

How To Set Up WebDAV With Apache2 On Debian Etch

By Falko Timme

Published: 2008-06-01 17:41

How To Set Up WebDAV With Apache2 On Debian Etch

Version 1.0

Author: Falko Timme <ft [at] falkotimme [dot] com>

Last edited 05/27/2008

This guide explains how to set up WebDAV with Apache2 on a Debian Etch server. WebDAV stands for **Web-based Distributed Authoring and Versioning** and is a set of extensions to the HTTP protocol that allow users to directly edit files on the Apache server so that they do not need to be downloaded/uploaded via FTP. Of course, WebDAV can also be used to upload and download files.

I do not issue any guarantee that this will work for you!

1 Preliminary Note

I'm using a Debian Etch server with the IP address `192.168.0.100` here.

2 Installing WebDAV

If Apache is not already installed, install it as follows:

```
apt-get install apache2
```

Afterwards, enable the WebDAV modules:

```
a2enmod dav_fs
```

```
a2enmod dav
```

Reload Apache:

```
/etc/init.d/apache2 force-reload
```

3 Creating A Virtual Host

I will now create a default Apache vhost in the directory `/var/www/web1/web`. For this purpose, I will modify the default Apache vhost configuration in `/etc/apache2/sites-available/default`. If you already have a vhost for which you'd like to enable WebDAV, you must adjust this tutorial to your situation.

First, we create the directory `/var/www/web1/web` and make the Apache user (`www-data`) the owner of that directory:

```
mkdir -p /var/www/web1/web  
  
chown www-data /var/www/web1/web
```

Then we back up the default Apache vhost configuration (`/etc/apache2/sites-available/default`) and create our own one:

```
mv /etc/apache2/sites-available/default /etc/apache2/sites-available/default_orig  
  
vi /etc/apache2/sites-available/default
```

```
NameVirtualHost *  
<VirtualHost *>  
    ServerAdmin webmaster@localhost  
  
    DocumentRoot /var/www/web1/web/  
    <Directory /var/www/web1/web/>  
        Options Indexes MultiViews
```

```
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>

</VirtualHost>
```

Then reload Apache:

```
/etc/init.d/apache2 reload
```

4 Configure The Virtual Host For WebDAV

Now we create the WebDAV password file `/var/www/web1/passwd.dav` with the user `test` (the `-c` switch creates the file if it does not exist):

```
htpasswd -c /var/www/web1/passwd.dav test
```

You will be asked to type in a password for the user `test`.

We will later on use the URL `http://192.168.0.100/webdav` to connect to WebDAV. When you do this on a Windows XP client and type in the user name `test`, Windows translates this to `192.168.0.100test`. Therefore we create a second user account now (without the `-c` switch because the password file is already existing):

```
htpasswd /var/www/web1/passwd.dav 192.168.0.100\\test
```

[\(We must use a second backslash here in the user name to escape the first one!\)](#)

Now we change the permissions of the `/var/www/web1/passwd.dav` file so that only `root` and the members of the `www-data` group can access it:

```
chown root:www-data /var/www/web1/passwd.dav
```

```
chmod 640 /var/www/web1/passwd.dav
```

Now we modify our vhost in `/etc/apache2/sites-available/default` and add the following lines to it:

```
vi /etc/apache2/sites-available/default
```

```
[...]
Alias /webdav /var/www/web1/web

<Location /webdav>
  DAV On
  AuthType Basic
  AuthName "webdav"
  AuthUserFile /var/www/web1/passwd.dav
  Require valid-user
</Location>
[...]
```

The *Alias* directive makes (together with `<Location>`) that when you call `/webdav`, WebDAV is invoked, but you can still access the whole document root of the vhost. All other URLs of that vhost are still "normal" HTTP.

The final vhost should look like this:

```
NameVirtualHost *
<VirtualHost *>
  ServerAdmin webmaster@localhost
```

```
DocumentRoot /var/www/web1/web/  
<Directory /var/www/web1/web/>  
    Options Indexes MultiViews  
    AllowOverride None  
    Order allow,deny  
    allow from all  
</Directory>  
  
Alias /webdav /var/www/web1/web  
  
<Location /webdav>  
    DAV On  
    AuthType Basic  
    AuthName "webdav"  
    AuthUserFile /var/www/web1/passwd.dav  
    Require valid-user  
</Location>  
</VirtualHost>
```

Reload Apache afterwards:

```
/etc/init.d/apache2 reload
```

5 Testing WebDAV

We will now install *cadaver*, a command-line WebDAV client:

```
apt-get install cadaver
```

To test if WebDAV works, type:

```
cadaver http://localhost/webdav/
```

You should be prompted for a user name. Type in *test* and then the password for the user *test*. If all goes well, you should be granted access which means WebDAV is working ok. Type *quit* to leave the WebDAV shell:

```
server1:~# cadaver http://localhost/webdav/  
  Authentication required for test on server `localhost':  
  Username: test  
  Password:  
  dav:/webdav/> quit  
  Connection to `localhost' closed.  
server1:~#
```

