

[RFC 2.6.10 1/22] xfrm: Add direction information to xfrm_state

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-12/6190.html>

From: David Dillow (dave_at_thedillows.org)

Date: 12/30/04

Date: Thu, 30 Dec 2004 03:48:34 -0500

To: netdev@oss.sgi.com

```
# This is a BitKeeper generated diff -Nru style patch.
#
# ChangeSet
# 2004/12/30 00:27:15-05:00 dave@thedillows.org
# Add direction information to xfrm_state. This will be needed to
# offload xfrm processing to the NIC.
#
# Signed-off-by: David Dillow <dave@thedillows.org>
#
# net/xfrm/xfrm_state.c
# 2004/12/30 00:25:42-05:00 dave@thedillows.org +5 -0
# Add direction information to xfrm_state. This will be needed to
# offload xfrm processing to the NIC.
#
# Signed-off-by: David Dillow <dave@thedillows.org>
#
# net/ipv6/xfrm6_state.c
# 2004/12/30 00:25:42-05:00 dave@thedillows.org +9 -0
# Place holder for adding IPv6 direction mapping routine.
#
# net/ipv4/xfrm4_state.c
# 2004/12/30 00:25:42-05:00 dave@thedillows.org +10 -0
# Add direction information to xfrm_state. This will be needed to
# offload xfrm processing to the NIC.
#
# Signed-off-by: David Dillow <dave@thedillows.org>
#
# include/net/xfrm.h
# 2004/12/30 00:25:42-05:00 dave@thedillows.org +10 -0
# Add direction information to xfrm_state. This will be needed to
# offload xfrm processing to the NIC.
#
# Signed-off-by: David Dillow <dave@thedillows.org>
#
diff -Nru a/include/net/xfrm.h b/include/net/xfrm.h
```

Linux–Kernel: [RFC 2.6.10 1/22] xfrm: Add direction information to xfrm_state

```

--- a/include/net/xfrm.h 2004-12-30 01:12:08 -05:00
+++ b/include/net/xfrm.h 2004-12-30 01:12:08 -05:00
@@ -146,6 +146,9 @@
     /* Private data of this transformer, format is opaque,
      * interpreted by xfrm_type methods. */
     void *data;
+
+ /* Intended direction of this state, used for offloading */
+ int dir;
+ };

enum {
@@ -157,6 +160,12 @@
     XFRM_STATE_DEAD
};

+enum {
+ XFRM_STATE_DIR_UNKNOWN,
+ XFRM_STATE_DIR_IN,
+ XFRM_STATE_DIR_OUT,
+};
+
+ struct xfrm_type;
+ struct xfrm_dst;
+ struct xfrm_policy_afinfo {
@@ -194,6 +203,7 @@
     struct xfrm_state *(*find_acq)(u8 mode, u32 reqid, u8 proto,
                                   xfrm_address_t *daddr, xfrm_address_t *saddr,
                                   int create);
+ void (*map_direction)(struct xfrm_state *xfrm);
+ };

extern int xfrm_state_register_afinfo(struct xfrm_state_afinfo *afinfo);
diff -Nru a/net/ipv4/xfrm4_state.c b/net/ipv4/xfrm4_state.c
--- a/net/ipv4/xfrm4_state.c 2004-12-30 01:12:08 -05:00
+++ b/net/ipv4/xfrm4_state.c 2004-12-30 01:12:08 -05:00
@@ -106,12 +106,22 @@
     return x0;
}

+static void
+_xfrm4_map_direction(struct xfrm_state *x)
+{
+ if (inet_addr_type(x->id.daddr.a4) == RTN_LOCAL)
+ x->dir = XFRM_STATE_DIR_IN;
+ else
+ x->dir = XFRM_STATE_DIR_OUT;
+}
+
+ static struct xfrm_state_afinfo xfrm4_state_afinfo = {
     .family = AF_INET,

```

```

        .lock = RW_LOCK_UNLOCKED,
        .init_tempsel = __xfrm4_init_tempsel,
        .state_lookup = __xfrm4_state_lookup,
        .find_acq = __xfrm4_find_acq,
+ .map_direction = __xfrm4_map_direction,
    };

void __init xfrm4_state_init(void)
diff -Nru a/net/ipv6/xfrm6_state.c b/net/ipv6/xfrm6_state.c
--- a/net/ipv6/xfrm6_state.c 2004-12-30 01:12:08 -05:00
+++ b/net/ipv6/xfrm6_state.c 2004-12-30 01:12:08 -05:00
@@ -116,12 +116,21 @@
        return x0;
    }

+static void
+__xfrm6_map_direction(struct xfrm_state *x)
+{
+ /* XXX This needs to be implemented by someone who knows
+ * IPv6 better than I.
+ */
+}
+
static struct xfrm_state_afinfo xfrm6_state_afinfo = {
    .family = AF_INET6,
    .lock = RW_LOCK_UNLOCKED,
    .init_tempsel = __xfrm6_init_tempsel,
    .state_lookup = __xfrm6_state_lookup,
    .find_acq = __xfrm6_find_acq,
+ .map_direction = __xfrm6_map_direction,
};

void __init xfrm6_state_init(void)
diff -Nru a/net/xfrm/xfrm_state.c b/net/xfrm/xfrm_state.c
--- a/net/xfrm/xfrm_state.c 2004-12-30 01:12:08 -05:00
+++ b/net/xfrm/xfrm_state.c 2004-12-30 01:12:08 -05:00
@@ -186,6 +186,7 @@
        x->lft.soft_packet_limit = XFRM_INF;
        x->lft.hard_byte_limit = XFRM_INF;
        x->lft.hard_packet_limit = XFRM_INF;
+ x->dir = XFRM_STATE_DIR_UNKNOWN;
        spin_lock_init(&x->lock);
    }
    return x;
@@ -404,6 +405,8 @@
    if (unlikely(afinfo == NULL))
        return -EAFNOSUPPORT;

+ afinfo->map_direction(x);
+
    spin_lock_bh(&xfrm_state_lock);

```

Linux-Kernel: [RFC 2.6.10 1/22] xfrm: Add direction information to xfrm_state

```
x1 = ainfo->state_lookup(&x->id.daddr, x->id.spi, x->id.proto);
@@ -451,6 +454,8 @@
    ainfo = xfrm_state_get_ainfo(x->props.family);
    if (unlikely(ainfo == NULL))
        return -EAFNOSUPPORT;
+
+ ainfo->map_direction(x);

    spin_lock_bh(&xfrm_state_lock);
    x1 = ainfo->state_lookup(&x->id.daddr, x->id.spi, x->id.proto);
-
```

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/>