# 1 Anonymizing Your Activities

*In our daily lives we like to have a certain level of privacy. We have curtains on our windows, doors for our offices, and even special screen protectors for computers to keep out prying eyes. This idea of wanting privacy also extends to the use of the Internet. We do not want people knowing what we typed in Google, what we said in our Instant Message conversations, or what websites we visited. Unfortunately, your private information is largely available if someone is watching. When doing any number of things on the Internet, there are plenty of reasons you might want to go incognito. However, that does not mean you're doing anything wrong or illegal.*

The justification for anonymity when researching malware and bad guys is pretty straightforward. You do not want information to show up in logs and other records that might tie back to you or your organization. For example, let's say you work at a financial firm and you recently detected that a banking trojan infected several of your systems. You collected malicious domain names, IP addresses, and other data related to the malware. The next steps you take in your research may lead you to websites owned by the criminals. As a result, if you are not taking precautions to stay anonymous, your IP address will show up in various logs and be visible to miscreants.

If the criminals can identify you or the organization from which you conduct your research, they may change tactics or go into hiding, thus spoiling your investigation. Even worse, they may turn the tables and attack you in a personal way (such as identity theft) or launch a distributed denial of service (DDoS) attack against your IP address. For example, the Storm worm initiated DDoS attacks against machines that scanned an infected system (see `http://www.securityfocus.com/news/11482`).

This chapter contains several methods that you can use to conduct research without blowing your cover. We've positioned this chapter to be first in the book, so you can use the techniques when following along with examples in the remaining chapters. Keep in mind that you may never truly be anonymous in what you are doing, but more privacy is better than no privacy!

## The Onion Router (Tor)

A widely known and accepted solution for staying anonymous on the Internet is *Tor*. Tor, despite being an acronym, is written with only the first letter capitalized and stands for *The Onion Router* or *the onion routing network*. The project has a long history stemming from a project run by the Naval Research Laboratory. You can read all about it at `http://www.torproject.org`.

Tor is a network of computers around the world that forward requests in an encrypted manner from the start of the request until it reaches the last machine in the network, which is known as an exit node. At this point, the request is decrypted and passed to the destination server. *Exit nodes* are specifically used as the last hop for traffic leaving the Tor network and then as the first hop for returning traffic. When you use Tor, the systems with which you are communicating see all incoming traffic as if it originated from the exit node. They do not know where you are located or what your actual IP address is. Furthermore, the other systems in the Tor network cannot determine your location either, because they are essentially forwarding traffic with no knowledge of where it actually originated. The responses to your requests will return to your system, but as far as the Tor network is concerned, you are just another hop along the way. In essence, you are anonymous. Figure 1-1 shows a simplified view of the Tor network.

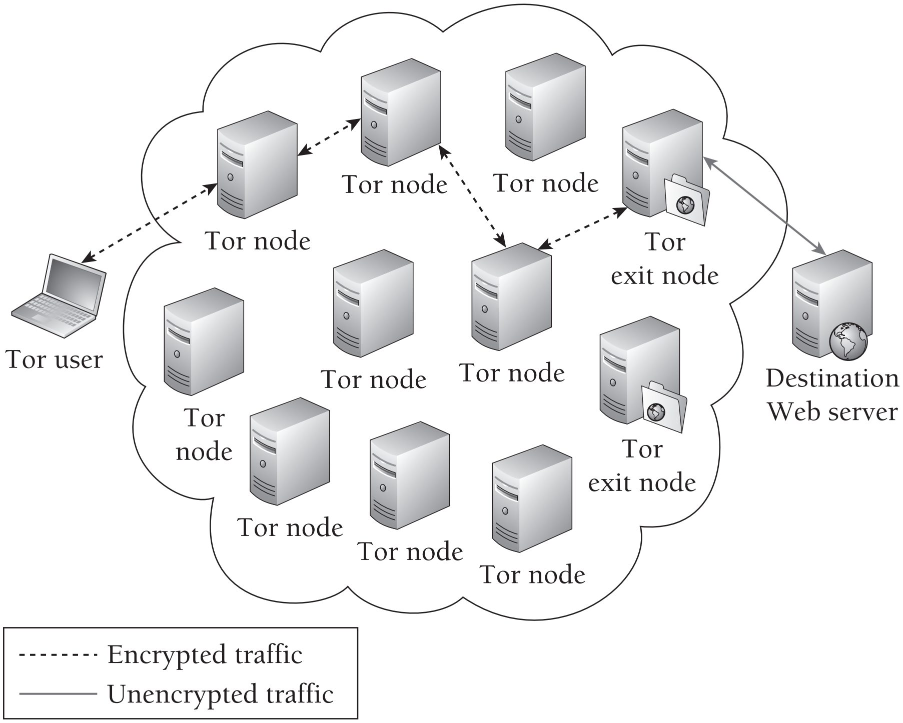


**Figure 1-1:** Simplified Tor Diagram

## RECIPE 1-1: ANONYMOUS WEB BROWSING WITH TOR

The Tor software is free to use and available for most computing platforms. You can install Tor on your Ubuntu system by typing `apt-get install tor`. For other platforms, such as Windows or Mac OS X, you can download the appropriate package from the Tor download page.[1] In most cases, the "Installation Bundle" for your operating system is what you want to install. If you need additional help, the website also has step-by-step instructions and videos.

The remainder of this recipe assumes you're installing Tor on Windows; however, the steps are largely the same for other platforms. Once it is installed, you can immediately start using Tor to anonymize your activity on the Web. Chances are that a lot of your investigative activities will be conducted through a web browser, and as a result you need your web requests to go through Tor. This is quite simple to do, because recent versions of the Tor bundles come with a Firefox extension called Torbutton.[2] Figure 1-2 shows what the button looks like when it is turned on and turned off. This button is located in the bottom right-hand corner of the browser once it is installed.

