SPICE
Simple Privacy-Preserving Identity-Management for Cloud Environment

Sherman S.M. Chow
Yi-Jun He
Lucas C.K. Hui
Siu Ming Yiu

Applied Cryptography and Network Security (ACNS ‘12)
Session 12: Security and Privacy in Cloud Systems
June 29th, 2012
• Cloud, and Digital Identity Management (DIM)
• Existing DIMs and their Limitations
• Our Goals
• World of Group Signatures
• SPICE!
• Simple Showcase
Moving into the Cloud

- Cloud computing has been envisioned as the next-generation architecture of IT enterprise
  - On-demand self-service
  - Ubiquitous network access
  - Location independent resource pooling
  - Rapid resource elasticity
  - Usage-based pricing
  - Transference of risk
An organization employs the cloud service
Many members belong to this organization
Many cloud service providers

Authentication expected
Threats to Cloud Computing

- Abuse and Nefarious Use of Cloud Computing
- Insecure Interfaces and APIs
- Malicious Insiders
  - “You cannot always trust your cloud provider’s employees.”
- Shared Technology Issues
- Data Loss or Leakage
- Account or Service Hijacking
  - “Identity theft -- it’s not just a consumer problem.”
- Unknown Risk Profile
- [Cloud Security Alliance (CSA), March 2010]
Digital Identity Management
Desirable Properties

- Unlinkability
- Delegatable Authentication
  - Each CSP has its own way
  - Still Unlinkable
  - Useful in Cloud Environment
- Anonymity
- Accountability
- User Centric Access Control
- Single Registration
Bertino et al.’s DIM System

- Registrar stores a set of signatures
  - each sign on a commitment of a user’s attribute
- User first retrieves (some of) them
  - according to CSP’s need
- CSP verifies user’s Zero-Knowledge Proof (ZKP)
- CSP then generates a credential to the user
- User may present this to another CSP
- Mechanism to deal with Heterogeneous Naming
Limitations

- Registrar remains (practically always) online
  - Single point: Scalability and Security
- Signatures are linkable
  - ZKP of signatures? Yes, but stay tuned
- Delegation (of authentication / verification) is not supported
  - Not to say in an unlinkable way
Shibboleth

• Open source reference implementation
  – A similar authentication mechanism
• Single sign-on (~delegatable authentication)
  – CSPs form a federation a priori
  – User authorizes CSP\(_1\) to access CSP\(_2\) on behalf
  – Registrar needs to be online during the access to issues two certificates to each of CSPs
  – Unlinkability is not a concern
Our Goal

- Unlinkability and Delegatable Authentication
- Secure yet Efficient
- Simple
- Privacy-Preserving
- Identity-Management for
- Cloud Environment
Group Signatures

- Group-oriented signatures with anonymity
- A group manager (GM) issues credentials
- Any member can sign for the group
  - remain anonymous within the group
  - signatures are unlinkable
  - but, unconditional anonymity may be abused
- An opening authority can “open” a group signature to reveal its true signer
World of Group Signatures

“Traceable Signatures”  
[KTY04]

“Real Traceable Signatures”  
[Chow09]

“Double-Trapdoor Anonymous Tag”  
[ACHO11]

Efficient Tracing

+ Authorship Claiming

+ Anonymous Tracing

+ Authorship Deniability

Verifier-Local Group Signatures  
(tracing)

Group Signatures  
(opening)

(Slides Courtesy: Masayuki Abe)
Basic Algorithms

- Setup() $\rightarrow$ public key = $pk$, secret = $(msk, osk)$
- UG($msk$, $id$) $\rightarrow$ user secret key $sk_i$
- Sign($sk_i$, $m$) $\rightarrow$ $\sigma$
- Verify($\sigma$, $m$) $\rightarrow$ “True”/”False”
- Open($\sigma$, $osk$) $\rightarrow$ $id$
Additional Algorithms

