Publishing A Website On IPFS

- This presentation is posted at

- An example for websites with direct links in pages to media (mp3, mp4, etc.) files
  - Even if on a media server outside of the website.
  - Links to media platforms like Bitchute, Odysee.com and others, not ending with .mp3 or .mp4, will not allow the media to be included in the import.
Publishing A Website On IPFS

- FreeMansPerspective.com as example
  - http://ipfs.io/ipns/
    k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19
    ijjmqddumof12o3ys/
    freemansperspective.com/
- Need HTML and Linux familiarity
  - https://dev.w3.org/html5/html-author/
- Linux only for this example
• Get Website Info

• Open website and go to a page listing media like Podcast link in main menu
  • FreeMansPerspective.com/podcast/

• To see page source code, enter before url view-source:

• Search in source (ctrl-f) for media type (mp3 in this case) [screenshot]

• Note the media source(s) domain(s) (libsyn.com in this case)
On Shame

Shame is the great barrier that keeps people from acting as individuals and from expanding beyond a minimum-energy state. The expectation of shame is paralyzing to the vast majority of humans at some points of their life, and to some more frequently than others. Many people have spent their lives seeking shame-avoidance positions.

All of this is a horrific waste, and in this podcast, we take it on.

Shame, strikes deepest into the heart of man; it is felt as a sickness of the soul which leaves man naked, defeated, alienated, and lacking in dignity.
• Create directory for website with subdirectory
  • FreeMansPerspective/Subdir in this example
Prepare wget command

- Import 2-levels deep for first evaluation of site
- With desired domains to include linked page requisites.
- To bypass most cookie access restrictions, include:
  - --user-agent="Mozilla"
- In this example, to access the mp3 files, it was necessary to ignore robot exclusions with ‘-e robots=off’
  - This can cause a lot of things you don’t want to also be imported. These can be excluded from import with:
    - --exclude-domains subdomain1.domain.tld,subdomain2.domain.tld
- This example’s CLI (command line interface) command is:
  - If needed, install wget with: sudo apt install wget
  - wget command manual is at https://www.gnu.org/software/wget/manual/
Publishing A Website On IPFS

- Enter the `wget` command with the CLI in the Subdir created
  - `FreeMansPerspective/Subdir` in this example
- As it executes, you’ll see the related subdomains and domains appear in Subdir.
  - `subdomain.domains.tld`
  - This example creates:
    - `freemansperspective.com`
    - `traffic.libsyn.com`
Publishing A Website On IPFS
Publishing A Website On IPFS

Inspect each domain and subdomain folder

- In each, see if there’s anything significant which the website needs.
- If the content in a folder is all insignificant:
  - to save time, disregard it in the website mass edits.
- If folder contains a LOT of unnecessary stuff which delays processing:
  - Exclude it from the import by adding in the `wget` command ‘--exclude-domains domain-list’
    - domains in ‘domain-list’ separated by commas only
Publishing A Website On IPFS

Inspect each domain & subdomain folder

• In the folders, see if files ending in .htm are used.
  • If so, the mass edit commands will have to be duplicated for these
Publishing A Website On IPFS

- Copy all of the folders in Subdir to a new folder having the same parent as Subdir
  - 2LevelOrig in this example
  - This backup can allow bypassing the wget import when running revised mass edit commands.
Publishing A Website On IPFS

- In the Subdir folder, create a temp.sh shell (text) file to run a series of mass editing commands.
Publishing A Website On IPFS

- First in the temp.sh file enter a line to find all http:// instances in the site’s HTML files and change to https:// as follows:
  - `find . -type f -name "*.html" -exec sed -i 's|http://|https://|g' {} \;
  - `echo “comment” lines are included which may be helpful with debugging
- If .htm files are also used, add another command line as follows:
  - `find . -type f -name "*.htm" -exec sed -i 's|http://|https://|g' {} \;
  - `find command info is at https://www.gnu.org/software/findutils/
- If needed, install with: `sudo apt install find`
Publishing A Website On IPFS

• sed command manual is at https://www.gnu.org/software/sed/manual/
  • also https://linuxhandbook.com/sed-reference-guide/

• If needed, install with:
  • sudo apt-get install sed

• Note that sed uses ‘regular expressions’ in the search (regex) text which has more info at
• The only folders created under Subdir in this example were freemansperspective.com and traffic.libsyn.com

