SFTP Server s0 v1

Overview

becke.ch

email: ssh--s0-v1@becke.ch

homepage: https://becke.ch/app/becke-ch--sftp-server--s0-v1/

download: https://play.google.com/store/apps/details?id=ch.becke.sftp_server__s0_v1

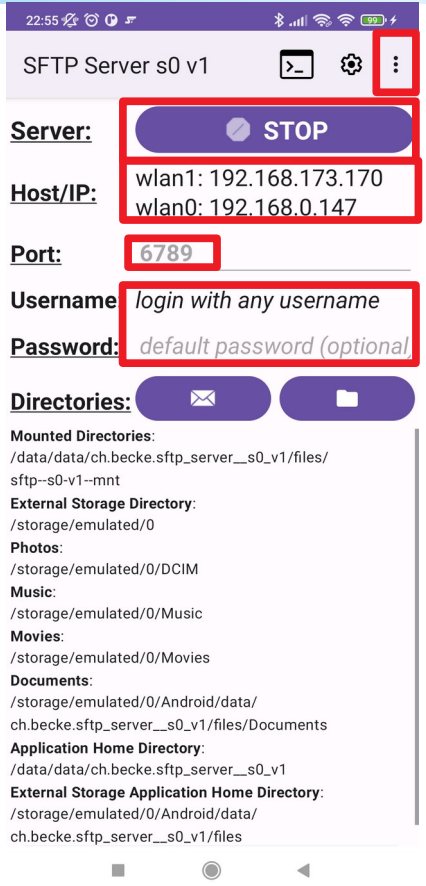
Summary

SFTP Server s0 v1 is an Android **Secure File Transfer** Protocol **Server App**, compatible with all Android devices down to Android Version 4.4/5.0* “KitKat/Lollipop*” (API Level 19/21*), covering **more than 99.9% of all android devices world-wide**.
*)Mounting of drives (document providers) & directories using Android Scoped Storage requires at least Android Version 5.0 “Lollipop” (API Level 21)

- SFTP Server s0 v1 offers **automatic public key handling** according to preference settings
- SFTP Server s0 v1 has been tested and is working well with different SFTP Client applications like: **SSHFS** (network file-system, mount for: Linux, Mac, Windows), **GIO/GVfs** (virtual file-system, mount for: Linux), **SFTP** (Linux-Client, Windows/Cygwin), **FileZilla** (Windows-, Mac-, Linux-Client), **WinSCP** (Windows-Client), **PSFTP** (**Putty SFTP**, Windows-Shell), **Cyberduck** (Windows- & Mac-Client), **Mountain Duck** (Windows- & Mac-Client), **Total Commander SFTP Plugin** (Windows-Client).
- SFTP Server s0 v1 is purely Java based (no 3rd party and native libraries) and therefore portable on different platforms and operating systems.

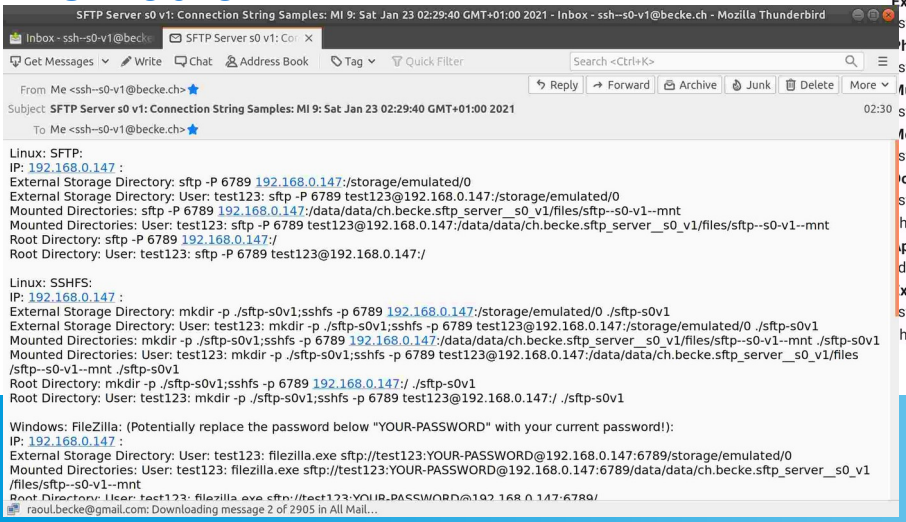
Getting Started

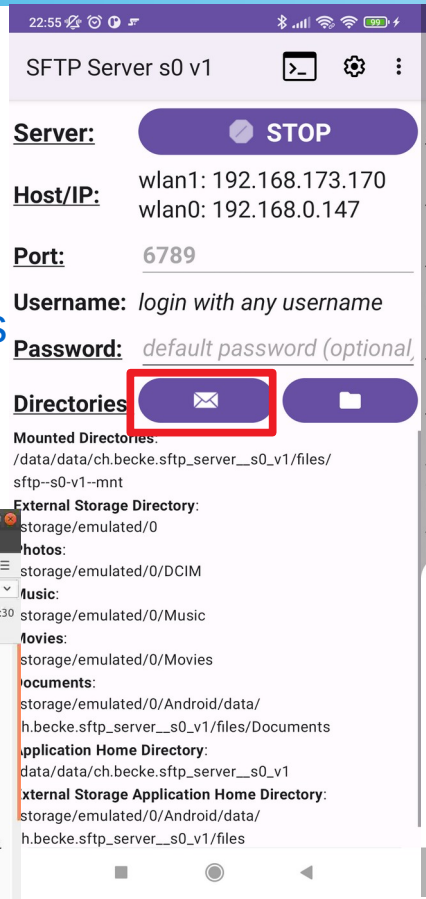
- **Server: STOP/START**: When the app launches the **SFTP server starts automatically**. Tapping the “**STOP**” button, stops the SFTP server and tapping the button, now called “**START**”, starts the SFTP server.
- **Host/IP & Port**: The SFTP server will listen for incoming connections on all listed IP Address(es) (e.g. “192.168.173.170” and “192.168.0.147”) on a certain Port (e.g. “6789”) shown on the main screen. The Port Value needs to be between 1 and 65535. Ports below 1024 require a rooted device (i.e. the ability to run the "su" command with root permissions). To **modify the port** the server needs to be “**STOPPED**” first!
  For **optimal speed** connect to **5GHz WLAN** and use a **fast SFTP Client** Program!
- **Username & Password**: You can login with any username. You can enter a default password or leave the the password field empty. If empty a random password will be generated and displayed on the screen during the first login. The passwords can be changed afterwards on the “SETTINGS” page (click the “Gear” symbol on the top). The minimal password length is 0 (empty) or at least 6 characters long.
- **Help and Overview-Video**: Clicking on the Help or Overview-Video entry (in the overflow menu, top right) brings up this presentation respective the corresponding video.
- **Context Help**: Clicking on the underlined text labels: “Server:”, “Host/IP:”, “Port:”, “Password:” and “Directories:” brings up a short context related help dialog.

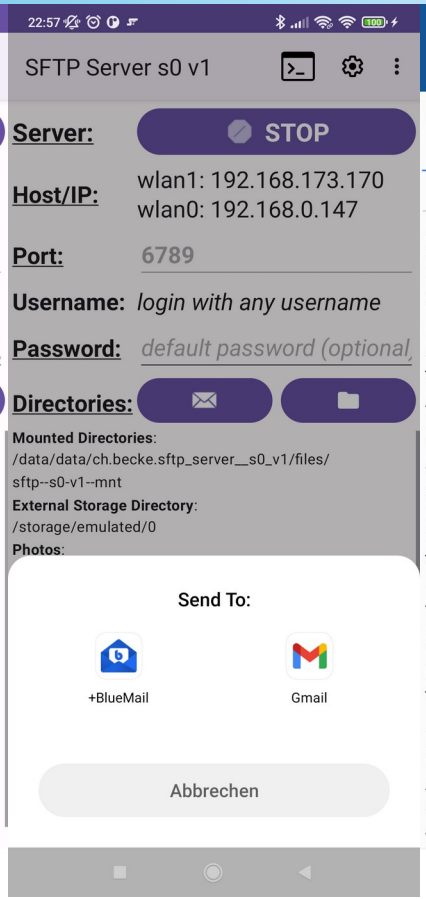


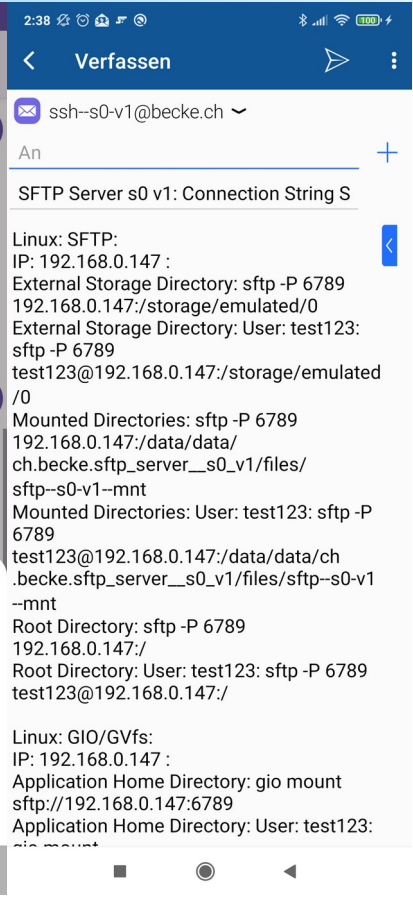
MAIL

MAIL: Pushing the “MAIL” button, launches the E-Mail App. The user can then choose where to send the e-mail containing the SFTP Server Connection Strings to.
With the help of these connection strings the user can then easily connect from his PC to the SFTP Server running on his mobile.





