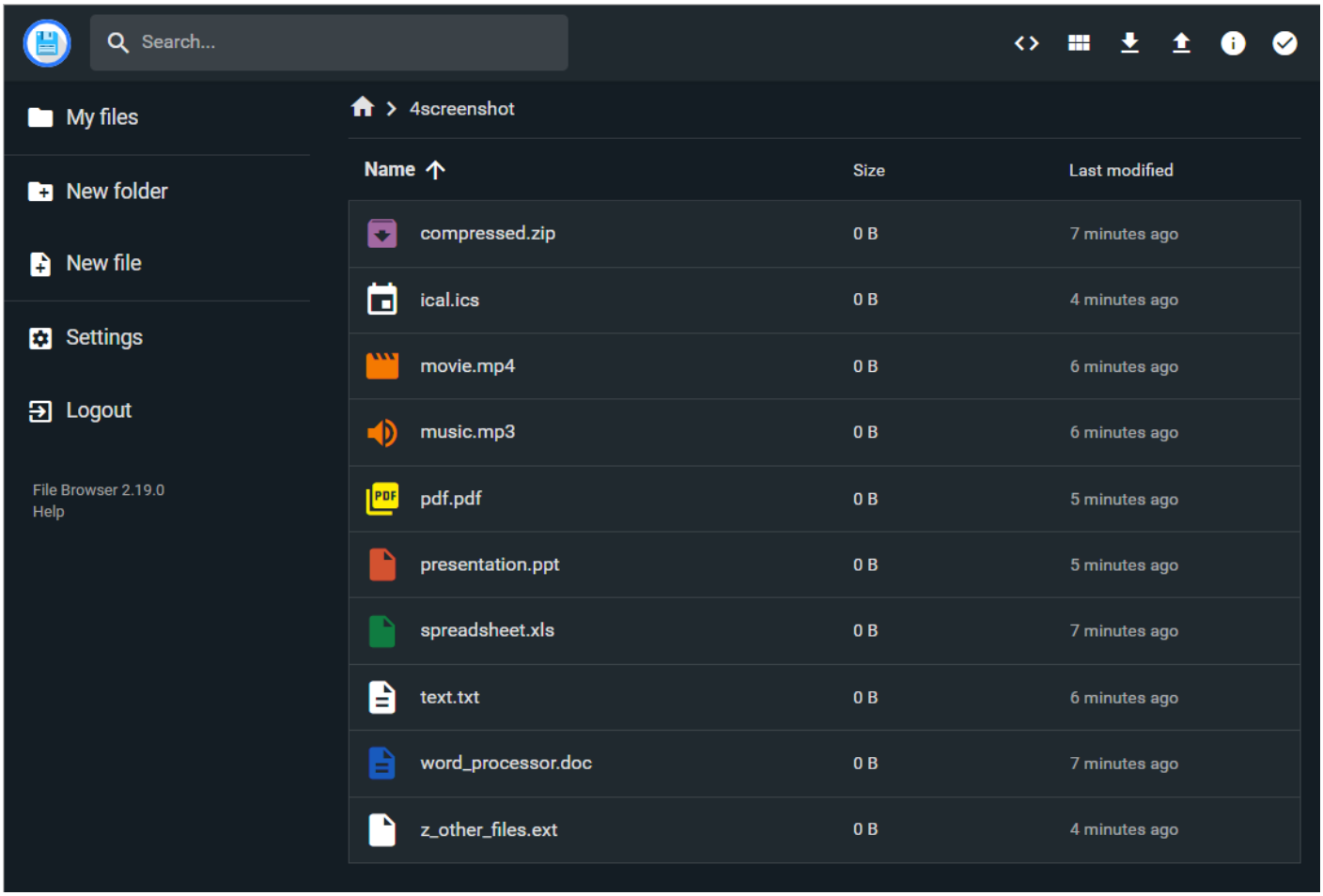
# Self-Hosted Files

- Filebrowser Custom Icon Colors - How To
- Comprehensive Guide: Setting up Tailscale for Self-Hosting
- Comprehensive Guide: Setting up DynDNS Updater for Cloudflare
- Understanding Prometheus: A Time Series Database for Monitoring
- The World of Self-Hosting: Exploring Alternative Options to Nextcloud
- Choosing the Right Media Server: Plex vs. Jellyfin
- Radarr/Sonarr Permission issues on Windows
- My Journey into Self-Hosting: A Raspberry Pi Tale

# Filebrowser Custom Icon Colors - How To

This css *colorizes* some file types' icons, so different files can be distinguished more easily. It also *changes some icons* to make distinguishing them even more easy.




