

# Flowise Docker Setup Tutorial

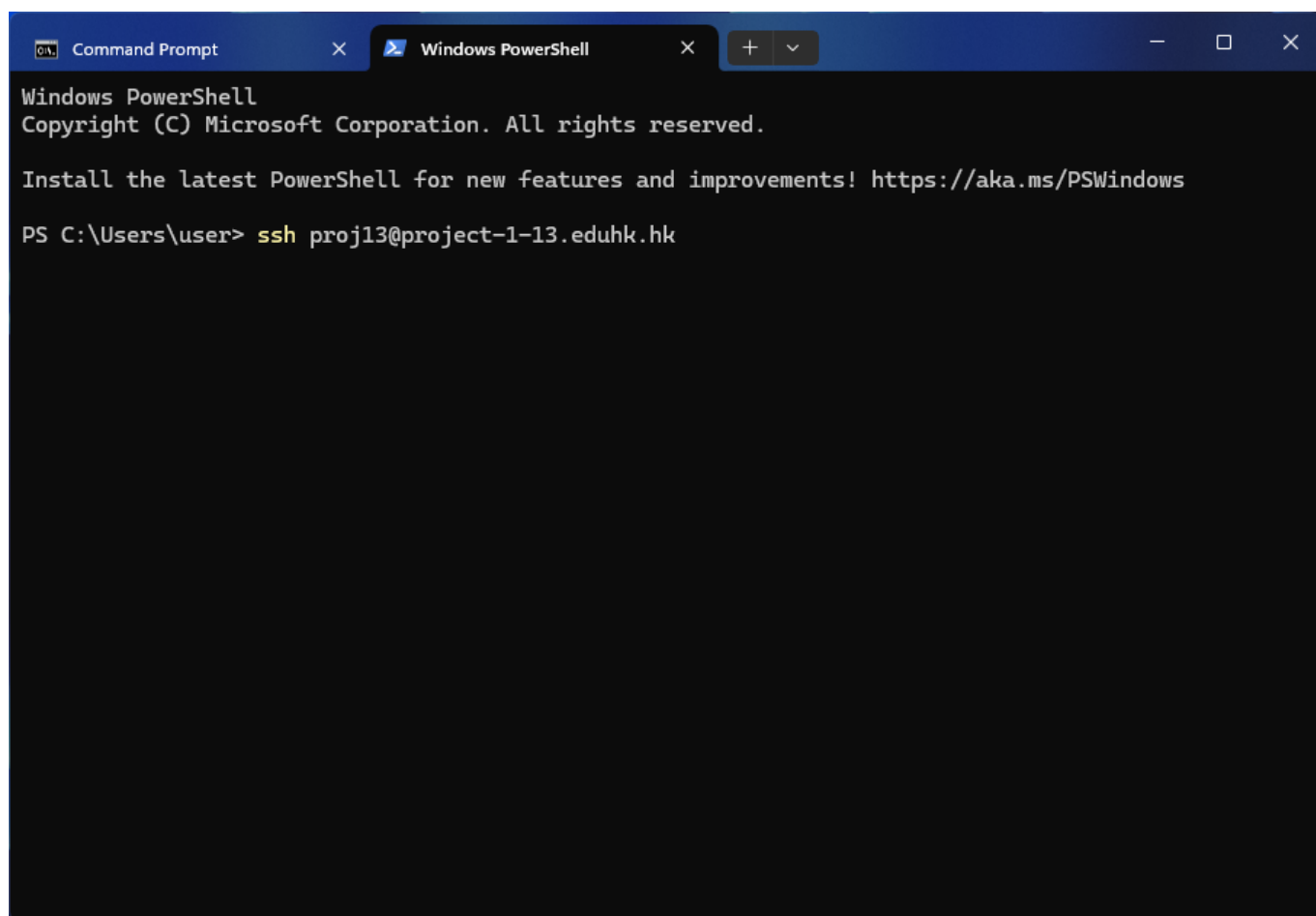
---

This tutorial guides you through setting up Flowise using Docker, and configuring it to be served via Nginx.

