# strongSwan - Bug #2373

## farp ignores the IP addresses of a road warrior

30.06.2017 09:51 - Harald Dunkel

| | | | |
|---|---|---|---|
| **Status:** | Closed | **Start date:** | |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | Tobias Brunner | **Estimated time:** | 0.00 hour |
| **Category:** | libcharon | | |
| **Target version:** | 5.6.0 | | |
| **Affected version:** | 5.5.3 | **Resolution:** | Fixed |

**Description**

Sometimes farp ignores the arp requests for one (or more?)
IP address bound to a child SA.

I saw this on my IPsec gateway, for example:

```
# ipsec statusall
:
 IPSec-IKEv2[2772]: ESTABLISHED 85 minutes ago, 2001:db8:13b0:ffff::63[gate.example.com]...2001:db
8:30:fff0:ed29:6621:7cf8:6387[ppcm005.example.de]
 IPSec-IKEv2[2772]: IKEv2 SPIs: b6cf4a195efbe943_i 90855da73ac9b14c_r*, public key reauthenticatio
n in 22 hours
 IPSec-IKEv2[2772]: IKE proposal: AES_CBC_256/HMAC_SHA2_256_128/PRF_HMAC_SHA2_256/MODP_2048
 IPSec-IKEv2{4107}:  INSTALLED, TUNNEL, reqid 2323, ESP SPIs: c68adec6_i 0bf997dc_o
 IPSec-IKEv2{4107}:  AES_CBC_256/HMAC_SHA2_256_128/MODP_2048, 741797 bytes_i (5250 pkts, 0s ago),
1479117 bytes_o (1653 pkts, 1666s ago), rekeying in 21 minutes
 IPSec-IKEv2{4107}:   x.xxx.142.192/26 10.47.11.0/24 yy.yy.169.96/27 10.19.96.0/19 10.22.111.0/24
10.23.15.0/24 zzz.zz.32.0/27 === 10.19.97.55/32
:
```

The IP address bound to the peer is 10.19.97.55 in this case. tcpdump
shows the incoming arp request, but they are not answered:

```
# tcpdump -i eth1 -env arp and host 10.19.97.55
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
15:12:17.654382 00:16:5a:xx:ce:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.124, length 46
15:12:17.785501 00:20:8c:xx:51:83 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.11, length 46
15:12:18.805539 00:20:8c:xx:51:83 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.11, length 46
15:12:19.869832 00:1e:67:xx:9b:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.98.253, length 46
15:12:20.677828 00:1e:67:xx:9b:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.98.253, length 46
15:12:20.826525 00:20:8c:xx:51:83 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.11, length 46
15:12:21.676258 00:1e:67:xx:9b:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.98.253, length 46
15:12:21.845542 00:20:8c:xx:51:83 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.11, length 46
15:12:22.869629 00:20:8c:xx:51:83 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.11, length 46
15:12:23.611081 00:16:5a:xx:ce:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.124, length 46
15:12:24.610385 00:16:5a:xx:ce:a9 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: Ethernet
 (len 6), IPv4 (len 4), Request who-has 10.19.97.55 tell 10.19.96.124, length 46
^C
11 packets captured
12 packets received by filter
0 packets dropped by kernel
```

Farp answers the arp requests for the other road warrior IP addresses (not shown here). "arp" on the gateway shows "incomplete" for all road warrior IP addresses.

The problem in this example came up after more than 70 minutes uptime of the IPsec connection to this peer (AFAIK). The peer is a Mac. DPD packets are sent and received in the expected regular intervals.

By now I haven't found a way to reproduce this problem, but we see it once or twice per week or so. The affected road warriors are pissed because they don't like to reset their network connection, but usually this helps.

### Associated revisions

**Revision 6138b8d6 - 27.07.2017 13:07 - Tobias Brunner**

farp: Only remove one tracked entry

Multiple CHILD_SAs sharing the same traffic selectors (e.g. during make-before-break reauthentication) also have the same reqid assigned. If all matching entries are removed we could end up without entry even though an SA exists that still uses these traffic selectors.

