Secure Data Preservers for Web Services

Byung-Gon Chun
Yahoo! Research

Joint work with
Jayanthkumar Kannan (Google) and Petros Maniatis (Intel Labs)
Users Entrust Web Services with Their Data

- Credit card number
- Trading strategy
- Health records
- Web click logs
Users Entrust Web Services with Their Data

- Credit card number
- Health records

- How their data will be used
- What parts will be shared
- With whom they will be shared
Exposure of Sensitive Data

• dataloss.db lists 400 data loss incidents in 2009; on average exposed half-a-million customer records
Exposure of Sensitive Data

- dataloss.db lists 400 data loss incidents in 2009; on average exposed half-a-million customer records
Exposure of Sensitive Data

- dataloss.db lists 400 data loss incidents in 2009; on average exposed half-a-million customer records
Exacerbated by Giving Up Data Usage Control

Individuals

Health records
Exacerbated by Giving Up Data Usage Control

Individuals → Health records

WebMD
Exacerbated by Giving Up Data Usage Control

- How their data will be used
- What parts will be shared
- With whom they will be shared
Give Control Back to Users

- How their data will be used
- What parts will be shared
- With whom they will be shared

Personalizable trust

Individuals

WebMD

Health records
Roadmap

• Motivation
• Secure Data Preserver
• Design
• Evaluation
Our Approach

• Entrusting raw data violates least privilege

• Encapsulate sensitive data and enforce well-defined interface for service to access data
Secure Data Preserver (SDaP)

(a) Service + User Data

Service

Service Code

← User Data

OS

HW

(b) Service + Preserver

Service

Service Code

Data Interface

OS

HW

Preserver

Preserver Code

User Data

isolation boundary

access control
Preserver Deployment Scenarios

Trusted third party or client

Service app
OS
HW

SDaP
OS
HW

Co-location

Service app
OS
VMM
HW

SDaP
Mini-OS

Faulty service app
Faulty service operator

Faulty service app
Faulty service operator

Secure co-processor
What Apps Are Suitable?

• Sensitive query
  – User provides sensitive query, service provides data stream
  – E.g., Trading, Health

• Analytics on sensitive data
  – Service performs data mining on user’s sensitive data
  – E.g., Targeted advertising, Recommendation

• Proxy
  – User provides credentials to another service
What Apps Are Suitable?

• **Sensitive query**
  – User provides sensitive query, service provides data
  
  * Limitation

• **Data-centric service** reading and updating users’ data at fine granularity
  – E.g., Docs, Social networking apps

• **Proxy**
  – User provides credentials to another service
Roadmap

- Motivation
- Secure Data Preserver
- Design
- Evaluation
Preserver Design Goals

- Simple Interface
- Flexible deployment
- Fine-grained use policy
- Trust but mitigate risk
Preserver Operational View

1. Pick Preserver
2. Specify policy
3. Install Preserver

Preserver
Policy
Data

E*Trade
app
OS

4. API
Ticker()
Preserver Architecture

Hosting  Invocation  Transformation

Client  Service  Preserver

Service  Host Hub

OS  Service Data

Data Layer

User Data  Interface  Policy Data

Policy Engine  Base Layer  Host Facilities

Install  Invoke  Install/xform
Preserver Hosting

- Which services can host users’ preservers
- Hosting policy
  - Declarative language based on SecPAL
  1. *alice* SAYS CanHost(M) IF OwnsMachine(*amazon*, M)
- Hosting mechanism
  - Hosting protocol based on Diffie-Hellman protocol
Preserver Hosting

• Which services can host users’ preservers
• Hosting policy
  – Declarative language based on SecPAL

2. *alice* SAYS CanHost(M) IF TrustedService(S), OwnsMachine(S,M), HasCoprocessor(M)

• Hosting mechanism
  – Hosting protocol based on Diffie-Hellman protocol
Preserver Hosting

• Which services can host users’ preservers
• Hosting policy
  – Declarative language based on SecPAL

3. alice SAYS amazon CANSAy TrustedService(S)

• Hosting mechanism
  – Hosting protocol based on Diffie-Hellman protocol
Preserver Invocation

• Constrain interface invocation parameters with SecPAL
• Two kinds: stateless, stateful

1. *alice* SAYS CanInvoke(*amazon*, A) IF LessThan(A, 50)

• Transfer of invocation policies: exo-leasing
Preserver Invocation

- Constrain interface invocation parameters with SecPAL
- Two kinds: stateless, stateful

2. *alice* SAYS CanInvoke(*doubleclick*, A) IF LessThan(A, Limit), Between(Time, ”01/01/10”, ”01/31/10”) STATE (Limit=50, Update(Limit, A))

- Transfer of invocation policies: exo-leasing
Preserver Invocation

• Constrain interface invocation parameters with SecPAL
• Two kinds: stateless, stateful

3. *alice* SAYS *amazon* CANSA Y CanInvoke(S,A) IF LessThan(A,Limit) STATE (Limit=50,Update(Limit,A))

• Transfer of invocation policies: exo-leasing
Preserver Transformation

- Filtering: retain a subset of data
  - E.g., only the web history in the last six months

- Aggregation: merging of raw data from mutually trusting users of a service
  - E.g., ad-click history of users
Roadmap

- Motivation
- Secure Data Preserver
- Design
- Evaluation
Evaluation

• Deployment options:
  – TTP, client, Xen-based co-location

• Three sample preservers:
  – Stock trading, targeted advertising, credit card xact

• Main results:
  – Cost of preserver
  – Comparison of deployment options
  – Security analysis: LS2-based theoretical analysis, Trusted Computing Base (TCB) comparison
Cost of Basic Invocation (Latency)

![Graph showing latency versus invocation size for different cases: TTPCase, XenCase, and BaseCase.]
Cost of Stock Trading (Latency)

![Graph showing latency against number of invocations for different cases: TTPCase, XenCase, BaseCase. The graph illustrates an increasing trend in latency with the number of invocations.](image-url)
Discussion

• Find appropriate interfaces, verify them

• Easy refactoring
  – Even automated

• Apps with rich interfaces
  – Information flow control
Related Work

• Wilhelm’s mobile agent

• CLAMP

• BSTORE

• Decentralized privacy frameworks

• Information flow control
Conclusion

• Rearchitect web services around the principle of giving data usage control back to users

• Secure Data Preserver achieves this goal via data encapsulation and interface-based access control
Thank you!

Q & A