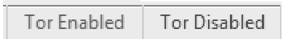


**Figure 1-2:** Firefox Torbutton

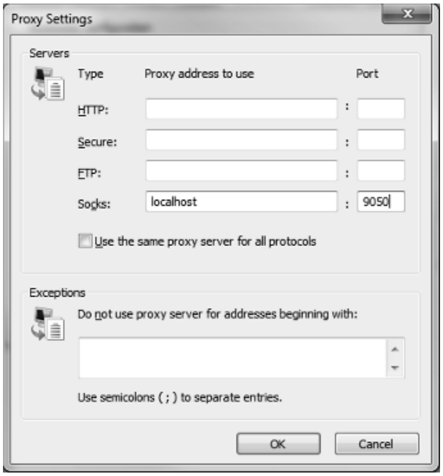
A simple click of the mouse allows you to enable or disable the use of Tor in the browser.

If you are using a browser other than Firefox, or you opt not to use the Torbutton add-on, you need to set up your browser to use Tor as a SOCKS4 or SOCKS5 proxy. Tor should bind to the localhost (127.0.0.1) on TCP port 9050 in its default configuration. This means it only accepts connections from your local computer and not from other systems on your network or on the Internet.

### Internet Explorer Configuration

To configure Internet Explorer (IE) to use Tor, follow these steps:

1. Click Tools ➪ Internet Options ➪ Connections ➪ LAN settings ➪ [x] "Use a proxy server for your LAN" ➪ Advanced. The Proxy Settings dialog appears.
2. In the Socks field, enter **localhost** in the first box for the proxy address and then **9050** for Port.
   Figure 1-3 shows how the IE Proxy Settings page should look once configured.



Proxy Settings

Servers

| Type | Proxy address to use | Port |
|---|---|---|
| HTTP: | | |
| Secure: | | |
| FTP: | | |
| Socks: | localhost | 9050 |

□ Use the same proxy server for all protocols

Exceptions

Do not use proxy server for addresses beginning with:

Use semicolons ( ; ) to separate entries.

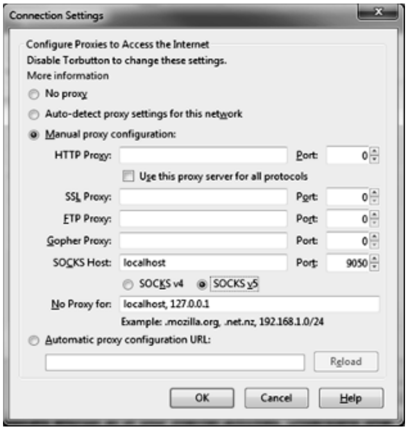
**Figure 1-3:** Internet Explorer Proxy Settings

## Firefox Configuration

You can configure Firefox to use Tor as a SOCKS proxy in the following manner:

1. Click Tools ➪ Options ➪ Advanced ➪ Network ➪ Settings ➪ Manual proxy configuration. The Connection Settings dialog appears.
2. For the SOCKS Host, enter **localhost** and for Port enter **9050** (you can select either SOCKS v4 or SOCKS v5).

Figure 1-4 shows how the Firefox Connection Settings page should look once configured.



Connection Settings

Configure Proxies to Access the Internet
Disable Torbutton to change these settings.
More information

○ No proxy
○ Auto-detect proxy settings for this network
● Manual proxy configuration:

| | | | |
|---|---|---|---|
| HTTP Proxy: | | Port: | 0 |
| | □ Use this proxy server for all protocols | | |
| SSL Proxy: | | Port: | 0 |
| FTP Proxy: | | Port: | 0 |
| Gopher Proxy: | | Port: | 0 |
| SOCKS Host: | localhost | Port: | 9050 |
| | ○ SOCKS v4 ● SOCKS v5 | | |
| No Proxy for: | localhost, 127.0.0.1 | | |
| | Example: .mozilla.org, .net.nz, 192.168.1.0/24 | | |

○ Automatic proxy configuration URL:

**Figure 1-4:** Firefox Connection Settings

At this point, you are up and running and can start browsing the Web, conducting research, and accessing content anonymously. To validate that your activities are now anonymous, we recommend that you quickly pull up a website such as `www.ipchicken.com` or `www.whatsmyip.org` and verify that the IP address returned by the website is not the IP address of your system. If this is the case, then everything is working fine and you can move along with your business anonymously.

**NOTE**

The *Tor Browser Bundle* is a self-extracting archive that has standalone versions of Tor, Vidalia (the Tor GUI), Polipo, and Firefox. It does not require any installation, and can be saved to and used from a portable storage device such as a USB drive. This can be very useful if you cannot install files on a system or want to quickly be up and running on a new machine without needing to install anything.

[1] `http://www.torproject.org/easy-download.html.en`

[2] `https://addons.mozilla.org/en-US/firefox/addon/2275`

## Malware Research with Tor

When researching malware, you may often need to anonymize more than just your web browsing. Tor can be used with command-line URL-fetching tools such as `wget`, or when connecting to SSH, FTP, or IRC servers. This section looks at tools that can be used to wrap Tor around your applications to ensure their connections appear to come from the Tor network and not directly from your system.

Recipe 1-2

### RECIPE 1-2: WRAPPING WGET AND NETWORK CLIENTS WITH TORSOCKS

ON THE DVD *You can find supporting material for this recipe on the companion DVD.*

In a Linux environment, you can use Torsocks[3] to wrap SOCKS-friendly applications with Tor. Torsocks ensures that your application's communications go through Tor, including DNS requests. It also explicitly rejects all (non DNS) UDP traffic from the application you are using in order to protect your privacy. To install Torsocks, use the following command:

```
$ sudo apt-get install torsocks
```

Once installed, you can begin using Torsocks, so long as Tor is running. By default, Torsocks sends its connections to TCP port 9050 on the localhost. This is the default port to which Tor binds. You can now leverage `usewithtor` to execute `wget`, `ssh`, `sftp`, `telnet`, and `ftp`, and their requests will be routed through the Tor network.