6 Configure A Windows XP Client To Connect To The WebDAV Share

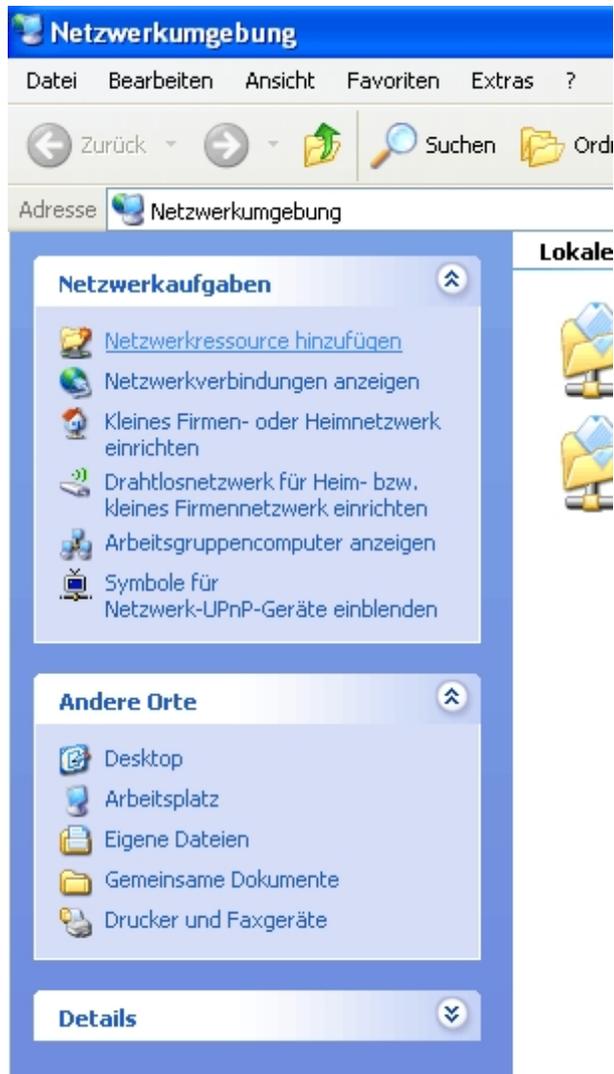
According to <http://www.heise.de/netze/WebDAV-mit-Apache--/artikel/77605/3> (in German!), Windows XP clients with SP2 don't support the *Basic* authentication in conjunction with WebDAV unless you download this file and tweak the Windows registry: <ftp://ftp.heise.de/pub/ct/listings/0504-202.zip>

The file contains a *.reg* file. Double-click on it to install it, then **restart Windows**.

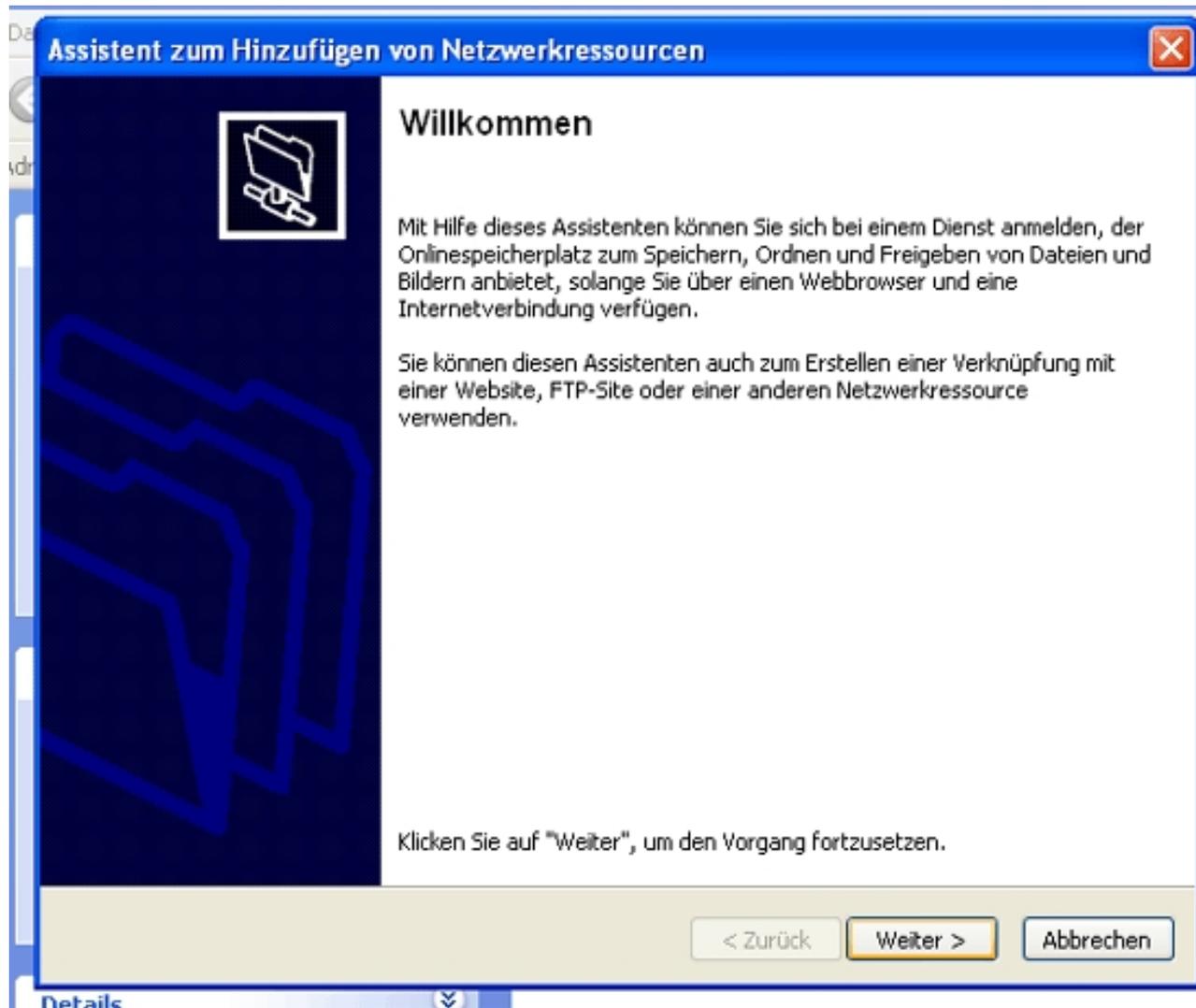
After the restart, click on *My Network Places* on your desktop (I have a German Windows, so the names are a bit different in the screenshots):



