
Subject: Re: [PATCH][ICMP]: Dst entry leak in icmp_send host re-lookup code (v2).
Posted by [Herbert Xu](#) on Wed, 02 Apr 2008 12:40:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wed, Apr 02, 2008 at 12:19:06PM +0300, Julian Anastasov wrote:

>

> OK, we found there was a leak, but why it happens? Initially,
> I thought it was caused by saddr=0 provided to ip_route_input. Some
> debugging shows that in the case with forwarded skb (with attached
> input route) saddr is set to 0 but later xfrm_decode_session_reverse
> rebuilds fl with addresses from packet. So, it was not that we
> play with saddr=0. In my test setup with 2 interfaces ip_route_input
> failed because I don't have route to the original destination
> which is now provided as saddr to ip_route_input. No ICMP was sent
> to sender while previous kernels send ICMP.

Yes this was an oversight.

[ICMP]: Ensure that ICMP relookup maintains status quo

The ICMP relookup path is only meant to modify behaviour when appropriate IPsec policies are in place and marked as requiring relookups. It is certainly not meant to modify behaviour when IPsec policies don't exist at all.

However, due to an oversight on the error paths existing behaviour may in fact change should one of the relookup steps fail.

This patch corrects this by redirecting all errors on relookup failures to the previous code path. That is, if the initial xfrm_lookup let the packet pass, we will stand by that decision should the relookup fail due to an error.

This should be safe from a security point-of-view because compliant systems must install a default deny policy so the packet would'nt have passed in that case.

Many thanks to Julian Anastasov for pointing out this error.

Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>

Cheers,

--

Visit Openswan at <http://www.openswan.org/>

Email: Herbert Xu ~{PmV>Hl~} <herbert@gondor.apana.org.au>

Home Page: <http://gondor.apana.org.au/~herbert/>

PGP Key: <http://gondor.apana.org.au/~herbert/pubkey.txt>

--

```

diff --git a/net/ipv4/icmp.c b/net/ipv4/icmp.c
index a944e80..40508ba 100644
--- a/net/ipv4/icmp.c
+++ b/net/ipv4/icmp.c
@@ -591,7 +591,7 @@ void icmp_send(struct sk_buff *skb_in, int type, int code, __be32 info)
 }

    if (xfrm_decode_session_reverse(skb_in, &fl, AF_INET))
-   goto ende;
+   goto relookup_failed;

    if (inet_addr_type(net, fl.fl4_src) == RTN_LOCAL)
        err = __ip_route_output_key(net, &rt2, &fl);
@@ -601,7 +601,7 @@ void icmp_send(struct sk_buff *skb_in, int type, int code, __be32 info)

    fl2.fl4_dst = fl.fl4_src;
    if (ip_route_output_key(net, &rt2, &fl2))
-   goto ende;
+   goto relookup_failed;

    /* Ugh! */
    odst = skb_in->dst;
@@ -614,21 +614,21 @@ void icmp_send(struct sk_buff *skb_in, int type, int code, __be32 info)
 }

    if (err)
-   goto ende;
+   goto relookup_failed;

    err = xfrm_lookup((struct dst_entry **)&rt2, &fl, NULL,
        XFRM_LOOKUP_ICMP);
-   if (err == -ENOENT) {
+   switch (err) {
+   case 0:
+       dst_release(&rt->u.dst);
+       rt = rt2;
+       break;
+   case -EPERM:
+       goto ende;
+   default:
+relookup_failed:
+       if (!rt)
+           goto out_unlock;
-       goto route_done;
+       break;
    }
-
-   dst_release(&rt->u.dst);

```

```

- rt = rt2;
-
- if (err)
- goto out_unlock;
}

```

route_done:

```
diff --git a/net/ipv6/icmp.c b/net/ipv6/icmp.c
```

```
index f204a72..893287e 100644
```

```
--- a/net/ipv6/icmp.c
```

```
+++ b/net/ipv6/icmp.c
```

```
@@ -436,24 +436,26 @@ void icmpv6_send(struct sk_buff *skb, int type, int code, __u32 info,
}
```

```

    if (xfrm_decode_session_reverse(skb, &fl2, AF_INET6))
- goto out_dst_release;
+ goto relookup_failed;

```

```

    if (ip6_dst_lookup(sk, &dst2, &fl))
- goto out_dst_release;
+ goto relookup_failed;

```

```

    err = xfrm_lookup(&dst2, &fl, sk, XFRM_LOOKUP_ICMP);
- if (err == -ENOENT) {
+ switch (err) {
+ case 0:
+ dst_release(dst);
+ dst = dst2;
+ break;
+ case -EPERM:
+ goto out_dst_release;
+ default:
+relookup_failed:
    if (!dst)
        goto out;
- goto route_done;
+ break;
}

```

```

- dst_release(dst);
- dst = dst2;
-
- if (err)
- goto out;
-

```

route_done:

```

if (ipv6_addr_is_multicast(&fl.fl6_dst))
    hlimit = np->mcast_hops;

```