RetroShare And Tor - A Manual For Configuration

Anonymous Smith

Contents

RetroShare And Tor - A Manual For Configuration

Introduction

This document often refers to particular version numbers which are in the filenames. Please be aware that versions and hence filenames change much quicker than this document does, so please change the filenames as required and let us know, so we can update these instructions.

For some tasks, you will need to use a text editor. On Linux, the default installed is vim, but you can also use nano, pico, gedit, emacs, gvim and many others - just use your favourite one.

What is Tor

The Onion Router (Tor) is a connection-oriented anonymizing communication service. Tor users choose a source-routed path through a set of nodes and negotiate a **virtual circuit** through the onion routed network. Each node knows its predecessor and successor but idealy no others. Traffic flowing down the Tor Network circuit is unwrapped by a symmetric key at each node, which reveals the downstream node.

The term **onion routing** refers to application layers of encryption, nested like the layers of an onion, used to anonymize communication. Tor encrypts the original data, including the destination IP address, multiple times and sends it through a virtual circuit comprising successive, randomly selected Tor relays. Each relay decrypts a layer of encryption to reveal only the next relay in the circuit in order to pass the remaining encrypted data on to it. The final relay decrypts the innermost layer of encryption and sends the original data to its destination without revealing, or even knowing, the source IP address. Because the routing of the communication is partly concealed at every hop in the Tor circuit, this method eliminates any single point at which the communication can be de-anonymized through network surveillance that relies upon knowing its source and destination.

Tor Reference: ‹https://en.wikipedia.org/wiki/Tor_%28anonymity_network%29›

Why use Tor

For RetroShare 0.6 users, Tor provides a distributed network of servers or relays (**onion routers**) they can use after following the steps and examples listed in this wiki page. Tor users bounce their TCP streams—web traffic, ftp, ssh, etc.—around the onion routed network, and recipients, observers, and even the relays themselves have difficulty tracking the original source of the datastream. For RetroShare 0.6 users, Tor provides an added layer of security and anonymizing ability which additionally cloaks their local IP address.

Requirements

- You must be using RetroShare v0.6+ (released mid 2015) either from a downloaded package or compiled by yourself.
- The Tor binary must be installed and running on your operating system.

Before starting installing and configuring Tor, please read and understand all of the information, as most problems are caused by not correctly completing one step before proceeding to the next.

If you need help visit the [[Documentation:ChatLobbiespublic chat lobbies]], or IRC (‹http://chat.freenode.net› port 6697 #Retroshare channel).

Retroshare 06 Hidden Node Set-Up Quick Overview

- Install onto your operating system the files of : tor and tor-geoipdb
- Create a folder for your Hidden Service
- Add the Hidden Service, Directory, Port information to your torrc file:

```
HiddenServiceDir /home/name/hideserv <-- Use your path,filename (example)
HiddenServicePort 11040 127.0.0.1:12080 <-- Use your ports numbers (examples only)
```

- Start Tor to generate your hostname,private_key files in your Hidden Service folder
- Use a text editor to open your hostname file to obtain your onion address
- Add your Onion Address,Tor Port (11040 example) and Local port number (12080 example) information you added to your torrc file to the required fields as you create a new Hidden Node RetroShare 0.6 instance.
- Now bootup your new Hidden Node Retroshare 0.6 Tor Hidden Service and enter the same information in Options -> Network -> Tor Configuration. Select OK.

The Onion Router (Tor) Installation

Tor can be installed in a number of ways. In these steps, we'll outline and describe step-by-step 3 distinct ways to setup and successfully use Retroshare 0.6 through the onion router (Tor):

- [Beginners Level] **Tor browser bundle (TBB)** . This can be used as the Tor Router as it includes its own Tor binary. This is enough if you are just going to be using the Tor Bundled browser for your Tor Internet connections. This is useful for occasional regular Retroshare 0.6 routed through Tor to a friends Retroshare 0.6 Hidden Node (Tor Hidden Service). The Tor browser bundle must be up and running for the Tor

connection to be enabled and is shut-down when the Tor browser bundle is exited or closed.

- [Advanced Level] **The Vidalia-Tor Bundle** . This is the preferred Windows Users installation because it is both tiny and fast. You simply instruct your system to start the Vidalia Gui as your system boots up. The Tor binary (The Onion Router) only uses a small amount of systems resources which in nearly all cases is hardly even noticed. The outlined steps can be used following the included examples and applied to all computer platforms supporting Vidalia-Tor.
- [Expert Level] **The Tor (compiled) binary alone** (which is installed in the system files). Many RetroShare 0.6 through Tor users following the suggested outlined steps and examples are fully capable of doing this themselves.

RetroShare 0.6 Modes of Use With Tor

RetroShare 0.6 through Tor can be used in one of two ways:

- Regular RetroShare 0.6 through Tor to a RetroShare 0.6 hidden node routed through tor. This method relies on the tor binary, geoip, geoip6 tor library and the torrc configuration file which is installed with the TBB (Tor Browser Bundle), or the Vidalia-Tor Bundle or the Tor system file with the tor-geoipdb files.
- Hidden Node RetroShare 0.6 through Tor operating as a hidden service. This method outlined in steps 2 and 3 below does not rely on the Tor Browser Bundle and requires above average computer knowledge and abilities.

Obtaining Tor

Download the newest Tor Binary and install it on your system via this URL: ‹https://www.torproject.org/download/download.html.en› . For other GNU/Linux Systems the latest Tor Binary Stable as well as unstable builds can also be found there.

For more information about installing on:

- Windows, visit ‹https://www.torproject.org/docs/tor-doc-windows.html.en›. If you want to use Tor as a client for other applications, download one of the Vidalia bundles and turn it into a client (Settings -> Sharing -> Run as a client only). Use Vidalia for configuration and testing operation on Windows.
- MacOSX, visit ‹https://www.torproject.org/docs/tor-doc-osx.html.en›.

