

---

# Fixing IPv6 on Hetzner Cloud: the story of a lifetime

Published by

Stanislas

Estimated reading time

7 min

Added on

2024-08-12

Address

<https://stanislas.blog/2019/08/fix-ipv6-hetzner-cloud/>

# Fixing IPv6 on Hetzner Cloud: the story of a lifetime



IPv6 works out of the box on [Hetzner Cloud](#) VMs. However, once you start adding IPv6 to an existing or new interface, IPv6 connectivity completely falls apart.

I started to notice this [a year and half ago](#), when I was trying to setup LXDM. With the help of a [fellow sysadmin](#), we tried to hunt down the issue, without success, but found a dirty workaround.

Then it was reported again on my [openvpn-install](#) script, and more recently on my [wireguard-install](#) script, so I decided to dive into this issue once more.

## Scope of the issue

This issue affects the Debian and Ubuntu “ready-to-go” images made by Hetzner, not CentOS or Fedora. It does not affect ISOs or installations made with [installimage](#) in rescue mode.

## Issue explanation

Using tcpdump, I noticed that when adding another inet6, i.e. another IPv6 network to an existing or new interface, the source IPv6 changes from the usual public address to the one that was added.

Here is a ping6 to Google viewed by tcpdump:

```
IP6 (flowlabel 0xbdb1d, hlim 64, next-header ICMPv6 (58) payload length: 64) 2a01:4f8:c010:1031::1 > fra16s25-in-x0e.1e100.net: [icmp6 sum ok] ICMP6, echo request, seq 18
IP6 (flowlabel 0xbdb1d, hlim 54, next-header ICMPv6 (58) payload length: 64) fra16s25-in-x0e.1e100.net > 2a01:4f8:c010:1031::1: [icmp6 sum ok] ICMP6, echo reply, seq 18
```

---

The ICMP packet is correctly emitted and going back.

Now, let's say I add a new WireGuard interface with an IPv6 address (fd42:42:42::1). Here's the output:

```
IP6 (flowlabel 0x1e23e, hlim 64, next-header ICMPv6 (58) payload length: 6
4) fd42:42:42::1 > fra16s25-in-
x0e.1e100.net: [icmp6 sum ok] ICMP6, echo request, seq 17
```

Now, the source IPv6 is my WireGuard private IPv6, hence the ping not coming back.

At this point, public IPv6 connectivity is completely broken.

The source address was confirmed by:

```
root@debian-2gb-fsn1-1:~# ip route get 2a00:1450:4001:820::200e
2a00:1450:4001:820::200e from :: via fe80::1 dev eth0 src fd42:42:42::1 me
tric 1024 pref medium
```

Although it doesn't make sense by looking at the routes:

```
root@debian-2gb-fsn1-1:~# ip -6 r
::1 dev lo proto kernel metric 256 pref medium
2a01:4f8:c010:1031::/64 dev eth0 proto kernel metric 256 pref medium
fd42:42:42::/64 dev wg0 proto kernel metric 256 pref medium
fe80::/64 dev eth0 proto kernel metric 256 pref medium
default via fe80::1 dev eth0 metric 1024 onlink pref medium
```

The actual issues is visible here:

```
root@debian-2gb-nbg1-1:~# ip -6 -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 state UNKNOWN qlen 1000
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 state UP qlen 1000
   inet6 2a01:4f8:c2c:8ebe::1/64 scope global deprecated
       valid_lft forever preferred_lft 0sec
   inet6 fe80::9400:ff:fe2d:532c/64 scope link
       valid_lft forever preferred_lft forever
3: wg0: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1420 state UNKNOWN qlen 1000
   inet6 fd42:42:42::1/64 scope global
       valid_lft forever preferred_lft forever
```

---

To enhance privacy, IPv6 addresses can be auto-generated and renewed client-side without a DHCP server, according to [RFC 3041](#), at least that's my understanding of it.

Two values are important here:

- `preferred_lft`: number of seconds until the address is deprecated and another address is added
- `valid_lft`: number of seconds until the deprecated address is removed

As we can see here, our public IPv6 address has, **by default**, a `valid_lft` of forever and a `preferred_lft` of 0sec.