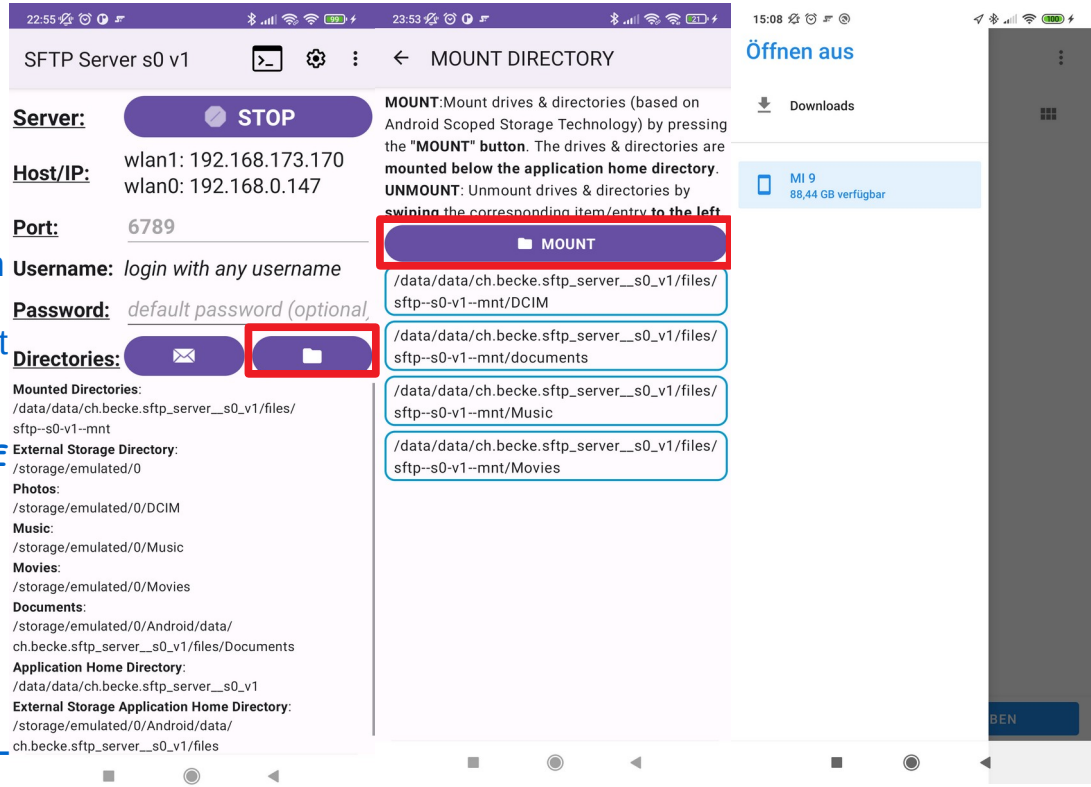




MOUNT

“MOUNT” (Folder Icon): Pushing the “MOUNT” (Folder Icon) button on the main screen, opens the “MOUNT DIRECTORY” activity screen, showing the mounted drives (document providers) and directories.

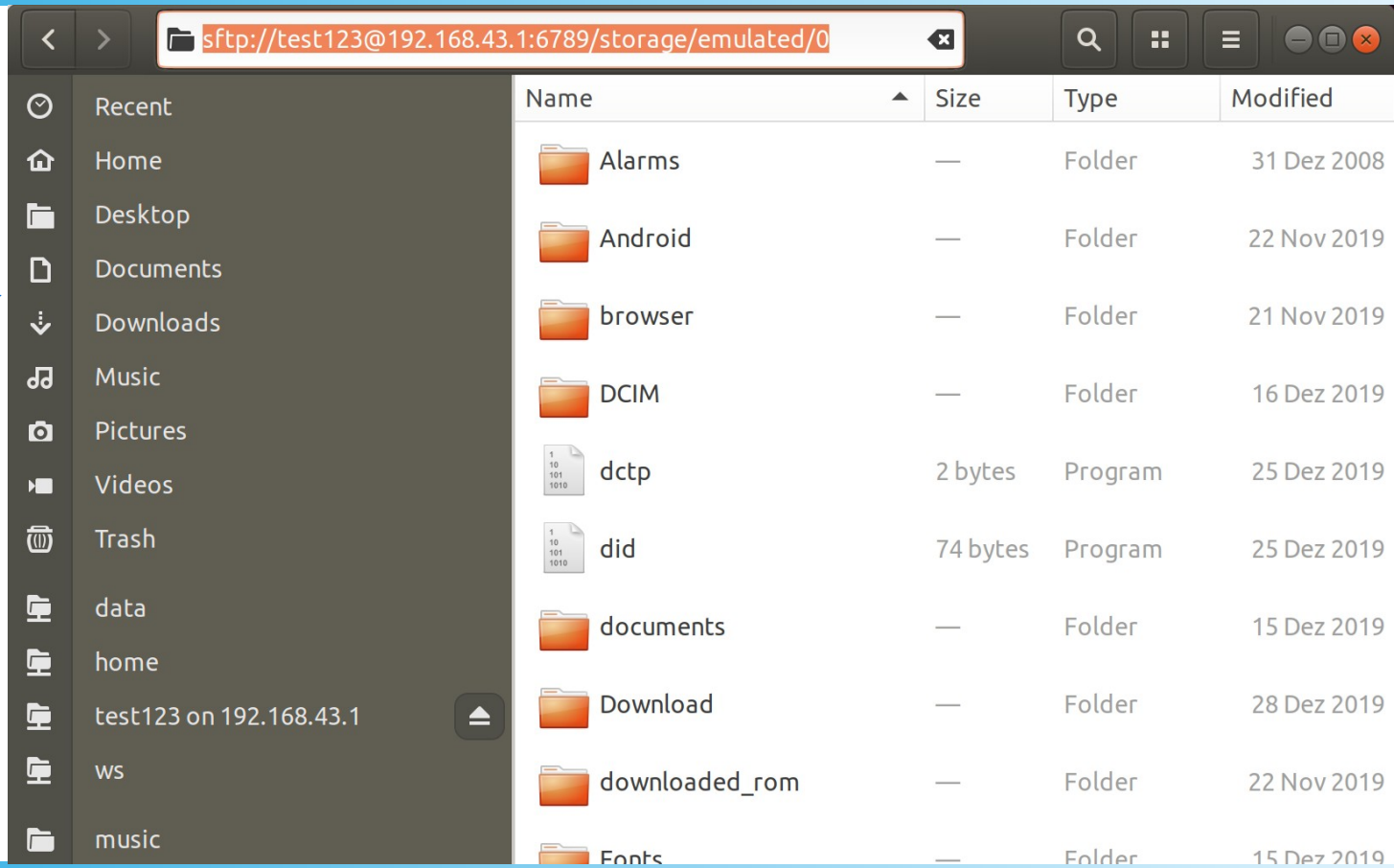
- **MOUNT**: On the “MOUNT DIRECTORY” activity screen, pushing the **“MOUNT” button**, opens the **drive & directory picker**. Drives and directories that have been picked and confirmed will then be added to the drives and directories list on the “MOUNT DIRECTORY” activity screen. The **drives & directories are mounted** at the following **location** (you “cannot navigate” to it because some directories are “hidden” instead you should access it (from remote) using the full path): **`/data/data/ch.becke.sftp_server__s0_v1/files/sftp--s0-v1--mnt`**
- **UNMOUNT**: Drives and directories are unmounted by **swiping** the corresponding item/entry **to the left**.
- **Remote Access**: The mounted directories can be accessed remotely for example as follows (SFTP client example): **`sftp -P 6789 <PHONE-IP>:/data/data/ch.becke.sftp_server__s0_v1/files/sftp--s0-v1--mnt`**
  The different possible connection strings are part of the MAIL Functionality.



SFTP Client: GIO/GVfs

GIO/GVfs: GVfs is GNOME's userspace virtual filesystem designed to work with the I/O abstraction of GIO:
(optional: use the fully qualified path to run the commands! e.g. `/usr/bin/gio mount sftp://...`)