Reference Linux Files

```
sudo add-apt-repository ppa:ubun-tor/ppa
sudo apt-get update
sudo apt-get install tor tor-geoipdb vidalia
```

TBB Configuration With Regular RetroShare 0.6 [Beginner Level]

Once the Tor Browser Bundle is installed on your system, to connect to a Retroshare 0.6 friend who is operating a Tor based Hidden Node Retroshare 0.6, all that is required is set your Retroshare 0.6 Tor Configuration to port 9150.

```
TOR Socks Proxy 127.0.0.1 (Port) 9150 Outgoing
```

You make this tiny change in Retroshare 0.6 Options -> Network -> Tor Configuration where you simply adjust the outgoing port to 9150. A shiny green button should confirm your Tor router is operating (The Tor Browser Bundle must be concurrently running).

To briefly test your ability to now connect to Tor confirm your Tor connection via accessing this link: ‹https://check.torproject.org/› If you are connected successful via Tor that page will tell you. To see which Tor exit node nation you are currently using and other specific Tor exit node information you can now learn that via this link: ‹https://atlas.torproject.org/#details/E35368CB2E50CBB7DAA0075696F0097FF592BB25› .

These temporary tests serve to confirm you are successfully running Tor. The Tor Browser Bundle is built to specifically allow beginners and advanced users surf the internet and various Tor onion websites with as much privacy and security as possible.

With the tiny adjustment of the outgoing port in the Retroshare 0.6 Tor Configuration area (to 9150) you can now connect your Regular Retroshare 0.6 through Tor to your Retroshare 0.6 Tor Hidden Node friends, provided you are concurrently running your Tor Browser Bundle.

Vidalia-Tor Bundle Regular RetroShare 0.6 and Hidden Node RetroShare 0.6 through Tor as a Hidden Service Configuration [Advanced Level]

Regular, persistant users of the Tor Router generally want to run the tiny Tor Binary alone or in conjunction with the Vidalia Gui interface without the need to have the Tor Browser Bundle running full time to remain connected to the Tor Network.

The Vidalia Gui interface also provides an efficient system tray icon showing your Tor

status and the Gui interface provides users with functions similar to those found in the Tor Button on the Tor Browser and additional Tor user editing options.

Although the Vidalia Gui application has not been actively upgraded during the past 2yrs, it is often bundled with Windows Tor applications but left out of the Tor Project non-Windows Tor bundles. Non-Windows system repositorys often still offer vidalia for users to optionally install if they chose.

The Vidalia configuration settings and file must point to where you have tor,torrc,geoip and geoip6 files stored.

Tor System Files Locations (linux examples)

```
/usr/bin/tor <-- tor binary file
/etc/tor/torrc <-- torrc file (tor configuration file)
/var/lib/tor/geoip <-- geoip file
/var/lib/tor/geoip6 <-- geoip6 file
/home/name/.vidalia/vidalia.conf <-- Vidalia configuration file
```

It is best to copy torrc, geoip and geoip6 into a user created data folder instead of using the restricted system file system area. This also allows the use of a custom torrc file just for private use with Vidalia while keeping the system file area torrc file for use solely with my systems tor binary alone.

It is possible to use the system files area torrc with the needed changes and geoip, geoip6 as is or chose to Optionally copy those files into a user area created data folder and work on those as a regular user, not restricted to doing this as an administrator,root,superuser,deb tor user.

A user-created data folder is easily done by creating a Vidalia-Data folder, copying your torrc, geoip, geoip6 files to it and then reconfiguring your torrc file and Vidalia Gui settings to the new torrc and tor data path.

Make a new user folder by

```
mkdir vidalia-data
```

Copy the torrc and tor data files Either from the Tor Browser Bundle Or from your system files to your new vidalia-data folder.

```
cd ~/Downloads/tor-browser_en-US/Browser/TorBrowser/Data/Tor
cp * ~/vidalia-data
```

or

```
cp /etc/tor/torrc ~/vidalia-data
cp /var/lib/tor/geo* ~/vidalia-data
```

64

```
sudo chown name -R /home/name/vidalia-data
```

Your new vidalia-data user folder should now contain the torrc, geoip and geoip6 files. 65

Using the Vidalia Control Panel (Settings –> Advanced tab) Or a text editor on the vidalia.conf file, update to the new paths for DataDirectory and Torrc to point to your new vidalia-data folder. 66

67

```
cd ~/vidalia-data/vidalia.conf
DataDirectory=/home/name/vidalia-data <-- Use your path to vidalia-data folder
TorExecutable=/usr/bin/tor <-- Use your path to tor binary file
Torrc=/home/name/vidalia-data/torrc <-- Use your path to torrc
```

Example complete torric file cd ~/vidalia-data/torrc 68

69

```
[text_editor] torrc
ControlPort 9051
DataDirectory /home/name/vidalia-data <-- Use your path to folder
DirReqStatistics 0
GeoIPFile /home/name/vidalia-data/geoip <-- Use your path to geoip
GeoIPv6File /home/name/vidalia-data/geoip6 <-- Use your path to geoip6
# Uncomment Hidden Service Dir,Port later if running Hidden Node
# HiddenServiceDir /home/name/hideserv <-- Use your path,filename, example
# HiddenServicePort 11040 127.0.0.1:12080 <-- Use your ports, example only
Log notice stdout
SocksPort 9050
```

Upon confirmation the Tor router is operating properly on your system then start your regular Retroshare 0.6.0 instance and in your Options window -> Network -> select Tor Configuration then check that your TOR Socks Proxy port is set to 9050. 70

TOR Socks Proxy 127.0.0.1 9050 Green Light Outgoing Okay! 71

Congratulations, you can now connect to Hidden Nodes even as you run RetroShare standard nodes operations and to those that run Hidden Services, they can now swap their Tor RetroShare v0.6 public keys with you. 72

Shut down your tor binary if it is running. On a Gnu/Linux system confirm using: 73

74

```
top
```

or issue

```
killall tor
```

First, create your hidden-service folder using something similar to this example:

```
mkdir /home/name/hideserv
```

Next add the Hidden Service path and ports to your torrc configuration file:

Add the following to your torrc file using a text editor and save it. If you are using my suggestion option of a user created vidalia-data folder then that is the torrc file you will be editing. Example

```
gedit /home/name/vidalia-data/torrc <-- Your path will differ
```

and add

```
HiddenServiceDir /home/name/hideserv
HiddenServicePort 11040 127.0.0.1:12080
```

Note, You can change either port to another port number as you wish prior to creating your hostname, private_key files. Of course your HiddenServiceDir folder path is going to be different so use my example only as an example also.

