

# Instalação Hub Beszel

Link: <https://www.beszel.dev/guide/hub-installation>

## Hub Installation

Beszel supports installation via Docker/ Podman or single binary file.

TIP

Check the [Getting Started](#) guide if you're setting up Beszel for the first time.

## Docker or Podman

All methods will start the Beszel service on port `8090` and mount the `./beszel_data` directory for persistent storage.

docker-compose.yml  docker run  podman run

```
services:
  beszel:
    image: henrygd/beszel
    container_name: beszel
    restart: unless-stopped
    environment:
      - APP_URL=http://localhost:8090
    ports:
      - 8090:8090
    volumes:
      - ./beszel_data:/beszel_data
```

IMPORTANT

This configuration should work out of the box, but you must follow these steps when adding the system in the web UI:

1. Update the `KEY` and `TOKEN` environment variables with your public key and token, then restart the agent:

■

```
docker compose up -d
```

2. Use the unix socket path as the **Host / IP** in the web UI:

■

```
/beszel_socket/beszel.sock
```

docker-compose.yml

■

```
services:
  beszel:
    image: henrygd/beszel:latest
    container_name: beszel
    restart: unless-stopped
    environment:
      APP_URL: http://localhost:8090
    ports:
      - 8090:8090
    volumes:
      - ./beszel_data:/beszel_data
      - ./beszel_socket:/beszel_socket

  beszel-agent:
    image: henrygd/beszel-agent:latest
    container_name: beszel-agent
    restart: unless-stopped
    network_mode: host
    volumes:
      - ./beszel_agent_data:/var/lib/beszel-agent
      - ./beszel_socket:/beszel_socket
      - /var/run/docker.sock:/var/run/docker.sock:ro
    environment:
```

```
LISTEN: /beszel_socket/beszel.sock
HUB_URL: http://localhost:8090
TOKEN: <token>
KEY: "<key>"
```

## NOTE

If you prefer to set up containers in a different way, please feel free to do so.

1. Install [Docker](#) and [Docker Compose](#) if you haven't already.
2. Copy the `docker-compose.yml` content.
3. Create a directory somewhere to store the `docker-compose.yml` file.

```
mkdir beszel
cd beszel
```

4. Create a `docker-compose.yml` file, paste in the content, and save it.

nano  vim  emacs  vscode

■

```
nano docker-compose.yml
```

5. Change the `APP_URL` environment variable to the URL you'll use to access the Hub (for example, a domain name or public IP including port if needed)
6. Start the service.

```
docker compose up -d
```

# Binary

Beszel is written in pure Go and can be easily compiled (or cross-compiled) if a prebuilt binary isn't available.

## 1. Linux / FreeBSD install script

This command downloads and runs our `install-hub.sh` script. The script installs the latest binary and creates a systemd service to keep it running after reboot.

- `-u` : Uninstall
- `-p <port>` : Specify a port number (default: 8090)
- `-c <url>` : Use a custom GitHub mirror URL (e.g. <https://ghfast.top/>)
- `--auto-update` : Enable automatic daily updates
- `-h` : Show help

```
curl -sL https://get.beszel.dev/hub -o /tmp/install-hub.sh && chmod +x /tmp/install-hub.sh && /tmp/install-hub.sh
```

## 2. Manual download and start (Linux, FreeBSD, others)

### Download

Download the latest binary from [releases](#) that matches your server's CPU architecture and run it manually. You will need to create a service manually to keep it running after reboot.

```
curl -sL "https://github.com/henrygd/beszel/releases/latest/download/beszel_$(uname -s)_$(uname -m | sed -e 's/x86_64/amd64/' -e 's/armv6l/arm/' -e 's/armv7l/arm/' -e 's/aarch64/arm64/').tar.gz" | tar -xz -O beszel | tee ./beszel >/dev/null && chmod +x beszel
```

### Start the hub

```
./beszel serve --http "0.0.0.0:8090"
```

### Update the hub

```
./beszel update
```

### Create a service (optional)

If your system uses systemd, you can create a service to keep the hub running after reboot.

1. Create a service file in `/etc/systemd/system/beszel.service`, replacing `{/path/to/working/directory}` with the path to the working directory. A non-root user can be used if the user has write access to the working directory.

```
[Unit]
```

```
Description=Beszel Hub
```

```
After=network.target
```

```
[Service]
```

```
Type=simple
```

```
Restart=always
```

```
RestartSec=3
```

```
User=root
```

```
WorkingDirectory={/path/to/working/directory}
```

```
ExecStart={/path/to/working/directory}/beszel serve --http "0.0.0.0:8090"
```

```
[Install]
```

```
WantedBy=multi-user.target
```

2. Enable and start the service.

```
sudo systemctl daemon-reload
sudo systemctl enable beszel.service
sudo systemctl start beszel.service
```

## 3. Manual compile and start (any platform)

### Compile

See [Compiling](#) for information on how to compile the hub yourself.

### Start the hub

```
./beszel serve --http "0.0.0.0:8090"
```

### Update the hub

```
./beszel update
```

### Create a service (optional)

If your system uses systemd, you can create a service to keep the hub running after reboot.

1. Create a service file in `/etc/systemd/system/beszel.service`. A non-root user can be used if the user has write access to the working directory.

```
[Unit]
Description=Beszel Hub
After=network.target

[Service]
Type=simple
Restart=always
RestartSec=5
User=root
WorkingDirectory={/path/to/working/directory}
ExecStart={/path/to/working/directory}/beszel serve --http "0.0.0.0:8090"

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

2. Enable and start the service.

```
sudo systemctl daemon-reload
sudo systemctl enable beszel.service
sudo systemctl start beszel.service
```

[Edit this page on GitHub](#)

Last updated: 31/03/2026, 20:10

Pager

[Previous pageGetting Started](#)

[Next pageAgent Installation](#)

---

Revision #1

Created 2 April 2026 22:06:44 by Administrador

Updated 2 April 2026 22:09:28 by Administrador