Introduction

IPFS stands for **Inter Planetary File System**.

IPFS is a peer-to-peer distributed file system that seeks to connect all computing devices with the same system of files.

It also is a protocol designed to create a permanent and decentralized method of storing and sharing files.

Most people store their files on their local computer or in the cloud through a cloud provider (like Dropbox, AWS S3, Azure Cloud etc). If you have a website, the assets might be served from a CDN which geographically spreads/replicates your files for faster access. These are all very good solutions because they do a good job of storing and serving your files with high availability and no loss of data.

However, they also have a few drawbacks which can be concerning. If these service providers have outage (and it happens), your files can be inaccessible. If you store files against the company policy, they have the right to remove/block it. Depending on the number of files, the cost of storing could get very high.

In the IPFS world, these service providers with centralized servers are replaced by peer-to-peer network of computers. Just like anyone can run an Ethereum node, anyone can run an IPFS node and join the network to form a global file system. The files are replicated across many nodes and it is almost impossible to lose access to your files and is also censorship resistant. Anyone in the world can run an IPFS node and in the future can get compensated through Filecoin.

Installation

Here we are using Windows system. For other system, the installation is more or less same.

- Go to https://dist.ipfs.io/#go-ipfs and download ***go-ipfs*** *.*
- Extract the zip and copy the folder and paste into Program Files.
- Set the environment variable i.e. set **%PATH%**.
- Open cmd and type command ***ipfs version*** to ensure that ipfs is successfully installed in your system or not.
- **Instantiate the ipfs node :** Type command ***ipfs init*** to instantiate the ipfs node in your system.
- **Going online :** Type command ***ipfs daemon*** to connect to the live ipfs network.
- Browse and check the url ***localhost:5001/ipfs/webui*** to check the network connected or not.

Linux:

- Go to https://dist.ipfs.io/#go-ipfs and download ***go-ipfs*** *.*
- Go to that directory and type command : sudo ./install.sh
- ipfs version
- ipfs init
- ipfs daemon
- Open different terminal
- ipfs add ~/Downloads/videoplayback.mp4

Writing Files into IPFS Network

- Type command ***ipfs add fileName.extension*** to add file

You will get a hash of the file after executing the above command. E.g QmfUdW45xat5T4xJy9jz2D9yfrg6f5KLakZ3nAnqNu28Ft

Reading Files from IPFS network

Once you get the hash of the file append it with https://ipfs.io/ipfs/ to access it from the network.

- Ex.https://ipfs.io/ipfs/QmfUdW45xat5T4xJy9jz2D9yfrg6f5KLakZ3nAnqNu28Ft or http://localhost:8080/ipfs/QmfUdW45xat5T4xJy9jz2D9yfrg6f5KLakZ3nAnqNu28Ft

Note :

- To add more than one file with *add* command of ipfs, you can use something like this ***ipfs add -r .***
- ***ipfs swarm peers*** command to check number of nodes I am connected to.
- Video streaming can also be possible on IPFS. Just add the video into IPFS and browser the url like this
- https://ipfs.io/ipfs/QmaweTZFmcn1dZdovhLSufVM2W6nZ2exvQbx94DQFYskZ8/#play
- More : https://ipfs.io/docs/

Hosting your website on IPFS

https://ipfs.io/docs/examples/example-viewer/example#../websites/README.md

- Create a folder (application folder where allPFS DocI the static files would be, i.e.

mytestapp) and put all your static files into it.

- Go to command line, make sure you are connected to the live network (using daemon command) and type command : ***ipfs add -r mytestapp***
    - Notice all your files along with root directory are added to the IPFS. Copy the has of last file (which is the hash of the mytestapp folder) adn browse in the browser like this : https://ipfs.io/ipfs/Qma86bMvGkVuEUg5WXTsKKxcq23377T2ZgA9B5gfCaNbcA/test.html

Notice here domain name is in hash which is not readable at all. We need to change it into human readable format. - Needs to be figured out.

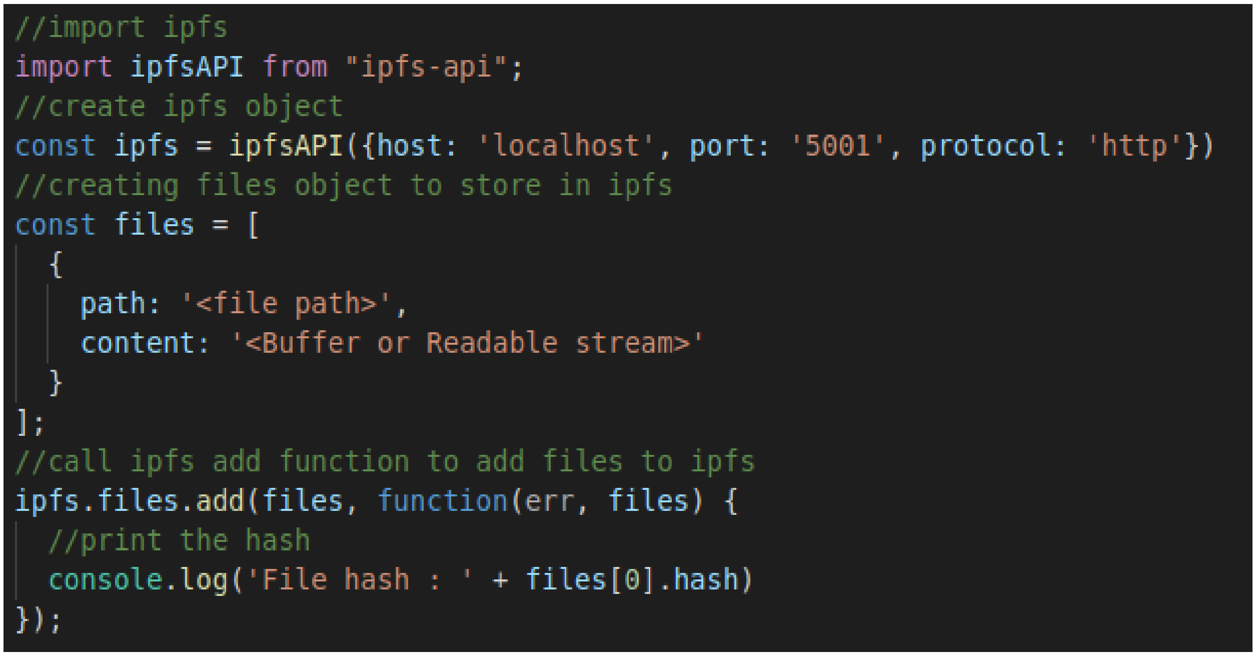
Connecting Javascript with IPFS node

Steps

1. Installation

```
npm install --save ipfs-api
```

2. Run IPFS daemon
3. Upload file using IPFS instance object



```
//import ipfs
import ipfsAPI from "ipfs-api";
//create ipfs object
const ipfs = ipfsAPI({host: 'localhost', port: '5001', protocol: 'http'})
//creating files object to store in ipfs
const files = [
  {
    path: '<file path>',
    content: '<Buffer or Readable stream>'
  }
];
//call ipfs add function to add files to ipfs
ipfs.files.add(files, function(err, files) {
  //print the hash
  console.log('File hash : ' + files[0].hash)
});
```

References

Web3 Js API

Web3.net

Web3.eth

Web3.shh

Web3.db

Web3 Js Doc

IPFS Doc

IPFS API