

1. Notification Services

This chapter discusses all kinds of notifications you can receive on your phone or laptop regarding your home lab. You want to be notified when a hard drive fails, when a disk is almost full, when backup failed, when a new update is available or any other alert. This will provide you with the ease of use of your home lab. Once it is running it is running and you only need to pay attention again to your homelab when you get a notification ;)

We will focus on two main notification strategies which will be **NTFY** and **Email**, but there are bunch of others out there as well. Maybe we will add these as well to this book later down the road. As always it is up to you which ones you prefer and if you can't choose you can also do both obviously.

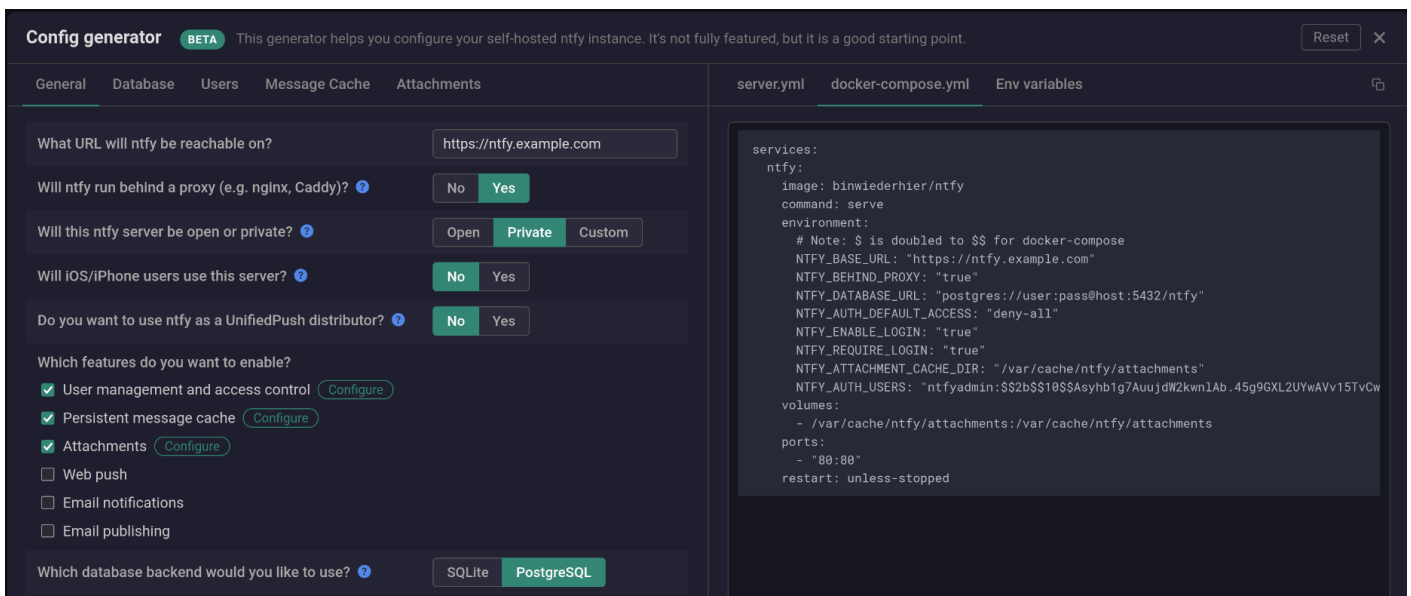
- [1. NTFY](#)
- [2. Classic Email SMTP Setup](#)

1. NTFY

Docker VM Deployment via Dockhand

Ntfy is a great notification application that provides you with alarms on your phone or laptop if anything happens inside your infrastructure. When resource limits are reached or when a new update for a container is available. It is really handy to have instead of using your Email for all your notifications. In this way your personal email can stay nice and clean and is only used for sending email and not being bombarded with notifications that clogs up your inbox.

