User Manual - Wirespace

Yash Shah, Kumar Saunack, Kumar Saurav

1 Introduction

Wirespace is a decentralized file server application which, if permitted, allows clients to directly access file storage on the server side. To put it simply, the software creates a local file server which clients can connect to and read/write directly. The project runs on the browser, so only the server needs to have a copy of the software. The only exception is when the client wants to use the in-place editor.

1.1 Features

- **Authentication**: A unique key is generated for each file/folder shared, which is to be entered once, following which the entire session is authenticated for the client. Needless to say, no client can work without the key.
- **Access hierarchy**: There are types of access levels: **Read** and **Read/Write**. In the former, no write access is allowed.
- **Local editing**: The software allows clients to collaboratively edit files shared, given that write access is allowed

2 Getting Started

1. Install the package:
   (a) Clone the/ github repository `https://github.com/ys1998/Wirespace.git` using

   ```
   git clone https://github.com/ys1998/Wirespace.git
   ```

2. Install dependencies: Navigate to the folder containing the package. Run the following command to install the dependencies required (this requires root priveleges):

   ```
   ./wirespace.sh --install
   ```

   Alternatively, check the dependencies which are listed in *requirements.txt* and install them manually.

3. Execute the program: The script *wirespace.sh* has 6 possible flags:
   - `-i` or `--install` Installs the required dependencies for the program to run and creates a new superuser (admin) with user input
   - `-s` or `--start` Detects the IP address of the computer and runs the server on that IP address on port 8000

- `-l` or `--local` Starts a localhost server (or 127.0.0.1) on port 8000
- `-c` or `--custom` Inputs an IP address and a port number and attempts to start a server on that address and port
- `-n` or `--newuser` Creates a new superuser
- `-e` or `--editor` Opens the editor program
- `-h` or `--help` Display the list of possible flags

The usage is `./wirespace.sh [-i|-s|-l|-c|-n|-e|-h]`. Once the server has started, you can minimise the terminal window. All requests from clients are logged in the terminal.

3 Server side

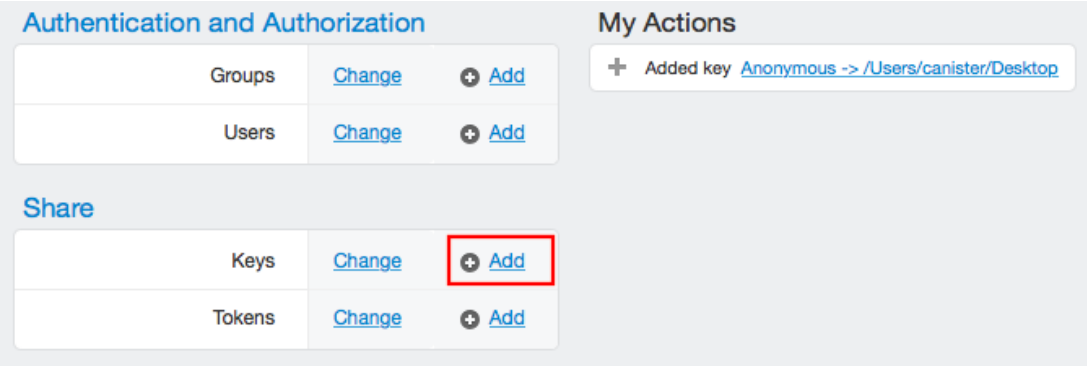
On the server side, download and install the Wirespace software and create a superuser. Then run the server using either the *local*, *start* or *custom* flag. Then follow these steps to set up sharing:

1. Open the *host* url: On starting the server successfully, you get a message similar to the following in the terminal:

```
Django version 1.11.5, using settings 'wirespace.settings'
Starting development server at http://192.168.0.4:8000/
Quit the server with CONTROL-C.
```

   Open the URL + `host` (here it would be `http://192.168.0.4:8000/host`) to visit the admin login page

2. Login using the superuser/admin credentials created
3. Under the share group, click on `Add` under `Keys`:



4. The fields are:
   - Expiry: Set the time on which the sharing link should expire
   - Path shared: The *absolute* path (this is very important) to the directory which you want to share, with a forward slash at the end
   - Permission: Set the access rights for the client - `Read` or `Read/Write`

- Space allotted: If `Write` access is granted, the amount of space the client is allowed to write to

5. On saving, you will be redirected to a page which stores all your created keys. You can manage/edit these keys by clicking on the hyperlinked links under the `Link` headers