- **`gio mount sftp://test123@192.168.43.1:6789/`**

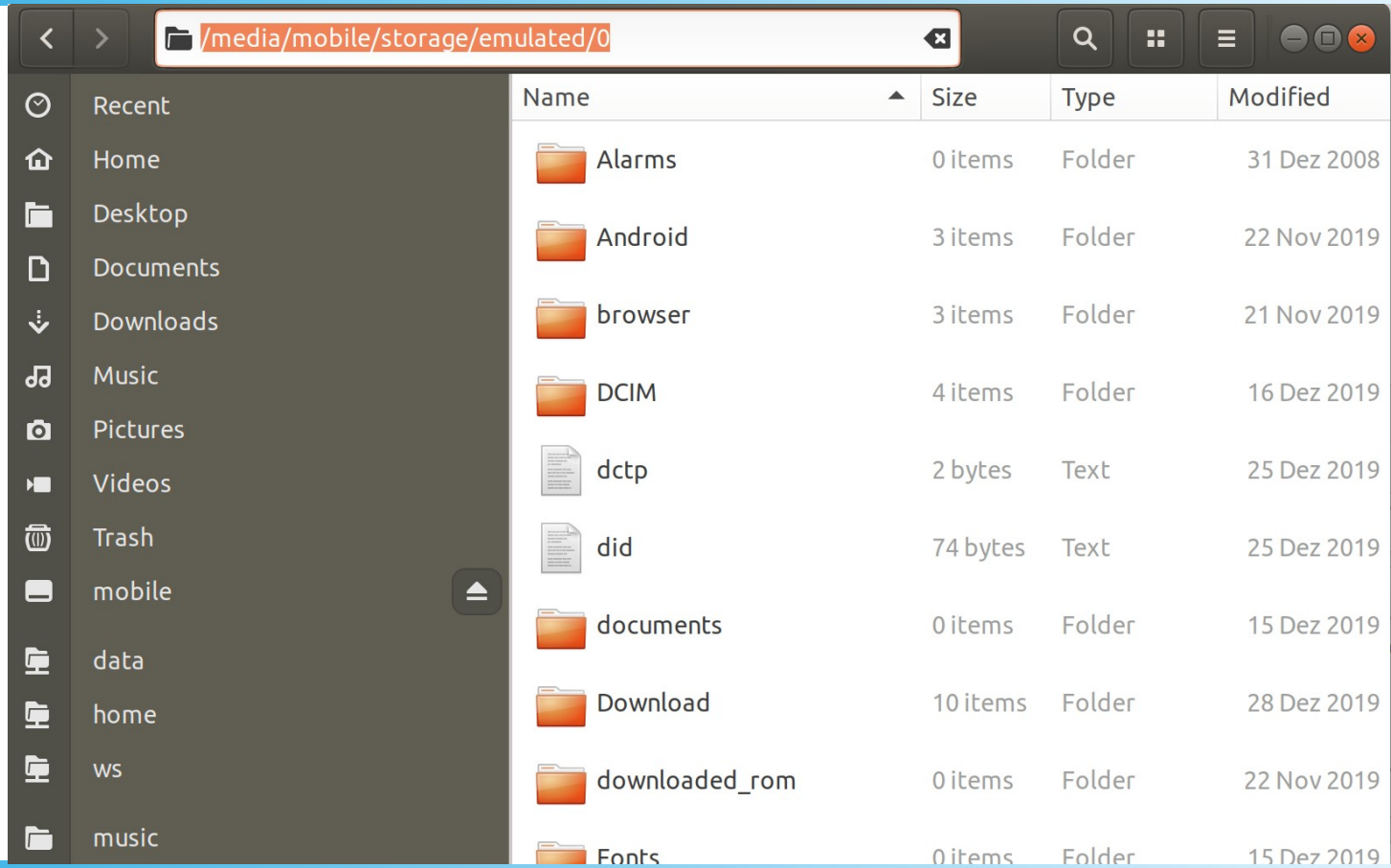


SFTP Client: SSHFS

SSHFS: SSHFS (Secure SHell FileSystem) is a file system for Linux (and other operating systems with a FUSE implementation, such as Mac OS X or FreeBSD) capable of operating on files on a remote computer:

(optional: use the fully qualified path to run the commands! e.g. `/usr/bin/sshfs -p 6789 ...`)

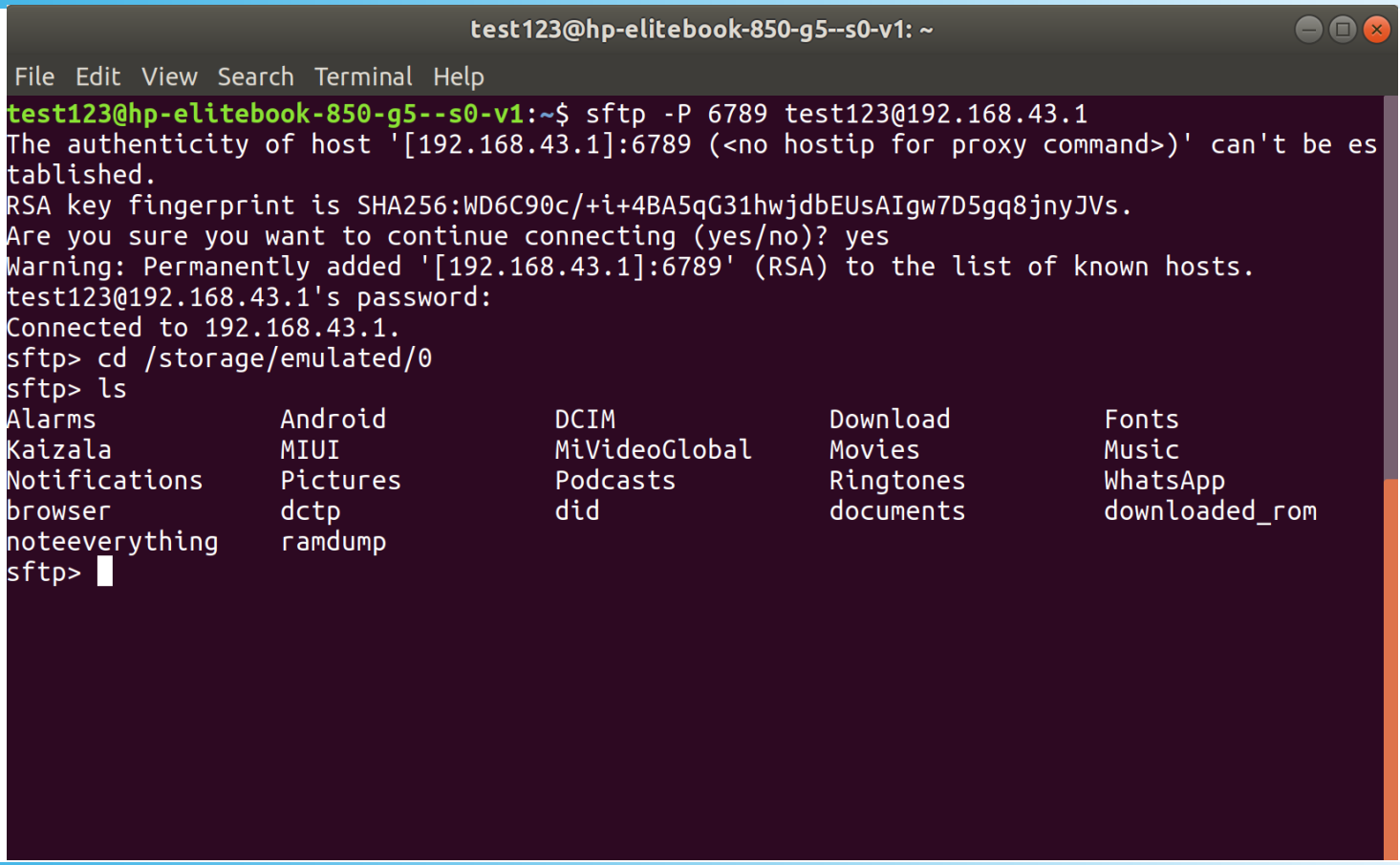
- **`sshfs -p 6789 test123@192.168.43.1:/ /media/mobile`**
- **`sshfs -p 6789 test123@192.168.43.1:/storage/emulated/0 /media/mobile`**



SFTP Client: SFTP

SFTP: sftp is an interactive file transfer program, similar to ftp, which performs all operations over an encrypted ssh transport:
(optional: use the fully qualified path to run the commands! e.g. `/usr/bin/sftp -P 6789 ...`)

- **`sftp -P 6789 test123@192.168.43.1`**
- **`sftp -P 6789 test123@192.168.43.1:/storage/emulated/0`**

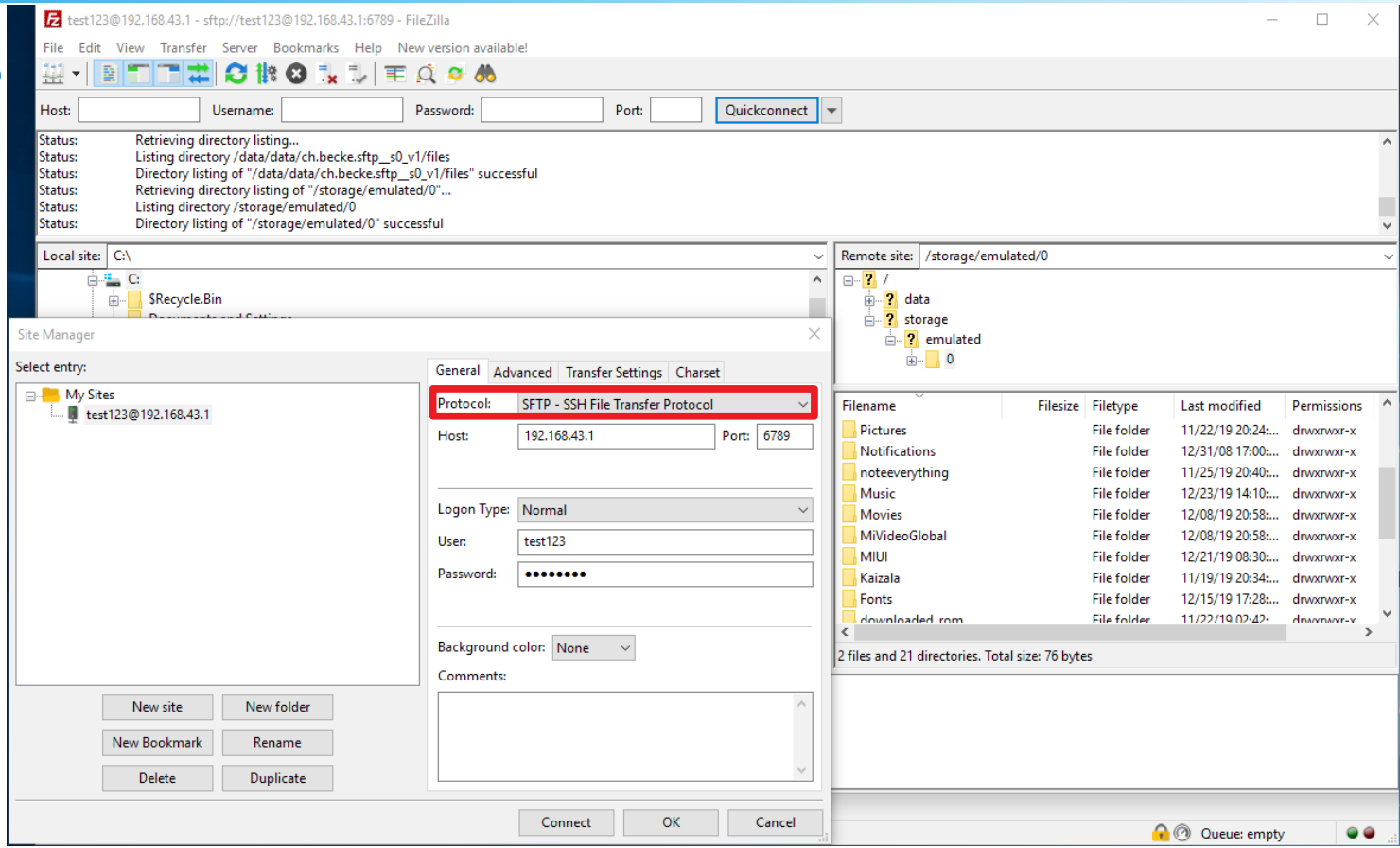


SFTP Client: FileZilla

FileZilla: FileZilla is a free software, cross-platform FTP application. Both server and client support FTP and FTPS, while the client can in addition connect to SFTP servers.

