

Open5gs bug report3

☆ 0 stars 🍴 0 forks

☆ Star

🔔 Notifications

<> Code ⌚ Issues 🔗 Pull requests ▶ Actions 📁 Projects ⚠ Security 📈 Insights

🔑 main ▾

Go to file



ToughRunner Update README.md ...

on Oct 10 ⌚ 7

[View code](#)

☰ README.md

Open5gs - A memory leak in PFCP protocol processing crashes UPF causing DoS

Recently, we discovered a logic vulnerability that may cause Open5gs UPF to crash during a code audit of Open5gs Ver2.4.11. The specific causes of the vulnerability are as follows:

Vulnerability description

When processing PFCP packet, a memory leak in UPF `src/upf/pfcp-path.c` from open5gs causing a DoS vulnerability.

UPF pfcp-path

Function `pfcp_rcv_cb` from `src/upf/pfcp-path.c` will be called when receiving pfcp connection.

`src/upf/pfcp-path.c`

```
static void pfcp_rcv_cb(short when, ogs_socket_t fd, void *data)
{
```

...

`pfc_pnode` will be allocated by calling `ogs_pfc_pnode_add`.

`src/upf/pfc-path.c`

```
node = ogs_pfc_pnode_find(&ogs_pfc_self()->pfc_peer_list, &from);
if (!node) {
    node = ogs_pfc_pnode_add(&ogs_pfc_self()->pfc_peer_list, &from);
    ogs_assert(node);

    node->sock = data;
    pfc_pnode_fsm_init(node, false);
}
...
```

`pfc_pnode` is allocated from `ogs_pfc_pnode_pool` and appended to `pfc_peer_list` in `ogs_pfc_pnode_add`.

`lib/pfc/context.c`

```
ogs_pfc_pnode_t *ogs_pfc_pnode_new(ogs_sockaddr_t *sa_list)
{
    ogs_pfc_pnode_t *node = NULL;

    ogs_assert(sa_list);

    ogs_pool_alloc(&ogs_pfc_pnode_pool, &node);
    ogs_assert(node);
    memset(node, 0, sizeof(ogs_pfc_pnode_t));

    node->sa_list = sa_list;

    ogs_list_init(&node->local_list);
    ogs_list_init(&node->remote_list);

    ogs_list_init(&node->gtpu_resource_list);

    return node;
}

ogs_pfc_pnode_t *ogs_pfc_pnode_add(
    ogs_list_t *list, ogs_sockaddr_t *addr)
{
    ogs_pfc_pnode_t *node = NULL;
```

```

ogs_sockaddr_t *new = NULL;

ogs_assert(list);
ogs_assert(addr);

ogs_assert(OGS_OK == ogs_copyaddrinfo(&new, addr));
node = ogs_pfcpc_node_new(new);

ogs_assert(node);
memcpy(&node->addr, new, sizeof node->addr);

ogs_list_add(list, node);

return node;
}

```

Instead of freeing the nodes after using or encountering an error, these nodes are freed only after the termination of UPF by calling function `ogs_pfcpc_context_final`.

So making more than 64 pfcpc connections will crash the UPF causing DoS.

ogs_pfcpc_node_pool

The size of `ogs_pfcpc_node_pool` is defined as 64.

`lib/app/ogs-context.c`

```

#define MAX_NUM_OF_UE          1024    /* Num of UEs */
#define MAX_NUM_OF_PEER        64      /* Num of Peer */

self.max.ue = MAX_NUM_OF_UE;
self.max.peer = MAX_NUM_OF_PEER;

static void recalculate_pool_size(void)
{
    ...
    self.pool.nf = self.max.peer;
    ...
}

```

`lib/pfcpc/context.c`

```

ogs_pool_init(&ogs_pfcpc_node_pool, ogs_app()->pool.nf);