# How to use this

- Download the custom.css file from here or copy from below.

```
/*
    filebrowser-css - a custom stylesheet for filebrowser
    that adds colors and changes some icons
```

```
    https://github.com/jniggemann/filebrowser-css

    Copyright (C) 2021-2022 Jan Niggemann
        With kind contributions of
        * Richard Asplin (https://github.com/richneptune)

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU Affero General Public License as
    published by the Free Software Foundation, either version 3 of the
    License, or (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU Affero General Public License for more details.

    You should have received a copy of the GNU Affero General Public License
    along with this program.  If not, see <https://www.gnu.org/licenses/>.

*/


/* remove a bit of padding from the file list entries */
#listing.list .item {
    padding: 0.7em;
}

/* make folders yellow-ish */
#listing .item[data-dir=true] div i {
    color: #ffc84b;
}

/* packed files - colorize and change icon */
#listing .item[aria-label$=".7z"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".7z"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".7z"] .material-icons::before {
```

```
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".arj"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".arj"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".arj"] .material-icons::before {
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".zip"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".zip"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".zip"] .material-icons::before {
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".gz"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".gz"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".gz"] .material-icons::before {
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".tar"] div i {
    color: #a268a1;
}
```

```
#listing .item[aria-label$=".tar"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".tar"] .material-icons::before {
    content: "archive";
    visibility: visible;
}



#listing .item[aria-label$=".bz"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".bz"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".bz"] .material-icons::before {
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".bz2"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".bz2"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".bz2"] .material-icons::before {
    content: "archive";
    visibility: visible;
}

#listing .item[aria-label$=".xz"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".xz"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".xz"] .material-icons::before {
    content: "archive";
    visibility: visible;
```

```
}

#listing .item[aria-label$=".tbz"] div i {
    color: #a268a1;
}
#listing .item[aria-label$=".tbz"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".tbz"] .material-icons::before {
    content: "archive";
    visibility: visible;
}



/* office files */
/* PDF - colorize and change icon /*
/* Note: This is yellow because I use SumatraPDF */
#listing .item[aria-label$=".pdf"] div i {
    color: #FFEE00;
}
#listing .item[aria-label$=".pdf"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".pdf"] .material-icons::before {
    content: "picture_as_pdf";
    visibility: visible;
}

/* word processors - colorize and change icon */
/* Word */
#listing .item[aria-label$=".doc"] div i {
    color: #185ABD;
}
#listing .item[aria-label$=".doc"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".doc"] .material-icons::before {
    content: "description";
    visibility: visible;
}
```

```
#listing .item[aria-label$=".docx"] div i {
    color: #185ABD;
}
#listing .item[aria-label$=".docx"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".docx"] .material-icons::before {
    content: "description";
    visibility: visible;
}

/* OpenOffice Writer */
#listing .item[aria-label$=".odt"] div i {
    color: #185ABD;
}
#listing .item[aria-label$=".odt"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".odt"] .material-icons::before {
    content: "description";
    visibility: visible;
}

/* LibreOffice Writer */
#listing .item[aria-label$=".sxw"] div i {
    color: #185ABD;
}
#listing .item[aria-label$=".sxw"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".sxw"] .material-icons::before {
    content: "description";
    visibility: visible;
}



/* PowerPoint */
#listing .item[aria-label$=".ppt"] div i {
    color: #D35230;
```

```
}
#listing .item[aria-label$=".pptx"] div i {
    color: #D35230;
}
#listing .item[aria-label$=".pps"] div i {
    color: #D35230;
}
/* OpenOffice Impress */
#listing .item[aria-label$=".odp"] div i {
    color: #D35230;
}


/* Excel */
#listing .item[aria-label$=".xls"] div i {
    color: #107C41;
}
#listing .item[aria-label$=".xlsx"] div i {
    color: #107C41;
}
#listing .item[aria-label$=".ods"] div i {
    color: #107C41;
}
#listing .item[aria-label$=".sxc"] div i {
    color: #107C41;
}
#listing .item[aria-label$=".wri"] div i {
    color: #336eff;
}


/* sound files - colorize */
#listing .item[data-type=audio] div i {
    color: #F47900;
}

/* video files - colorize */
#listing .item[data-type=video] div i {
    color: #F47900;
}
```

```
/* text files - change icon*/
#listing .item[aria-label$=".txt"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".txt"] .material-icons::before {
    content: "description";
    visibility: visible;
}
#listing .item[aria-label$=".md"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".md"] .material-icons::before {
    content: "description";
    visibility: visible;
}

/* various other files*/
/* iCal - change icon*/
#listing .item[aria-label$=".ics"] .material-icons {
    visibility: hidden;
}
#listing .item[aria-label$=".ics"] .material-icons::before {
    content: "event";
    visibility: visible;
}
```