(optional: use the fully qualified path to run the commands! e.g. "`C:\Program Files\FileZilla FTP Client\filezilla.exe" sftp://...`)

- **`filezilla.exe sftp://test123:becke.ch@192.168.43.1:6789/`**
- **`filezilla.exe sftp://test123:becke.ch@192.168.43.1:6789/storage/emulated/0/`**

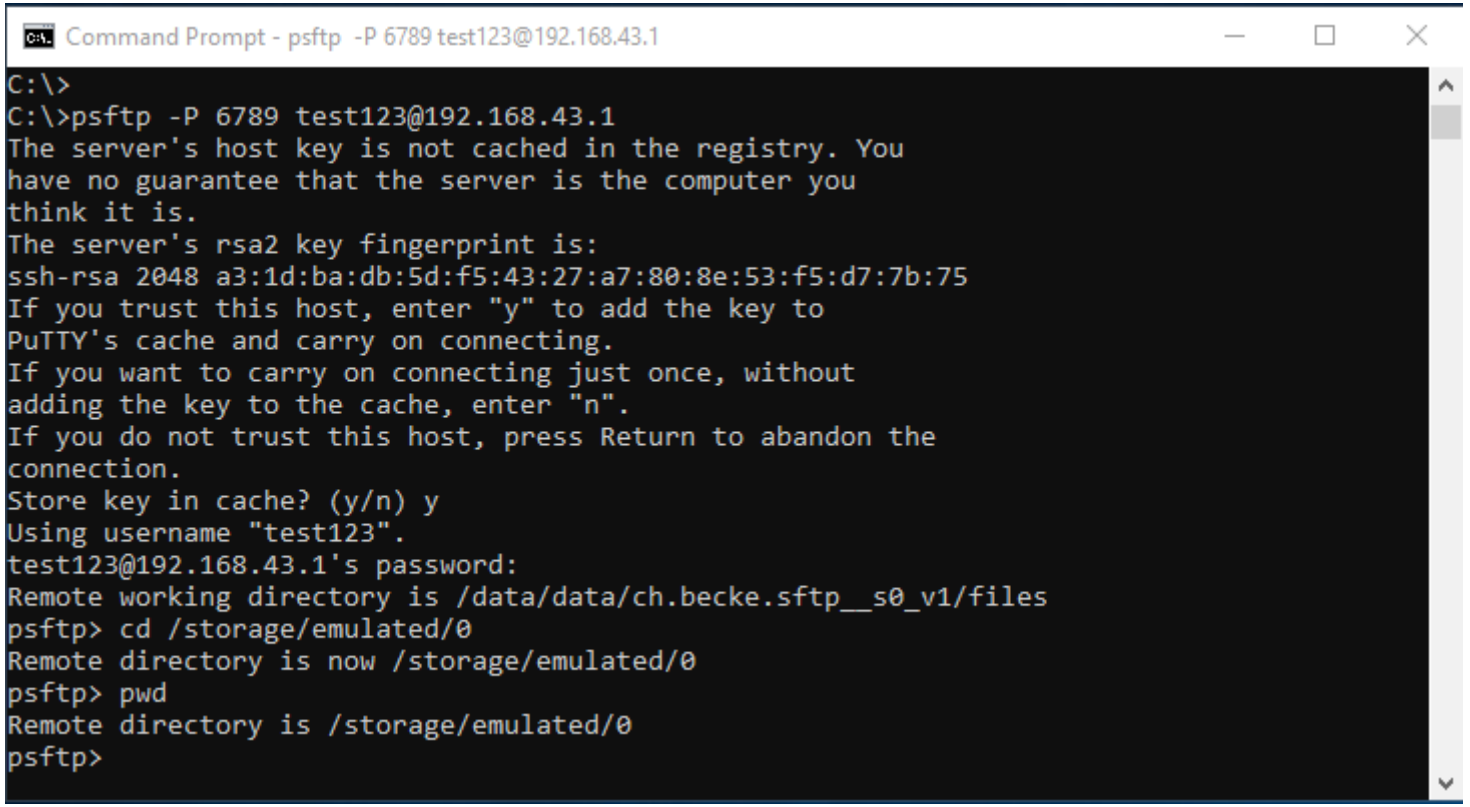


SFTP Client: PuTTY SFTP

PSFTP: PuTTY SFTP client, is a tool for transferring files securely between computers using an SSH connection:

(optional: use the fully qualified path to run the commands! e.g. "`C:\Program Files\PuTTY\psftp.exe" -P 6789 ...`)

- **`psftp -P 6789 test123@192.168.43.1`**



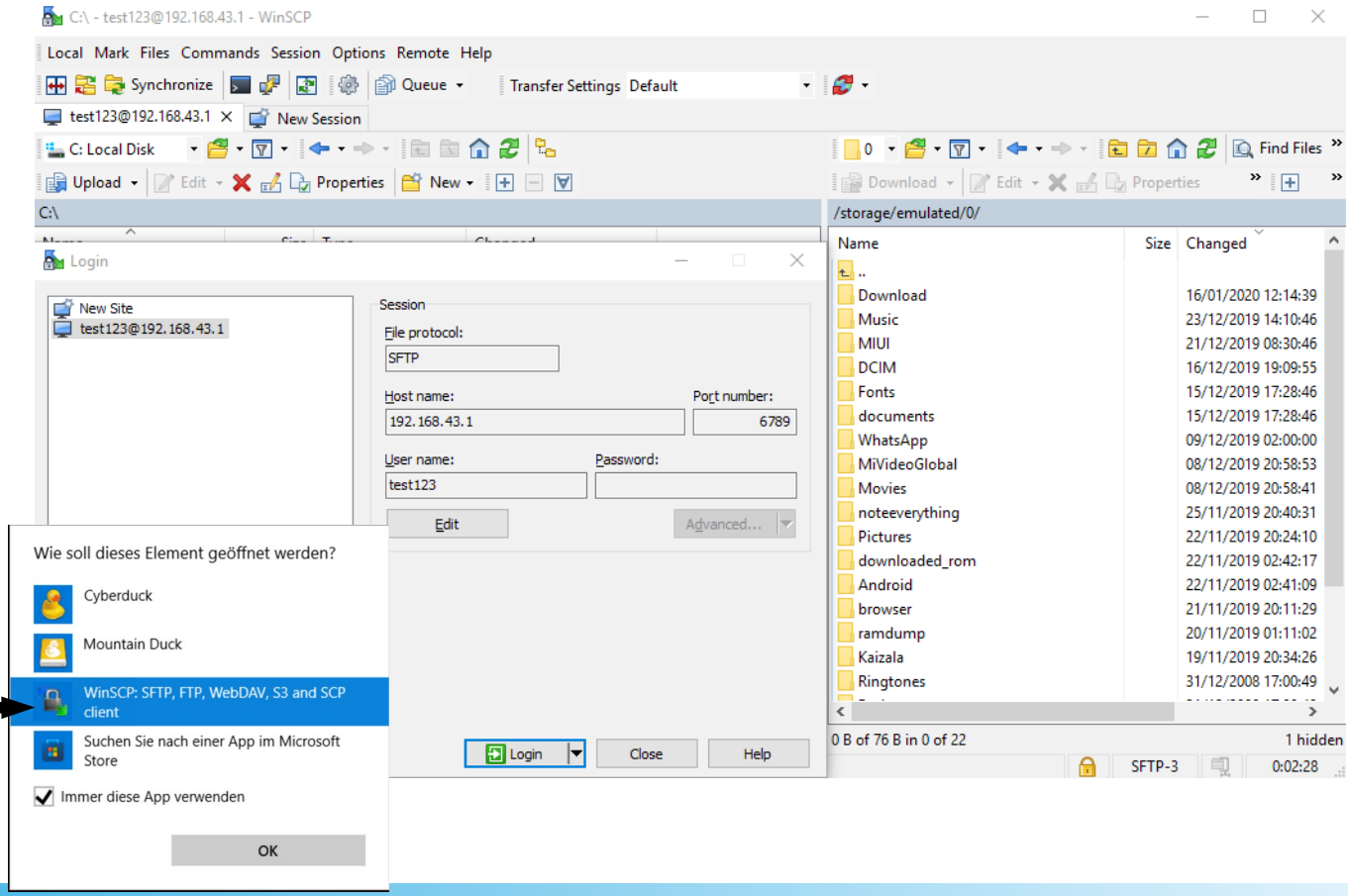
SFTP Client: WinSCP

WinSCP: WinSCP is a free and open-source SFTP, FTP, WebDAV, Amazon S3 and SCP client for Microsoft Windows.

(optional: use the fully qualified path to run the commands! e.g. "`C:\Program Files (x86)\WinSCP\winscp.exe" sftp://...`)

- **`winscp.exe sftp://test123@192.168.43.1:6789/`**
- **`winscp.exe sftp://test123@192.168.43.1:6789/storage/emulated/0/`**

