Management and Application of Data Provenance in the Cloud

Adam Bates, Ben Mood, Masoud Valafar, and Kevin Butler

Computer and Information Science Department, University of Oregon

As organizations become increasingly reliant on outsourced cloud services, concerns over data security and management increase. For example, the need to govern access control at finer granularities becomes particularly important. **Data provenance, the metadata history detailing the derivation of an object, provides the necessary support to address these challenges, but collecting and securing provenance in distributed environments is difficult.** We have proposed components to manage and validate the metadata of a provenance-aware cloud system, and introduced protocols that allow for secure transfer of provenance metadata between end hosts and cloud authorities. Using these protocols, we developed a provenance-based access control mechanism for Cumulus cloud storage, capable of processing thousands of operations per second. This work establishes the practicality of provenance for security-critical cloud applications.

CLOUD PROVENANCE AUTHORITY

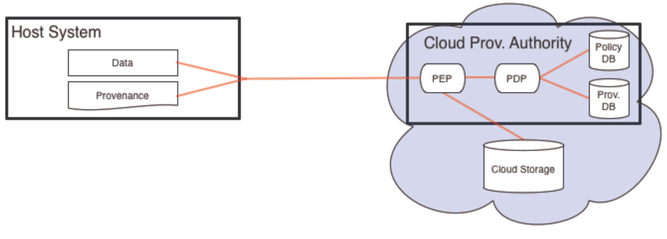

Fig. 1. Cloud Provenance Authority Components.

Cloud Provenance Authority components form an overlay to standard cloud storage. **Policy enforcement points (PEPs)** collect provenance from hosts and act as the arbiters of whether to allow reading or writing to cloud storage. **Policy decision points (PDPs)** are responsible for ensuring provenance validity and policy compliance of data prior to allowing it into cloud storage. **Provenance and Policy databases** are queried by the PDPs.

SYSTEM PROTOCOLS

When a user tries to write to or read from a file in cloud storage, a query consisting of the data and its provenance is sent to the PEP. It checks the integrity of the request and sends it to the PDP, which evaluates it against a set of pre-defined policies before returning an access decision.

Provenance Commitment Protocol

1. $C \rightarrow CPA:\ nonce_C,\ ID_C,\ ID_{Obj}$
2. $CPA \rightarrow C:\ nonce_{CPA},\ Prov_{Obj,t-1},\ Sign[K^-_{CPA}, Prov_{Obj,t-1}||nonce_C]$
3. $C \rightarrow CPA:\ Obj,\ Prov_{Obj,t},\ Sign[K^-_C, Prov_{Obj,t}||nonce_C]$
4. $CPA \rightarrow C:\ Sign[K^-_{CPA}, Prov_{Obj,t}]$

Fig. 2. Provenance Commitment Protocol for Write Operation.

Our protocol makes use of the *secure provenance chain* cryptographic construction to ensure integrity and non-repudiability. To write an object, a Client (C) sends a request to the Cloud Provenance Authority (CPA) with their user ID and an object ID. The CPA responds with a signed copy of the object's existing provenance chain. C then sends a new object version along with updated provenance. The write is committed when the CPA returns a signature for the new provenance. Read operations function similarly.

Provenance Delegation and Retrieval Protocol

Provenance metadata can grow arbitrarily large, even larger than the object it describes. To improve performance and scalability, we introduce a method of on-demand rights delegation between CPAs at different organizations. In this manner, cloud provenance authorities can cooperate with each other to track data. For environments that require end users to inspect provenance data, this interface can extend to allow hosts outside of the CPA to query Provenance Databases (through a PEP gateway).

$$Sign[K^-_{O_0}, Delegation, O_1]$$
$$Sign[K^-_{O_1}, Delegation, O_2]$$
$$Sign[K^-_{O_2}, read, U]$$

Fig. 3. Delegated Read permission from organizations (O_1, O_2) to user U.

PROVENANCE ACCESS CONTROL

Using provenance-based labeling, access control (AC) mechanisms can dynamically adapt to sudden changes in policy, and express access decisions at different levels of granularity. We implemented three lightweight AC mechanisms, including a Bell-LaPadula-like MLS confidentiality model. In evaluation, this mechanism processed 1000 access requests per second, independent of the size of the associated provenance data.

PERFORMANCE

Given the nature of our overlay design, performance was initially poor as the PEP effectively doubled the amount of data transmission. We overcame this limitation by distributing the PEP component, resulting in an overhead of just 14% on read operations.

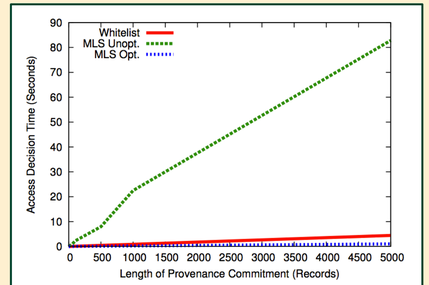

Fig. 4. Performance of different ACs as provenance chain length increases.

We then tested our different ACs against a major web server trace. Through caching, we were able to avoid repetitive parsing of old provenance entries, achieving amortized constant time while still benefiting from provenance context. Reparsing old provenance occurred infrequently after changes to policy.

OSRIS Oregon Systems Infrastructure Research & Information Security Laboratory

Adam Bates, Ben Mood, Masoud Valafar, and Kevin Butler.
Towards Secure Provenance-based Access Control in Cloud Environments.
3rd ACM Conf. on Data and Application Security and Privacy (CODASPY 2013).
San Antonio, TX, USA, February 19, 2012.

Research supported by NSF grant CNS-1118046

For further information, contact Adam Bates (*amb@cs.uoregon.edu*).