Some self hosted applications integrate NTFY natively and others you can add it via *Webhooks*. Now just as with the other docker vm applications we will deploy NTFY with a simple docker compose file. Now, installation is very straightforward by using their [Configuration Generator](#).



Config generator BETA This generator helps you configure your self-hosted ntfy instance. It's not fully featured, but it is a good starting point. Reset ✕

General Database Users Message Cache Attachments

What URL will ntfy be reachable on?

Will ntfy run behind a proxy (e.g. nginx, Caddy)? ? No Yes

Will this ntfy server be open or private? ? Open Private Custom

Will iOS/iPhone users use this server? ? No Yes

Do you want to use ntfy as a UnifiedPush distributor? ? No Yes

Which features do you want to enable?

- ☒ User management and access control Configure
- ☒ Persistent message cache Configure
- ☒ Attachments Configure
- ☐ Web push
- ☐ Email notifications
- ☐ Email publishing

Which database backend would you like to use? ? SQLite PostgreSQL

server.yml **docker-compose.yml** Env variables 📄

```
services:
  ntfy:
    image: binwiederhier/ntfy
    command: serve
    environment:
      # Note: $ is doubled to $$ for docker-compose
      NTFY_BASE_URL: "https://ntfy.example.com"
      NTFY_BEHIND_PROXY: "true"
      NTFY_DATABASE_URL: "postgres://user:pass@host:5432/ntfy"
      NTFY_AUTH_DEFAULT_ACCESS: "deny-all"
      NTFY_ENABLE_LOGIN: "true"
      NTFY_REQUIRE_LOGIN: "true"
      NTFY_ATTACHMENT_CACHE_DIR: "/var/cache/ntfy/attachments"
      NTFY_AUTH_USERS: "ntfyadmin:$2b$5$10$SAsyhb1g7AuujdW2kwn1Ab.45g9GXL2UYwAVv15TvCw"
    volumes:
      - /var/cache/ntfy/attachments:/var/cache/ntfy/attachments
    ports:
      - "80:80"
    restart: unless-stopped
```

The above screenshot is how I have set it up, which is fairly basic. I don't need Web Push, Email notifications, nor Email pushing but that are possibilities if you want to. Also by default an NTFY server is public and anyone can write and read from topics. For most instances you would like to have some form of authentication to prevent random people from flooding your topics. Hence, you can make your instance private. Finally, I would recommend to choose a different port then port 80 since it is most often already used by another application in your Docker VM.

Users

Now, if your instance is private you also need to create users to login to your instance if they want to interact with it. Also this is easily done in the [Configuration Editor](#)

Config generator

BETA

This generator helps you configure your self-hosted ntfy instance. It's not fully featured, but it is a good starting point.

Reset

✕

General Database Users Message Cache Attachments

server.yml docker-compose.yml Env variables

Configure user management, access control, and provisioned users/ACLs. See [access control](#) for details.

Where should the user database be stored?

Using PostgreSQL

What should the default access policy be?

Deny All

Should login to the web app be enabled?

Disabled Enabled Required

Should it be possible to sign up via the web app?

No Yes

Provisioned users

ntfyadmin h6BqKEXhA0PtIUdu Admin

ntfyuser sqNyDztnE0mTw0ot User

+ Add user

Provisioned ACLs

+ Add ACL

Provisioned tokens

+ Add token

```

services:
  ntfy:
    image: binwiederhier/ntfy
    command: serve
    environment:
      # Note: $ is doubled to $$ for docker-compose
      NTFY_BASE_URL: "https://ntfy.example.com"
      NTFY_BEHIND_PROXY: "true"
      NTFY_DATABASE_URL: "postgres://user:pass@host:5432/ntfy"
      NTFY_AUTH_DEFAULT_ACCESS: "deny-all"
      NTFY_ENABLE_LOGIN: "true"
      NTFY_REQUIRE_LOGIN: "true"
      NTFY_ATTACHMENT_CACHE_DIR: "/var/cache/ntfy/attachments"
      NTFY_AUTH_USERS: "ntfyadmin:$$2b$$10$$Asyhb1g7AuujdW2kwn1Ab.45g9GXL2UYwAVv15TvCw"
    volumes:
      - /var/cache/ntfy/attachments:/var/cache/ntfy/attachments
    ports:
      - "80:80"
    restart: unless-stopped