## 1. SSH into your server

Connect to your server using SSH.

```
ssh proj13@project-1-13.eduhk.hk
```



## 2. Change Docker's Default IP Configuration

Clone the repository to change Docker's default IP, make the script executable, and run it.

```
git clone https://github.com/enoch-sit/change-docker-default-ip.git
cd change-docker-default-ip/
chmod +x script.sh
sudo ./script.sh
cd ..
```

```
Command Prompt
proj13@project-1-13: ~/chang

*** System restart required ***
Last login: Sat Sep 20 02:22:16 2025 from 172.18.120.181
proj13@project-1-13:~$ ls
nginxsetupscriptssl
proj13@project-1-13:~$ git clone https://github.com/enoch-sit/change-docker-default-ip.git
Cloning into 'change-docker-default-ip'...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 18 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (18/18), 8.66 KiB | 4.33 MiB/s, done.
Resolving deltas: 100% (4/4), done.
proj13@project-1-13:~$ ls
change-docker-default-ip  nginxsetupscriptssl
proj13@project-1-13:~$ cd change-docker-default-ip/
proj13@project-1-13:~/change-docker-default-ip$ ls
README.md  script.sh
proj13@project-1-13:~/change-docker-default-ip$ sudo chmod +x script.sh
[sudo] password for proj13:
proj13@project-1-13:~/change-docker-default-ip$ ls
README.md  script.sh
proj13@project-1-13:~/change-docker-default-ip$ sudo ./script.sh
=====
Docker Install, IP Change, Verify, and Cleanup Script
Target IP range: 10.20.0.0/16 (pools), BIP: 10.20.1.1/24
=====
✓ Detected Snap Docker.
✓ Backup created.
Updating Docker configuration...
✓ Configuration updated.
```

### 3. Clone the Flowise Docker Setup Repository

Clone the repository containing the Flowise Docker setup.

```
git clone https://github.com/enoch-sit/flowisedockersetup202509.git
cd flowisedockersetup202509/
```

```
Command Prompt x proj13@project-1-13: ~/flowis x + v
kersetup202509.git
Cloning into 'flowisedockersetup202509'...
remote: Enumerating objects: 83, done.
remote: Counting objects: 100% (83/83), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 83 (delta 45), reused 26 (delta 8), pack-reused 0 (from 0)
Receiving objects: 100% (83/83), 32.70 KiB | 837.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
proj13@project-1-13:~/change-docker-default-ip$ ls
flowisedockersetup202509 README.md script.sh
proj13@project-1-13:~/change-docker-default-ip$ sudo rm -r flowisedockersetup202509/
[sudo] password for proj13:
proj13@project-1-13:~/change-docker-default-ip$ ls
README.md script.sh
proj13@project-1-13:~/change-docker-default-ip$ cd ../
proj13@project-1-13:~$ git clone https://github.com/enoch-sit/flowisedockersetup202509.git
Cloning into 'flowisedockersetup202509'...
remote: Enumerating objects: 83, done.
remote: Counting objects: 100% (83/83), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 83 (delta 45), reused 26 (delta 8), pack-reused 0 (from 0)
Receiving objects: 100% (83/83), 32.70 KiB | 881.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
proj13@project-1-13:~$ ls
change-docker-default-ip flowisedockersetup202509 nginxsetupscriptssl
proj13@project-1-13:~$ cd flowisedockersetup202509/
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh docker-compose.yml monitor.sh README.md
deploy.sh init-db nginx-integration.conf secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$
```

```
proj13@project-1-13: ~/flowis x + v
remote: Compressing objects: 100% (74/74), done.
remote: Total 83 (delta 45), reused 26 (delta 8), pack-reused 0 (from 0)
Receiving objects: 100% (83/83), 32.70 KiB | 837.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
proj13@project-1-13:~/change-docker-default-ip$ ls
flowisedockersetup202509 README.md script.sh
proj13@project-1-13:~/change-docker-default-ip$ sudo rm -r flowisedockersetup202509/
[sudo] password for proj13:
proj13@project-1-13:~/change-docker-default-ip$ ls
README.md script.sh
proj13@project-1-13:~/change-docker-default-ip$ cd ../
proj13@project-1-13:~$ git clone https://github.com/enoch-sit/flowisedockersetup202509.git
Cloning into 'flowisedockersetup202509'...
remote: Enumerating objects: 83, done.
remote: Counting objects: 100% (83/83), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 83 (delta 45), reused 26 (delta 8), pack-reused 0 (from 0)
Receiving objects: 100% (83/83), 32.70 KiB | 881.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
proj13@project-1-13:~$ ls
change-docker-default-ip flowisedockersetup202509 nginxsetupscriptssl
proj13@project-1-13:~$ cd flowisedockersetup202509/
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh docker-compose.yml monitor.sh README.md
deploy.sh init-db nginx-integration.conf secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$ sudo chmod +x deploy.sh
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh docker-compose.yml monitor.sh README.md
deploy.sh init-db nginx-integration.conf secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$
```

