Crawling records on the InterPlanetary Name System

By Axel Gard

https://github.com/AxelGard/seek-ipns/tree/master/doc

ipfs://QmWRkWcdYcnTvG856xptCvQ3aLcNp1sPApbwGePWq8Wgdp

Road map

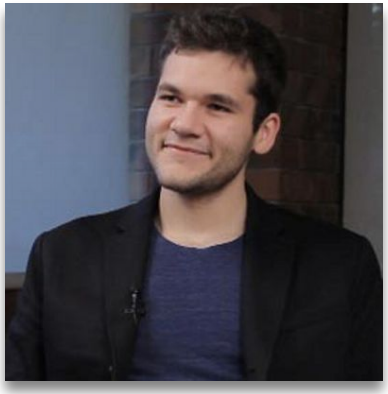
- IPFS
- IPNS
- The system
- Results

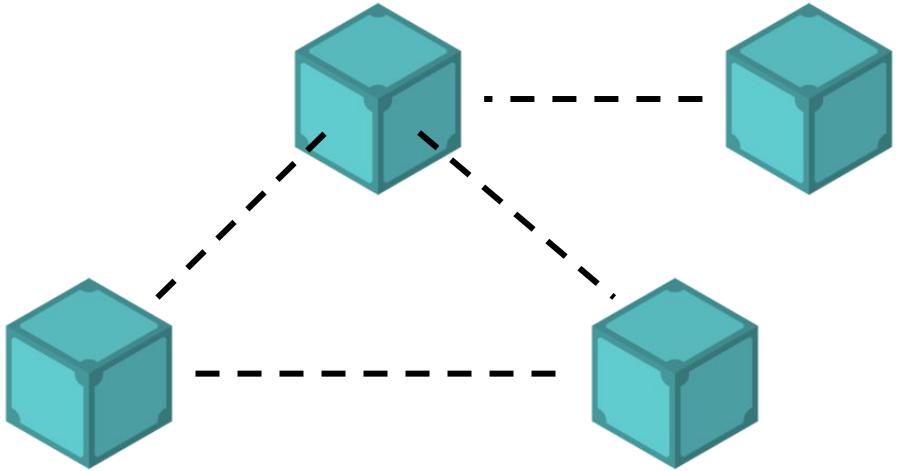
How do you host content on the internet?

InterPlanetary File System (**IPFS**)

IPFS

Created by Juan Benet, in 2015





Node / Peer

Peer ID

Private Key

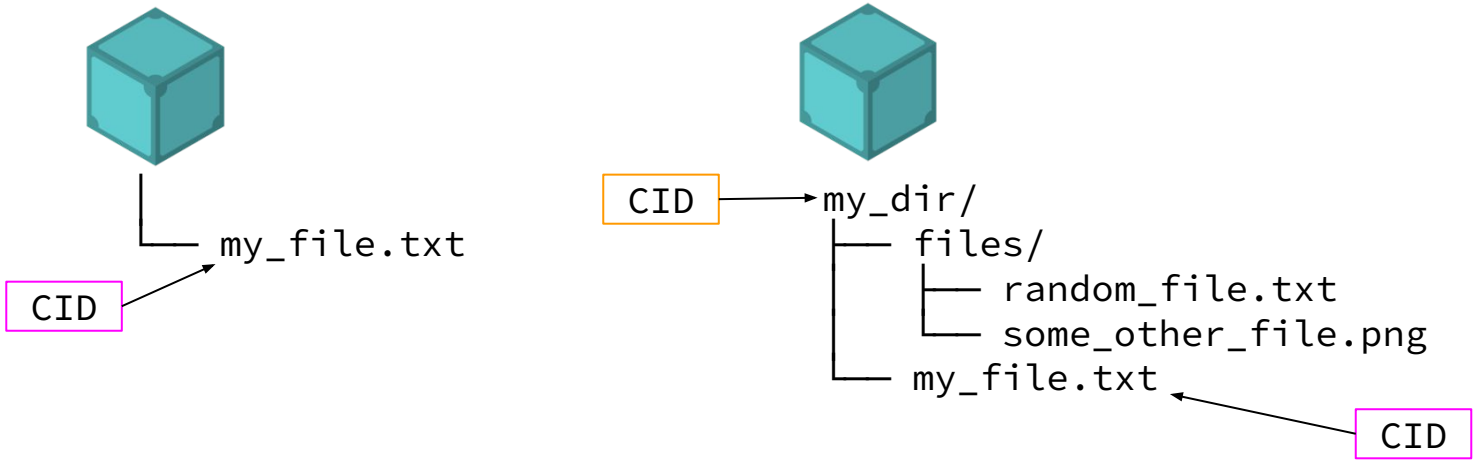
Public key

Files

Hash functions

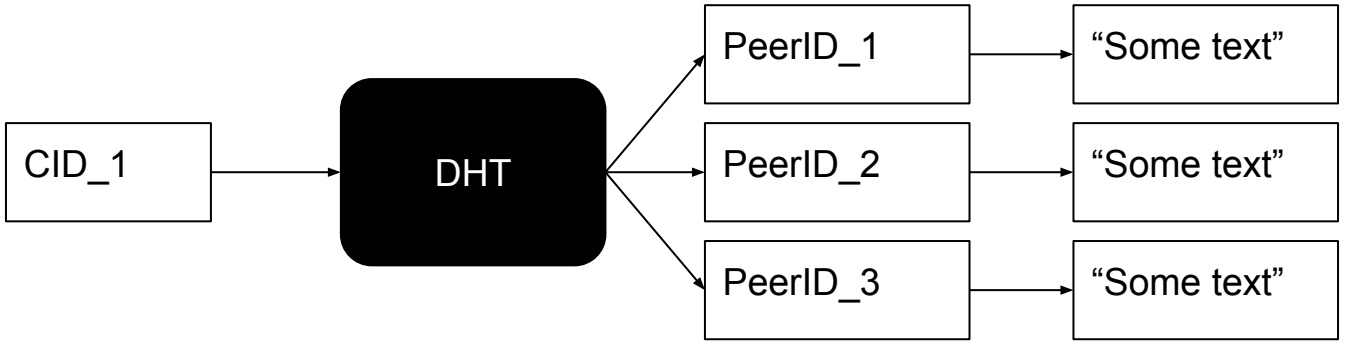
Content Identifier (CID)

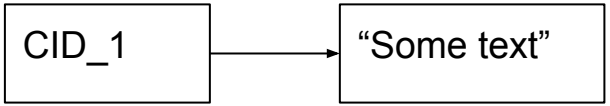
CID exemple: “” ⇒ QmUNLLsPACCz1vLxQVkXqqLX5R1X345qqfHbsf67hvA3Nn



Distributed hash tables (DHT)

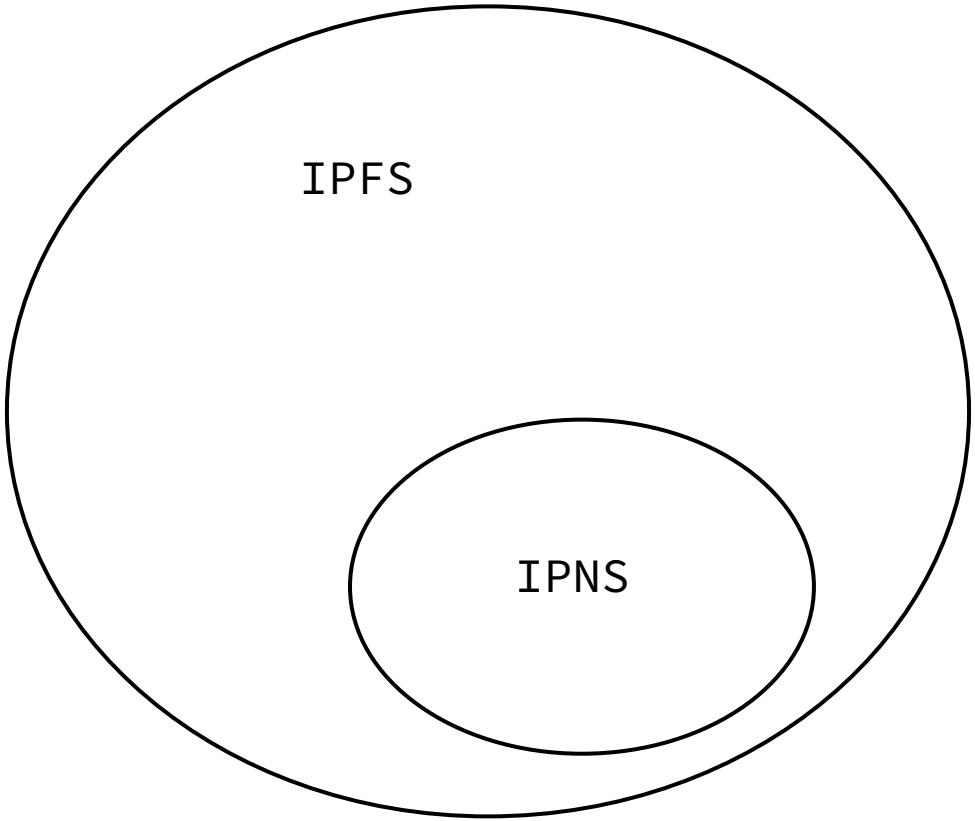
IPFS DHT uses Kademlia



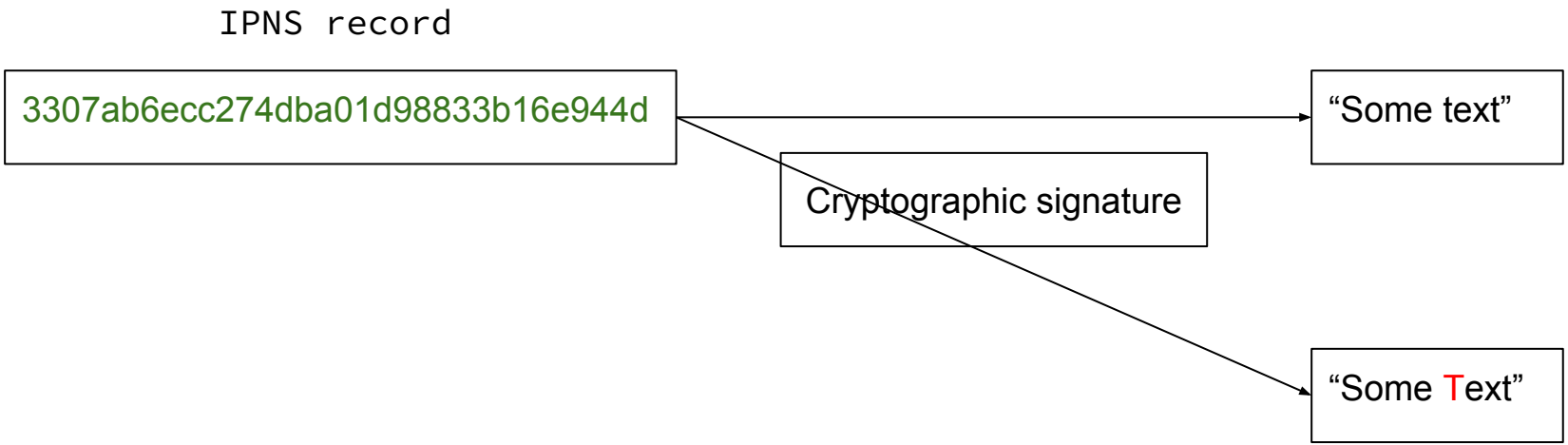


Content can be found
on the network if the
unique CID is known
for that content.