Which means by the time the VM has booted, the main and only IPv6 (besides the loopback) is already deprecated. Though, another IPv6 is not added so I guess the mechanism described in RFC 3041 needs some configuration.

Even though the address is deprecated, it works perfectly, and that's why most users won't see any issues with IPv6 connectivity.

But why does it work? According to [davidc](#):

non-deprecated address(es) will be favored

So if only one address is present, it is still favored. However once we add another `inet6`, which will not be deprecated by default if configured correctly, this `inet6` becomes the favored address, thus all packets will use it as the `src` address. That explains the example above with WireGuard.

But the real question is: why does our public address have a `preferred_lft` of 0sec by default?

I got the answer by exposing my issue on [ServerFault](#).

Here how the interfaces are configured:

```
root@debian-2gb-nbg1-1:~# cat /etc/network/interfaces.d/50-cloud-init.cfg
# This file is generated from information provided by
# the datasource. Changes to it will not persist across an instance.
# To disable cloud-init's network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
    dns-nameservers 213.133.99.99 213.133.98.98 213.133.100.100
auto eth0:0
iface eth0:0 inet6 static
    address 2a01:4f8:c2c:6daa::1/64
    gateway fe80::1
    post-up route add -net :: netmask 0 gw fe80::1%eth0 || true
```

---

```
pre-down route del -net :: netmask 0 gw fe80::1%eth0 || true
```

Notice the eth0:0. Apparently this is called a virtual interface, but it makes no sense here.

This is the issue. When attaching the IPv6 to eth0 directly, the preferred\_lft is forever. When using eth0:0, it's 0sec. I haven't figured the relation between the virtual interface and the preferred\_lft. If anyone knows why, I'm all ears!

By the way this is why it only affects Debian and Ubuntu, since both use ifupdown on Hetzner's images.

During my research I found that a quick temporary fix was the following:

```
ip -6 addr change <ipv6>/64 dev eth0 preferred_lft forever
# or even simpler, for some reason
ip -6 addr change <ipv6>/64 dev eth0
```

This sets the preferred\_lft to forever and the address is not deprecated anymore.

## Permanent fix

To fix this issue across reboots, the network configuration has to be updated, by moving the IPv6 off the virtual interface to the main one.

This:

```
root@debian-2gb-nbg1-1:~# cat /etc/network/interfaces.d/50-cloud-init.cfg
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
    dns-nameservers 213.133.99.99 213.133.98.98 213.133.100.100
auto eth0:0
iface eth0:0 inet6 static
    address 2a01:4f8:c2c:6daa::1/64
    gateway fe80::1
    post-up route add -net :: netmask 0 gw fe80::1%eth0 || true
    pre-down route del -net :: netmask 0 gw fe80::1%eth0 || true
```

Becomes this:

```
root@debian-2gb-nbg1-1:~# cat /etc/network/interfaces.d/50-cloud-init.cfg
auto lo
iface lo inet loopback
```

---

```
auto eth0
iface eth0 inet dhcp
    dns-nameservers 213.133.99.99 213.133.98.98 213.133.100.100
iface eth0 inet6 static
    address 2a01:4f8:c2c:6daa::1/64
    gateway fe80::1
    post-up route add -net :: netmask 0 gw fe80::1%eth0 || true
    pre-down route del -net :: netmask 0 gw fe80::1%eth0 || true
```

After a reboot, the preferred\_lft will be forever by default. No issues anymore, with LXD, OpenVPN, WireGuard... ?

## Hetzner will fix this soon

I first reached to them about the preferred\_lft on 02/2018, and they answered:

we are not setting this value from our side. If you look at the network configuration, just the IP is configured, nothing else.

Which is technically true!

After gathering more information and finding that the issue was the virtual interface, I reached out to them again yesterday and they said:

this issue is caused by the way cloudinit is configuring the network interfaces. We are already working on a fix that will configure the IPv6 address correctly. Currently we can't say when this will be released.

Hopefully this will be fixed by default in the future!

At least I got to learn more about IPv6. By the way, please correct me if I said something incorrect in this blog post, I'm not a network engineer.

---

Produced by wallabag with tcpdf

Please open [an issue](#) if you have trouble with the display of this E-Book on your device.