Flexible and Secure Services using Sensitive Information in the Cloud

Sponsor: NSF IUCRC Fundamental Research Program

In partnership with Microsoft and IBM

Faculty: Doug Blough, Ling Liu, Dan Russler (Adjunct)

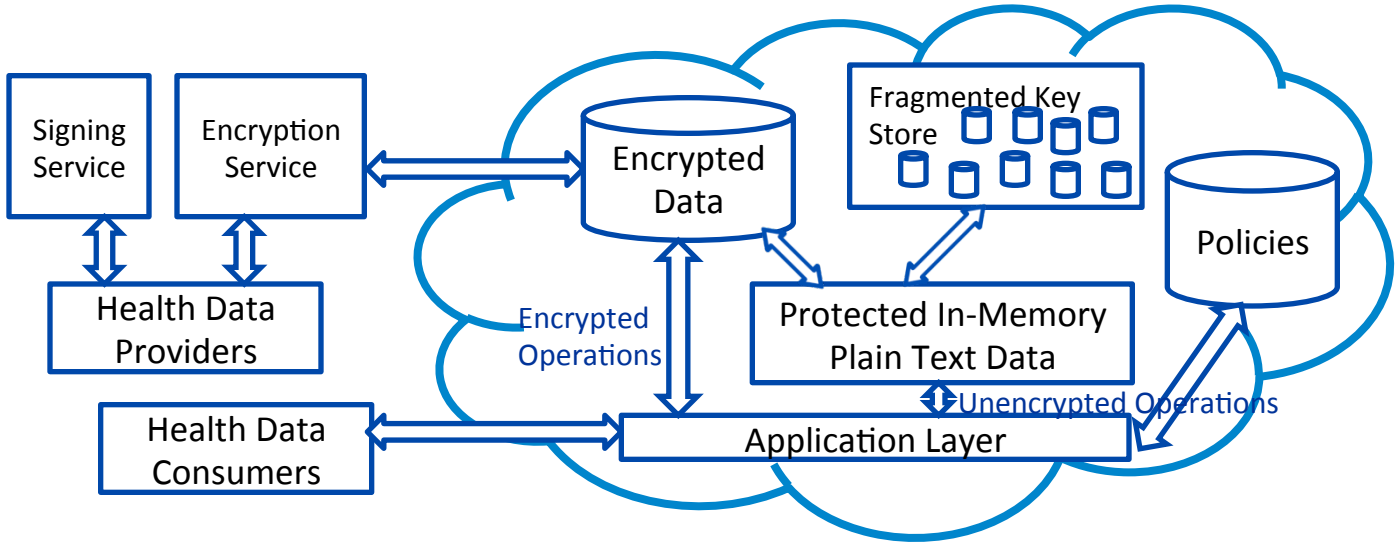
Graduate Students: Kyle Bilbray, Emily Bragg, Jordan Brown, Michael Darakananda, Brian Laub, Yuzhe Tang, Jiaqi Xue

Background

- Reluctance to host services with business-critical information and/or sensitive personal information in clouds
  - Cloud resources are shared with other users, possibly even direct competitors
  - Cloud providers typically have full and unfettered access to user data
- *Project goal:* Develop and evaluate an architecture and underlying mechanisms for flexible, secure and privacy-preserving execution of services in clouds (use health as the driving application area)

Research Thrusts and Architecture

- Information trust, integrity and provenance - signature schemes, information flow monitoring/control
- Leveraging cloud resources
  - Remove host OS from trusted computing base
  - Data fragmentation across multiple servers
- Operating on encrypted data



Current Research Emphases

- Flexible and secure object storage
  - Encode object into multiple fragments (replicas, secret shares, erasure-coded pieces, etc.)
  - Support many different encodings, including user-defined ones
  - Store fragments across availability zones, administrative zones, and virtual nodes per user-specified requirements
- Programming support for use of protected memory
  - Protected memory allocated by hypervisor, not visible to dom0
  - Create versions of malloc() and free() for allocation/deallocation of protected memory
  - Provide support for “protected code segments” and “protected function execution”
- Granular information flow control
  - Ability to perform fine-grain tracking of sensitive information flow and block unauthorized accesses

Primary Research Platform

- 82-node cluster operated by GT-CERCS and running OpenStack (www.openstack.org)
- OpenStack was started by Rackspace but now includes numerous companies, organizations, and individuals (~6700 people)
- OpenStack provides compute, networking, and storage services (Swift) on top of any major hypervisor (KVM, Xen, VMWare, Hyper-V, etc.)