InterPlanetary Name System (**IPNS**)


IPFS
IPNS

IPNS

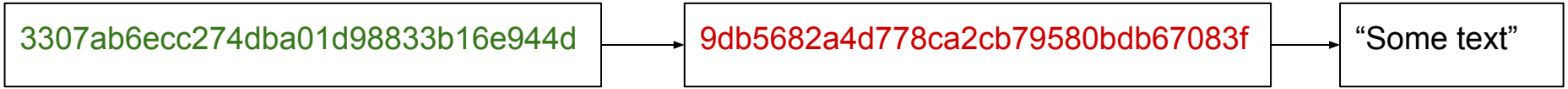

IPNS record
3307ab6ecc274dba01d98833b16e944d
“Some text”
Cryptographic signature
“Some Text”

Hash functions

IPFS


9db5682a4d778ca2cb79580bdb67083f
“Some text”

IPNS


3307ab6ecc274dba01d98833b16e944d
9db5682a4d778ca2cb79580bdb67083f
“Some text”


Cryptographic signature
3307ab6ecc274dba01d98833b16e944d


3307ab6ecc274dba01d98833b16e944d
56e57920c3c8c727bfe7a5288cdf61c4
“Some Text”

IPFS


9db5682a4d778ca2cb79580bdb67083f
“Some text”
56e57920c3c8c727bfe7a5288cdf61c4
“Some Text”

IPNS


3307ab6ecc274dba01d98833b16e944d
“Some text”
“Some Text”

IPNS **allows** for content to **change** but still a **static link** to the content

How do we find CIDs or records?

You need to know the CID, or the IPNS record.

But how do we find them?

We need a search engine that lets us search for content and gives us IPNS records.

So we need to be able to **search** and **find IPNS records** so that we can find **content**

The **content** on IPNS can become **available** to others and make the network much **more useful**

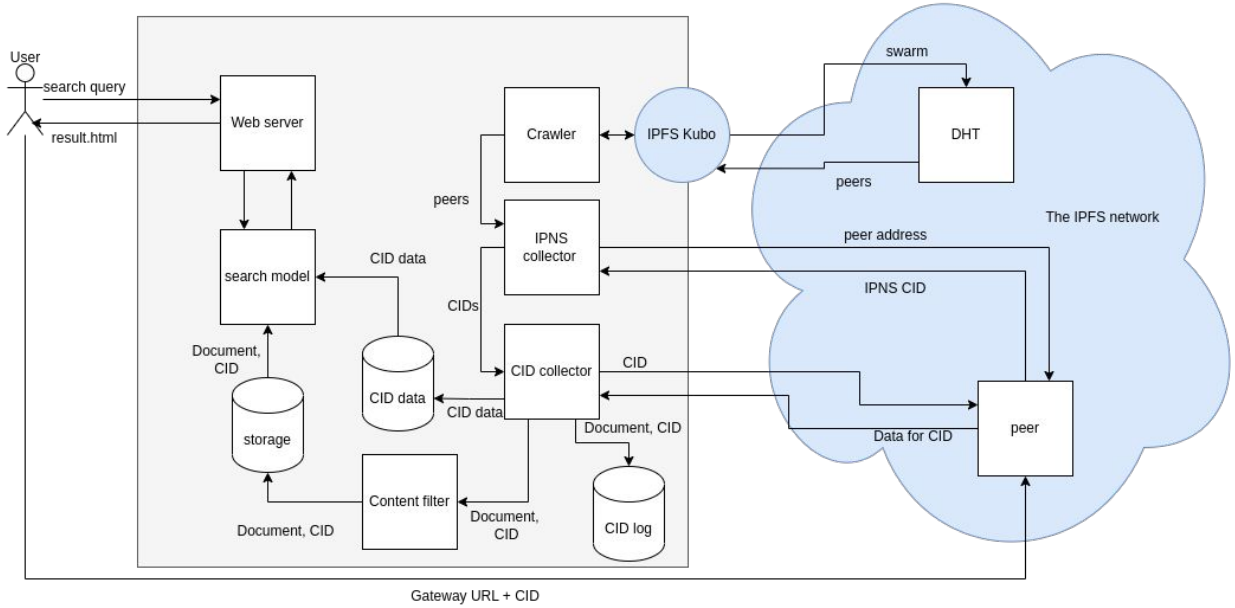
Research questions

1. How does the rate of change in default IPNS records of peers on the IPFS network impact the reliability of an synchronized indexing of CIDs over time?

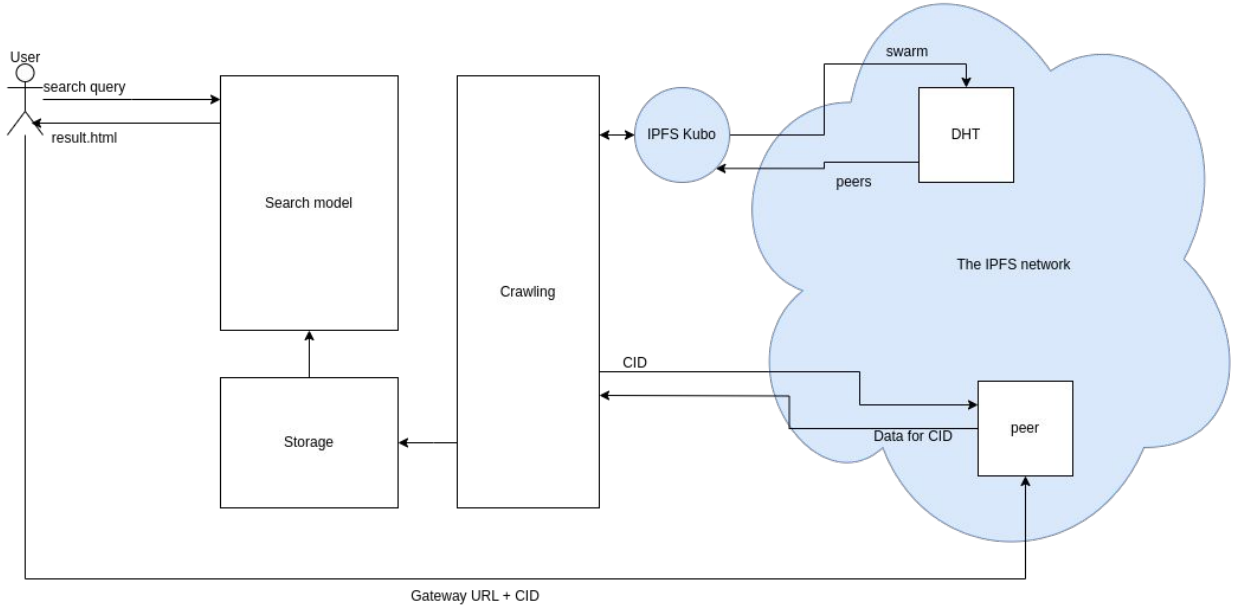
2. What file formats and data types should be prioritized for information extraction compatibility with the default IPNS records to ensure effective integration into a search engine system?