Run (Windows-Key+R):
`sftp://test123@192.168.43.1:6789/`



SFTP Client: Cyberduck

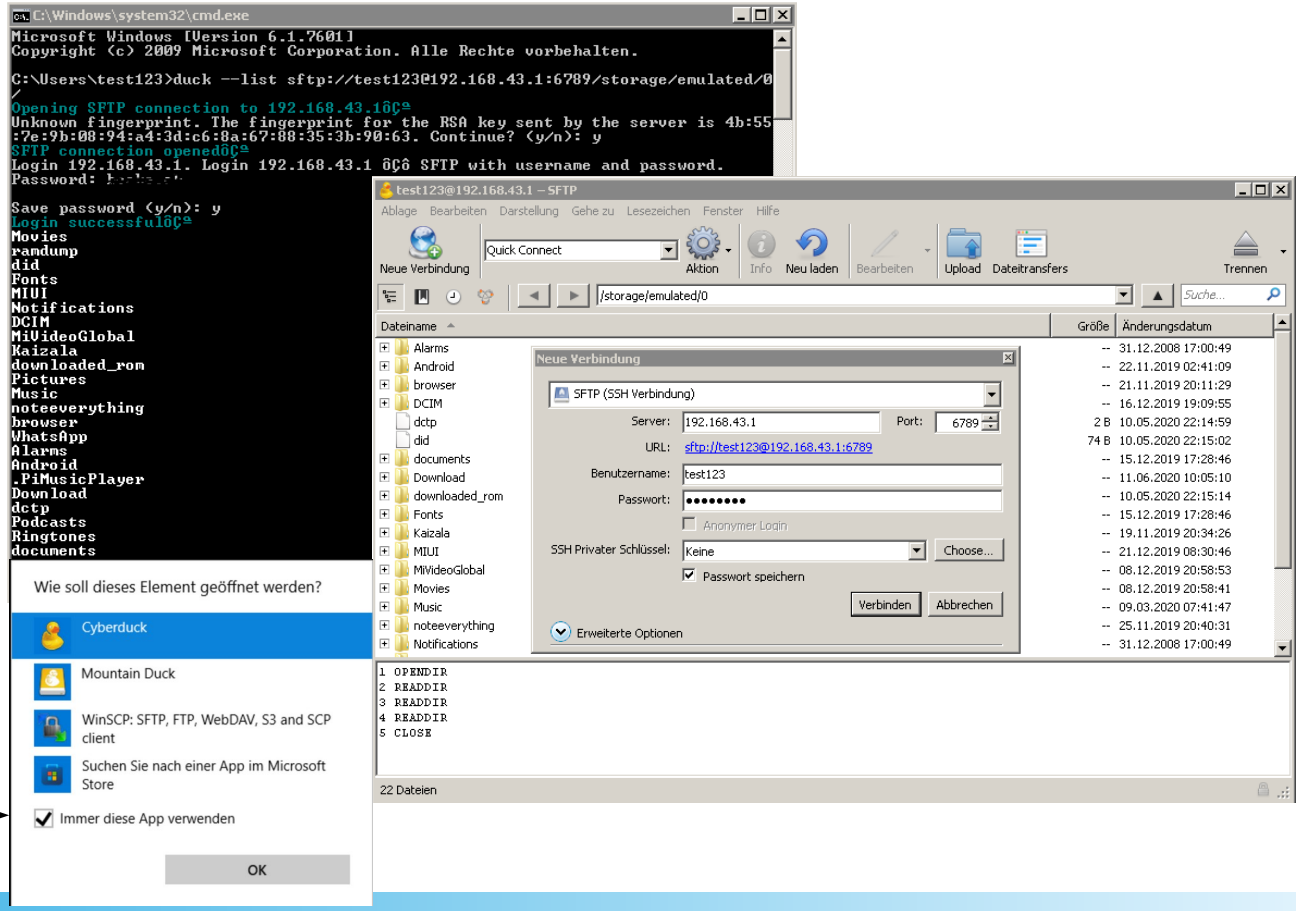
Cyberduck: Cyberduck is an open-source client for FTP and SFTP, WebDAV, and cloud storage, available for macOS and Windows.

CLI: Command Line Interface: "`duck`":

(optional: use the fully qualified path to run the commands! e.g. "`C:\Program Files\Cyberduck\Cyberduck.exe" sftp://...`)

- **`duck --list sftp://test123@192.168.43.1:6789/storage/emulated/0/`**
- **`duck --download sftp://test123@192.168.43.1:6789/storage/emulated/0/DCIM/Camera/*.* .`**

Run (Windows-Key+R):
`sftp://test123@192.168.43.1:6789/`



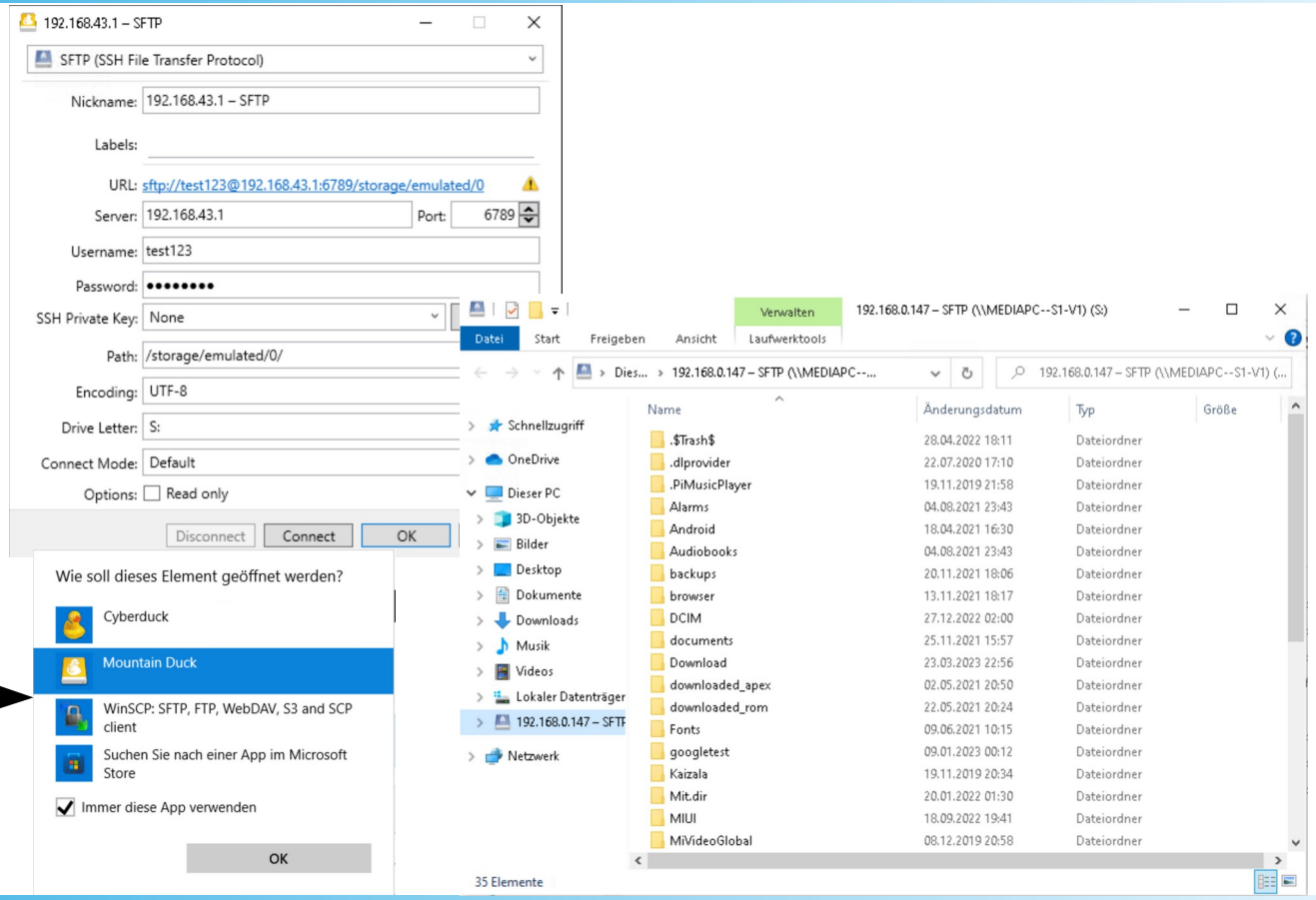
SFTP Client: Mountain Duck

Mountain Duck: Mountain Duck lets you mount server and cloud storage as a disk in Finder on macOS and the File Explorer on Windows:

(optional: use the fully qualified path to run the commands! e.g. `"C:\Program Files\Mountain Duck\Mountain Duck.exe" "sftp://..."`)

- **`"Mountain Duck.exe" sftp://test123@192.168.43.1:6789/`**
- **`"Mountain Duck.exe" sftp://test123@192.168.43.1:6789/storage/emulated/0/`**

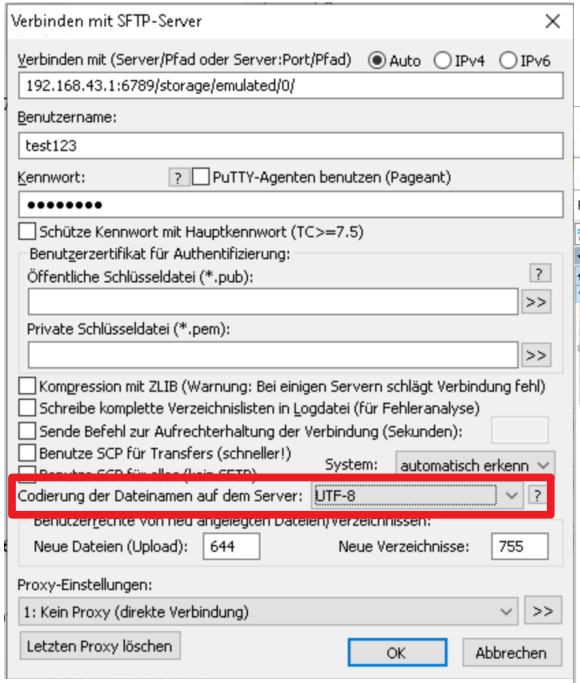
Run (Windows-Key+R): **`sftp://test123@192.168.43.1:6789/`**

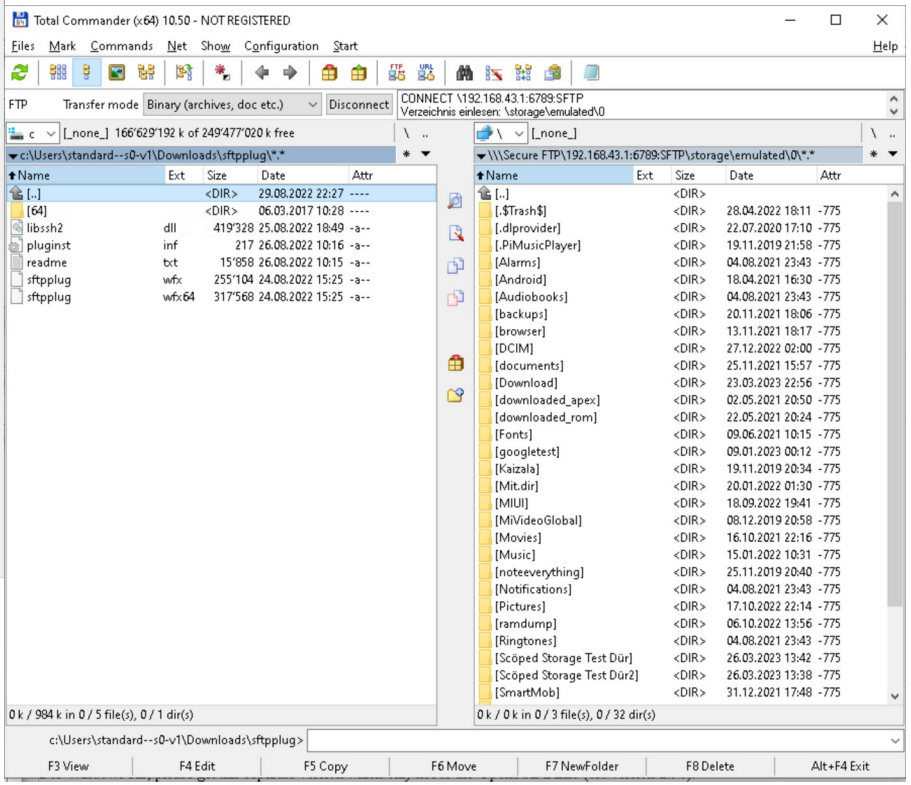


SFTP Client: Total Commander

Total Commander: Is a Shareware file manager for Windows.

