Apps

These are the productivity apps in Infinity Tools. Note that unlike in the previous chapter, there is no specific order to these apps. You can choose what you need. The chapter numbers are only for organizational purposes.

8: Vaultwarden - Password Manager

Vaultwarden is a self-hosted password manager that lets you store and manage all your passwords securely on your own server. It's compatible with all Bitwarden apps, so you can use it with your phone, computer, and web browser. For comprehensive usage instructions and advanced features, please refer to the official Vaultwarden documentation.

What is Vaultwarden?

Vaultwarden is like a digital safe for all your passwords. Instead of remembering dozens of different passwords, you only need to remember one master password to access all your accounts.

Why Vaultwarden is useful:

- **Store passwords securely** - All passwords are encrypted and safe
- **Generate strong passwords** - Creates secure passwords for you
- **Works everywhere** - Phone, computer, web browser
- **Sync across devices** - Your passwords are available everywhere
- **You own your data** - Everything stays on your server

Think of it like this: Instead of writing passwords on sticky notes or using the same password everywhere, Vaultwarden keeps them all safe in one encrypted vault that only you can access.

Prerequisites

Before installing Vaultwarden, make sure you have:

- **Traefik installed** (from Chapter 4)
- **Docker running** (from Chapter 3)
- **A domain name** (recommended for security)
- **An email address** (for SSL certificates)

Why These Prerequisites Matter

Traefik: Provides secure HTTPS access to your password manager

Docker: Runs Vaultwarden in a secure container

Domain name: Makes it easier to access and more secure

Email: Required for SSL certificates to keep your passwords safe

Step 1: Start Infinity Tools

Make sure you're connected to your server via SSH, then start Infinity Tools:

```
sudo infinity-tools
```

Using the Infinity Tools GUI

When you start Infinity Tools, you'll see the main menu. Look for the □□ **APPLICATIONS** section - this is where all your apps are located.

- □ **Color-coded sections** - Applications have their own section
- □ **Clear descriptions** - Each app shows what it does
- □ **Status indicators** - Shows if apps are installed or running
- □ **Easy navigation** - Use arrow keys to move around

Step 2: Navigate to Vaultwarden

In the Infinity Tools menu:

1. Use your arrow keys to navigate to □□ **APPLICATIONS**
2. Press **Enter** to open the Applications menu
3. Look for **Vaultwarden** in the list
4. Select it and press **Enter**

Understanding the Application Menu

The Applications section shows you all available apps:

- □□ **Security Apps** - Vaultwarden, WireGuard, Warpgate
- ☁ **Cloud Apps** - Nextcloud, Syncthing
- □□ **Web Apps** - WordPress, Matomo
- □□ **Monitoring Apps** - Netdata, Uptime Kuma

Look for the turquoise cursor - it shows what you're about to select!

Step 3: Install Vaultwarden

When you select Vaultwarden, you'll see installation options. Choose **Install Vaultwarden**.

What Happens During Installation

Vaultwarden installation will:

- Create a secure container for your password data
- Set up SSL certificates for secure access
- Configure the web interface
- Generate an admin token for management
- Start the service

This usually takes 2-3 minutes.

Step 4: Configure Vaultwarden

During installation, you'll be asked several questions:

SSL Configuration

You'll see a prompt like:

```
□□SSL Certificate Configuration
==============================
Do you want to use Traefik for SSL certificates and domain routing?

Y) Yes (default) - Use Traefik with Let's Encrypt SSL and domain
N) No - Standalone with self-signed certificate and direct port access
```

Choose "Y" (Yes) - This uses Traefik for secure HTTPS access (recommended).

Domain Configuration

You'll be asked for your domain name:

```
Enter the domain name for Vaultwarden:
[vault.example.com]
```

What to enter: Use a subdomain like `vault.yourdomain.com` or `passwords.yourdomain.com`

Examples:

- `vault.myinfinitytools.com`
- `passwords.mydomain.com`
- `vault.home.local` (for local testing)

User Signup Policy

You'll be asked about user signups:

```
□□User Signup Policy
==================
Do you want to allow new users to sign up?

Y) Yes - Allow anyone to create an account
N) No - Only admin can create accounts (recommended)
```

For beginners: Choose "N" (No) - This keeps your password manager private and secure.

Step 5: Wait for Installation

After answering the questions, Vaultwarden will install and start. You'll see messages like:

```
□□Installing Vaultwarden...
□□Creating configuration...
□□Setting up SSL certificates...
□□Generating admin token...
□ Vaultwarden installed successfully!
```

Step 6: Get Your Admin Token

After installation, you'll see important information:

```
□□Vaultwarden Admin Information
==============================

Admin Token: abc123def456ghi789...
Web Vault: https://vault.yourdomain.com
Admin Panel: https://vault.yourdomain.com/admin
```

Save Your Admin Token

IMPORTANT: Save your admin token in a safe place! You'll need it to:

- Access the admin panel
- Manage users and settings
- Configure advanced options

How to save it:

- Copy it to a secure note on your phone
- Write it down and store it safely
- Don't share it with anyone

Step 7: Verify Vaultwarden is Working

Let's make sure Vaultwarden is running properly.

Check Status in Infinity Tools

In the Infinity Tools menu, go to:

- □□ **STATUS & HEALTH → STATUS**

You should see Vaultwarden listed as "RUNNING" or "ACTIVE".

Using the Status Dashboard

The Status & Health section shows you:

- □ **Service Status** - Which applications are running
- □ **System Health** - CPU, memory, and disk usage
- □ **Docker Info** - All your containers and their status
- □ **Network Status** - Shows your Docker networks

Look for the green checkmarks - they indicate everything is working properly!

Test Your Web Vault

Open your web browser and visit your Vaultwarden URL:

- Go to `https://vault.yourdomain.com`
- You should see the Vaultwarden login page
- Look for the lock icon in your browser (secure connection)

Step 8: Create Your First Account

Now it's time to set up your password manager!

Sign Up Process

1. Visit your Vaultwarden URL in your browser
2. Click **"Create Account"**
3. Enter your email address
4. Create a strong master password
5. Confirm your password
6. Click **"Create Account"**

Choosing a Strong Master Password

Your master password protects all your other passwords. Make it:

- **At least 12 characters long**
- **Mix of letters, numbers, and symbols**
- **Easy for you to remember**
- **Unique (don't use it anywhere else)**

Example: `MyDog@2024!Loves#Treats`

Step 9: Understanding What Was Created

Vaultwarden installation creates several important files and configurations:

Data Storage

Your password data is stored in:

- `/opt/speedbits/vaultwarden/data/` - Your encrypted password database
- `/opt/speedbits/vaultwarden/admin_token.txt` - Your admin access token

Web Access

Vaultwarden provides:

- **Web Vault:** `https://vault.yourdomain.com` - Main interface
- **Admin Panel:** `https://vault.yourdomain.com/admin` - Management interface

Security Features

Vaultwarden includes:

- **End-to-end encryption** - Your passwords are encrypted
- **HTTPS access** - Secure connection
- **Admin controls** - Manage users and settings
- **Backup ready** - Data can be backed up easily

What's Next?

Congratulations! You now have your own password manager running securely on your server.

Next Steps

- **Download Bitwarden apps** - For your phone and computer
- **Import existing passwords** - From other password managers
- **Set up two-factor authentication** - For extra security
- **Create your first password** - Start using your vault

Getting Help

For detailed usage instructions, advanced features, and troubleshooting, please refer to the official Vaultwarden documentation.

Troubleshooting

Can't Access Vaultwarden

If you can't access your Vaultwarden:

- Check that Traefik is running: `docker ps | grep traefik`
- Verify your domain points to your server
- Wait a few minutes for SSL certificates to be issued
- Check Vaultwarden logs: `docker logs vaultwarden`

SSL Certificate Issues

If you see security warnings:

- Make sure your domain is correctly configured
- Check that ports 80 and 443 are open
- Wait for certificates to be generated (can take 5-10 minutes)
- Try refreshing the page after a few minutes

Can't Create Account

If signup is disabled:

- Use the admin panel to create accounts
- Go to `https://vault.yourdomain.com/admin`
- Use your admin token to log in
- Create user accounts from the admin interface

Quick Reference

Check Vaultwarden status:

```
docker ps | grep vaultwarden
```

View Vaultwarden logs:

```
docker logs vaultwarden
```

Restart Vaultwarden:

```
docker restart vaultwarden
```

Access admin panel:

```
https://vault.yourdomain.com/admin
```

You're Ready!

Vaultwarden is now installed and running! You have your own secure password manager that you control completely.

What you accomplished:

- ✅ Installed and configured Vaultwarden
- ✅ Set up secure HTTPS access
- ✅ Created your admin account
- ✅ Secured your password data

Next step: Download the Bitwarden apps for your devices and start using your new password manager!

What You Learned

- **Vaultwarden** - A self-hosted password manager
- **Password Security** - How to store passwords safely
- **Admin Management** - How to control access to your vault
- **SSL Security** - How HTTPS protects your data

You now have a professional-grade password manager running on your own server!

Next: Installing Your Next Application (Coming Soon)

9: Passbolt - Team Password Manager

Passbolt is a team-oriented, self-hosted password manager built on OpenPGP. It lets you securely store and share passwords with your team. For comprehensive usage instructions, browser extension setup, and advanced features, please refer to the official Passbolt documentation.

What is Passbolt? (Simple Explanation)

Passbolt helps teams store and share passwords securely. It uses strong encryption and a browser extension to keep your secrets safe and easy to use.

Why Passbolt is useful:

- **Team sharing** - Share secrets with specific people
- **Strong security** - OpenPGP-based encryption
- **Browser extensions** - Easy access in Chrome/Firefox
- **Self-hosted** - You control your data

Prerequisites

Before installing Passbolt, make sure you have:

- **Traefik installed** (from Chapter 4)
- **Docker running** (from Chapter 3)
- **Borgmatic installed** (from Chapter 5) - Your data will be automatically backed up (optional but recommended)
- **Subdomain ready** (from Chapter 4.5), e.g., `pass.yourdomain.com`
- **An email address** (for SSL certificates)

Why These Prerequisites Matter

Traefik: Provides secure HTTPS access

Docker: Runs Passbolt securely in containers

Borgmatic: Automatically backs up your Passbolt data and database

Subdomain: Easy, secure access for your team

Step 1: Start Infinity Tools

Connect via SSH and start Infinity Tools:

```
sudo infinity-tools
```

Using the Infinity Tools GUI

From the main menu, go to the **APPLICATIONS** section.

- **Clear categories** - Applications are grouped logically
- **Status indicators** - Shows if services are installed
- **Easy navigation** - Arrow keys + Enter

Step 2: Install Passbolt

1. Open **APPLICATIONS**
2. Select **Passbolt**
3. Choose **Install Passbolt**

What Happens During Installation

- Creates Passbolt and database containers
- Generates secure database passwords
- Configures SSL via Traefik (recommended)
- Sets up data directories in `/opt/speedbits/passbolt`

Step 3: Configure Passbolt

SSL & Domain

You'll be asked whether to use Traefik and for your domain. Recommended:

- Use Traefik: **Yes**
- Domain: e.g., `pass.yourdomain.com`

Admin Account

After installation, you'll finish setup in the browser by creating the first admin user and installing the Passbolt browser extension.