```

Here you can define as many users and admins that you want up front. Now, after you have created your instance and you want to add more users to it later on then this is also possible from inside the container. Simply go to Dockhand and enter the container. In there you can use the following [commands](#)

```

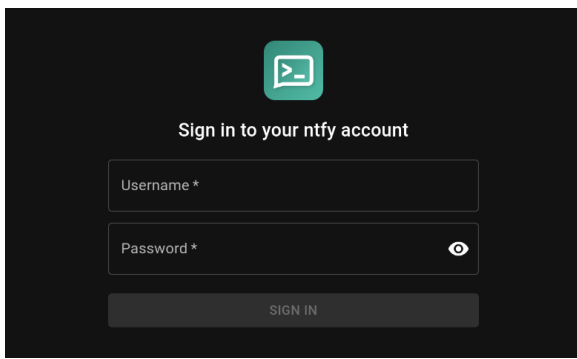
ntfy user list           # Shows list of users (alias: 'ntfy access')
ntfy user add phil      # Add regular user phil
ntfy user add --role=admin phil  # Add admin user phil
ntfy user del phil      # Delete user phil
ntfy user change-pass phil  # Change password for user phil
ntfy user change-role phil admin  # Make user phil an admin
ntfy user change-tier phil pro  # Change phil's tier to "pro"
ntfy user hash          # Generate password hash, use with auth-users config option

```

Alright so that was basically it for installing NTFY on your instance.

Configuration

Sweet, so you got NTFY installed, and you have a reverse proxy pointing to your IP-DOCKERVM:Port then you will be able to reach it on the domain name you have set in the configuration file. Then if all goes well you should be prompted with a login screen as such



Then simply login with the Admin account which brings you to the home screen. Then to create a topic is also pretty simple by just clicking on Subscribe to topic. Now, you can choose a name related to your app but that would be less secure. Hence, I would click on the **Generate Name** button. This will generate a random string of numbers that is not easily to guess by anyone.

The great thing is that after you have created the topic you can create an alias in human readable form. However, make sure that you note down the real topic e.g

<https://ntfy.example.com/qCLtfl7rW6LPZ9yp> since that will be the topic that your application will subscribe to.

Access Tokens

The final thing we need to setup are access tokens which applications will need to authenticate for your NTFY instance. For this go to Account > Access Tokens and simply click on **Create Access Token**. I would recommend one access token for each application for security reasons.

Perfect! You have now setup your own NTFY instance and can start using it in the webbrowser, desktop app and of course especially on your phone. Make sure you save your login credentials in Vaultwarden or another password manager s.t you can always login from any device.

2. Classic Email SMTP Setup

Email is definitely one of the notification systems that you need to setup for notifications. Lots of applications use email for user verification, to setup accounts, but also to get notified for events, changes etcetera.

Now, every Email provider has an SMTP (For Outgoing Mail) and IMAP (For Incoming mail) server, for instance Porkbun follows the following [Email Client configuration](#). However, one could also have setup his own email server obviously with Mailcow for instance.

Service	Hostname	Port	Security
SMTP	smtp.porkbun.com	587	STARTTLS
SMTP	smtp.porkbun.com	50587	STARTTLS Alt.
SMTP	smtp.porkbun.com	465	Implicit TLS
IMAP	imap.porkbun.com	993	SSL (SSL/TLS)
POP	pop.porkbun.com	995	SSL (SSL/TLS)

In this Porkbun example the SMTP server sits on port 587 with STARTTLS and the IMAP server on port 993 with SSL. Then the only thing left is to write down your username i.e email address and password. These configurations are the ones we will use for applications to notify us for any kind of change.