

# Webdav using Apache2

## Step 1: Install Required Packages

First, ensure that you have Apache2 and the necessary modules installed. You can install them using the package manager for your distribution. For example, on Ubuntu or Debian, you can run:

```
sudo apt install apache2 apache2-utils libapache2-mod-dav libapache2-mod-dav_fs
```

## Step 2: Enable Required Apache Modules

You need to enable the WebDAV modules in Apache. Run the following commands:

```
sudo a2enmod dav  
sudo a2enmod dav_fs  
sudo a2enmod auth_digest  
sudo a2enmod auth_basic  
sudo a2enmod authn_file
```

## Step 3: Create a Directory for WebDAV

Create a directory that will be used for WebDAV storage. For example:

```
sudo mkdir /var/www/webdav
```

Set the appropriate permissions for this directory:

```
sudo chown -R www-data:www-data /var/www/webdav
sudo chmod -R 755 /var/www/webdav
```

# Step 4: Configure Apache for WebDAV

Create a new configuration file for your WebDAV site. You can create a new file in the `/etc/apache2/sites-available/` directory. For example, create a file named `webdav.conf`:

```
sudo vim /etc/apache2/sites-available/webdav.conf
ln -s /etc/apache2/sites-available/webdav.conf /etc/apache2/sites-enabled
```

Add the following configuration to the file:

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    DocumentRoot /var/www/webdav
    ServerName example.com # Change to your domain or IP

    <Directory /var/www/webdav>
        Options Indexes FollowSymLinks
        AllowOverride None

        Dav On
        AuthType Basic
        AuthName "WebDAV Login"
        AuthUserFile /etc/apache2/webdav.password
        Require valid-user
    </Directory>
</VirtualHost>
```

# Step 5: Set Up Authentication (Optional)

If you want to secure your WebDAV with a username and password, you can create a password file. Use the `htpasswd` command to create a password file:

```
sudo apt install apache2-utils # If not already installed
sudo htpasswd -c /etc/apache2/webdav.password username
```

Replace `username` with your desired username. You will be prompted to enter a password.

# Step 6: Enable the Site Configuration

Enable the new site configuration and reload Apache:

```
sudo a2ensite webdav.conf
sudo a2dissite 000-default.conf
sudo systemctl reload apache2
```

# Step 7: Access Your WebDAV Server

You can now access your WebDAV server through the Nginx reverse proxy. Open a web browser and navigate to:

```
http://example.com
```

# Step 8: Troubleshooting

If you encounter issues, check the Apache error logs for more information or check your config integrity:

```
sudo tail -f /var/log/apache2/error.log
sudo apachectl configtest
```

## Step 9 (Optional): Configure Nginx as a Reverse Proxy

Create a New Nginx Configuration File: You can create a new configuration file for your WebDAV reverse proxy. For example, create a file named `webdav.conf` in the `/etc/nginx/sites-available/` directory:

```
server { # Change to 443 for HTTPS
    listen 80;
    server_name url.tld; # Replace with your domain or IP

    location / {
        proxy_pass http://ip-address:80; # Replace with your WebDAV server address
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        # WebDAV specific headers
        proxy_set_header Depth $http_depth;
        proxy_set_header Destination $http_destination;
        proxy_set_header If $http_if;
        proxy_set_header Lock-Token $http_lock_token;
        proxy_set_header Translate $http_translate;
        proxy_set_header If-None-Match $http_if_none_match;
        proxy_set_header If-Match $http_if_match;
        proxy_set_header If-Modified-Since $http_if_modified_since;
```

```
}  
}
```

---

Revision #1

Created 20 January 2025 04:16:49 by tinfoil-hat

Updated 20 January 2025 04:17:41 by tinfoil-hat