6. To share your folders with a client, make sure that client is on the same network and copy and send the link created which is listed under the `Link` header

That's it from the server side. Additionally, you can manage the list of superusers under the `Users` section in the `Authentication and Authorization` sidebar

4 Client side

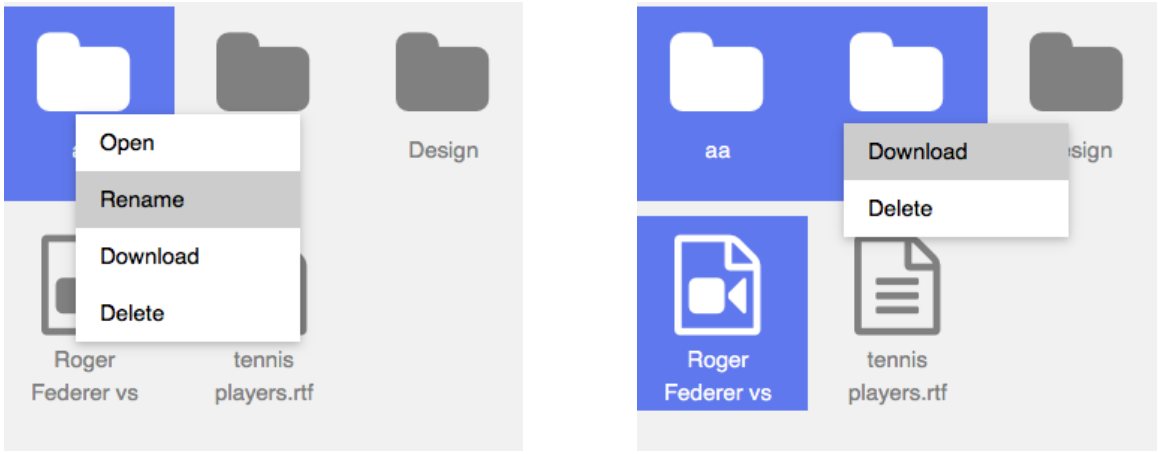
The server must have provided you with a link which would look something like this:

$$\underbrace{\texttt{192.168.0.4:8000}}_{\textit{Server's IP address}} / \underbrace{\texttt{8368427f2d3db49f}}_{\textit{Unique key}}$$

Visit this URL to authenticate your session and browse the shared directories. Browsing is similar to the default options for browsing in your file explorer.

4.1 Gestures

- Single left click to select an icon. To select multiple files/folders, hold down modifier keys (`Shift`, `Alt`, or `Ctrl`)
- Double left click to open file or folders. By default, clicking on folder opens up its contents while clicking on a file views its content, or if unable to do so, downloads it
- Right click to open context menu. If no item is selected, or right clicking on an item not selected, the first right click selects that item and the second right click opens the context menu. Clicking elsewhere cancels the menu
- Drag files/folders into another folder to move the former inside the latter
- Drag and drop files from the desktop/file explorer to upload, provided that write access is granted.



(a) single selection

(b) multiple selection

Figure 1: Context Menu

4.2 Supported actions

Open	Double left click
Rename	Right click → Context menu → Rename
Go back to a directory	Click on the corresponding directory in the address bar
Instantaneous search	Type in the search bar
Clear search	Clear search bar
Download entire directory	Download icon on top right
Download selected files/directories	Right click → Context menu → Download
Delete selected files/directories	Right click → Context menu → Delete
Upload files	New button (top left) → Upload Files
Upload files	Drag and drop files
Upload folders	New button (top left) → Upload Folders
Move file/folders	Select required files/folders → Drag to destination folder
Create new folder	New button (top left) → New Folder

A few notes here:

- On searching, files and folders are searched recursively. And since we do not support indexing or caching, the program can hang while searching for fairly deep nested directory structures.
- Multiple files can be uploaded at once
- Folder upload is supported in only select browsers such as Mozilla Firefox and Google Chrome
- Files are not moved to the trash on deletion. Instead, they are permanently deleted
- Folders or multiple selections are downloaded as compressed files
- The application is a single page web-app. So, back button does not work as expected

4.3 Local Editing

To support local editing, you need to have a copy of the software on your system.

1. Navigate to the software folder in terminal
2. Run `./wirespace.sh -e`
3. Enter the link that was provided to you by the server in the URL box and click on `Connect`
4. Navigate to the file you want to edit and click on `Edit`. The file should open up in the default editor
5. After editing, save the file and exit the editor. Click on `Save file remotely`