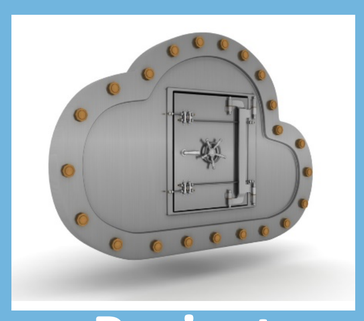


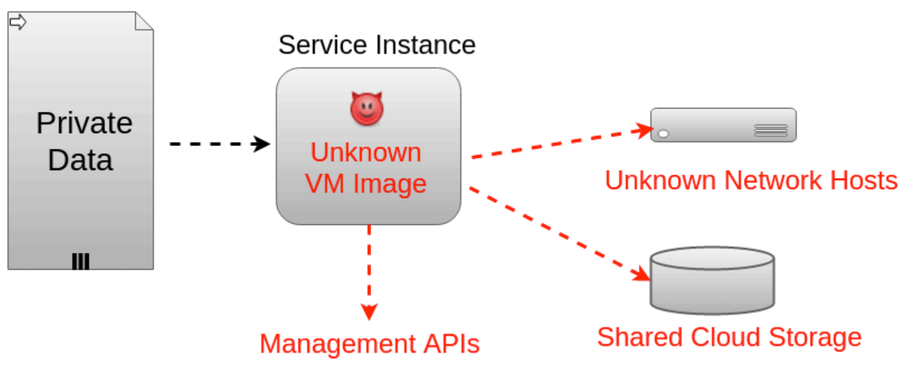
Project Silver

CQSTR: Containing your Data in the Cloud

Yan Zhai, Lichao Yin, Jeffrey Chase, Thomas Ristenpart, Michael Swift
yanzhai@cs.wisc.edu, lichaoyin@gmail.com, chase@cs.duke.edu,
ristenpart@cornell.edu, swift@cs.wisc.edu

Problems and Goals

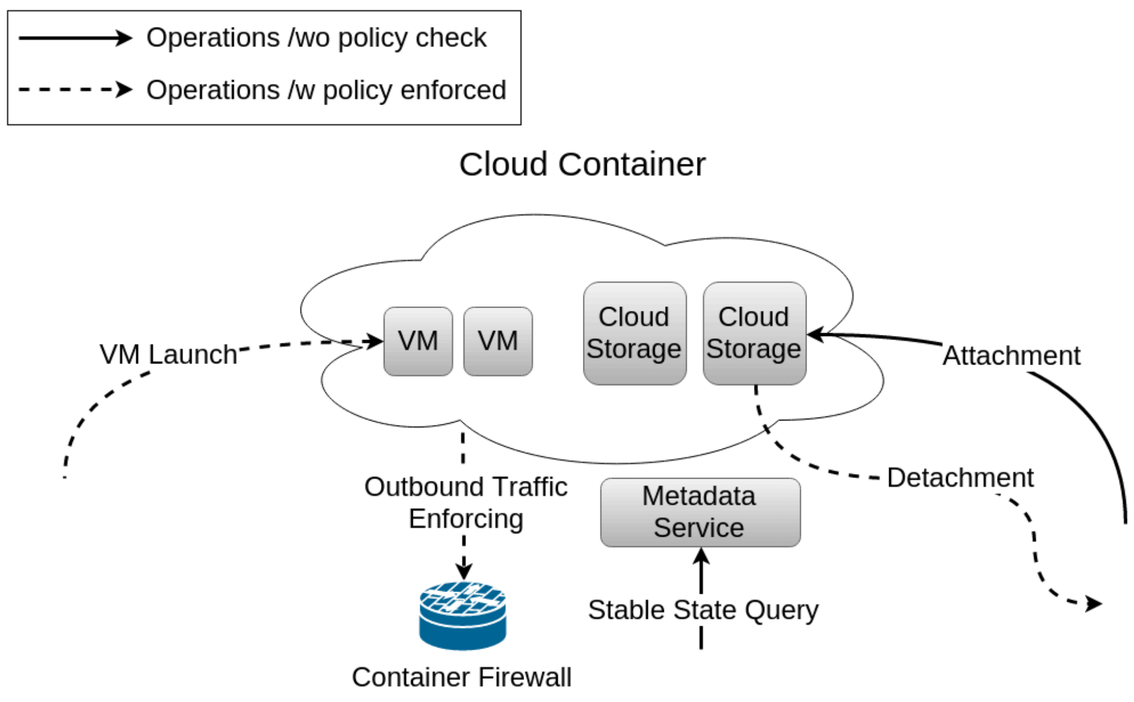
- Outsourcing private data to cloud services is risky



Infrastructure Level Threats for Outsourcing Data

- Goal: improving trust on existing IaaS platform so that
  - data owners can
    - **verify** service setup and data containment
    - tightly **control** data release
  - service providers can
    - deploy **unmodified** applications with simple additional configurations
    - run services with **no performance penalty**

