Current Research Topics in Data Security
 Dan Suciu

Seminar on Trustworthy Computing
 9/29/2004

Data Security

Dorothy Denning, 1982:

• Data Security is the science and study of methods of protecting data (...) from unauthorized disclosure and modification

Traditional Data Security

• Access control

• Security in statistical databases

Access Control

Discretionary Access Control

• The System R authorization model [Griffith and Wade'76], [Fagin'78]

• Became the SQL security model

• Extended, generalized to OO data

Mandatory Access Control

• Has been tried, but semantics becomes too complex

Access Control in SQL

GRANT privileges ON object TO users [WITH GRANT OPTIONS]

privileges = SELECT | INSERT(column-name) | DELETE | REFERENCES(column-name)  
object = table | attribute

Examples

GRANT INSERT, DELETE ON Reserves TO Yuppy WITH GRANT OPTIONS

GRANT SELECT ON Reserves TO Michael

GRANT SELECT ON Sailors TO Michael WITH GRANT OPTIONS

GRANT UPDATE (rating) ON Sailors TO Leah

GRANT REFERENCES (bid) ON Boats TO Bill
Views and Security

- David has SELECT rights on table Students
- Creates a VIEW BrightStudents
- Grants SELECT rights on BrightStudents to Dan

Revokation

```sql
REVOKE [GRANT OPTION FOR] privileges
ON object FROM users { RESTRICT | CASCADE }
```

Administrator says:

```sql
REVOKE SELECT ON Students FROM David CASCADE
```

Dan loses SELECT privileges on BrightStudents

Summary

- Access control:
  - great success story of the DB community...
  - ...or spectacular failure
    - SAP uses its own security layer
- Main assumption: data on trusted server
- The real challenge: securing the server
  - But this is not my job

Security in Statistical DBs

Goal:
- Allow aggregate queries
- Hide confidential data

Why it’s hard:
- Allow arbitrary aggregate queries, as long as no compromise

New Challenges in Data Security

- Traditional security: limited to client-server
- New Challenges: complex data management scenarios
  - Global sharing of data and services

Two Famous Attacks

- SQL injection
  - Chris Anley, Advanced SQL Injection In SQL Server Applications, www.ngssoftware.com
- Latanya Sweeney’s finding
**SQL Injection**

Go to your favorite shopping Website and login:

<table>
<thead>
<tr>
<th>Search order by date:</th>
<th></th>
</tr>
</thead>
</table>

Normal use:

<table>
<thead>
<tr>
<th>Search order by date:</th>
<th>9/15/04</th>
</tr>
</thead>
</table>

Now this:

<table>
<thead>
<tr>
<th>Search order by date:</th>
<th>9/15/04'; drop table user; --</th>
</tr>
</thead>
</table>

• The DBMS works perfectly. So why is SQL injection possible so often?

**Latanya Sweeney’s Finding**

• In Massachusetts, the Group Insurance Commission (GIC) is responsible for purchasing health insurance for state employees
• GIC has to publish the data:

\[
\text{GIC}(\text{zip, dob, sex}, \text{diagnosis, procedure, ...})
\]

• Sweeney paid $20 and bought the voter registration list for Cambridge Massachusetts:

\[
\text{GIC}(\text{zip, dob, sex}, \text{diagnosis, procedure, ...}) \\
\text{VOTER(name, party, ..., zip, dob, sex)}
\]

• William Weld (former governor) lives in Cambridge, hence is in VOTER
• 6 people in VOTER share his dob
• only 3 of them were man (same sex)
• Weld was the only one in that zip
• Sweeney learned Weld’s medical records!

• All systems worked as specified, yet an important data has leaked

• How do we protect against that?

Some of today’s research in data security address breaches that happen even if all systems work correctly
Research Topics in Data Security

1. Fine-grained access control
2. Database encryption
3. Privacy
4. Shared computation
5. Information Leakage
6. Watermarking
7. Integrity

Seems a random list of topics. How do we classify them?

Classification 1

• Confidentiality
  – Control who gets the data
  – Most research focuses here
  – Privacy falls here, but more complex

• Integrity
  – Control where the data comes from
  – Data provenance: a partial answer, not enforceable
  – I won’t cover it for lack of time

Classification 2

1. Fine-grained Access Control

Simple idea: control access at the tuple, even attribute level.

No big deal. What are the research questions?