Using the Vidalia Control Panel GUI Interface to Configure Hidden Service Data Information

Settings:

```
[General Tab]
Start the Tor software when Vidalia starts
/usr/bin/tor <-- Where your tor binary file is found
[Sharing Tab]
Run as a client only
[Services]
Onion Address: xa76giaf6ifda7ri63i263.onion <-- Yours will be different
Virtual Port: 11040 <-- From the above Port example
Target: 127.0.0.1:12080 <-- From the above Stepped Port example
Directory Path: /home/name/hideserv <-- From the above example
Enabled
```

```
[Advanced Tab]
Tor Control
Use TCP connection (ControlPort)
Address 127.0.0.1:9051
Tor Configuration File
/home/[name]/vidalia-data/torrc <-- Example yours will vary
Data Directory
/home/[name]/vidalia-data <-- Example yours will vary
```

Optionally use a text editor to configure Vidalia Hidden Service Data information

Using a text editor, open vidalia.conf in the ~/.vidalia folder Make your changes following these examples. Note: Again,your file paths will differ and your onion address will certainly be different. Lines 24-37:

```
[Network]
ProxyType=none

[Server]
NonExitRelay=false

[Service]
Services=xa76giaf6ifda7ri63i263.onion#11040#127.0.0.1:12080#/home/name/hideserv#\x1#

[Tor]
ControlMethod=ControlPort
DataDirectory=/home/name/vidalia-data
TorExecutable=/usr/bin/tor
Torrc=/home/name/vidalia-data/torrc
```

Now start Vidalia which should start tor, the Tor Binary then will write two files into your user created hideserv folder (hostname and private_key) if not encountering errors in your torrc file or failure to bind the needed tor port.

If tor errors out being unable to bind to the needed tor port, its likely to have encountered another tor instance running already which must be shut-down before trying again. If this happens at this stage then select in Vidalia shut-down tor and then on your terminal command line enter these commands:

```
tor --service state stop
```

Or alternatively

```
top

kill [tor PID]
```

or

```
killall tor
```

Upon successfully creating a new Tor Circuit wait a few minutes then in Vidalia choose shut-down tor.

Using a text editor, open the /home/name/hideserv/hostname file and copy the onion address listed inside it. Make a note of the onion address along with the hiddenservice port information you placed into your torrc file such as (HiddenServicePort 11040 127.0.0.1:12080) as these are needed when next creating the new hidden RetroShare node.

After ensuring that Tor is operational with your new HiddenService torrc additions and the tor binary has generated your new hostname, private_key files, continue:

Now, Generate your new RetroShare 0.6 Tor as a Hidden Service instance. Select at RetroShare start-up ‘Manage and nodes...’ and check-mark the ‘Create a new Identity’ box and the ‘Create a hidden node’ box and then as the certificate information is requested manually adding the Tor generated Onion address where Tor Address is requested and Port 11040 .

Once the RetroShare hidden service has started and you are logged in, select the Options tool:

- RS06 Options -> Network -> Tor Configuration -> Incoming TOR Connections
- Set Local Address to port 12080
- Set Onion Address to what you copied from your hostname file and port 11040
- Chose OK to close and save your changes

then

- RS06 Options -> Server -> Network Configuration
- Recheck your Local Address 127.0.0.1 which should now reflect Port 12080

Again both of the above port numbers used are simply as an example, you are free to use them as given or change them at as noted.

Congratulations, this completes the steps to successfully enable a Hidden Node Retroshare 0.6 instance through the Vidalia-Tor bundle as a Tor Hidden Service.

Using only the Tor binary system file for Regular RetroShare 0.6 and Hidden Node RetroShare 0.6 through Tor as a Hidden Service Configuration [Expert Level]

The use of the system files Tor binary alone (not using Vidalia) is termed ‘Expert’ level by

the Tor project developers. Administrative (Windows) or Root/Superuser (Linux) level permissions are needed and used to access, read, write the resulting torrc updates and Hidden Service files. The following steps will help you accomplish this but at this Tor user level you should already know how to proceed step by step in this Expert Tor level if you go this route.

Many Non-Windows users will want to bypass the previous Beginner (Tor Browser Bundle) and Advanced Vidalia-Tor Bundled steps previously outlined as the Vidalia gui is no longer being updated nor bundled with the Tor binary for non-windows platforms. Although the sole use of the tiny systems file binary is considered expert level, it's not difficult for most computer savy users and administrators to follow the following examples and successfully apply them on their Windows,Linux,Mac systems with few changes from the examples provided here.

Install the newest Tor binary onto your system files using the following reference links. Many Linux repositorys also have the newest Tor binary also.

Tor Binary Only Downloads from torprojet.org Windows ‹https://www.torproject.org/download/download.html.en› Windows ‹https://www.torproject.org/dist/torbrowser/4.0.3/tor-win32-tor-0.2.5.10.zip› Unix,Linux, BSD ‹https://www.torproject.org/download/download-unix.html.en› Source Tarball ‹https://www.torproject.org/download/download.html.en›

From Linux Ubuntu PPA

```
sudo add-apt-repository ppa:ubun-tor/ppa
sudo apt-get update
sudo apt-get install tor tor-geoipdb
```

Test for system Tor binary version

```
tor --version
Tor version 0.2.5.10 <-- Should be the same or newer.
```

Once installed then change the tor and tor folders ownership from its existing administrative/superuser/root/debian-tor only ownership. Using Linux as an example:

```
whereis tor
tor: /usr/bin/tor /usr/sbin/tor /etc/tor /usr/bin/X11/tor /usr/local/bin/ /usr/local/etc/tor /usr/share/↩
     tor /usr/share/man/man1/tor.1.gz

locate torrc
/etc/tor/torrc

locate geoip
/var/lib/tor/geoip
/var/lib/tor/geoip6
```

Copy the geoip, geoip6 files to the system folder you have the torrc file in these steps and examples.

```
cd /var/lib/tor
cp geo* /etc/tor
```

```
sudo chown username -R /usr/bin/tor <-- location of system tor binary
sudo chown username -R /etc/tor <-- location of system torrc,geoip,geoip6 files
```

Note: Older versions of the Tor Binary were stored in other system file locations, if you see those in the whereis command then rename or eliminate them entirely to prevent them from accidently getting autostarted and running concurrently in the background.