The following commands access `www.unlockedworkstation.com/ip.php` with and without the Tor network. The ip.php script returns the IP address of the connecting client and can be used to validate that your request went through Tor. The output shows that our IP without Tor is x.x.44.192 (sanitized for privacy) and the IP with Tor is 59.31.236.91.

```
$ wget www.unlockedworkstation.com/ip.php
$ cat ip.php
x.x.44.192

$ usewithtor wget www.unlockedworkstation.com/ip.php
$ cat ip.php
59.31.236.91
```

As long as the returned IP address is not that of your system, you know the request has worked. Keep in mind that `wget`, by default, will leak information about your system. For example, the following line may appear in the target website's access logs:

```
59.31.236.91 - - [03/Apr/2010:10:04:41 -0400] "GET /ip.php HTTP/1.0" \
                              200 12 "-" "Wget/1.12 (linux-gnu)"
```

The request told the web server that you were using `wget` version 1.12 and were sending it from a Linux-based system (Ubuntu in this case). This may not be a big deal, as your browser normally indicates the user agent and operating system being used. However, you may still wish to obfuscate this by providing a different user agent. You can do this with `wget` by using the `-U` flag.

```
$ usewithtor wget www.unlockedworkstation.com/ip.php \
    -U "Mozilla/5.0 (Windows NT; en-US) Gecko/20100316 Firefox/3.6.2"
```

This makes your request appear as if it came from a Firefox browser on a Windows 7 system. The more generic or common you make the user agent, the less likely it is that your requests can be distinguished from others. A simple bash script can be set up on your system to always use Torsocks, `wget`, and an alternate user agent. You can find a copy of the script named tgrab.sh on the book's DVD. Before using it, change the file's access permissions so that it can be executed.

```
$ cat tgrab.sh
#!/bin/bash

TSOCKS=`which usewithtor`
WGET=`which wget`
```

```
if [ $# -eq 0 ]; then

  echo "Please enter a URL to request";
  exit;

fi

$TSOCKS $WGET $1 -U "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; \
                                Trident/4.0; GTB6; .NET CLR 1.1.4322)"

$ chmod +x tgrab.sh
```

Now you can grab files with the command that follows without having to type out the user agent each time or having to precede the `wget` command with `usewithtor` each time.

```
$./tgrab.sh www.unlockedworkstation.com/ip.php
```

You can also wrap other applications with Torsocks just as you did with the `wget` command. Launch the applications as you would typically, but make sure to add `usewithtor` in front of your requests.

```
$ usewithtor ssh username@your-site-here.edu
$ usewithtor ftp user@your-site-here.edu
$ usewithtor sftp user@your-site-here.edu
$ usewithtor telnet your-site-here.edu 8000
```

Consider setting up small bash scripts, as we demonstrated in the previous code segment, for any commands that you run repetitively. You can easily paste any command you frequently run into a file, give it executable access permissions, and then run that file directly. This can save you time and prevent you from accidentally forgetting to send a particular request through `usewithtor`.

[3] http://code.google.com/p/torsocks/

## RECIPE 1-3: MULTI-PLATFORM TOR-ENABLED DOWNLOADER IN PYTHON

Recipe 1-3

ON THE DVD *You can find supporting material for this recipe on the companion DVD.*

In the previous recipe, you learned how to wrap `wget` requests with Torsocks. However, Torsocks does not support Mac OS X or Windows environments. This recipe shows you how to create a simple Tor-enabled file downloader in Python. As long as you can install Tor, Python, and the SocksiPy module (a generic SOCKS client), you can use this program to grab files from remote web servers without exposing your IP address.

To install the SocksiPy module, download the archive, extract socks.py from the Zip, and copy it into your site-packages directory.

```
$ unzip SocksiPy.zip
Archive:  SocksiPy.zip
  inflating: LICENSE
  inflating: BUGS
  inflating: README
  inflating: socks.py

$ cp socks.py /usr/lib/python2.5/site-packages/
```

The path to your site-packages directory will vary depending on your operating system. Here are the most likely locations for the correct site-packages directory on each platform (assuming you run Python 2.5):

- **Linux:** /usr/lib/python2.5/site-packages/
- **Mac OS X:** /Library/Python/2.5/site-packages/
- **Windows:** C:\Python25\site-packages\

Ensure that Tor is up and running on your system and locate the torwget.py script from the companion DVD. You may need to configure the following two variables at the top of torwget.py if you changed the default IP and port for Tor during set up.

```
TOR_SERVER = "127.0.0.1"
TOR_PORT = 9050
```

The script uses those variables to initialize a SOCKS proxy that sends all traffic through Tor. Then it overrides the default Python socket object with the class from SocksiPy. Any code used or imported from your Python script that uses sockets will then automatically send traffic through the Tor-enabled socket. In particular, since the script imports the `httplib` module (which uses sockets) to fetch URLs, the HTTP requests will be able to use Tor.

```
# Override the socket object with a Tor+Socks socket

socks.setdefaultproxy(socks.PROXY_TYPE_SOCKS5, TOR_SERVER, TOR_PORT)
socket.socket = socks.socksocket
```

You can print the script's usage by passing the `-help` flag, like this:

```
$ python torwget.py -help

usage: torwget.py [options]

options:
  -h, --help            show this help message and exit
  -r REFERRER, --referrer=REFERRER
                        use this Referrer
  -u USERAGENT, --useragent=USERAGENT
                        use this User Agent
  -c SITE, --connect=SITE
                        Connection string (i.e. www.sol.org/a.txt)
  -z, --randomize       Choose a random User Agent
```

If you want to download a file using a particular referrer and a random user agent, you can specify the following arguments. The user agent isn't truly random, it is just randomly selected from a hard-coded list in the torwget.py source code, which you can configure to your liking.

```
$ python torwget.py -c http://xyz.org/file.bin -r http://msn.com -z

Hostname: xyz.org
Path: /file.bin
Headers: {'Referrer': 'msn.com', 'Accept': '*/*', 'User-Agent':
'Opera/9.80 (Windows NT 5.1; U; cs) Presto/2.2.15 Version/10.00'}
Saving 21569 bytes to xyz.org/file.bin
Done!
```

The current version of torwget.py only supports fetching URLs using HTTP, however future versions may support FTP and other protocols.