The system, seekIPNS

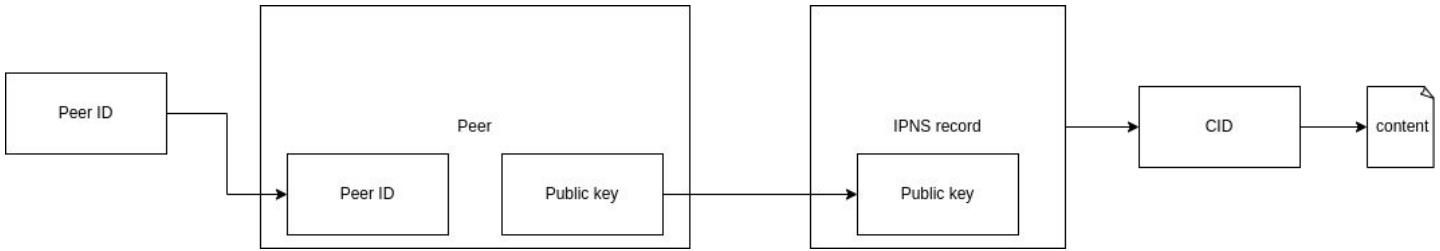
System architecture



System architecture, simplified

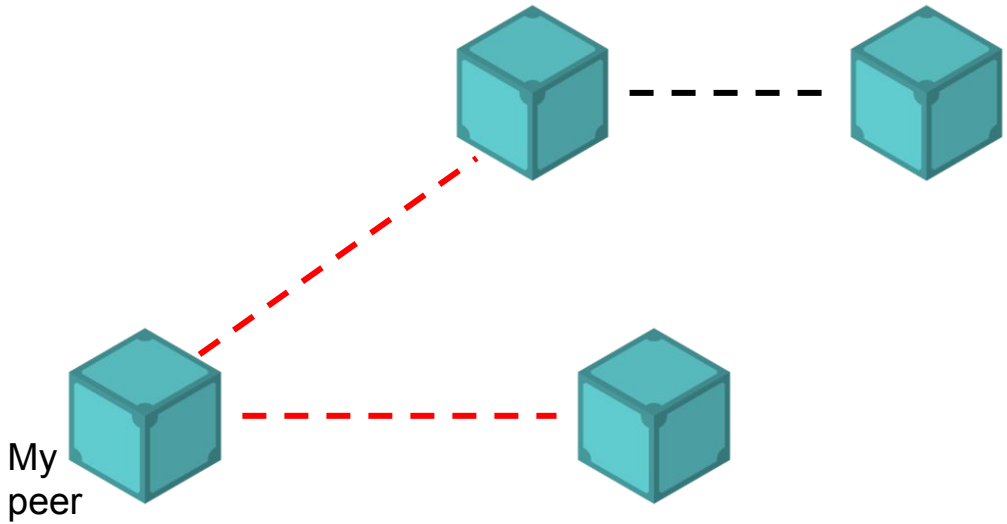


Peer ID to CID



How do we find peers

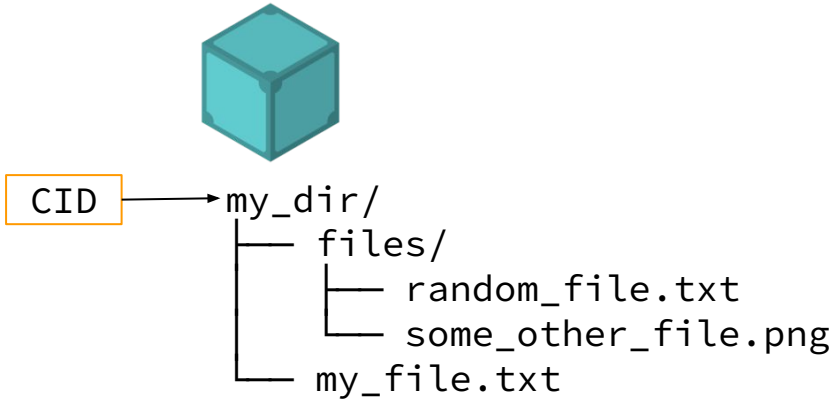
Iterate over the open connections using IPFS swarm command.



Get all open connections that the host node has.

Crawling a peer

- Peer ID
- CID
- All file names
- All file path
- All file sizes
- All file formats
- Number of files
- Number of peers hosting the CID
- Time when peer was found

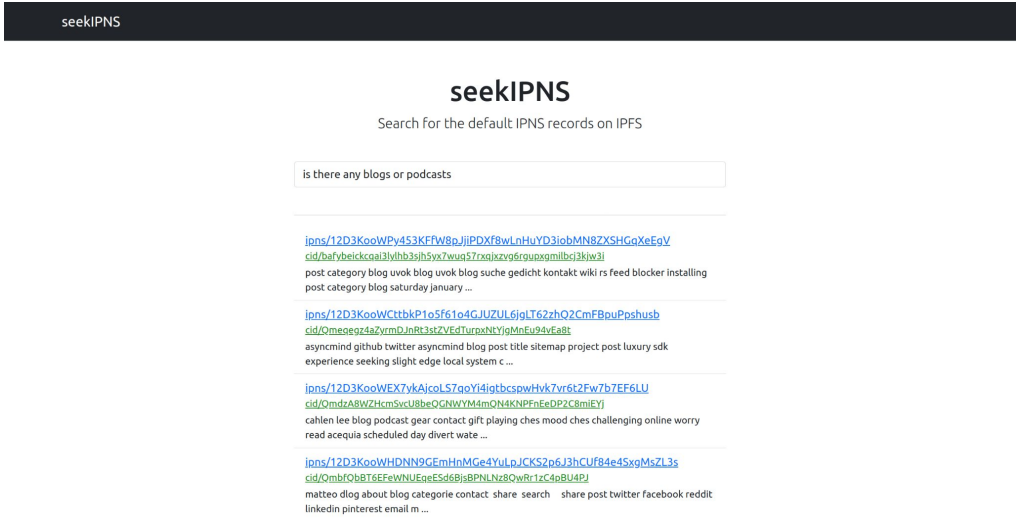


Some files found on the peer were saved if they are needed for the search model.

Search model

term frequency–inverse document frequency (TF–IDF)

Weights of words in each document, which can be then accumulated for each query word.



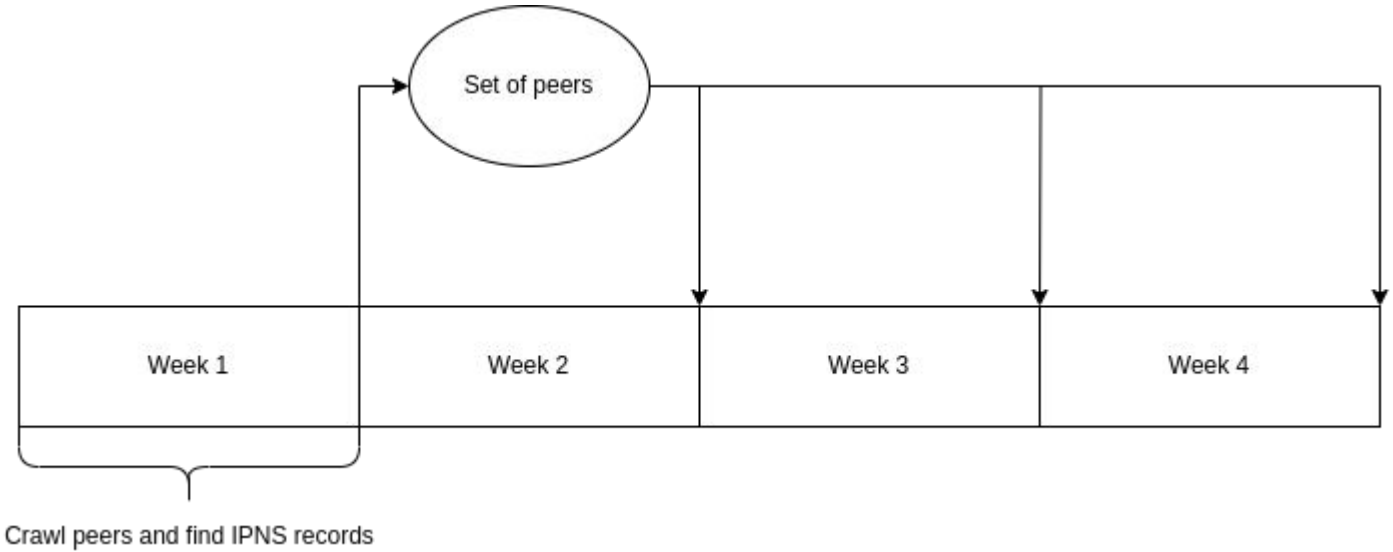
Exploring the data on the IPNS records

Research questions

1. How does the **rate of change** in default IPNS records of peers on the IPFS network impact the reliability of an synchronized indexing of CIDs over time?

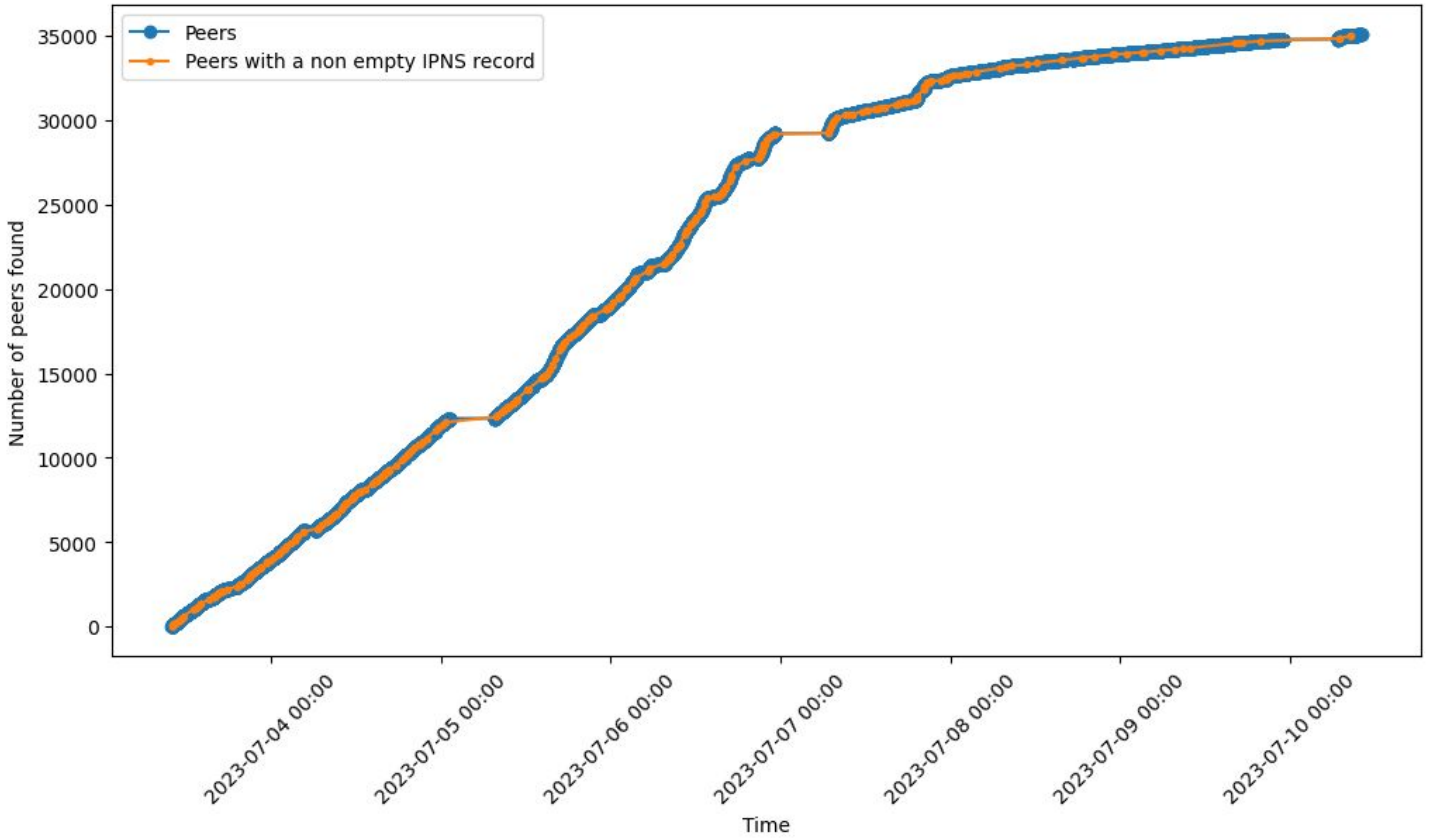
2. What **file formats and data types** should be prioritized for information extraction compatibility with the default IPNS records to ensure effective integration into a search engine system?