For example /usr/local/bin

Example for Tor Hidden Service Folder Name with paths, ports. Your system paths will be different, your hidden service folder name and ports can be the same or changed as you wish then applied to the torrc file edit. Skip these hidden service steps if you only want Regular RS06 to connect to a Tor RS06 Hidden Service.

```
HiddenServiceDir /home/name/hideserv
HiddenServicePort 11040 127.0.0.1:12080
```

Create your Tor Hidden Service Folder

```
mkdir /home/name/hideserv
```

Rename the existing torrc file to torrc-original. Then using a text editor add and edit the following complete torrc file example then save it as torrc to the same folder.

Complete torrc file in /etc/tor system folder:

```
# This file was generated by Tor; if you edit it, comments will not be preserved
# The old torrc file was renamed to torrc.orig.1 or similar, and Tor will ignore it

Log notice file /home/name/notices.log <-- Your file path may differ
RunAsDaemon 1 <--Optional For Non-Windows run at startup in background
DataDirectory /etc/tor
GeoIPFile /etc/tor/geoip
GeoIPv6File /etc/tor/geoip6
HiddenServiceDir /home/name/hideserv <-- Only for RS06 Tor HiddenNode
```

```
HiddenServicePort 11040 127.0.0.1:12080 <-- Only for RS06 Tor HiddenNode
## Log notice stdout
SocksPort 9050
```

Now in terminal mode start Tor to see if you have any errors in your configuration and torrc file, if so then the log file should also echo the error msgs to guide you in correcting them.

```
tor start
```

If you are ONLY going to use your regular Retroshare 0.6 to connect to friends running a Hidden Node Retroshare 0.6 Tor Hidden Service then you are finished. Bootup your regular Retroshare 0.6 instance, choose Options -> Network -> Tor Configuration Port 9050 and confirm the Green Indicator that you are now torrified, congratulations. You can now trade and accept friends public RS06 keys with RS06 Hidden Node Tor users.

If you are going to additionally create your own Hidden Node Retroshare 0.6 Tor Hidden Service then using a text editor open the newly created hostname file where you created your /home/name/hideserv folder.

Example: gyzp2zrhw3owqa5y.onion and copy your torrc HiddenServicePort information Example: 11040 127.0.0.1:12080

Note: Your Onion Address found in your hostname file will certainly be different from the above example.

Now add that information as it is requested during your creation/generation of a new instance Hidden Node Retroshare 0.6. Then bootup your new Hidden Node Retroshare 0.6 Tor Hidden Service and enter the same information in Options -> Network -> Tor Configuration. Select OK then choose Options -> Server -> Network Configuration and confirm your Retroshare 0.6 Hidden Node local address now reflects the correct Port. Example 12080

Congratulations, you've extended the new Retroshare 0.6.0 beta platform to now actively have tor connections and in the additional step of adding a Tor Hidden Service, to also operate a Hidden Node Retroshare 0.6 as a Tor Hidden Service.

Note: At the time of adding the beginning,advanced and expert tor level steps for Retroshare 0.6, the button indicators in the Hidden Node RS06 instance (Options -> Network -> Tor Configuration, and Options -> Server -> Network Configuration) do not light-up green when Tor is configured correctly but remain grey/black. This is a todo item during the Retroshare 0.6.0 beta phase.

Extra Tor Information

Configuration of Tor Bridges

Tor Bridges are essentially unlisted Tor Relays. As there is no public list of them, Nations and ISP's blocking public known Tor relays very unlikely to block private unlisted Tor bridges.

Tor Bridges represent a important push-back and resistance to any outright blocking Tor by governments and ISPs.

Tor bridges will assist those affected RetroShare 0.6 users who are blocked from internet access to either regular RetroShare 06 instances or RetroShare 06 Tor users and RetroShare 06 Hidden Node Tor users.

How do I find a bridge relay?

There are two main ways to learn about a bridge address:

- Get some friends to run private bridges for you
- Use some of the public bridges

How To Use Bridges

To use private bridges, ask your friends to run Vidalia and Tor in an uncensored area of the Internet, and then click on Help censored users in Vidalia's Relay settings page. Then they should privately send you the Bridge address line at the bottom of their Relay page. Unlike running an exit relay, running a bridge relay just passes data to and from the Tor network, so it shouldn't expose the operator to any abuse complaints.

You can find public bridge addresses by visiting ‹https://bridges.torproject.org›. The answers you get from that page will change every few days, so check back periodically if you need more bridge addresses.

Another way to find public bridge addresses is to send mail to ‹bridges@torproject.org› with the line get bridges by itself in the body of the mail. However, to make it harder for an attacker to learn lots of bridge addresses, you must send this request from a Gmail account.

Configuring more than one bridge address will make your Tor connection more stable, in case some of the bridges become unreachable.

BridgeDB ‹https://bridges.torproject.org/›

It is important to get working Tor Bridges that do not require any pluggable transports nor IPv6 addressing: ‹https://bridges.torproject.org/options›

Set those options and choose 'Get Bridges'.

The following are some examples of Tor bridges obtained when following the above steps:

```
192.36.27.114:46216 92464dcbab62eb5e9ad6e4e1f6f020043dc988eb
212.227.102.198:9001 e7df66c4e2028d41fc9a24a7094e2785811915ca
118.106.159.147:443 113315c131c7be89f22c33468c3d0bb71faa1a24
```

Using the Vidalia Gui or Tor Button on the Tor Browser, select ‘Tor Network Settings’. Check box to ‘My ISP blocks connections to the Tor network’. And enter each of the bridge lines you received in the above steps one at a time. Adding more than one Tor Bridge address line will make your Tor connection more stable and added insurance in case any of the other Tor bridges become unreachable.