[4] http://socksipy.sourceforge.net

## Tor Pitfalls

While Tor is a great service to use, it does have its pitfalls. These pitfalls may affect your speed of browsing, the security and integrity of data sent over the network, and your ability to access resources. Do not let these issues get in your way, but do make sure you are aware of them.

### Speed

At the time of this writing, the chief complaint against Tor is how slow browsing can be for the end user. This is a very well-known issue and exists for a few reasons. Your connection might be bouncing all over the world adding latency along the way—not to mention some Tor nodes may be low on bandwidth or already saturated. Fortunately, there are currently plans underway aimed at improving the speed and performance of the Tor network. You can't complain though, right? The service is free, after all. Of course you can—this is the Internet and everyone complains!

### Untrustworthy Tor Operators

Unscrupulous people have been known to run Tor exit nodes. What does that mean to you? It means there may be a Tor operator running an exit node that is specifically looking to monitor your traffic and in some cases modify it to their benefit. If you log into an application that does not use SSL to encrypt its passwords or session data, your credentials may be available to a snooping exit node operator.

Also, beware that Tor exit node operators, in their capacity to act as a man-in-the-middle, can inject traffic into unencrypted sessions. For example, should you be browsing a normal website, the unscrupulous exit node operator could inject an iframe or JavaScript reference that points to a malicious exploit website. If the code attempts to exploit something your system is vulnerable to, you may find your system infected with malware.

### Tor Block Lists

Several websites and services on the Internet specifically track what systems are acting as Tor exit node servers. This means that you may find yourself unable to access certain websites during your research if you are using Tor. While the majority of Tor usage may be legitimate, people can also use Tor to hide illegal and/or immature activities. As a result, some site administrators choose to block access from these IP addresses to cut down on this activity.

## Proxy Servers and Protocols

One of the original ways to stay anonymous on the Internet was through the use of proxy servers, or proxies. A *proxy server* is a system designed to work as an intermediary between a client making a request and the server responding to it. Organizations commonly use proxies to speed up traffic and save bandwidth through web caching, and to block unwanted content through content filtering. However, they can also be used for the specific purpose of remaining anonymous on the Internet.

When you use a proxy, all of your requests are first sent to the proxy and then to their destination. The proxy essentially acts as a man-in-the-middle between you and your destination. This set up may sound a lot like Tor. In reality, there are two very important differences.

- Unlike Tor, which has a whole network of systems, the proxy server you are communicating with is generally the only system between you and your destination, besides networking equipment and similar devices.
- Most importantly, there is no privacy between you and the proxy server. The proxy server knows who you are and knows that each request it receives is actually coming

from you. Compare that with Tor, where the exit node has no idea where the original request came from.

It is important that you know there are several proxy types. While proxies do act as a man-in-the-middle, they do not necessarily provide you full anonymity. Figure 1-5 shows how proxy servers work.

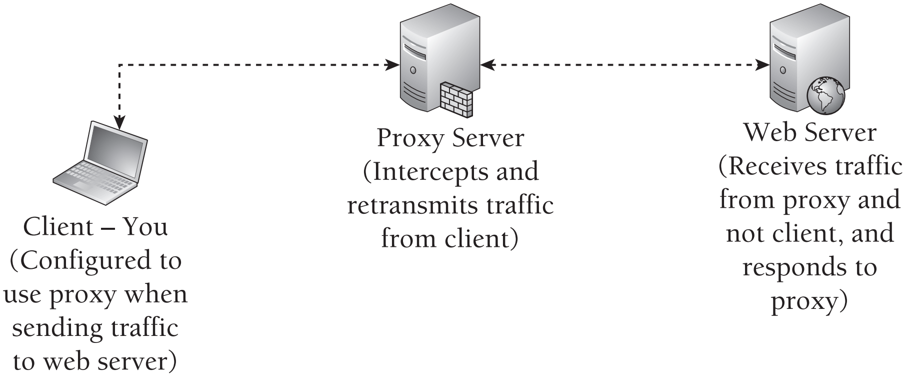

Client – You (Configured to use proxy when sending traffic to web server)
Proxy Server (Intercepts and retransmits traffic from client)
Web Server (Receives traffic from proxy and not client, and responds to proxy)

**Figure 1-5:** Proxy Server Diagram

Different proxies support a few different protocols. The three protocols you will see frequently are HTTP, SOCKS4, and SOCKS5. If you are just attempting to anonymize the research you are doing through a web browser, the protocols may not concern you. However, the following sections highlight some of the key differences between the three.

## HTTP

HTTP proxies support specially crafted requests that they will proxy and forward along to the requested resource. HTTP proxies are generally used for non-encrypted connections, but some may support SSL. They may also support FTP and HTTP methods such as CONNECT, which allow non-HTTP communication.

## SOCKS4

SOCKS4 is a protocol that is designed to handle traffic between a client and server by way of an intermediary proxy. SOCKS4 only supports the TCP communication protocol. It does not contain a method for authentication. SOCKS4 is not the most recent version of the SOCKS protocol, but it is still widely used and accepted. It is worth noting that SOCKS4A is an extension to SOCKS4 that added support for resolving DNS names.

## SOCKS5

SOCKS5 is the current version of the SOCKS protocol and is an extension of the SOCKS4 protocol. It supports both the TCP and UDP protocols for communication. It also adds on methods to support authentication from the client to the proxy server.

## RECIPE 1-4: FORWARDING TRAFFIC THROUGH OPEN PROXIES

ON THE DVD *You can find supporting material for this recipe on the companion DVD.*

The first thing you need to do before setting up and using a proxy is to find one that works. To do this, you can consult several websites that provide a list of free proxies to use. These websites generally list the IP address of the proxy, its port, protocol, and type. Below are a few websites that contain a list of free proxies that you can use.

- `http://www.xroxy.com`
- `http://www.proxy4free.com`
- `http://aliveproxy.com/`
- `http://www.freeproxylists.com`

Once you locate a proxy, you can configure your web browser to use it by following the steps detailed in Recipe 1-1 for configuring Tor. Just enter the IP address of the proxy and the port that the proxy is listening on. You can validate that the proxy is working in the same manner as you validated Tor—by going to a website that will return back your IP address (e.g. `http://www.ipchicken.com`).