- Install the SFTP Plugin.
- Set the **encoding** to **UTF-8** (SSH default encoding)!
- Strange the connection requires 2 attempts before connecting (only happens with Total Commander)?!

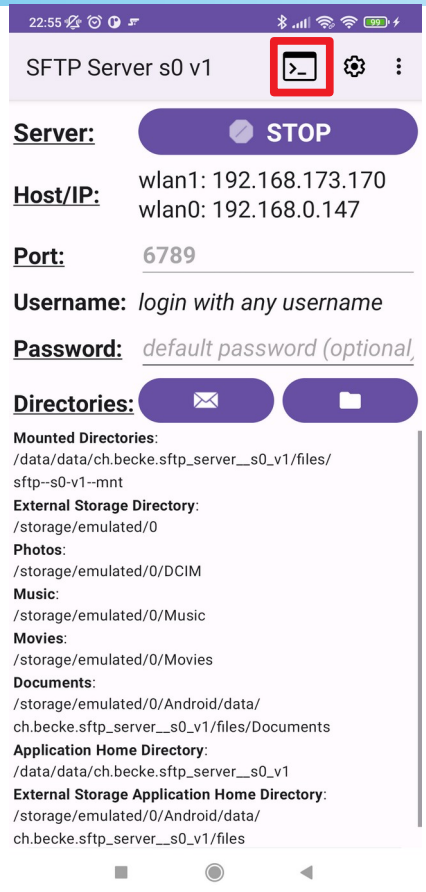




SYSTEM OUT (Password & Public Key)

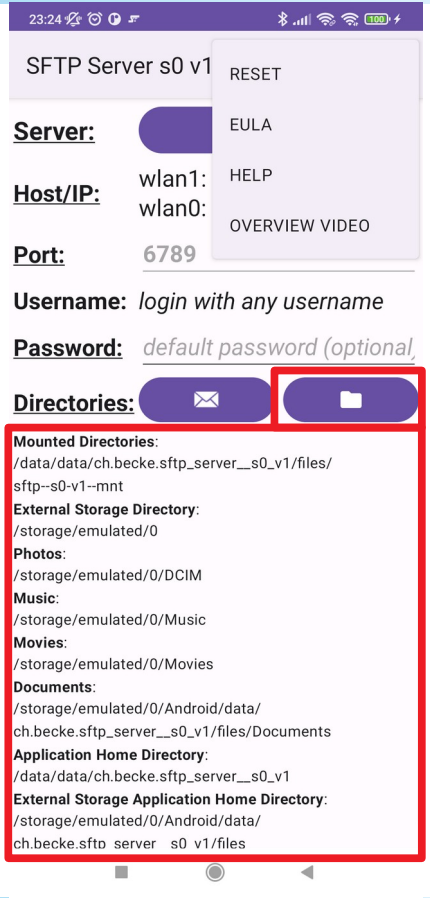
System Out: Clicking on the Terminal Icon in the top app bar brings the user to the System Out Screen where different System and Log related Information (e.g. password during the login process) is displayed.

- **Password**: During the first login the password (default or generated, depending on the SETTINGS) is shown on screen.
- **Public Key**: After successful login the client public key (if any and if of type **RSA or ECDSA**) is stored on the SFTP server automatically (if “Save Public-Key” is activated in the SETTINGS). And from then no password is required anymore (when connecting from this client/public-key).
- Click on the “**<-**” (back) button to get back to the main screen!
- Clicking on the “**SYSTEM OUT**” menu item (in the overflow menu, top right), on the main screen, brings you back to this screen.



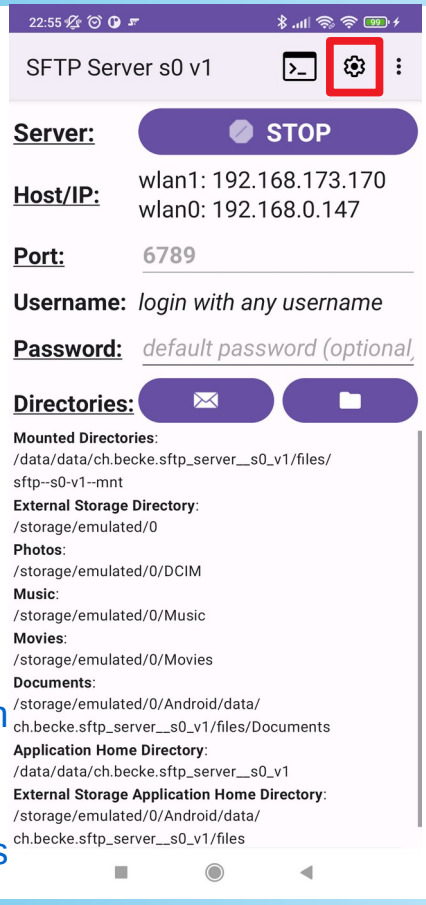
Directories

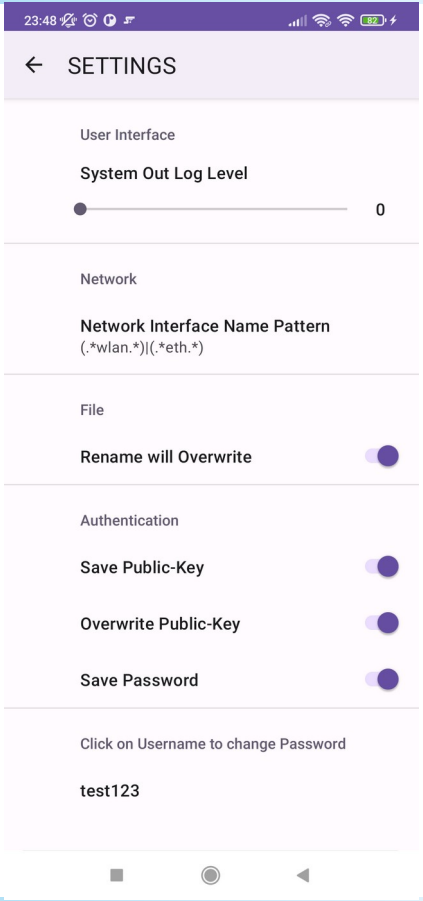
- **Directories**: Most directories on the android file system are hidden and therefore some of the most important directories are listed on the main screen in the lower (scrollable) part. With the help of these directory paths the user can easily find the files he is interested in. For example one of the most important directories is the “External Storage Directory” (e.g. “/storage/emulated/0”) where all the public data is stored.
- **Mounted Directories**: Clicking on **MOUNT** (Folder Icon) lets you mount additional drives & directories below the application home directory. (For additional information see previous slide “MOUNT”)



SETTINGS

- **System Out Log Level**: The higher the Log Level the more information will be written to the SYSTEM OUT Screen.
- **Network Interface Name Pattern**: A regular expression filtering the network interface names. On the main screen only the Host/IP addresses of the interfaces matching this regular expression will be shown.
- **Rename will Overwrite**: Renaming a file will overwrite another file with the same name if there exists one (this option is required in the context of SSHFS).
- **Save Public-Key**: If activated the client public-key (if any and if of type **RSA or ECDSA**) is stored after successful login. And from then no password is required anymore (from this client/public-key).
- **Overwrite Public-Key**: If activated a non matching client public-key is overwritten with the new client public-key after successful password login.
- **Save Password**: If activated the Username & Password is stored (hash value, not clear text!)
- **Click on Username to change Password**: By clicking on the username you can change the password (stored as hash value!) for this user.