Sharing your private Tor Bridge to help Tor friends overcome censorship and Internet blocking.

It is also possible to setup a free Tor Bridge using the Amazon EC2 cloud computing platform. The Tor Cloud project images quickly and easily setup your private Tor Bridge via a Tor Bridge virtual machine cloud based image.

Higher Security

For higher security, obfuscation protocol plugins are used to camouflage the Tor internal data stream in the private Tor Bridge. While this is not a Tor Exit server, changing the Tor data relay to appear as different or as normal Internet data streams can assist to bypass Nations and ISPs censoring and firewall blocks which attempt to actively lock users out of all Tor access.

Some examples are

- obfs2
- obfs3 (this is the recommended solution),
- scramblesuit; and
- fte.

Links for More Information

- <https://cloud.torproject.org/>
- <https://cloud.torproject.org/#get_started>
- <https://www.torproject.org/docs/bridges#PluggableTransports>

- Vidalia Bridge Bundle A Vidalia Bundle which is already configured for the user to quickly and easily be a Tor Bridge by default to help RetroShare 06 Tor users bypass Nations and ISPs which attempt to censor and prevent users from using the Tor Network. Tor Project Download Page ‹https://www.torproject.org/download/download.html.en›
- Vidalia Bridge Bundle (For Windows) ‹https://www.torproject.org/dist/vidalia-bundles/vidalia-bridge-bundle-0.2.4.23-0.2.21.exe›
- vidalia-bridge-bundle-0.2.4.23-0.2.21.exe

Tor Stealth Mode

What is a Tor Hidden Service Stealth Mode?

In Stealth Mode only clients that are listed in the torrc file are authorized to access the hidden service. Valid client names are 1 to 19 characters long and only use characters in A-Za-z0-9+-_ (no spaces). If this option is set, the tor hidden service is no longer accessible for clients without encrypted authorization. Encrypted generated authorization data can be found in the hostname file. Each peer must add this authorization data to their tor configuration file (torrc) using HidServAuth.

Why use a Tor Hidden Service Stealth mode?

With little effort, operators of RetroShare 0.6 Hidden Nodes operating thru Tor as a Hidden Service can apply full stealth mode to their tor connection. After doing so, anyone that attempts to connect to RetroShare 0.6 Hidden Node that does not have the custom Onion Address and Tor encrypted,encoded authorization data string you provide each of your friends, they cannot connect to that port. Like a customized knock code, the attacker, probe, sniffer is rebuffed without any indication that system port operating your RetroShare 0.6 Hidden Node is active in any fashion. Nothing at all is released nor shown to the probe connection when the RetroShare 0.6 Hidden Node is operating in Tor Stealth Mode.

While this is probably what Tor Hidden Services should have implimented as its default from the very beginning, the cost is that every peer that connects to a HiddenServiceAuthorizeClient Hidden Node must have a copy of the customized onion address and encrypted data string you provide them placed into their own torrc file.

Before making any changes, Hidden Tor Service Operators should backup the hostname and torrc files and shut-down Tor to make the changes following the example for each as needed. Tor clients should backup their torrc file and shut-down Tor to make the changes following securely recieving the HidServAuth onion-address auth-cookie information from the Hidden Tor Service Stealth Mode Operator.

Tor Hidden Service Server Side HiddenServiceAuthorizeClient auth-type client-name,client-name,…

If configured the hidden service is accessible for authorized clients only. The auth-type can either be ‘basic’ for a general-purpose authorization protocol or ‘stealth’ for a less scalable protocol that also hides service activity from unauthorized clients. Only clients that are listed here are authorized to access the hidden service. Valid client names are 1 to 16 characters long and only use characters in A-Za-z0-9+-_ (no spaces). If this option is set, the hidden service is not accessible for clients without authorization any more. Generated authorization data can be found in the hostname file. Clients need to put this authorization data in their configuration file using HidServAuth.

Example

```
HiddenServiceAuthorizeClient stealth ASm,Gor,Hid,001,002
```

Tor Client Side

```
HidServAuth onion-address auth-cookie [service-name]
```

Client authorization for a hidden service. Valid onion addresses contain 16 characters in a-z2-7 plus “.onion”, and valid auth cookies contain 22 characters in A-Za-z0-9+/. The service name is only used for internal purposes, e.g., for Tor controllers. This option may be used multiple times for different hidden services. If a hidden service uses authorization and this option is not set, the hidden service is not accessible. Hidden services can be configured to require authorization using the HiddenServiceAuthorizeClient option

Example

```
HidServAuth sqc4gjoq4aw2qufl.onion Y1vLvgTC1rOc2+JIZVZEjS
```

Reference Tor Manual <https://www.torproject.org/docs/tor-manual.html.en>

Direct Examples Hidden Node Operator and Clients torrc file additions.

```
HiddenServiceAuthorizeClient stealth client-name,name,name,name <-- Hidden Service torrc Stealth command ↩
    configuration
HiddenServiceAuthorizeClient stealth ASm,Gor,Hid,001,002 <-- Hidden Service Stealth torrc Example
HidServAuth onion-address auth-cookie <-- Client Stealth Tor Hidden Service torrc command configuration
HidServAuth g4efrxjzqs3li3un.onion lXOmu05+lIPoL46FCUuNQB <-- Client Stealth Tor Hidden Service torrc Example
```

Notice In the Tor Hidden Service torrc examples running in Stealth Mode for RS06 Friends ASmith,Gordon,Hidden, I use short nicnames or numbers for each Hidden Node Retroshare 0.6 Hidden Service Stealth Mode Tor Connection friends to your HiddenServiceAUthorizeClient where you see abc,001. I suggest you use 3 letters from RetroShare 0.6 friends usernames to keep this line compact however you can use numbers if you wish. You just must connect each short nicname or number to a specific friend/client to your Tor Hidden Service running in Stealth Mode.

HiddenServiceAuthorizeClient stealth ASm,Gor,Hid,001,002

Then the Tor Hidden Service Operator restarts Tor which generates a new hostname and client_keys file in their hidden server folder for each of your clients by nicnames or numbers.