### Choosing a Proxy Type

The most important factor when choosing a proxy is to determine what type to use. When we say *proxy type*, we are not referring to what protocol it is using, but rather the level of anonymity that you have as a proxy user. Proxy types include *transparent*, *anonymous*, and *highly anonymous*.

In this recipe, we are going to introduce you to the various proxy types and show you examples of additional artifacts that they may add to your requests. We will show you how you can test the proxies and see what HTTP fields they modify (if any) and what information may potentially be leaked as a result. Aside from protecting your own identity, you can use this knowledge when tracking attackers who are hiding behind proxies.

NOTE

There is no way to guarantee that the proxy you are using hasn't been set up by miscreants to sniff traffic or is not a misconfigured device that has been discovered on the Internet. Use caution when selecting and using proxies found on these websites.

### Validating Proxy Type

To test a proxy, you'll need to capture what the target website sees when the proxy forwards your requests. You can do this by setting up a PHP script on a web server that you own, and visiting it while using the proxy. For convenience, we created a script called header_check.php, which can be found on the companion DVD. Below you will find the contents of the header_check.php script. Place this file in an accessible directory on your web server to use it.

```
<?php

$get_headers = apache_request_headers();

echo $_SERVER['REQUEST_METHOD'] . " " .
     $_SERVER['REQUEST_URI'] . " " .
     $_SERVER['SERVER_PROTOCOL'] . "<br/>";

foreach ($get_headers as $header => $value) {
    echo "$header: $value <br/>\n";
}

echo "<br/><br/>Your IP address is: " . $_SERVER['REMOTE_ADDR'];

?>
```

Requesting this file from a web browser will result in it returning the request you made along with all HTTP headers. By using the `REMOTE_ADDR` variable, it can also print the IP address of the client machine.

In the following examples, we sanitized the IP addresses of the proxies we used for privacy. Here is a list that you can use for reference:

- 192.168.5.88 is the IP address of the system we are making the requests from.
- 10.20.30.40 is the IP address of a transparent proxy.
- 10.20.30.50 is the IP address of an anonymous proxy.
- 10.20.30.60 is the IP address of a highly-anonymous proxy.

Before moving on, you should use the script to generate a baseline of what requests look like from your browser without the use of a proxy. The output below shows the headers printed by header_check.php.

```
GET /header_check.php HTTP/1.1
Host: www.unlockedworkstation.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.5) \
                                           Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Your IP address is: 192.168.5.88
```

The above request returned our baseline header information, which we can compare to the other requests that are made with proxies enabled. This will allow us to see what types of elements might be added by different proxy types. As the output shows, the server sees our connection originating from our real IP address.

## Transparent Proxies

RFC 2617 defines a *transparent proxy* as a proxy that does not modify the request or response beyond what is required for proxy authentication and identification. In other words, most fields should not be modified. However, transparent proxies—at least most of the ones you find on the Web—often do not conceal information about the source of their requests. When a client uses a transparent proxy, all requests to the server still come from the IP address of the proxy server. However, the proxy server adds an additional HTTP header indicating the original source of the request.

The request that follows is what a web server sees from a browser that is using a transparent proxy:

```
GET /header_check.php HTTP/1.1
Host: www.unlockedworkstation.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.5) \
                                           Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Via: 1.1 proxy:3128 (squid/2.5.STABLE11)
X-Forwarded-For: 192.168.5.88
Cache-Control: max-age=259200
Connection: keep-alive

Your IP address is: 10.20.30.40
```

To the target web server, our connection appears to have originated from the IP address of the proxy. 10.20.30.40 is the address that will show up in the web access logs. However, as you can see, several HTTP header fields were added to this request. In particular, the `X-Forwarded-For` and `Via` headers identify our real IP address and which proxy software is being used. This provides little to no anonymity.

### Anonymous Proxies

*Anonymous proxies* do not reveal your IP address to the server to which you are making a request. However, they normally add in some form of additional information that will indicate that the request is coming from a proxy server. They may still contain an `X-Forwarded-For` header but the IP address that is supplied will likely contain the IP address of the proxy server or a value that is otherwise not your IP address. If the supplied value is a real IP address but does not belong to you or the proxy server, the proxy is said to be a *distorting proxy*.

Compare the following request that a web server sees from a browser using an anonymous proxy to the baseline request that did not use a proxy.

```
GET /header_check.php HTTP/1.1
Host: www.unlockedworkstation.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.5) \
                                          Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Via: 1.1 x81prx00 (NetCache NetApp/6.0.7)

Your IP address is: 10.20.30.50
```

Now you can see that your IP address was not passed along in this request. However, an additional HTTP header called `Via` was added to the request, which identifies the proxy software being used (x81prx00). Some identifiers that are passed by anonymous proxies might be unique to you. This means that while the target web server might not be capable of converting this information back to your IP address, it may still distinguish all of your requests from others.

### Highly Anonymous Proxies

*Highly anonymous proxies* do not reveal your IP address or any other information to a target web server. These are the most desired of the proxy types because they provide the highest level of anonymity. When you use a highly anonymous proxy, request headers

Recipe 1-4

from the proxy server appear no different from those you make yourself. However, they are coming from the IP address of the proxy server.

```
GET /header_check.php HTTP/1.1
Host: www.unlockedworkstation.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT; en-US; rv:1.9.1.5) \
                                          Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive

Your IP address is: 10.20.30.60
```

Compare this request with the one sent without a proxy; you'll notice they look identical. The only difference is that the web server saw the connection coming from the proxy IP instead of your IP. This is not to say that all highly anonymous proxies do not make some modifications to headers, but the modifications should not identify you or the fact that the server is a proxy.

