Assumes these are already secure

Distinct from systems and network security

Data Security = Confidentiality + Integrity

Data Security is the science and study of methods of protecting data from unauthorized disclosure and modification

Data Security
GRANT INSERT, DELETE ON Customers TO Yahia WITH GRANT OPTIONS

Queries allowed to Yahia:

```
INSERT INTO Customers(cid, name, address)
```

Queries denied to Yahia:

```
DELETE Customers WHERE LastPurchaseDate > 1993
```

**Examples**

```
GRANT INSERT, DELETE, REFERENCES ON Customers TO Yahia
```

```
INSERT INTO Customers(cid, name, address)
```

```
DELETE Customers WHERE LastPurchaseDate > 1993
```

```
SELECT Customer.address FROM Customer WHERE name = 'Joe Blow'
```

```
SELECT Customer.address FROM Customer WHERE name = 'Joe Blow'
```

**Discretionary Access Control in SQL**
GRANT SELECT ON Customers TO Mosaed

Now Mosaed can SELECT on Customers

Now Yahia can SELECT on Customers

Yahia can say this:

Example:

Example
Now Basel can INSERT tuples into Orders.

**Examples**

BASEL has INSERT/UPDATE rights to Orders.

**Examples**

Leila can update, but only Product.price, but not Product.name.
CREATE VIEW PublicCustomers
SELECT Name, Address
FROM Customers
GRANT SELECT ON PublicCustomers TO
Farid

CREATE VIEW PublicCustomers
SELECT Name, Address
FROM Customers
GRANT SELECT ON PublicCustomers TO
Farid, Dawood

Dawood says

CREATE VIEW BadCreditCustomers
SELECT *
FROM Customers
WHERE Balance > 0
GRANT SELECT ON BadCreditCustomers TO
Johaina

CREATE VIEW BadCreditCustomers
SELECT *
FROM Customers
WHERE Balance > 0
GRANT SELECT ON BadCreditCustomers TO
Johaina

Davood says

Dawood owns Customers:
Johaina is allowed to see balances > 0.

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ann</td>
<td>Portland</td>
<td></td>
</tr>
<tr>
<td>Joan</td>
<td>Seattle</td>
<td></td>
</tr>
<tr>
<td>Sue</td>
<td>Seattle</td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td>Houston</td>
<td>450.99</td>
</tr>
</tbody>
</table>

Views and Security: Customers
CREATE VIEW CustomerMary
SELECT * FROM Customers
WHERE name = 'Mary'
GRANT SELECT ON CustomerMary TO Mary

CREATE VIEW CustomerMary
SELECT * FROM Customers
WHERE name = 'Mary'
GRANT SELECT ON CustomerMary TO Mary

Each customer should see only her/his record.

Views and Security
According to SQL, everyone keeps the privilege.

What happens?

GRANT OPTION

Same object, same privilege.

Reversion

Reversion
Access control = great success story of the DB community...

- And then to protect entire tables, not columns
- Only assign privileges to users/roles

... or spectacular failure:

Table creator owns the data: that's unfair!

- No row level access control

Limitations:

Summary of SQL Security


WHERE
FROM PERSON
(SELECT AVG(SALARY)
information
with where clause which enables the user to get this
The SQL can be formed as statistical command accompanied
not allowed to get some data (e.g. salary of specific employee),
Suppose it is allowed to issue statistical commands on K and it is
EX

Statistics DB Security (cont)

Statistics DB Security

same time provide certain level of security.

Statistics DB commands on DB and on the
The problem is how to allow users to issue

that satisfy some selection conditions.

A population is a set of tuples of a relation

(count, max, min, etc).

(commands as sum,

statistics on various populations
statistics DB are used mainly to produce

Statistics DB Security

Statistics DB Security
Inserting slight noise into the results.

• Tuples refer repeatedly to the same population of

Prohibiting the sequence of queries that condition fails below certain threshold.

Population specified by the selection whenever the number of tuples in the

No statistical queries are permitted.

Some solutions:

Statistical DB Security (cont)