Rate of change experiment



Results

Crawling


Peers
Peers with a non empty IPNS record
Number of peers found
35000
30000
25000
20000
15000
10000
5000
0
2023-07-04 00:00
2023-07-05 00:00
2023-07-06 00:00
2023-07-07 00:00
2023-07-08 00:00
2023-07-09 00:00
2023-07-10 00:00
Time

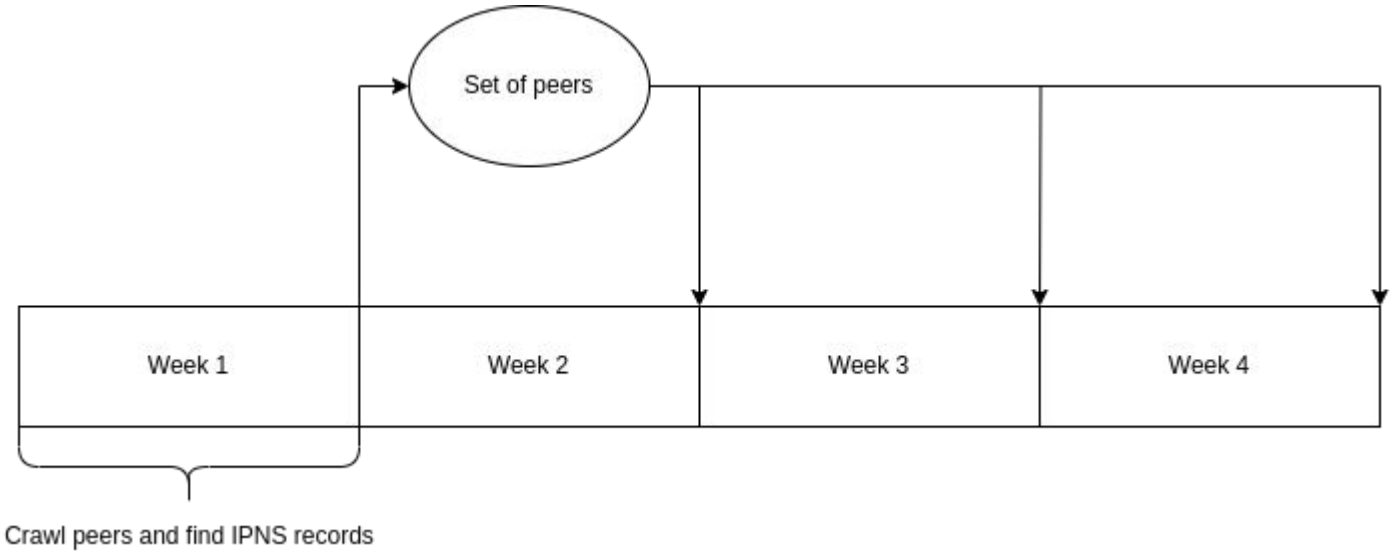
Data

- 35K unique peers were found
- 333 peers had a non-empty IPNS record
- 82K files were found
- 175.3 GB of files

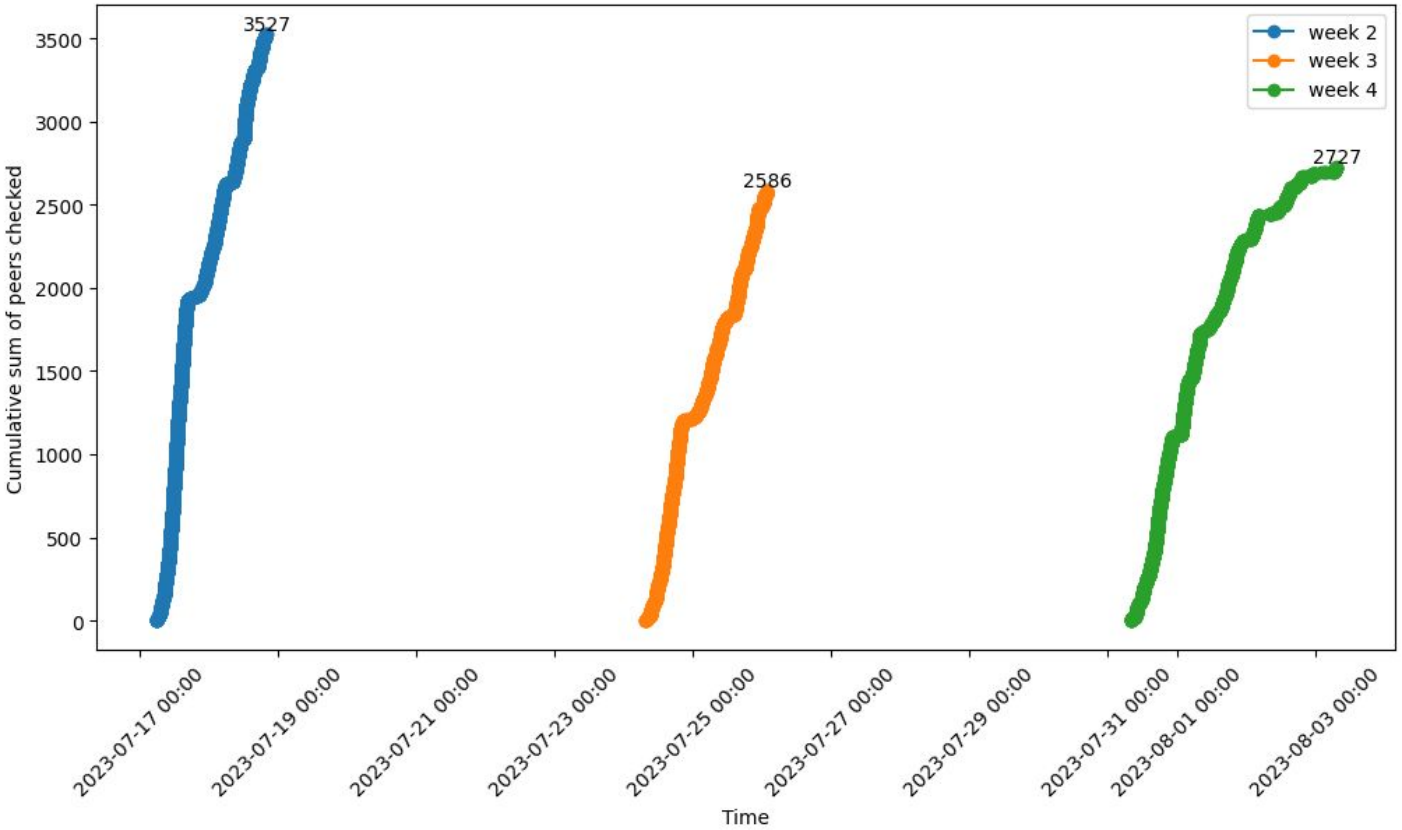
“Related work” by Erik Daniel and Florian Tschorsch found the network size to be **40-60K peer size** in **2022**

A significant amount of the **network was crawled**

Revisiting


Set of peers
Week 1
Week 2
Week 3
Week 4
Crawl peers and find IPNS records

Revisiting peer



Rate of change

Week	New CIDs
0	0
1	234
2	98
3	114

Total: 446 peers changed their content
In the first week, there were 333 non-empty records

The **content** on the peers **changes** a lot over time

Evaluating the content on the records

Formats

My_file.txt

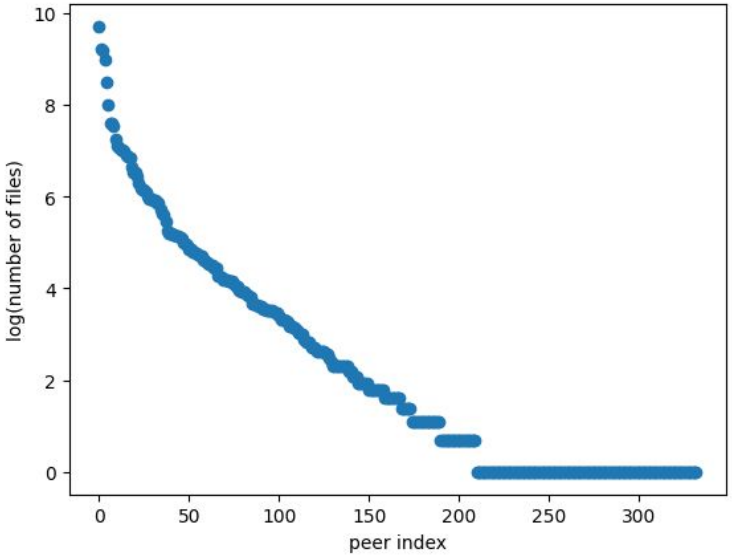
My_file.txt ⇒ .txt

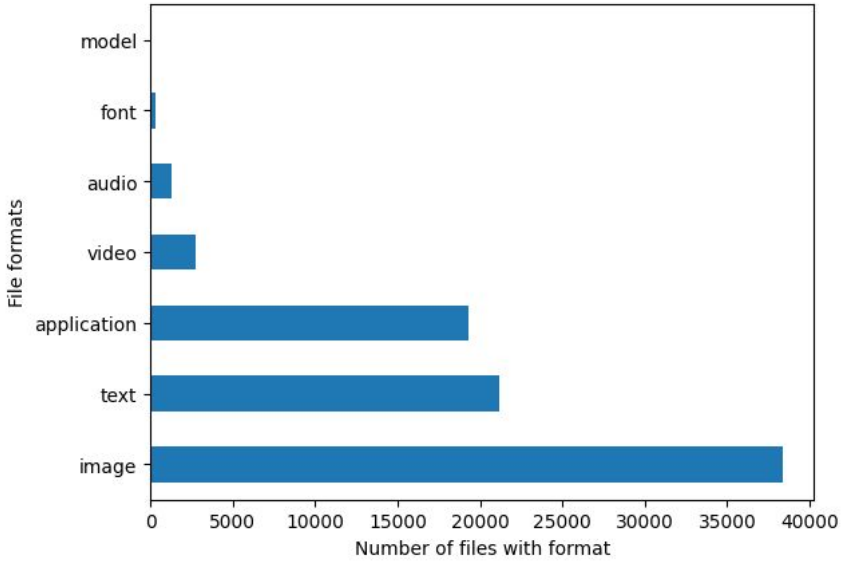
My_file.txt ⇒ text/plain (go package mimetype)
text/plain ⇒ text

