DARUMA: Regaining Trust in Cloud Storage

Doron Shapiro, Michelle Socher, Ray Lei, Sudarshan Muralidhar

Advisors: Boon Thau Loo, Nadia Heninger

Penn Engineering

Background

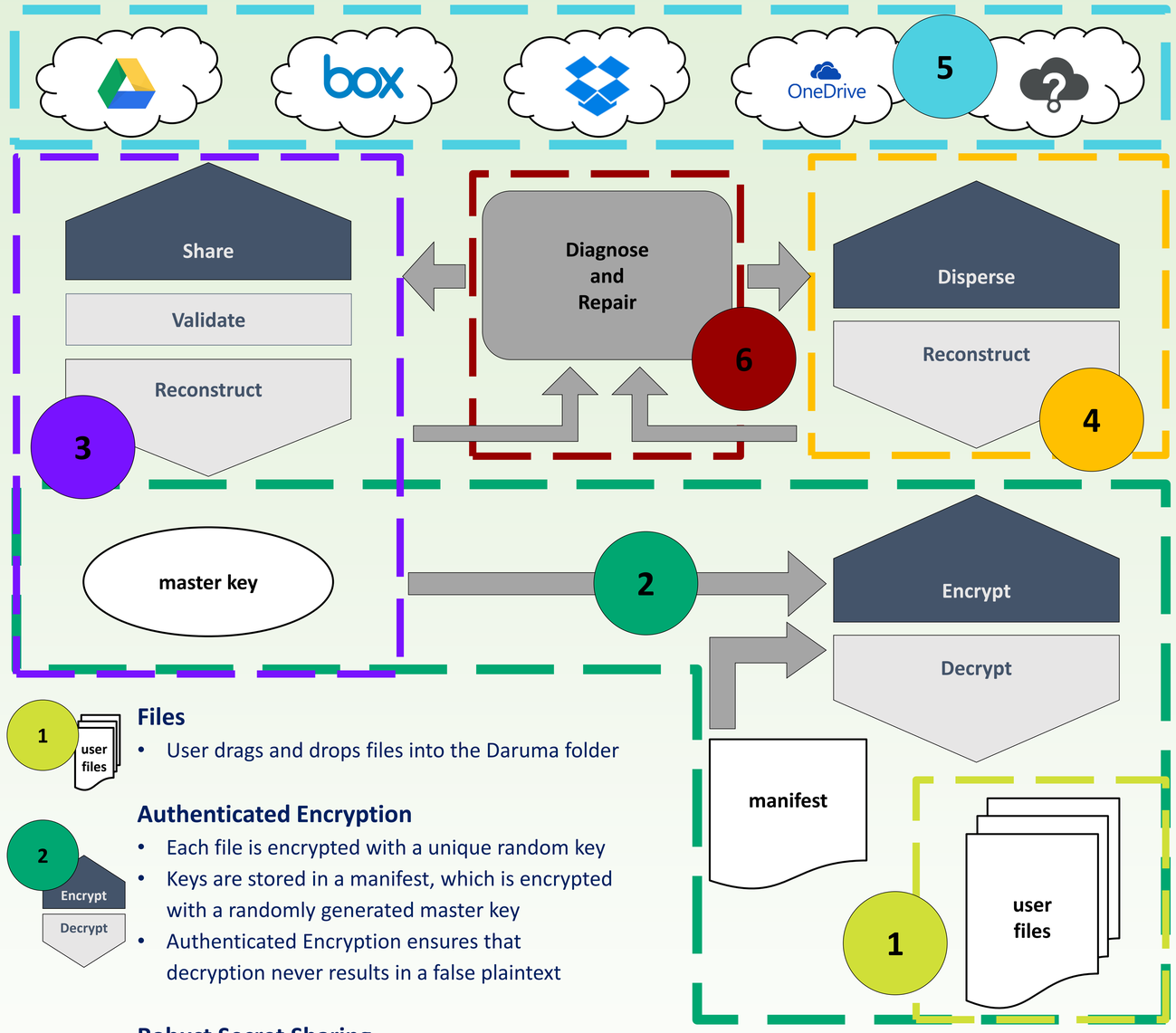
- Users trust cloud storage providers (Dropbox, Google Drive, OneDrive, Box, etc) to keep important documents safe, secure, and available
- Providers cannot guarantee this
  - Downtime is inevitable
  - Hackers or software bugs can compromise data
  - Providers or governments may read user files

Abstract

- **Daruma eliminates the need to trust providers**
- Daruma provides new guarantees: **no one provider can read, change, or delete user files**
- Daruma efficiently combines the storage on a user's existing provider accounts
- Daruma runs entirely locally and is open-source, so users don't have to trust it either
- Daruma provides the **benefits of cloud storage without its inherent risks**, through a standard and familiar interface
- Current solutions only provide one or two of the so-called "CIA" properties:
  - Confidentiality (secrecy)
  - Integrity (tamper-resistance)
  - Availability (uptime)
- Daruma guarantees all three – and more

How are we different?