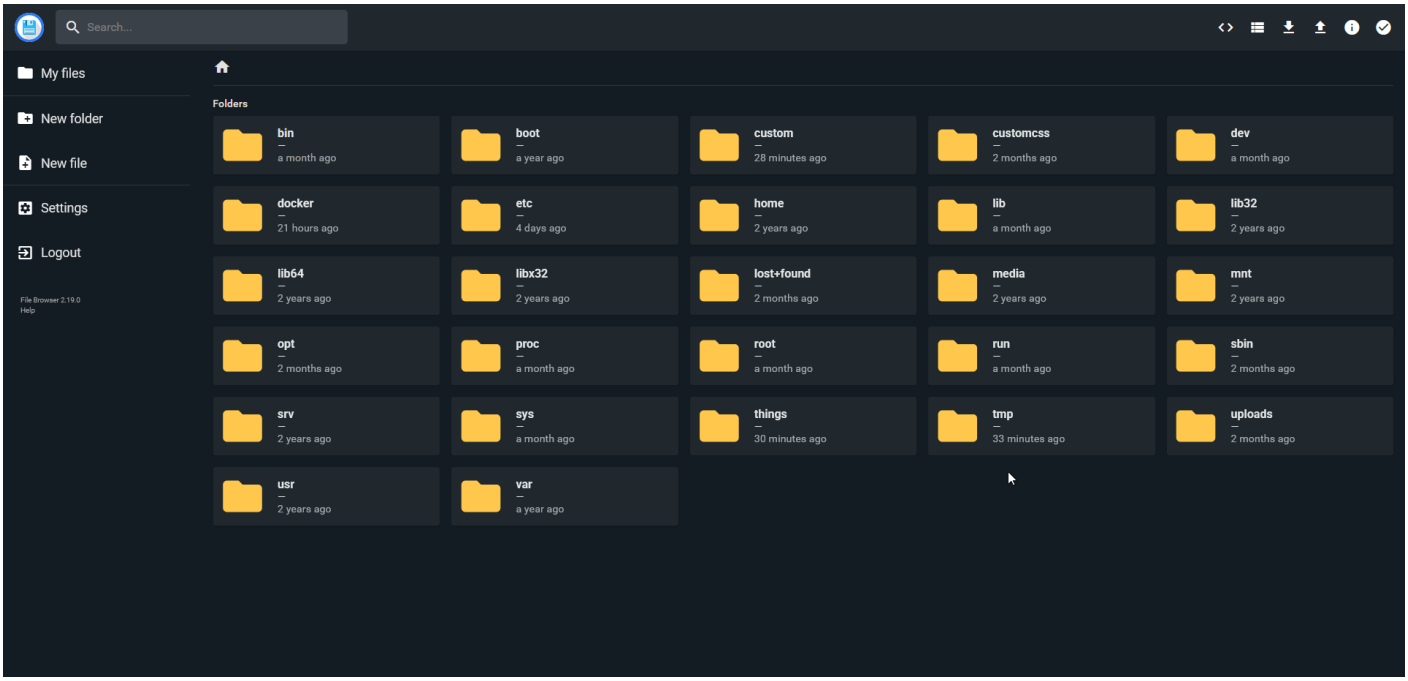
- In Filebrowser, go to Settings → Global Settings and enter the path to where you saved the custom.css in "Branding Directory path"

You can also set this on the commandline.