Recipe 1-5

## RECIPE 1-5: USING SSH TUNNELS TO PROXY CONNECTIONS

A great way to proxy your connections is to use port forwarding through an SSH tunnel. SSH tunnels allow you open up a listening port on your local workstation, connect to your server via SSH, and then use your server as a SOCKS4/5 proxy. You can then use any application that supports SOCKS4/5 proxies to access resources using the IP address of the server you have logged into via SSH.

The first step in this process is to have a shell account on a remote SSH server that you would like to use for your tunneling. Several companies offer cheap shell accounts that can be used for this purpose. The Super Dimension Fortress (SDF) Public Access UNIX System[5] offers SSH tunneling/port forwarding as a part of their MetaARPA membership for $36 a year.

Setting up an SSH tunnel to be used as a SOCKS4/5 proxy in Linux or Mac OS X is simple. Just follow these steps:

1. From a shell on your workstation, launch `ssh` to your server with the `-D` flag.

```
$ ssh user@shell-server.net -D1080
```

This sets up dynamic application-level port forwarding by binding a listening socket to your system on TCP port 1080. If the connection succeeded, you should see the SSH client listening on the port specified.

```
$ sudo netstat -tnlp | grep 1080
tcp     0      0 127.0.0.1:1080     0.0.0.0:*    LISTEN   17190/ssh
```

2. You can now configure applications that support SOCKS4/5 proxies to use your workstation (localhost or 127.0.0.1) and TCP port 1080 for connections. Your SSH server will effectively be a SOCKS proxy accessible to your local system.
3. You can be more specific with SSH tunneling by forwarding connections to a certain local port to a specific IP and port combination. For example, if you only wanted to proxy your SSH connections to `unlockedworkstation.com` on TCP port 80, you would do the following:

   ```
   $ ssh user@shell-server.net -L2080:unlockedworkstation.com:80
   ```

4. Now you can make connections to your localhost on TCP port 2080 and they will be proxied through your SSH server to the IP address for `unlockedworkstation.com` on TCP port 80.

   ```
   $ wget http://localhost:2080
   ```

When you use `ssh` to set up a tunnel, it will result in a command shell on the SSH server. You may not want to keep this window open, but if you close it, your tunnel will no longer persist. To alleviate this problem, you can keep the connection alive and throw it in the background. The following is a modified version of one of our earlier examples.

```
$ ssh user@shell-server.net -D1080 -f -N
```

The `-f` flag requests that the SSH client process goes into the background just before command execution. The `-N` flag tells SSH not to execute any remote commands (just maintain an open tunnel).

### SSH Proxies on Windows

The steps to accomplish an SSH tunnel on a Windows workstation are very different, but can still be easily accomplished with the PuTTY[6] SSH client. The Web Hosting Talk website has a good post with step-by-step instructions[7] for doing this with PuTTY.

[5] http://sdf.lonestar.org

[6] http://www.chiark.greenend.org.uk/~sgtatham/putty/

[7] http://www.webhostingtalk.com/showthread.php?t=539067

## RECIPE 1-6: PRIVACY-ENHANCED WEB BROWSING WITH PRIVOXY

If you are interested in enhancing your privacy while browsing the Internet, with or without anonymity, you may want to consider looking into Privoxy.[8] *Privoxy* is a non-caching web proxy that filters out ads and other unwanted content. The software is highly configurable, but by default it can:

- filter banner ads, web bugs, and HTML annoyances
- bypass click-tracking scripts and redirections
- remove animation from GIFs

You can run Privoxy on your local system or you can set it up on a server on your network that multiple users can access. Privoxy does not support authentication, so you should only use it in a trusted network or otherwise apply some form of access restriction to the system.

On an Ubuntu system, you can install Privoxy by typing `apt-get install privoxy`. Then you can start it by using the `service` command or by launching /etc/init.d/privoxy.

```
$ service privoxy start
Starting Privoxy, OK.
```

If the service started properly, you'll see a process listening on port 8118 of localhost (127.0.0.1).

```
$ sudo netstat -tnlp | grep privoxy
tcp     0     0 127.0.0.1:8118  0.0.0.0:*  LISTEN  28270/privoxy
```

### Configuring Privoxy for Multiple Clients

As previously mentioned, you can configure Privoxy to act as a server so that multiple clients can access it. To do this, modify the `listen-address` parameter in the Privoxy configuration file (/usr/local/etc/privoxy/config on most systems). The default is shown in the following code:

```
listen-address  127.0.0.1:8118
```

Modify `127.0.0.1` to be the IP address of your server that is accessible to the other clients on your network. If your IP address is `192.168.1.200`, edit the config to look like the following:

```
listen-address  192.168.1.200:8118
```

### Configuring Browsers to Use Privoxy

Once clients configure the HTTP proxy setting of their browsers to use `192.168.1.200:8118`, all web requests will go through Privoxy. If you want to use Privoxy and Tor, you can do that, too. Simply modify the Privoxy config file to point to the Tor listener as a SOCKS5 proxy. If the system running Privoxy is also running Tor, you can uncomment the following from the config file:

```
forward-socks5   /               127.0.0.1:9050 .
```

If this is uncommented, Privoxy will send all outbound requests through Tor (assuming Tor is running and bound to the server locally on port 9050), giving you both anonymity and a higher level of privacy.

[8] http://www.privoxy.org/

## Web-Based Anonymizers

*Web-based anonymizers* are essentially HTTP proxies wrapped up into a web interface. Instead of configuring the proxy settings of your browser, you visit an anonymizer site and tell it where you want to go. This is often easier and quicker than the proxies we described in Recipe 1-4. The web-based anonymizer sends your request to the destination and displays the web pages back to you, as if you visited the destination directly. You will notice that the URL bar on your browser still contains the address for the anonymizer site.

The set up and configuration of various web-based anonymizers vary from site to site. They will likely only work for HTTP or HTTPS communication. Depending on the site, you may have restrictions on common HTTP methods (POST requests may not be allowed), download sizes, allowed ports, cookies, and other limitations imposed by the server. Much like other proxy types we discussed earlier in the chapter, web-based anonymizers often add fields to your requests that make it readily apparent you are using a proxy. However, most web-based anonymizers do not have fields that present your IP address to the destination server.