• Replace absolute links to be relative with commands:
  • `find . -type f -name "*.html" -exec sed -i 's|https://freemansperspective.com|/freemansperspective.com|g' {} \;`
  • `find . -type f -name "*.html" -exec sed -i 's|https://traffic.libsyn.com|/traffic.libsyn.com|g' {} \;`

• If *.htm files are in the site, these commands will need duplicating as described for the earlier find command.
Publishing A Website On IPFS

```bash
1 echo "find and replace http:// with https://"
2 find . -type f -name "*.html" -exec sed -i 's|http://|https://|g' {} \
3 echo "find and replace https://freemansperspective.com with /freemansperspective.com"
4 find . -type f -name "*.html" -exec sed -i 's|https://freemansperspective.com|/freemansperspective.com|g' {} \
5 echo "find and replace https://traffic.libsyr.com with /traffic.libsyr.com"
6 find . -type f -name "*.html" -exec sed -i 's|https://traffic.libsyr.com|/traffic.libsyr.com|g' {} \
```
Publishing A Website On IPFS

Change the temp.sh file Properties/Permissions to allow execution
Publishing A Website On IPFS

- Create empty ipfs_hash_log.txt file in Subdir
  - FreeMansPerspective/Subdir in this example
- With CLI in the Subdir, execute temp.sh with command:
  - ./temp.sh
Publishing A Website On IPFS

- Change the CLI directory to the parent of Subdir with
  - cd ..
- Add Subdir with all its contents to your IPFS node with the following command:
  - ipfs add -r -p Subdir
- When the add command ends, copy the last hash displayed to the left of Subdir
Publishing A Website On IPFS
Publishing A Website On IPFS

- Paste/append the Subdir hash into ipfs_hash_log.txt
  - This is to keep track of hashes created which can be unpinned later
Publishing A Website On IPFS

Alternatively, you can
1) open the IPFS GUI,
2) select FILES,
3) then select +Import in the upper right corner,
4) select New Folder in the drop-down,
5) enter a folder name,
6) select the new folder,
7) select +Import,
8) select From IPFS in the drop-down,
9) paste the copied IPFS hash for Subdir in the entry field,
10) after the hash appears in the folder, select … to the right of the hash,
11) then select unpin
12) It may be necessary to backup a level and re-enter the folder for the pin status to appear correctly.
Publishing A Website On IPFS
Publishing A Website On IPFS
Publishing A Website On IPFS

- Open your browser with IPFS Companion installed
- Verify IPFS is active with the blue icon in the upper right having a number of peers indicated
Publishing A Website On IPFS

- Disconnect your internet connection to verify the site on IPFS works without it.
- Enter website on your IPFS node by pasting the IPFS_hash in the url address:
  - https://ipfs.io/ipfs/IPFS_hash
  - Note: url will change when page opens
  - Subdir folder contents will appear
Publishing A Website On IPFS

Index of /ipfs/bafybeigiovmc7gslcofgtjgrwqpp5wa4fhd7t2xdncmd4zjq4qnggh5kcb2e

<table>
<thead>
<tr>
<th>File</th>
<th>Hash</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>..</td>
<td></td>
<td></td>
</tr>
<tr>
<td>freemansperspective.com</td>
<td>QmWf...CX7G</td>
<td>26 MB</td>
</tr>
<tr>
<td>temp.sh</td>
<td>Qma4...62t4</td>
<td>493 B</td>
</tr>
<tr>
<td>traffic.libsyn.com</td>
<td>Qmbx...o9Gd</td>
<td>578 MB</td>
</tr>
</tbody>
</table>

mp3
Publishing A Website On IPFS

- Select primary domain in directory to view site
  - freemansperspective.com in this example
- Next is the tricky part requiring HTML knowledge
- Investigate the functionality of pages in site,
  - especially menus, page continuations and next page links.
  - Look in the source code to see if there are clues about additional mass editing commands needed.
- Keep in mind this import was only 2-levels deep.
  - You may run into broken links as you go deeper which will be ok with more levels
Publishing A Website On IPFS

- More of the tricky part
- Execute any new find/sed commands necessary to resolve significant issues
  - Determined on case by case basis
- After mistakes, copy the originals over the changed directories in Subdir
  - Eliminates need to repeat wget command
  - From 2LevelOrig in this example (ref. slide 12)
Publishing A Website On IPFS

