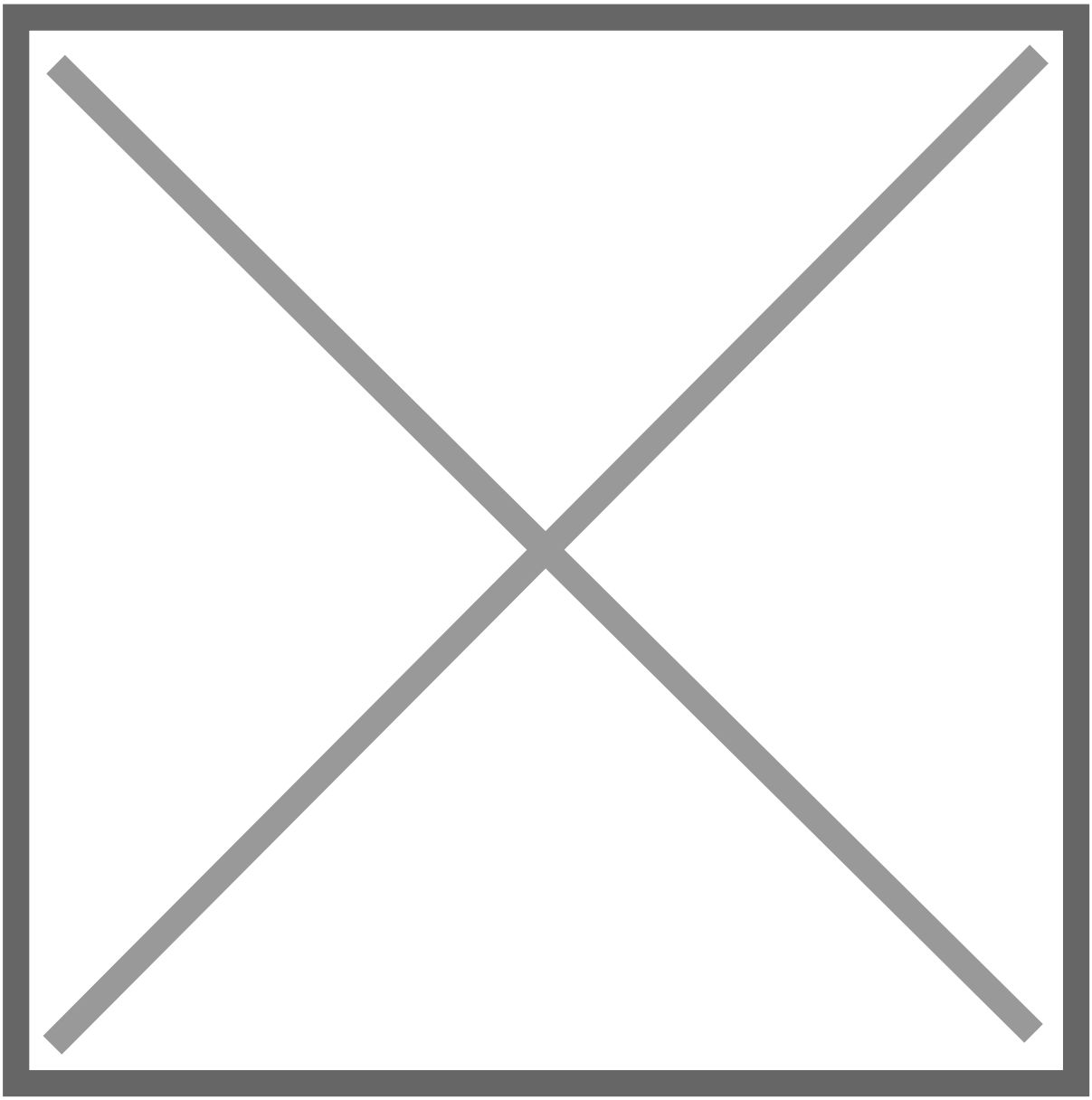
Pi-hole Docker Howto

Solved: Error starting userland proxy: listen tcp4 0.0.0.0:53: bind: address already in use

Published by

Problem

While configuring the Duo Network Gateway (DNG) on a Ubuntu device for RDP, I came across the following error while trying to configure a DNS service:



The error is presented because the Ubuntu system is using the UDP port 53 (DNS). This is because there is a network name resolution service called systemd-resolved running by default. This service provides name resolution to local applications using the loopback IP of the device and acts by default as a DNS stub listener. It can also validate DNS/DNSSEC and can be configured for Link-Local Multicast Name Resolution (LLMNR) which when enabled will become a full LLMNR responder and resolver. There are a few other things that systemd-resolved can do but for this article, we won't discuss those as they're not relevant. You can find out more about systemd-resolved here.