```

POC

The vulnerability can be triggered simply by sending more than 64 invalid pfcp packets through different sockets.

```
upf | 09/16 08:30:02.618: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:34815 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:02.618: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:03.634: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:54635 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:03.634: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:04.654: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:41418 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:04.654: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:05.642: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:47755 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:05.642: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:06.729: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:59306 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:06.729: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:07.671: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:37689 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:07.671: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
scscf | 5(52) DEBUG: ims_dialog [dlg_handlers.c:1923]: print_all_dlg() ***** 5(52) DEBUG: ims_dialog [dlg_handlers.c:1924]: print_all_dlg(): printing 4096 dialogs
scscf | 5(52) DEBUG: ims_dialog [dlg_handlers.c:1934]: print_all_dlg() ***** 5(52) DEBUG: ims_auth [authorize.c:187]: reg_wait_timer(): Looking for expired/unused at 57532221
scscf | 5(52) DEBUG: ims_auth [authorize.c:232]: reg_wait_timer(): [DONE] Looking for expired/unused at 57532221
upf | 09/16 08:30:08.713: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:51233 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:08.713: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:09.758: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:68333 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:09.758: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:10.760: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:40534 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:10.760: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:11.829: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:54263 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:11.829: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:12.865: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:47995 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:12.865: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:13.774: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:33681 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:13.774: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:14.798: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:46665 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:14.798: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:15.892: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:60567 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:15.892: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:16.917: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:41650 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:16.917: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:17.968: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:33493 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:17.968: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
scscf | 5(52) DEBUG: ims_usrlc_scscf [ims_usrlc_scscf_mod.c:481]: timer(): Syncing cache
scscf | 5(52) DEBUG: ims_usrlc_scscf [udomain.c:291]: mem_timer_udomain(): *** mem_timer_udomain - checking contacts - START ***
scscf | 5(52) DEBUG: ims_usrlc_scscf [udomain.c:353]: mem_timer_udomain(): *** mem_timer_udomain - checking contacts - FINISHED ***
scscf | 5(52) DEBUG: ims_usrlc_scscf [udomain.c:359]: mem_timer_udomain(): *** mem_timer_udomain - checking IMPUS - START ***
scscf | 5(52) DEBUG: ims_usrlc_scscf [udomain.c:384]: mem_timer_udomain(): *** mem_timer_udomain - checking IMPUS - FINISHED ***
scscf | 5(52) DEBUG: ims_dialog [dlg_handlers.c:1923]: print_all_dlg() ***** 5(52) DEBUG: ims_dialog [dlg_handlers.c:1924]: print_all_dlg(): printing 4096 dialogs
scscf | 5(52) DEBUG: ims_dialog [dlg_handlers.c:1934]: print_all_dlg() ***** 5(52) DEBUG: ims_auth [authorize.c:187]: reg_wait_timer(): Looking for expired/unused at 57532231
scscf | 5(52) DEBUG: ims_auth [authorize.c:232]: reg_wait_timer(): [DONE] Looking for expired/unused at 57532231
upf | 09/16 08:30:18.995: [pfcp] INFO: ogs_pfcpc_connect() [172.22.0.1]:38214 (../lib/pfcp/path.c:61)
upf | 09/16 08:30:18.995: [upf] WARNING: cannot handle PFPC message type[1] (../src/upf/pfcp-sm.c:139)
upf | 09/16 08:30:20.057: [pfcp] FATAL: ogs_pfcpc_node_new: Assertion 'node' failed. (../lib/pfcp/context.c:642)
upf | 09/16 08:30:20.057: [core] FATAL: backtrace() returned 9 addresses (../lib/core/ogs-abort.c:37)
upf | /openSgs/install/lib/x86_64-linux-gnu/libogs-pfcp.so.2(ogs_pfcpc_node_new+0x1b3) [0x7f8def6d083c]
upf | /openSgs/install/lib/x86_64-linux-gnu/libogs-pfcp.so.2(ogs_pfcpc_node_add+0x181) [0x7f8def6d0b98]
upf | ./openSgs-upfd(+0x118c2) [0x55c0f7d6b8c2]
upf | /openSgs/install/lib/x86_64-linux-gnu/libogs-core.so.2(+0x24c43) [0x7f8def5f1c43]
upf | ./openSgs-upfd(+0x72e0) [0x55c0f7d612e0]
upf | /openSgs/install/lib/x86_64-linux-gnu/libogs-core.so.2(+0x117e5) [0x7f8def5de7e5]
upf | /lib/x86_64-linux-gnu/libpthread.so.0(+0x8609) [0x7f8def2c6609]
upf | /lib/x86_64-linux-gnu/libc.so.6(clone+0x43) [0x7f8def1eb133]
upf | ./openSgs_init.sh: line 96: 58 Aborted (core dumped) ./openSgs-upfd
upf exited with code 134
coredump | coredump
```

Update

We have reported this vulnerability to the vendor through email at 19 Sep 2022, but this bug has not been fixed yet.

Acknowledgment

Credit to @ToughRunner, @HenryzhaoH, @leonW7 from Shanghai Jiao Tong University.

Releases

No releases published

Packages

No packages published