Step 4: Open Passbolt

Once installation completes:

- Go to `https://pass.yourdomain.com`
- Follow the on-screen setup wizard
- Install the Passbolt browser extension when prompted

Step 5: Verify and Basics

- **Service running:** Check **STATUS & HEALTH → STATUS**
- **Backup active:** Borgmatic will include Passbolt data automatically
- **Login works:** Use your admin account

Troubleshooting

Can't Access the Site

- Check Traefik is running: `docker ps | grep traefik`
- Make sure your subdomain points to your server
- Wait a few minutes for SSL certificates

Database Issues

- Check database container status in **STATUS & HEALTH → DOCKER INFO**
- Review logs: `docker logs passbolt`, `docker logs passbolt-db`

Quick Reference

Web UI: `https://pass.yourdomain.com`

Data directory: `/opt/speedbits/passbolt/`

Database credentials: `/opt/speedbits/passbolt/db_password.txt`

You're Ready!

Passbolt is now installed and ready for your team. Manage users and shared passwords from the web interface and browser extension.

Next: Add your team, create groups, and start sharing passwords securely. For how-to guides and best practices, see the official Passbolt documentation.

10: Syncthing - File Synchronization

Syncthing keeps folders on your devices in sync (PCs, servers, laptops). It’s private, fast, and peer-to-peer. For detailed usage and device pairing guides, see the official Syncthing documentation.

Dependency check

- Required: **Docker** (Chapter 3)
- Optional: **Traefik** (Chapter 4) + subdomain (Chapter 4.5) for easy HTTPS access
- Optional: **Borgmatic** (Chapter 6) to back up Syncthing data

What is Syncthing? (Simple Explanation)

Syncthing lets you pick a folder (e.g., Documents) and keep it automatically synchronized between your devices. You choose which devices and folders to sync—nothing is uploaded to third-party clouds.

Interdependencies

Optional but recommended: Traefik for secure HTTPS access with your domain (easier to reach your server). **Borgmatic** will back up Syncthing data as part of your regular backups.

Prerequisites

- Docker running (Chapter 3)
- Optional: Traefik installed (Chapter 4) + subdomain (Chapter 4.5), e.g., `sync.yourdomain.com`
- Optional: Borgmatic installed (Chapter 6) for backups

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install Syncthing

1. Go to □□ **APPLICATIONS**
2. Select **Syncthing**
3. Choose **Install Syncthing**

Using the Infinity Tools GUI

- Use ↑/↓ to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll be asked how you want to access Syncthing's web interface:

- **Traefik (recommended)**
  - **What it is:** Use your domain with a trusted HTTPS certificate
  - **What you need:** Subdomain like `sync.yourdomain.com` pointing to your server
  - **What you get:** Clean URL like `https://sync.yourdomain.com`
  - **Pick this if:** You want easy, secure access from anywhere
- **Standalone HTTPS (self-signed)**
  - **What it is:** HTTPS with a self-signed certificate (browser warning appears once)
  - **What you need:** A free port (e.g., 8384)
  - **What you get:** URL like `https://SERVER_IP:8384`
  - **Pick this if:** You use it only within your home/office network
- **Standalone HTTP (not encrypted)**
  - **What it is:** Plain HTTP
  - **Pick this only if:** Quick local testing on a private network

Rule of thumb: Use **Traefik** if you have a domain. Otherwise use **Standalone HTTPS** for local networks.

Step 3: Open Syncthing

- **Traefik mode:** Visit `https://sync.yourdomain.com`
- **Standalone:** Visit `https://SERVER_IP:8384` (or the port you chose)

Step 4: First?Time Basics

1. **Change the GUI password:** In Settings → GUI, set a username/password
2. **Set the device name:** Give your server a friendly name (e.g., “Home-Server”)
3. **Create your first folder:** Click “Add Folder” → pick a folder path (e.g., `/opt/speedbits/syncthing/Documents`)

Step 5: Pair a Device

1. Install Syncthing on your computer/phone (see official downloads)
2. On your device, copy the **Device ID** (a long string)
3. On the server web UI, click “Add Remote Device” → paste the Device ID → give it a name
4. Accept the pairing request on the other device
5. Share a folder: Select your folder → “Share With Devices” → pick the device you added

Step 6: Verify It Works

- Create a test file (e.g., `Test.txt`) in the synced folder on one device
- Within seconds, the file should appear on the other device

Troubleshooting

- Can’t open the web UI? Check the URL/port and that the Syncthing container is running: `docker ps | grep syncthing`
- Browser warning? That’s normal for self-signed HTTPS—accept once to proceed
- Folder not syncing? Make sure both devices shared the same folder and have write permissions
- Firewall issues? Ensure ports are open or use Traefik mode for easier access

Quick Reference

- **Web UI (Traefik):** `https://sync.yourdomain.com`
- **Web UI (Standalone):** `https://SERVER_IP:8384`
- **Change GUI password:** Settings → GUI

You’re ready to keep your files in sync across devices—privately and securely.

11: Nextcloud - Private Cloud

Nextcloud is a self-hosted cloud platform for files, photos, calendars, and more — think of it like your own private Dropbox or Google Drive. For full usage instructions and advanced features, please refer to the official Nextcloud documentation.

What is Nextcloud? (Simple Explanation)

Nextcloud lets you store and share files, view photos, sync calendars and contacts, and access everything from mobile and desktop apps — all running on your own server, under your control.

- **Store and share files** with family, friends, or your team
- **Access anywhere** using web, desktop, and mobile apps
- **Sync calendars and contacts** across devices
- **Extend with apps** like Photos, Notes, Tasks, and more
- **You own your data** — everything stays on your server

Prerequisites

Before installing Nextcloud, make sure you have:

- **Traefik installed** (from Chapter 4) for secure HTTPS and domain routing
- **Docker running** (from Chapter 3)
- **Apprise installed** (from Chapter 5) for notifications
- **Borgmatic installed** (from Chapter 6) for automated backups (optional but recommended)
- **A domain name** (recommended) and an email address for SSL certificates

Interdependencies: Backups for Nextcloud use Borgmatic (Chapter 6). Borgmatic notifications rely on Apprise (Chapter 5). If you skip backups now, you can add them later.

Step 1: Start Infinity Tools

Connect to your server via SSH and start Infinity Tools:

```
sudo infinity-tools
```

Using the Infinity Tools GUI

In the main menu, go to the □□ **APPLICATIONS** section.

- □ **Applications** are grouped with clear descriptions
- □ **Status indicators** show if apps are installed/running
- □ Navigate with the arrow keys, press **Enter** to select

Step 2: Open Nextcloud in Applications

1. Go to □□ **APPLICATIONS**
2. Select **Nextcloud**
3. Choose **Install Nextcloud**

What the installer does

- Creates a PostgreSQL database
- Sets up the Nextcloud service
- Connects to Traefik for HTTPS (if selected)
- Prepares data folders under `/opt/speedbits/nextcloud`
- Generates secure admin and database passwords

Step 3: Choose HTTPS Mode

When asked about SSL/HTTPS:

```
Use Traefik for SSL? (Y/n)
```

- **Y (recommended):** Uses your domain with automatic SSL via Traefik
- **N:** Standalone mode using a direct port (HTTP or self-signed HTTPS)

Step 4: Enter Your Domain (Traefik Mode)

Example domains:

- `cloud.yourdomain.com`
- `files.yourdomain.com`

If you're not using a domain, the installer will ask you to pick a port for local access.

Step 5: Set Default Storage Quota

The installer offers to set a default per-user quota.

- **Recommended:** Choose per-user quota (e.g., **5 GB** to start)
- You can change quotas later in the web interface
- This helps prevent your server from running out of disk space

Step 6: Wait for Installation

First-time setup takes about **2–5 minutes**. The installer will show progress while Nextcloud initializes.

Step 7: Save Your Admin Credentials

When installation finishes, you'll see an admin **username** and **password**. Write them down and keep them safe.

You can also find them in `/opt/speedbits/nextcloud/.env` (root-only).

Step 8: Open Nextcloud

- **Traefik mode:** Visit `https://your-domain` (e.g., `https://cloud.yourdomain.com`)
- **Standalone mode:** Visit the IP and port shown by the installer (e.g., `http://SERVER_IP:PORT`)

Step 9: Verify It's Running

In Infinity Tools, go to □□ **STATUS & HEALTH → STATUS**. You should see Nextcloud and its database running.

Step 10: Recommended Next Steps

- **Set up email** in Nextcloud (for password resets and notifications)
- **Install useful apps**: Calendar, Contacts, Notes, Deck
- **Configure user quotas** (Settings → Users)
- **Enable backups** with Borgmatic (Chapter 6)
- **Install mobile apps** (iOS/Android) and desktop sync client

Troubleshooting

Can’t access the site

- Check that Traefik is running: `docker ps | grep traefik`
- Verify your domain points to your server (DNS)
- Wait a few minutes for SSL certificates to be issued
- Check Nextcloud logs: `docker logs nextcloud`

Running out of disk space

- Reduce user quotas or free storage
- Monitor usage: `df -h /opt/speedbits/nextcloud`

SSL warning in standalone HTTPS

Self-signed certificates show a browser warning. Click “Advanced → Proceed” to continue, or switch to Traefik with a real domain for trusted HTTPS.

Quick Reference

Check containers:

```
docker ps | grep -E "nextcloud|nextcloud-db"
```

View logs:

```
docker logs nextcloud
```

Restart service:

```
cd /opt/speedbits/nextcloud && docker compose restart
```

Helpful Resources

- Official Nextcloud documentation
- Nextcloud desktop & mobile clients

12: WordPress - Build Your Website

WordPress is the most popular platform for building websites and blogs. With Infinity Tools, you can install WordPress securely on your own server with just a few steps. For everything beyond installation and basic usage, see the official WordPress documentation.

What is WordPress? (Simple Explanation)

WordPress lets you create a website or blog using themes and plugins — no coding required. You manage posts, pages, and media from a friendly dashboard, and extend features with plugins (contact forms, SEO, e-commerce, and more).

- **Easy to use** dashboard for content
- **Thousands of themes** to change the look
- **Plugins** for features like forms, SEO, shops
- **You own it** — runs on your server

Prerequisites

Before installing WordPress, make sure you have:

- **Traefik installed** (from Chapter 4) for HTTPS and domains
- **Docker running** (from Chapter 3)
- **Apprise installed** (from Chapter 5) for notifications
- **Borgmatic installed** (from Chapter 6) for automated backups (optional but recommended)
- **A domain name** (recommended) and email for SSL certificates

Interdependencies: WordPress uses a database (MariaDB). Database backups integrate with Borgmatic (Chapter 6), and Borgmatic notifications rely on Apprise (Chapter 5).

Step 1: Start Infinity Tools

Connect to your server via SSH and start Infinity Tools:

```
sudo infinity-tools
```

Open Applications

1. Go to □□ **APPLICATIONS**
2. Select **WordPress**
3. Choose **Install WordPress**

Step 2: Choose HTTPS Mode

When prompted:

```
Use Traefik for SSL? (Y/n)
```

- **Y (recommended):** Uses your domain and automatic SSL via Traefik
- **N:** Standalone mode using a direct port (HTTP or self-signed HTTPS)

Step 3: Enter Your Domain (Traefik Mode)

Examples:

- `myblog.yourdomain.com`
- `www.yourdomain.com` (will redirect to the main domain)

No domain? Pick a port when asked (for local access only).

Step 4: Optional Redis Cache

You can enable a performance cache called **Redis**. If you enable it during installation, Infinity Tools sets up a Redis container for you.

Recommended free plugin

Install the free plugin **Redis Object Cache** to speed up WordPress:

- From your dashboard: **Plugins → Add New**
- Search for **Redis Object Cache** by Till Krüss
- Click **Install**, then **Activate**
- Go to **Settings → Redis** and click **Enable Object Cache**

Redis Object Cache plugin (wordpress.org)

Step 5: Wait for Installation

Setup usually takes a few minutes. WordPress, the database, and (optionally) Redis will be created.