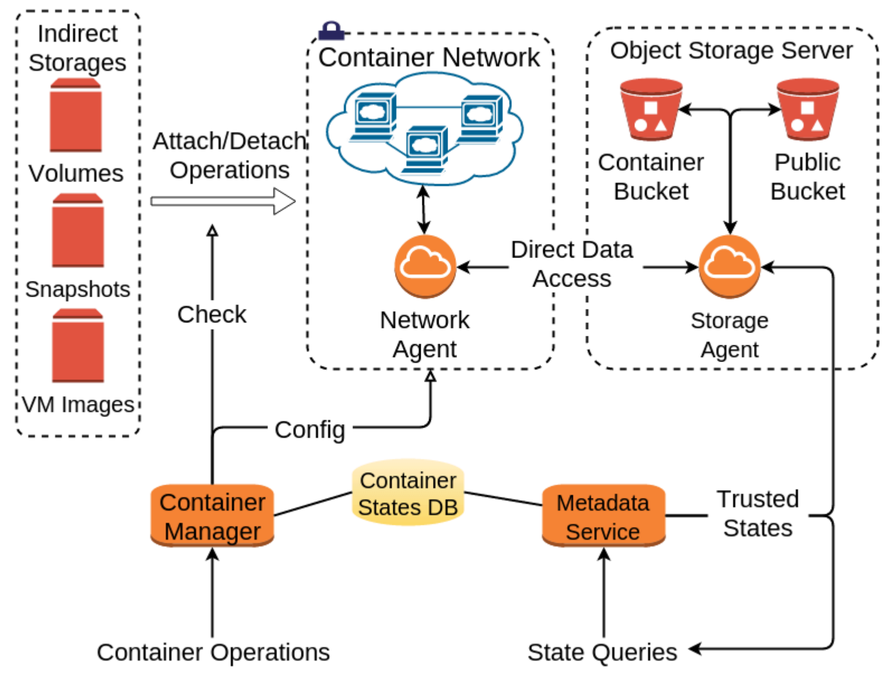
Approach



Cloud Container Data Flow Overview

- New abstraction: **Cloud Container**
  - adapt practical features of *information flow control* at infrastructure level
  - Cloud containers define isolation and containment over cloud VM instances and resources
    - VM restrictions: images, environment
    - Network restrictions: firewall, bandwidth
    - Storage restrictions: accessibility
- **Stable State Assertions**
  - attest status and configuration of a cloud container
  - only monotonic states; no ToCToU problems
- Fix **Cloud API semantics**
  - additional checks to enforce the cloud container rules

Implementation

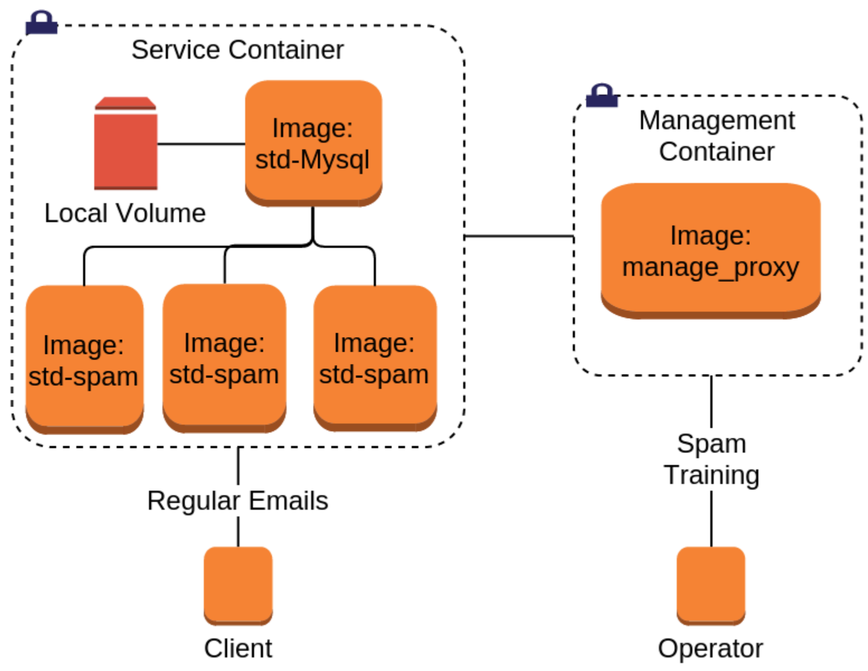


CQSTR Implementation on Openstack Kilo

- CQSTR implemented on Openstack
  - less than 6k LOC
  - fully compatible for non-container usage
  - attestation based access control

Applications and Results

- Ported 3 real applications to CQSTR
  - SpamAssassin, PacketPig, and PredictionIO
  - no application code modification
  - additional “management proxies” added to perform non-data-centric management



Setup of SpamAssassin in CQSTR.
(For remaining application set ups see tech report)

- Low Overhead
  - CQSTR introduced less than **3%** performance difference for all application level experiments
  - microbenchmarks show low cost to frequent operations

- Conclusion
  - CQSTR provides data control over outsourced data by leveraging existing IaaS infrastructure. It can be directly applied to a wide range of data analysis applications and services. It also provides a basement for upper level software to enforce more fine grained control.