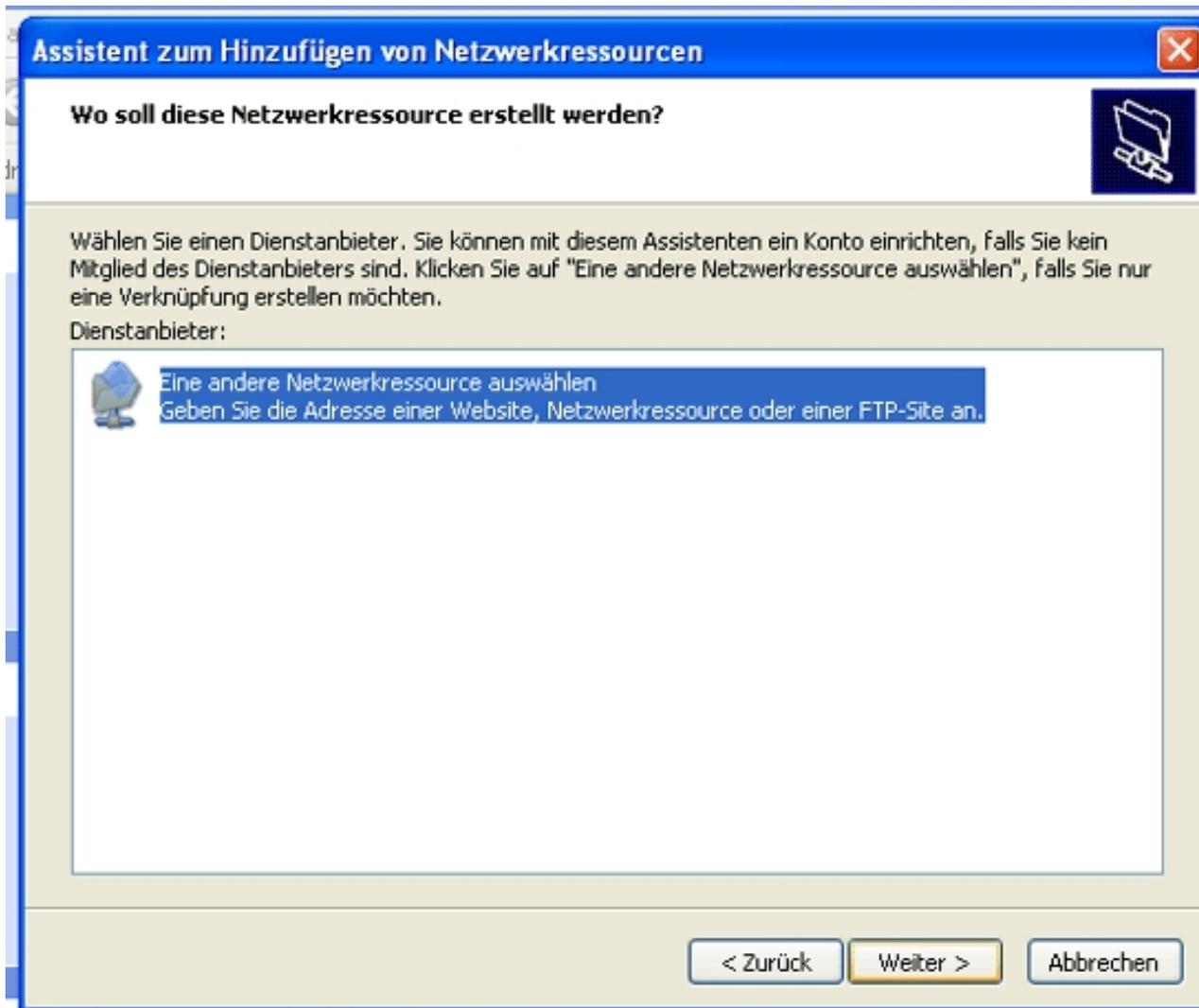
Select *Add a Network Place* from the *Network Tasks* menu (on the left):