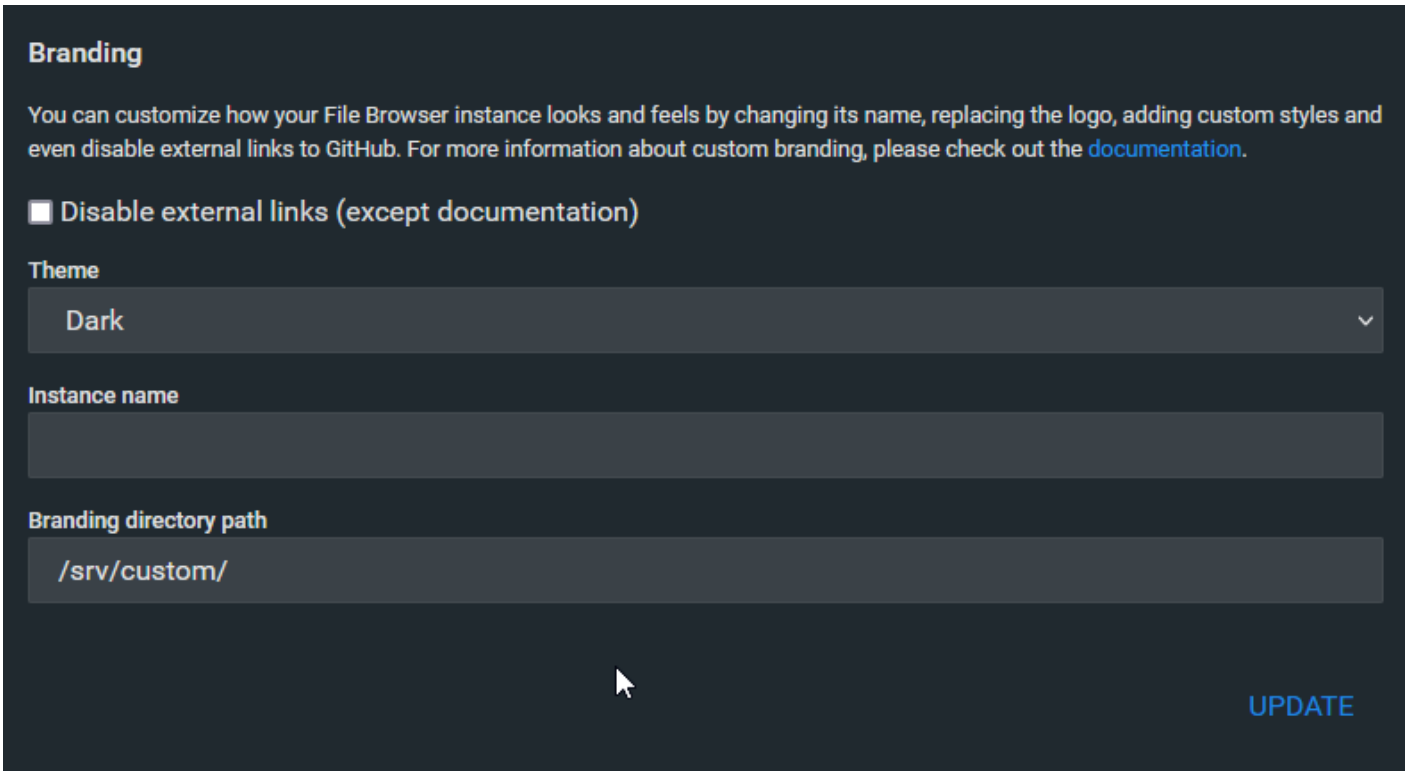
Since no one else showed exactly how to do this, I will be kind and explain it for you. Since I am using a docker image of Filebrowser, it uses the 'srv' directory as the volume mount for the file directory.

```
docker run \
    -v /path/to/root:/srv \
    -v /path/filebrowser.db:/database.db \
    -v /path/.filebrowser.json:/.filebrowser.json \
    -u $(id -u):$(id -g) \
    -p 8080:80 \
    filebrowser/filebrowser
```

I created a new folder in my root Filebrowser called custom then placed the custom.ss file there.




It's important to leave the trailing slash so the path can call and append the custom.css file.




Once you save this path, you might need to clear your browser cache to see the changes.

# Comprehensive Guide: Setting up Tailscale for Self-Hosting

**Step 1: Install Tailscale Server**

- Download the Tailscale server binary for your platform: https://tailscale.com/downloads
- Run the server with the following command: `./tailscaled --server-addr 0.0.0.0:8345`
- This starts the Tailscale server on port 8345 and listens on all interfaces.

**Step 2: Obtain Public and Private Keys**

- Generate a Tailscale network key pair: `./tailscaled --network-key-gen`
- Store the **public key** in a secure location for clients to connect.
- Keep the **private key** confidential.

**Step 3: Configure Clients**

- Download the Tailscale client binary for each device: https://tailscale.com/downloads
- Run the client with the following command: `./tailscaled --node-addr <public_ip>:8345 --network-key <network_key_public>`
- Replace `<public_ip>` with your Tailscale server's IP address and `<network_key_public>` with the public key obtained in Step 2.

**Step 4: Join the Network**

- Once clients are configured, they will join the Tailscale network and be able to communicate with each other over any network.