Most web-based anonymizers are available for free. However, there are pay services that offer additional features, such as content filtering and protection from known phishing and exploit websites. The same pitfalls and risks mentioned in the Tor and Proxies sections apply here, especially when using the free services.

## RECIPE 1-7: ANONYMOUS SURFING WITH ANONYMOUSE.ORG

The website `www.anonymouse.org` is a free web-based anonymizer that can be used from virtually any browser. When you visit the site, enter your destination URL and press the Surf anonymously button, as shown in Figure 1-6.

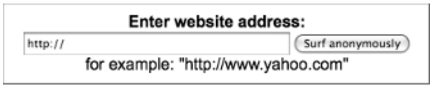


**Figure 1-6:** Anonymouse.org Web Form

You are anonymously redirected to the website you entered and the page loads as if you visited it directly, only with a few minor changes. The website's title has the text [Anonymoused] appended to it. Additionally, the HTML source for the website has an iframe at the bottom that loads an advertisement on the page. You can close the advertisement, but it will reappear each time you browse to a new page. Alternatively, you may sign up to use the Anonymouse service without advertisements for a small monthly fee.

The `Anonymouse.org` website is an anonymous proxy. The website hides your IP address, browser type, and operating system when making requests to websites on your behalf. However, it modifies the HTTP headers, which makes it obvious that you used a proxy service. The following example shows what a web server sees when a request is made to it through the Anonymouse proxy service. We used the header_check.php script described in Recipe 1-4 to capture the data.

```
GET /header_check.php HTTP/1.1
Host: www.unlockedworkstation.com
User-Agent: http://Anonymouse.org/ (Unix)
Connection: keep-alive

Your IP address is: 193.200.150.137
```

The IP you see in the output is the address of a proxy server owned by Anonymous .org. The service makes it apparent through the user agent string that your request is coming from the `Anonymouse.org` website. This keeps your identity safe but makes it readily apparent to anyone that is looking that you are using a web-based proxy service for your requests.

## Alternate Ways to Stay Anonymous

There are a few alternate ways to stay relatively anonymous while doing your research. In particular, the use of cellular Internet connections and virtual private networks (VPNs) can be great options. You may have to shell out a few dollars for either solution, but in the end it may be well worth it. Both solutions provide a certain level of anonymity as far as the outside world can tell. You will not have to worry about leaked DNS queries, or configuring browsers or applications to use proxies with either of these two methods.

## Cellular Internet Connections

The main benefit to using a cellular Internet connection to stay anonymous is that the IP address by itself cannot be tied directly back to you by any outside party. Your cellular carrier, of course, has the capability to link the IP address to you. Each time you connect, you will likely receive a different, dynamically assigned IP address. If someone is tracking your previous activity based on your IP address, they will run into trouble, because you can change your IP by simply reconnecting.

The strength of the signal and the quality of the coverage in your area may have a drastic impact on the type of speeds you see when you connect to a cellular network. However, you should be able to do light investigative work. Because you are already relatively anonymous, it may not be necessary to use one of the other anonymizing services such as Tor or a proxy. Should you choose to use one of these other services on top of your cellular Internet connection, you may find your browsing and related activities become very slow.

Some computing devices, such as laptops, often have cellular modems built into them these days. However, cell phone companies generally provide you with a cellular modem (often at a cost) to use their service. These modems plug right into your laptop or computer and allow you to connect to the Internet with additional software. USB-based cellular modems allow you the most flexibility because you can use them with most laptop and desktop computers.

### RECIPE 1-8: INTERNET ACCESS THROUGH CELLULAR NETWORKS

The first step to connecting anonymously with a cellular Internet provider is to sign up for the service and obtain a cellular card or device. Most cellular cards come with software that helps you connect to the service. Some cards may automatically configure themselves, such as PCI-X and PCMCIA cards for Mac OS X. Figure 1-7 shows an example of the Verizon VZAccess Manager that is used for connecting to Verizon's cellular network.

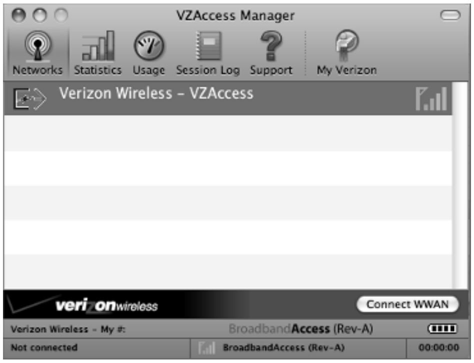


**Figure 1-7:** Verizon VZAccess Manager

The bars on the right side under the menu bar work the same as they do on your cellular phone and indicate signal strength. Click the Connect WWAN button to initiate the connection. Once connected, Verizon Wireless supplies you with an IP address from a large pool of addresses that they own. You can now browse the Internet anonymously.

A final item to keep in mind is that you can still essentially be profiled while using a cellular Internet connection. Your IP address may change all the time, but it is still possible for someone to figure out your general location. In addition, someone looking into your activity can tell that you are using a cellular Internet connection for your access. If you continually do research from these services, the bad guys may also determine that the research you do on subsequent visits is related to past research, even if the IP address has changed.

## Virtual Private Networks

There are many different types of VPNs and ways to both authenticate and connect to them. When you use a VPN, you are setting up a connection with a remote server that allows you to send traffic through it, similar to how a proxy works. However, the main difference is that your system is generally assigned an IP address on the VPN's network and all the traffic between your machine and the VPN is encrypted.