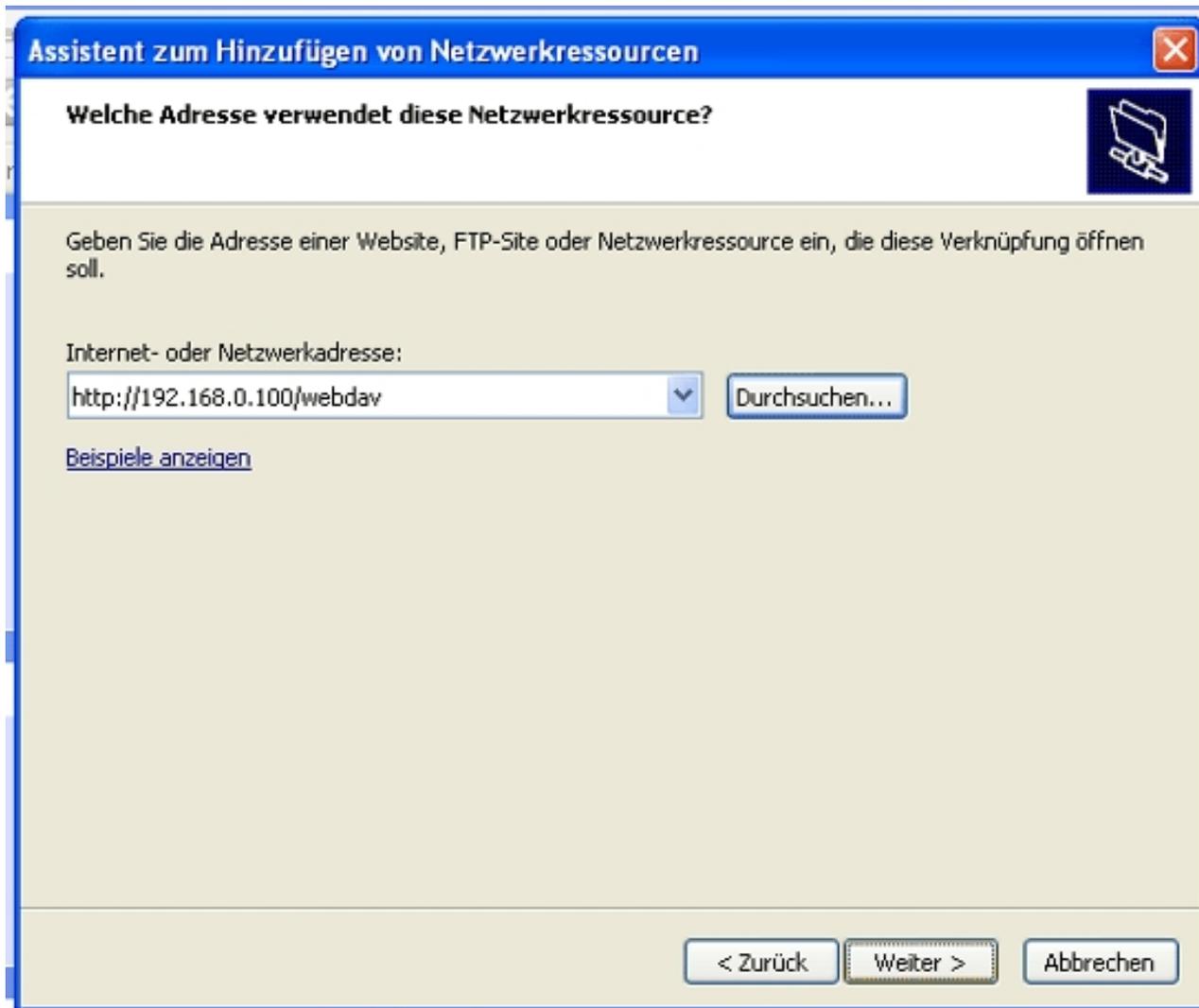
The *Add Network Place Wizard* comes up. Click on the *Next* button:



