



Abstract

Decentralized cloud storage represents a fundamental shift in the efficiency and economics of large-scale storage. Eliminating central control allows users to store and share data without reliance on a third-party storage provider. Decentralization mitigates the risk of data failures and outages, while simultaneously increasing the security, read performance and privacy of object storage. Decentralization also allows market forces to optimize for less expensive storage at a greater rate than any single provider could afford. Although there are many ways to build such a system, there are some specific responsibilities any given implementation should address, including: security, compatibility, building resilience against bad actors, ensuring favorable economics for both providers and end users, and setting incentives. Based on our experience with petabyte-scale storage systems, we introduce a modular framework for considering these responsibilities and for building our distributed storage network. Additionally, we describe an initial concrete implementation for the entire framework.

The V3 white paper also goes into significant depth on the design constraints we took into consideration when designing the network, walkthroughs on how to perform certain tasks on the network, future plans for the platform, and calculations that informed many of our design decisions.

Discussion

Decentralized cloud storage has emerged as a potential solution to the world's growing data needs. The amount of digital data that the world creates doubles every year; by some estimates, it will reach 44 zettabytes per year by 2020. At the same time, the vast majority of storage devices are operating at less than 25% capacity, and the price of cloud storage has declined by less than 10% annually over the past three years. Moreover, the traditional cloud model has significant issues with security, availability, and performance--particularly in regions far from major data centers. The inherent benefits of decentralized cloud storage can address these needs. By using existing, underutilized hard drives and bandwidth while maintaining SLAs that are comparable to traditional data centers, decentralized cloud storage stands out as a new solution that is both cost effective and performant. Moreover, adopting a decentralized approach enables us to create a system that is significantly more durable and more resistant to bad or unreliable actors.

The Storj platform can address several key segments within the market today, particularly those related to long-term archival storage and S3-compatible object storage. However, we have designed a system to address a much wider range of use cases, from basic object storage to

content delivery networks (CDN). The V3 Storj network is the next evolution of cloud storage and will be a key influencer and innovation driver in the developing web 3.0 era.

Network Improvements Based on Experience

The new V3 network sets itself apart from other decentralized platforms in several ways. Our team has prioritized simplicity in every aspect of the design. Most of the network incorporates proven, widely-used technologies, but deploys them in innovative ways. Our commitment to function includes our decision to avoid using a blockchain or distributed ledger for storing files or metadata. Storage consumers are used to platforms with horizontal scaling, like AWS S3, which gain performance as more hardware is added. However, distributed ledgers cannot easily achieve this, so we avoid using them for the actual storage of data.

Another key differentiator of the V3 network is its use of erasure codes for resiliency, rather than replication. Because bandwidth is a limiting factor on decentralized cloud storage networks, replication is a poor choice as a tool to guarantee resiliency. Based on our research and experience operating our previous network (which achieved a scale larger than any other decentralized cloud storage network in the world), we've found that systems which use erasure codes to achieve six 9s of durability utilize five times less storage capacity than systems using replication to achieve the same durability level.

The Challenges of Decentralized Networks

There are several design constraints that must inform the requirements, network implementation and overall architecture of a scalable, durable decentralized cloud storage platform. Many platforms fail to adequately address these design constraints, delivering decentralized cloud storage networks that have unreasonable latency, low file durability, high costs and low performance as a result. By solving for the design constraints, Storj's V3 network delivers a network that can outperform centralized cloud storage platforms in many ways. The design constraints we have considered include:

- The need for AWS S3 compatibility to ensure easy migration
- Device failure and churn, which are tightly coupled with durability and bandwidth
- Minimizing bandwidth usage due to bandwidth caps imposed by ISPs
- The need for enterprise-grade security and privacy for data stored on the network
- Object storage vs database use cases
- Byzantine fault tolerance across the decentralized cloud storage network
- General attack resistance to combat data breaches and DDoS attacks
- Achieving decentralization to ensure maximum reliability
- Economic viability to keep costs competitive with centralized platform offerings
- Building coordination avoidance systems instead of coordination dependant systems

An important goal of our platform is to deliver cloud storage that is easy to incorporate into existing infrastructure and applications. It must also deliver on security, encryption, reputation management (the ability to weed out bad actors), trustlessness (minimizing the amount of trust required from any single entity on the network), durability and resilience. Without delivering these essential capabilities, a decentralized network will ultimately fail.

Storj's V3 Architecture

We have designed a specific framework of eight components that provide an optimal implementation of decentralized storage. The architecture we outline operates within the limits of the design constraints, and provides the essential capabilities expected - all while passing savings on to storage users.

The framework of components discussed in this white paper are:

- Storage nodes
- Peer-to-peer communication and discovery
- Redundancy
- Metadata
- Encryption
- Audits and reputation
- Data repair
- Payments

This framework is fundamental to the overarching Storj platform and as the network matures and evolves, we do not expect the framework or components to change. For this reason, we do not anticipate that there will be a need for a complete rework of the network, rather, we expect the concrete implementation of the individual components to evolve. As this occurs, we will also update the white paper accordingly.

These eight components are incorporated into three different parts of the network.

- An *Uplink* is any software or service which invokes LibUplink in order to communicate with Satellites and Storage Nodes. Examples of Uplinks include the Uplink CLI and the Gateway.
- The *Storage Node*, which stores the data for the network. Each one is independently operated and does not share bandwidth, power or other resources.
- The *Satellite*, which operates as a heavy client that connects Uplinks to the Storage Node network and manages metadata for files. It also handles file audits, repair and other crucial network tasks. There will be many satellites on the network and companies and community members will be able to operate their own satellites as well.

Bridging Open Source and Decentralization

In addition to being open source, our new V3 network also financially empowers open source companies by enabling them to generate revenue every time their users store data on the cloud. This supports the open source companies interested in monetizing their product's use in cloud, while also helping Storj grow adoption of its platform within the innovative open source community.

The new program is enabled by the network through connectors built in conjunction with each open source partner. The connectors track data usage either by storage bucket or by user. When data flows through one of these connectors, the open source company is given credit for the usage and a percentage of the revenue generated flows back to the corresponding project. Partners also earn revenue for bandwidth usage on the network.

The new Storj white paper will be continually updated, as new functionality is made possible through advances in research and improvements in technology. We expect that the main eight components will remain the same, however their concrete implementation will evolve to maximize security, reliability, efficiency, performance and take advantage of other benefits.