Step 6: Open Your Site

- **Traefik mode:** Visit `https://your-domain`
- **Standalone mode:** Visit the IP and port shown by the installer (e.g., `http://SERVER_IP:PORT`)

Complete the WordPress setup wizard and create your admin account.

Step 7: Verify It's Running

In Infinity Tools, go to □□ **STATUS & HEALTH → STATUS**. You should see WordPress, the database, and (if used) Redis running.

Recommended Next Steps

- □ **Choose a theme** that fits your site
- □ **Install key plugins** (SEO, forms, security)
- □ **Enable Redis Object Cache** if you set up Redis
- □ **Backups**: confirm Borgmatic includes WordPress data

Where Your Data Lives

- `/opt/speedbits/wordpress/wp_data/` — WordPress files
- `/opt/speedbits/wordpress/db_data/` — Database files (MariaDB)
- `/opt/speedbits/wordpress/redis_data/` — Redis data (if enabled)

Troubleshooting

Can’t access the site

- Check Traefik is running: `docker ps | grep traefik`
- Verify your domain points to your server (DNS)
- Wait a few minutes for SSL certificates
- View logs: `docker logs wordpress`

Database connection error

- Check database container: `docker ps | grep wp-db`
- Confirm password file: `/opt/speedbits/wordpress/db_password.txt`
- Restart: `cd /opt/speedbits/wordpress && docker compose restart`

Enable HTTPS in standalone mode

Standalone HTTPS uses a self-signed certificate and may show a browser warning. Click “Advanced → Proceed”, or switch to Traefik for trusted HTTPS.

Quick Reference

Check containers:

```
docker ps | grep -E "wordpress|wp-db|redis"
```

View logs:

```
docker logs wordpress
```

Restart services:

```
cd /opt/speedbits/wordpress && docker compose restart
```

Helpful Resources

- Official WordPress documentation

- Redis Object Cache plugin

13: Matomo - Privacy?Friendly Analytics

Matomo is a self-hosted web analytics platform (an alternative to Google Analytics) that lets you track website visits while keeping full control of your data. For detailed usage and advanced features, please refer to the official Matomo documentation.

What is Matomo? (Simple Explanation)

Matomo shows you how people use your website: how many visitors you have, what pages they view, where they come from, and more — all without sending data to third parties.

- **Privacy-friendly** and GDPR-compliant
- **Real-time stats** and easy dashboards
- **Full control** — runs on your own server
- **Unlimited sites** can be tracked

Prerequisites

Before installing Matomo, make sure you have:

- **Traefik installed** (from Chapter 4) for HTTPS and domains
- **Docker running** (from Chapter 3)
- **Apprise installed** (from Chapter 5) for notifications
- **Borgmatic installed** (from Chapter 6) for automated backups
- **A domain name** (recommended) and email for SSL certificates

Interdependencies: Matomo uses a MariaDB database. Database backups are handled by Borgmatic (Chapter 6). Borgmatic notifications rely on Apprise (Chapter 5).

Step 1: Start Infinity Tools

Connect to your server via SSH and start Infinity Tools:

```
sudo infinity-tools
```

Open Applications

1. Go to ◫ **APPLICATIONS**
2. Select **Matomo**
3. Choose **Install Matomo**

Step 2: Choose HTTPS Mode

When prompted:

```
Use Traefik for SSL? (Y/n)
```

- **Y (recommended):** Uses your domain and automatic SSL via Traefik
- **N:** Standalone mode using a direct port (HTTP or self-signed HTTPS)

Step 3: Enter Your Domain (Traefik Mode)

Examples:

- `analytics.yourdomain.com`
- `stats.yourdomain.com`

No domain? The installer will ask you to pick a port for local access.

Step 4: Wait for Installation

First-time setup takes a few minutes. Matomo and its database will be created and started.

Step 5: Open Matomo and Complete the Wizard

- **Traefik mode:** Visit `https://your-domain`
- **Standalone mode:** Visit the IP and port shown by the installer (e.g., `http://SERVER_IP:PORT` or `https://SERVER_IP:PORT`)

Follow the Matomo setup wizard:

1. System check → Next
2. Database setup → The installer shows your database credentials
3. Create your admin account
4. Add your first website to track
5. Copy the tracking code (you'll paste it into your website later)

Step 6: Verify It's Running

In Infinity Tools, go to □□ **STATUS & HEALTH → STATUS**. You should see Matomo and its database running.

Recommended Next Steps

- □ **Add the tracking code** to your website (before `</head>`)
- □ **Set up the Matomo cron** for reports and archiving (see below)
- □ **Confirm backups** include Matomo's database
- □ **Review privacy settings** (IP anonymization, Do Not Track)

Cron for Archiving (Recommended)

Add this to your server's crontab to keep reports up-to-date:

```
*/5 * * * * docker exec matomo /usr/local/bin/php /var/www/html/console core:archive
>/dev/null 2>&1
```

Troubleshooting

Can't access the site

- Check Traefik is running: `docker ps | grep traefik`
- Verify your domain points to your server (DNS)
- Wait a few minutes for SSL certificates
- View logs: `docker logs matomo`

Database connection error

- Check database container: `docker ps | grep matomo-db`
- Use the database credentials shown during installation
- Restart: `cd /opt/speedbits/matomo && docker compose restart`

Quick Reference

Check containers:

```
docker ps | grep -E "matomo|matomo-db"
```

View logs:

```
docker logs matomo
```

Restart services:

```
cd /opt/speedbits/matomo && docker compose restart
```

Helpful Resources

- Official Matomo documentation
- Matomo user guides

14: Webmin - Visual Server Management

Webmin is a web-based interface for managing your Linux server. Instead of using the command line, you can manage users, services, files, system settings, and more through a friendly web browser interface. Think of it as a control panel for your entire server.

For advanced features, module documentation, and detailed guides, see the official Webmin documentation.

Why Webmin?

- **Visual server management** - Manage your server without command line
- **User management** - Create, edit, and manage user accounts visually
- **File browser** - Browse and edit files with a web interface
- **Service management** - Start, stop, and configure system services
- **System monitoring** - View system resources and logs
- **Package management** - Install and update software packages
- **Network configuration** - Configure network settings

Prerequisites

- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `webmin.yourdomain.com`

Note: Webmin is typically accessed via SSH tunnel for security. The installation script will guide you through this. Traefik mode is optional and allows direct web access.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install Webmin

1. Go to □□ **APPLICATIONS**
2. Select **Webmin**
3. Choose **Install Webmin**

Using the Infinity Tools GUI

- Use ↑/↓ to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (optional)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `webmin.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Direct web access at `https://webmin.yourdomain.com`
  - **Pick this if:** You want direct web access without SSH tunneling
- **Standalone (recommended)**
  - **What it is:** Uses HTTPS with a self-signed certificate, accessed via SSH tunnel
  - **What you need:** SSH access to your server
  - **What you get:** Secure access via SSH tunnel (more secure than direct web access)
  - **Pick this if:** You want the most secure setup (recommended)

Simple rule of thumb: Use **Standalone** for security (SSH tunnel), or **Traefik** if you want direct web access.

Step 2.2: If You Choose Traefik

1. Enter your subdomain, e.g., `webmin.yourdomain.com`
2. Ensure the subdomain's DNS A record points to your server (see Chapter 4.5)
3. Infinity Tools will configure HTTPS automatically via Let's Encrypt

After install: Your Webmin will be available at `https://webmin.yourdomain.com`

Step 2.3: If You Choose Standalone

1. Pick a port (default: 8443)
2. You'll access Webmin via SSH tunnel (instructions shown after installation)

Step 2.4: Host Filesystem Access

You'll be asked about host filesystem access. This controls whether Webmin can browse files on your actual server (not just inside the container).

- **No (default)**
  - **What it is:** Webmin can only browse files inside its own container
  - **Pick this if:** You want maximum security and only need Webmin for system management (not file browsing)
- **Yes (Read-Only)**
  - **What it is:** Webmin can browse host files at `/host/` but cannot edit them
  - **Pick this if:** You want to view server files safely without risk of accidental changes
- **Yes (Read-Write)**
  - **What it is:** Webmin can browse AND edit host files at `/host/`
  - **Pick this if:** You want full file management capabilities
  - ⚠ **Warning:** This gives Webmin full access to your server's filesystem - use with caution!

What Happens During Installation

- Webmin container is created
- System user `webminadmin` is created with a random password
- Data directory is set up at `/opt/speedbits/webmin`
- Optional domain + Traefik HTTPS routing (if using Traefik)
- Service starts and becomes accessible

Step 3: Access Webmin

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://webmin.yourdomain.com` in your browser
3. Login with the credentials shown after installation

If Using Standalone (SSH Tunnel)

⚠ **IMPORTANT:** Webmin requires an SSH tunnel for secure access. You cannot access it directly from the internet.

On your local computer (not the server), run:

```
ssh -L 8443:localhost:10000 your-username@your-server-ip
```

Replace:

- `your-username` - Your SSH username
- `your-server-ip` - Your server's IP address
- `8443` - The port you chose during installation (or default 8443)

Then in your browser, open:

```
https://localhost:8443
```

You'll see a security warning (normal for self-signed certificates). Click "Advanced" → "Proceed" to continue.

Step 4: Login to Webmin

After installation, you'll see credentials like:

- **Username:** `webminadmin`
- **Password:** A randomly generated password (shown only once!)

⚠ **CRITICAL:** Write down or save this password immediately! It will NOT be shown again. Use a password manager (like Vaultwarden from Chapter 7) to store it securely.

Login Steps

1. Enter username: `webminadmin`
2. Enter the password shown after installation
3. Click "Login"

Note: Any user with sudo privileges can also login to Webmin using their system username and password.

Step 5: Understanding the File Manager

 IMPORTANT: When you first open Webmin's File Manager, you're browsing files **inside the Webmin container**, not your actual server!

How to Access Host System Files

To browse files on your actual server (the host system), you need to navigate to the `/host/` folder:

1. Go to **Other** → **File Manager** in Webmin
2. You'll see the container's filesystem (usually empty or minimal)
3. **To access host files:** Type `/host/` in the path bar at the top
4. Press Enter or click "Go"
5. Now you'll see your actual server's filesystem!

Understanding the Path Structure

- **Container files:** `/` (root of container)
- **Host files:** `/host/` (mounted host filesystem)

Examples

- **Host home directories:** Navigate to `/host/home/`
- **Host system logs:** Navigate to `/host/var/log/`
- **Host Docker data:** Navigate to `/host/opt/`
- **Host Infinity Tools:** Navigate to `/host/opt/InfinityTools/`
- **Host Speedbits data:** Navigate to `/host/opt/speedbits/`

□□ **Tip:** Bookmark `/host/` or add it to your favorites in Webmin for quick access!

File Access Modes

Depending on what you chose during installation:

- **Read-Only:** You can view files at `/host/` but cannot edit or delete them
- **Read-Write:** You can view, edit, create, and delete files at `/host/`
- **No Access:** The `/host/` folder won't exist - you can only browse container files

What You Can Do in Webmin

System Management

- □□ **User Management** - Create, edit, delete user accounts
- □□ **Password Management** - Change user passwords
- ⚙ **Service Management** - Start, stop, restart system services
- □□ **Package Management** - Install and update software

File Management

- □□ **File Browser** - Browse and edit files (remember to use `/host/` for host files!)
- □□ **Text Editor** - Edit configuration files
- □□ **File Upload/Download** - Transfer files to/from server

Monitoring

- □□ **System Information** - View CPU, memory, disk usage
- □□ **Log Viewer** - View system logs
- □□ **Network Configuration** - Configure network settings

Security Recommendations

- □ **Use SSH tunnel** - Standalone mode with SSH tunnel is more secure than direct web access
- □ **Strong password** - The generated password is strong, but you can change it in Webmin
- □ **Store credentials securely** - Use a password manager (Vaultwarden recommended!)
- □ **Limit filesystem access** - Use "Read-Only" unless you need to edit files
- □ **Regular updates** - Keep Webmin updated for security patches
- ⚠ **Protect access** - Webmin has powerful system management capabilities - keep it secure!

