Provenance based Security: Towards Designing Provenance Aware Secure Systems

Ragib Hasan, Ph.D.

Assistant Professor, Computer and Information Sciences

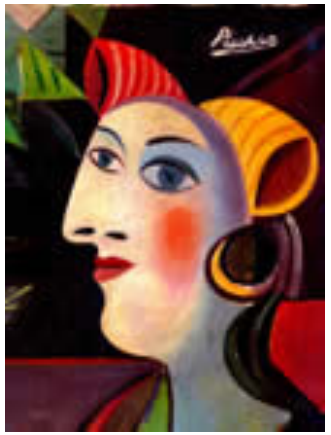
University of Alabama at Birmingham

secret.cis.uab.edu | **ragibhasan.com** | **ragib@uab.edu**

2016 NSF CyberSecurity Summit

Let’s play a game

Real, worth $**101.8** million



Fake, listed at eBay, worth nothing

Can you spot the fake **Picasso**?

So, how do art buyers **authenticate art**?

Among other things, they look at **Provenance records**

Provenance: from Latin *provenire* 'come from', defined as

> "(i) *the fact of coming from some particular source or quarter; origin, derivation.*
>
> *(ii) the history or pedigree of a work of art, manuscript, rare book, etc.; a record of the ultimate derivation and passage of an item through its various owners*" (Oxford English Dictionary)

In other words, **Who** owned it, **what** was done to it, **how** was it transferred …

Widely used in arts, archives, and archeology, called the Fundamental Principle of Archival

L'artiste et son modèle (1928), at Museum of Modern Art

http://moma.org/collection/provenance/items/644.67.html

What is the provenance of data?

Definitions

- Description of the **origins** of data and the **process** by which it arrived at the database. [Buneman et al.]
- Information that helps determine the **derivation history** of a data product, starting from its original sources. [Simmhan et al.]

- Unlike physical objects, digital data can be modified as it changes hands
- Besides custody or lineage, we also need to record transformation

Provenance in large scale systems

The "Why"?

- **Trustworthiness** – knowing the sources and owners over time helps determine the trustworthiness
- **Data Quality** – output data depends on input data and actions taken on data
- **Replication** – a recipe for recreating data
- **Informational** – learn how data was derived
- **Audit Trails** –proof of due diligence; investigation of problem sources, system intrusions, proof of prior work in patents

Issues and Challenges of provenance in large scale systems

- **Collection:** Where is provenance collected? At what granularity? Who collects it? Is the collector trustworthy? Can they maliciously misreport provenance and get away undetected?
- **Processing:** How is provenance data processed? Is it aggregated before storage?

- **Storage:** How and where is provenance stored? (With data? Centrally?) What is the performance?
- **Security:** How is provenance secured? Can the owner of the data object modify it? Delete it? How do we verify the integrity? How do we ensure confidentiality and availability?
- **Retrieval:** How fast can we retrieve provenance?
- **Usage:** What is the provenance used for?

Two aspects of provenance security

- **Securing provenance:** How can we secure provenance to provide integrity, confidentiality, and availability assurances for the provenance data

- **Provenance-based security:** How can provenance be used for security related decisions

Ragib Hasan, Radu Sion, and Marianne Winslett, "Introducing Secure Provenance: Problems and Challenges", In Proceedings of ACM StorageSS, Alexandria, VA, October 2007.

U. Braun, A. Shinnar, and M. Seltzer. Securing Provenance. In Proceedings of the USENIX Workshop on Hot Topics in Security (HotSec), San Jose, CA, 2008



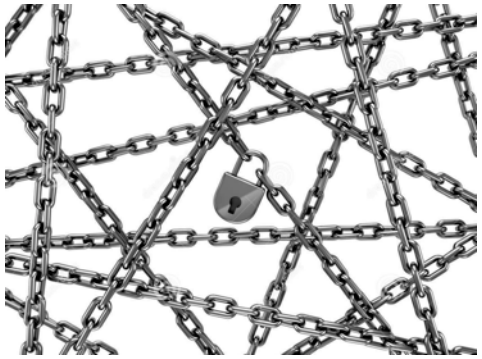
Securing Provenance

Challenges in securing provenance

- **Tampering:** In distributed settings, it is difficult to prevent different parties from tampering with provenance
- **Replacement:** It is possible to create a fake but entirely plausible provenance history by replacing provenance entirely
- **Confidentiality:** Provenance history itself should not leak confidential information
- **Availability:** Removal of provenance should be prevented
- **Performance:** Provenance Security mechanisms should not make the system performance go down

