Automaton Segmentation: A New Approach to Preserve Privacy in XML Information Brokering

Fengjun Li, Bo Luo, Peng Liu, Dongwon Lee and Chao-Hsien Chu

College of Information Sciences and Technology
The Pennsylvania State University

Outline

• Motivation
• Solution
  – The broker-coordinator overlay approach
  – Automaton segmentation scheme
  – Query encryption scheme
• Evaluation
• Conclusion

Motivation

Information Brokering Scenario

Naïve approach

Privacy Threats

Privacy Threat: Curious Insider
Our Solution: Overview

- To block probing queries
  - in-network access control
- To protect data location privacy, patient location privacy, and metadata privacy
  - automaton segmentation
- Defeat all the aforementioned privacy threats.
- Achieve superior privacy protection.

Preliminary: How the proxy works?

- **Routing rules**

  \[ R^{index} = \{object, \text{destination(s)}\} \]

  - Object is an XPath expression
  - Destination is an IP address
  - Routing rules represent physical distribution of data objects
Example Routing Rules

R1: (/site/people, 192.168.0.2)
R2: (/africa/item, 192.168.0.15)
R3: (/asia/item, 192.168.0.16)

Solution

Evaluation

Conclusion

An Example Routing Automaton

• Query routing: one-to-many XPath matching.
• **Routing automaton**: A Non-deterministic Finite Automaton that captures routing rules.

R1: (/site/people, 192.168.0.2)
R2: (/africa/item, 192.168.0.15)
R3: (/asia/item, 192.168.0.16)

Solution

Evaluation

Conclusion

How to block probing queries?

Global Routing Automaton

Access Control Automaton

Integrated Global Automaton

The Integrated Automaton

Integrated Automaton

System Architecture

Brokers and Coordinators

• **Brokers**
  - Connect user
  - Forward query to the root-coordinator
  - Before forwarding, do pre-protection

• **Coordinator**
  - Root-coordinator, coordinator, and leaf-coordinator
  - They form a coordinator tree
  - The leaf-coordinator does not hold any automaton piece, but the other two do.

• **The Super Node**
  - Initiation and offline maintenance
Motivation

Broker 3 → root-coordinator

Query Segment Encryption

Bob → "/*/regions/asia/item[&name='Abacavir']/location"

Broker 3 → "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"

Broker 3, "/*/regions/asia/item[&name='Abacavir']/location"

Broker 3, "xxxxxxxxxxxxxxxxxxxxx/asia/item[&name='Abacavir']/location"

Broker 3, "xxxx/regions/asia/item[&name='Abacavir']/location"

Broker 3, "/*/regions/asia/item[&name='Abacavir']/location"

Privacy Analysis

- Answers are returned through leaf-coordinator and broker.
- Unauthorized queries are rejected by the coordinators: probing queries are blocked.
- Leaf-coordinators know data server addresses, but nothing about the data
  - Leaf coordinators cannot see queries.
  - Leaf coordinators only hold accept states.
- Other coordinators have partial routing rule, but NO location information

✔ Data Location Privacy
Privacy Analysis

• Query content is encrypted in the communication.
• For all proxy components, ONLY the root-coordinators can see the content of the query.
✓ Query Content Privacy (Partially)

Solution

• NO coordinator knows user location, and ONLY the broker does
• But the broker does not know query content
✓ User Location Privacy

Evaluation

• The automaton is split into pieces, NO proxy knows an entire (access control/routing) rule
✓ Metadata Privacy

Performance Evaluation

• Settings
  – Coordinators
    • Java (JDK5.0)
    • Windows desktop
  – Data
    • XMark DTD and XML documents
    • Synthetic rules
    • Synthetic queries

• Network Assumption
  – It's unfair to use our Gigabyte intranet to measure network latency
  – Internet latency

End-to-End Query Processing Time

• Average query brokering time ($T_C$)
  – AC enforcement, routing to next coordinator and encryption
  – Average value from experiment: 1.9 ms

• Average network transmission time ($T_N$)
  – Estimated using Internet latency, average latency between two coordinators: 100ms

• Average number of hops ($N_{hop}$)
  – Estimated from experiment: 5.7

$$T_{forward} = T_C \times N_{hop} + T_N \times (N_{hop} + 1) = 681 \text{ (ms)}$$

• Without any privacy protection, $T_{forward} = 211 \text{ (ms)}$
• Average query evaluation time at data server ($T_E$)
• Average backward data transmission time ($T_{back}$)

System Scalability

• Total computation from all the coordinators
  – Measured by $N_{seg}$: total number of query segments in PPIB system.

• Parameters:
  – Query frequency
  – Size of rules
Conclusion

- Design the first privacy-preserving information brokering system (PPIB).
  - Integrate query routing with in-broker access control
- Design automaton segmentation scheme to preserve query content privacy.
  - Integrate automaton segmentation with query routing and access control
- Provide most comprehensive privacy protection for IBSs with insignificant performance degradation and scalability.

Questions?

- Thank you!

Privacy Analysis

- If the root coordinator is compromised:
  - PPIB vs. centralized proxy
    - We still protect query location and data location privacy
  - Full query segment encryption
    - Also encrypt un-processed XPath steps
      - Relaxed: encrypt all predicates
    - Coordinators need to be authenticated to decrypt an XPath step
    - Extra overhead: very complicated authentication process and key management scheme

A Real Example

- Assume 5 hospitals are sharing data; Each hospital has 10,000 patients
  - Assume each hospital has 10 roles;
  - How large is the total amount of index data?
    - Assume we only index on patient name: 10000*5=50K
    - could be greatly reduced at the router.
  - How large is the data server for each hospital?
    - TB Level: 100M*10K=1T
  - How many rules are there for each hospital?
    - 10-30 rules per role, 100-300 per hospital, 500-1500 in total
    - Rules may be identical or similar
  - How large is the global access control automata?
    - A fair guess (MRQ [SUTC 06]): 50 distinct paths

Privacy Analysis

- If one coordinator is compromised,
  - What he obtains:
    - A segment of the integrated automaton: an XPath step
    - Public resource: XML schema
  - What he can infer:
    - Pre-path from root to itself
      - Multiple paths available: k-anonymity
    - Post-path from itself to the accepted states
      - Multiple accept states available: l-diversity
    - Together t-concealment

A Real Example

- How many coordinators are needed in each category?
  - A fair guess: 50 leaf coordinators, 10 intermediate coordinators
  - Replicates needed
- What is the size for each automaton segment?
  - One XPath step per segment
  - Memory consumption of Java implementation: 10KB level
  - What is the average size of a query?
    - Consider the size of health care schema (e.g. HL7)
    - A fair guess: 4-8 paths
<table>
<thead>
<tr>
<th>Motivation</th>
<th>Solution</th>
<th>Evaluation</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Different with other anonymizing services</td>
<td>• Target destinations are known beforehand</td>
<td>• User don't know where to send the query</td>
<td>Query routing is necessary and unavoidable</td>
</tr>
<tr>
<td></td>
<td>Æ Query routing is necessary and unavoidable</td>
<td>Æ Proxy with the routing automaton knows too much</td>
<td></td>
</tr>
</tbody>
</table>