**Step 5: Advanced Configuration (Optional)**

- **Custom Network Name:** Use `--network-name <network_name>` to assign a specific name to your Tailscale network.
- **Remote Access:** Enable remote access by running `./tailscaled --remote-access` on the server. This allows non-Tailscale devices to connect to the network using a VPN.
- **Authentication:** Implement authentication using `./tailscaled --auth-file <path/to/auth_file>` on the server. This requires creating an authentication file containing user credentials.

**Tips:**

- Use a static IP address for your Tailscale server to avoid connectivity issues.
- Configure firewall rules to allow traffic on port 8345 for Tailscale communication.
- Consider using a dedicated machine or container for the Tailscale server for better performance and security.

**Further Resources:**

- Tailscale Documentation: https://tailscale.com/docs/
- Tailscale Community: https://tailscale.com/community

**Note:** This guide provides a basic overview of the process. For more advanced configurations and specific scenarios, refer to the official Tailscale documentation and resources.

# Comprehensive Guide: Setting up DynDNS Updater for Cloudflare

**Step 1: Choose a Dynamic DNS Provider**

- DynDNS offers a free dynamic DNS service that automatically updates your domain's IP address when it changes.
- Sign up for a DynDNS account and obtain your domain name and authentication credentials.

**Step 2: Install and Configure DynDNS Updater**

- Download the DynDNS Updater binary for your platform: https://dynupdate.com/downloads/
- Run the updater with the following command: `./dynupdate -u <username> -p <password> -h <host>`
- Replace `<username>`, `<password>`, and `<host>` with your DynDNS credentials and domain name.

**Step 3: Configure Cloudflare DNS Records**

- Log in to your Cloudflare account and manage your domain's DNS records.
- Create two TXT records:
  - One record with the name `_dynupdate` and value `[DynDNS token]`
  - One record with the name `<your domain name>` and value `[ Cloudflare IP]`

**Step 4: Obtain Dynamic IP Address**

- Use a script or external service to retrieve your current IP address.
- This can be done using tools like `ipinfo.io` or `curl ipinfo.io/ip`.

**Step 5: Update DynDNS Record**

- Write a script that runs the DynDNS updater every time the IP address changes.
- The script should:
  - Retrieve the current IP address.
  - Update the `_dynupdate` TXT record with the new token.

- Update the `<your domain name>` A record with the new IP address.

**Step 6: Automate the Process**

- Use a cronjob or systemd service to automatically run the update script at regular intervals.
- This ensures that your DNS records are always up-to-date and point to the correct IP address.

**Tips:**

- Choose a reliable script or service for IP address retrieval to minimize downtime.
- Update the frequency of updates based on your network's stability and requirements.
- Consider using a dedicated machine or container for DynDNS Updater to ensure consistent performance.

**Further Resources:**

- DynDNS Documentation: https://dynupdate.com/docs/
- Cloudflare DNS Documentation: https://developers.cloudflare.com/dns/

**Note:** This guide provides a basic overview of the process. For more advanced configurations and specific scenarios, refer to the official DynDNS and Cloudflare documentation.

# Understanding Prometheus: A Time Series Database for Monitoring

Monitoring and logging are crucial components in ensuring the smooth functioning of applications and systems. One such tool that has gained immense popularity in recent years is Prometheus, a time series database designed specifically for monitoring. In this article, we will delve into what Prometheus is, how it works, and its implementation with Grafana.

**What is Prometheus?**

Prometheus is an open-source time series database that was originally developed by SoundCloud. It is designed to collect metrics from applications and systems, storing them in a time series format for later querying and analysis. The primary goal of Prometheus is to provide a scalable and flexible solution for monitoring and logging, allowing users to track the performance and behavior of their applications over time.

**How Does Prometheus Work?**

Prometheus operates on the concept of scraping metrics from targets (applications or systems) at regular intervals. These targets are configured using YAML files, which define the scrape interval, job name, and other relevant details. The Prometheus server then scrapes these targets, collecting metrics such as CPU usage, memory consumption, and network traffic.

Once collected, these metrics are stored in a time series format within the Prometheus database. This allows for efficient querying and analysis of historical data. Prometheus also includes support for alerting and notification features, enabling users to set thresholds for specific metrics and receive alerts when those thresholds are exceeded.

**Implementation with Grafana**

Grafana is an open-source platform that provides visualization tools for monitoring and logging data. By combining Prometheus with Grafana, users can create a comprehensive monitoring solution that offers real-time insights into their applications and systems.