Ragib Hasan, Radu Sion, and Marianne Winslett, "Introducing Secure Provenance: Problems and Challenges", In Proceedings of ACM StorageSS, Alexandria, VA, October 2007

Solution approaches



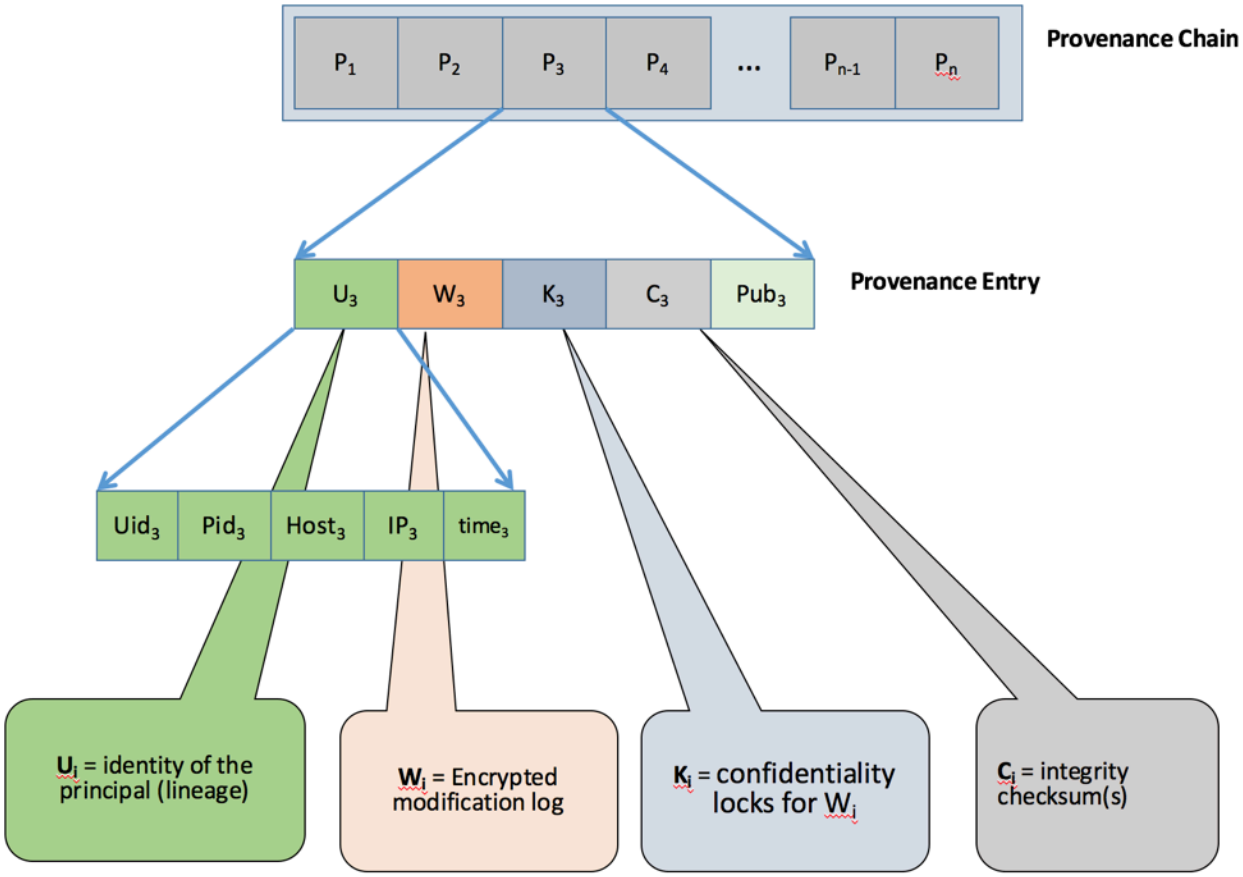
Local solution: provenance chains

- Maintain **provenance chain** as metadata for each data object/file
- Have each user who edits the data append a **provenance record** to the provenance chain, and also sign it
- Use a **signed hash** to protect chronology of the chain

Ragib Hasan, Radu Sion, and Marianne Winslett, "Preventing History forgery with Secure Provenance", ACM Transactions on Storage, Vol. 5, Issue 4, pp. 12:1-12:43, December, 2009, ACM Press.

