

new/usr/src/uts/common/inet/ip/ip.c

1

448794 Mon Nov 10 08:23:16 2014
new/usr/src/uts/common/inet/ip/ip.c
4674 No need to undefine rptr in ip.c

_____unchanged_portion_omitted_____

```
12038 /*
12039  * Prepend an M_DATA fastpath header, and if none present prepend a
12040  * DL_UNITDATA_REQ. Frees the mblk on failure.
12041  *
12042  * nce_dlur_mp and nce_fp_mp can not disappear once they have been set.
12043  * If there is a change to them, the nce will be deleted (condemned) and
12044  * a new nce_t will be created when packets are sent. Thus we need no locks
12045  * to access those fields.
12046  *
12047  * We preserve b_band to support IPQoS. If a DL_UNITDATA_REQ is prepended
12048  * we place b_band in dl_priority.dl_max.
12049  */
12050 static mblk_t *
12051 ip_xmit_attach_llhdr(mblk_t *mp, nce_t *nce)
12052 {
12053     uint_t hlen;
12054     mblk_t *mp1;
12055     uint_t priority;
12056     uchar_t *rptr;
12057
12058     rptr = mp->b_rptr;
12059
12060     ASSERT(DB_TYPE(mp) == M_DATA);
12061     priority = mp->b_band;
12062
12063     ASSERT(nce != NULL);
12064     if ((mp1 = nce->nce_fp_mp) != NULL) {
12065         hlen = MBLKL(mp1);
12066         /*
12067          * Check if we have enough room to prepend fastpath
12068          * header
12069          */
12070         if (hlen != 0 && (rptr - mp->b_datap->db_base) >= hlen) {
12071             rptr -= hlen;
12072             bcopy(mp1->b_rptr, rptr, hlen);
12073             /*
12074              * Set the b_rptr to the start of the link layer
12075              * header
12076              */
12077             mp->b_rptr = rptr;
12078             return (mp);
12079         }
12080         mp1 = copyb(mp1);
12081         if (mp1 == NULL) {
12082             ill_t *ill = nce->nce_ill;
12083
12084             BUMP_MIB(ill->ill_ip_mib, ipIfStatsOutDiscards);
12085             ip_drop_output("ipIfStatsOutDiscards", mp, ill);
12086             freemsg(mp);
12087             return (NULL);
12088         }
12089         mp1->b_band = priority;
12090         mp1->b_cont = mp;
12091         DB_CKSUMSTART(mp1) = DB_CKSUMSTART(mp);
12092         DB_CKSUMSTUFF(mp1) = DB_CKSUMSTUFF(mp);
12093         DB_CKSUMEND(mp1) = DB_CKSUMEND(mp);
12094         DB_CKSUMFLAGS(mp1) = DB_CKSUMFLAGS(mp);
12095         DB_LSOMSS(mp1) = DB_LSOMSS(mp);
12096         DTRACE_PROBE1(ip_xmit__copyb, (mblk_t *), mp1);
12097     }
```

new/usr/src/uts/common/inet/ip/ip.c

2

```
12097     /*
12098      * XXX disable ICK_VALID and compute checksum
12099      * here; can happen if nce_fp_mp changes and
12100      * it can't be copied now due to insufficient
12101      * space. (unlikely, fp mp can change, but it
12102      * does not increase in length)
12103      */
12104     return (mp1);
12105 }
12106 mp1 = copyb(nce->nce_dlur_mp);
12107
12108 if (mp1 == NULL) {
12109     ill_t *ill = nce->nce_ill;
12110
12111     BUMP_MIB(ill->ill_ip_mib, ipIfStatsOutDiscards);
12112     ip_drop_output("ipIfStatsOutDiscards", mp, ill);
12113     freemsg(mp);
12114     return (NULL);
12115 }
12116 mp1->b_cont = mp;
12117 if (priority != 0) {
12118     mp1->b_band = priority;
12119     ((dl_unitdata_req_t *) (mp1->b_rptr))->dl_priority.dl_max =
12120         priority;
12121 }
12122 return (mp1);
12123 #undef rptr
12123 }
12124 _____unchanged_portion_omitted_____
```