STORAGE DEVELOPER CONFERENCE

SDC 21

BY Developers FOR Developers

Virtual Conference
September 28-29, 2021

A SNIA. Event

Compression, Deduplication, Encryption Conundrums in Cloud

Presented by: Tejas Chopra, Netflix, Inc.

Agenda

- Introduction
- Cloud Storage costs and footprint
- Compression, Deduplication, Encryption: Why & How?
- Right order of applying them
- Learnings & Takeaways

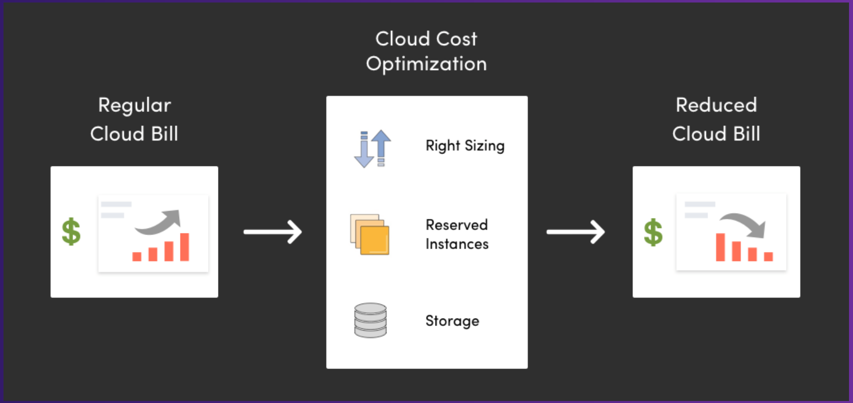
About me

- Senior Software Engineer, Netflix
- Keynote speaker: Distributed systems, Cloud, Blockchain
- Senior Software Engineer, Box
- Datrium, Samsung, Cadence, Tensilica

NETFLIX

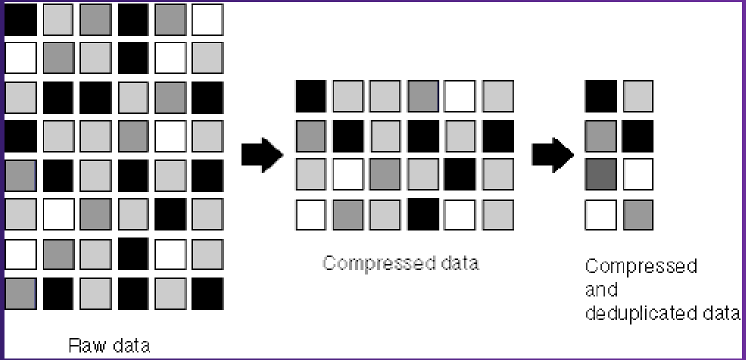
Cloud

- Why? Manageability, scalability, pay per use
- Costs: Per request cost, data at rest cost, egress cost
- Ex: For a given file stored in cloud, if the file is read many times, you pay egress cost each time
- Box: 100s of petabytes of data; Netflix: exabytes of data
- On-premise file systems: Compression, Deduplication to reduce file footprint



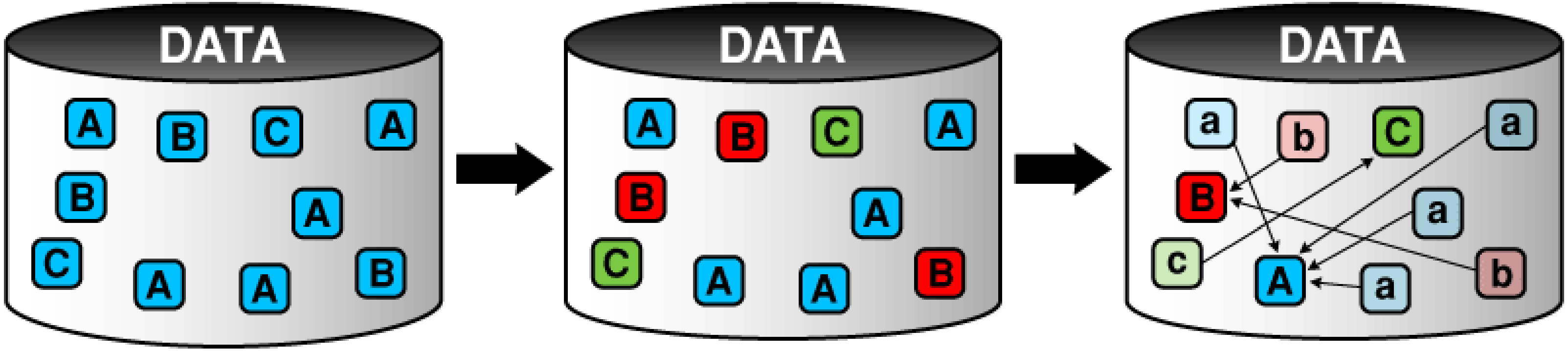
Compression

- Several techniques: gzip, lzip, lz4, etc.
- Not all data is compressible
- Overhead is generally < 2%, which is acceptable for file storage, not very acceptable for streaming videos
- Box: Compression ratios > 5X