- Repeat the import and find/sed commands going 3 levels deep
  - After all find/sed commands are tested and validated for level 2
- Wget command can be added at start of the temp.sh file
  - Along with any new find/sed commands found to be necessary.
- For Wordpress sites, delete all of the old *.html files, to not accumulate
  - Instead of deleting the entire directories
  - Avoids again downloading media files and other things which didn’t change
  - Find *.html delete command is added in this example before the wget command
  - Items with a newer remote timestamp will overwrite the old files.
  - This is only needed for dynamic websites like Wordpress, Joomla and Drupal.
- At the end, the ipfs add command can also be added, following cd ..
- Then, to return to the original folder include:
  - Cd Subdir
Publishing A Website On IPFS

1 echo "delete old *.html files"
2 find . -type f -name "*.html" -delete
3 # Import website infinite levels deep, bypassing robot exclusion to get media files from
   # libsyn.com and with --user-agent="Mozilla" to bypass cookie restrictions
4 wget --user-agent="Mozilla" -e robots=off --convert-links --recursive --level=3 --page-
   requisites --adjust-extension --timestamping --restrict-file-names=windows --no-parent -
5 echo "find and replace http:// with https://"
6 find . -type f -name "*.html" -exec sed -i 's|http://|https://|g' {} \
7 echo "find and replace https://freemansperspective.com with /freemansperspective.com"
8 find . -type f -name "*.html" -exec sed -i 's|https://freemansperspective.com/|
   freemansperspective.com|g' {} \
9 echo "find and replace https://traffic.libsyn.com with /traffic.libsyn.com"
10 find . -type f -name "*.html" -exec sed -i 's|https://traffic.libsyn.com/|
   traffic.libsyn.com|g' {} \
11 # Add to IPFS
12 cd ..
13 ipfs add -r -p Subdir
14 cd Subdir
15 |
Publishing A Website On IPFS

- Change to Subdir and execute ./temp.sh
- Repeat all appropriate steps above
  - especially copying the new hash for Subdir
Preservation of content quickly is the 1st goal
  • Perfection is difficult to achieve on IPFS
  • If it’s better than alternatives, publish and improve later

When satisfied the program is good enough, create an IPNS static address for the website with a command similar to the following example:
  • `ipfs key gen --type=rsa --size=2048 FreeMansPerspective.com`

The new k2k... hash displayed is the IPNS address for the website which we will publish to.
  • Copy this and save it.
Publishing A Website On IPFS

dev@dev:~$ ipfs key gen --type=rsa --size=2048 FreeMansPerspective.com
k2k4r8opjzlvy6tcsmpl80bigim0tty4f9dm1y19iijmqdumof12o3ys
dev@dev:~$
Publishing A Website On IPFS

• An easy way to save it is assigning it as a variable at the beginning of the program with a command like:
  
  • IPNS_hash=k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19ijjmqddumof12o3ys

• The website folder address will be:
  
  • ipfs.io/ipns/
    k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19ijjmqddumof12o3ys

• The website home page address will have the domain added at the end, followed by /
  
  • ipfs.io/ipns/
    k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19ijjmqddumof12o3ys/freemansperspective.com/
Publishing A Website On IPFS

- Change the levels in the `wget` command to 0 or `inf` (infinite) to import the entire site.
- We can automate assigning a variable the new Subdir IPFS hash with:
  - `IPFS_hash=$(ipfs add -r -Q Subdir)`
- The hash can then be automatically appended to the `ipfs_hash_log.txt` file with this command:
  - `echo $IPFS_hash >> ipfs_hash_log.txt`
- With variables assigned to the IPNS and IPFS hashes, the publish to IPNS command becomes:
  - `ipfs name publish --key=$IPNS_hash $IPFS_hash`
Publishing A Website On IPFS

```
1 echo "assign variable for IPNS hash"
2 IPNS_hash=k2k4r8opjzlv6tcsmp80bigim0tty4f9mdml1y19ijmqddumofl2o3ys
3 echo "delete old *.html files"
4 find . -type f -name "*.html" -delete
5 # Import website infinite levels deep, bypassing robot exclusion to get media files from
   libsyn.com and with --user-agent="Mozilla" to bypass cookie restrictions
7 echo "find and replace http:// with https://"
8 find . -type f -name "*.html" -exec sed -i 's/http://https://g' {} \
9 echo "find and replace https://freemansperspective.com with /freemansperspective.com/
10 find . -type f -name "*.html" -exec sed -i 's=https://freemansperspective.com//freemansperspective.com/g' {} \
11 echo "find and replace https://traffic.libsyn.com with /traffic.libsyn.com"
12 find . -type f -name "*.html" -exec sed -i 's=https://traffic.libsyn.com//traffic.libsyn.com/g' {} \
13 # Add to IPFS and assign hash to variable
14 cd ..
15 IPFS_hash=$(ipfs add -r -Q Subdir)
16 cd Subdir
17 # append to ipfs_hash_log.txt
18 echo $IPFS_hash >> ipfs_hash_log.txt
19 echo "publishing $IPFS_hash to IPNS"
20 ipfs name publish --key=$IPNS_hash $IPFS_hash
21 echo "freemansperspective.com has completed publishing to IPNS"
```
Publishing A Website On IPFS

