Challenges for Provenance in Cloud Computing

Imad Abbadi and John Lyle

Department of Computer Science
University of Oxford
Outline

1. Why cloud computing? Why provenance?
2. The structure of clouds.
3. Challenges for provenance.
   – Examples.
   – TClouds project.
4. Bonus material:
   – Provenance for mobile privacy and usability.
‘Cloud computing is a model for enabling convenient, on-demand network access to a **shared pool of configurable computing resources** (e.g., networks, servers, storage, applications, and services) that can be **rapidly provisioned** and released **with minimal management effort** or service provider interaction.’
Why cloud computing?

• Popular
  – Low barrier to entry
  – Cost effective
  – Incredibly scalable
  – Resilient and reliable (in theory)
LAST YEAR WE RECOGNIZED THAT OUR PROCESSES WERE FAR TOO COMPLEX

SO WE PUT THEM INTO THE CLOUD

LET THE CLOUDS MAKE YOUR LIFE EASIER

Why provenance?

• Clouds hide complexity
  – Sometimes the complexity matters.
  – Common request: “In which country is my computation happening?”
  – Forensics, billing, security

• Clouds go wrong
  – Errors can be very difficult to track down without provenance
How are clouds structured?

[Saas/PaaS]

Cloud Users

[Infrastrucure properties]

Network layer

Server layer

Physical Layer

Virtual Layer

Application Layer

User Properties

Hospital Service Provider Application

Application Backend (DBMS)

Application Middle-tier
Challenges

• Building a logical sequence of events
  – Involves data from every layer, at multiple time intervals
  – Combining this data currently very difficult, often ad-hoc.
  – Not just storage, but all levels of the cloud.
• Requires common data structures and semantics at all layers
• Need to trust the cloud providers
• Protecting log data
• Not losing the usability benefits
Example scenario

Application Domain

Application Layer

Virtual Layer

Physical Layer

Hosted by $VD_1$

Hosted by $L_1$
TClouds Project

• Building trustworthy, resilient cloud systems
• Two example cases
  – Healthcare
  – Public lighting
• Provenance opportunity
  – Top-down approach (design phase)
  – Bottom-up approach (pragmatic)
Conclusions

• Clouds are really dynamic, and hide a lot of complexity
• Errors, security incidents and privacy requirements require this complexity to be revealed
• Often data and execution provenance doesn’t exist, or could be false
• What is the best approach for providing provenance in the cloud?
Collecting context data in webinos

• Cross-device application environment
  – Mobile, Car, Set-top-box, PC
  – Think Java but for web applications
• Use & creation of contextual data
  – Location, social graph, proximity sensors, etc
  – Shared between devices in a big, synchronised database
• Used for:
  – Better user interface & experience
  – Analytics and advertising
  – Making access control decisions
• Privacy and reliability concerns!
App running on Smartphone

App makes context query

Policy query: access control

Make query for context data

Context Store

Smartphone

Car

Smart TV

PC

Sensor data, location, user data, social network data, ...
Real conclusion

• I have two projects which would benefit from introducing provenance
  – Webinos
  – TClouds

• Can this audience provide any suggestions or warnings?
Thank you.

Any questions?