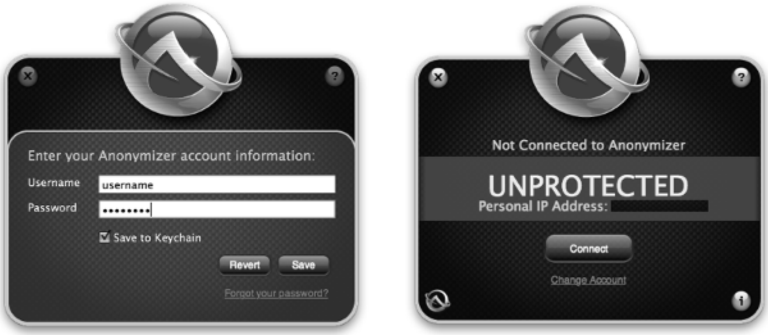
If you want to build your own VPN infrastructure, you can purchase a virtual private server from a hosting provider such as Linode (`http://www.linode.com`) or Amazon's EC2 (`http://aws.amazon.com/ec2/`). Then install and configure a free, open source product such as OpenVPN (`http://openvpn.net/`) onto your server. Alternately, you can use a commercial solution, which cuts down on the set up and maintenance that you'll need to perform.

## RECIPE 1-9: USING VPNS WITH ANONYMIZER UNIVERSAL

Anonymizer, Inc. offers a service called Anonymizer Universal,[9] which provides an encrypted L2TP/IPSec VPN service that has a pool of tens of thousands of constantly rotating "untraceable IP addresses" for approximately $79.99 a year. It allows you to connect in an instant and start conducting all of your activities from one of the untraceable IP addresses. Anonymizer does not modify your traffic to include identifying information that might lead back to you or your real IP address.

After you obtain an Anonymizer account, you'll be able to download client software and configuration files for Windows, Mac OS X, and the iPhone. The set ups for Windows and Mac OS X are very straightforward. You can just launch the Anonymizer Universal application, as shown in Figure 1-8.

Enter your account information and save it. You will then be brought to a screen that displays your IP address. It shows that you are "unprotected," as all of your network activity will come from the personal IP address that is displayed. Now click Connect and let Anonymizer establish a VPN connection with its back-end service. Once the connection succeeds, you are assigned a new IP address, as shown in Figure 1-9.



Enter your Anonymizer account information:

Username: username

Password: ••••••••

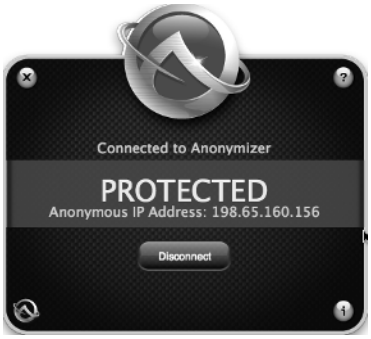
☑ Save to Keychain

Not Connected to Anonymizer

UNPROTECTED

Personal IP Address: [redacted]

**Figure 1-8:** Anonymizer—Account Info and Unprotected



Connected to Anonymizer

PROTECTED

Anonymous IP Address: 198.65.160.156

**Figure 1-9:** Anonymizer—Protected

You now have an IP address that is not tied back to you. In this case, the IP address the Anonymizer service has assigned to you is registered to NTT America. The GeoLocation for the IP address says it is in Colorado and the WHOIS information points to Delaware and California. Nothing about this IP address reveals that is a proxy. You can now perform your investigations over the Internet and all of the activity will come from the IP address 198.65.160.156.

[9] http://www.anonymizer.com

## Being Unique and Not Getting Busted

This chapter discussed a few ways you might be fingerprinted or otherwise stand out while trying to remain anonymous. Whether it is through a proxy-modified HTTP header or an IP address range, repeated activity can clearly make you stand out to someone that is watching.

Your browser and the various plug-ins can reveal a lot of information. Often a simple request to a website can result in passive fingerprinting that can determine your operating system, browser type and version, language settings, and more. Various plug-ins—Adobe Flash, Acrobat, QuickTime, Java, and even Facebook—can also probe your system.

The Electronic Frontier Foundation (EFF) has a website called Panopticlick (http://panopticlick.eff.org/) that helps determine how unique your browser is when compared to others. This website uses code from BrowserSpy (http://browserspy.dk/) to determine how much information is revealed about your computer through your web browser. Using these tools, it may be possible for someone to fingerprint each of your visits to their website, despite the fact that you visited on different days using a different IP address each time—and they can do this without the use of cookies or any persistent data set by the website. If you are interested in understanding more about how fingerprinting works and how you can be identified and tracked, it's definitely worth taking a look at the Panopticlick website.

Other techniques that attackers may use can reveal your real IP address even if you're using a highly anonymous proxy. For example, code on a web page can often instruct Flash to make a connection that does not go through your proxy, thus revealing your real IP address. Other methods may reveal your DNS server. Potentially, you could do anonymous research from your place of business and someone could watch your activities, see that your DNS lookup came from ns1.your-company-name-here.com, and bust you as a result. The website for the Metasploit Decloaking Engine (http://decloak.net/) has a tool to demonstrate several of these issues. Use this website to see if they can, in fact, decloak you while you're behind a proxy.

Despite all of this, you can do several things to defend yourself against these methods of fingerprinting. A simple measure that can go a long way is to disable JavaScript during your anonymous research activities. You can further manage and control this, even during your non-research activities, through the NoScript (`http://noscript.net`) Firefox extension. This add-on for Firefox can protect you from exploits using JavaScript, Java, Flash, or other browser plug-ins.

You should follow a few other general rules and practices to stay anonymous during research activities. The following is a list of considerations to take into account before starting any research:

- When signing up for various accounts, do not use an account name that identifies you or your organization. Additionally, do not use a password that you use elsewhere in your normal day-to-day activity.
- If you come across something that seems questionable or if your own activities worry you, even though they are anonymous, you should stop.

Although you think you're doing all you can to stay anonymous during your activities, consider that your research might reduce your level of anonymity. For example, your organization may have been targeted with a piece of malware that, when run, connects to bad-website.com/connection/report.php. If you were to attempt to access this domain yourself, even while taking all the right steps to stay anonymous, you might still end up revealing yourself to the bad guys. Unknown to you, the bad guys may have used the domain name specifically to attack your organization and no others. So searching, probing, or otherwise revealing the existence of this domain shows the bad guys that the activity is coming from someone at your company. Although you did not provide any information to directly identify yourself or use an IP address with ties to your organization, you have been indirectly identified and your cover has been blown.