To implement Prometheus with Grafana, follow these steps:

1. **Install Prometheus:** Download and install the Prometheus server on your chosen platform.
2. **Configure Targets:** Define targets for Prometheus to scrape metrics from using YAML files.
3. **Create a Data Source in Grafana:** In Grafana, create a new data source by selecting "Prometheus" as the type. Enter the URL of your Prometheus server and configure any additional settings as needed.
4. **Visualize Metrics:** Use Grafana's dashboard features to visualize Prometheus metrics, creating panels for CPU usage, memory consumption, network traffic, or other relevant metrics.

**Benefits of Using Prometheus with Grafana**

The combination of Prometheus and Grafana offers numerous benefits, including:

- Real-time monitoring: Prometheus provides real-time data collection, allowing users to track performance and behavior in near-real-time.
- Scalability: Prometheus is designed for scalability, enabling it to handle large volumes of data and high traffic loads.
- Flexibility: With Prometheus and Grafana, users can create customized dashboards and panels tailored to their specific monitoring needs.
- Alerting and notification: Prometheus' alerting features enable users to set thresholds and receive notifications when performance issues arise.

Prometheus is a powerful time series database designed specifically for monitoring applications and systems. By implementing Prometheus with Grafana, users can create comprehensive monitoring solutions that offer real-time insights into their infrastructure. Whether you're looking to monitor CPU usage, memory consumption, or network traffic, the combination of Prometheus and Grafana provides a scalable, flexible, and customizable solution for all your monitoring needs.

# The World of Self-Hosting: Exploring Alternative Options to Nextcloud

As the popularity of cloud storage services continues to grow, many users are looking for alternative options that allow them to maintain control over their data. One such option is self-hosting a cloud storage solution, which provides a secure and private environment for storing and sharing files. In this article, we'll explore some Nextcloud alternatives that can be used for self-hosting, as well as provide guidance on how they can be utilized.

**1. ownCloud**

ownCloud is one of the most well-known open-source cloud storage solutions available. Developed by Frank Karlitschek and others, ownCloud allows users to create a private cloud storage solution that can be accessed from anywhere. With ownCloud, you can store and share files, collaborate with others, and even integrate third-party apps.

ownCloud is highly customizable, allowing you to tailor the installation to your specific needs. You can also extend its functionality by adding plugins and themes. One of the most significant advantages of ownCloud is its scalability, making it an excellent choice for large-scale deployments.

**2. Pydio**

Pydio (formerly known as AjaXplorer) is another popular open-source cloud storage solution that offers a high degree of customization. Developed by Alexandre Morin, Pydio provides a robust set of features for storing and sharing files, including collaboration tools and version control.

Pydio's architecture is designed to be highly scalable, making it an excellent choice for large-scale deployments. It also supports multiple protocols, such as FTP, SFTP, and WebDAV, allowing you to integrate it with other storage solutions.

**3. Seafile**

Seafile is a cloud storage solution that focuses on collaboration and file sharing. Developed by Seafile Ltd., this open-source platform provides a high degree of customization and scalability.

One of the key features of Seafile is its ability to handle large files and folders, making it an excellent choice for teams working with multimedia content. It also includes advanced version control and collaboration tools, allowing multiple users to work together on projects.

**4. FileRun**

FileRun is a modern, open-source cloud storage solution that focuses on simplicity and ease of use. Developed by FileRun Inc., this platform provides a user-friendly interface for storing and sharing files, as well as collaborating with others.

One of the unique features of FileRun is its focus on mobile app support. The platform includes native apps for Android and iOS devices, allowing users to access their files from anywhere. It also includes advanced security features, such as end-to-end encryption and two-factor authentication.

**5. Yandex Disk**

Yandex Disk is a cloud storage solution developed by Yandex, one of the largest technology companies in Russia. This platform provides a free, open-source alternative to commercial cloud storage solutions.

One of the key features of Yandex Disk is its focus on simplicity and ease of use. The platform includes a user-friendly interface for storing and sharing files, as well as collaborating with others. It also supports advanced file management features, such as version control and folder organization.

**Guidance on Using Nextcloud Alternatives**

When selecting a Nextcloud alternative for self-hosting, it's essential to consider the specific needs of your use case. Here are some general guidelines to keep in mind:

1. **Scalability**: If you anticipate storing large amounts of data or expect a high volume of users, look for platforms that prioritize scalability.
2. **Customization**: If you need to tailor the platform to your specific requirements, consider options with a high degree of customization.
3. **Security**: If security is a top concern, look for platforms that prioritize encryption and two-factor authentication.
4. **Collaboration**: If collaboration is a key aspect of your use case, consider platforms that include advanced version control and sharing features.

