Towards an inference detection system against multi-database attacks

27 August 2020

Authors:
Paul Lachat¹,²
paul.lachat@insa-lyon.fr
Veronika Sonigo³
Nadia Bennani¹
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

Functional Dependency

\[ \text{Rank} \Rightarrow \text{Salary} \]

Security rule

\[ \times \text{Name and Salary} \]

Q₁: \text{SELECT Rank, Salary FROM Employee}

Q₂: \text{SELECT Name, Rank FROM Employee}

John, Clerk \Rightarrow 38,000 \times
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

Functional Dependency

\[ \text{Rank} \Rightarrow \text{Salary} \]

Security rule

\[ \times \text{Name and Salary} \]

Q₁

\[
\text{SELECT Rank, Salary}
\]
\[
\text{FROM Employee}
\]

Q₂

\[
\text{SELECT Name, Rank}
\]
\[
\text{FROM Employee}
\]

John, Clerk \Rightarrow 38,000 \times
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

Functional Dependency

\[ \text{Rank} \Rightarrow \text{Salary} \]

Security rule

\[ \times \text{Name and Salary} \]

Q₁

\[
\text{SELECT Rank, Salary}
\text{FROM Employee}
\]

Q₂

\[
\text{SELECT Name, Rank}
\text{FROM Employee}
\]

John, Clerk $\Rightarrow$ 38,000 $\times$
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

Functional Dependency

Rank $\Rightarrow$ Salary

Security rule

$\times$ Name and Salary

$Q_1$ SELECT Rank, Salary FROM Employee

$Q_2$ SELECT Name, Rank FROM Employee

John, Clerk $\Rightarrow$ 38,000 $\times$
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

Functional Dependency

\[ \text{Rank} \Rightarrow \text{Salary} \]

Security rule

\[ \times \text{Name and Salary} \]

Q1: SELECT Rank, Salary
FROM Employee

Q2: SELECT Name, Rank
FROM Employee

John, Clerk $\Rightarrow$ 38,000 $\times$
Inference detection in databases – Motivation

<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Mary</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Chris</td>
<td>Secretary</td>
<td>28,000</td>
</tr>
<tr>
<td>Joe</td>
<td>Manager</td>
<td>45,000</td>
</tr>
<tr>
<td>Sam</td>
<td>Clerk</td>
<td>38,000</td>
</tr>
<tr>
<td>Eve</td>
<td>Manager</td>
<td>45,000</td>
</tr>
</tbody>
</table>

**Functional Dependency**

**Rank ⇒ Salary**

**Security rule**

- **Name** and **Salary**

**Q₁**

```sql
SELECT Rank, Salary
FROM Employee
```

**Q₂**

```sql
SELECT Name, Rank
FROM Employee
```

John, Clerk ⇒ 38,000 ✗
Inference detection system

Example:
- **Logical framework**: Horn Clauses
  - Tyrone S Toland et al. (2010).

Example:
- **Probabilistic framework**: Bayesian Network
  - Yu Chen et al. (2006).
Inference detection system

Example:

![Diagram of data flow and inference detection system](image)
Inference detection system

Protect a single database against inference attacks.

User → Data Controller → Data Processor

Give her data

Query

Answer

Inference?

Answer

Logical framework
Example: Horn Clauses

Probabilistic framework
Example: Bayesian Network
Limits of existing solutions

- **IDS\(_1\)**
  - Security rule: Give her **data**
  - Functional Dependency: Rank, Age $\Rightarrow$ Salary

- **DC\(_1\)**
  - User
  - Give her data

- **IDS\(_2\)**
  - Security rule: $\times$ Name and Salary

- **DC\(_2\)**
  - DP
  - Give her data

- **DC\(_1\)** and **DC\(_2\)** contain:
  - age, name, salary
  - rank, name, height
Limits of existing solutions

Functional Dependency

Rank, Age \(\Rightarrow\) Salary

Security rule

\[\times\] Name and Salary

IDS\(_1\)

DC\(_1\)

Give her data

age, name

User

Give her data

rank, name, height

IDS\(_2\)

DC\(_2\)

Give her data

age, name

DP
Limits of existing solutions

User

Give her data

DC1

IDS1

age, name

Give her data

DC2

IDS2

DP

Security rule

× Name and Salary

Functional Dependency

Rank, Age ⇒ Salary

age, name

rank, name, height

name, salary
Limits of existing solutions

User

IDS$_1$

DC$_1$

IDS$_2$

DP

Functional Dependency

Rank, Age $\Rightarrow$ Salary

Security rule

$\times$ Name and Salary

Give her data

age, name, salary

Give her data

rank, name, height

Ok!
Limits of existing solutions

- **User**
  - Give her data
  - age, name, salary

- **IDS$_1$**
  - Functional Dependency
    - Rank, Age $\Rightarrow$ Salary
  - Give her data
  - age, name

