

tcp_push_pending_frames() without TCP_CORK or TCP_NODELAY

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-07/6094.html>

From: Robert White (*rwhite_at_casabyte.com*)

Date: 07/30/04

To: <linux-kernel@vger.kernel.org>

Date: Thu, 29 Jul 2004 19:19:00 -0700

Greetings,

I have several environments where I have two or more boxes tied together with short/private Ethernet segments. The applications on these boxes toss events back and forth using that network. Reliability is most important so I use TCP, but performance is also important as a close second. The events are often quite short and are, as often as not assembled on the fly in the code via multiple writes.

The protocol(s) also have "known complete" moments where the sender knows that a complete event has been written.

If I turn Nagle off, then the segmented write-on-assembly generates a lot of very short (3 to 30 etc bytes) packets; if I leave it on then Nagle tends to delay the complete events (because, of course, the stack isn't psychic 8-).

I currently flush these event boundaries by turning nagle off and then back on using back-to-back calls to setsockopt(). The extra syscalls seem like a waste. To that end I am looking into a patch to add a SIOCFLUSH ioctl or similar.

The below [untested] patch is my first-take on the question. I am interested in knowing whether this looks useful to others (or ill conceived etc 8-) before I try to add this to the tree pervasively.

[Sorry about the mis-indented line to stifle outlook word-wrap. I "have to" use this bloody program here at work... /sigh 8-)]

==== Begin Patch =====

--- linux.orig/include/asm-i386/sockios.h 2004-06-15 22:19:02.000000000 -0700

+++ linux/include/asm-i386/sockios.h 2004-07-29 18:37:22.000000000 -0700

@@ -8,5 +8,6 @@

#define SIOCGPGRP 0x8904

#define SIOCATMARK 0x8905

#define SIOCGSTAMP 0x8906 /* Get stamp */

+#define SIOCFLUSH 0x8907

Linux-Kernel: tcp_push_pending_frames() without TCP_CORK or TCP_NODELAY

```
#endif
diff --recursive -u linux.orig/net/ipv4/tcp.c linux/net/ipv4/tcp.c
--- linux.orig/net/ipv4/tcp.c 2004-06-15 22:19:03.000000000 -0700
+++ linux/net/ipv4/tcp.c 2004-07-29 19:09:07.000000000 -0700
@@ -526,6 +526,14 @@
     else
         answ = tp->write_seq - tp->snd_una;
     break;
+ case SIOCFLUSH:
+ {
+ __u8 scratch = tp->nonagle;
+ tp->nonagle = (scratch & ~TCP_NAGLE_CORK) | TCP_NAGLE_OFF | TCP_NAGLE_PUSH;
+ tcp_push_pending_frames(sk, tp);
+ tp->nonagle = scratch;
+ }
+ return 0;
     default:
         return -ENOIOCTLCMD;
     };
==== End Patch ====
```

Robert White,
Casabyte, Inc.

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>