Tunneling SSH over an HTTP-Proxy Server

Can't use SSH on the standard port 22? Need to tunnel through a proxy server? Work behind a draconian firewall and can't SSH directly? No problem. This document will hopefully show you how to tunnel through an http-proxy server without any server-side modifications.

Build and Configure an HTTP-Proxy Application

1. **Get Corkscrew:** available from corkscrew home page.

   I've tried other http-tunnel programs, but this is truly the easiest one I've found and it doesn't require server-side applications (such as are required by httptunnel, which is a good program otherwise). Furthermore, **`corkscrew`** works on every UNIX platform I've tried and even compiles and runs flawlessly under Cygwin on Windows.

2. **Unpack and Compile `corkscrew`:**

   ```
   tar -xzvf corkscrew.tar.gz
   # [..]
   cd corkscrew
   ./configure
   make install
   ```

   Presuming no errors, **`corkscrew`** is now installed in **`/usr/local/bin`** on your machine. If you want to put it somewhere else, use the **`--prefix=`*`path`*** flag to the **`configure`** script.

3. **Add `ProxyCommand` to your SSH config file:**

   You may or may not have a configuration file for SSH already. It should be located in **`$HOME/.ssh/config`** and is a simple text file. Create one if it does not exist and add lines such as these to it:

   ```
   Host *
     ProxyCommand corkscrew http-proxy.example.com 8080 %h %p
   ```

   … replacing ***`http-proxy.example.com`*** with the name or address of your http proxy and possibly replacing **`8080`** with the port on which the proxy listens, which may be 80 or even some other port. The **`%h`** and **`%p`** will be replaced automatically by SSH with the actual destination host and port.

   These two lines tell the SSH client to start another program (**`corkscrew`**) to make the actual connection to the SSH server. The **`Host *`** line says that this will be done for **ALL** hosts. If you wish to restrict the hosts for which this will be done, you can put a limited form of regular expression there. See the **`ssh_config(5)`** man page for more information. If you don't have **`corkscrew`** in your path or have put it in a non-standard location, you may specify an absolute path to **`corkscrew`** in that file as well.

4. **Try it out…**

   ```
   ssh example.net
   ```

   … replacing example.net with the name of a host to which you can connect using SSH. Presumably this host will be outside your local network and therefore require the use of the proxy server. If it is not outside your local network, then the connection may fail as the proxy-server or some firewall may be configured to **not** redirect proxy connections back into your local network.

   Either of the following two errors probably indicate an error in your **`~/.ssh/config`** file, most likely the name or port of the proxy server.

```
ssh_exchange_identification: Connection closed by remote host
[ OR ]
ssh: connection to host example.net port 22: Connection timed out
```

Congratulations - you are using an http-proxy server with SSH. Anything you can do with SSH you should now be able to do through the proxy server, including tunneling of other ports or even ppp.

Other Tricks with HTTP-Proxies

1. **Authenticated proxy connections**

   Some proxy servers require authentication. In this case, you can add authentication credentials to the **`ProxyCommand`** line:

   ```
   Host *
     ProxyCommand corkscrew http-proxy.example.com 8080 %h %p ~/.ssh/p
   ```

   In the **`~/.ssh/proxyauth`** file, put your proxy login and password like this:

   ```
   <username>:<passwd>
   ```

   `Corkscrew` should now happily use that authentiaction information and tunnel your connection through the proxy.

2. **Automatically detect proxy-server availability.**

   Sometimes you might be on a portable computer and only **sometimes** behind that firewall or on the network with the proxy server? Note that the **`ProxyCommand`** configuration item can be just about anything you like, as long as it reads from standard-input and writes to standard-output. Using that fact, we can write a wrapper around **`corkscrew`** when the proxy tunnel is needed.

   i. download my **`ssh-proxy`** script and place it in your **`~/.ssh`** directory.

   ii. Change your **`~/.ssh/config`** file to include the following:

   ```
   Host *
     ProxyCommand $HOME/.ssh/ssh-proxy http-proxy.example.com 8080
   ```

   The relevant line is of course the **`ProxyCommand`** line and it looks darn similar to the previous version. All that this script does is attempt to connect *directly* to the destination host first, falling back to using the proxy server specified if a direct connection is not possible.

   Note that the script uses another program called **`netcat`** (sometimes just **`nc`**) to test and make direct connections. If you don't have **`netcat`**, you can look here, but any decent system, including Cygwin, should have it installed by default.

   iii. Shorten the timeout for trying a direct connection:

   Note that the **`ssh-proxy`** script defines a default timeout (8 seconds) for testing direct connections to the remote host. If that timeout seems too long to you, you can shorten it by adding a **`-w <seconds>`** flag in the **`ProxyCommand`** line of your **`~/.ssh/config`** file, like this:

   ```
   Host *
     ProxyCommand $HOME/.ssh/ssh-proxy -w 2 http-proxy.example.com
   ```

   If on the other hand, 2 seconds is too short, you can make it longer too.

   iv. Specify the location of **`netcat`** or **`corkscrew`**:

   Just like you can specify a alternate timeout, you can use two other options to specify the name and/or location of the **`netcat`** and **`corkscrew`** programs:

```
-n path-to-netcat/direct-connect-program
-t path-to-corkscrew/http-tunnel-program
```

One could even specify a completely different direct-connect or proxy-tunnel programs, but then you are probably going to have to modify the source as the arguments are not likely to be the same. Just look at the source.

Last Modified: Tue, Jan 27 2009, 10:29:24 -0500