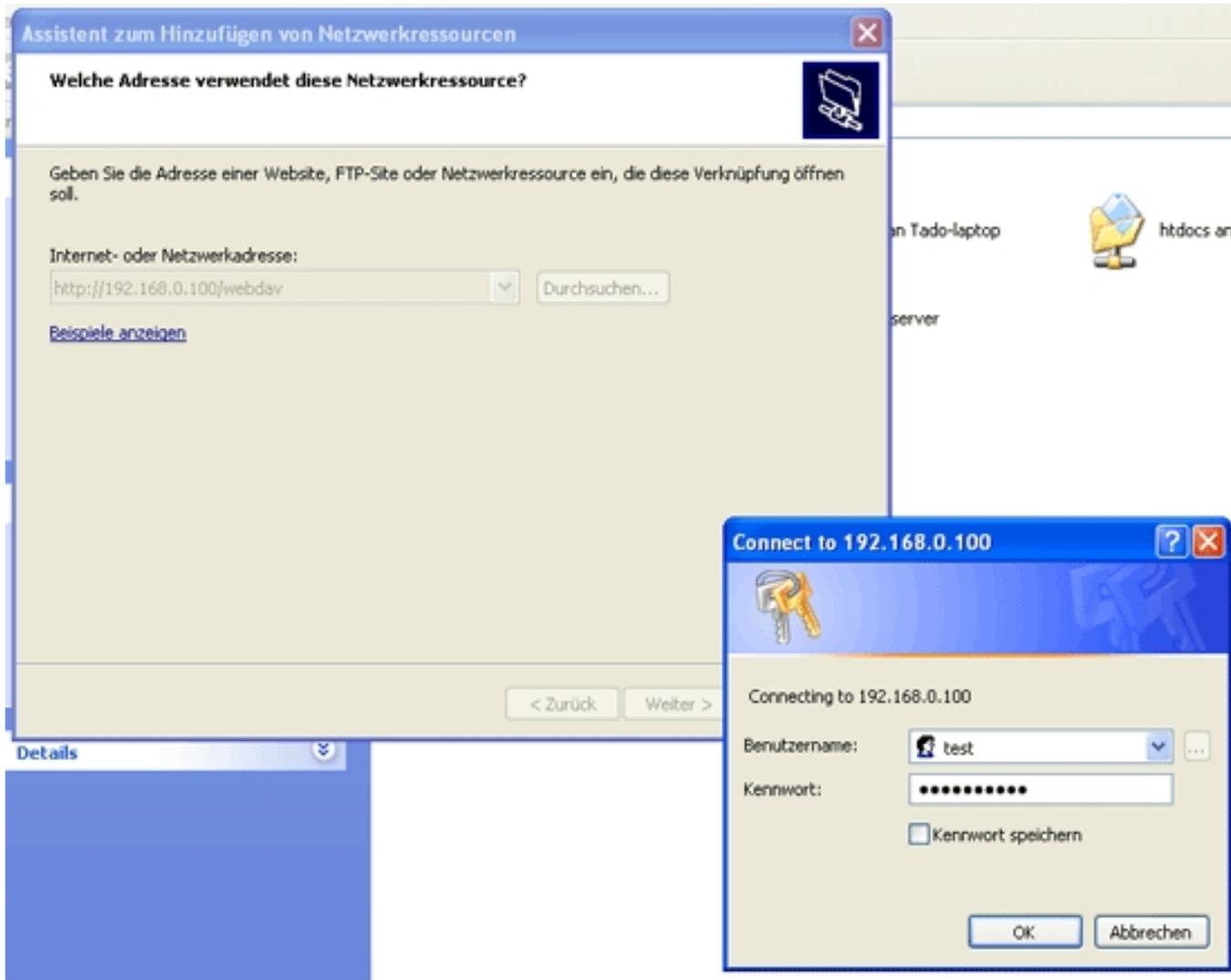
Select *Choose another network location*, and click on *Next*:



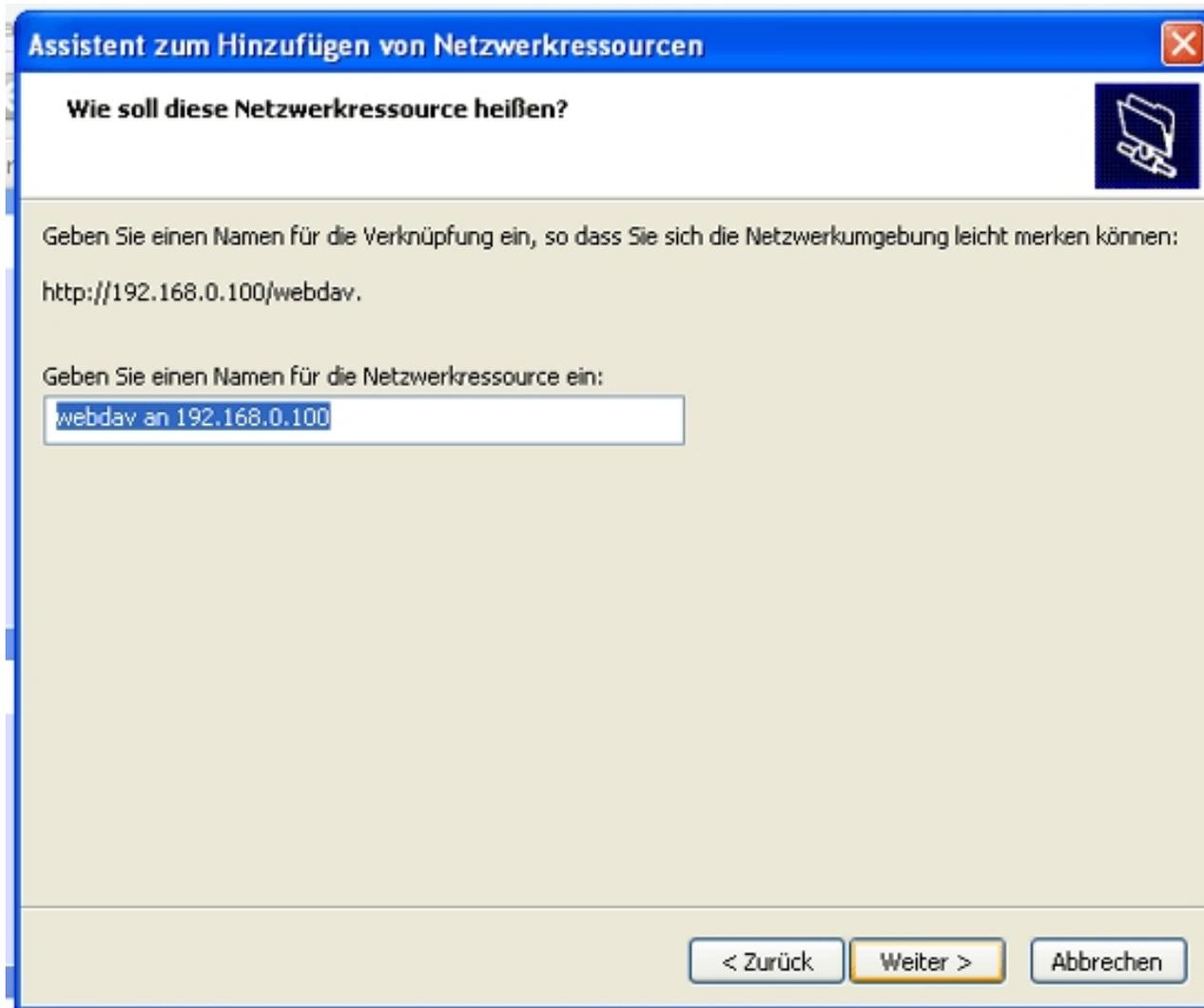
Enter `http://192.168.0.100/webdav` as the location and click on *Next*:



