

Publishing Your Website Or A Directory Of Content On IPFS (InterPlanetary File System) For Linux, Windows or Apple PCs

How to create free unblockable storage and website protection from interference by tyrants

Revised 7 Feb 2021 to add additional Windows install step 7a for publishers. Also added steps 12a-c for content management and increasing initial access speed. Plus in step 9 a note for remote publishing.

This document is available on IPFS at

<https://ipfs.io/ipns/k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzt6mobpumbiruhv/FreedomPosts/PublishingYourWebsiteOnIPFS.html>

If you're not familiar with using the terminal command line, **don't be intimidated**. You will follow the **same simple 3-step process every time after doing it the first time**. Get a friend to help you accomplish the setup and first upload. **Be sure to save the information specified to go in a text file**. Then, **you or your grandmother will be able to do this solo thereafter!**

PC requirements: VPN is supported. Quad core processor with 8Gb ram (an old laptop may do). USB3 or higher i/o ports will access the external hard drive(s) faster, but USB2 will work. Preferably, the PC will be dedicated to this task. If not, other programs may be slowed. A Raspberry Pi 4 with it's light OS and 4Gb ram may also be used, which can make cloning to other Raspberry Pi locations easier.

The Setup: Only done once, maybe with a little help from a friend

If you're only putting a complete directory/folder of files on IPFS, skip to step 2.

1) **To upload a website which can be completely browsed**, it must be entirely contained within a single folder. Also, all links to your website content must be relative, not referencing a domain name or other location outside of the folder. Links to locations outside the folder, or using the domain name of the website, will not be mirrored with an IPFS file location hash and preserved from blocking. Any script which references files outside the folder or is processed on the website host server will lose functionality. Addressing these limitations is the first step toward completely mirroring a website on IPFS. However, you may proceed with your site as is and make the site improvements later, knowing some of it won't be on IPFS at first.

2) If the folder containing the website or content is not on the machine you'll be running IPFS on (which is preferable), a **static network path to the folder needs to be established**. This path needs to not change over time for the simple three-step update process to not change.

3) The machine which the IPFS node program will be running on **needs to always stay on**, to the maximum extent possible. It is the primary node for your content before it's seeded by people accessing it, or duplicated on another node (addressed below).

4) It's best to have the IPFS data storage and program installed on a **physically connected external drive to facilitate cloning**. The drive should be at least 1Tb and preferably double the size of the folder being put on IPFS.

5) On the IPFS drive, **create a folder named IPFS** in the root.

6) **Download the IPFS installation files** from [IPFS.io](https://ipfs.io) into the IPFS folder from step 5 and **run the installation file**. This assures your IPFS data storage is where you want it.

7) Open the IPFS program to the status screen and **verify peers are being connected**. Do not use the graphical user interface to try and upload your content because it can't create or update the static IPNS (InterPlanetary Name System) address you need.

7a) For Windows 7 (and possibly later) installs to support publishing. The latest Windows IPFS Desktop does not include command-line capability. For versions after Win 7, you can check this by entering the command-line "ipfs --version" (note that is a double dash before version). If there's no response, it's necessary to do the following:

In the IPFS Desktop install directory there is a hidden directory .ipfs where the IPFS repository is stored. An environment variable needs to be added, IPFS_PATH, with the complete path to the repo so that go-ipfs will find it. Restart the machine for this to activate.

Install go-ipfs for Windows from <https://docs.ipfs.io/install/command-line/#official-distributions> and be sure to add the ipfs.exe location to the system environment variable PATH. Restart again for the PATH addition to activate.

Now you should be able to enter "ipfs --version" and see the current version displayed.

8) **Open a terminal window**

9) **Create the IPNS hash which will become the static address** for your website or content file. You will not do this again, unless you have a different folder to upload which will have it's own IPNS hash address. To do this, enter the following command, replacing YourContentName (no spaces or within quotes) with what you want to appear in IPNS searches for your site:

ipfs key gen --type=rsa --size=2048 YourContentName (or "Your Folder Name")

Copy the hash created which is the public key/address to the folder containing your site or content. It will look like:

k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzl6mobpumbiruhv

Save this in a text file where it will be handy and keep the file open to add more information below. Also, write down how you copied it, if you're not familiar with doing so.

Three Simple Steps You'll Repeat Each Time You Update Your Website Or Content

Some custom script by a friend might allow this to become a single step

Note: Publish 360p versions with a modest frame rate to IPFS, unless higher resolution is really needed. Bitchute is 360p and good enough quality at only 3.3Mb/minute. Better yet, if the video only adds your face, upload much smaller mp3 audio files which can be about 1 Mb/minute. Either way, play with the recording setting to optimize the file size and quality.