Troubleshooting

Can't Access Webmin

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Standalone mode:** Make sure you're using SSH tunnel (not direct access)
- **Check container status:** Run `docker ps | grep webmin` to see if it's running
- **Check logs:** Run `docker logs webmin` to see error messages

Can't Login

- Make sure you're using the correct username: `webminadmin`
- Check that you saved the password correctly (it's shown only once!)
- Try logging in with a system user that has sudo privileges
- Check if user exists: `docker exec webmin grep webminadmin /etc/passwd`

Can't See Host Files

- **Remember:** You must navigate to `/host/` in the File Manager path bar
- If `/host/` doesn't exist, you chose "No" for host filesystem access during installation
- To enable it, reinstall Webmin and choose "Yes (Read-Only)" or "Yes (Read-Write)"
- Check installation: Look for `/host` in the docker-compose.yml file

SSH Tunnel Not Working

- Make sure SSH is working: `ssh your-username@your-server-ip`
- Check the port number matches what you chose during installation
- Try using the container IP directly: `ssh -L 8443:CONTAINER_IP:10000 user@server`
- Check Webmin is running: `docker ps | grep webmin`

Where to Find Webmin After Install

- On the finish screen, Infinity Tools prints the access URL and SSH tunnel command
- You can also see it in □□ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/webmin`
- View credentials: The password is shown only once during installation - save it!

You're Ready!

Webmin is now installed and ready to use. You can manage your Linux server visually through the web interface. Remember:

- □ Use `/host/` in File Manager to access your actual server files
- □□ Save your login credentials securely
- □□ Use SSH tunnel for the most secure access

Next steps: Explore Webmin's features, manage your server users, browse files, and configure system settings. Webmin makes server management much easier than using the command line!

15: BookStack - Documentation Platform / Wiki

BookStack is a beautiful, simple documentation and wiki platform. It helps you organize information into Books, Chapters, and Pages - just like a real book! You can write documentation, create knowledge bases, and share information with your team or the world.

For advanced features, API documentation, and customization options, see the official BookStack documentation.

Why BookStack?

- **Easy to use** - Write like you would in Word or Google Docs
- **Organized structure** - Books → Chapters → Pages
- **Rich editor** - WYSIWYG editor with Markdown support
- **Full-text search** - Find anything quickly across all your content
- **Image uploads** - Add pictures, diagrams, and attachments
- **User permissions** - Control who can view and edit what
- **Export options** - Download as PDF, HTML, or Markdown
- **Beautiful design** - Clean, modern interface that's easy on the eyes
- **Fun fact**: the platform you are seeing right now is BookStack – it hosts our docs.

Prerequisites

- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `docs.yourdomain.com`

Note: BookStack works best with Traefik and a domain name. It's designed for sharing documentation, so having a friendly URL like `docs.yourdomain.com` makes it much easier to access.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install BookStack

1. Go to □□ **APPLICATIONS**
2. Select **BookStack**
3. Choose **Install BookStack**

Using the Infinity Tools GUI

- Use **↑/↓** to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (recommended)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `docs.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://docs.yourdomain.com` with trusted SSL
  - **Pick this if:** You want to share documentation with others (recommended)
- **Standalone HTTPS (self-signed)**
  - **What it is:** Uses HTTPS with a self-signed certificate (your browser will warn it's not trusted)
  - **What you need:** Just a free port (default: 8092)
  - **What you get:** URL like `https://SERVER_IP:8092` with a warning you must accept
  - **Pick this if:** You're just testing or only using it on your local network

Simple rule of thumb: Use **Traefik** if you have a domain and want to share your documentation. Use **Standalone HTTPS** only for testing or private use.

Step 2.2: If You Choose Traefik

1. Enter your subdomain, e.g., `docs.yourdomain.com`
2. Ensure the subdomain's DNS A record points to your server (see Chapter 4.5)
3. Enter your email address (for SSL certificate notifications)
4. Infinity Tools will configure HTTPS automatically via Let's Encrypt

After install: Your BookStack will be available at `https://docs.yourdomain.com`

Step 2.3: If You Choose Standalone

1. Pick a port (default: 8092)
2. You'll access BookStack via `https://SERVER_IP:8092`
3. Accept the browser security warning (it's safe for private use)

What Happens During Installation

- BookStack container is created
- MariaDB database container is created
- Database passwords are generated and saved securely
- Data directory is set up at `/opt/speedbits/bookstack`
- Optional domain + Traefik HTTPS routing (if using Traefik)
- Services start and become accessible
- Initial setup takes 2-3 minutes (database initialization)

Step 3: Access BookStack

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://docs.yourdomain.com` in your browser
3. You'll see the BookStack welcome page

If Using Standalone

1. Open `https://SERVER_IP:8092` in your browser
2. You'll see a security warning (normal for self-signed certificates)
3. Click "Advanced" → "Proceed to site" to continue
4. You'll see the BookStack welcome page

Step 4: First Login

⚠ **CRITICAL SECURITY STEP:** BookStack comes with default admin credentials that MUST be changed immediately!

Default Credentials (First Time Only)

- **Email:** `admin@admin.com`

- **Password:** `password`

⚠ **CHANGE THESE IMMEDIATELY!** These are public defaults - anyone can guess them!

Login Steps

1. Click "Login" in the top right corner
2. Enter email: `admin@admin.com`
3. Enter password: `password`
4. Click "Log In"

Change Your Password Immediately

1. After logging in, click your name in the top right corner
2. Select "My Profile"
3. Click "Change Password"
4. Enter your current password (`password`)
5. Enter a strong new password (use a password manager!)
6. Confirm the new password
7. Click "Save"

□□ **Tip:** Use a password manager (like Vaultwarden from Chapter 7) to generate and store a strong password!

Step 5: Create Your First Book

Now that you're logged in, let's create your first documentation book!

Creating a Book

1. Click "Create Book" (usually a big button on the home page)
2. Enter a book name, e.g., "My Server Documentation"
3. Add a description (optional but helpful)
4. Click "Save Book"

Adding Chapters

1. Inside your book, click "Add Chapter"
2. Enter a chapter name, e.g., "Getting Started"
3. Add a description (optional)
4. Click "Save Chapter"

Creating Pages

1. Inside a chapter, click "Add Page"
2. Enter a page title
3. Start writing! Use the editor toolbar to format text, add images, create lists, etc.
4. Click "Save Page" when done

Using the Editor

The BookStack editor is like Word or Google Docs:

- **Bold** and *Italic* buttons for formatting
- Headings dropdown (H1, H2, H3, etc.)
- Image upload button (□□ icon)
- Link button to add hyperlinks
- Code blocks for technical content
- Markdown support (you can type Markdown if you prefer)

What You Can Do in BookStack

Content Organization

- □□ **Books** - Top-level containers (e.g., "Server Setup Guide")
- □□ **Chapters** - Sections within books (e.g., "Installation", "Configuration")
- □□ **Pages** - Individual documentation pages
- □□ **Reordering** - Drag and drop to reorganize content

Content Features

- ✍ **Rich text editor** - WYSIWYG editing with Markdown support
- □□ **Image uploads** - Add screenshots, diagrams, photos
- □□ **Attachments** - Upload files (PDFs, documents, etc.)
- □□ **Full-text search** - Search across all books and pages
- □□ **Tags** - Organize content with tags

Sharing & Export

- □□ **User roles** - Control who can view/edit (Admin, Editor, Viewer)
- □□ **Public links** - Share specific pages publicly
- □□ **Export** - Download as PDF, HTML, Markdown, or Plain Text

- □□ **Print** - Print-friendly view

Security Recommendations

- □ **Change default password immediately** - This is critical!
- □ **Use Traefik mode** - Provides trusted SSL certificates
- □ **Strong passwords** - Use a password manager to generate strong passwords
- □ **User permissions** - Set appropriate roles (don't make everyone an admin!)
- □ **Regular backups** - BookStack data is stored in `/opt/speedbits/bookstack`
- □ **Keep updated** - Re-run the install script to get updates
- ⚠ **Public content** - Be careful what you make public if using public links

Troubleshooting

Can't Access BookStack

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Check containers:** Run `docker ps | grep bookstack` to see if containers are running
- **Check logs:** Run `docker logs bookstack` to see error messages
- **Database issues:** Check database container: `docker logs bookstack-db`

Can't Login

- Make sure you're using the exact default credentials: `admin@admin.com` / `password`
- Check that BookStack finished initializing (wait 2-3 minutes after installation)
- Check container logs: `docker logs bookstack`

Slow Loading

- First-time setup takes 2-3 minutes (database initialization)
- Large images can slow down pages - optimize images before uploading
- Check server resources: `docker stats bookstack`

Lost Password

- If you forgot your password, you can reset it via the database
- Or reinstall with `--deleteall` flag (⚠ this deletes all content!)
- Better: Keep your password in a password manager!

Where to Find BookStack After Install

- On the finish screen, Infinity Tools prints the access URL
- You can also see it in □□ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/bookstack`
- Database password saved in: `/opt/speedbits/bookstack/db_password.txt`

Backing Up Your Documentation

Your BookStack content is stored in:

- **Database:** `/opt/speedbits/bookstack/db_data/`
- **Config:** `/opt/speedbits/bookstack/config/`
- **Uploads:** Inside the config directory

To backup:

```
cd /opt/speedbits
tar czf bookstack-backup.tar.gz bookstack/
```

To restore: Extract the backup and restart BookStack containers.

Email Configuration (Optional)

After installation, you'll be asked if you want to configure email (SMTP). This is optional but useful for:

- Password reset emails
- Notification emails when pages are updated
- User invitation emails

You can skip this and configure it later from the Infinity Tools menu or web interface.

You're Ready!

BookStack is now installed and ready to use. You can start creating beautiful documentation! Remember:

- □□ Change the default password immediately!
- □□ Create books to organize your content

- ✍ Use the rich editor to write beautiful documentation
- □□ Use search to find content quickly
- □□ Invite team members and set appropriate permissions

Next steps: Create your first book, write some pages, upload images, and explore all the features. BookStack makes documentation fun and easy!

16: Uptime Kuma - Uptime Monitoring & Status Pages

Uptime Kuma is a beautiful, self-hosted monitoring tool that watches your websites, servers, and services 24/7. It tells you immediately when something goes down, shows you uptime statistics, and can even create public status pages (like status.github.com) to show your users that everything is working.

For advanced features, API documentation, and customization options, see the official Uptime Kuma documentation.

Why Uptime Kuma?

- **24/7 monitoring** - Know immediately when something breaks
- **Beautiful dashboard** - See all your services at a glance with colorful graphs
- **90+ notification options** - Get alerts via Discord, Slack, Email, Telegram, and many more
- **Public status pages** - Create beautiful status pages to share with your users
- **Docker monitoring** - Monitor your Docker containers automatically
- **Multiple monitor types** - Websites, APIs, ports, DNS, and more
- **Uptime statistics** - See how reliable your services are over time
- **Free and open source** - No subscription fees, runs on your server

Prerequisites

- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `status.yourdomain.com`

Note: Uptime Kuma works great with Traefik and a domain name. Having a friendly URL like `status.yourdomain.com` makes it easy to access your monitoring dashboard and share status pages.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install Uptime Kuma

1. Go to □□ **APPLICATIONS**
2. Select **Uptime Kuma**
3. Choose **Install Uptime Kuma**

Using the Infinity Tools GUI

- Use **↑/↓** to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (recommended)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `status.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://status.yourdomain.com` with trusted SSL
  - **Pick this if:** You want secure, easy access with a domain name (recommended)
- **Standalone HTTP**
  - **What it is:** Uses HTTP with direct port access (no SSL)
  - **What you need:** Just a free port (default: 3001)
  - **What you get:** URL like `http://SERVER_IP:3001`
  - **Pick this if:** You're just testing or only using it on your local network

Simple rule of thumb: Use **Traefik** if you have a domain and want secure access. Use **Standalone HTTP** only for testing or private use.