## 4. Check Port Availability and Stop Existing Services

Before deploying Flowise, ensure that port 3000 is free. If you have other Node.js applications running, for example with PM2, you should stop them.

```
# Check if any process is using port 3000
sudo lsof -i :3000

# List all processes managed by PM2
pm2 list

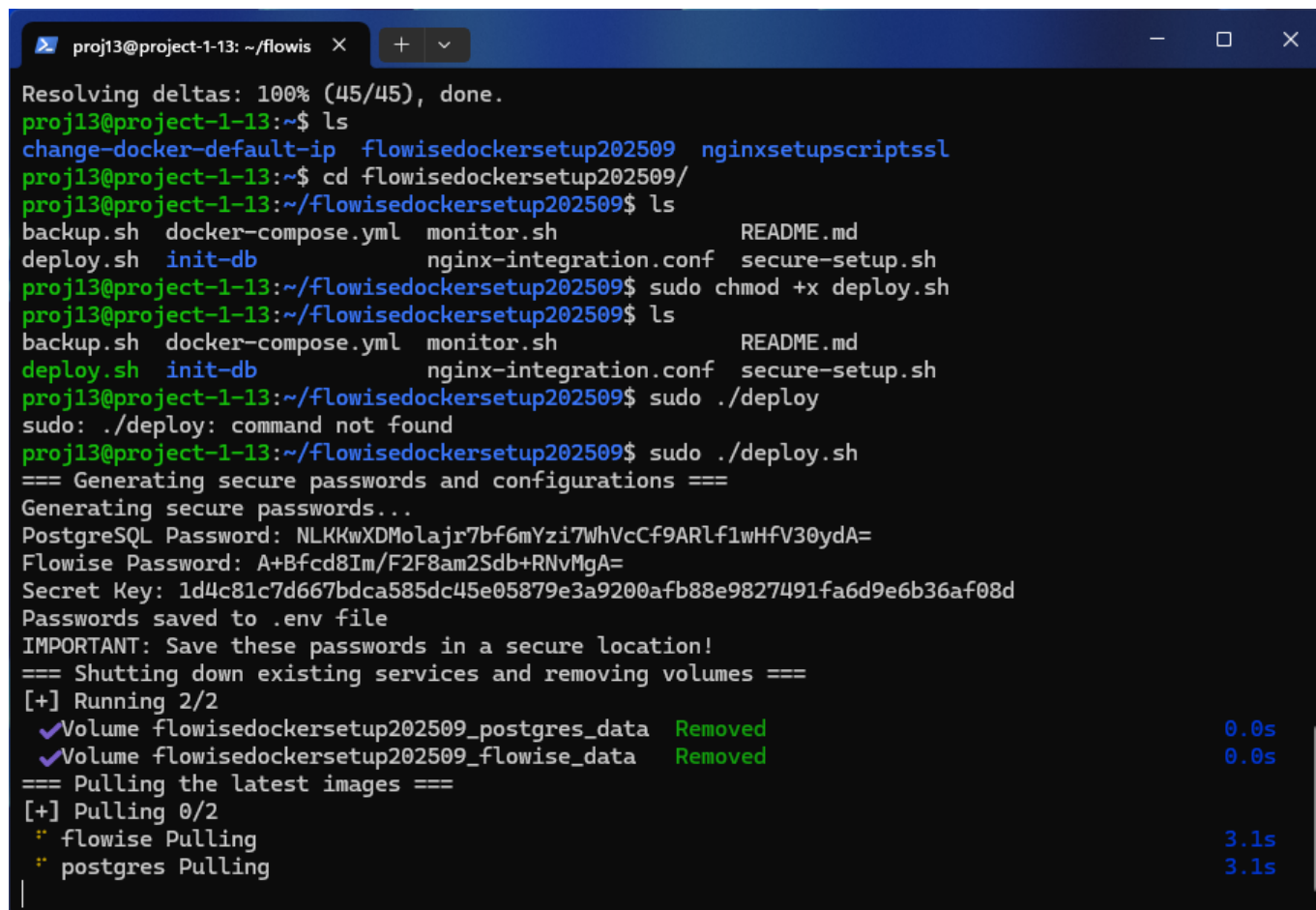
# Stop all PM2-managed processes
pm2 stop all

# Or, to delete them completely
pm2 delete all
```