image/jpeg, image/png are put into the image category

Application, Audio, Font, Image, Model, Text, and Video

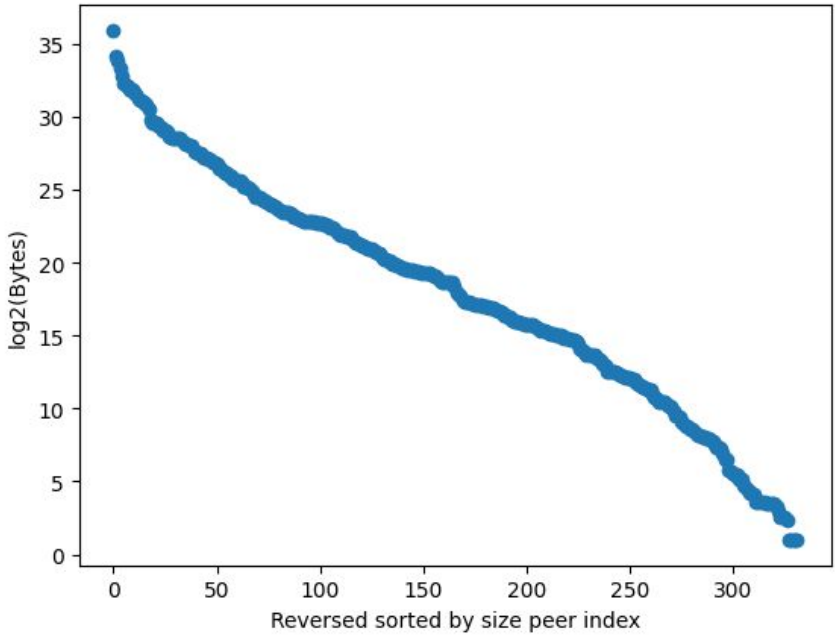
Number of files

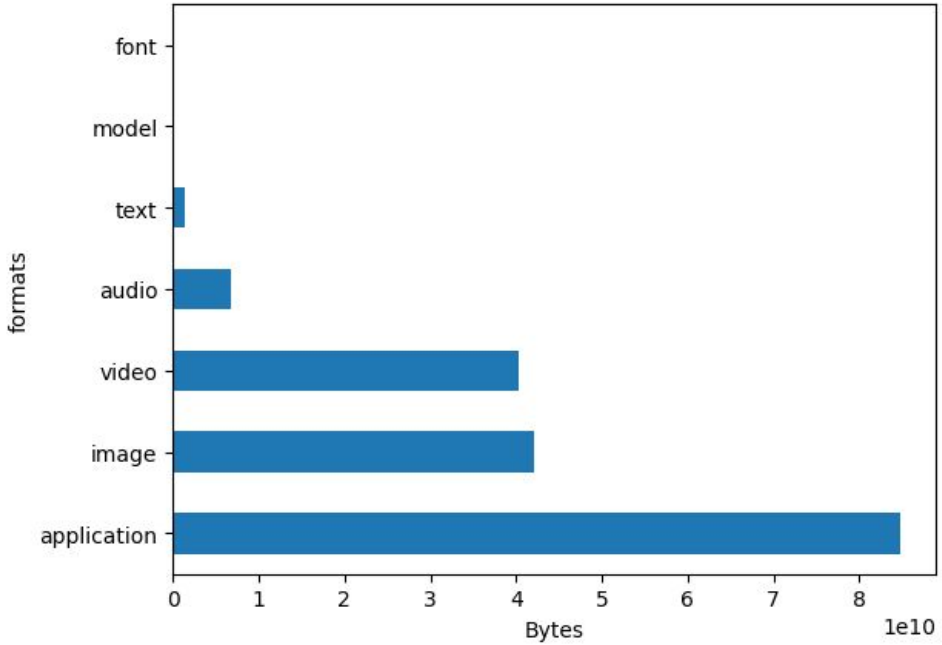




Mean: 249.5 files per peer

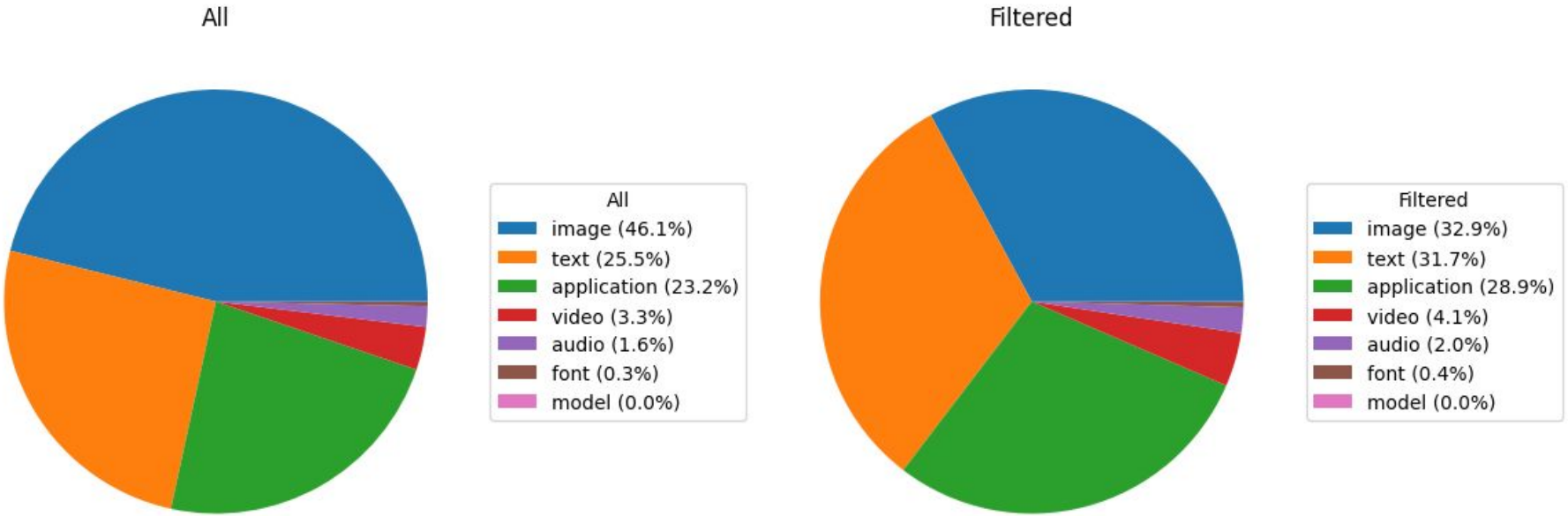
File sizes





Mean: 0.26 GB

Dominated metrics

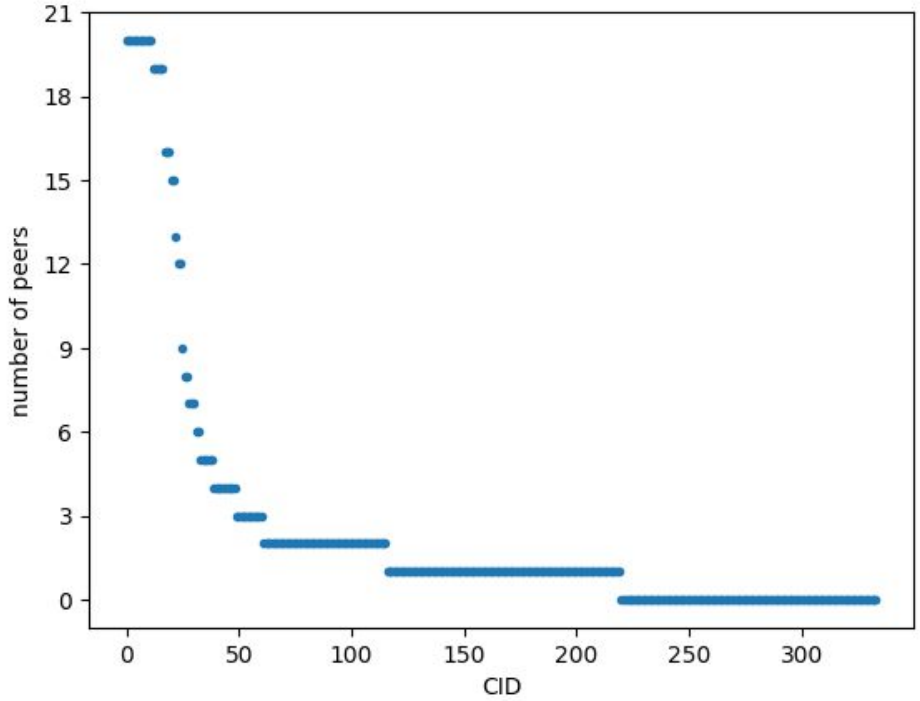


Number of files with format, removing the peer with the most number of files.