By considering these factors and selecting the right Nextcloud alternative for self-hosting, you can create a private cloud storage solution that meets your specific needs and provides a secure environment for storing and sharing files.

# Choosing the Right Media Server: Plex vs. Jellyfin

When it comes to hosting a media server, two popular options are Plex and Jellyfin. Both platforms offer unique features and advantages that can help you create a personalized entertainment experience. In this article, we'll delve into the key differences between Plex and Jellyfin, helping you decide which one is best for your needs.

**Plex: The Comprehensive Media Server**

Plex has established itself as a leading media server solution with its extensive device support, advanced features, and cloud syncing capabilities. Here are some of the key pros and cons to consider:

Pros:

- Wide device support: Plex has apps for most major devices, including smart TVs, streaming boxes, and mobile devices.
- Large community: Plex has a massive user base, which means there are many resources available online (e.g., forums, documentation, and plugins).
- Advanced features: Plex offers advanced features like live TV streaming, DVR capabilities, and transcoding for incompatible formats.
- Cloud syncing: Plex allows you to sync your media library across multiple devices.

Cons:

- Paid subscription required: While the core Plex app is free, some features require a paid subscription (Plex Pass).
- Resource-intensive: Plex can be demanding on system resources, which might impact performance if your server isn't powerful enough.
- Limited free version: The free version of Plex has limited functionality and may not meet all your needs.

**Jellyfin: The Lightweight Media Server**

Jellyfin is a free and open-source media server solution that offers a more lightweight alternative to Plex. Here are some of the key pros and cons to consider:

Pros:

- Free and open-source: Jellyfin is completely free and open-source, with no paid subscriptions or limitations.
- Lightweight: Jellyfin is designed to be efficient and won't strain your system resources.
- Simple and easy to use: Jellyfin has a minimalistic interface that's easy to navigate, even for those without extensive technical expertise.

Cons:

- Limited device support: While Jellyfin supports many devices, its compatibility list is not as extensive as Plex'.
- Fewer advanced features: Jellyfin lacks some of the advanced features available in Plex, such as live TV streaming and DVR capabilities.
- No cloud syncing: Jellyfin doesn't offer cloud syncing or media library management across multiple devices.

**Choosing Between Plex and Jellyfin**

When deciding between Plex and Jellyfin, consider the following questions:

1. Do you prioritize having a wide range of device support? If so, Plex might be the better choice for you.
2. Are advanced features like live TV streaming and DVR capabilities important to you? If yes, Plex is likely the way to go.
3. Are you willing to pay for a subscription to unlock additional features in Plex? Or do you prefer the free and open-source nature of Jellyfin?

Ultimately, if you value a wide range of device support, advanced features, and cloud syncing, Plex might be the better choice for you. However, if you're looking for a lightweight, easy-to-use, and completely free media server solution, Jellyfin is an excellent option.

Both Plex and Jellyfin offer unique strengths and weaknesses that can help you create a personalized entertainment experience. By considering your needs and preferences, you can make an informed decision about which media server solution is best for you.

# Radarr/Sonarr Permission issues on Windows

If you are running Radarr or Sonarr on Windows but find you are getting permission issues, try this.