Ragib Hasan, Radu Sion, and Marianne Winslett, "The Case of the Fake Picasso: Preventing History Forgery with Secure Provenance", In Proceedings of the 7th USENIX Conference on File and Storage Technologies (FAST 2009)

SPROV: Secure Provenance Chains for local files



Pros: Provides integrity and confidentiality assurances, any modification or tampering after the fact can be detected, only authorized users can access provenance data

Cons: Cannot prevent replacement attack where attacker replaces entire provenance chain with fake but plausible history, Hard to maintain in distributed setting

Distributed/Cloud based provenance

- A cloud based large scale system poses special challenges
- There are multiple parties involved: users, cloud service providers, and potential attackers

- The cloud itself may be a malicious party
- Must ensure the provenance already collected cannot be replaced

Shams Zawoad and Ragib Hasan, "SECAP: Towards Securing Application Provenance in the Cloud", In Proceedings of the 9th IEEE International Conference on Cloud Computing (IEEE CLOUD), San Francisco, CA, July 2016.

Three types of provenance in cloud based large scale systems

Data provenance:

- Provenance of the data produced in different parts of the cloud

Application provenance:

- Provenance of applications running on, and migrating around in the cloud

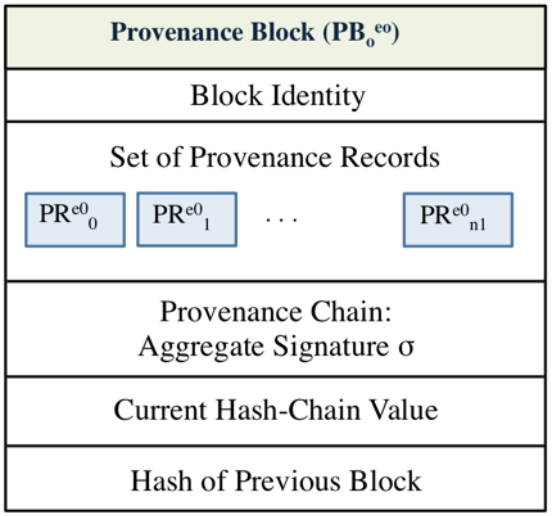
System provenance:

- Provenance of the cloud itself, its own configurations/state etc.

Preventing deletion attacks

- **Solution:** Entangle provenance chains/ records so that the cloud provider cannot go back and replace provenance

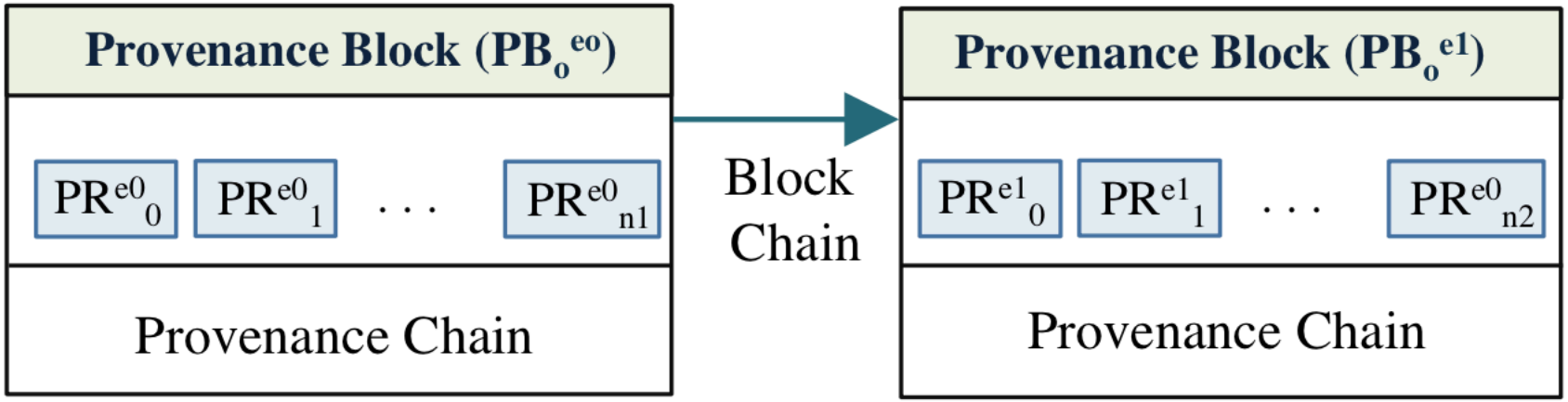
- **How?** Group provenance chain entries from same epoch into blocks, Have the cloud provider periodically and publicly publish hashes of current provenance chain blocks, have a third party monitor the hashes
  - Cloud provider cannot go back and modify the chain, the modification will break the entangled entries