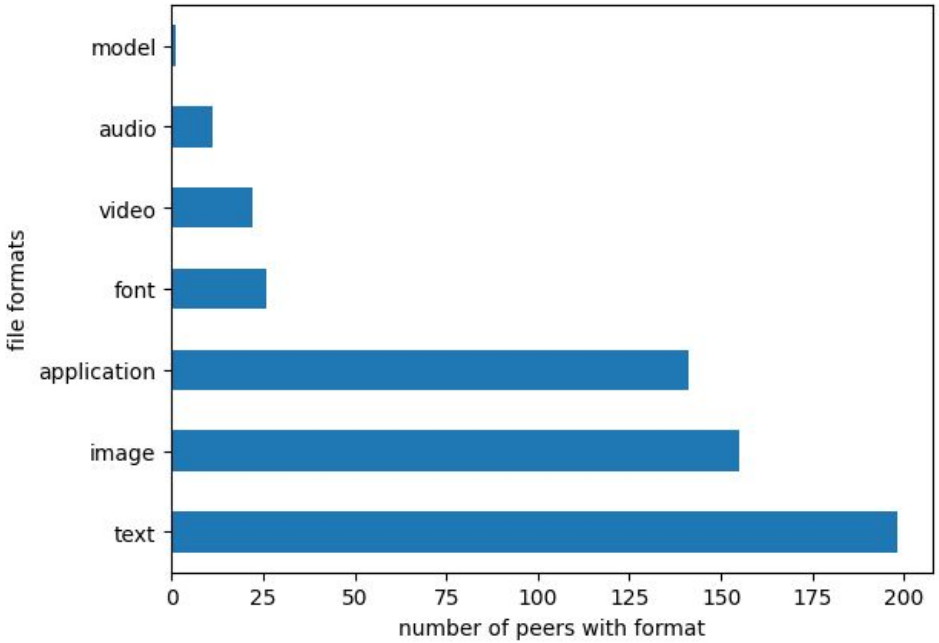
Decentralization of records

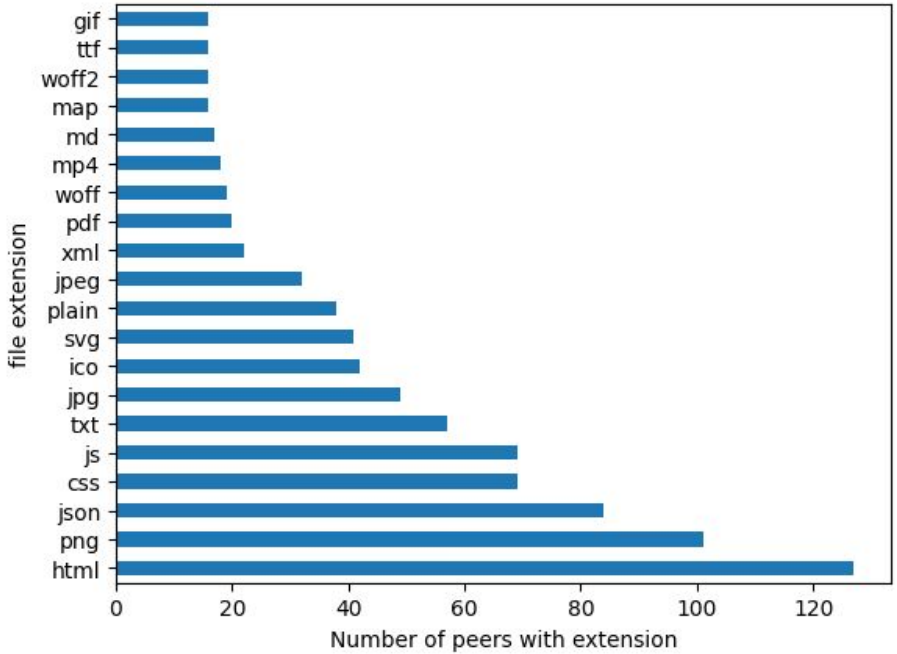
Mean: 2.47

0 number of peers hosting, due to DHT look up taking too long.

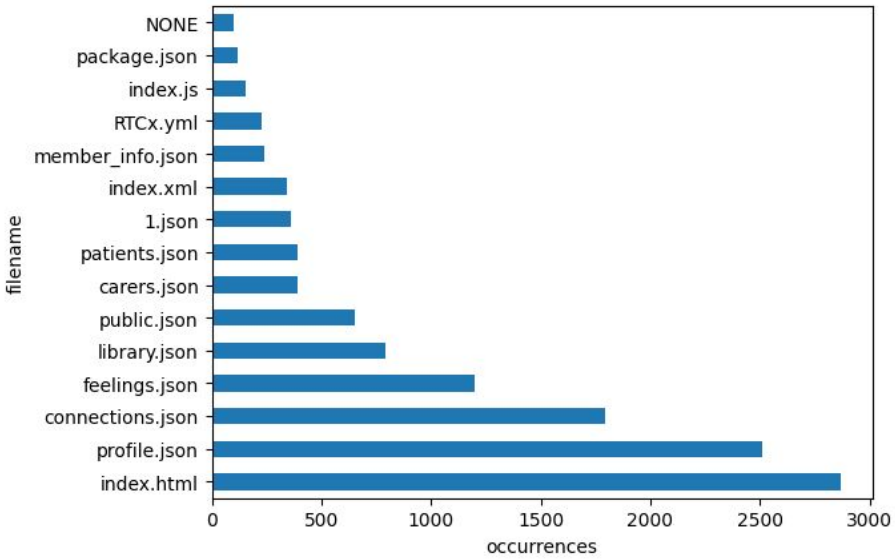


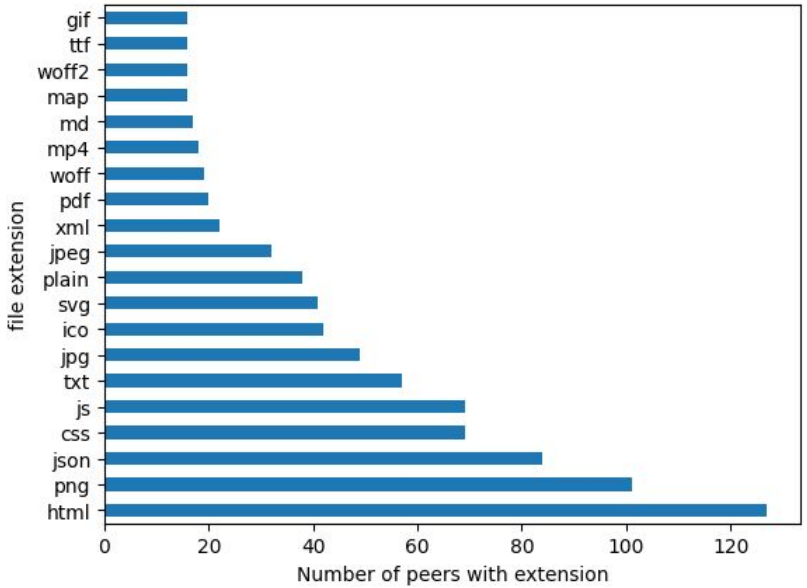
Distribution among peers





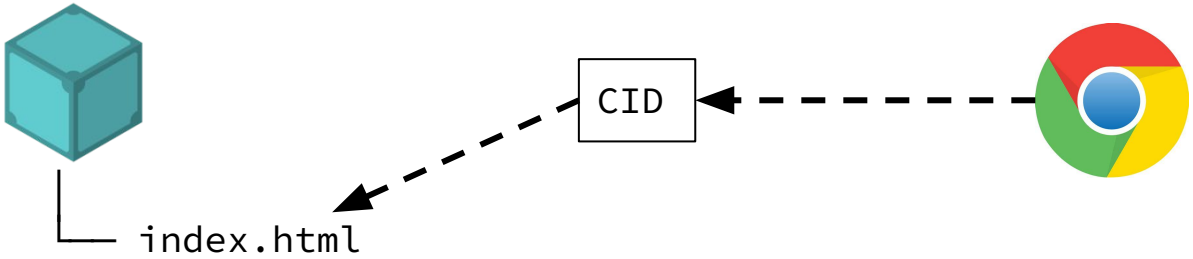
Number of times a file name was found

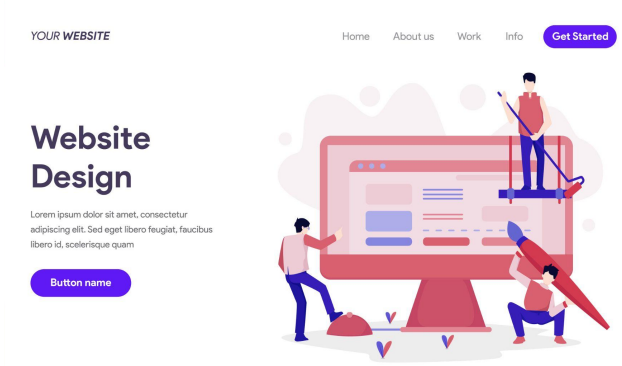




The most **common** use case for the **IPNS** records have files relate to **websites.**

Why do websites make sense?