## 5. Deploy Flowise

Make the deployment script executable and run it. This will generate passwords and start the Flowise and Postgres containers.

```
chmod +x deploy.sh
sudo ./deploy.sh
```

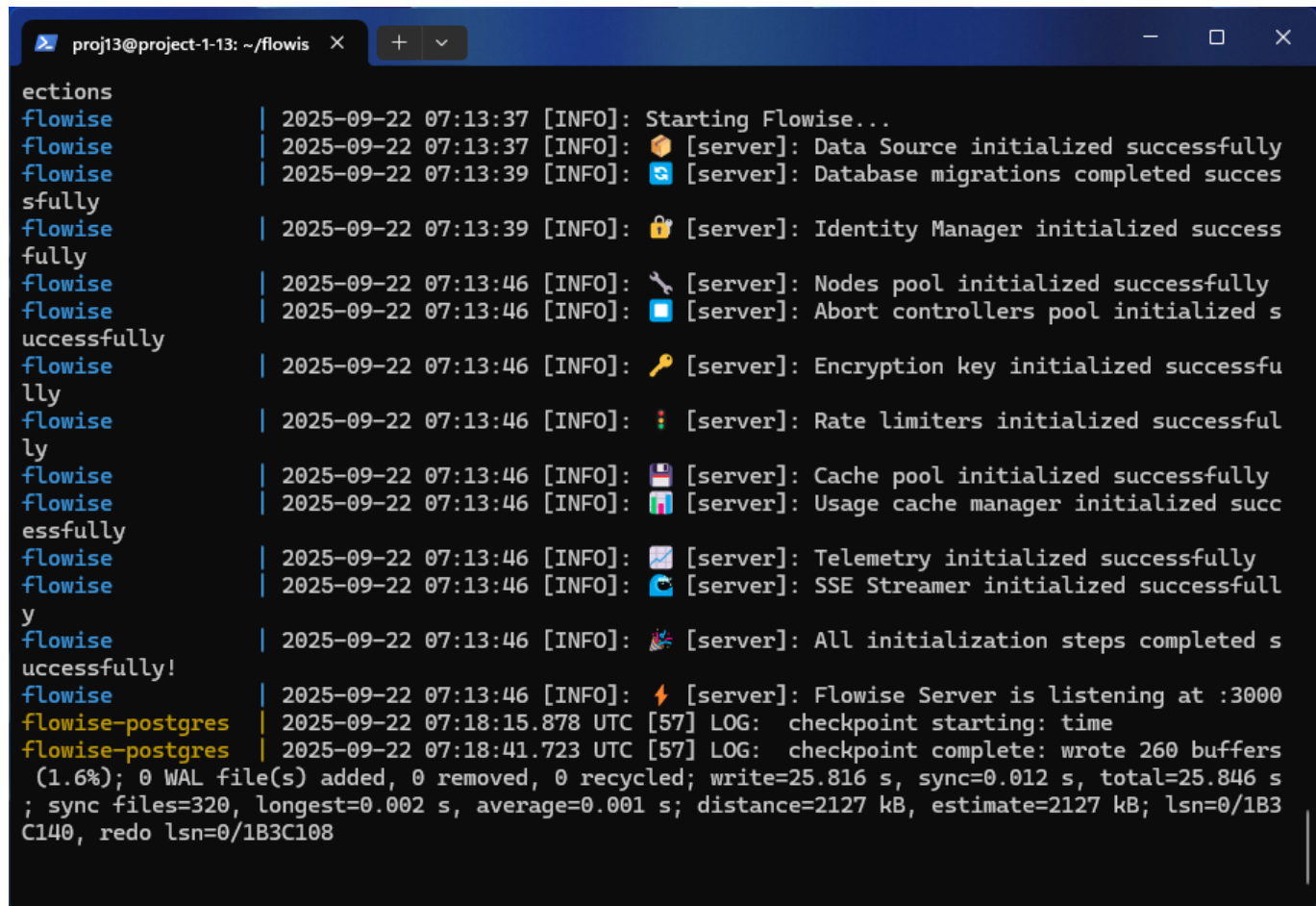


```
proj13@project-1-13: ~/flowis x
Resolving deltas: 100% (45/45), done.
proj13@project-1-13:~$ ls
change-docker-default-ip  flowisedockersetup202509  nginxsetupscriptssl
proj13@project-1-13:~$ cd flowisedockersetup202509/
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh  docker-compose.yml  monitor.sh  README.md
deploy.sh  init-db             nginx-integration.conf  secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$ sudo chmod +x deploy.sh
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh  docker-compose.yml  monitor.sh  README.md
deploy.sh  init-db             nginx-integration.conf  secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$ sudo ./deploy
sudo: ./deploy: command not found
proj13@project-1-13:~/flowisedockersetup202509$ sudo ./deploy.sh
=== Generating secure passwords and configurations ===
Generating secure passwords...
PostgreSQL Password: NLKKwXDMolajr7bf6mYzi7WhVcCf9ARlf1wHfV30ydA=
Flowise Password: A+Bfcd8Im/F2F8am2Sdb+RNvMgA=
Secret Key: 1d4c81c7d667bdca585dc45e05879e3a9200afb88e9827491fa6d9e6b36af08d
Passwords saved to .env file
IMPORTANT: Save these passwords in a secure location!
=== Shutting down existing services and removing volumes ===
[+] Running 2/2
  ✓ Volume flowisedockersetup202509_postgres_data  Removed  0.0s
  ✓ Volume flowisedockersetup202509_flowise_data   Removed  0.0s
=== Pulling the latest images ===
[+] Pulling 0/2
  • flowise Pulling 3.1s
  • postgres Pulling 3.1s
```

