Data Security and Protection

Juncheng Yang

March 11

Harvard John A. Paulson
School of Engineering
and Applied Sciences

HARVARD UNIVERSITY
MADSYS
Measurement and Design of Systems Lab

Agenda

- Data security
- Data protection

Data Security

Data security

- How does storage system protect your data from being accessed by unauthorized users?
  - multiple layers of protection
- Access control
  - authentication and authorization
  - RBAC & ABAC: role-based access control, attribute-based access control
  - access control list (ACLs): POSIX permissions

Encryption

- Data encryption
  - encryption in-flight: protect data being transmitted
  - encryption at-rest: server encrypts data before writing to disk
  - key management: critical for rotation and revocation
    - how to rotate? naive approach requires re-encrypts everything
      - hierarchical management
        - data encryption key (DEK): encrypted and stored with data
        - key encryption key (KEK): master key to encrypt DEK
- Hardware protection
  - self-encrypting drives (SEDs): protect data on device
  - trusted execution environment: protect data being computed

Data Protection

The hierarchy of protection

- **High Availability (HA)**
  - survive a *component* failure, e.g., disk/server
  - goal: uptime
- **Backup**
  - survive a *corruption* or *deletion* event
  - goal: restoration
- **Disaster Recovery (DR)**
  - survive a *site* failure, e.g., flood/fire
  - goal: business continuity
- **Archive**
  - long-term retention for compliance
  - goal: low cost

Backup metrics

- **RPO (Recovery Point Objective)**: "How much data can we afford to lose?"
  - RPO = 0: require synchronous replication (expensive, latency penalty)
  - RPO = 24 hours: nightly backup is sufficient
- **RTO (Recovery Time Objective)**: "How long can we be down?"
  - RTO = 0: require Active-Active clusters (failover).
  - RTO = 4 hours: require enough bandwidth to reload data from disk/tape

Backup mechanics

- **Full**
  - copy everything
  - slow backup, high storage
- **Incremental**
  - copy only what changed since the *last backup*
  - fast backup, slow restore
- **Differential**
  - copy only what changed since the *last Full backup*
  - medium backup, fast restore
- **Synthetic Full**
  - the modern standard
  - backup server merges incrementals into a full image offline

Deduplication

- Backup data is highly redundant, deduplication improves efficiency
- Source-side vs. target-side:
  - *source:* client calculates hashes and only sends unique blocks, saves bandwidth
  - *target:* server accepts all data and filters it, saves client CPU
- Inline vs. post-process:
  - *inline:* dedup happens as data arrives, slows down ingest
  - *post-process:* write raw data first, dedup later, requires spare disk

Ransom protection

- Entropy analysis (measurement)
  - regular documents: low entropy
  - encrypted data: high entropy
  - modern arrays sample I/O stream in real-time, take a snapshot if upon an entropy spike
- Immutability & WORM
  - write once, read many, e.g., object lock
- Common best practices
  - 3-2-1: three copies of data, 2 media types, 1 offsite
  - air gapping: physically or logically disconnect backup from network

Summary

- Data security
- Data protection