(static website)

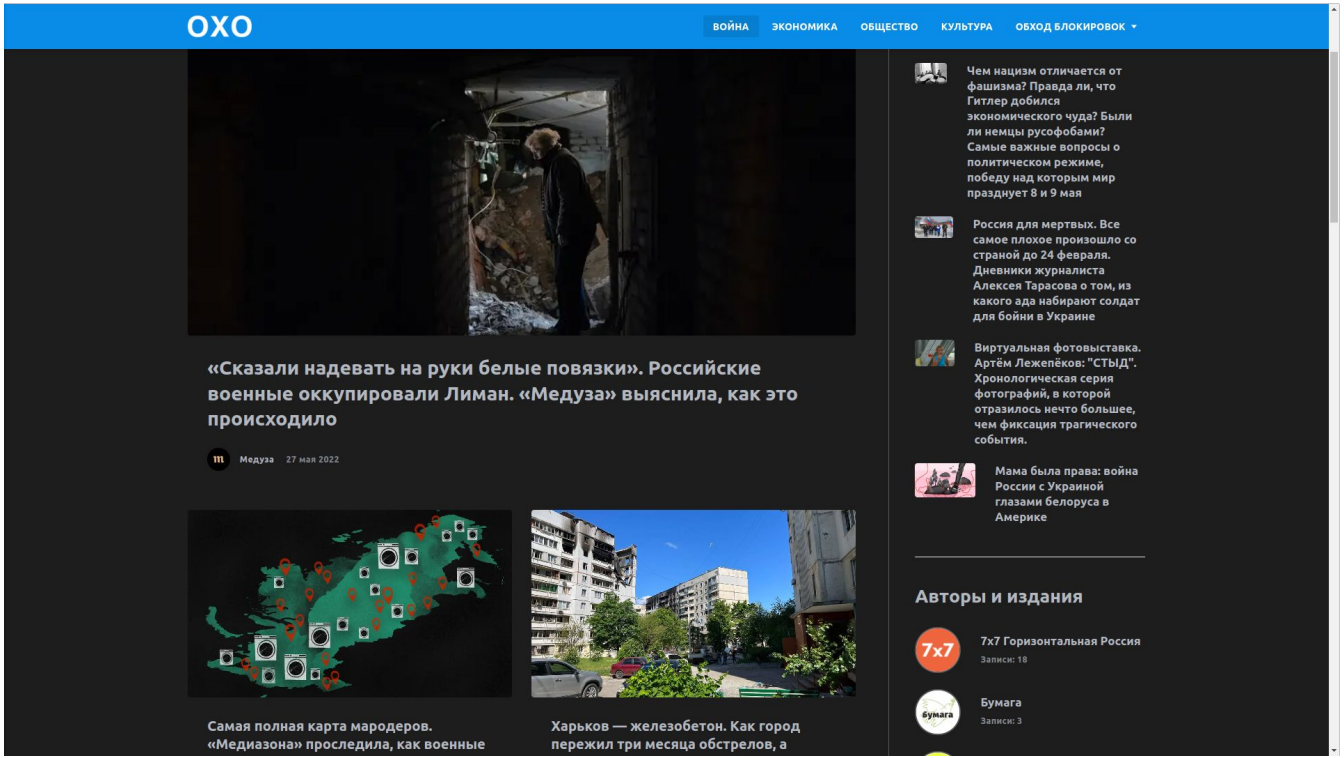
Some **examples of content**
found on the on some records