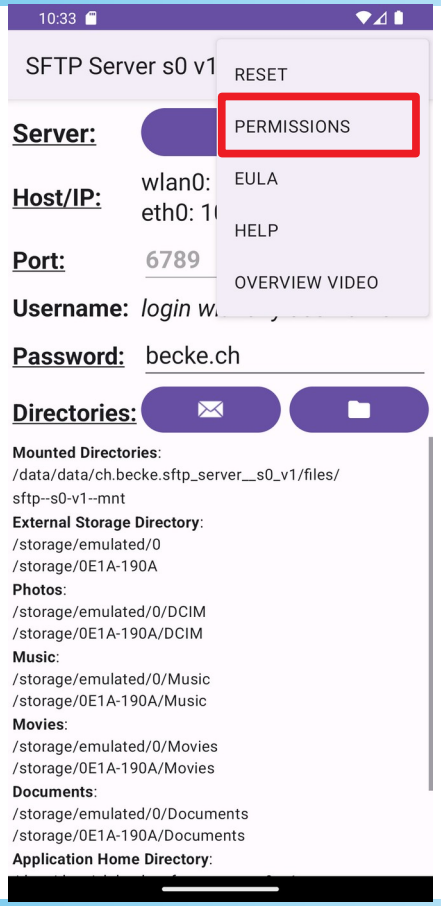


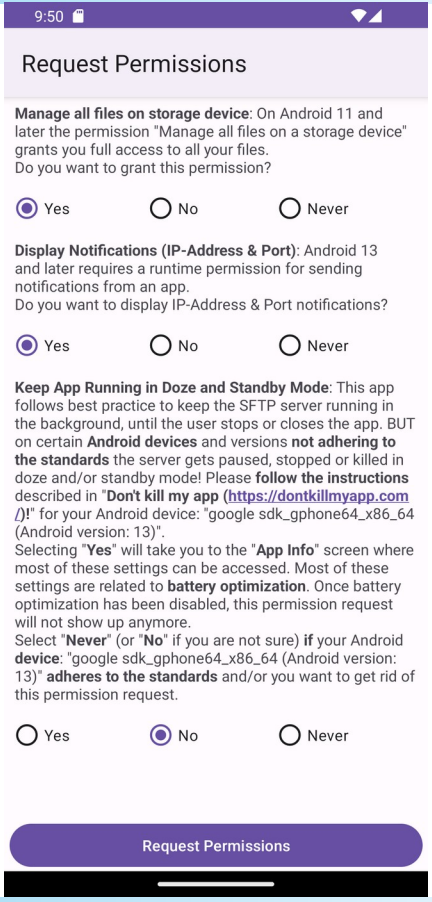


PERMISSIONS

Clicking on the “**PERMISSIONS**” **Menu Entry** in the overflow menu will display all Permission Requests that have not been granted until now:

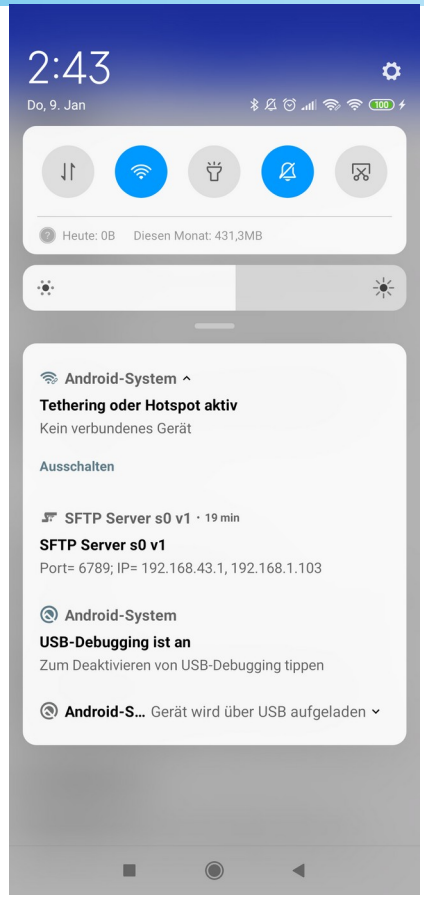
- **Manage all files on storage device**: *On Android 11 and later the permission "Manage all files on a storage device" grants you full access to all your files*.
- **Display Notifications (IP-Address & Port)**: *Android 13 and later requires a runtime permission for sending notifications from an app*.
- **Keep App Running in Doze and Standby Mode**: *This app follows best practice to keep the SFTP server running in the background, until the user stops or closes the app. BUT on certain **Android devices** and versions **not adhering to the standards** the server gets paused, stopped or killed in doze and/or standby mode! Please **follow the instructions** described in "**Don't kill my app (https://dontkillmyapp.com/)!**" for your Android device: "...".*

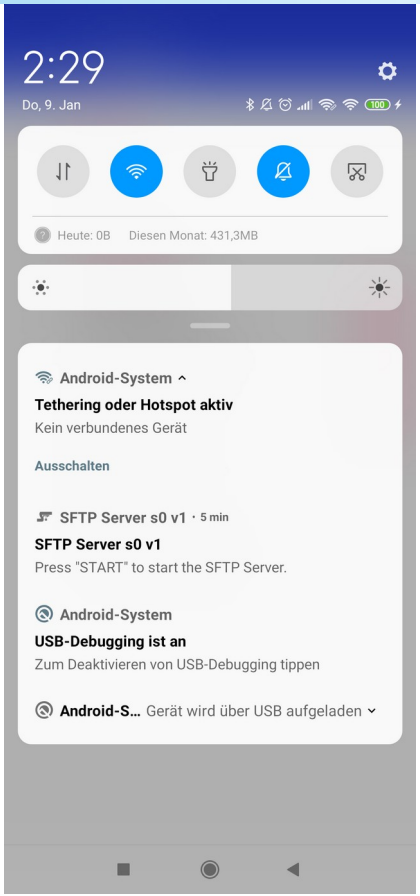




Notification

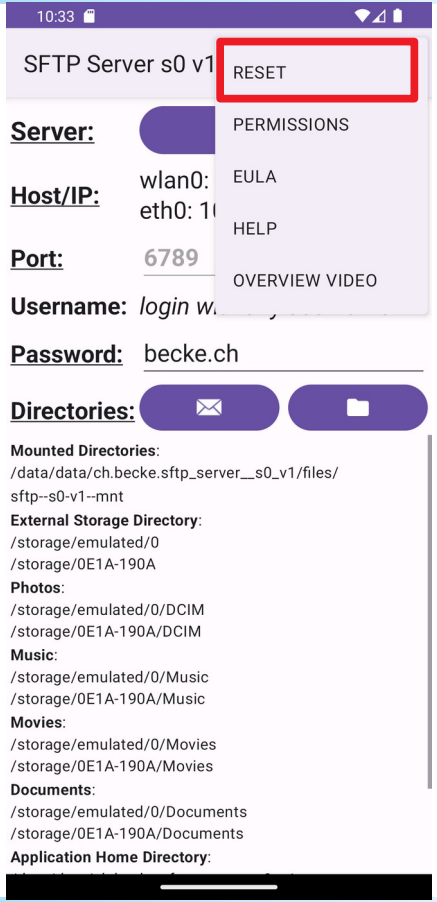
Notification: Every time the “STOP” respective “START” button is pressed the Notification message is updated accordingly.





RESET

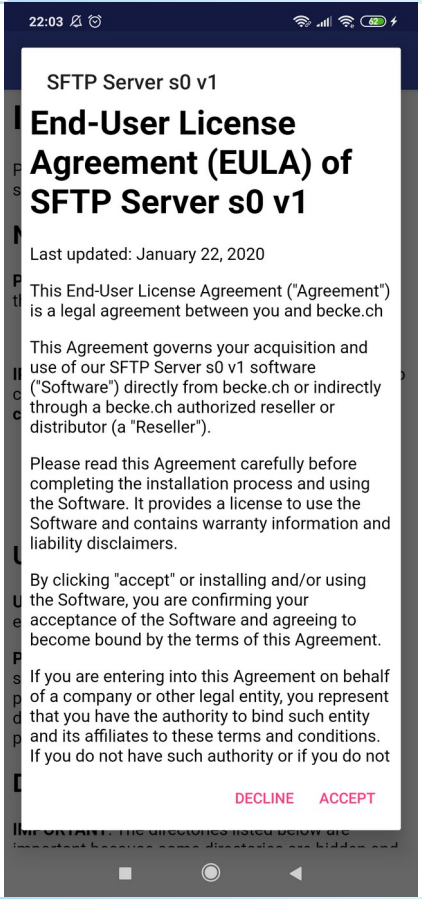
RESET: Pushing the “RESET” menu entry in the overflow menu, will delete all usernames & passwords, all client public-keys, all mounted directories and will set back the settings to their initial state.



Privacy Policy and End-User License Agreement

Privacy Policy and End-User License Agreement (EULA) of SFTP Server s0 v1:

- **Privacy Policy**: https://www--s0-v1.becke.ch/app/becke-ch--sftp-server--s0-v1/legal/becke-ch--sftp-server--s0-v1--privacy-policy.html
- **EULA**: https://www--s0-v1.becke.ch/app/becke-ch--sftp-server--s0-v1/legal/becke-ch--sftp-server--s0-v1--eula.html

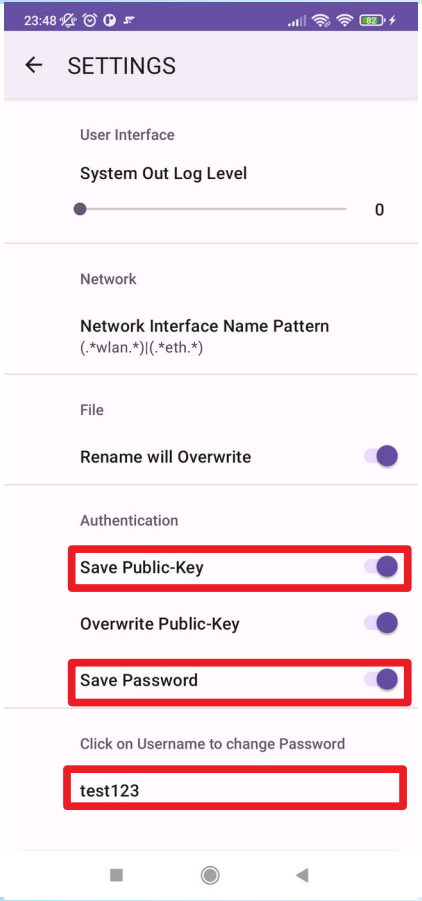


APPENDIX

PASSWORD & PUBLIC KEY FUNCTIONALITY

Open the SETTINGS screen:

- **Change User (e.g. “test123”) Password**: Scroll to the bottom and click on the Username (e.g. “test123”) to change his password. The **minimal password length is 6 characters**!
- **Save Password**: If activated the **Username & Password is stored** (**hash value**, not clear text!). Otherwise each time the user logs in he has to enter the default respective a newly generated password.
- **Save Public-Key**: If activated the client **public-key** (if any and **if** of **type RSA or ECDSA**) is **stored after successful login**. And from then on **no password is required anymore** (from this client/public-key).