Shams Zawoad, Amit Dutta, and Ragib Hasan, "Towards Building Forensics Enabled Cloud Through Secure Logging-as-a-Service", IEEE Transactions on Dependable and Secure Computing (TDSC), SI-Cyber Crime, vol. 13, no. 2, pp. 148-162, 2016.

Further improvements: Block chains

In distributed large scale systems, managing provenance chain integrity can be done by utilizing the **Block chain** technology



Other issues

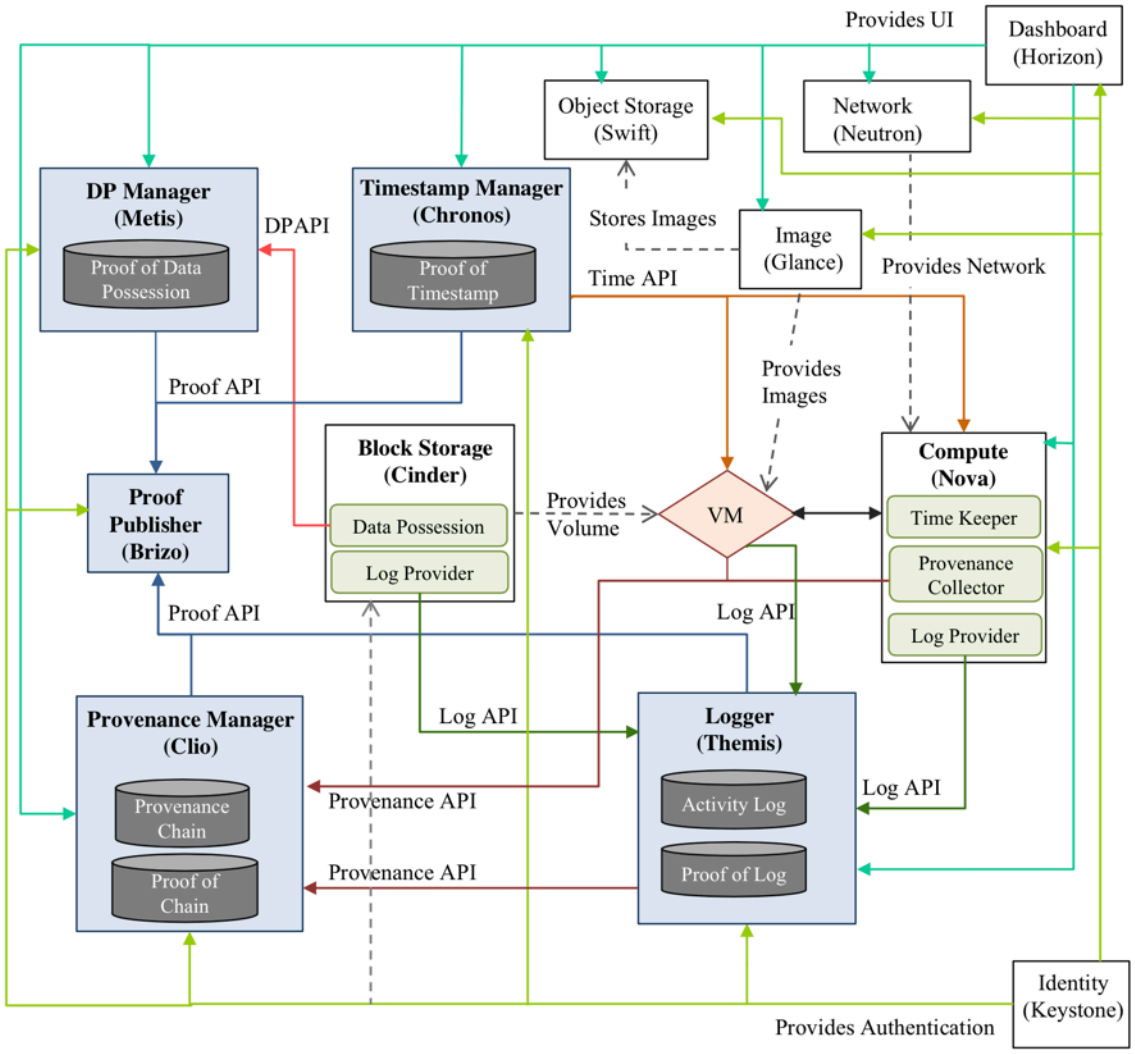
- **Timestamp:** In distributed environments, timestamps of provenance entries can be misleading or untrustworthy
- **Confidentiality:** Chronology can be protected In a privacy preserving method by using Bloom filter or RSA accumulators