In the new hostname file the Tor Hidden Service Operator finds

```
e2ffrxjzqs3li3um.onion mYOmu05+lIPoL46FCUuNQB # client: ASm
aqc9gjoq4aw2qufl.onion Z1vLvgTC1rOc2+JIZVZEjR # client: Gor
3phmajq5m5mnae3x.onion AyEgnImcoO+/IxFI/OORjx # client: Hid
4aw2quflaqc9gjoq.onion ZVZEjRZ1vLvgTC1rOc2+JI # client: 001
5mnae3x3phmajq5m.onion FI/OORjxAyEgnImcoO+/Ix # client: 002
```

Then the Tor Hidden Service Operator securely provides to ASmith this line to add to their torrc file HidServAuth e2ffrxjzqs3li3um.onion mYOmu05+lIPoL46FCUuNQB

Provides to Gordon this line to add to their torrc file HidServAuth aqc9gjoq4aw2qufl.onion Z1vLvgTC1rOc2+JIZVZEjR

Provides to Hidden this line to add to their torrc file HidServAuth 3phmajq5m5mnae3x.onion AyEgnImcoO+/IxFI/OORjx

Provides to someone you designate as 001 this line to add to their torrc file HidServAuth 4aw2quflaqc9gjoq.onion ZVZEjRZ1vLvgTC1rOc2+JI

Provides to someone you designate as 002 this line to add to their torrc file HidServAuth 5mnae3x3phmajq5m.onion FI/OORjxAyEgnImcoO+/Ix

Notice the encoded,encrypted Onion address is different for each client in the Stealth Mode. However each new Onion Address still connects back to the same Tor based Hidden Service.

Clients can also add additional HidServAuth lines to their torrc file to enable their authorization to additional Tor Hidden Services operating in stealth modes without problems. The additional Tor Hidden Service Operators running in stealth modes would provide the additional HidServAuth torrc line to add. Clients can also continue to connect to non-stealth Tor Hidden Services with and after these additions to their torrc file.

Misc. Stopping/Killing all Tor Processes If you already have tor running, stop all tor applications:

Issue a torrc file configuration check for errors and for Tor Hidden Service Operators to automatically generate your new hostname and client_key files in your Hidden Service folder.

```
$ tor <-- simply start tor
```

If successful (not erroring out) and following the examples above, Tor Hidden Service Operators running in stealth mode find inside their new client_keys file in your Hidden Service folder a entry like this:

client-name ASm
descriptor-cookie lXOmu05+lIPoL46FCUuNQB==
client-key
—–BEGIN RSA PRIVATE KEY—–
MIIDXQIBABKBgQDeaafSq5OlsFE00IQgeU0xuz1AuDqzWy4c3xdhG2dcB68i1R80
jt/YmtcMo6+8Pb/6PJT2oFuEPgAAIVNOxxgEtJ43krb6T/NMUVnL2yzc5EDsBsuW
yQuIkLcTeKq5aoasmwsNudAH0FuGer2qKXOPScU2Ldx/leQIusREnRVJ6wIDAQAB
AoGBAIe+loYW4BttHEv/A9VKE5o5vC3HA1S6Xs+SPZOwBQp/Sh4JU21hki+CeaQm
Nzi9u3EADq7juldg++EYdd8L1pTNLq9rsrNzE6losY67oWhlRIO9lCIozfHvjGi8
xd8d3vRFZv7qwUfg5MfbT3/CDfAQ1N2fps0/MwFn/Jix0XYRAkEA904iPXXichpf
Jmjxn+scfb03zh5V2ZHGvaqCrSZYHYjZ++3w4PmQnYlYzirG6vaIlKTAzrqGS7PX
JGAIyVnW+QJBAOY7ejQYIZpHuQ/PjlzenOhAsBe6xB0QM3F7m0b8L3U8OTAvN3kp
Fm7iP0+QexP0ci40mMyxv2TZhndoiWGuLQMCQBQFtWv4cV7/l5iKgpWJ+YKMoUoE
1rqIvOh6N55BLn0lqVBmUUxFqFm41wHKINTBJQmz1RBntZvHcbG2PL/gkDECQGWj
dDtzO8JkT2qPcoXFMPfHNvYD2XkLIerDarVXt4vF3mlLLu4qGWEyOEjju7H/hDip
t5sGDTx0W0xztdKNlgkCQQDAczBbamwkn4+W1qqS4gg11TzQD6oqnp0KYbz2aTeK
dkbuPpnZRhDAY0/EXPJOi5A7jkJK5IRm91YaQIDX7AQW
—–END RSA PRIVATE KEY—–

And you'll find in your new hostname file in your Hidden Service folder a entry like this:

g4efrxjzqs3li3un.onion lXOmu05+lIPoL46FCUuNQB # client: ASm

To authorize ASm (shorthand for ASmith) in this example, you'd supply ASmith with the following to insert into their torrc file:

HidServAuth g4efrxjzqs3li3un.onion lXOmu05+lIPoL46FCUuNQB

Repeat handing out each specific HidServAuth to each of your authorized friends (peers) following these examples collected from your hostname file inside your Hidden Service folder:

```
sqc4gjoq4aw2qufl.onion Y1vLvgTC1rOc2+JIZVZEjS # client: Gor
3phmajq5m5mnae2x.onion ByEgnImcoO+/IxFI/OORjy # client: 001
```

To authorize Gor (shorthand for Gordon) in this example, you'd supply Gordon with the following to insert into their torrc file:

```
HidServAuth sqc4gjoq4aw2qufl.onion Y1vLvgTC1rOc2+JIZVZEjS
```

To authorize 001 (shorthand for Jenster) in this example, you'd supply Jenster with the following to insert into their torrc file:

```
HidServAuth 3phmajq5m5mnae2x.onion ByEgnImcoO+/IxFI/OORjy
```

You can later add additional authorized friends (clients) to your HiddenServiceAuthorizeClient stealth xyz, torrc entry

You can later delete a previous authorized friend (client) from your:

```
HiddenServiceAuthorizeClient torrc entry
client_keys file
hostname file
```

```
$ tor --verify-config  <-- Terminal command to check and update your tor configuration files
```

More Information For The Curious

The RS06 developers did not automate the process of modifying the torrc file and creating the hidden service folder because there are different restrictions using the system files tor binary versus the user located Tor binary in the Tor Browser Bundle. The system

files Tor binary requires administrative/root/debian-tor permissions to read/write/modify the torrc file in the system file and in the Hidden Service folder.

