



author Alexander Potapenko <glider@google.com> 2022-11-04 11:32:16 +0100
committer David S. Miller <davem@davemloft.net> 2022-11-07 12:26:15 +0000
commit [c23fb2c82267638f9d206cb96bb93e1f93ad7828](#) (patch)
tree [05cae64d7c51e99706c929554d610716350297d7](#)
parent [0c175da7b0378445f5ef53904247cfbfb87e0b78](#) (diff)
download [linux-c23fb2c82267638f9d206cb96bb93e1f93ad7828.tar.gz](#)

diff options

context:
space:
mode:

ipv6: addrlabel: fix infoleak when sending struct ifaddrlblmsg to network

When copying a `struct ifaddrlblmsg` to the network, __ifal_reserved remained uninitialized, resulting in a 1-byte infoleak:

```
BUG: KMSAN: kernel-network-infoleak in __netdev_start_xmit ./include/linux/netdevice.h:4841
__netdev_start_xmit ./include/linux/netdevice.h:4841
netdev_start_xmit ./include/linux/netdevice.h:4857
xmit_one net/core/dev.c:3590
dev_hard_start_xmit+0x1dc/0x800 net/core/dev.c:3606
__dev_queue_xmit+0x17e8/0x4350 net/core/dev.c:4256
dev_queue_xmit ./include/linux/netdevice.h:3009
__netlink_deliver_tap_skb net/netlink/af_netlink.c:307
__netlink_deliver_tap+0x728/0xad0 net/netlink/af_netlink.c:325
netlink_deliver_tap net/netlink/af_netlink.c:338
__netlink_sendskb net/netlink/af_netlink.c:1263
netlink_sendskb+0x1d9/0x200 net/netlink/af_netlink.c:1272
netlink_unicast+0x56d/0xf50 net/netlink/af_netlink.c:1360
nlmsg_unicast ./include/net/netlink.h:1061
rtnl_unicast+0x5a/0x80 net/core/rtnetlink.c:758
ip6addrlbl_get+0xfad/0x10f0 net/ipv6/addrlabel.c:628
rtnetlink_rcv_msg+0xb33/0x1570 net/core/rtnetlink.c:6082
...
Uninit was created at:
slab_post_alloc_hook+0x118/0xb00 mm/slab.h:742
slab_alloc_node mm/slub.c:3398
__kmem_cache_alloc_node+0x4f2/0x930 mm/slub.c:3437
__do_kmalloc_node mm/slab_common.c:954
__kmalloc_node_track_caller+0x117/0x3d0 mm/slab_common.c:975
kmalloc_reserve net/core/skbuff.c:437
__alloc_skb+0x27a/0xab0 net/core/skbuff.c:509
alloc_skb ./include/linux/skbuff.h:1267
nlmsg_new ./include/net/netlink.h:964
ip6addrlbl_get+0x490/0x10f0 net/ipv6/addrlabel.c:608
rtnetlink_rcv_msg+0xb33/0x1570 net/core/rtnetlink.c:6082
netlink_rcv_skb+0x299/0x550 net/netlink/af_netlink.c:2540
rtnetlink_rcv+0x26/0x30 net/core/rtnetlink.c:6109
netlink_unicast_kernel net/netlink/af_netlink.c:1319
netlink_unicast+0x9ab/0xf50 net/netlink/af_netlink.c:1345
netlink_sendmsg+0xebc/0x10f0 net/netlink/af_netlink.c:1921
...
```

This patch ensures that the reserved field is always initialized.

Reported-by: syzbot+3553517af6020c4f2813f1003fe76ef3cbffe98d@syzkaller.appspotmail.com

Fixes: 2a8cc6c89039 ("[IPV6] ADDRCONF: Support RFC3484 configurable address selection policy table.")

Signed-off-by: Alexander Potapenko <glider@google.com>
Reviewed-by: David Ahern <dsahern@kernel.org>
Signed-off-by: David S. Miller <davem@davemloft.net>

Diffstat

-rw-r--r-- net/ipv6/addrlabel.c 1

1 files changed, 1 insertions, 0 deletions

diff --git a/net/ipv6/addrlabel.c b/net/ipv6/addrlabel.c

index 8a22486cf27020..17ac45aa7194ce 100644

--- a/net/ipv6/addrlabel.c

+++ b/net/ipv6/addrlabel.c

```
@@ -437,6 +437,7 @@ static void ip6addrlbl_putmsg(struct nlmsg_hdr *nlh,
 {
     struct ifaddrlblmsg *ifal = nlmsg_data(nlh);
     ifal->ifal_family = AF_INET6;
+    ifal->__ifal_reserved = 0;
     ifal->ifal_prefixlen = prefixlen;
     ifal->ifal_flags = 0;
     ifal->ifal_index = ifindex;
```