Solution

II've seen articles on the Internet recommending that systemd-resolved should be disabled if you encounter this issue. However, by disabling systemd-resolved, the name resolution will not work, so we need to take another approach. The approach that we're going to take in this article is to

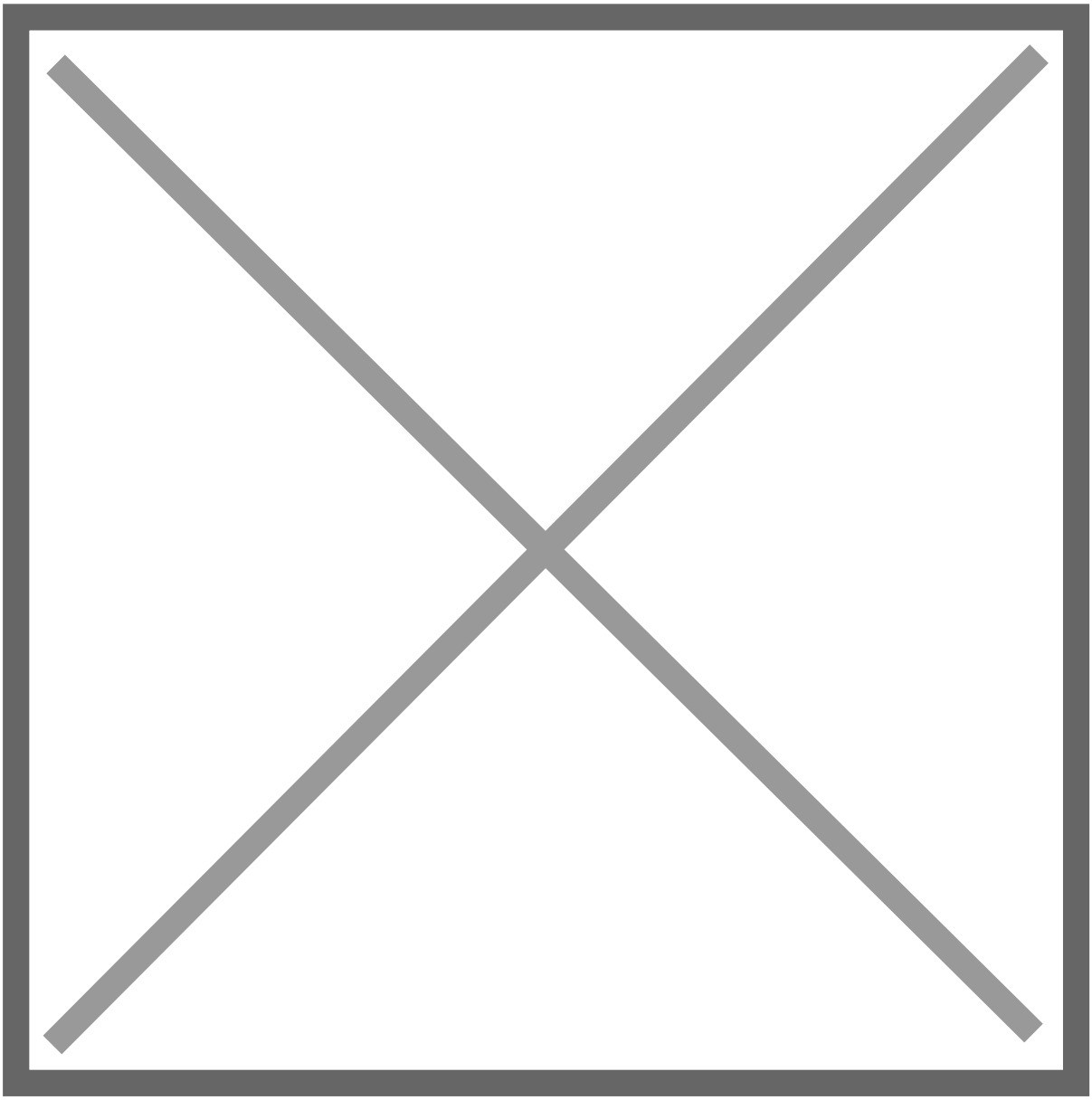
modify the resolved.conf for systemd-resolved. We will modify the configuration so that it no longer listens for DNS requests but rather uses the configured DNS servers for that task.