• Policy specification languages
• Enforcement

Policy Specification Language

(Too) many exists. The good ones re-use a declarative query language, e.g. SQL, Xpath, XQuery

CREATE AUTHORIZATION VIEW PatientsForDoctors AS
SELECT Patient.*
FROM Patient, Treats, Doctor
WHERE Patient.pid = Treats.pid
  and Treats.did = Doctor.did
  and Doctor.uid = %userId
  and %accessMode in ('local', 'ssh')

[Oracle 7i], [Rizvi et al.2004]
Several policy languages for XML
Enforcement by query analysis/modification

```
SELECT Patient.name, Patient.age
FROM Patient
WHERE Patient.disease = 'flu'
```

```
SELECT Patient.name, Patient.age
FROM Patient
WHERE Patient.disease = 'flu'
```

```
SELECT Patient.name, Patient.age
FROM Patient, Treats, Doctor
WHERE Patient.disease = 'flu'
and Patient.pid = Treats.pid
and Treats.did = Doctor.did
and Doctor.userID = %currentUser
```

```
SELECT Patient.name, Patient.age
FROM Patient,
Treats, Doctor
WHERE Patient.disease = 'flu'
and Patient.pid = Treats.pid
and Treats.did = Doctor.did
and Doctor.userID = %currentUser
```

e.g. Oracle

Semantics

- The Truman Model: transform reality
  - ACCEPT all queries
  - REWRITE queries
  - Sometimes misleading results

- The non-Truman model: reject queries
  - ACCEPT or REJECT queries
  - Execute query UNCHANGED
  - Subtle semantics: instance dependent or independent

[Rizvi et al. SIGMOD 2004]

Semantics

- The Truman Model: transform reality
  - ACCEPT all queries
  - REWRITE queries
  - Sometimes misleading results

- The non-Truman model: reject queries
  - ACCEPT or REJECT queries
  - Execute query UNCHANGED
  - Subtle semantics: instance dependent or independent

Implementation with Accessibility Maps

- Enforce at query execution time
- High flexibility but high space cost
- Research issue: compress it
- E.g. for XML data, exploit locality:

2. Encryption in DBMS

Some scenarios:

- Untrusted storage, trusted server
  - Required by legislation
  - Malicious DBAs

- Untrusted storage, untrusted server
  - Database as a Service

2. Encryption in DBMS

Issues:

- Granularity of the encryption:
  - E.g. attribute, tuple, disk page/block

- What is encrypted:
  - E.g. user data (always), metadata, indexes, logs

3. Privacy

- “Is the right of individuals to determine for themselves when, how and to what extent information about them is communicated to others” [Agrawal, VLDB’03]

- More complex than confidentiality

Approaches to Privacy in Data Management

- Hippocratic Databases [Agrawal et al. VLDB’04]
  - Make DBMS privacy-aware
  - Protects against negligence, ignorance
    - arguably the most common
  - No protection against malicious attacks

- Privacy for the paranoids [Aggarwal et al. VLDB’04]
  - DIFFERENT Aggrawal !!

Hippocratic Databases

Ten principles:
- Purpose specification
- Consent
- Limited collection
- Limited use
- Limited disclosure
- Limited retention
- ...

Privacy for Paranoids

[G. Aggarwal et al., VLDB’2004]
- not Agrawal
- Idea: rely on trust agents to control private data

Example 1
- Replace email alice@aliceHost.com with aly1@agentHost.com

Example 2
- Replace a credit card number with a one-time use number: pseudonum
4. Shared Processing

- Alice has a database DB_A
- Bob has a database DB_B
- How can they compute Q(DB_A, DB_B), without revealing their data?

---

4. Shared Processing

Solution 1: one-way hash function h(-)
- Alice and Bob compute h(DB_A) and h(DB_B)
- Exchange
- Intersect
- What’s wrong?

[Agrawal et al. SIGMOD'03]

Solution 2: commutative encryption E_{key}(-)
- Alice, Bob compute E_A(DB_A) and E_B(DB_B)
- Exchange
- Alice, Bob compute E_A(E_B(DB_B)) and E_B(E_A(DB_A))
- Exchange
- Compute the intersection
  - Possible because E_A(E_B(x)) = E_B(E_A(x))

[Agrawal et al. SIGMOD'03]

---

5. Watermarking

- Want to sell a database instance
- But want to be able to trace the source
- Watermark:
  - small, hidden perturbations in the database that prove its origin
- How can one do that?
  - Possible for numeric values that tolerate some loss in precision
- Variation: fingerprinting

[Agrawal, Kiernan VLDB’02]

6. Information Leakage

Single source:
- Alice publishes two views:
  - V1(PatientName, BuildingNumber) - for guests
  - V2(BuildingNumber, Disease) - for CDC control
- Malory wants to know if ‘Joe Doe’ has ‘measles’
- Is there a leakage?

Approach: using information theory
[Agrawal, S 2004], [Miklau, Dalvi, S 2005] [Yang and Li 2004]
6. Information Leakage

Multiple sources
- Latanya Sweeney’s example

Approach: k-anonymity
- Replace values with NULL until every tuple appears at least k times in the table
- NP-hard to anonymize optimally [Meyerson, Williams, PODS’2004]

7. Integrity

- Next week; Come to Gerome’s talk.

Summary

- Traditional data security
  - Access control in SQL
  - Statistical databases

- Current research in data security
  - Very varied
  - Reflects the varied data management tasks we face
  - Database researchers are consumers of both cryptography and systems security