## 6. Verify Deployment

Check the container logs to ensure that the deployment was successful and Flowise is running.

```
docker-compose logs -f
```

A terminal window titled 'proj13@project-1-13: ~/flowis' displays the output of 'docker-compose logs -f'. The logs show the Flowise server starting and initializing various components. The 'flowise' container logs include: 'Starting Flowise...', 'Data Source initialized successfully', 'Database migrations completed successfully', 'Identity Manager initialized successfully', 'Nodes pool initialized successfully', 'Abort controllers pool initialized successfully', 'Encryption key initialized successfully', 'Rate limiters initialized successfully', 'Cache pool initialized successfully', 'Usage cache manager initialized successfully', 'Telemetry initialized successfully', 'SSE Streamer initialized successfully', and 'All initialization steps completed successfully'. The 'flowise-postgres' container logs show: 'checkpoint starting: time' and 'checkpoint complete: wrote 260 buffers (1.6%); 0 WAL file(s) added, 0 removed, 0 recycled; write=25.816 s, sync=0.012 s, total=25.846 s; sync files=320, longest=0.002 s, average=0.001 s; distance=2127 kB, estimate=2127 kB; lsn=0/1B3C140, redo lsn=0/1B3C108'.

```
actions
flowise      | 2025-09-22 07:13:37 [INFO]: Starting Flowise...
flowise      | 2025-09-22 07:13:37 [INFO]: 🍷 [server]: Data Source initialized successfully
flowise      | 2025-09-22 07:13:39 [INFO]: 🗄️ [server]: Database migrations completed successfully
