TP 3: IPFS (60 - 80 minutes)

You are part of a **disaster response team** working in a remote area where internet access is unavailable, and traditional centralized servers are unreliable. The mission is to establish a **resilient, decentralized communication system** for sharing critical information between team members. You will use IPFS to create a peer-to-peer network for exchanging emergency files, broadcasting team updates, and ensuring file integrity—all without relying on centralized infrastructure or HTTP gateways.

You will use kubo (ipfs command line) and only the official documentation (you won't be needing anything else) https://docs.ipfs.tech/how-to/command-line-quick-start/

Scenario

Your team has just arrived at a remote location. The initial tasks include:

1. **Setting up a local peer-to-peer network** to ensure seamless communication among team members.
2. **Distributing critical information**, such as evacuation routes, weather updates, and medical protocols, using IPFS.
3. **Broadcasting real-time updates** to all team members using a decentralized messaging system.
4. **Ensuring data integrity** of sensitive files, such as medical instructions or maps, so everyone has consistent and verified information.
5. **Demonstrating the resilience of the system** by simulating a scenario where one node is temporarily offline and later reconnects.

Questions

1. **Initial Setup (15 min):**
    - Set up a local IPFS node for each participant.
    - Configure the nodes to communicate within the same local network and discover each other automatically.
2. **File Sharing (20 min):**
    - Share a critical text file (e.g., evacuation routes) between peers, ensuring no HTTP gateway is used.
    - Verify that the file can be retrieved by others using only the CID.
    - Discuss how CIDs ensure file integrity and demonstrate their cryptographic properties.
3. **Decentralized Messaging (20 min):**
    - Establish a topic for team communication (e.g., `disaster-updates`).
    - Use Pub/Sub to broadcast real-time messages to all peers.
    - Simulate a scenario where a node temporarily leaves the network, re-subscribes, and verifies it can still receive updates.
4. **Simulating Resilience (15 min):**
    - Add a larger file (e.g., a map image) to the network.
    - Pin the file on multiple nodes to ensure availability even if one node goes offline.
    - Test the retrieval of the file after a node comes back online.
5. **Final Wrap-Up (10 min):**
    - Reflect on how this system differs from centralized methods (e.g., HTTP servers).
    - Discuss the potential real-world applications of IPFS for disaster recovery, censorship resistance, or offline networks.

Chapitre 03: IPFS

Aller à...

Chapitre 04: Introduction aux Smart Contracts