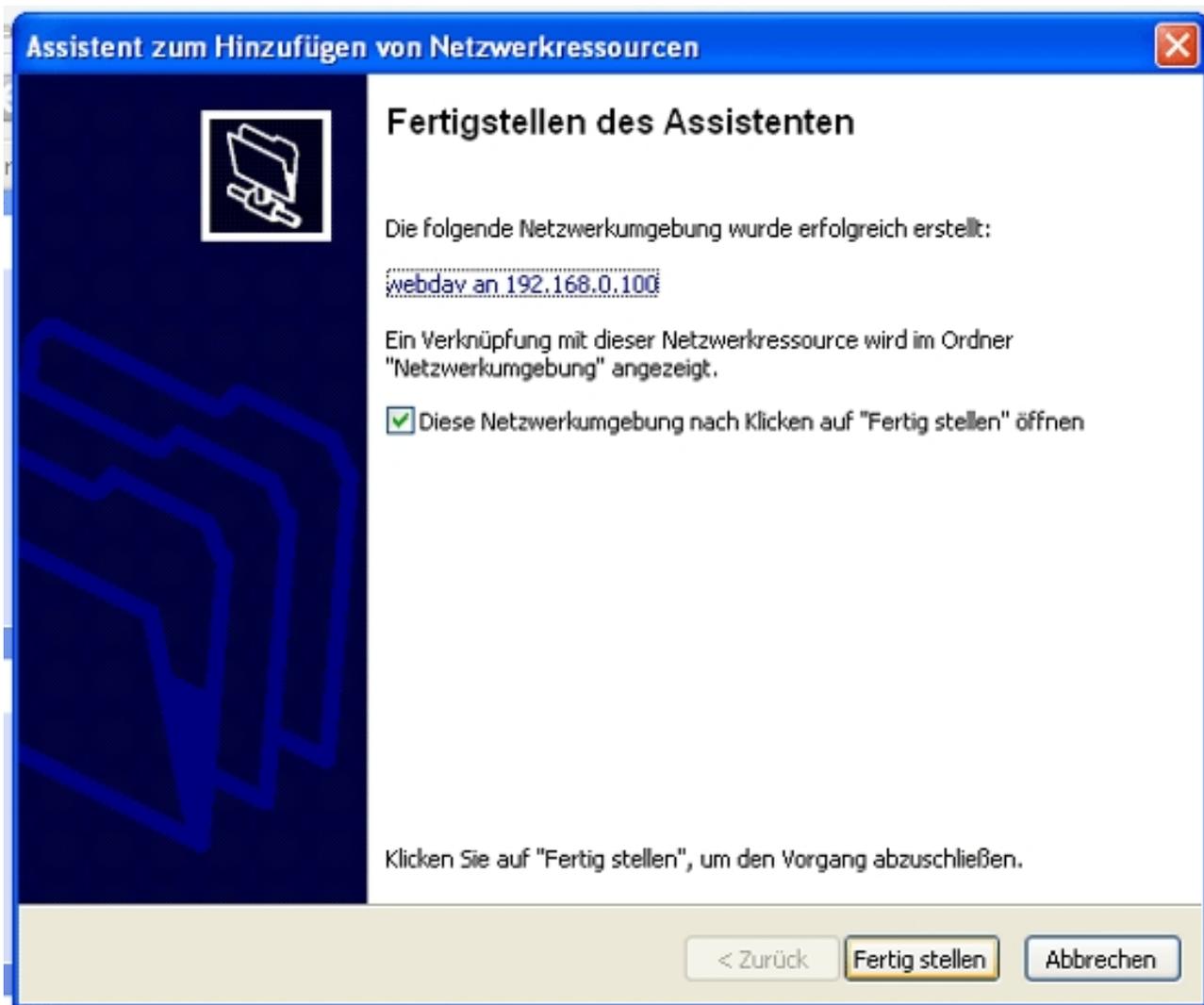
You will be prompted for a user name and a password. Type in the user name *test* and the password for the user *test*:



