

Store with IPFS – Decentralized File Upload  
  
**Objective/Aim:**  
  
 To understand how the InterPlanetary File System (IPFS) works and to perform a practical demonstration



of uploading, storing, and retrieving files on a decentralized peer-to-peer network.

**Apparatus/Software Used:**

* Laptop/PC
* Internet connection
* IPFS Desktop or IPFS Command-Line Interface (CLI)
* Browser with IPFS gateway access (e.g., https://ipfs.io)

**Theory/Concept:**

* What is IPFS?  
   IPFS is a peer-to-peer distributed file system that seeks to make the web faster, safer, and more open. Instead of storing files on a central server, IPFS uses content addressing (hashing) to uniquely identify files.
* Key Concepts:
* Content Identifier (CID): A unique cryptographic hash of a file.
* Distributed Storage: Files are split, stored, and shared across multiple peers.
* Decentralization: No central server, improves resilience and censorship resistance.
* Persistence: As long as one peer “pins” the file, it remains available.
* Why use IPFS?
* Immutable storage (file hash changes with modification).
* Reduces duplication and improves bandwidth efficiency.
* Foundation for decentralized apps (dApps), NFTs, and blockchain-based storage.



**Procedure:**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* **Install IPFS Desktop or CLI** from https://ipfs.tech.
* Start the IPFS daemon to connect your system to the IPFS network.
* Add a file to IPFS using the CLI:
* ipfs add filename.txt
* Note the **CID** generated for the file.
* Access the uploaded file using a public gateway:
* https://ipfs.io/ipfs/<CID>

**Observation Table:**

* Each uploaded file is assigned a unique CID (hash).
* Even a small change in file content results in a completely different CID.
* Files remain retrievable from any IPFS node using the CID.
* Decentralized storage ensures file persistence without dependency on a central server.