Crypto Hobos NFT

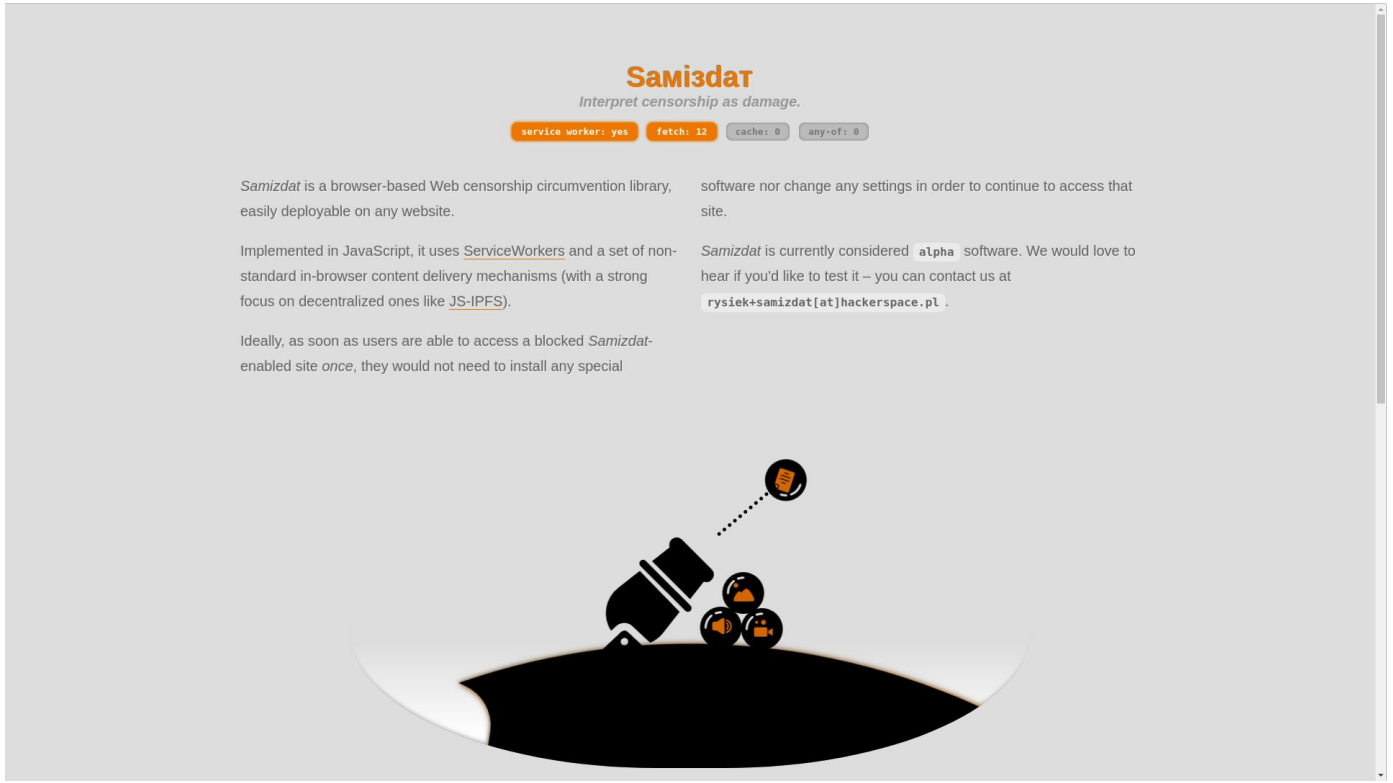
7981 files

2759.png

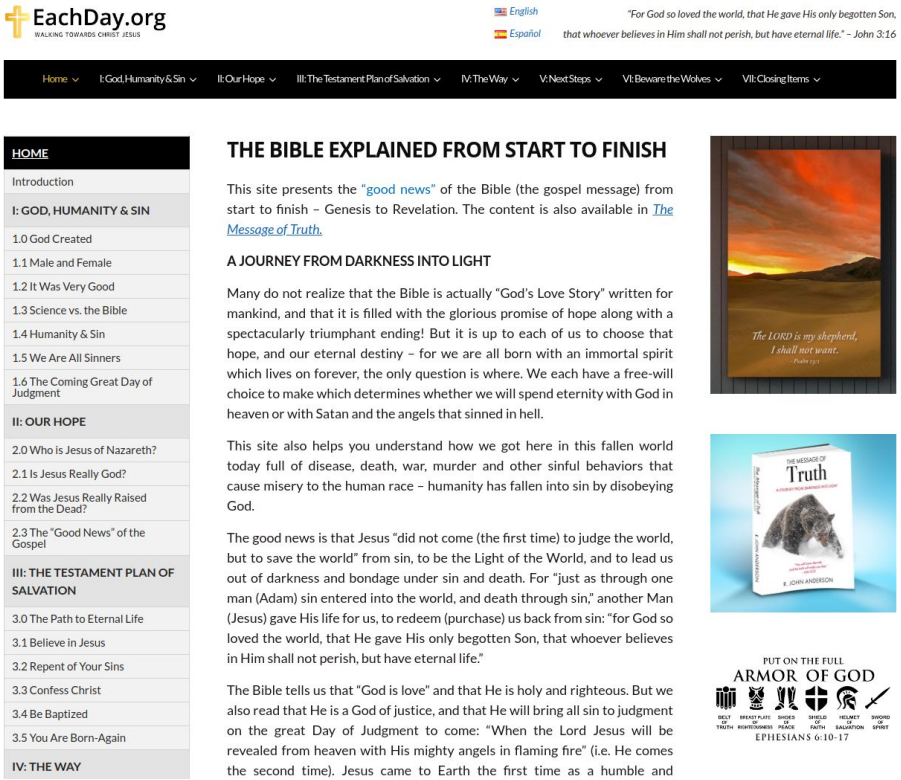
Russian news website



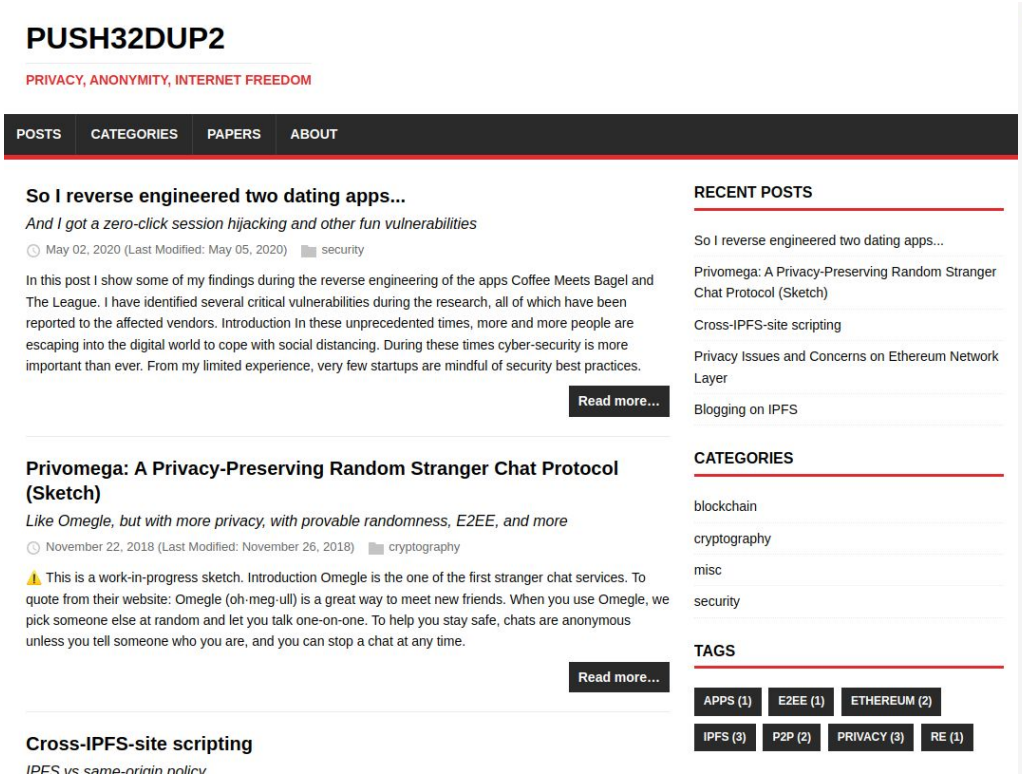
JS library

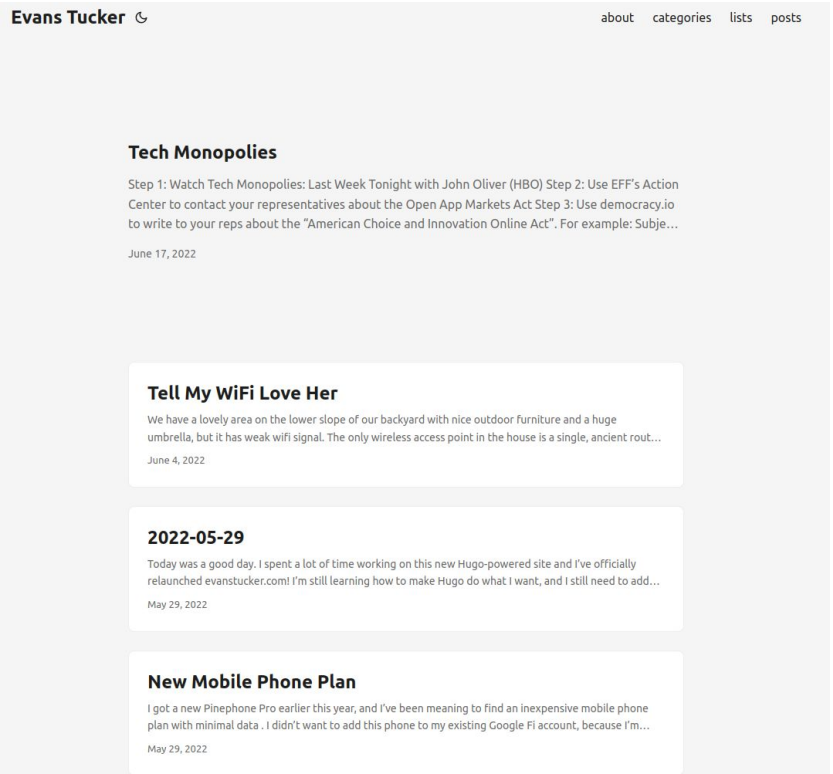

Samizdat
Interpret censorship as damage.
service worker: yes
fetch: 12
cache: 0
any-of: 0
Samizdat is a browser-based Web censorship circumvention library, easily deployable on any website.
Implemented in JavaScript, it uses ServiceWorkers and a set of non-standard in-browser content delivery mechanisms (with a strong focus on decentralized ones like JS-IPFS).
Ideally, as soon as users are able to access a blocked Samizdat-enabled site once, they would not need to install any special software nor change any settings in order to continue to access that site.
Samizdat is currently considered alpha software. We would love to hear if you'd like to test it – you can contact us at rysiek+samizdat[at]hackerspace.pl.

Religious website

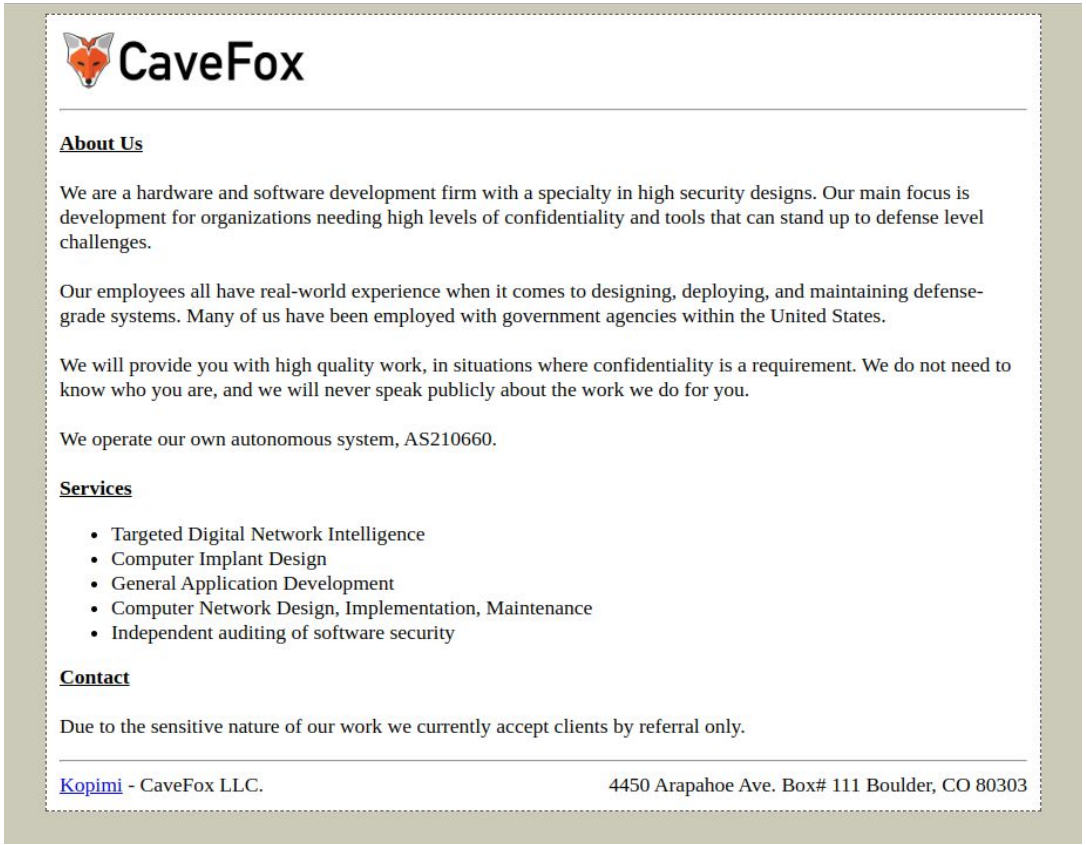


Bloggs


PUSH32DUP2
PRIVACY, ANONYMITY, INTERNET FREEDOM
POSTS
CATEGORIES
PAPERS
ABOUT
So I reverse engineered two dating apps...
And I got a zero-click session hijacking and other fun vulnerabilities
May 02, 2020 (Last Modified: May 05, 2020)
security
In this post I show some of my findings during the reverse engineering of the apps Coffee Meets Bagel and The League. I have identified several critical vulnerabilities during the research, all of which have been reported to the affected vendors. Introduction In these unprecedented times, more and more people are escaping into the digital world to cope with social distancing. During these times cyber-security is more important than ever. From my limited experience, very few startups are mindful of security best practices.
Read more…
Privomega: A Privacy-Preserving Random Stranger Chat Protocol (Sketch)
Like Omegle, but with more privacy, with provable randomness, E2EE, and more
November 22, 2018 (Last Modified: November 26, 2018)
cryptography
⚠️ This is a work-in-progress sketch. Introduction Omegle is the one of the first stranger chat services. To quote from their website: Omegle (oh·meg·ull) is a great way to meet new friends. When you use Omegle, we pick someone else at random and let you talk one-on-one. To help you stay safe, chats are anonymous unless you tell someone who you are, and you can stop a chat at any time.
Read more…
Cross-IPFS-site scripting
IPFS vs same-origin policy
RECENT POSTS
So I reverse engineered two dating apps...
Privomega: A Privacy-Preserving Random Stranger Chat Protocol (Sketch)
Cross-IPFS-site scripting
Privacy Issues and Concerns on Ethereum Network Layer
Blogging on IPFS
CATEGORIES
blockchain
cryptography
misc
security
TAGS
APPS (1)
E2EE (1)
ETHEREUM (2)
IPFS (3)
P2P (2)
PRIVACY (3)
RE (1)


Evans Tucker
about
categories
lists
posts
Tech Monopolies
Step 1: Watch Tech Monopolies: Last Week Tonight with John Oliver (HBO) Step 2: Use EFF's Action Center to contact your representatives about the Open App Markets Act Step 3: Use democracy.io to write to your reps about the "American Choice and Innovation Online Act". For example: Subje…
June 17, 2022
Tell My WiFi Love Her
We have a lovely area on the lower slope of our backyard with nice outdoor furniture and a huge umbrella, but it has weak wifi signal. The only wireless access point in the house is a single, ancient rout…
June 4, 2022
2022-05-29
Today was a good day. I spent a lot of time working on this new Hugo-powered site and I've officially relaunched evanstucker.com! I'm still learning how to make Hugo do what I want, and I still need to add…
May 29, 2022
New Mobile Phone Plan
I got a new Pinephone Pro earlier this year, and I've been meaning to find an inexpensive mobile phone plan with minimal data . I didn't want to add this phone to my existing Google Fi account, because I'm…
May 29, 2022

CaveFox



CaveFox

About Us

We are a hardware and software development firm with a specialty in high security designs. Our main focus is development for organizations needing high levels of confidentiality and tools that can stand up to defense level challenges.

Our employees all have real-world experience when it comes to designing, deploying, and maintaining defense-grade systems. Many of us have been employed with government agencies within the United States.

We will provide you with high quality work, in situations where confidentiality is a requirement. We do not need to know who you are, and we will never speak publicly about the work we do for you.

We operate our own autonomous system, AS210660.

Services

- Targeted Digital Network Intelligence
- Computer Implant Design
- General Application Development
- Computer Network Design, Implementation, Maintenance
- Independent auditing of software security

Contact

Due to the sensitive nature of our work we currently accept clients by referral only.

Kopimi - CaveFox LLC. 4450 Arapahoe Ave. Box# 111 Boulder, CO 80303

So the **IPNS records** are an alterative way of **hosting** a **static website**

Conclusion

RQ: Synchronized indexing of CIDs over time

1. How does the **rate of change** in default IPNS records of peers on the IPFS network impact the **reliability of an synchronized indexing** of CIDs **over time**?

RQ: File formats and data types

2. What **file formats and data types** should be **prioritized** for information extraction compatibility with the default IPNS records to ensure effective **integration** into a **search engine system?**

IPNS records are being used as an alternative way of **hosting websites** and the **content** is often **changing**

The **content** on IPNS can become **available** to others and make the network much **more useful**

Thanks for listening!
Any **questions**?

Opposition!