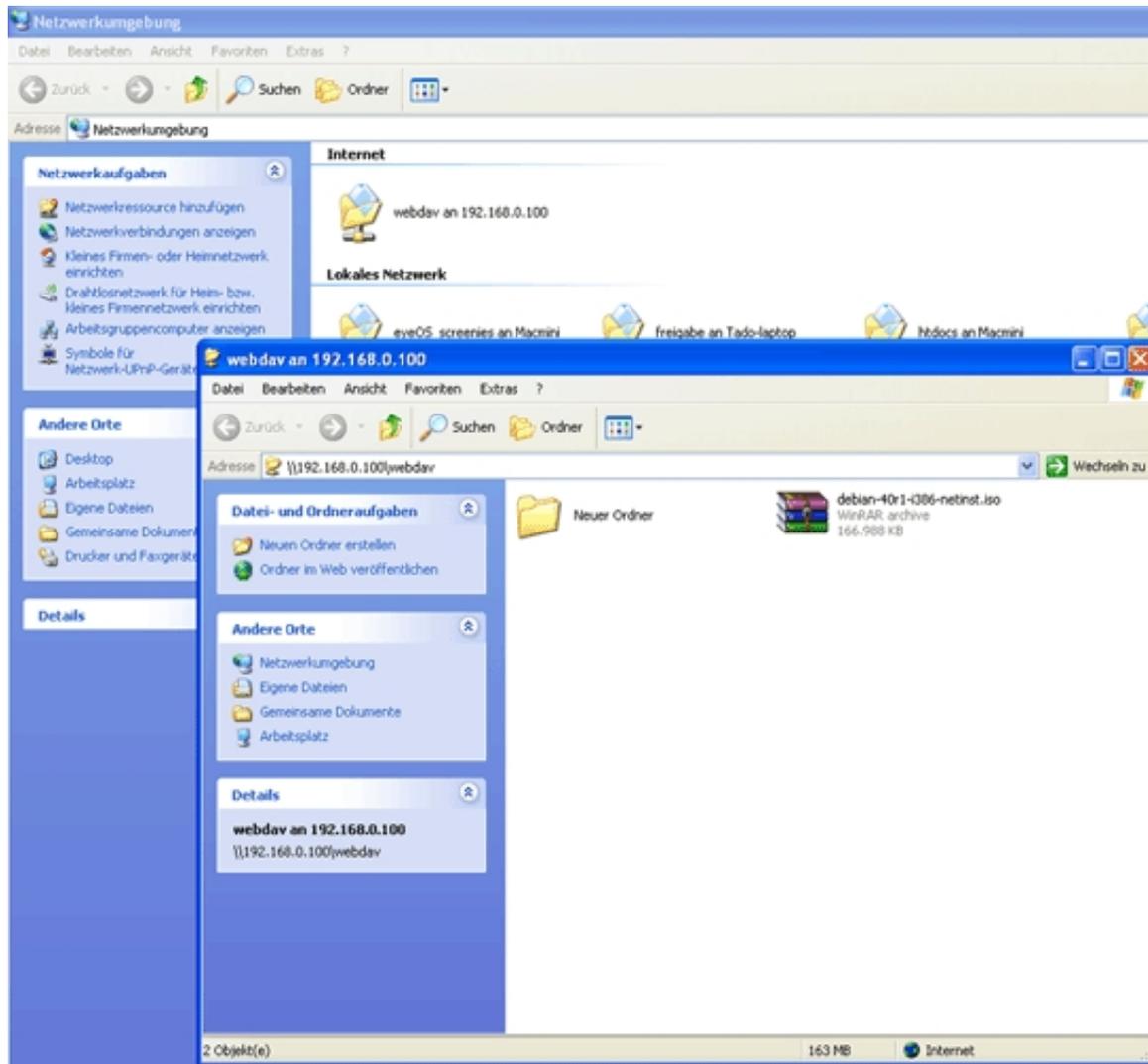
Then type in a name for the WebDAV folder:



To open the new connection, keep the *Open this network place when I click Finish* box checked, and click on *Finish*:



The WebDAV folder will then open where you can browse the contents of the `/var/www/web1/web` directory and its subdirectories on the server, and you will find an icon for your new WebDAV share in the *My Network Places* folder:



7 Configure A Linux Client (GNOME) To Connect To The WebDAV Share

If you want to connect to the WebDAV share from a GNOME desktop, go to *Places > Connect to Server...*:



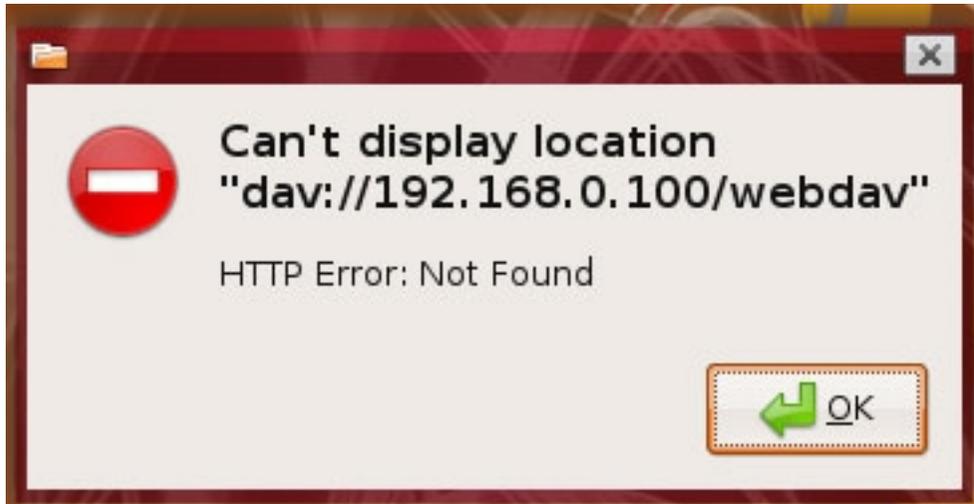
Select *WebDAV (HTTP)* as the *Service type*, type in the *Server (192.168.0.100 in this example)* and then the *Folder (webdav)*. Do not fill in a *User Name* yet because otherwise the connection will fail. Click on *Connect* afterwards:



