DepSky: Dependable and Secure Storage in a Cloud-of-Clouds
Alysson Bessani, Miguel Correia, Bruno Quaresma
Fernando André, Paulo Sousa

http://www.navigators.di.fc.ul.pt
http://www.tclouds-project.eu

Cloud-of-Clouds Storage

Rationale
✓ Everyone loves the cloud
✓ Not everyone trusts cloud providers
✓ Why not use several clouds instead of one?

The DepSky storage protocol suite

Design principles
✓ No trust on any single cloud
✓ No protocol code running on the clouds

Assumptions
✓ Clouds and readers can be Byzantine
✓ Crash-only writers
✓ Asynchronous system* 
  *synchrony is needed for the lease protocol

Employed techniques
✓ f-disseminating Byzantine Quorum Systems
✓ Information-optimal erasure codes
✓ Secret sharing

Write

Confidentiality & Storage-efficiency

Original data can be recovered combining the file fragments and the key shares:
- If secret sharing is needed:
  Each server stores at least f+1 fragments/shares.

Read

Other protocols
✓ Create, destroy, reconfigure and old versions removal
✓ Lock/lease protocol for ensuring data unit single writer

Consistency-proportional storage
✓ Regular if the clouds are at least regular
✓ Read-your-writes (RYW) if clouds are at least RYW
✓ Eventual if the cloud is eventually consistent

Evaluation

Setup
✓ Four clouds setup (Amazon S3, Rackspace, Windows Azure and Nirvanix), tolerating a single fault.
✓ Two DepSky configurations: A (replication-only) and CA (replication + confidentiality)
✓ Eight PlanetLab clients reading and writing from different locations around the world during a month.
✓ Three data unit sizes: 100kb, 1Mb and 10Mb.

Results highlight
✓ DepSky $$$ costs are twice the average costs of using a single cloud.
  - Can be better: data can be stored only on f+1 (A) or 2f+1 (CA) clouds.
✓ Read latency and throughput better than single clouds (in general).
✓ Write latency and throughput worse than single clouds (in general).
✓ Data availability (not surprisingly) better.