- Execute ./temp.sh in Subdir again
- After completing the CLI displays:

```bash
Converting links in freemansperspective.com/wp-includes/css/dashicons.min.css@ver=5.6.3.css... 3-0
Converting links in freemansperspective.com/wp-content/plugins/mad-mimi-wp/assets/vendor/tooltipster/tooltipster.bundle.min.css@ver=2.5.1.css... nothing to do.
Converting links in freemansperspective.com/wp-content/themes/twentysixteen-fmp-child/style.css@ver=20190507.css... 1-0
Converting links in freemansperspective.com/wp-admin/css/l10n.min.css@ver=5.6.3.css... nothing to do.
Converting links in freemansperspective.com/wp-admin/css/login.min.css@ver=5.6.3.css... 2-0
Converting links in freemansperspective.com/wp-includes/js/mediaelement/wp-mediaelement.min.css@ver=5.6.3.css... nothing to do.
Converting links in freemansperspective.com/wp-content/plugins/mad-mimi-wp/assets/vendor/featherlight/release/featherlight.min.css@ver=2.5.1.css... nothing to do.
Converted links in 6748 files in 9.3 seconds.
find and replace http:// with https://
find and replace https://freemansperspective.com with /freemansperspective.com
find and replace https://traffic.libsyn.com with /traffic.libsyn.com
publishing Qme4vB11aQ146P2paY6EcKm8Cby7L1PPEAPeTeuuvuPaTS to IPNS
Published to k2k4r8opjzlv6tcmmp80bigim0tty4f9dm1y19ijmqddumof12o3ys: /ipfs/Qme4vB11aQ146P2paY6EcKm8Cby7L1PPEAPeTeuuvuPaTS
freemansperspective.com has completed publishing to IPNS
dev@dev:/media/dev/2nd500/sites/FreMansPerspective/Subdir$ 1
```
Publishing A Website On IPFS

• Essentials of the website should function without internet connectivity at:
  • ipfs.io/ipns/
    k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19ijjmqddum
    of12o3ys/freemansperspective.com/

• Again, preservation of content ASAP is the 1st goal
  • Perfection is difficult to achieve on IPFS
  • If it’s better than alternatives, publish and improve later

• See the last page of this presentation about listing your site in the IPFS Website Directory
After this, you should keep your IPFS node online as much as possible, or at least until you know others have pinned the entire site.

You may want to rename the temp.sh file and add more internal documentation at this point, like:

- FreeMansPerspective.sh available at https://ipfs.io/ipns/k2k4r8n098cwalcc7rdntd19nsjyzh6rku1hvgsmkjzvnw582mncc4b4/Shells/FreeMansPerspective.sh
Publishing A Website On IPFS

• This True Story’s Outcome Is At
  • http://ipfs.io/ipns/k2k4r8opjzlv6tcsmp80bigim0tty4f9mdm1y19ijjmqddumof12o3ys/freemansperspective.com/

• Also listed in the IPFS Websites Directory at
  • https://ipfs.io/ipns/k2k4r8n098cwalcc7rdntd19nsjyzh6rku1hvgsmkjzvnw582mncc4b4/IPFS_WebsitesDirectory.html

• Which can be found at
  https://agorist.market/IPFS.html
Publishing A Website On IPFS

- This presentation is posted at

- Contact Mike Swatek at:
  - Telegram: @ppmSilver
  - Email: Web3Only@Agorist.Market
  - Send your new website IPNS addresses to be included in the IPFS Websites Directory at
    - https://ipfs.io/ipns/k2k4r8n098cwalcc7rdntd19nsjyzh6rku1hvgsmkzvnnw582mncc4b4/IPFS_WebsitesDirectory.html
    - Also linked from https://agorist.market/IPFS.html