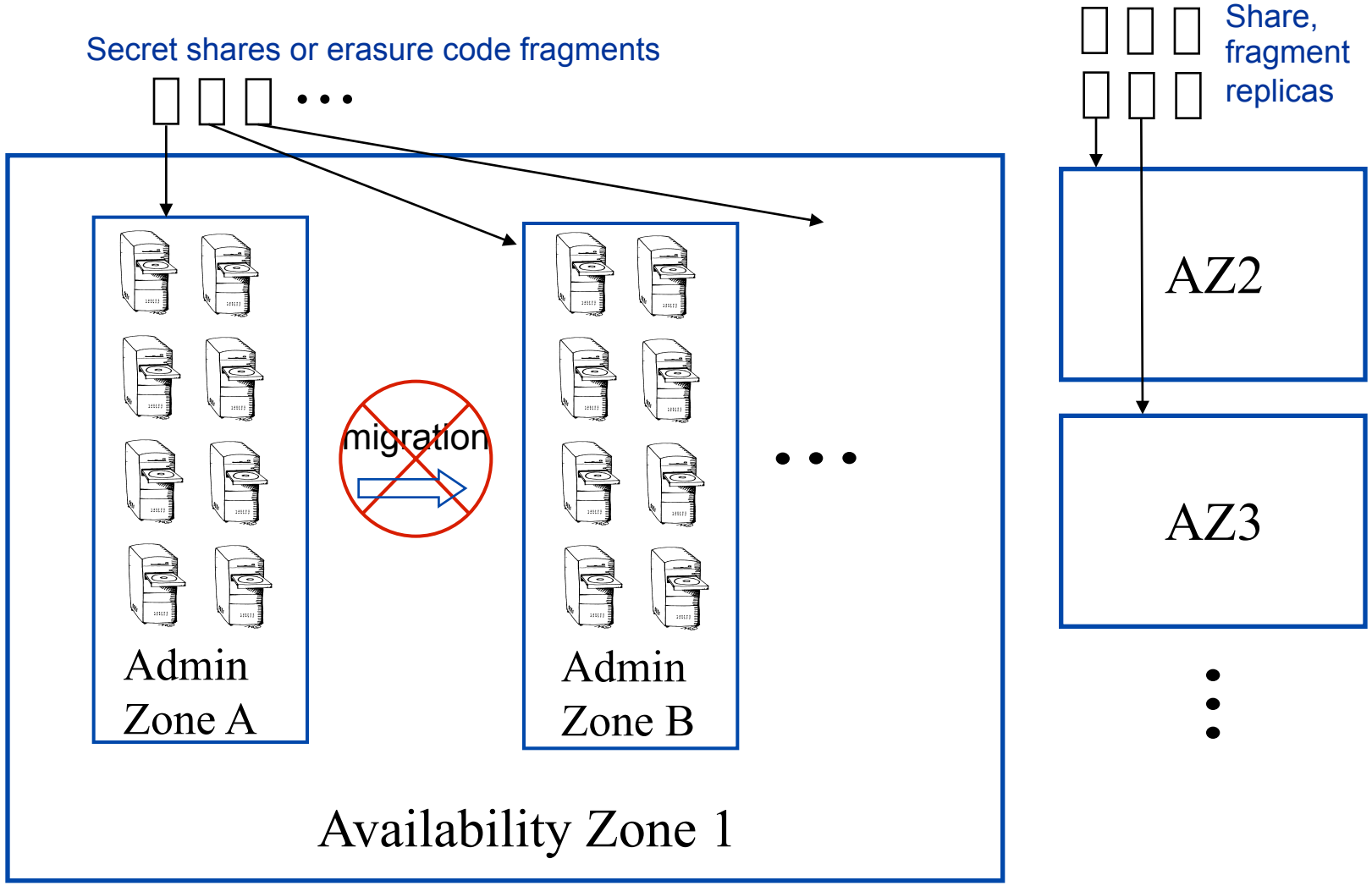
A Flexible and Secure Cloud Object Storage Service

- Cloud service that provides options for data storage to maintain privacy and robustness of sensitive data that needs to be operated on in the cloud
- *Flexibility:* give client control over how data is physically stored
- *Security:* protect against “insider” threats; minimize trust when surrendering data to cloud domain
- *Availability:* maintain “always there” property of cloud storage

Threats

- *Failures*
  - Individual servers, racks, other physical infrastructure
  - Entire data center
- *Attacks*
  - Other cloud users/applications (VM isolation)
  - Cloud provider's administrators
    - Hypervisor-protected memory
    - Separate administrative domains
  - Service-level
    - Service vulnerabilities
    - Service provider's administrator's