- Current solutions prevent "regret"-based attacks, but do not prevent **future attacks** (i.e., after the various parties become malicious, they can intentionally misreport provenance)
  - Hardware attestation based approaches might be a solution

Shams Zawoad and Ragib Hasan "Chronos: Towards Securing System Time in the Cloud for Reliable Forensics Investigation", In Proceedings of the 40th IEEE COMPSAC, Atlanta, Georgia, June 2016.

Ragib Hasan, Rasib Khan, Shams Zawoad, Md Haque, "WORAL: A Witness Oriented Secure Location Provenance Framework for Mobile Devices", IEEE Transactions on Emerging Topics in Computing (TETC), vol. 4, no. 1, pp. 128-141, 2016

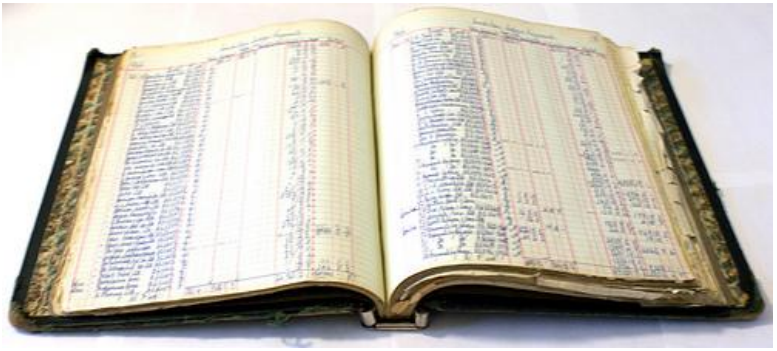
Bringing it all together:
Provenance Aware OpenStack



Created by adding 5 modules on top of OpenStack, and provides **automated provenance management** for applications and data

Creates various **security proofs** for provenance, data possession, and log integrity

Includes **secure timestamp** management

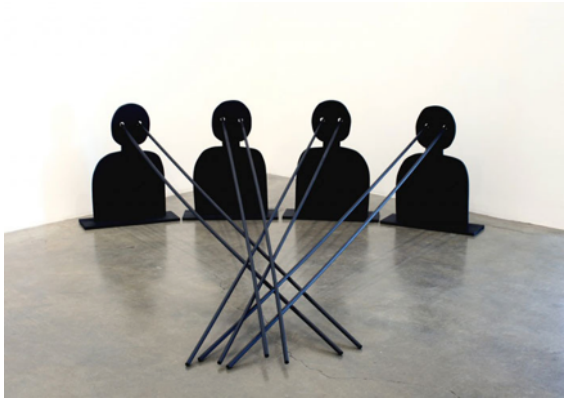




Provenance-based Security

Provenance based security

- Can we use **provenance for security**?
- **Yes we can**! That's because **all** of our security decisions are **causal effects of past events**
  - Authentication: We authenticate based on past interactions
  - Access control and authorization: We don't do access control based on what hasn't happened! In other words, all access control is based on some event in the past

