Read divert sockets in java?

Asked 16 years, 3 months ago Modified 12 years, 1 month ago

Viewed 340 times

0

If I was to create a ipfw divert rule to forward all FTP traffic to a specific socket, is it possible to use Java to connect to the socekt and read the packet information? If so, how would i go about reading/writing to the scoket?

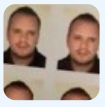
java ftp tcp ipfw

Share

Improve this question

Follow

asked Sep 21, 2008 at 12:41



abnev

469 ● 1 ● 8 ● 20

3 Answers

Sorted by: Highest score (default)

0

not sure what you mean. If you're using a divert rule, then all you have to do is listen on that ip:port combination in your java app and you're all set. If you want to read the actual destination endpoint information, you'll need to use JNI for that.

Share Improve this answer

Follow

answered Sep 21, 2008 at 13:19

Shachar

943 ● 1 ● 9 ● 18

That's not true. When you create the socket, you actually need to specify it's a divert port. – ianatha Oct 26, 2012 at 1:59

0

Yes, it's like a normal socket, you can read/write from/to it, but on Mac OS X, if you do want to modify the packet and insert it back, you need to recalculate the tcp check sum first.

http://blog.loudhush.ro/2006/08/using-divert-sockets-on-mac-os-x.html

This is a good post that introduce the basic usage of the divert socket on Mac OS X. You can actually create the rule in your C code.

For you case, just scan the packet for TCP or IP header and parse for whatever you want

Share Improve this answer

Follow

answered Nov 9, 2010 at 16:42

Spike

302 ● 2 ● 4

0

Java doesn't natively support binding onto a Divert socket. It's not exactly a cross-platform feature (it's a BSD/Mac OS X) thing...

You could try using the "socket" native function call through JNA.

Your code will look something like this:

```
public interface LibC extends Library {
  int socket(int domain, int type, int protocol);
  int close(int fd);

  /* stuff from /System/Library/Frameworks/Kernel.fram
... */
  /* ... /sys/socket.h */
  static int AF_INET = 2;
  static int SOCK_RAW = 3;
  /* .. netinet/in.h */
  static int IPPROTO_DIVERT = 254;
}

public static void main(String[] args) {
  LibC lib = (LibC) Native.loadLibrary("C", LibC.class
  int fd = lib.socket(LibC.AF_INET, LibC.SOCK_RAW, Lib
  if (fd <= 0) {
    throw new RuntimeException("couldn't open divert s
  }
  System.out.println(fd);
  System.out.println(lib.close(fd));
}
```

Share Improve this answer Follow

edited Oct 26, 2012 at 2:23

answered Oct 26, 2012 at 2:00

ianatha

307 ● 3 ● 9