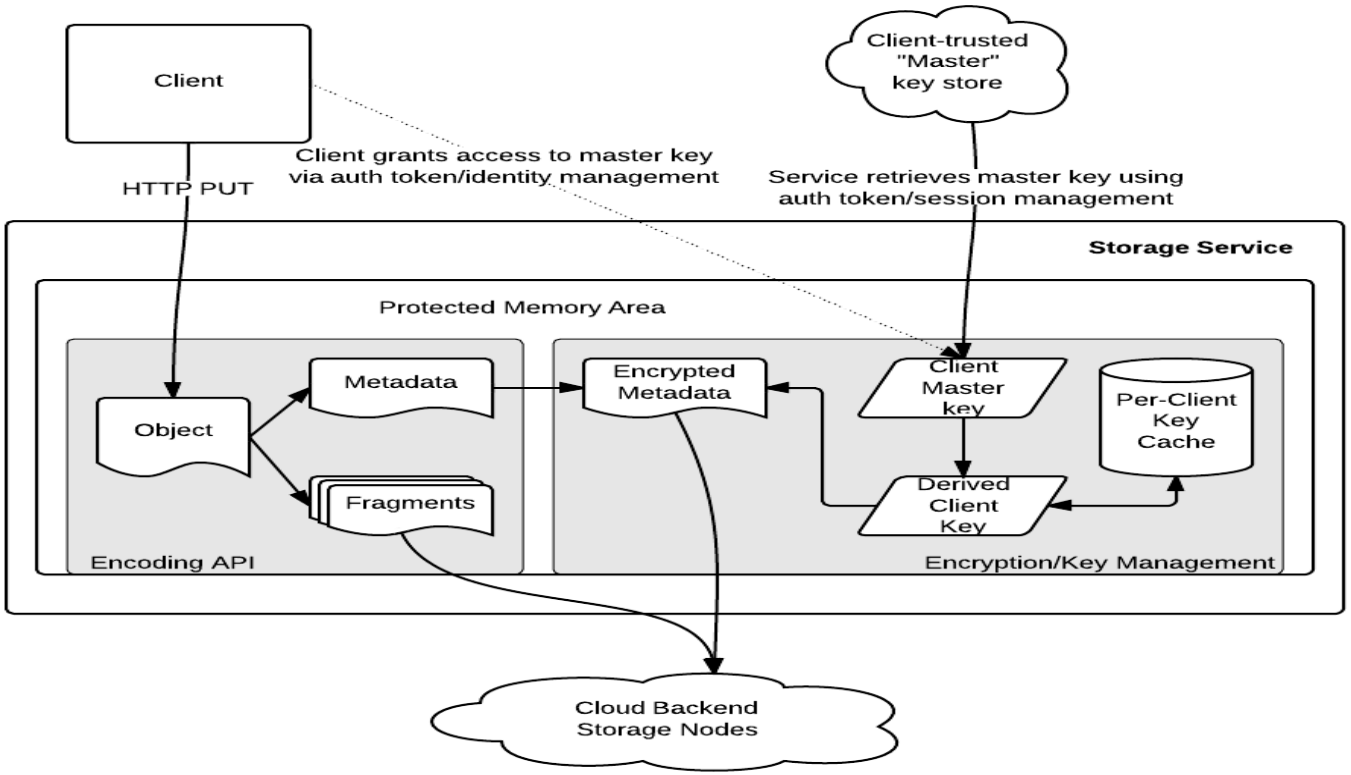
Cloud Storage Node Organization



Example Use Cases

- Strong confidentiality, availability
  - Encrypt data on disk
  - Secret share encryption key(s) across administrative zones (so data can be decrypted and operated on in protected memory)
  - Replicate across availability zones
- Strong availability, moderate confidentiality, better performance
  - Erasure code data across administrative zones
  - Replicate across availability zones

Storage Service Architecture



Metadata about object encoding, fragment locations stored in protected memory and encrypted on disk

Implementation Status

- Modified build of OpenStack Swift 1.9.2
- Client dictates per-bucket encoding
- Modular encoding API
  - Secret sharing: xor (n,n), Shamir (k,n)
  - Erasure coding
  - Replication
- Have demonstrated simple PUT/GET interface with HTTPS communication, DELETE in progress
- Metadata encrypted via 256-bit AES using service master key, per-client key derivation/management in progress

Future Work

- Evaluation of robustness, performance, security
- Modeling protected memory
- New encoding strategies, e.g. “GridSharing”, intelligent fragment placement

Questions?