flowise      | 2025-09-22 07:13:39 [INFO]: 🔑 [server]: Identity Manager initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🗑️ [server]: Nodes pool initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🛑 [server]: Abort controllers pool initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🔑 [server]: Encryption key initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🚦 [server]: Rate limiters initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🗄️ [server]: Cache pool initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🗄️ [server]: Usage cache manager initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 📡 [server]: Telemetry initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🗣️ [server]: SSE Streamer initialized successfully
flowise      | 2025-09-22 07:13:46 [INFO]: 🎉 [server]: All initialization steps completed successfully!
flowise      | 2025-09-22 07:13:46 [INFO]: ⚡ [server]: Flowise Server is listening at :3000
flowise-postgres | 2025-09-22 07:18:15.878 UTC [57] LOG: checkpoint starting: time
flowise-postgres | 2025-09-22 07:18:41.723 UTC [57] LOG: checkpoint complete: wrote 260 buffers (1.6%); 0 WAL file(s) added, 0 removed, 0 recycled; write=25.816 s, sync=0.012 s, total=25.846 s; sync files=320, longest=0.002 s, average=0.001 s; distance=2127 kB, estimate=2127 kB; lsn=0/1B3C140, redo lsn=0/1B3C108
```

## 7. Update Nginx Configuration

To expose Flowise through a domain or subdomain, configure Nginx to act as a reverse proxy. Edit your site's Nginx configuration file:

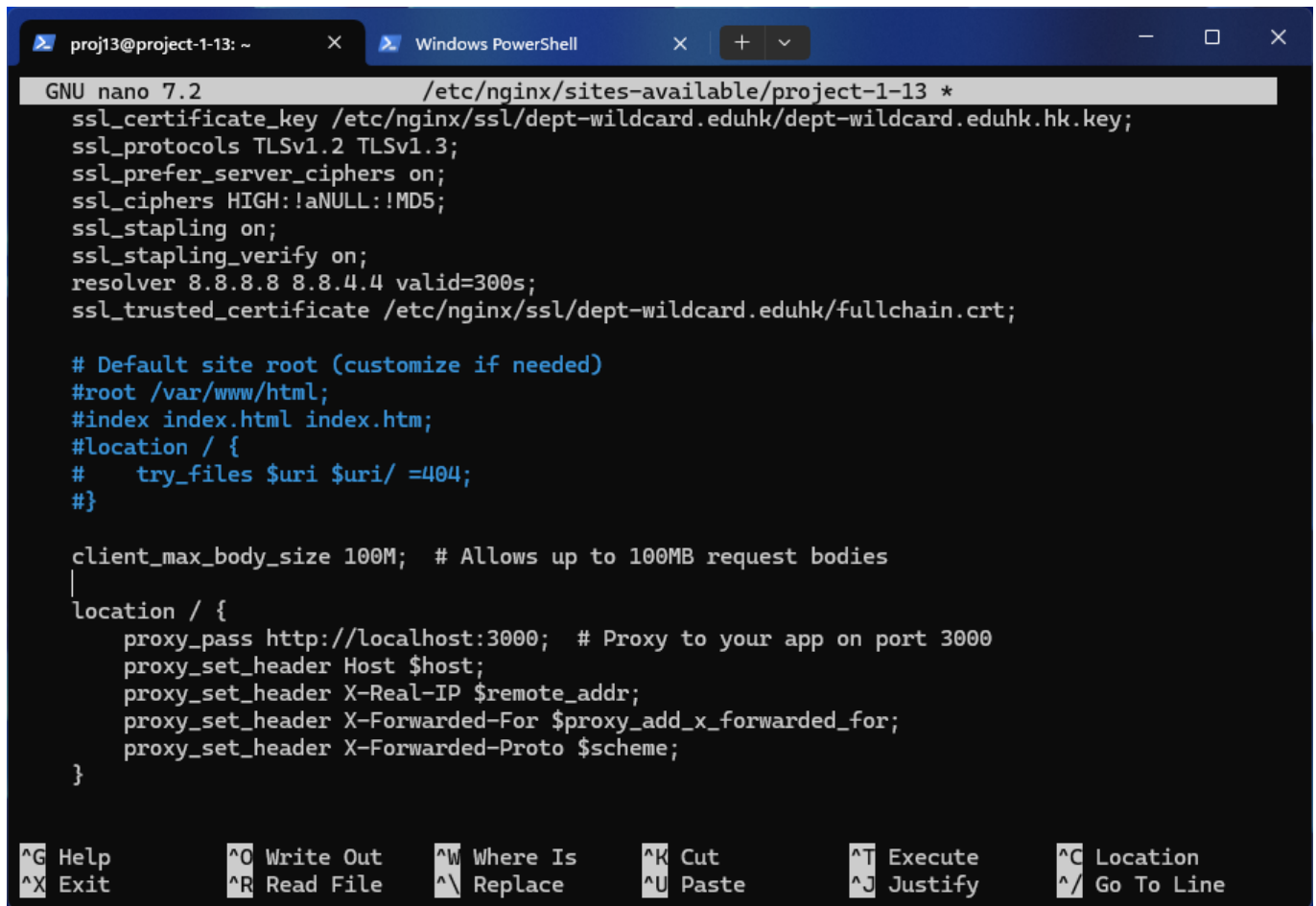
```
sudo nano /etc/nginx/sites-available/project-1-13.eduhk.hk
```