It is important for computer security to prevent third party application with administrative/- 234
root/debian-tor permissions as a superuser on computer systems. To use the Tor binary in system files and not have to concurrently use the Tor Browser bundle while running multiple Tor applications including the RS06 Hidden Node.

The Tor Browser bundled Tor binary does not reside in the system files folders and 235
therefore does not need administrative/root/debian-tor permissions to access,write and modify the Hidden Service folder.

An undocumented feature in Vidalia was discovered which automatically changed the 236
permissions by the system files Tor binary to access/read/write/modify a user folder torrc file and write, access, modify the hidden service folder hostname, private_key files in a regular user created folder (for example in the notes previously is named which I choose to name vidalia-data).

This allows the advanced Tor user to use the system file Tor binary with a user stored, 237
created, changed, updated torrc (tor configuration file). As outlined in the previous RS06 Tor steps, other files need to be added such as geoip, geoip6 and cert caches to the user created vidalia-data folder by simply copying those over from the ones created and supplied with the Tor Browser Bundle. (/tor-browser_en-US/Browser/TorBrowser/-Data/Tor folder files)

If the system files Tor binary doesn't have the correct permissions to read, write and 238
modify the torrc file then it simply supplys some dumbed-down default values which in the case of a Hidden Service operation are going to fail everytime.

These issues can be solved by adding the tiny Vidalia Gui interface which also allows 239
the user instant debugging messages as well as other features for the user. The previous steps detailed above in the RS06 Tor pages how to use the system files tor binary with the Vidalia Gui interface.

FAQ 240

Question Does a hidden node attempt to connect directly to a regular node, thus 241
leaking its IP address, because that would defeat the whole idea of using RS over tor?

Answer A RetroShare 0.6 Hidden Node operating as a Tor Hidden Service is commu- 242
nicating inside the Tor Network itself. Regular RetroShare 0.6 instances running the Tor configuration communicate to them by going through the Tor Network and there to the Hidden Node itself. The Hidden Node isn't reaching out trying to go through the clearnet persay to find any Regular RetroShare 0.6 instance running Tor.

Question I have been told that hidden services were not really intended as a core 243
feature of Tor, and that Tor was really meant to be anonymous access to clear-net sites. However, hidden services works just fine so far as I can tell. **Answer** Yes,

the Onion Router was designed from the beginning to be a encrypted network of proxy servers exiting to the clear-net to allow the USA Military, undercover agents and various informants working in a foreign nation to be able to anonymously communicate without being traced back to their point of origin.

Onion based Hidden Service website abilities were added later and seen as a ancillary service. In a security application, essential applications and tools need to be built and added at the core level itself. Ancillary is like tossing a tool into your vehicle trunk. You have it in your car but at best, its not optimal and likely never will be because its not at the very core level on the application itself. I2P which is newer than Tor, designed its Hidden Services epsites at its very core and beginning with utmost focus on them and their security at the very foundation and beginning of the Invisible Net I23P project. 244

Question There MAY be scalability issues, since only one instance of Tor can accept connections to one .onion address at a time. 245

Answer It is possible to run 3-4 different applications through a single Tor Binary router all concurrently. 1-2 Tor Hidden Services, Opera Browser through Tor, 1-2 Regular Nodes of RetroShare 0.6 also routed through Tor. That single tor binary is working concurrently with Dozens of Onion Addresses at the same time. Some Tor Purists that run a single tor binary router per their hidden services and other tor routed packages. With the tor binary taking so little resources and the possibility of further security running 1 per use, its a personal choice not a ‘you must do it this way’. 246

Total Darknet via the Tor Network is not ideal. Although Tor developers are working to shore up Tor Hidden Services security and providing new much stronger alternatives to the way,way weak and ancient 1024bit asymmetric key pairs still being used, the reality is that Russia, China, USA and at least 32 other Nations State Sponsor attacks on the Tor Network, Tor Hidden Services, Exit Server Nodes and Tor Users themselves. Private business’s also provide tools to discern Tor users, Tor users locations and to ban or blacklist tor users. 247

Tor now is a virtual minefield. The risks can be somewhat mitigated with additional steps in security and realistic use. Let’s go through some suggestions and observations for both Tor Hidden Services as well as Tor Bundle Browser use. 248

1. Do not run the Tor Binary in Root/Administrator level. If a hacker through Tor breaks into your system they would gain root/administrative level control. 249

2. Weigh the benefits of running a Tor Hidden Service in True Stealth Mode if considering setting up and running a Tor Hidden Service and see if that is the approach you need Retroshare 0.6 Hidden Node Tor Stealth Mode 250

3. Consider generating additional encoded, encrypted .onion address’s along with their pathetic 1024bit keys to throw off any State Sponsored posse of thugs that somehow obtained a long used .onion address. This security step essentially is to change your .onion address now and then. 251

4. Pretending Tor use is entirely anonymous is entirely unrealistic. Posting threats of various natures online even via tor is likely to be responded to and then you’d be hunted 252

down. Commanding your torrc file to exit a widely used Tor exit node or nodes helps to disperse you among a crowd of other specific Tor users but you still are not invisible so don't act like you are entirely invisible and could get away with anything.

