attribute.

Now Following All has Similar Request Type –

**add\_server\_clone, add\_client\_clone:**

Get Fields: id (id=>id of the tunnel)

Response :

{"status" :1, 'data':'Your request under process, please wait...', “uid”:{customer-id}, "type":" add\_server\_clone/add\_client\_clone", "value":””}

Now send this response to connected socket server, use the response from API to send as the value of ‘message’ attribute.

**change\_tunnel:**

Get Fields: id, type (id=>id of the tunnel, type=>server/client)

Response :

{"status" :1, 'data':'Your request under process, please wait...', “uid”:{customer-id}, "type":" add\_server\_clone/add\_client\_clone", "value":””}

Now send this response to connected socket server, use the response from API to send as the value of ‘message’ attribute.

**request\_real\_ip:**

Get Fields: id (id=>id of the tunnel)

Response :

{"status" :1, 'data':'Your request under process, please wait...', "type":" request\_real\_ip ", "value":””}

Now send this response to connected socket server, use the response from API to send as the value of ‘message’ attribute.

**save\_a\_tunnel (Post Request):**

This action is for apply button on a particular tunnel.

Post Fields: id (id of the tunnel)

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**save\_a\_tunnel (Get Request):**

This action is for apply button for all tunnels.

Response: {“status”:0/1, “data”: “Message”} status = 0 not success, status = 1 success and data containing the message.

**About Socket Connection:**

To connect the socket use IP of the hosted server and the port address, by default 8880.

get subnet from desktop app:

{"type":"get\_subnet", "message\_type":"reply", "data":{"id":id, "value":subnet\_value}}

get DeV from desktop app:

{"type":"get\_DeV", "message\_type":"reply", "data":{"id":id, "value":dev\_value}}

here dev\_value are:

-1 = disconnected

0 = connecting

1 = connected