Add a **location** block to proxy requests to the Flowise container (running on port 3000 by default).

```
client_max_body_size 100M;

location / {
    proxy_pass http://localhost:3000;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
```

```
} proxy_set_header X-Forwarded-Proto $scheme;
```



The screenshot shows a Windows PowerShell terminal window with two tabs: 'proj13@project-1-13: ~' and 'Windows PowerShell'. The active tab is 'Windows PowerShell', which displays the GNU nano 7.2 editor. The editor is editing the file '/etc/nginx/sites-available/project-1-13 \*'. The configuration includes SSL settings, a default site root, and a proxy configuration for localhost:3000. The proxy configuration sets headers for Host, X-Real-IP, X-Forwarded-For, and X-Forwarded-Proto. The bottom of the terminal shows a row of keyboard shortcuts for nano editor commands.

```
GNU nano 7.2 /etc/nginx/sites-available/project-1-13 *
ssl_certificate_key /etc/nginx/ssl/dept-wildcard.eduhk/dept-wildcard.eduhk.hk.key;
ssl_protocols TLSv1.2 TLSv1.3;
ssl_prefer_server_ciphers on;
ssl_ciphers HIGH:!aNULL:!MD5;
ssl_stapling on;
ssl_stapling_verify on;
resolver 8.8.8.8 8.8.4.4 valid=300s;
ssl_trusted_certificate /etc/nginx/ssl/dept-wildcard.eduhk/fullchain.crt;

# Default site root (customize if needed)
#root /var/www/html;
#index index.html index.htm;
#location / {
#    try_files $uri $uri/ =404;
#}

client_max_body_size 100M; # Allows up to 100MB request bodies
location / {
    proxy_pass http://localhost:3000; # Proxy to your app on port 3000
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

## 8. Test and Reload Nginx

Test the Nginx configuration and then reload it to apply the changes.

```
sudo nginx -t
sudo systemctl reload nginx
```

```
proj13@project-1-13: ~  
ssl_protocols TLSv1.2 TLSv1.3;  
ssl_prefer_server_ciphers on;  
ssl_ciphers HIGH:!aNULL:!MD5;  
ssl_stapling on;  
ssl_stapling_verify on;  
resolver 8.8.8.8 8.8.4.4 valid=300s;  
ssl_trusted_certificate /etc/nginx/ssl/dept-wildcard.eduhk/fullchain.crt;  
  
# Default site root (customize if needed)  
root /var/www/html;  
index index.html index.htm;  
location / {  
    try_files $uri $uri/ =404;  
}  
}  
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$host_name  
[sudo] password for proj13:  
proj13@project-1-13:~$ echo $hostname  
  
proj13@project-1-13:~$ echo $host_name  
  
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$HOSTNAME  
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$hostname  
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$HOSTNAME  
[sudo] password for proj13:  
proj13@project-1-13:~$ sudo nginx -t  
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok  
nginx: configuration file /etc/nginx/nginx.conf test is successful  
proj13@project-1-13:~$ sudo systemctl reload nginx  
proj13@project-1-13:~$ |
```

## 9. Get Credentials

View the `.env` file to get the `FLOWISE_USERNAME` and `FLOWISE_PASSWORD`.

