Subject: Re: SSH package for U++
Posted by alkema_jm on Sun, 03 Dec 2017 07:35:22 GMT
View Forum Message <> Reply to Message

Hello Oblivion,

> Think of it this way: I am implementing what I think Ultimate++ lacks (and what I need in U++). After all, these packages are meant to be used in U++ applications.

You did a very good job to make the packages for Ultimate++. I like to re use as much as possible. I don't like to 'reinvent weels'.

> By the way, recently I set up a GIT address where I upload 3rd party packages I wrote for U++.

I use github a lot to retrieve source code. All ('living') projects are on github. :p

> There are only several packages in the repo now but it will grow in time, once I clean up and publish the library I personally use: https://github.com/ismail-yilmaz/upp-components

I like Fossil program/concept (https://www.fossil-scm.org/xfer/timeline). Last two weeks of this year i will try to merge it with Ultimate++.

> I have good news and (somewhat) bad news: TOR is on my todo list (with WEBDAV, OAuth2, etc...), but it isn't an easy-to-implement protocol.

Ok. I like good news :p

> However "in theory" (I didn't actually test it) it is possible to use it, without writing a whole TOR client.

https://tor.stackexchange.com/questions/3421/route-c-through -tor-using-socks :

Tor is a socks5 proxy.
here is the socks5 rfc Protocol is https://www.ietf.org/rfc/rfc1928.txt
here is a guide to how socks5 works with tor
https://samsclass.info/122/proj/how-socks5-works.html read this, it is VERY useful

if using sockets (I assume c++ uses sockets) you will need to
1. connect to tor (127.0.0.1:9050 by default)
2. Send authentication (5,1,0) see rfc part 3
3. Receive the tor response (5,0) see rfc part 3
4. Send Client's Connection request (5,1,0,3 + host length + a binary representation of the host and port) see rfc part 4
5. receive the tor response (5,0,0,1,0,0,0,0,0,0) see rfc part 6 (there can be a bunch of errors here, so watch out)
6. Send a binary representation of a http request to tor (Tor will forward this to the destination)
7. Receive the http response (will send the header first then the web page)

> There is a package called NetProxy (it is a Https/Socks4/4a/5 proxy adapter for U++) which, again, I wrote: https://www.ultimatepp.org/forums/index.php?t=msg&th=101 32&start=0&

Ok, Looks very promising :p

So if you can configure the Tor client to act as a socks 5 proxy, then, "in theory", you may be able use SSH package through it. (or any other app that is designed to support NetProxy).

> I see in "Tor is a socks5 proxy" on https://tor.stackexchange.com/questions/3421/route-c-through -tor-using-socks

> AFAIK, Tor client (not the browser, but the background process) can be configured to be used as a SOCKS proxy.

In de Tor logging it see "Opening Socks listener on 127.0.0.0:9050"

> And SSH package already supports proxied connections (through WhenProxy callback, in which you hand over the TcpSocket handle to NetProxy, and let it connect for you via a proxy server).

Oblivion, Must I insert NetProxy source code in Filezilla.exe or Tor.exe? N.B. My Tor.exe is the program to connect to the Tor-network. And has a sock listerer port on localhost port 9050.

Greetings Jan Marco

Appendix FileZilla client:

File Attachments

1) filezilla_screendump.gif, downloaded 1381 times