10) In the terminal, **change to the directory containing the folder to be uploaded** (not the folder itself). In the open text file, **note whatever commands were necessary to get there**. This will help assure you and others can do the same each time an upload is processed.

11) In the terminal **enter the following command to upload all of the individual files to your IPFS node** using your own folder name (no spaces or within quotes):
ipfs add -r -p FolderName (or “Folder Name”)

Note: repeating this operation will not create IPFS duplicates for unchanged files

After this completes, which may take a long time initially (overall progress % is indicated on the right of each line), you’ll see the folder name which contains all of the files **near the bottom of the displayed complete results. To the left of the folder name is the folder’s IPFS hash** (the last hash displayed). Note: it will change each time you prepare for the website or folder publishing with this step. **Copy the new folder’s IPFS hash into the open text file.**

12) In the terminal, enter the following command to **publish your IPFS files into your IPNS static address** which will always be used to access your content.

ipfs name publish --key=YourIPNS_StaticHashAddress YourIPFS_Folder’sLatestHash

It will look something like

```
ipfs name publish --key=k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzl6mobpumbiruhv
QmVReNwmpLXFxBu2wttHxRneqP5VZfc6rD6pRBbY4ds9g5
```

No progress is displayed and the **cursor will return when publishing is complete** which may take a while.

Note: If publishing on a remote server, the key needs to be generated where the publish command will be executed. A private key is created there to keep others from being able to publish to the same public key.

After doing this the first time, to be included in the IPFS directory at

[https://ipfs.io/ipns/k2k4r8n098cwalcc7rdntd19nsjyzh6rku1hvgismkjzvnw582mncc4b4/
IPFS_WebsitesDirectory.html](https://ipfs.io/ipns/k2k4r8n098cwalcc7rdntd19nsjyzh6rku1hvgismkjzvnw582mncc4b4/IPFS_WebsitesDirectory.html)

email your YourIPNS_StaticHashAddress to Freedom@Agorist.Market using the subject New IPNS Site.

12a) **To see your published files listed in IPFS Desktop**, in the files menu select +Import in the upper right corner and add a New Folder (you only need to do this once per original content directory). Enter that folder, select +Import/From IPFS, enter YourIPFS_Folder’sLatestHash, select Import and the new hash should appear (if not, select Status in the upper left, then Files just below it, then the new folder you created). The pin icon to the right of the new hash verifies it’s pinned to your node. **Note:** after newer hashes are added for your content, you can select the ... to the far left of an old hash and easily **unpin older hashes, allowing newer content to use the space.**

12b) **For faster initial access to your new content**, go to another machine and access your updated content at either with ipfs.io/ipns/YourIPNS_StaticHashAddress, preferably with an IPFS Companion enabled browser as described below.

12c) **To speed up initial access to your content even more**, partner with another publisher to trade pins. Ask them to execute the following command which pins it to their node:
ipfs pin add -r --progress YourIPFS_Folder'sLatestHash
Exchanging these favors with more publishers is even better.

NOTE: If you have a LOT of older content, the ipfs pin may take hours and might not even successfully complete. If so, see steps 15-20.

Things Viewers Need To Do For Assured Access, Maybe With Help From A Friend

13) Your website or content will be at the same IPNS address after each upload using steps 10-12 above. **Using a browser, the address will be:**
`https://ipfs.io/ipns/ YourIPNS_StaticHashAddress` (add /filename.ext if needed, see note below)

Give this web address to everyone. Ask them to use it and help seed your content across the decentralized IPFS network for safe keeping.

Note: it may be necessary to add the home page file at the end of the address to display correctly.

It will look like

`https://ipns.io/ipns/k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzt6mobpumbiruhv`

or maybe something like

`https://ipns.io/ipns/k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzt6mobpumbiruhv/index.html`

14) **To assure your viewers will always have access, they need to do the following on a PC**

Caution: Brave desktop browser's IPFS integration in version 1.19 or later on a desktop, is largely dysfunctional and slow due to link handling issues. Also, it may redirect you to the dweb.link gateway which can be easily blocked at the DNS level. **The other methods of browsing with IPFS below are superior.**

For desktop browsers Firefox, Chrome, Brave Opera and Edge

a) **Install IPFS Desktop** (node) from ipfs.io on their PC. Start it and let the node run in the background to support the IPFS browser companion (see next step). Keeping the node on will help make the decentralized network stronger.

