

Re: parallelizing ipfw table

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/net/2005-11/0260.html>

From: Gleb Smirnoff (glebius_at_FreeBSD.org)

Date: 11/28/05

Date: Mon, 28 Nov 2005 13:52:50 +0300

To: ". "@babolo.ru

On Mon, Nov 28, 2005 at 01:42:41PM +0300, ". "@babolo.ru wrote:

.> > On Mon, Nov 28, 2005 at 08:27:32AM +0200, Ruslan Ermilov wrote:

.> > R> > Can you try my patch? Since it reduces the total number of mutex

.> > R> > operations it should be a win on UP, too.

.> > R> We're currently based on 4.x. You can try it yourself: create

.> > R> a table with 10000 entries and with value 13. Then write a

.> > R> ruleset with 13 rules that look up this table so that the last

.> > R> rule looks it up with value 13, and do a benchmark. Let me

.> > R> know what are results with and without caching.

.> > Such kind of firewall looks like unoptimized. Why should we optimize the

.> > code for non-optimized setups. Can't we avoid looking into one table

.> > 13 times each packet?

.>

.> add 47400 pipe 47400 ip from table(0, 0) to any

.> add 47401 pipe 47401 ip from table(0, 1) to any

.> add 47402 pipe 47402 ip from table(0, 2) to any

.> add 47403 pipe 47403 ip from table(0, 3) to any

.> add 47404 pipe 47404 ip from table(0, 4) to any

.> add 47405 pipe 47405 ip from table(0, 5) to any

.> add 47406 pipe 47406 ip from table(0, 6) to any

.> add 47407 pipe 47407 ip from table(0, 7) to any

.> add 47408 pipe 47408 ip from table(0, 8) to any

.> add 47409 pipe 47409 ip from table(0, 9) to any

.>

.> for different traffic consumers listed in table(0)

I understand now. Ruslan has sent me a sample setup, too. Anyway, the current optimization is broken on SMP, because it stores the cache in the table itself. Parallel processing of the different packets on SMP breaks the optimization, since different instances of ipfw_chk() trash the cached addr one after another.

I have two ideas about this. First, store the cache on stack. Second, utilize the table entry value in the rule. In this case your block can be converted to:

```
add N pipe \${val} ip from table(0) to any
```

freebsd-net: Re: parallelizing ipfw table

\\$val means the value of the entry in the table.

--

Totus tuus, Glebius.
GLEBIUS-RIPN GLEB-RIPE

freebsd-net@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-net>

To unsubscribe, send any mail to "freebsd-net-unsubscribe@freebsd.org"