Step 2.2: If You Choose Traefik

1. Enter your subdomain, e.g., `status.yourdomain.com`
2. Ensure the subdomain's DNS A record points to your server (see Chapter 4.5)
3. Infinity Tools will configure HTTPS automatically via Let's Encrypt

After install: Your Uptime Kuma will be available at `https://status.yourdomain.com`

Step 2.3: If You Choose Standalone

1. Pick a port (default: 3001)
2. You'll access Uptime Kuma via `http://SERVER_IP:3001`

Step 2.4: Docker Container Monitoring (Optional)

You'll be asked if you want to enable Docker container monitoring:

- **Yes**
  - **What it does:** Uptime Kuma can monitor your Docker containers automatically
  - **What you get:** Alerts when containers stop, health status monitoring
  - **Pick this if:** You want to monitor your Docker containers (recommended if you use Docker)
- **No**
  - **What it does:** Only monitors websites, APIs, and ports (not Docker containers)
  - **Pick this if:** You don't use Docker or don't need container monitoring

Step 2.5: Timezone (Optional)

You can set your timezone for monitoring logs and graphs. Examples:

- `America/New_York`
- `Europe/London`
- `Asia/Tokyo`

Leave empty for UTC (default).

What Happens During Installation

- Uptime Kuma container is created
- Data directory is set up at `/opt/speedbits/uptime-kuma`
- Optional domain + Traefik HTTPS routing (if using Traefik)
- Optional Docker socket access (if Docker monitoring enabled)
- Service starts and becomes accessible

Step 3: Access Uptime Kuma

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://status.yourdomain.com` in your browser
3. You'll see the Uptime Kuma setup wizard

If Using Standalone

1. Open `http://SERVER_IP:3001` in your browser
2. You'll see the Uptime Kuma setup wizard

Step 4: Create Your Admin Account

⚠ **CRITICAL:** Uptime Kuma requires you to create admin credentials on your FIRST login. There is NO default password!

Setup Steps

1. You'll see: "Create your admin account"
2. Enter a username (choose any username you like)
3. Enter a password:
   - Minimum: 8 characters
   - Recommended: 12+ characters
   - Best: 20+ characters (use a password manager!)
4. ⚠ **WRITE DOWN YOUR CREDENTIALS IMMEDIATELY!**
   - This is your ONLY chance to set the initial password
   - There is NO "forgot password" on first setup!
   - Use a password manager (like Vaultwarden from Chapter 7) to store it securely
5. Click "Create"
6. Done! You'll see the monitoring dashboard

If You Forget Your Password

Don't worry! You can reset it using the command line:

1. Run: `docker exec -it uptime-kuma npm run reset-password`
2. Enter your username
3. Enter a new password
4. Log in with your new password

Step 5: Add Your First Monitor

Now that you're logged in, let's start monitoring something!

Adding a Monitor

1. Click **"Add New Monitor"** (big button on the dashboard)
2. Choose monitor type:
   - **HTTP(s)** - Monitor websites and APIs
   - **TCP Port** - Monitor if a port is open (SSH, databases, etc.)
   - **Ping** - Check if a server responds
   - **Docker Container** - Monitor Docker containers (if enabled)
   - **DNS** - Check DNS records
   - And more!
3. Enter the URL or IP address to monitor
4. Set check interval (default: 60 seconds - how often to check)
5. Click **"Save"**

Example: Monitor Your Website

1. Type: **HTTP(s)**
2. URL: `https://yourdomain.com`
3. Check interval: 60 seconds
4. Click "Save"

Uptime Kuma will now check your website every 60 seconds and show you if it's up or down!

Step 6: Set Up Notifications

To get alerts when something goes down, you need to configure notifications.

Setting Up Notifications

1. Go to: **Settings** → **Notifications**
2. Click **"Setup Notification"**
3. Choose a provider:
   - **Discord** - Get alerts in Discord
   - **Slack** - Get alerts in Slack
   - **Telegram** - Get alerts via Telegram
   - **Email** - Get alerts via email
   - **Apprise** - Use Apprise for 80+ services (if you have Apprise installed)
   - And 80+ more!
4. Follow the setup instructions for your chosen provider
5. Test the notification

6. Click "Save"

Using Apprise (If Installed)

If you have Apprise installed (Chapter 5), you can use it for notifications:

1. Type: **Apprise (Self-hosted)**
2. URL: `http://apprise:8000/notify/{YOUR-KEY}`
3. This lets you use all 80+ Apprise notification services!

Step 7: Create a Status Page (Optional)

Status pages let you show your users that your services are working. They're public (no login required) and look professional.

Creating a Status Page

1. Go to: **Status Pages**
2. Click **"New Status Page"**
3. Enter a name, e.g., "My Services Status"
4. Choose which monitors to display publicly
5. Customize the appearance (colors, logo, etc.)
6. Click **"Save"**
7. Share the public URL with your users!

What You Can Monitor

Monitor Types

- □□ **HTTP(s) websites** - Check if websites are online
- □□ **TCP ports** - Check if ports are open (SSH, databases, etc.)
- □□ **Ping (ICMP)** - Check if servers respond
- □□ **DNS records** - Check DNS configuration
- □□ **Docker containers** - Monitor container health (if enabled)
- □□ **Keyword detection** - Check if a page contains specific text
- □□ **SMTP email servers** - Check if email servers work
- □□ **gRPC services** - Monitor gRPC APIs

What You'll See

- □□ **Dashboard** - All monitors at a glance with colorful status indicators
- □□ **Uptime graphs** - See uptime percentages over time
- □ **Response times** - See how fast your services respond
- □□ **Incident history** - See when things went down and came back up
- □□ **Alerts** - Get notified immediately when something breaks

Security Recommendations

- □ **Use Traefik mode** - Provides trusted SSL certificates
- □ **Strong password** - Use a password manager to generate strong passwords
- □ **Enable 2FA** - Go to Settings → Security → Two-Factor Auth
- □ **Regular backups** - Uptime Kuma data is stored in `/opt/speedbits/uptime-kuma`
- □ **Protect admin panel** - Monitoring data can be sensitive, keep it secure!
- □ **Use status pages** - Share public status pages instead of giving access to the admin panel

Troubleshooting

Can't Access Uptime Kuma

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Check containers:** Run `docker ps | grep uptime-kuma` to see if container is running
- **Check logs:** Run `docker logs uptime-kuma` to see error messages

Can't Create Admin Account

- Make sure you're accessing Uptime Kuma for the first time (no account exists yet)
- Check container logs: `docker logs uptime-kuma`
- Try resetting: `docker restart uptime-kuma`

Monitors Not Working

- Check that the URL or IP address is correct
- Verify the service is actually running
- Check firewall rules (ports might be blocked)
- Look at monitor details for error messages

Notifications Not Sending

- Test the notification in Settings → Notifications
- Check notification provider settings (Discord webhook, email SMTP, etc.)
- Verify network connectivity from the container

Where to Find Uptime Kuma After Install

- On the finish screen, Infinity Tools prints the access URL
- You can also see it in □□ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/uptime-kuma`
- Data stored in: `/opt/speedbits/uptime-kuma/data`

Backing Up Your Monitoring Data

Your Uptime Kuma data is stored in:

- **Data directory:** `/opt/speedbits/uptime-kuma/data`

To backup:

```
cd /opt/speedbits
tar czf uptime-kuma-backup.tar.gz uptime-kuma/
```

To restore: Extract the backup and restart the Uptime Kuma container.

Or use Uptime Kuma's built-in backup: Go to Settings → Backup → Download Backup

You're Ready!

Uptime Kuma is now installed and ready to monitor your services! Remember:

- □□ Create your admin account on first login (no default password!)
- □□ Add monitors to start tracking uptime
- □□ Set up notifications to get alerts
- □□ Create status pages to share with users
- □□ Enable Docker monitoring if you use Docker containers

Next steps: Add monitors for your websites and services, configure notifications, and create a status page. Uptime Kuma will help you keep everything running smoothly!

17: Netdata - Real-time Performance Monitoring

Netdata is a powerful, real-time monitoring tool that shows you exactly what's happening on your server right now. It displays beautiful graphs of CPU, memory, disk, network, and Docker containers - updating every single second! Think of it as a real-time health dashboard for your entire server.

For advanced features, API documentation, and customization options, see the official Netdata documentation.

Why Netdata?

- **Real-time monitoring** - See updates every second, not every minute
- **Beautiful graphs** - Colorful, easy-to-read charts for everything
- **Zero configuration** - Works immediately after installation
- **Auto-discovery** - Automatically finds and monitors all Docker containers
- **Low resource usage** - Uses only ~50MB of RAM
- **Comprehensive metrics** - CPU, RAM, disk, network, processes, and more
- **Alert notifications** - Get notified when something goes wrong
- **Historical data** - See trends over time

Prerequisites

- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `monitor.yourdomain.com`
- **Optional: Apprise installed** (from Chapter 5) for alert notifications

Note: Netdata works great with Traefik and a domain name. Having a friendly URL like `monitor.yourdomain.com` makes it easy to access your monitoring dashboard.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install Netdata

1. Go to □□ **APPLICATIONS**
2. Select **Netdata**
3. Choose **Install Netdata**

Using the Infinity Tools GUI

- Use ↑/↓ to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (recommended)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `monitor.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://monitor.yourdomain.com` with trusted SSL
  - **Pick this if:** You want secure, easy access with a domain name (recommended)
- **Standalone HTTP**
  - **What it is:** Uses HTTP with direct port access (no SSL)
  - **What you need:** Just a free port (default: 19999)
  - **What you get:** URL like `http://SERVER_IP:19999`
  - **Pick this if:** You're just testing or only using it on your local network

Simple rule of thumb: Use **Traefik** if you have a domain and want secure access. Use **Standalone HTTP** only for testing or private use.

Step 2.2: If You Choose Traefik

1. Enter your subdomain, e.g., `monitor.yourdomain.com`
2. Ensure the subdomain's DNS A record points to your server (see Chapter 4.5)
3. Infinity Tools will configure HTTPS automatically via Let's Encrypt

After install: Your Netdata will be available at `https://monitor.yourdomain.com`

Step 2.3: If You Choose Standalone

1. Pick a port (default: 19999)
2. You'll access Netdata via `http://SERVER_IP:19999`

Step 2.4: Multi-Server Monitoring (Optional)

You'll be asked if you want to stream metrics to a Netdata Director (parent server):

- **Yes**
  - **What it does:** Sends all metrics to a central dashboard
  - **What you need:** A Netdata Director server already set up
  - **What you get:** Centralized monitoring of multiple servers
  - **Pick this if:** You have multiple servers and want one dashboard for all
- **No (default)**
  - **What it does:** Standalone monitoring (this server only)
  - **Pick this if:** You're just monitoring one server

Step 2.5: Apprise Notifications (Optional)

If you have Apprise installed (Chapter 5), you can enable alert notifications:

- **Yes**
  - **What it does:** Sends alerts to Apprise when CPU, RAM, or disk usage is high
  - **What you get:** Notifications via Discord, Slack, Email, etc. (all Apprise services)
  - **Pick this if:** You want to get alerts when something goes wrong
- **No**
  - **What it does:** No alert notifications
  - **Pick this if:** You just want to view metrics, no alerts

What Happens During Installation

- Netdata container is created
- Data directories are set up at `/opt/speedbits/netdata-client`
- Optional domain + Traefik HTTPS routing (if using Traefik)
- Docker socket access configured (for container monitoring)
- Custom alerts configured (CPU, RAM, disk)
- Optional Apprise integration (if enabled)
- Service starts and becomes accessible

Step 3: Access Netdata

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://monitor.yourdomain.com` in your browser
3. You'll see the Netdata dashboard immediately!

