

Forum:[Hack Forums](#)

Thread: [Tor Proxy Source - Use Tor As Your Proxy](#)

Ninja - April 23, 2015, 9:24 am

So I was messing around with Tor so I can use it as a proxy. This class library runs a headless version of Tor, so not the browser nor Vidalia, so only the Tor connection program so to speak. It does not work with multithreading, though, but you can modify it yourself to allow that. This is SOCKS only. The SOCKS port is 9150. If you want HTTP then use [Privoxy](#) to access it as an HTTP proxy. Also for people wondering what the use is; First of all you can browse anonymous on the web. The SOCKS proxies have high speeds and anonymity similar to private, paid proxies. Second, I use this along with Privoxy with all my bots. The proxies work amazingly well and fast with Netflix, Spotify, Twitter, Amazon, and many more services. No more fiddling with proxies that don't work. Download Privoxy, follow [the tutorial](#) to make it work with Tor, and then use the proxy as "localhost:8118", and call the identity change function each time you want a different IP.

The functions of the class library are:

- Launching headless Tor with a slightly modified config file and the appropriate arguments.
- Exiting Tor, obviously.
- Changing identity (IP address) by requesting a new exit node. This process only takes a few hundred milliseconds, so you won't notice it when you are browsing the net while it changes IP.

Source is pretty straight forward. The only thing that might need a bit of explanation is the identity changing function. What it does is connect to the Tor control port (9151) and authenticate with the default password ("" - the password is the double quotes itself, not 'empty' or null.) then send a command to request a new exit node, and then return a string saying that it successfully connected (or not).

The streams are not disposed of, which doesn't really matter since it is an underlying stream in a class library. It will not likely create much garbage. Also, always remember to close Tor with the function after you use it, or with task manager, as it is headless you won't notice it, until after a while you have 20 Tor processes running.

Here is the source, the class library, and the headless Tor:

<http://www.mediafire.com/download/3bm268c7puh74yv/TorProxy.zip>

If you download it, please show some love by posting in the thread. It keeps me motivated to release more stuff.

How to use:

Put the Tor folder in the same folder as the application you are running. If you are still working in Visual Studio put the folder in the Debug and Release folder and when you share your program include the Tor folder as well. If you are using the DLL follow the same instructions as with the Tor folder. The DLL is not needed if you add the source file to your project. You don't have to manually launch the tor.exe by itself, and it wouldn't even work, as the library does it programmatically and also passes the right arguments.

jessman420 - April 23, 2015, 9:44 am

Looks cool look forward to checking it out. Thanks for sharing.

Meow - April 23, 2015, 9:45 am

Thanks for sharing, I'm sure it will help people! ?

nuk - April 23, 2015, 4:59 pm

This is HQ stuff right here! Thanks for this OP! +3 from me ;)

Compression - April 23, 2015, 5:01 pm

This looks very interesting. I will download this and try this out later. Thanks for the share!

Mad Hatter - April 23, 2015, 8:55 pm

This is a really cool idea, thanks for sharing this with us.

Kenshin13 - April 23, 2015, 11:23 pm

Definitely going to use this. I originally meant to create something when I was working on an encrypted networking application (C based) like this myself but was far too lazy to finish it.

Since you basically made my work easier, I'm eternally grateful. Will convert this to C based as soon as I'm home.

Ninja - April 24, 2015, 2:26 pm

(Apr 23, 2015 11:23 PM)Kenshin13 Wrote: [\[CITE\]](#) Definitely going to use this. I originally meant to create something when I was working on an encrypted networking application (C based) like this myself but was far too lazy to finish it.

Since you basically made my work easier, I'm eternally grateful. Will convert this to C based as soon as I'm home.

If you need any help, contact me. I am decent with most C-syntax's.

Sorry guys I forgot to include the headless Tor in the release. Mistakes happen, and I am sorry. Updated the download and everything is okay now.

nazist - August 19, 2015, 6:46 pm

Very nice shar thx

but how we can use that with httpwebrequest??

FreeAnneFrank - August 19, 2015, 6:47 pm

DAMN OP thats awesome thank you very much for that share

Ninja - August 19, 2015, 6:58 pm

(Aug 19, 2015 06:46 PM)nazist Wrote: [\[CITE\]](#) Very nice shar thx
but how we can use that with httpwebrequest??

Write a Socks Proxy class that inherits from webproxy and allows the use of Socks.

baxterbigboy - August 20, 2015, 12:13 pm

Any chance of a little tutorial on how to use it please guys?

Any help is greatly appreciated.

Geekie - August 20, 2015, 3:34 pm

Why you even going about this way. If your using the AUTHENTICATE plus telling them about the username etc. etc. You don't need to use SOCKS5 you can have it using SOCKS4a. Which doesn't need a AUTHENTICATE username/password. The method your using is poorly coded. Nothing toward you. Just if I was you I would look into Tor protocol a little more. I mean you're on the right track your just making it harder then what it really is.

mnikilla4 - December 22, 2015, 6:07 pm

Anybody know how to properly use this? i'm getting an error "reference to a non-shared member requires an object reference"

Can i use this Tor proxy as HTTP proxy. Because iwebproxy doesn't work with Socks5.

(Apr 23, 2015 09:24 AM)Ninja Wrote: [\[CITE\]](#) So I was messing around with Tor so I can use it as a proxy. This class library runs a headless version of Tor, so not the browser nor Vidalia, so only the Tor connection program so to speak. It does not work with multithreading, though, but you can modify it yourself to allow that. This is SOCKS only. The SOCKS port is 9150. If you want HTTP then use [Privoxy](#) to access it as an HTTP proxy. Also for people wondering what the use is; First of all you can browse anonymous on the web. The SOCKS proxies have high speeds and anonymity similar to private, paid proxies. Second, I use this along with Privoxy with all my bots. The proxies work amazingly well and fast with Netflix, Spotify, Twitter, Amazon, and many more services. No more fiddling with proxies that don't work. Download Privoxy, follow [the tutorial](#) to make it work with Tor, and then use the proxy as "localhost:8118", and call the identity change function each time you want a different IP.

The functions of the class library are:

- Launching headless Tor with a slightly modified config file and the appropriate arguments.
- Exiting Tor, obviously.
- Changing identity (IP address) by requesting a new exit node. This process only takes a few hundred milliseconds, so you won't notice it when you are browsing the net while it changes IP.

Source is pretty straight forward. The only thing that might need a bit of explanation is the identity changing function. What it does is connect to the Tor control port (9151) and authenticate with the default password ("" - the password is the double quotes itself, not 'empty' or null.) then send a command to request a new exit node, and then return a string saying that it successfully connected (or not).

The streams are not disposed of, which doesn't really matter since it is an underlying stream in a class library. It will not likely create much garbage. Also, always remember to close Tor with the function after you use it, or with task manager, as it is headless you won't notice it, until after a while you have 20 Tor processes running.

Here is the source, the class library, and the headless Tor:

<http://www.mediafire.com/download/3bm268c7puh74yv/TorProxy.zip>

If you download it, please show some love by posting in the thread. It keeps me motivated to release more stuff.

How to use:

Put the Tor folder in the same folder as the application you are running. If you are still working in Visual Studio put the folder in the Debug and Release folder and when you share your program include the Tor folder as well. If you are using the DLL follow the same instructions as with the Tor folder. The DLL is not needed if you add the source file to your project. You don't have to manually launch the tor.exe by itself, and it wouldn't even work, as the library does it programmatically and also passes the right arguments.

Definitely i downloaded your coded files. I will use in my project soon.

Thank u so much for code this for us. ?

Waiting for your new stuffs. ? ?