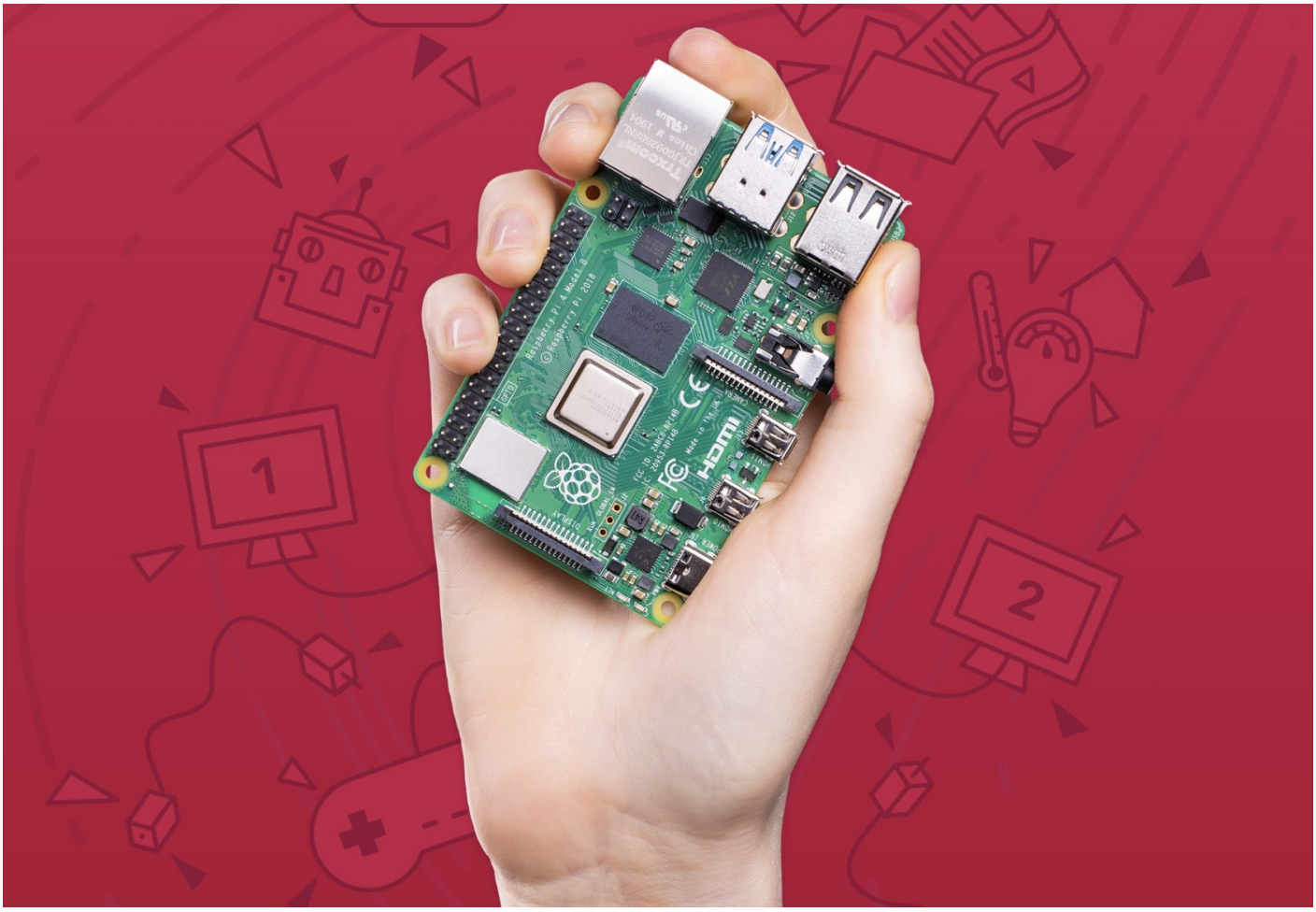
First, open up your search bar and type in "services" - this will take you to the Services window on your Windows machine. From there, navigate over to Radarr and right-click on it. In the context menu that pops up, select "Properties". This will bring up the Properties window for Radarr.

Next, click on the Log On tab at the top of the window. You should see a list of available users here - simply select the one you're currently logged in as on your Windows machine. Once you've done this, Radarr should now be able to use your current login credentials and you'll be all set! You may need to restart Radarr for permissions to take effect.

# My Journey into Self-Hosting: A Raspberry Pi Tale

Hey there fellow tech enthusiasts! Today, I want to share with you my own exciting journey into the world of self-hosting and how a little device called the Raspberry Pi played an essential role in it.

As someone who spends most of their time online, I've always been concerned about data privacy and the control I have over my digital life. So, when I discovered that I could set up my own self-hosted services using a Raspberry Pi, I was intrigued. This compact, yet powerful single-board computer offered the perfect opportunity to take matters into my own hands.



The learning process was challenging at times, but it was incredibly rewarding. With each hurdle overcome and each service successfully set up, I felt a sense of accomplishment that I had never experienced before. I found myself gaining a deeper understanding of how the digital world works, which in turn empowered me to customize my online experience to fit my needs perfectly.

One of the best parts about self-hosting with a Raspberry Pi is that it allowed me to enjoy improved data privacy. Gone were the days of relying on third-party services whose intentions may not have aligned with mine. With my own servers, I could ensure that my data stayed safe and secure, giving me peace of mind.

Another benefit I discovered during this journey was the ability to create tailored solutions for unique problems. When you use off-the-shelf services, you're often limited by their features and capabilities. But with self-hosting, I could design my own tools from scratch, ensuring they were a perfect fit for my requirements.

In closing, if you're looking to take control of your digital destiny, I wholeheartedly recommend diving into the world of self-hosting with a Raspberry Pi. It offers an unparalleled learning experience and empowers you to customize your online life to fit your exact needs. Plus, there's something incredibly satisfying about setting up and maintaining your own servers!