

Re: Socket holding pattern

Source: <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.misc/2004-09/0785.html>

From: Uri Guttman (uri_at_stemsystems.com)

Date: 09/09/04

Date: Thu, 09 Sep 2004 13:49:29 GMT

>>>> "AS" == Anno Siegel <anno4000@lublin.zrz.tu-berlin.de> writes:

AS> Another possibility is to open the relevant file handles (sockets)
AS> so that they are immune to close-on-exec. (Set \$^F to something
AS> huge while they are opened). The old server process would have
AS> to tell the new one which file descriptors are in use, so the new
AS> one can take them over. Besides the compatibility issues already
AS> mentioned, there may be synchronization problems when both processes
AS> hold file handles to active connections.

it can be much harder than that too. most servers of this type (long lived connections) keep state about each connection so that has to be stored in a place away from the socket itself. if that data is in objects and you reload that class, who knows what will happen to those objects? the newly loaded code would always need to know it is fresh and reload those objects and such. this idea of reloading code on the fly and keeping connections is not for the faint of heart nor the weak in coding and design skills.

and i agree, since this is only for a mud, who cares? kick all the players and let them reconnect.

uri

--

Uri Guttman ----- uri@stemsystems.com ----- <http://www.stemsystems.com>
--Perl Consulting, Stem Development, Systems Architecture, Design and Coding-
Search or Offer Perl Jobs ----- <http://jobs.perl.org>