- Providers encrypt files upon storage
  - Providers hold the encryption keys, so data is vulnerable to irresponsible employees and government subpoenas
  - With Daruma, cloud services can't ever recover the content or even the names of files stored
- Software can encrypt files before storing them online
  - The user is forced to remember a private key; if forgotten, all files are permanently lost
  - Daruma doesn't require the user to remember any new passwords – it securely distributes all encryption keys across providers
- Some tools automatically copy files across providers
  - This is space inefficient as it requires the entire file to be stored on every provider used
  - Daruma uses intelligent redundancy algorithms to guarantee availability while using significantly less storage space



Files

1 user files

- User drags and drops files into the Daruma folder

Authenticated Encryption

2 Encrypt / Decrypt

- Each file is encrypted with a unique random key
- Keys are stored in a manifest, which is encrypted with a randomly generated master key
- Authenticated Encryption ensures that decryption never results in a false plaintext

Robust Secret Sharing

3 Share / Validate / Reconstruct

- The master key is split into shares (one per provider)
- A collection of all but one of the shares will be required to reconstruct the secret key
- Any smaller collection reveals no information about the secret key
- Shares are resistant to tampering and deletion

Reed-Solomon Encoding

4 Disperse / Reconstruct

- Encrypted files are split into chunks (one per provider)
- Chunks are resistant to tampering and deletion

Providers

5

- Shares and chunks are stored on providers
- Providers aren't trusted with sensitive data
- Daruma works with any cloud provider

Diagnostics and Repair

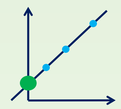
6 Diagnose and Repair

- Failing providers are marked as unstable
- Corrupted or missing data is replaced via the standard flow
- Daruma tolerates unexpected provider behavior at any time – even during repair
- With time, providers can regain reputation

Implemented Cryptography

Shamir Secret Sharing (SSS)

- Goal: store master key (*secret*) on *n* unreliable providers without leaking information
- To allow recovery from *k* of *n* providers, choose a random polynomial *P* such that $deg(P) = (k-1)$ and $P(0) = secret$
- Distribute a $(x, P(x))$ share to each provider



  - Any *k* shares are required to reconstruct *P* (and thus *secret*)
  - Left: With *n=3* and *k=2*, *secret* (green) can only be recovered by at least 2 shares (blue)
- Tolerant of share deletion, but not share tampering

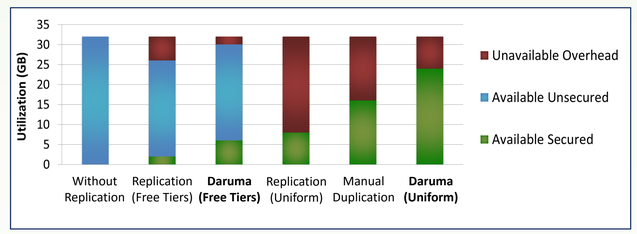
Robust Secret Sharing

- Goal: Provide authentication for SSS in order to tolerate and identify up to $(k-1)$ tampered shares
- Implements Rabin & Ben-Or's "Verifiable Secret Sharing"
- Generate *check vectors* for each pair of providers
- On recovery, validate shares before SSS reconstruction

Performance

Capacity Utilization

- Secured space on Daruma is limited by the capacity of the smallest registered provider
- Two motivating cases demonstrate Daruma's advantages
  - *Free Tiers* – Provider free account capacity varies greatly, ranging from 2 (Dropbox) to 15 GB (Drive)
  - *Uniform* – Providers have the same capacity (8 GB)
- Compared to other redundancy solutions, Daruma makes the same guarantees with less replication overhead
- In near-uniform cases, Daruma maximizes secured space



Speed Benchmarks

- Daruma's speed is always influenced by a constant algorithmic cost and by the slowest provider
- For small files, Daruma is only a few seconds slower
- For large files, Reed-Solomon encoding and parallelization give Daruma an advantage