b) **Install the IPFS Companion** from docs.ipfs.io/install/ipfs-companion/ in their browser and activate it. If the IPFS Desktop node was restarted, wait for the number of peer connections to stop increasing.

c) **Browse to the IPNS address** you provided.

d) **Verify the IPFS companion is on.** If so, the address bar will display something like
`http://k2k4r8jllj4k33jxoa4vaeleqkrwu8b7tqz7tgczhptbfkhqr2i280fm.ipns.localhost:8080/index.html` (any filename at the end is also in the IPNS address from step 13)

- e) **Security settings may need to be reduced** for this mirror site to see all the content at the new IPNS address. There are no certificates to accompany your IPFS mirror site.
- f) **Bookmark the IPNS home page** for direct access.
- g) **Ask them to please visit all of the pages** in the menu bar to seed them across the network and make them more unstoppable. Visiting more pages and content in the site helps even more.
- h) **Encourage your viewers to share this instruction** and the one below with their friends to help them assure access to your site and other IPFS mirrors, even when tyrants try to interfere.
- i) The IPFS companion may cause normal web browsing to be much slower. If so, **it may be disabled when not needed**.
- j) Likewise, the IPFS Desktop may degrade site visitor's PC performance. If so, exit it and only **use it when needed to accompany the IPFS browser companion**.

An example of implementing this step is at [Agorist.Market/IPFS.html](https://agorist.market/ipfs.html) and <https://ipfs.io/ipns/k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzt6mobpumbiruhv/IPFS.html>

Preservation Of Older Content

Recent content should be seeded across the IPFS decentralized network and have quicker access as a result of visitors browsing your mirror site. Older content won't benefit from this and additional measures may be needed for assured downloads and protection from tyrants. The following steps will be necessary for very large folders of content and dedicated mirror nodes may be the best solution. Smaller amounts of older content may be adequately preserved with step 12c. The following can be done on Raspberry Pi 4 units with 4Gb of ram. If you provide these and have one yourself, the new clone drives may be sent out without the need for step 18.

- 15) **Create device to device clones of your IPFS data hard drive**. It's important that they be clones and not images, so all the data can be accessed. CloneZilla is a good program for this. Caution: write down the source drives identity and make certain you don't use it as the target and overwrite it.
- 16) **Send hard drive clones to friends** at other locations who want to protect your content.
- 17) The machine your friends' IPFS node program will be running on **needs to always stay on**, to the maximum extent possible.
- 18) Your friends will then **download the IPFS installation file** from [IPFS.io](https://ipfs.io) into the IPFS folder on the cloned hard drive and run the installation file. This assures your IPFS data storage is recognized in that location. It also allows for them to be running a different desktop or Raspberry Pi operating system.
- 19) They should then open the IPFS program to the status screen to **verify peers are being connected**. Also, after scanning the entire data directory, the storage space used will also be displayed on the status screen.

20) **If your new content is being accessed on IPFS**, it may not be necessary to preserve older content this way again. If your not sure, repeat this section occasionally.

Closing Statements

After the initial setup (steps 1-9), the three-step process (steps 10-12) to update an IPFS mirror site is very easy.

Visitor use of the IPFS mirror's static IPNS address (step 13) should be encouraged and frequently linked as the primary site location. Otherwise, the seeding may not occur, newer content will be slower to access and there will be more risk of tyrants blocking access by taking out the source node.

Visitors can only be assured of access to IPFS content by installing the IPFS Desktop on a PC (step 14). Otherwise, they will lose access any day tyrants want to delist a site from the DNS (domain name servers). It's the same as the wisdom of using the LBRY app, instead of LBRY.tv and Odysee.com which can both be easily blocked at the DNS. All the other platforms, like Flote, which don't use an app to directly access a decentralized network are all at high risk of being blocked. LBRY and IPFS are the only highly decentralized content hosts which can't be easily disabled by tyrants. The latter is the only one which can also host a website mirror, for free.

Assured preservation of older content requires a few more steps and others at different locations willing to have IPFS nodes operating with your content (steps 15-20).

Hopefully, the above will be helpful in protecting much of the content at increasing risk from tyrants today.

Republication

This document is yours to use as you see fit. Improvements will be appreciated. Credit back to [Agorist.Market](https://agorist.market) and first publication at <https://ipfs.io/ipns/k2k4r8mlcz4sn7rvh45zq2uu1b883mrs6a9hmhn4dzl6mobpumbiruhv/FreedomPosts/PublishingYourWebsiteOnIPFS.html> will be appreciated. You may contact the author at Web3Only@Agorist.Market with any improvements.