- **DC$_1$**

- **User**
  - Give her data
  - rank, name, height

- **IDS$_2$**

- **DC$_2$**

- **DP**
  - Security rule
    - $\times$ Name and Salary
Limits of existing solutions

Functional Dependency

$\text{Rank, Age} \Rightarrow \text{Salary}$

Security rule

$\times \text{Name and Salary}$

IDS$_1$

$\cdots$

DC$_1$

Give her data

age, name

User

Give her data

rank, name

DC$_2$

Give her data

rank, name

IDS$_2$

age, name

DP

name, salary

name, salary

name, salary

height

name, salary

name, salary
Limits of existing solutions

User

IDS$_1$

DC$_1$

Give her data

DC$_2$

Give her data

IDS$_2$

DP

Functional Dependency

Rank, Age $\Rightarrow$ Salary

Security rule

$\times$ Name and Salary

age, name, salary

rank, name, height

rank, name

name

salary

name
Limits of existing solutions

IDS$_1$

DC$_1$

User

Give her data

age, name, salary

DC$_2$

Give her data

rank, name

IDS$_2$

DP

OK!

Functional Dependency

Rank, Age $\Rightarrow$ Salary

Security rule

$\times$ Name and Salary

age, name

rank, name
Limits of existing solutions

Functional Dependency

\[
\text{Rank, Age} \Rightarrow \text{Salary}
\]

Security rule

\[
\times \text{Name and Salary}
\]

IDS\(_1\)

User

Give her data

DC\(_1\)

Give her data

DC\(_2\)

Give her data

IDS\(_2\)

age, name, salary

age, name

rank, name

rank, name

Name and Salary
Our proposition

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Our proposition

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Our proposition

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Our proposition

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Our proposition

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Chen et al. 2006 – Solution

User \hspace{2cm} Data Controller \hspace{2cm} Data Processor

Give her data \hspace{2cm} Query \hspace{2cm} Answer

Answer \hspace{2cm} Inference?

Data Controller \hspace{2cm} Query \hspace{2cm} Answer

User gives data to the Data Controller, which then queries the Data Processor. The Data Controller checks the dependencies at schema level. The Data Processor answers the query based on the data provided. The answer is then linked to an inference model (IDS) for further analysis.

**Semantic Inference Model (SIM)**
- Dependencies at schema level.

**Semantic Instance Graph (SIG)**
- Dependencies at instances level.

**SIM**
- Computes dependencies at schema level.

**SIG**
- Instantiates the dependencies into a graph.
Chen et al. 2006 – Solution

**Semantic Inference Model (SIM)**

Dependencies at schema level.

**Semantic Instance Graph (SIG)**

Dependencies at instances level.
Chen et al. 2006 – Solution

Inference Detection System

compute

SIM

Table1
Attr1
Attr2

Table2
Attr1
...

Table3
Attr1
Attr2

Dependencies at schema level.

Semantic Inference Model (SIM)

instantiate into

SIG

Pierre Mar...

Semantic Instance Graph (SIG)

Dependencies at instances level.
Inference channels within a database: Example of a SIG

Inference threshold set to 70%.

Dependencies between instances of the Los Angeles airport database. **LAX** is an airport, **R1** a runway, and **C5** an aircraft.
Inference channels within a database: Example of a SIG

Dependencies between instances of the Los Angeles airport database. **LAX** is an airport, **R1** a runway, and **C5** an aircraft.

Inference threshold set to 70%.
Contribution – *Global Instance Graph* (GIG) computation

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph* (GIG) computation

**Challenges**
- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**
- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph (GIG)* computation

**Challenges**

- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph (GIG)* computation

**Challenges**

- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph (GIG)* computation

**Challenges**
- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**
- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph* (GIG) computation

**Challenges**
- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**
- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph* (GIG) computation

### Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

### Hypothesis

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Contribution – *Global Instance Graph* (GIG) computation

**Challenges**

- Identify similar instances.
- The system can have an honest-but-curious behavior.

**Hypothesis**

- Data controllers collaborate with our system.
- The system is centralised to extend the solution of Chen et al. 2006.
Conclusion & Future work

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Next steps

- Optimise the GIG computation.
- Take databases updates into account.
Conclusion & Future work

Challenges

- Identify similar instances.
- The system can have an honest-but-curious behavior.

Next steps

- Optimise the GIG computation.
- Take databases updates into account.