Fixes #2373.

### History

**#1 - 30.06.2017 11:53 - Tobias Brunner**

*- Tracker changed from Bug to Issue*

*- Category set to libcharon*

*- Status changed from New to Feedback*

> Farp answers the arp requests for the other road warrior IP addresses (not shown here).

That could have been interesting, though.

> The problem in this example came up after more than 70 minutes uptime of the IPsec connection to this peer (AFAIK). The peer is a Mac. DPD packets are sent and received in the expected regular intervals.

Perhaps an issue with rekeying and tracking the SAs.

> By now I haven't found a way to reproduce this problem, but we see it once or twice per week or so.

Then again, if it only occurs so rarely it might be something else.

You might have to build a customized version of the farp plugin with some log messages added to find out what happens exactly (or use gdb to debug it). For instance, is the packet received at all by the packet socket? If so, is there a tracked CHILD_SA with matching remote traffic selector found? And if that's the case, is there a response sent?

**#2 - 04.07.2017 13:37 - Harald Dunkel**

Attached you can find the logging patch for farp_spoofer.c . Do you expect any problems with this (besides being slow)?

Maybe you would like to add it with DBG3 or DBG4 to the official source tree.

**#3 - 04.07.2017 14:48 - Tobias Brunner**

> Attached you can find the logging patch for farp_spoofer.c

Hm, I don't see any attachments.

#### #4 - 05.07.2017 12:55 - Harald Dunkel

*- File farp_debug.diff added*

now there is (hopefully)

#### #5 - 05.07.2017 14:27 - Tobias Brunner

> Do you expect any problems with this (besides being slow)?

Looks OK. The overhead shouldn't be that problematic (unless you have a huge amount of ARP requests every second). Maybe you could also add an else block with a log message for the len == sizeof(arp) check (in case invalid data is read from the socket). And in send_arp() the ioctl() calls could potentially fail, so either add an else block there too, or add a log message if sending the response was successful.

> Maybe you would like to add it with DBG3 or DBG4 to the official source tree.

Maybe.

#### #6 - 05.07.2017 15:04 - Harald Dunkel

The suggested changes are not in yet, but we saw this problem a few minutes ago again. Is it OK to send you log files per EMail? I wouldn't like to see all these IPv6 and mac and email addresses in the internet for everybody to scan. Obfuscating these log files would be a **huge** effort.

Apparently the

```
if (this->listener->has_tunnel(this->listener, local, remote))
{
    DBG1(DBG_NET, "arp received, sender is %#H, target is %#H", local, remote);
    send_arp(this, &arp, &addr);
} else {
    DBG1(DBG_NET, "arp ignored, sender is %#H, target is %#H", local, remote);
}
```

ran into the else branch, even though "swanctl --list-sas" shows IKE and child SAs for the IPv4 address.

#### #7 - 05.07.2017 15:08 - Tobias Brunner

> Is it OK to send you log files per EMail?

Sure.

> ran into the else branch, even though "swanctl --list-sas" shows IKE and child SAs for the IPv4 address.

Interesting. Was there a rekeying before for these SAs?

#### #8 - 18.07.2017 11:53 - Tobias Brunner

*- Tracker changed from Issue to Bug*

*- Assignee set to Tobias Brunner*

*- Target version set to 5.6.0*

*- Resolution set to Fixed*

Further analysis showed that the problem was caused when a peer had more than one CHILD_SA with the same traffic selectors/reqid established concurrently (e.g. one via IPv4 and one via IPv6, but make-before-break reauthentication is also affected). If that's the case and one of the SAs is terminated all the state in the *farp* plugin, which is bound to the reqid, was removed, even though state was added for each created CHILD_SA. A fix can be found in the *2373-farp-fix* branch.

#### #9 - 08.08.2017 17:59 - Tobias Brunner

*- Status changed from Feedback to Closed*

### Files

| | | | |
|---|---|---|---|
| ipsec.conf | 2.85 KB | 30.06.2017 | Harald Dunkel |