Using ProtonMail Bridge in an LXC container

Step 1: Create a LXC container

To create and manage an LXC container, you can use the app "Containers" which is available from Chum. After launching the app, you can create a new container by tapping on the "+" icon. This brings up a page where you give your container a name and decide on one of the available Linux-distros and the corresponding version. This will be setup for you automatically. The "Architecture"-field can (normally) left at the default architecture. Do not check "Setup desktop" as this is not required. You can choose any distro you like as we will use the Nix package manager to install ProtonMail Bridge. Why? Because Proton doesn't provide an *aarch64*-build of Bridge and compiling it from source (in my experience) often doesn't work correctly.

> **ATTENTION:** If you want to use NixOS itself in an LXC container then there are special steps required to run ProtonMail Bridge. Instructions for these steps can be found here

Step 2: Install the Nix package manager

After the container has been created by the app, you need to attach a terminal session to it. This can easily be done by clicking on the container you created and clicking on the button "Attach". This will open a new terminal window where you can execute commands *inside* your container. If you have a terminal session attached, the Nix package manager can be installed by executing (in the attached session):

```
sh <(curl -L https://nixos.org/nix/install) --daemon
```

(more info here: https://nixos.org/download)

> **Note:** Make sure *curl*, *tar* and *xz* are installed for this script to work. If you get an error that it can't find some package, just install the package it says is missing

Step 3: Installing the necessary programs

Next, we will need to install a few programs (with Nix). Obviously, we will need ProtonMail Bridge itself which can be installed with

```
nix-env -iA nixpkgs.protonmail-bridge
```

Then, we will also need pass and screen. These can be either installed via your distro's package manager or via Nix:

```
#Install screen:
nix-env -iA nixpkgs.screen
#Install pass
nix-env -iA nixpkgs.pass
```

Step 4: Setup Bridge

> **Disclaimer**: For this step, I'm following this tutorial, where you can find more information if you want

Step 4a: Setting up pass

Generate a PGP key pair for the new user with an empty passphrase.

The empty passphrase is required to run ProtonMail Bridge on the background on system startup without being prompted for the password and hence causing the process to fail.

```
gpg --full-generate-key
```

Choose 1 for RSA, type in 2048 for the length, choose 0 to make the key not expire, type in your name and your e-mail address, leave comment and passphrase empty (you will be asked for it two times).

Next, you can initialize pass. This can be done by executing

```
pass init <E-Mail address>
```

You need to use the same E-Mail address you typed in previously when generating the key.

Step 4b: Setting up Bridge

For Bridge to work, you will need to execute

```
export HOME=/home/root
```

(the directory must exist, of course).

After that, you can launch Bridge with:

```
protonmail-bridge --cli
```

You are now in the ProtonMail Bridge command line interface. Here, you can add a new account by typing in *add*. This will prompt you for your username and password (and Two-Factor-Token if enabled). After that, it will begin synchronising which can take a while. You than can list the credentials for IMAP and SMTP by typing in *info*. These details you can now use to setup the Jolla E-Mail app. The server URL is *127.0.0.1*. You can type in exit to quit Bridge.

> **Note**: closing the terminal window and/or Container-app does NOT stop the container, so it can conveniently run in the background and receive e-mails without you even noticing.

Step 5: Make Bridge automatically launch (optional)

If you want to start Bridge automatically, you can follow a few simple steps (some things again taken from here)

First, create a basic script which will launch ProtonMail Bridge in the background:

```
mkdir /var/lib/protonmail
nano /var/lib/protonmail/protonmail.sh
```

~~In this file, you can copy the content from here. At the top of the script (after the shebang *#!/bin/bash*), you need to add the *export* statement we used before.~~

> Since this tutorial seems to not be available anymore, I'll post the contents of the script here. But all credits go to the original authors, this is not my work.

```
#!/bin/bash
export HOME=/home/root
```

```
case "$1" in start)
# will create an screen in detached mode (background) with name "protonmail"
/nix/var/nix/profiles/default/bin/screen -S protonmail -dm protonmail-bridge --cli echo "Service started."
;;
status)
# ignore this block unless you understand how screen works and that only lists the current user's screens
result=$(/nix/var/nix/profiles/default/bin/screen -list | grep protonmail)
if [ $? == 0 ]; then
echo "Protonmail bridge service is ON."
else
echo "Protonmail bridge service is OFF."
fi
;;
stop)
# Will quit a screen called "protonmail" and therefore terminate the running protonmail-bridge process
/nix/var/nix/profiles/default/bin/screen -S protonmail -X quit echo "Service stopped."
;;
*) echo "Unknown command: $1" exit 1
;;
esac
```

After that, you need to create a systemd-service that starts and stops ProtonMail Bridge with the container. This can be done with:

```
nano /etc/systemd/system/protonmail.service
```

~~In there, copy the content from here~~

> Again since this tutorial is not available anymore, I'll post the content here but it's not my work and the credits go to the original authors

```
[Unit]
Description=Service to run the Protonmail bridge client
After=network.target

[Service]
Type=oneshot
User=root
ExecStart=/var/lib/protonmail/protonmail.sh start
ExecStop=/var/lib/protonmail/protonmail.sh stop
```

```
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

All you have to do now is to enable the service with

```
#Enable autostart
systemctl enable protonmail
#Start the service manually
systemctl start protonmail
```

Now ProtonMail Bridge will start every time you start your container. If you want, you can also configure the app "Containers" to automatically launch after every boot with the app "Takeoff" available from OpenRepos)

Revision #4
Created 15 December 2023 18:17:26 by NIS
Updated 30 April 2026 18:37:15 by NIS