Verify that port 53 is used on your DNG

Although you have experienced the aforementioned error that more than likely brought you here, let’s not jump the gun! We want to start by checking what is using port 53. Enter the following command to valid port 53 is indeed in use.

```
sudo lsof -i udp:53
```

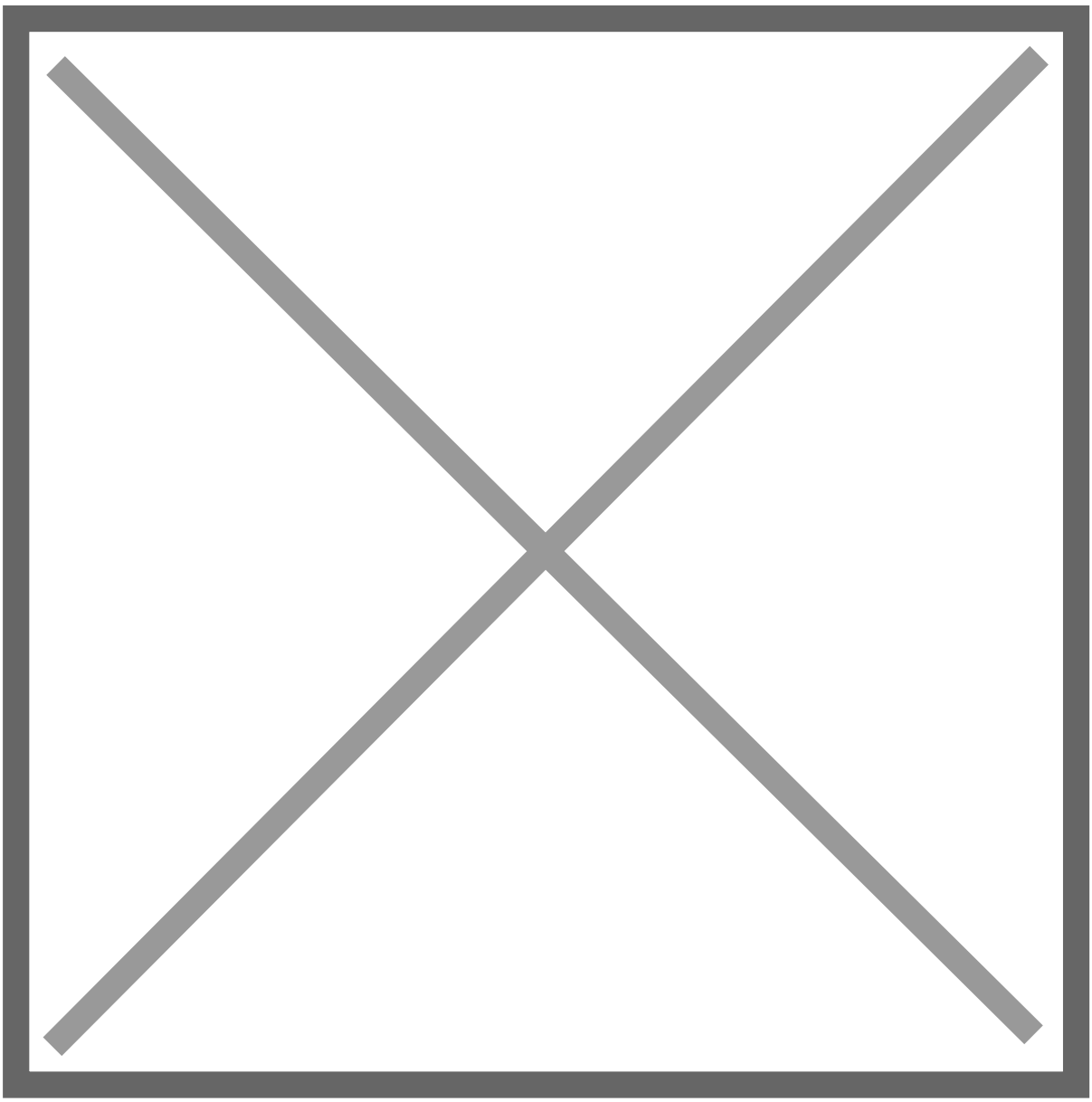
You should recieve a similar output to the one shown in the screenshot below. Here we confirm that systemd-resolved is indeed using UDP 53 (DNS).