If Using Standalone

1. Open `http://SERVER_IP:19999` in your browser
2. You'll see the Netdata dashboard immediately!

⚠ **IMPORTANT SECURITY NOTE:** Netdata has NO username/password protection by default! Anyone who can access the URL can see your monitoring data. If using Traefik, strongly consider adding Basic Auth protection. If using standalone mode, keep it on a private network only!

Step 4: Understanding the Dashboard

When you first open Netdata, you'll see a beautiful dashboard with lots of graphs. Here's what everything means:

Main Sections

- □□ **System Overview** - CPU, RAM, disk, network at a glance
- □□ **Docker Containers** - All your containers with individual metrics
- □□ **Disk I/O** - How fast your disks are reading/writing
- □□ **Network** - Network traffic and connections
- ⚙ **System Load** - How busy your server is
- □□ **Processes** - Individual programs and their resource usage

Reading the Graphs

- **Green** - Normal, healthy values
- **Yellow** - Warning (getting high)
- **Red** - Critical (too high!)
- **Time axis** - Shows last hour by default (can zoom in/out)
- **Real-time** - Updates every second automatically

Key Metrics to Watch

- □□ **CPU Usage** - Should be under 80% most of the time
- □□ **RAM Usage** - Should be under 80% most of the time
- □□ **Disk Usage** - Should be under 80% (watch for low disk space!)
- □□ **Network Traffic** - Shows incoming/outgoing data

- □□ **Container Status** - All containers should be running

Step 5: Docker Container Monitoring

One of Netdata's best features is automatic Docker container discovery and monitoring!

What You'll See

- All your Docker containers listed automatically
- Individual CPU, RAM, disk, and network usage for each container
- Container health status
- Real-time graphs for each container

How to Use It

1. Click on "Docker" in the left sidebar
2. You'll see all your containers listed
3. Click on any container to see its detailed metrics
4. Watch for containers using too much CPU or RAM

Step 6: Alert Notifications (If Enabled)

If you enabled Apprise notifications, Netdata will automatically send alerts when:

- ⚠ **CPU usage** > 80% (warning) or > 95% (critical)
- ⚠ **RAM usage** > 80% (warning) or > 95% (critical)
- ⚠ **Disk space** > 80% (warning) or > 90% (critical)

How Alerts Work

1. Netdata detects a problem (e.g., CPU too high)
2. Sends alert to Apprise
3. Apprise forwards to your configured channels (Discord, Slack, Email, etc.)
4. You get notified immediately!

Customizing Alerts

You can customize alert thresholds by editing configuration files:

```
nano /opt/speedbits/netdata-client/netdata/health.d/cpu_usage.conf
```

Change the warning/critical thresholds to your preferences.

Security Recommendations

- **Use Traefik mode** - Provides trusted SSL certificates
- **Add Basic Auth** - Protect dashboard with username/password (strongly recommended!)
- **Private network only** - If using standalone mode, don't expose to internet
- **Use VPN** - Access via WireGuard VPN (Chapter 18) for secure remote access
- **SSH tunnel** - Use SSH tunnel for secure access: `ssh -L 19999:localhost:19999 user@server`
- ⚠ **No default protection** - Netdata has NO username/password by default - protect it!

Adding Basic Auth Protection

If using Traefik, you can add username/password protection:

1. Run: `sudo bash Infrastructure/websiteprotection.sh`
2. Select "netdata"
3. Enter username and password
4. Now your dashboard is protected!

Troubleshooting

Can't Access Netdata

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Check containers:** Run `docker ps | grep netdata` to see if container is running
- **Check logs:** Run `docker logs netdata` to see error messages

No Docker Containers Showing

- Make sure Docker is running: `docker ps`
- Check that Docker socket is accessible: `docker exec netdata ls /var/run/docker.sock`
- Restart Netdata: `docker restart netdata`

Alerts Not Working

- Check that Apprise is running: `docker ps | grep apprise`
- Verify Apprise network connectivity
- Check Netdata logs: `docker logs netdata`
- Test alert thresholds (trigger a test alert)

High Resource Usage

- Netdata uses ~50MB RAM (very low!)
- If seeing high CPU, check how many containers you're monitoring
- Consider reducing data retention period

Where to Find Netdata After Install

- On the finish screen, Infinity Tools prints the access URL
- You can also see it in ◫ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/netdata-client`
- Configuration files: `/opt/speedbits/netdata-client/netdata/`

Useful Features

Historical Data

Netdata stores historical data so you can see trends over time:

- Zoom in/out on graphs to see different time periods
- Compare current vs. past performance
- Identify patterns and trends

Exporting Data

You can export graphs and data:

- Click on any graph to see export options
- Export as image (PNG)
- Share graphs with others

Custom Dashboards

Netdata allows you to create custom dashboards:

- Focus on specific metrics
- Create views for different purposes
- Save favorite views

You're Ready!

Netdata is now installed and monitoring your server in real-time! Remember:

- □□ Protect your dashboard with Basic Auth (strongly recommended!)
- □□ Check the dashboard regularly to understand your server's health
- □□ Set up alerts to get notified of problems
- □□ Use Docker monitoring to track container health
- □□ Use historical data to identify trends and plan capacity

Next steps: Explore the dashboard, check your Docker containers, set up alerts, and use Netdata to keep your server running smoothly!

18: Netdata Director - Multi-Server Monitoring Hub

Netdata Director is a centralized monitoring dashboard that lets you monitor multiple servers from one place. Instead of opening separate dashboards for each server, you get one unified view showing all your servers' metrics, alerts, and performance data. Think of it as a command center for all your infrastructure!

⚠ **IMPORTANT:** Netdata Director is a Pro+ feature that requires a license. For single-server monitoring, use regular Netdata (Chapter 17) which is free.

For advanced features, API documentation, and customization options, see the official Netdata documentation.

Why Netdata Director?

- **One dashboard for all servers** - Monitor everything from one place
- **Unified view** - See all your servers' metrics together
- **Centralized alerts** - Get notifications from all servers in one place
- **Historical data** - Long-term storage for all monitored servers
- **Node comparison** - Compare performance across servers
- **Easy management** - Add or remove servers easily
- **Scalable** - Monitor unlimited servers

Director vs Regular Netdata

Regular Netdata (Free)

- Monitor individual servers separately
- Each server has its own dashboard
- Free and open source
- Must open multiple dashboards for multiple servers
- No unified view

Netdata Director (Pro+)

- Monitor ALL servers from one dashboard
- Unified monitoring interface
- Centralized alerts and management
- Historical data for all servers
- ⚠ Requires Pro+ license

Prerequisites

- **Pro+ License** - Netdata Director requires a Pro+ license
- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `monitoring.yourdomain.com`
- **Optional: Apprise installed** (from Chapter 5) for alert notifications
- **Multiple servers** - Director is most useful when monitoring 2+ servers

Note: Netdata Director works best with Traefik and a domain name. Having a friendly URL like `monitoring.yourdomain.com` makes it easy to access your centralized dashboard.

How It Works

Netdata Director uses a parent-child architecture:

Director (Parent)

- The central dashboard server
- Receives metrics from all child nodes
- Displays unified view of all servers
- Manages alerts for all servers

Child Nodes

- Regular Netdata installations on each server
- Stream metrics to the Director
- Can still have their own dashboards (optional)
- Automatically appear in Director dashboard

Step 1: Verify Pro+ License

Before installing, make sure you have a Pro+ license. The installation script will check for this automatically.

If you don't have a Pro+ license:

- Visit: https://speedbits.io/infinity-tools/
- Email: sales@speedbits.io
- Support: support@speedbits.io

Step 2: Start Infinity Tools

```
sudo infinity-tools
```

Step 3: Install Netdata Director

1. Go to □□ **APPLICATIONS**
2. Select **Netdata Director**
3. Choose **Install Netdata Director**

Using the Infinity Tools GUI

- Use **↑/↓** to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 3.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (recommended)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `monitoring.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://monitoring.yourdomain.com` with trusted SSL
  - **Pick this if:** You want secure, easy access with a domain name (recommended)
- **Standalone HTTP**
  - **What it is:** Uses HTTP with direct port access (no SSL)
  - **What you need:** Just a free port (default: 19999)

- **What you get:** URL like `http://SERVER_IP:19999`
- **Pick this if:** You're just testing or only using it on your local network

Step 3.2: If You Choose Traefik

1. Enter your subdomain, e.g., `monitoring.yourdomain.com`
2. Ensure the subdomain's DNS A record points to your server (see Chapter 4.5)
3. Infinity Tools will configure HTTPS automatically via Let's Encrypt

After install: Your Netdata Director will be available at `https://monitoring.yourdomain.com`

Step 3.3: Apprise Notifications (Optional)

If you have Apprise installed (Chapter 5), you can enable centralized alert notifications:

- **Yes**
  - **What it does:** Sends alerts from ALL monitored servers to Apprise
  - **What you get:** One notification channel for all servers
  - **Pick this if:** You want centralized alerting
- **No**
  - **What it does:** No alert notifications
  - **Pick this if:** You just want monitoring, no alerts

Step 3.4: Save Your Stream API Key

⚠ **CRITICAL:** During installation, a Stream API Key will be generated. This key is used to connect child nodes to the Director.

- Write down the API key immediately!
- Save it in a password manager (like Vaultwarden)
- You'll need this key when configuring child nodes
- The key is also saved in: `/opt/speedbits/netdata-director/stream-api-key.txt`

What Happens During Installation

- Netdata Director container is created
- Data directories are set up at `/opt/speedbits/netdata-director`
- Stream API key is generated and saved
- Optional domain + Traefik HTTPS routing (if using Traefik)
- Optional Apprise integration (if enabled)
- Service starts and becomes accessible

Step 4: Access Netdata Director

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://monitoring.yourdomain.com` in your browser
3. You'll see the Director dashboard!

If Using Standalone

1. Open `http://SERVER_IP:19999` in your browser
2. You'll see the Director dashboard!

⚠ **IMPORTANT SECURITY NOTE:** Netdata Director has NO username/password protection by default! However, you CANNOT use Basic Auth because it blocks child nodes from streaming data. Instead, use firewall rules, VPN access, or Netdata Cloud for security.

Step 5: Connect Child Nodes

Now that Director is running, you need to connect your other servers (child nodes) to it.

On Each Server You Want to Monitor

1. SSH into the server
2. Run: `sudo infinity-tools`
3. Go to ◫ **APPLICATIONS → Netdata → Install**
4. When asked about streaming, choose **Yes**
5. Enter Director details:
   - **Director hostname/IP:** The Director server's IP or domain
   - **Director port:** 19999 (or your custom port)
   - **Stream API key:** The key you saved during Director installation
6. Complete the installation

What Happens Next

- Child node starts streaming metrics to Director
- Wait 1-2 minutes for connection to establish
- Child node appears in Director dashboard dropdown
- You can now switch between servers in the Director dashboard!