A simple shift of view shows everything can be represented via provenance



- A password can be represented by the "memory" or provenance of setting the password
- A biometric can be represented by the "memory" or provenance of registering the biometric
- An access control list entry can be represented by the provenance history of setting the entry

- Clearly, **every security related object** or item can be represented via **provenance of an event of the past**

The unifying concept: Interaction provenance

Definition

The **secure**, **chronological** and **order-preserving** sequence of interactions and events between two of more parties

Rasib Khan and Ragib Hasan, "Fuzzy Authentication using Interaction Provenance in Service Oriented Computing", the 12th IEEE International Conference on Services Computing (SCC), New York, USA, June 2015

Using Interaction provenance as authentication factors

Example 1: Username/Password registration event

Event Type: Password Registration
PRINCIPAL$_1$: *Alice*, PRINCIPAL$_2$: *ServiceProvider* INTERACTION: [*Alice*, *ServiceProvider*, REGISTRATION [(USERID: *alice*), (PROVIDERID: *ServiceProvider*), (PASSWORD: *password*), (REGTIME: *timestamp*)]]

Later, the user presents the provenance of this interaction to authenticate

This example is very similar to social authentication!

Advantages include the ability to use temporal factors, strength of interaction, indirect interactions

Interaction provenance for access control

- Similarly, interaction provenance of past access or being granted access can be used for access control

- Another example is Path-based access control, where path provenance of the application or user is considered in access control

J. Park, D. Nguyen and R. Sandhu, "A provenance-based access control model," Privacy, Security and Trust (PST), 2012 Tenth Annual International Conference on, Paris, 2012, pp. 137-144

Rasib Khan and Ragib Hasan, “Fuzzy Authentication using Interaction Provenance in Service Oriented Computing”, the 12th IEEE International Conference on Services Computing (SCC), New York, USA, June 2015

Advantages of provenance based security

- A unified single mechanism for security
- Provides flexibility and new methods and paradigms of authentication, access control
  - Access policy may allow access if provenance of similar access shown in other systems
  - Authentication based on multiple factors can be combined seamlessly
  - Allows consideration of temporal factors (how long) and qualitative factors (how strong)
- Very useful for service composition with heterogeneous systems

Ideal world of provenance

- In an ideal world, **everything**, ranging from data objects to memory to network packets should have provenance
- Systems should provide **automatic support** for provenance
- Having such provenance will allow achieving trustworthiness, gaining and evaluating quality
- We can also have true rollback for every object

Margo Seltzer, "The State of Provenance", TaPP 2009.

Providing system support for provenance: the Middleware approach

PASS

- Provenance aware storage system: A library for collection and storage of provenance

Provenance kernel hooks and LSM:

- Enhancing Linux based systems to collect provenance via kernel hooks and Linux Security Modules

Kiran-Kumar Muniswamy-Reddy, David A. Holland, Uri Braun and Margo I. Seltzer. "Provenance-Aware Storage Systems." *USENIX* (2006).

Bates, Adam M., Dave Tian, Kevin R. B. Butler and Thomas Moyer. "Trustworthy Whole-System Provenance for the Linux Kernel." *USS* (2015).

Open problems and future work

- **Performance** and **system i**ssues: How do we make provenance collection efficient
- Provenance **explosion:** Based on granularity, provenance may grow orders of magnitude larger than actual data
- Provenance **holes**: How do we handle systems or users that do not record any provenance?
- Provenance **applications**: Supply chain provenance, IoT provenance, Big Data provenance

Acknowledgements

Sponsors:

- National Science Foundation CAREER Award CNS-1351038
- Department of Homeland Security Grant FA8750-12-2-0254.

Collaborators:

- Marianne Winslett (UIUC), Radu Sion (Stony Brook), Rasib Khan (Northern Kentucky), Shams Zawoad (Visa)

Contact: Ragib Hasan ragib@uab.edu
Web: UAB SECRETLab (http://secret.cis.uab.edu)