**Debugging UNIX sockets with socat**

Written by [Matthew Scharley](mailto:matt@scharley.me) at Monday, 19 March 2012 - 11:02

Last updated: Friday, 23 March 2012 - 14:41

A while ago, I came across an interesting problem: how do you listen in on UNIX sockets? For IP, there are tools like wireshark that will display in great detail all the network traffic that go through your system, but these tools can’t listen in on UNIX sockets.

**Introducing socat**

So, what is socat?

**Socat** is a command line based utility that establishes two bidirectional byte streams and transfers data between them. Because the streams can be constructed from a large set of different types of data sinks and sources (see address types), and because lots of address options may be applied to the streams, socat can be used for many different purposes.

[*socat manpage*](http://www.dest-unreach.org/socat/doc/socat.html)

In other words, socat sits in the middle of two different sources of IPC and passes data between them. I use the term IPC because socat understands so many different formats that it is a little rediculous. It can process and translate between the following formats (not even close to [a comprehensive list](http://www.dest-unreach.org/socat/doc/socat.html#ADDRESS_CREAT)):

* TCP with or without SSL
* UNIX sockets
* regular files
* stdin/stdout of another process

As you can see, it’s very flexible. It also supports logging all traffic that flows through it. Which brings us back to our original point.

**Debugging UNIX sockets**

UNIX sockets are great for all sorts of reasons, including ensuring that they are only available locally. However, this also means that they are hard to listen in on and debug. So, enter socat:

$ socat -v UNIX-LISTEN:/tmp/socat-listen UNIX-CONNECT:/path/to/real.socket

Now, point the client application at /tmp/socat-listen instead of the real socket and enjoy the conversation.

Or, if you wish to provide your own test data rather than simply listening in:

$ socat -v READLINE UNIX-CONNECT:/path/to/real.socket

With this, socat will use readline to get data from the console, and print back any replies.

**In closing**

Obviously, this is only one way to use socat. It’s a very powerful application, and can be put to lots of good uses. This is just one of the more obscure things you can do with it. You could even use it as a telnet replacement if you really wanted to with the second example. You should explore the [man page](http://www.dest-unreach.org/socat/doc/socat.html) to get other cool ideas.