

# AI-03025 Standard Notes Docker

■ Owner	Ⓒ Rigel Arcayan
■ Tags	Project
■ Created time	@September 23, 2025 6:47 AM

**What is Standard Notes? Explain the beginning ideas, history and dev story. Provide a step by step docker deployment of Standard Notes.**

Standard Notes is an open-source, end-to-end encrypted notes app designed for security, privacy, and longevity. It is available on all major platforms, including web, desktop, and mobile, and allows users to seamlessly sync their notes. Its core feature is the robust encryption that ensures only the user can read their notes, not even the company's employees.

## The Beginning Ideas & History

The founding idea behind Standard Notes was to create a "digital safe space" for users' data, especially in a time of growing concern over privacy and government surveillance. The project was founded in 2016 in the United States. Its core principles are:

- **Longevity:** Standard Notes was designed to be a durable and reliable tool, not a service that would disappear or change its business model in a way that compromises user trust. The company prioritizes sustainability over rapid, often risky, growth.
- **Trustless Security:** The architecture is designed so that the company's servers are completely "trustless" or "zero-knowledge." This means the data stored is unreadable to the company itself. All encryption and decryption happen on the user's device, with the user possessing the only key.
- **Open Source:** To validate its security claims, Standard Notes is completely open-source. This allows anyone to audit the code and verify that the security

protocols are implemented correctly. This transparency builds trust and provides peace of mind.

Initially, the app focused on a simple, plain-text experience. The idea was that a basic, stable foundation could be built upon with various editors and features, which are often provided through paid subscriptions. This strategy allows the core, secure application to remain free and open, while additional features, such as rich text, markdown, and spreadsheets, are available for those who need them and help sustain the project.

### **Step-by-Step Docker Deployment**

Self-hosting a Standard Notes sync server with Docker gives you complete control over your data. This is an advanced process and requires a server with Docker and Docker Compose installed.

#### **1. Create a Working Directory**

First, create a new directory for your Standard Notes server files. This is where you'll store all the configuration files.

```
mkdir standardnotes-server  
cd standardnotes-server
```

#### **2. Download Configuration Files**

Download the necessary environment variable template and Docker Compose file from the official Standard Notes server repository.

```
curl https://raw.githubusercontent.com/standardnotes/server/main/.env.sample > .env  
curl https://raw.githubusercontent.com/standardnotes/server/main/docker-compose.example.yml > docker-compose.yml
```

#### **3. Configure Environment Variables**

Edit the newly created `.env` file using a text editor like nano or vim. You must set several critical variables.

nano .env

- Database: Configure the database connection details.
  - DB\_HOST=db
  - DB\_PORT=3306
  - DB\_USERNAME=your\_db\_username
  - DB\_PASSWORD=your\_secure\_password
  - DB\_DATABASE=standard\_notes\_db
- Keys: Generate secure, random strings for the authentication and encryption keys. Do not use the example values. You can use `openssl rand -hex 32` to generate them.
  - AUTH\_JWT\_SECRET=<your\_secure\_string\_here>
  - SESSION\_TOKEN\_SECRET=<your\_secure\_string\_here>

#### 4. (Optional) Customize the Docker Compose File

If you want to change the exposed ports or other container settings, you can edit the `docker-compose.yml` file. For most users, the default settings are sufficient.

#### 5. Deploy the Server

With the configuration files set, you can now pull the Docker images and start the services. The `-d` flag runs the containers in detached mode, so they run in the background.

```
docker compose pull
docker compose up -d
```

After a few moments, the server will be running. You can check the logs to ensure everything started correctly with `docker compose logs -f`.

#### 6. Connect the Client

Finally, open the Standard Notes desktop or mobile app. In the login or registration

screen, go to Advanced options and select Custom Sync Server. Enter the URL or IP address of your server (e.g., `http://your_server_ip:3000`), then register a new account or log in to an existing one to begin using your self-hosted instance.