```
sudo cat .env
```



```
proj13@project-1-13: ~/flowise X + -
proj13@project-1-13:~$ echo $host_name
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$HOSTNAME
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$hostname
proj13@project-1-13:~$ sudo nano /etc/nginx/sites-available/$HOSTNAME
[sudo] password for proj13:
proj13@project-1-13:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
proj13@project-1-13:~$ sudo systemctl reload nginx
proj13@project-1-13:~$ cd flowisedockersetup202509/
proj13@project-1-13:~/flowisedockersetup202509$ ls
backup.sh  docker-compose.yml  monitor.sh  README.md
deploy.sh  init-db             nginx-integration.conf  secure-setup.sh
proj13@project-1-13:~/flowisedockersetup202509$ cat: .env: Permission denied
proj13@project-1-13:~/flowisedockersetup202509$ sudo cat .env
# Database Configuration
POSTGRES_DB=flowise_production
POSTGRES_USER=flowise_admin
PGPASSWORD=VcCf9ARlf1wHfV30ydA=

# Flowise Authentication
FLOWISE_USERNAME=admin
FLOWISE_PASSWORD=A+Bfcd8Im/F2F8am2Sdb+RNvMgA=

Encryption key for credentials
FLOWISE_ENCRYPTION_KEY=as85dc45e05879e3a9200afb88e9827491fa6d9e6b36af08d
proj13@project-1-13:~/flowisedockersetup202509$
```

# 10. Create Your Account

Open your browser and navigate to <https://project-1-13.eduhk.hk/organization-setup>. Use the `FLOWISE_USERNAME` and `FLOWISE_PASSWORD` from the `.env` file as the "Existing Username" and "Existing Password". Then create a new administrator account.

Flowise - Build AI Agents, Visually

project-1-13.eduhk.hk/organization-setup

Guest

Setup Account

Application authentication now requires email and password.  
Contact administrator to setup an account.

Account setup does not make any external connections, your data stays securely on your locally hosted server.

Existing Username \*

Existing Username

Existing username that was set as FLOWISE\_USERNAME environment variable

Existing Password \*

Existing Password

Existing password that was set as FLOWISE\_PASSWORD environment variable

New Account Details

Administrator Name \*

John Doe

Is used for display purposes only.

Administrator Email \*

user@company.com

Kindly use a valid email address. Will be used as login id.

# 11. Finalize Flowise UI Setup



Sign up with your new account details.

Flowise - Build AI Agents, Visually

project-1-13.eduhk.hk/organization-setup

Guest

admin

Existing username that was set as FLOWISE\_USERNAME environment variable

Existing Password \*

Existing password that was set as FLOWISE\_PASSWORD environment variable

New Account Details

Administrator Name \*

Enoch Sit

Is used for display purposes only.

Administrator Email \*

ecysit@eduhk.hk

Kindly use a valid email address. Will be used as login id.

Password \*

Password must be at least 8 characters long and contain at least one lowercase letter, one uppercase letter, one digit, and one special character.

Confirm Password \*

Reconfirm your password. Must match the password typed above.

Sign Up

## 12. Flowise is Ready

Congratulations! Your Flowise instance is now up and running, accessible through your domain.

Flowise

Star 43,837

Guest

Chatflows

Agentflows

Executions

Assistants

Marketplaces

Tools

Credentials

Variables

API Keys

Document Stores

Chatflows

Build single-agent systems, chatbots and simple LLM flows

Search Name or Category [ Ctrl + F ]

Add New

No Chatflows Yet