Improving Customer Trust in Cloud Services

Nuno Santos  
MPI-SWS

Rodrigo Rodrigues  
MPI-SWS

Krishna P. Gummadi  
MPI-SWS

Stefan Saroiu  
Microsoft Research

1. Problem
- Cloud computing is attractive to many companies
  - Ability to elastically launch virtual machines (VMs) such as Amazon EC2 and GoGrid
  - Can reduce costs by offloading IT to cloud providers
- But, data privacy and integrity is a serious concern
  - Data can leak out or be tampered with by privileged administrators of the cloud provider
  - Surveys show such concerns prevent companies from adopting cloud services
- Our approach:
  - Prevent data access by untrusted cloud nodes
  - Let customers decide which cloud nodes are trusted

2. Key Idea: Policy-Sealed Data
- Primitives restrict access to customer data in the cloud
  - Customer binds data to a policy (sheath)
  - Cloud nodes must satisfy policy to access data (unsheath)
  - Resilient to software admin activity
- Policy specifies configurations of cloud nodes
  - Software configuration (e.g., hypervisor)
  - Physical configuration (e.g., location, hw components)
- Used to design trustworthy services, which can:
  - Protect confidentiality and integrity of customer data
  - Restrict location of data processing
- We build Merlin to implement policy-sealed data

3. Usage Example of Policy-Sealed Data

Cloud Compute Service akin to EC2
- Sw platforms: Xen- or SeL4-based
- Datacenters: US or EU
- Let customers choose preferred configurations

Threats and trust assumptions
- Adversary: software admin
  - e.g., can migrate data, reboot nodes
- Mitigation: cloud service developer
  - Sw platforms restrict admin privileges and sheathe data before migrating data
- Cloud nodes physically secured

4. Merlin: Implementing Policy-Sealed Data

1. Install TPM chips on nodes
   - Provide unique cryptographic key
   - Store fingerprint of sw platform
2. Deploy dedicated monitor
   - Checks node configurations using node TPMs
   - Enforces policy-sealed data with CPABE
3. Upload certificates issued by certifier
   - Map TPM fingerprint to software attributes
   - Map TPM key to physical attributes
4. Monitor checks node configurations
   - Determine TPM key and fingerprint
   - Detect node attributes based on certificates
   - Send CPABE capability with attributes
5. Customer attests the monitor
   - Determine monitor key and fingerprint
   - Validate them against certificate
6. Sheathing and unsheathing
   - Customers sheathe using CPABE public key
   - Nodes unsheathe using CPABE capability

5. Status
- Full implementation and evaluation of Merlin
  - Verification of Merlin protocols using an automated verification tool: ProVerif
  - Proof-of-concept cloud compute service using Eucalyptus, an open-source cloud backend platform