- $\text{Sign}(sk_i, m_1, m_2, ..., m_n) \rightarrow \sigma$
- $\text{Sign}(\sigma_n, m_{n+1}) \rightarrow \sigma_{n+1}$
- $\text{Hide}(\sigma, m_j, j) \rightarrow \sigma'$
- $\text{Rand}(\sigma) \rightarrow \sigma'$
Simple Mechanism

- Registrar acts as the GM
- Multiples groups / instances of GrpSig
  - Users as the members
  - CSP as the members (more details later)
- Certificate = GrpSig on $n$ attributes
- Authentication = Append message $m$ to cert.
- Hide some blocks aka attributes if needed
- Re-randomize it
Type of Attributes

• Sensitive personal info.:
  – relatively stable
  – a common representation across different CSPs

• Service-specific attributes:
  – CSPs may employ heterogeneous naming

• Irrelevant attributes:
  – e.g., logging in social network (many info) for external location-based services (mainly location)
Heterogeneous Naming

- Considered in Bertino et al.’s work
- Syntactic
  - e.g., ID vs. “Identity”
- Terminological
  - e.g., “email address” vs. “email account”
- Semantic variations
  - e.g., “privacy level” vs. “sharing setting”
- Matching techniques
  - Dictionaries (WordNet 2.1 English Lexical DB)
  - Ontology mapping (Falcon-AO)
Resolving Naming Variations

- Registrar needs to know who are concerned with a selected subset of attributes
  - but not the ontology mapping between all CSPs
- “Related” CSPs form a group
- Append a group signature on the new “name”
- Some initial trust on the source CSP
- Open the signature if necessary
Randomization / Naming Conversion

Sensitive personal info.  Service-specific attribute  Irrelevant attribute

$A_1$  $A_2$  $A_3$

$A'_1$  $A'_2$  $A'_3$

$A_1$  $A_2$  $A_3$

$A_1$  $A_2$  $A_3$
Attribute-Hiding

Sensitive personal info. | Service-specific attribute | Irrelevant attribute

\[ A_1 \quad A_2 \quad A_3 \rightarrow A_1 \quad A_2 \quad A_3 \]
Design of Group Signatures

- Group signature is just a non-interactive zero-knowledge (NIZK) proof of an underlying “regular” signature
- Two-level structures
  - 1\textsuperscript{st} level is user identity (hidden)
  - 2\textsuperscript{nd} level is actual message (by appending)
Our Extensions

- The regular sig. should be re-randomizable
  - E.g., Waters signatures
- The NIZK proofs should be re-randomizable
  - Groth-Sahai proof system
- Extending $pk$ for a hierarchy of msg
  - 2-level to $n$-level
- Hiding can be achieved like how the 1$^{st}$ level (signing on the user’s ID) is hidden
SPICE for Web Authentication

Client's Web Browser

1. Service request
2. Auth. request
3. Randomized certificate

Source CSP
- Policy Repository
- Naming Management Service
- Request Management
- Vocabulary Conflicts Handler
- Randomized Certificate Verifier GS.Ver()
- Sanitized Certificate Issuer GS.Hide(), GS.Rand()

Receiving CSP
- Policy Repository
- Naming Management Service
- Request Management
- Sanitized Certificate Verifier GS.Ver()

User
- Attribute Record Vocabulary
- Vocabulary Conflicts Handler
- Randomized Certificate Issuer GS.Sign()

Registrar
- Attribute Record Storage
- Credential Issuer GS.UG(), GS.Sign()
Summary of Our Results

• Privacy and security have become a critical concern
  • w/ immense growth in the popularity of cloud computing
• Digital ID. Mgt. (DIM) is a critical component
• We proposed a privacy-aware interoperable DIM system for the cloud
  • solved two open problems left by Bertino et al.
  • (unlinkability and delegatable authentication)
Summary of Our Techniques

- Our scheme relies on the conceptually simple use of extended group signatures
- Most part of the operations can be performed offline
- We remove the need of contacting the registrar before every authentication
  - or storing a large amount of certificates
- We believe the overhead is quite minimal for the privacy concern
- smchow@math.uwaterloo.ca