We can further validate this by performing an NSLOOKUP to see what the system uses to resolve FQDN’s. Enter the following command.

```
sudo nslookup google.com
```

The results should be similar to the screenshot below. You can see that the DNG is using itself to cache and perform DNS lookups.



If you are using an Ubuntu or Fedora operating system to run the Pi-Hole Docker container, you may need to disable the DNS Stub listener that is built into the Systemd resolve service.

The operating system uses this service to provide network name resolution. As Pi-Hole will want to operate on the same part the resolve service does, we need to disable it.

To start this process, begin editing the "`/etc/systemd/resolved.conf`" configuration file by running the following command.

```
sudo nano /etc/systemd/resolved.conf
```

Copy

8. Within this file, you will want to find the following line. This setting basically allows us to control whether the DNS stub listener is turned on.

```
#DNSStubListener=yes
```

Copy

After finding this line you will want to remove the hashtag (`#`) from the front of this line and change "`yes`" to "`no`".

```
DNSStubListener=no
```

Copy

9. Once you have made this change, save and quit out of the file by pressing CTRL + X, Y, and then ENTER.

10. Our next step is to remove the existing "`resolv.conf`" file since it currently points your system's network to use the now-disabled DNS stub resolver.

You can delete this file using the rm command as shown below.

```
sudo rm /etc/resolv.conf
```

Copy

11. With the existing "`resolv.conf`" file removed, we will now create a symbolic link in its place pointing to the version setting in the "/run/systemd/resolve/" directory.

This version of the file is automatically updated using the DNS servers set within your Netplan.

```
sudo ln -s /run/systemd/resolve/resolv.conf /etc/resolv.conf
```

Copy

12. The last thing we need to do is restart the "`systemd-resolved`" service so that all of our changes will be loaded in.

Once the service has finished restarting, Pi-hole should now be able to utilize the DNS ports on your system.

```
sudo systemctl restart systemd-resolved
```

```
sudo systemctl stop systemd-resolved
nslookup google.com
```

Composefile:

version: "3"

Mehr Infos auf https://github.com/pi-hole/docker-pi-hole/ und https://docs.pi-hole.net/
services:
pihole:
container_name: pihole # Der Name des Containers wie er in Portainer unter "Containers" angezeigt wird
image: pihole/pihole:latest # Download des offiziellen Pihole-Images
hostname: Pihole # Diese Bezeichnung steht später oben rechts in der Weboberfläche von Pihole als Hostname
Für DHCP wird empfohlen, die folgenden Ports zu entfernen und stattdessen hinzuzufügen:
network_mode: "host"
ports:
- "53:53/tcp"
- "53:53/udp"
- "67:67/udp" # Wird nur benötigt, wenn ihr Pi-hole als DHCP-Server nutzen wollt
- "88:80/tcp"
environment:
TZ: 'Europe/Berlin' # Die Zeitzone wird auf Deutschland gestellt
WEBPASSWORD: 'pihole' # Das Passwort, mit dem ihr euch an der Pihole-Weboberfläche anmeldet
PIHOLE_DNS_: '9.9.9.9;149.112.112.112' # Stellt in Pihole "Quad9 (filtered, DNSSEC)" als Upstream-DNS-Server ein statt Google
"Volumes" speichert Daten zwischen Container-Upgrades
volumes:
- './etc-pihole:/etc/pihole'
- './etc-dnsmasq.d:/etc/dnsmasq.d'
#cap_add: # Wird nur benötigt, wenn ihr Pi-hole als DHCP-Server nutzen wollt
- NET_ADMIN # Wird nur benötigt, wenn ihr Pi-hole als DHCP-Server nutzen wollt
restart: unless-stopped # Wenn ihr den Container stoppt, startet er nicht automatisch neu

---

Revision #4
Created 2024-11-18 21:19:56 UTC by willi
Updated 2025-03-13 17:49:32 UTC by willi