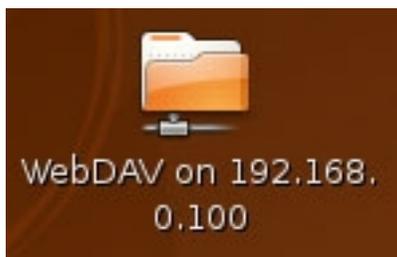
Now you are being prompted for a user name and password. Type in *test* along with the password, then click on *Connect*:



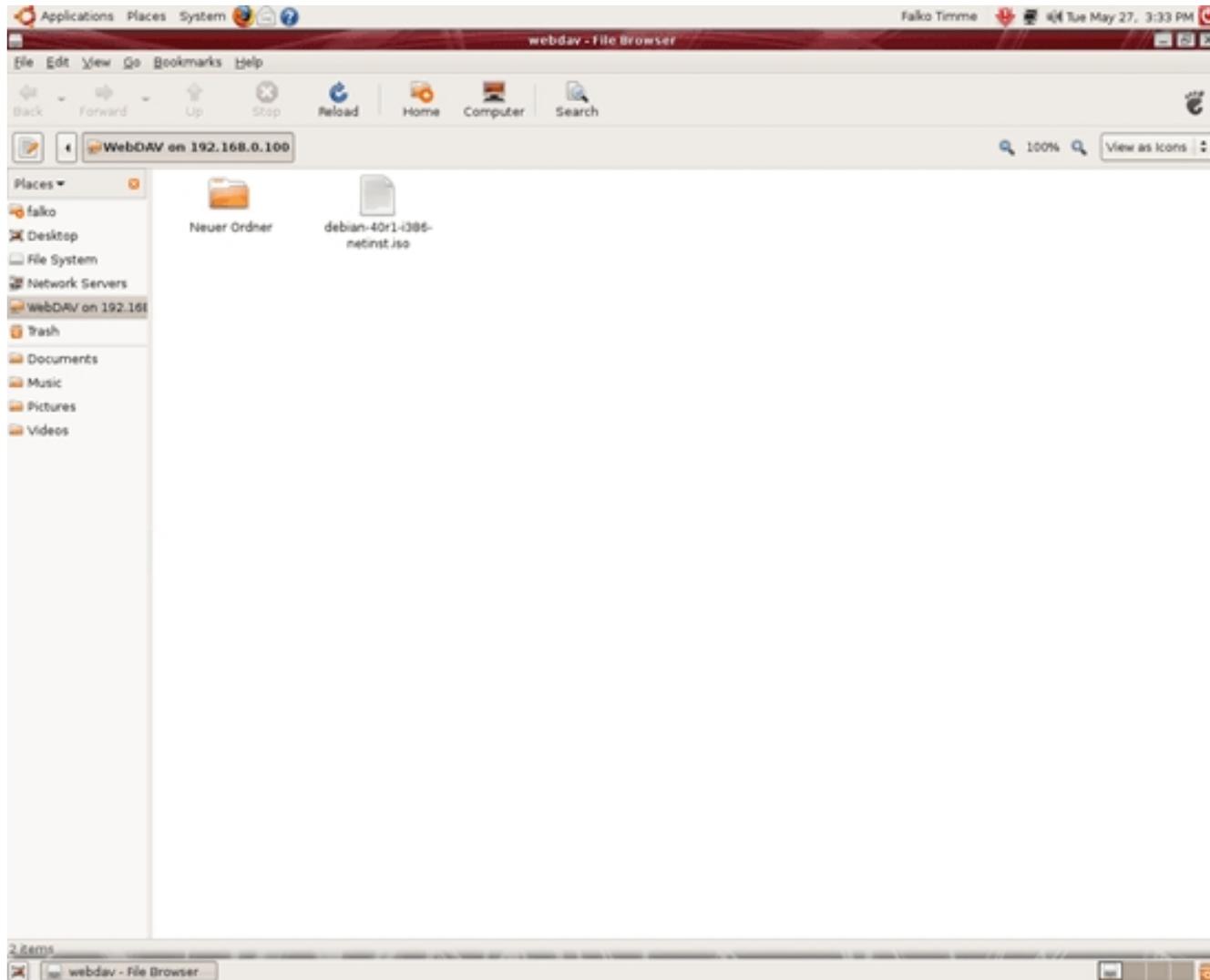
You might get the following error...



... but at the same time the WebDAV share should appear on the desktop, which means you can ignore the error:



Double-click on the icon to open the WebDAV share:



8 Links

- WebDAV: <http://www.webdav.org>

- Apache: <http://httpd.apache.org>
- Debian: <http://www.debian.org>