Step 6: Using the Director Dashboard

Switching Between Servers

1. Open the Director dashboard
2. Look for a dropdown menu (usually top-left or top-right)
3. Select a server from the dropdown
4. Dashboard updates to show that server's metrics

What You'll See

- □□ **Unified view** - All servers listed in dropdown
- □□ **Real-time metrics** - Same as regular Netdata, but for all servers
- □□ **Centralized alerts** - Alerts from all servers
- □□ **Historical data** - Long-term storage for all servers
- □□ **Node comparison** - Compare metrics across servers

Security Recommendations

- □ **Use Traefik mode** - Provides trusted SSL certificates
- □ **Firewall protection** - Restrict access to trusted IPs only
- □ **VPN access** - Use WireGuard VPN (Chapter 18) for secure remote access
- □ **Keep API key secret** - Only share with trusted servers
- ⚠ **NO Basic Auth** - Cannot use Basic Auth (blocks child nodes)
- ⚠ **Protect Director** - Director shows ALL server metrics - protect it!

Security Options

Since Basic Auth doesn't work with Director, use these alternatives:

- **Option 1: Firewall (Best)**
  - Use UFW to allow only trusted IPs
  - Example: `ufw allow from CHILD_NODE_IP to any port 443`
- **Option 2: VPN/WireGuard**
  - Access Director only through WireGuard VPN
  - Keep Director on internal network
- **Option 3: Netdata Cloud**
  - Use Netdata's official cloud service
  - Includes authentication and team management
  - Visit: https://app.netdata.cloud

Troubleshooting

Can't Access Director

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Check containers:** Run `docker ps | grep netdata-director` to see if container is running
- **Check logs:** Run `docker logs netdata-director` to see error messages

Child Nodes Not Appearing

- Wait 1-2 minutes after connecting (connection takes time)
- Verify API key is correct on child node
- Check network connectivity between child and Director
- Check Director logs: `docker logs netdata-director`
- Verify child node is streaming: Check child node logs

Lost API Key

- View saved key: `cat /opt/speedbits/netdata-director/stream-api-key.txt`
- Or check Director configuration: `cat /opt/speedbits/netdata-director/netdata/stream.conf`

Where to Find Netdata Director After Install

- On the finish screen, Infinity Tools prints the access URL
- You can also see it in ◫ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/netdata-director`
- API key saved in: `/opt/speedbits/netdata-director/stream-api-key.txt`
- Configuration files: `/opt/speedbits/netdata-director/netdata/`

Useful Features

Data Retention

Director stores historical data for all servers:

- **High-resolution:** 1 hour (1-second granularity)

- **Mid-resolution:** 1 day (1-minute granularity)
- **Low-resolution:** 30 days (15-minute granularity)

Centralized Alerts

If Apprise is enabled, you'll get alerts from all servers:

- Alerts include server hostname
- One notification channel for all servers
- Easier to manage than individual alerts

You're Ready!

Netdata Director is now installed and ready to monitor multiple servers! Remember:

- □□ Protect your Director with firewall or VPN (cannot use Basic Auth)
- □□ Keep your Stream API key secret
- □□ Connect child nodes to start monitoring
- □□ Set up centralized alerts for all servers
- □□ Use the unified dashboard to monitor everything

Next steps: Connect your first child node, verify it appears in the dashboard, and start monitoring all your servers from one place!

19: Installing WireGuard - Secure VPN Access

WireGuard is a modern, fast, and secure VPN (Virtual Private Network) that lets you access your server and its services securely from anywhere. Once connected, you can access internal services, manage your server, and browse securely - all encrypted and protected!

For advanced features, API documentation, and technical details, see the official WireGuard documentation.

Why WireGuard?

- **Secure access** - Access your server and services securely from anywhere
- **Easy to use** - Web interface makes managing clients simple
- **Fast and modern** - Uses modern encryption (ChaCha20) for speed and security
- **Mobile-friendly** - QR codes for easy mobile device setup
- **Multiple devices** - Connect phones, laptops, tablets - all from one server
- **Split tunneling** - Only VPN traffic goes through VPN, rest uses normal internet
- **Low overhead** - Minimal impact on your internet speed

Prerequisites

- **Docker running** (from Chapter 3)
- **Linux kernel 5.6+** - Most modern Linux distributions have this
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `vpn.yourdomain.com`
- **Firewall access** - Ability to open UDP port (default: 51820)

Note: WireGuard works great with Traefik and a domain name. Having a friendly URL like `vpn.yourdomain.com` makes it easy to access the web management interface.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install WireGuard

1. Go to □□ **APPLICATIONS**
2. Select **WireGuard**
3. Choose **Install WireGuard**

Using the Infinity Tools GUI

- Use ↑/↓ to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Network Configuration

You'll be asked to configure two networks:

VPN Network (Default: 10.13.13)

- **What it is:** The network used by WireGuard clients and Docker services
- **What you get:** Clients get IPs like 10.13.13.3, 10.13.13.4, etc.
- **Default:** 10.13.13 (usually fine to accept)
- **Pick this if:** You want the default setup (recommended)

Host Network (Default: 10.13.14)

- **What it is:** The network used for accessing host services (like Webmin, Apprise)
- **What you get:** Host services accessible at 10.13.14.1
- **Default:** 10.13.14 (usually fine to accept)
- **Pick this if:** You want the default setup (recommended)

□□ **Tip:** Unless you have a specific reason, accept the defaults (just press Enter).

Step 2.2: DNS Configuration

WireGuard will automatically detect your server's DNS settings. This ensures VPN clients use the same DNS as your server for consistency.

Usually, you can just accept the auto-detected DNS (press Enter).

Step 2.3: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (optional)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `vpn.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://vpn.yourdomain.com` with trusted SSL
  - **Pick this if:** You want secure, easy access with a domain name
- **Standalone (recommended)**
  - **What it is:** Uses HTTPS with a self-signed certificate and direct port access
  - **What you need:** Just a free port (default: 8445)
  - **What you get:** URL like `https://SERVER_IP:8445` with a warning you must accept once
  - **Pick this if:** You don't have a domain or prefer direct access (recommended)

Simple rule of thumb: Use **Standalone** for most cases. Use **Traefik** if you have a domain and want trusted SSL.

Step 2.4: VPN Port Configuration

You'll be asked for the UDP port for VPN connections:

- **Default:** 51820
- **What it is:** The port clients will connect to
- **Important:** You must open this port in your firewall!
- **Pick this if:** Default is fine (recommended)

Step 2.5: Server Endpoint

You'll be asked for your server's public IP address or domain name:

- **What it is:** How clients will find your server
- **Examples:** `123.45.67.89` or `vpn.yourdomain.com`
- **Important:** This must be accessible from the internet!

What Happens During Installation

- WireGuard kernel module is installed (if needed)
- WireGuard container is created
- Web management interface is set up
- Random password is generated for web UI
- Host network interface is created
- Network routing is configured
- Service starts and becomes accessible

Step 3: Open Firewall Port

⚠ **CRITICAL:** You MUST open the VPN port in your firewall, or clients cannot connect!

Opening the Port

```
sudo ufw allow 51820/udp
```

Replace `51820` with your custom port if you chose a different one.

Why This Matters

- Without this, VPN clients cannot connect to your server
- The port must be UDP (not TCP)
- This is the ONLY port you need to open for VPN access

Step 4: Access WireGuard Web Interface

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://vpn.yourdomain.com` in your browser
3. You'll see the WireGuard login page

If Using Standalone

1. Open `https://SERVER_IP:8445` in your browser
2. You'll see a security warning (normal for self-signed certificates)
3. Click "Advanced" → "Proceed to site" to continue
4. You'll see the WireGuard login page

Step 5: Login to Web Interface

⚠ **CRITICAL:** During installation, a random password was generated and displayed. Save it immediately!

Default Credentials

- **Username:** `admin`
- **Password:** Randomly generated (shown during installation)

If You Lost the Password

You can retrieve it from:

```
cat /opt/speedbits/wireguard/web-password.txt
```

Login Steps

1. Enter username: `admin`
2. Enter the password shown during installation
3. Click "Login"
4. You'll see the WireGuard dashboard!

Step 6: Create Your First VPN Client

Now that you're logged in, let's create your first VPN client!

Adding a Client

1. Click **"Add Client"** or the **"+"** button
2. Enter a name for your device, e.g., "My Phone", "Laptop", "Work PC"
3. Configure settings (or use defaults):
   - **Allowed IPs:** Usually auto-filled (VPN network + Host network)
   - **Use Server DNS:** Usually enabled (recommended)
4. Click **"Save"** or **"Create"**
5. You'll see a QR code and download options!

What You'll Get

- □□ **QR Code** - Scan with mobile devices
- □□ **Config File** - Download for Windows/Linux
- □□ **Client Details** - IP address, public key, etc.

Step 7: Set Up WireGuard on Your Device

Windows

1. Install WireGuard from Microsoft Store
2. Open WireGuard app
3. Click **"Add Tunnel"** → **"Import from file"**
4. Select the downloaded .conf file
5. Click **"Activate"** to connect

Android/iOS/macOS

1. Install WireGuard app from Play Store/App Store
2. Open WireGuard app
3. Tap **"+"** → **"Create from QR code"**
4. Scan the QR code from the web interface
5. Tap **"Activate"** to connect

Linux

1. Install WireGuard: `sudo apt install wireguard`
2. Copy the .conf file to: `/etc/wireguard/wg0.conf`
3. Start WireGuard: `sudo wg-quick up wg0`
4. Enable auto-start: `sudo systemctl enable wg-quick@wg0`

Step 8: Understanding VPN Networks

WireGuard creates two networks for different purposes:

VPN Network (10.13.13.0/24)

This network is for WireGuard clients and Docker services:

- □□ **Your devices** - Get IPs like 10.13.13.3, 10.13.13.4, etc.
- □□ **Docker services** - Accessible via their container names
- □□ **Examples:**
  - Vaultwarden: `http://vaultwarden:80`
  - WordPress: `http://wordpress:80`

    - Apprise: `http://apprise:8000`

Host Network (10.13.14.0/24)

This network is for accessing host services (services running directly on the server):

- **Host services** - Accessible at 10.13.14.1
- **Examples:**
    - Webmin: `https://10.13.14.1:8443`
    - Apprise: `http://10.13.14.1:8444`
    - SSH: `ssh user@10.13.14.1`

What You Can Access via VPN

Docker Services (VPN Network)

- All your Infinity Tools applications
- Access via container names (e.g., `http://vaultwarden:80`)
- No need to expose ports publicly!

Host Services (Host Network)

- Webmin (if installed)
- Apprise (if installed)
- SSH access
- Any other services running on the host

Security Recommendations

- **Open only VPN port** - Close other public ports (Webmin, Apprise, etc.)
- **Use strong password** - The generated password is strong, keep it safe!
- **Store password securely** - Use a password manager (Vaultwarden recommended!)
- **Limit client access** - Only create clients for trusted devices
- **Disable unused clients** - Turn off clients you're not using
- **Keep WireGuard updated** - Re-run install script periodically for updates
- ⚠ **Protect web interface** - The web UI manages all VPN clients - keep it secure!

Firewall Best Practices

After setting up WireGuard, you can close other public ports:

```
# Close Webmin public access (access via VPN instead)
sudo ufw delete allow 8443

# Close Apprise public access (access via VPN instead)
sudo ufw delete allow 8444

# Close WireGuard web UI public access (access via VPN instead)
sudo ufw delete allow 8445
```

Now access everything securely via VPN!

