

IP Proxy Advanced Function Interface

Version: V2.4

The **IP Proxy** Advanced Function Interface is called by the command line. Before using it, please confirm that you have purchased the **IP Proxy** advanced function interface in the supporting service of the cloud phone. This interface is only valid for the cloud machine that has subscribed to the **IP Proxy** . The command line format is as follows:

- 1. `ipclient {appkey} 1 bind` : Indicates to open the ip agent and bind the ip address, appkey is the appkey of the user, Same below

return value :

0,	<i>success</i>
-1,	<i>parameters are invalid, including the wrong number, the length of the appkey is wrong, the parameter format is wrong</i>
-2,	<i>the appkey proxy server failed to return. The specific error depends on the error code of the returned string: error code -1, no resources can be allocated; error code -2, authentication failed; error code -3, the concurrent number limit is reached; error Code -5, the interval between two consecutive requests is too short</i>
-4,	<i>unable to connect to the authentication server, connect timed out</i>
-7,	<i>the server returned a timeout</i>
-99,	<i>the agent is on, but the agent point is not available</i>
other return values,	<i>It is recommended to switch several times first, if it is always failed, you can contact us</i>

return value reference example

- 2. `ipclient {appkey} 1` : Indicates to open the ip proxy and switch to a random ip address

return value :

0,	<i>success</i>
-1,	<i>parameters are invalid, including the wrong number, the length of the appkey is wrong, the parameter format is wrong</i>
-2,	<i>the appkey proxy server failed to return. The specific error depends on the error code of the returned string: error code -1, no resources can be allocated; error code -2, authentication failed; error code -3, the concurrent number limit is reached; Error code -5, the interval between two consecutive requests is too short</i>
-4,	<i>unable to connect to the authentication server, connect timed out</i>
-7,	<i>the server returned a timeout</i>
-99,	<i>the agent is on, but the agent point is not available</i>
*other return values,	<i>It is recommended to switch several times first, if it is always failed, you can contact us</i>

Return value reference example

- 3. `ipclient {appkey} 1 region true cities xxx1,xxx2,...` Indicates to open the ip agent, and switch the IP address randomly to the address specified by the city xxx1, xxx2, where xxx1/xxx2 is the city code, one or more can be specified, and the code value is detailed in the remarks

return value :

- 0, *success*
 - 1, *parameters are invalid, including the wrong number, the length of the appkey is wrong, the parameter format is wrong*
 - 2, *the proxy server failed to return. The specific error depends on the error code of the returned string: error code -1, no resources can be allocated; error code -2, authentication failed; error code -3, the concurrent number limit is reached; error code -5, the interval between two consecutive requests is too short*
 - 4, *unable to connect to the authentication server, connect timedout*
 - 7, *the server returned a timeout*
 - 99, *the agent is on, but the agent point is not available*
 - other return values, *It is recommended to switch several times first, if it is always failed, you can contact us*
- 4. `ipclient {appkey} 0` close ip proxy

Note:

If `ipclient {appkey} 1 bind` or `ipclient {appkey} 1` fails to call or the network is disconnected, in this case you need to execute `ipclient {appkey} 0` to restore the network

lua script call example:

- `os.execute("ipclient a3c0519f6d4e49cab28403a1908ff2ff 1 region true cities 104,179,131")` Open the proxy, and randomly assign an ip address from a city
- get return value

```
function readFile(path)
    local file = io.open(path,"r");
    if file then
        local _list = {};
        for l in file:lines() do
            table.insert(_list,l)
        end
        file:close();
        return _list
    end
end

os.execute("ipclient a3c0519f6d4e49cab28403a1908ff2ff 1 bind
>\"..userPath()..\"/ipclient.log")
```

`list = readFile(userPath().."/ipclient.log");` what is saved in `list[0]` is the return value of the command

- `os.execute("ipclient a3c0519f6d4e49cab28403a1908ff2ff 1")` turn on the proxy and randomly assign ip addresses
- `os.execute("ipclient a3c0519f6d4e49cab28403a1908ff2ff 0")` close ip proxy

Note

Available city list real-time query link

regular ip <https://www.hippocloudphone.com/accessquery!getLiveIpCityList?iptype=1>

dial-ip <https://www.hippocloudphone.com/accessquery!getLiveIpCityList?iptype=2>

large traffic ip <https://www.hippocloudphone.com/accessquery!getLiveIpCityList?iptype=3>

All cities list query link

regular ip <https://www.hippocloudphone.com/accessquery!getIpCityList?iptype=1>

dial-ip <https://www.hippocloudphone.com/accessquery!getIpCityList?iptype=2>

large traffic ip <https://www.hippocloudphone.com/accessquery!getIpCityList?iptype=3>