LOGIN & PASSWORD-CHANGE

1. **Start “SFTP Server s0 v1”**: When the app launches the SFTP Server starts automatically.
2. **(Optional) Change Default Password**: On the **main screen**, click on “**Password**” field to change it. The **minimal default password length is 6 characters**! If you **clear all characters (0 length)** then **a random password gets generated** during login.
3. **SFTP Client: Connect to Server**: Use one of the SFTP Clients and connect to the SFTP Server on the designated IP-Address & Port-Number. (Using the “MAIL” functionality the user can send the SFTP connections strings to the Client PC and from there, with the help of these connection strings, easily connect to the SFTP Server).
4. **Login**: Enter the **Username** (e.g. “test123”) and **Default (respective Generated) Password** that is shown on the SFTP Server App Screen during the login process.
5. **(Optional) Change User (e.g. “test123”) Password**: Open the **SETTINGS screen**, scroll to the bottom and click on the Username (e.g. “test123”) to change his password. The **minimal password length is 6 characters**!

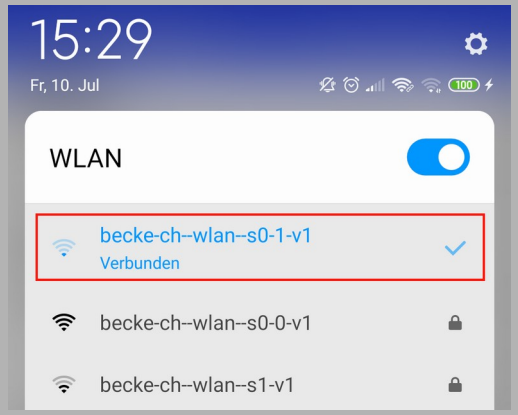
PUBLIC KEY (PASSWORDLESS) LOGIN

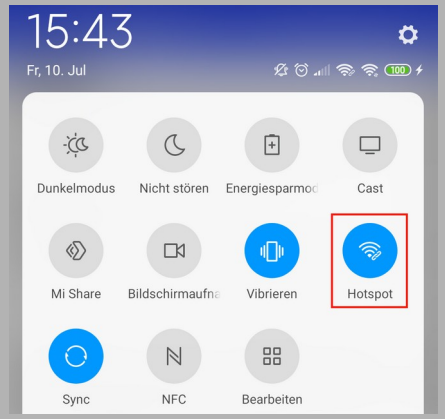
There exist many ways to generate a Public/Private Key-Pair of Type RSA (or ECDSA) (on Linux). Below is listed one of them:

1. **SFTP (Linux) Client: Terminal/Console**: On your SFTP Linux Client PC, open a new Terminal/Console window.
2. **`ssh-keygen -t rsa`**: Execute the command "**`ssh-keygen -t rsa`**" to start generating a **Public/Private Key-Pair of Type RSA**. (for **ECDSA** type "**`ssh-keygen -t ecdsa`**")
3. **Directory & Filename**: Enter the directory and filename where to store the Public/Priate Key-Pair (**`/home/<USERNAME>/.ssh/id_rsa`**) (respective **`/home/<USERNAME>/.ssh/id_ecdsa`**): (**I suggest to accept default directory and filename by just pressing the enter key!**).
4. (Optional) **Passphrase**: Enter passphrase (empty for no passphrase, up to you to decide)
5. **Repeat Passphrase**: Enter the same Passphrase again and you're done with the process of generating a Public/Private Key-Pair of Type RSA (or ECDSA).
6. **SFTP (Linux) Client: Login**: Login one last time with Username & Password. The Public Key will be stored on the SFTP Server and from then on Public Key (Passwordless) Login is enabled.

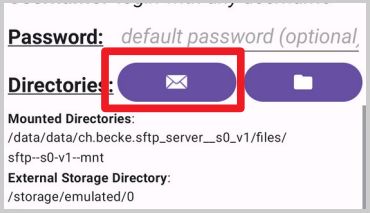
WALK THROUGH CONNECTIVITY SAMPLE

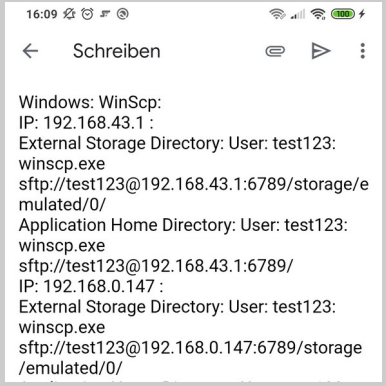
1. **Mobile**: Activate **WLAN** and/or **HotSpot**





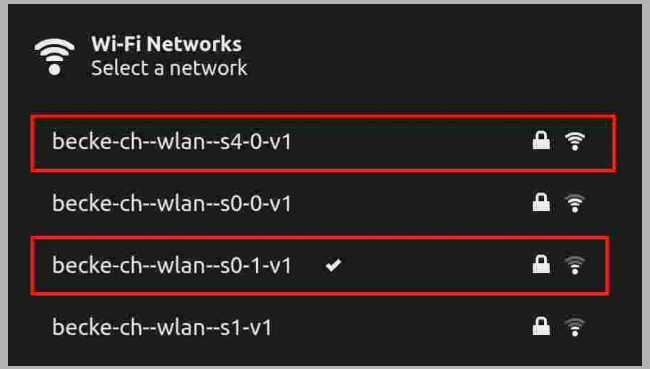
2. **Mobile**: “START” the SFTP Server (if not already started) and click on “**MAIL**” to email connection strings to your Laptop/PC.
If you change the port number of the server you need to STOP/START the server again!



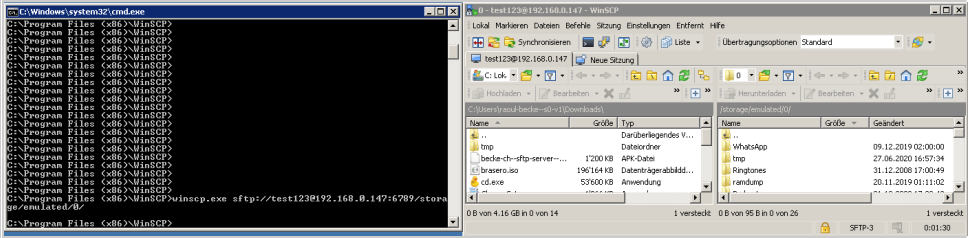


WALK THROUGH CONNECTIVITY SAMPLE

3. **Laptop/PC**: **Connect** to the **same WLAN / LAN** where the SFTP Server is running.
In a nutshell the only mandatory network requirement in this context is that the **Server IP Address and Port must be reachable from the Client Laptop/PC**!

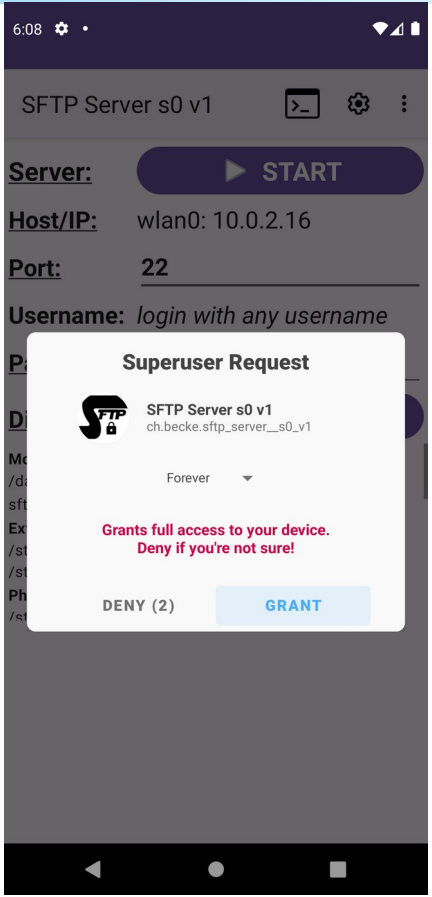


4. **Laptop/PC**: **Start** the **SFTP Client** using the **connection string** in the **email** that you sent previously in step 2. When prompted for the **password**, enter the password that is **shown on the Mobile App Screen** (default “becke.ch”).
Make sure that no **Firewall** is blocking the traffic!



PRIVILEGED PORT SUPPORT

- **Privileged port (ports below 1024) support for rooted devices**: To run the SFTP Server on the standard SFTP Port “22” respective in general to run the SFTP Server on a port below “1024” requires a rooted device, i.e. the ability to run the "su" command with root permissions!
- **Superuser Request**: (Once the device is rooted and) the first time when setting the Port to a value below “1024” the user is prompted with a “Superuser Request”. Granting this request will allow the SFTP Server App to run on the designated port below “1024”.



FAST SFTP VIA USB

As already mentioned: "*For optimal speed connect to 5GHz WLAN and use a fast SFTP Client Program!*"
OR alternatively to 5GHz WLAN use SFTP via USB (using ADB):

1. **Enable adb debugging on your device**: To use adb with a device connected over USB, you must enable **USB debugging** in the device system settings, under Developer options – see: https://developer.android.com/studio/command-line/adb#Enabling
2. **Download SDK Platform Tools / ADB (Android Debug Bridge)**: If not already installed (as part for Android Studio), download SDK Platform Tools containing the ADB from here: https://developer.android.com/studio/releases/platform-tools and follow the installation instructions.
3. **Start the ADB server**: "**`adb -a nodaemon server start`**". "**`-a`**": (optional) "listen on all network interfaces, not just localhost". "**`nodaemon`**": (optional) "start the server in the foreground, not as daemon".
4. **Connect USB cable**: Connect the mobile phone via USB to the computer (and select "data transfer" in the upcoming dialog).
5. **List attached devices:** "**`adb devices`**": The device ID is required to set up the network redirection.
6. **Set up network redirection (port forwarding):** "**`adb -s <deviceId> forward tcp:<localPort> tcp:<remotePort>`**" for example : "`adb -s dda0d48f forward tcp:56789 tcp:6789`"
7. Connect with your favourite SFTP Client to localhost (127.0.0.1) or any other IP address your computer exposes on "***`<localPort>`***" for example: "`sftp -P 56789 test123@localhost`"

THE END

becke.ch

email: ssh--s0-v1@becke.ch

homepage: http://becke.ch/app/becke-ch--sftp-server--s0-v1/

download: https://play.google.com/store/apps/details?id=ch.becke.sftp_server__s0_v1