Troubleshooting

Can't Connect to VPN

- **Check firewall:** Make sure UDP port 51820 (or your custom port) is open
- **Check server endpoint:** Verify the IP/domain is correct and accessible
- **Check client config:** Make sure you're using the correct .conf file
- **Check WireGuard status:** Run `docker logs wireguard` to see errors

Can't Access Web Interface

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Standalone mode:** Accept the self-signed certificate warning
- **Check container:** Run `docker ps | grep wireguard` to see if it's running
- **Check logs:** Run `docker logs wireguard` to see error messages

Can't Access Services via VPN

- **Check VPN connection:** Make sure WireGuard is connected on your device
- **Check IP address:** Verify you're using the correct IPs (10.13.13.x or 10.13.14.1)
- **Check Allowed IPs:** Make sure client config includes both VPN and Host networks
- **Check routing:** Verify network routing is configured correctly

Lost Web UI Password

- View saved password: `cat /opt/speedbits/wireguard/web-password.txt`
- If file doesn't exist, you'll need to reinstall WireGuard

Where to Find WireGuard After Install

- On the finish screen, Infinity Tools prints the web interface URL and password
- You can also see it in □□ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/wireguard`
- Password saved in: `/opt/speedbits/wireguard/web-password.txt`
- Client configs: `/opt/speedbits/wireguard/data/` (managed via web UI)

Managing VPN Clients

Adding More Clients

Simply repeat Step 6 for each device you want to connect. Each device gets its own unique IP address.

Disabling Clients

In the web interface, you can disable clients without deleting them. This is useful if you temporarily don't want a device to connect.

Viewing Connection Stats

The web interface shows connection statistics for each client, including data transferred and connection time.

You're Ready!

WireGuard is now installed and ready to use! Remember:

- □□ Open the firewall port (UDP 51820) - critical for connections!
- □□ Save your web UI password securely
- □□ Create clients via the web interface
- □□ Close other public ports and access everything via VPN
- □□ Use VPN network (10.13.13.x) for Docker services
- □□ Use Host network (10.13.14.1) for host services

Next steps: Create your first client, set up WireGuard on your device, test the connection, and start accessing your services securely from anywhere!

20: Warpgate - Secure SSH Gateway

Warpgate is a secure SSH gateway (also called a "bastion host") that provides a web interface for managing SSH access to your server. Instead of connecting directly to your server, you connect through Warpgate, which adds an extra layer of security and makes it easier to manage who can access what.

For advanced features, API documentation, and technical details, see the official Warpgate documentation.

Why Warpgate?

- **Secure SSH gateway** - All SSH connections go through Warpgate
- **Web-based management** - Easy-to-use web interface for managing access
- **User access control** - Control who can access which servers
- **Session recording** - Keep track of SSH sessions for security
- **No direct server access** - Server SSH port can be closed, only Warpgate port open
- **Centralized access** - Manage all SSH access from one place
- **Better security** - Reduces attack surface by closing direct SSH access

Prerequisites

- **Docker running** (from Chapter 3)
- **Optional: Traefik installed** (from Chapter 4) for HTTPS access with a domain
- **Optional: Subdomain** (from Chapter 4.5), e.g., `warpgate.yourdomain.com`
- **Firewall access** - Ability to open port 2222 (SSH) and optionally close port 22

Note: Warpgate works great with Traefik and a domain name. Having a friendly URL like `warpgate.yourdomain.com` makes it easy to access the web management interface.

Step 1: Start Infinity Tools

```
sudo infinity-tools
```

Step 2: Install Warpgate

1. Go to □□ **APPLICATIONS**
2. Select **Warpgate**
3. Choose **Install Warpgate**

Using the Infinity Tools GUI

- Use **↑/↓** to move, **Enter** to select, **Esc** to go back
- Look for the **turquoise cursor** indicating the current selection
- Each screen shows a short description at the top explaining what's needed

Step 2.1: Choose SSL Mode

You'll see two options. Here's what each means:

- **Traefik (recommended)**
  - **What it is:** Uses your domain name with a trusted HTTPS certificate from Let's Encrypt
  - **What you need:** A subdomain (e.g., `warpgate.yourdomain.com`) pointing to your server (see Chapter 4.5)
  - **What you get:** Professional URL like `https://warpgate.yourdomain.com` with trusted SSL
  - **Pick this if:** You have a domain and want secure, easy access (recommended)
- **Standalone**
  - **What it is:** Uses HTTPS with a self-signed certificate and direct port access
  - **What you need:** Just a free port (default: 8888)
  - **What you get:** URL like `https://SERVER_IP:8888` with a warning you must accept once
  - **Pick this if:** You don't have a domain or prefer direct access

Simple rule of thumb: Use **Traefik** if you have a domain (recommended). Use **Standalone** if you don't have a domain.

Step 2.2: Domain Configuration (Traefik Mode)

If you chose Traefik, you'll be asked for your domain:

- **What it is:** The subdomain where Warpgate will be accessible
- **Example:** `warpgate.yourdomain.com`
- **Important:** DNS must already point to your server (see Chapter 4.5)

Step 2.3: Port Configuration (Standalone Mode)

If you chose Standalone, you'll be asked for a port:

- **Default:** 8888
- **What it is:** The port for the web interface
- **Note:** SSH port (2222) is always exposed directly

What Happens During Installation

- Warpgate container is created
- Data directory is set up
- Web interface becomes accessible
- SSH gateway starts on port 2222
- Interactive setup prompts for admin credentials

Step 3: Set Up Admin Account

After installation, Warpgate will run an interactive setup. You'll be prompted to create an admin account:

Admin Setup Prompts

1. **Admin username:** Choose a username for the admin account (e.g., `admin`)
2. **Admin password:** Choose a strong password (you'll use this to log into the web interface)
3. **Confirm password:** Enter the password again to confirm

⚠ **IMPORTANT:** Save these credentials immediately! You'll need them to access the web interface.

Step 4: Access Warpgate Web Interface

If Using Traefik

1. Wait 30-60 seconds for SSL certificate generation
2. Open `https://warpgate.yourdomain.com` in your browser

3. You'll see the Warpgate login page

If Using Standalone

1. Open `https://SERVER_IP:8888` in your browser
2. You'll see a security warning (normal for self-signed certificates)
3. Click "Advanced" → "Proceed to site" to continue
4. You'll see the Warpgate login page

Step 5: Login to Web Interface

1. Enter the admin username you created during setup
2. Enter the admin password you created during setup
3. Click "Login"
4. You'll see the Warpgate dashboard!

Step 6: Understanding Warpgate

Warpgate acts as a gateway (or "bastion") between you and your server:

How It Works

- **Before Warpgate:** You connect directly to your server via SSH (port 22)
- **With Warpgate:** You connect to Warpgate (port 2222), which then connects you to your server
- **Benefits:** All SSH access goes through Warpgate, making it easier to manage and secure

What You Can Do

- **Manage users** - Add users who can access servers through Warpgate
- **Control access** - Decide which users can access which servers
- **View sessions** - See who's connected and what they're doing
- **Record sessions** - Keep logs of SSH sessions for security
- **Manage targets** - Add servers that users can connect to

Step 7: Add Your First Target (Server)

Before users can connect, you need to add a "target" (the server they'll connect to):

Adding a Target

1. In the web interface, go to **"Targets"** or **"Servers"**
2. Click **"Add Target"** or the **"+"** button
3. Enter target details:
   - **Name:** A friendly name (e.g., "My Server")
   - **Host:** The server's IP address or hostname (usually `localhost` or `127.0.0.1` for the same server)
   - **Port:** SSH port (usually `22`)
   - **Username:** The SSH username (e.g., your server username)
4. Click **"Save"** or **"Create"**

For Same-Server Access

If Warpgate is running on the same server you want to access:

- **Host:** `localhost` or `127.0.0.1`
- **Port:** `22` (or your server's SSH port)
- **Username:** Your server username

Step 8: Add Users

Now add users who can connect through Warpgate:

Adding a User

1. In the web interface, go to **"Users"**
2. Click **"Add User"** or the **"+"** button
3. Enter user details:
   - **Username:** A username for Warpgate (e.g., "john")
   - **Password:** A password for this user
   - **Email:** Optional email address
4. Click **"Save"** or **"Create"**

Granting Access

After creating a user, grant them access to targets:

1. Go to the user's profile
2. Find **"Access"** or **"Targets"** section
3. Select which targets this user can access

4. Save the changes

Step 9: Connect via SSH Through Warpgate

Now you can connect to your server through Warpgate:

SSH Connection

```
ssh -p 2222 warpgate-user@warpgate.yourdomain.com
```

Or if using standalone mode:

```
ssh -p 2222 warpgate-user@SERVER_IP
```

What Happens

1. You connect to Warpgate on port 2222
2. Warpgate asks for your Warpgate username and password
3. After authentication, Warpgate shows you available targets
4. You select which target (server) you want to connect to
5. Warpgate connects you to that server

First-Time Connection

On your first connection, you'll see:

1. Warpgate login prompt
2. Enter your Warpgate username and password
3. List of available targets
4. Select a target to connect
5. You're now connected to your server!

Step 10: Security Best Practices

Close Direct SSH Access

Once Warpgate is working, you can close direct SSH access to your server:

```
# Close port 22 (direct SSH)
sudo ufw delete allow 22/tcp

# Keep port 2222 open (Warpgate SSH)
sudo ufw allow 2222/tcp
```

⚠ **WARNING:** Only do this after testing Warpgate! Make sure you can connect through Warpgate before closing port 22.

Firewall Configuration

- **Open port 2222** - Required for Warpgate SSH access
- **Open port 80/443** - If using Traefik (for web interface)
- **Open port 8888** - If using standalone mode (for web interface)
- ⚠ **Close port 22** - After testing Warpgate (optional but recommended)

User Management

- **Use strong passwords** - For both admin and user accounts
- **Limit access** - Only grant access to targets users need
- **Regularly review users** - Remove users who no longer need access
- **Monitor sessions** - Check who's connecting and when

Troubleshooting

Can't Access Web Interface

- **Traefik mode:** Wait 30-60 seconds after installation for SSL certificate generation
- **Standalone mode:** Accept the self-signed certificate warning
- **Check container:** Run `docker ps | grep warpgate` to see if it's running
- **Check logs:** Run `docker logs warpgate` to see error messages

Can't Connect via SSH

- **Check firewall:** Make sure port 2222 is open: `sudo ufw status | grep 2222`
- **Check credentials:** Verify you're using the correct Warpgate username and password
- **Check target:** Make sure the target server is configured correctly
- **Check access:** Verify the user has access to the target

Forgot Admin Password

- You'll need to reinstall Warpgate with `--deleteall` flag
- This will wipe all data and let you create a new admin account
- Make sure to back up any important data first!

Target Connection Fails

- **Check target host:** Verify the host IP/name is correct
- **Check target port:** Verify the SSH port is correct (usually 22)
- **Check target credentials:** Verify the username is correct
- **Test direct connection:** Try connecting directly to the target to verify it's accessible

Where to Find Warpgate After Install

- On the finish screen, Infinity Tools prints the web interface URL
- You can also see it in ◫ **STATUS & HEALTH → STATUS**
- Check the installation directory: `/opt/speedbits/warpgate`
- Configuration file: `/opt/speedbits/warpgate/data/warpgate.yaml`
- Database: `/opt/speedbits/warpgate/data/db/`

Managing Warpgate

Adding More Users

Simply repeat Step 8 for each user you want to add. Each user can have access to different targets.

Adding More Targets

Add more servers by repeating Step 7. Users can then be granted access to these new targets.

Viewing Sessions

The web interface shows active SSH sessions, including who's connected and what they're doing.

Session Recording

Warpgate can record SSH sessions for security auditing. Check the settings in the web interface to enable this.

You're Ready!

Warpgate is now installed and ready to use! Remember:

- □□ Save your admin credentials securely
- □□ Connect via port 2222 (not port 22)
- □□ Add users and grant them access to targets
- □□ Close port 22 after testing (optional but recommended)
- □□ Monitor sessions in the web interface
- □□ Use strong passwords for all accounts

Next steps: Add your first target, create users, grant access, test SSH connection through Warpgate, and optionally close direct SSH access (port 22) for better security!