5. A great many clear-net websites are configured now to immediately recognize you as a Tor user. This through discerning you are using tor as a proxy directly or via tor exit IPv4 lookup databases. What many new Tor Browser Bundle users are learning and never told is discovering many sites list the Tor exit IPv4 they are using as a proxy is banned or blacklisted or is immediately recognized and challenged with a image puzzle which TBB cannot successfully respond to. 253

6. Some clear-net Tor Browser users in the past have spammed many blogs, forums chatrooms causing those website owners and developers to either outright ban,block all tor users or blacklist the popular Tor exit servers IPv4 itself. A great many clear-net websites immediately discern a visitor is using Tor and automatically go through security measures in case you are trying to hack or attack that website. Instead of your intention to be nearly anonymous and invisible at clear-net websites, in many its quite the reverse with a clear-net websites security measures triggered and watching your every move and action. Your access to additional areas in that website can be and often is restricted, in some cases the security measures can return you endlessly back to a registration page just to play with you as a tor user. 254

7. To milk out the remaining anonymous ability from the onion router, a tor browser bundle user would seek out a city public wifi hotspot and remotely use that public wifi connection without registering with any administrator or IT staff. The stealth user would not drive a car to the wifi hotspot as license plates would easily trace back to them and they'd be aware of and avoid CCTV cameras in that entire area. If a unwanted response was generated by this use, a 1 mile radius around the public area at a specific time would be targeted and scanned by State Sponsored goons. 255

8. These steps are worthless if your computer or smart device has been attacked and a long storage object used by Obama's State Sponsored Tor attackers has been placed on your device which serves as a 'beacon' IDing YOU, your IPv4 and other information about YOU. In short if your system is leaking through the clear-net identifying information about YOU. Anti-Virus, Anti-Maleware may or may not find these. Only a few can remove LSO beacons that can remain 2+ years. I side with Kaspersky Labs software because it is not 'Made in USA' and has been foremost in exposing Israeli-USA Gov created maleware,virus's and worms which 'Made in USA' firms either hushed up or were PoliceState ordered to ignore. Windows OS, Windows Apps have additional built-in hidden ID information that is designed to leak as well and maleware locators, anti-virus applications are largely going to ignore those Microsoft product ID code entries. 256

RetroShare FeedReader Via Tor 257

With the Tor Binary or Vidalia-Tor Bundle running on your computer system (socks5 port 9050), it's quite easy to add your Retroshare 0.6 FeedReader Plugin RSS feeds to provide updates from the http clearnet after traveling through Tor on your system. 258

When creating a new Feed or editing an existing Feed, uncheck the ‘Use standard proxy’ box. Then add:

```

```

Chose OK the select Update Feed. If you recieve a ‘wait, updating’ brief message without any errors being reported you are now using the Tor Network to obtain your FeedReader Feed.

This can also be additionally placed in Retroshare 0.6 Options - FeedReader where beneath Proxy you check the box near Use proxy and enter:

```
Server 127.0.0.1 : 9050
```

Tor Country Codes

Tor has the ability to let you choose which exit nodes you’d like to use - either by name, fingerprint, or country code. If you have a trusted list of nodes you’d like to use as a whitelist, you can use that, or if you have a list of nodes on a blacklist, it supports that configuration.

The configuration is simple, modify your torrc file to add the follwoing line to ensure Tor will only use a country based exit nodes making it appear you are in that nation and using one of that nations IP address’s:

Torrc (Tor Configuration File) insert examples:

```
ExitNodes {br}  <-- Use Only BRICs Founder Brazil Exit Nodes
StrictNodes 1 <-- You must enable StrictNodes

ExitNodes {in},{br},{za},{ar},{id},{bd},{my} <-- Use Only India,Brazil,Argentina,Indonesia,Bangladesh,↩
     Malaysia Exit Nodes
StrictNodes 1 <-- You must enable StrictNodes
```

Perhaps you want to bypass exits inside the West’s covert Urkaines warzone, insert example:

```

```

Perhaps you want to entirely exclude all USA node types fearing NSA poisoning including exits,relays,entry,bridge nodes, insert example:

BRIC's nations India,Brazil,South Africa (founders) and new BRICs members Argentina,Indonesia,Ba
are viewed as Secure and Respect users Privacys and Anonymous Data. Malaysia also appears to respect their online users privacys.

India {in} <– BRICs Founder Brazil {br} <– BRICs Founder South Africa {za} <– BRICs Founder Argentina {ar} <– Joining BRICs Indonesia {id} <– Joining BRICs Bangladesh {bd} <– Joining BRICs Malaysia {my} <– Protects Citizens Rights

Reference: ‹https://b3rn3d.herokuapp.com/blog/2014/03/05/tor-country-codes/›

External links Useful links to visit:

- ‹https://www.torproject.org/›
- ‹https://blog.torproject.org/›
- ‹https://www.torproject.org/projects/torbrowser.html.en›

Onion sites:

- Caves Tor hidden retrochat: ‹http://chat7zlxojqcf3nv.onion/›
- Silk Road 2.0 Url: ‹http://silkroad6ownowfk.onion›
- Agora Onion Address Tor ‹http://agorabasgefge4qo.onion›
- Utopia Market URL: ‹http://ggvow6fj3sehlm45.onion›
- RoadSilk url: ‹http://yjhzeedl5osagmmr.onion›
- White rabbit marketplace URL (Tor): ‹http://rabbittorvr74veg.onion›

Index

289

SiSU Metadata, document information

Document Manifest @:

‹http:///tor/en/manifest/rs-tor.html›

Title: RetroShare And Tor - A Manual For Configuration

Creator: Anonymous Smith

Prepared by: Morpheus Being

Subject: ebook,configuring tor

Publisher: SiSU ‹http://www.jus.uio.no/sisu› (this copy)

Date created: 2015

Date modified: 2015-07-15

Date: 2015-07-15

Topics Registered: Tor:RetroShare;RetroShare:MS Windows;RetroShare:Linux;RetroShare:MacOSX

Version Information

Sourcefile: rs-tor.sst

Filetype: SiSU text 7.1, UTF-8 Unicode text, with very long lines

Source Digest: SHA256(rs-tor.sst)=556f37b55521c0987d28c48232d8dfa046faf90753742a279ca46835583d46a0

Generated

Document (ao) last generated: 2015-07-17 16:04:00 +1000

Generated by: SiSU 7.1.5 of 2015w22/2 (2015-06-02)

Ruby version: ruby 2.1.5p273 (2014-11-13) [x86_64-linux-gnu]