Deduplication

- Store one copy and hand out references to users / apps
- File is not deleted from the backend unless all users / apps delete the file
- How to check for duplicates before storing the file?
    - Online deduplication
    - Offline deduplication
- Can deduplicate at finer granularities as well
- Deduplication boundaries are often also dictated by security



1 Hash values are generated from each piece of data.

2 The hash values are compared to identify duplicates

3 Duplicates are replaced with pointers to save storage space

Encryption

- Encryption over the wire is solved by TLS
- Two strategies:
  - Client Side Encryption
  - Server Side Encryption
- Clients generally prefer encrypting the data
- At Netflix, file sizes are often greater than max S3 size, so we have a layer on top of S3 which chunks a file into smaller pieces and uploads them.
- Each chunk is separately encrypted.
- NOTE: Encrypted data is not compressible, rarely dedupable.

Ordering Rule 1: Encrypt last

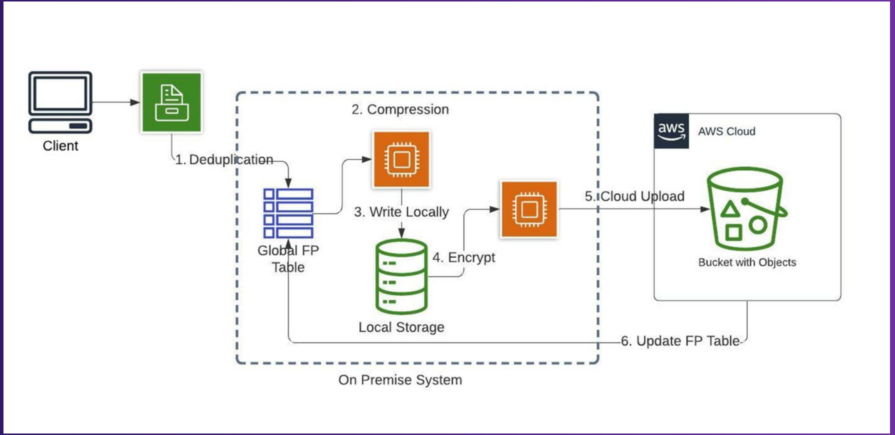
- If data is encrypted, you cannot compress or deduplicate
- If client side encryption is used, it means you have to employ online deduplication & compression on client side. Design can get complicated
- Per object encryption v/s maintaining a single encryption key per pool of objects: Implications on deduplication, security, etc.

Ordering Rule 2: Deduplicate first

- Employ online deduplication
- When data is streamed, maintain a global pool of fingerprints (hashes of data)
- Compare the current object/chunk's fp with the fps in the pool
- If data is present, no need to stream the data to cloud
- Increase the ref count of the fp, and return the reference to stored object on cloud
- Compression algorithms change, by deduping first, you're protecting against any change in algorithms in the future

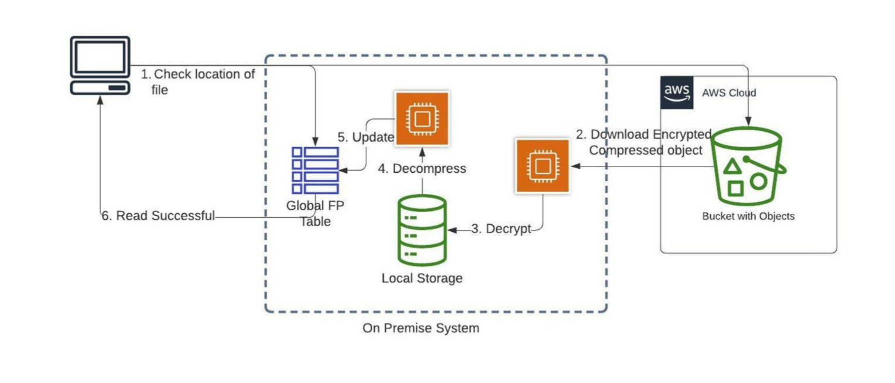
Optimum write flow

1. Generate a fp of data being stored, and check if the fp exists in a global pool. If so, this data is dedupable, if not, insert the fp in the pool
2. Compress the data on the client side
3. Encrypt the data after compression
4. Upload the compressed and encrypted bits to cloud, and update the global fp table with the cloud location of the uploaded file.



Optimum Read flow

1. When a read request is issued, fetch the compressed, client-side encrypted data from cloud locally.
2. Using the mapping of fileId → encryption key, decrypt the compressed bits.
3. Decompress the blob
4. Finally serve the reads.



Takeaways and Learnings

- Not all data is compressible & compression adds latency, so can remove the step of compression
- Many orgs (such as Box), first store the data on-premise and then upload it to cloud - saving egress costs, many files are temporary in nature
- So, encryption is applied at the point of uploads to cloud
- Online deduplication is tough -- state management, reference counting, concurrent accesses needs locking, so companies start off with offline deduplication to get some benefits

Thank you!

https://www.linkedin.com/in/chopratejas

chopratejas@gmail.com

chopra_tejas