INFO2222:
INtro to Security & Usability

Database security

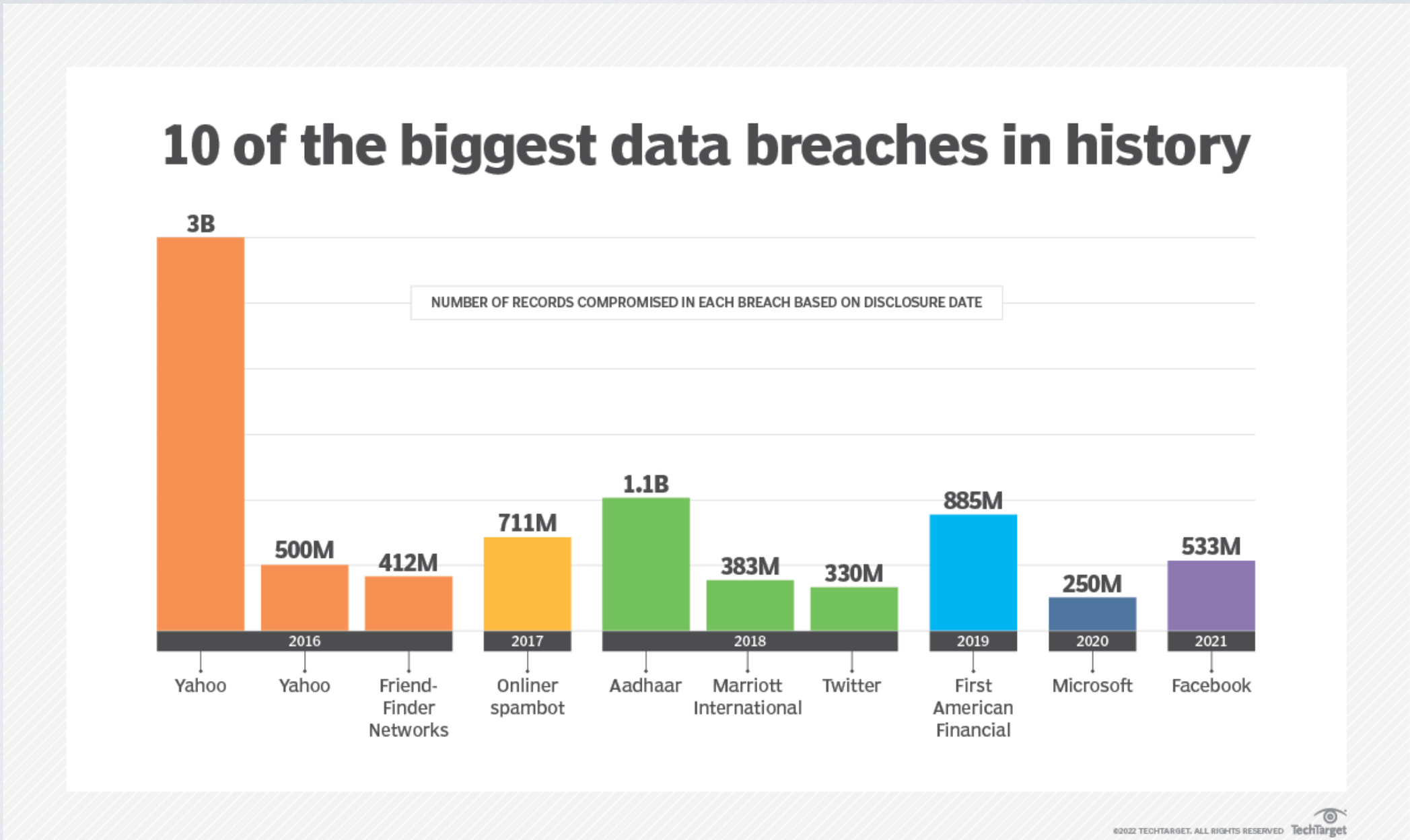
Database Security

Confidentiality, integrity and availability of an organization's databases.

Why It Matters?

Databases often hold the backbone of an organization; Its transactions, customers, employee info, financial data are held in databases, often left to the power of a databased administrator with no security training.

Why It Matters?



https://www.techtarget.com/searchsecurity/feature/10-biggest-data-breaches-in-history-and-how-to-prevent-them

Database Security: Example

Confidentiality, integrity and availability of an organization's databases.

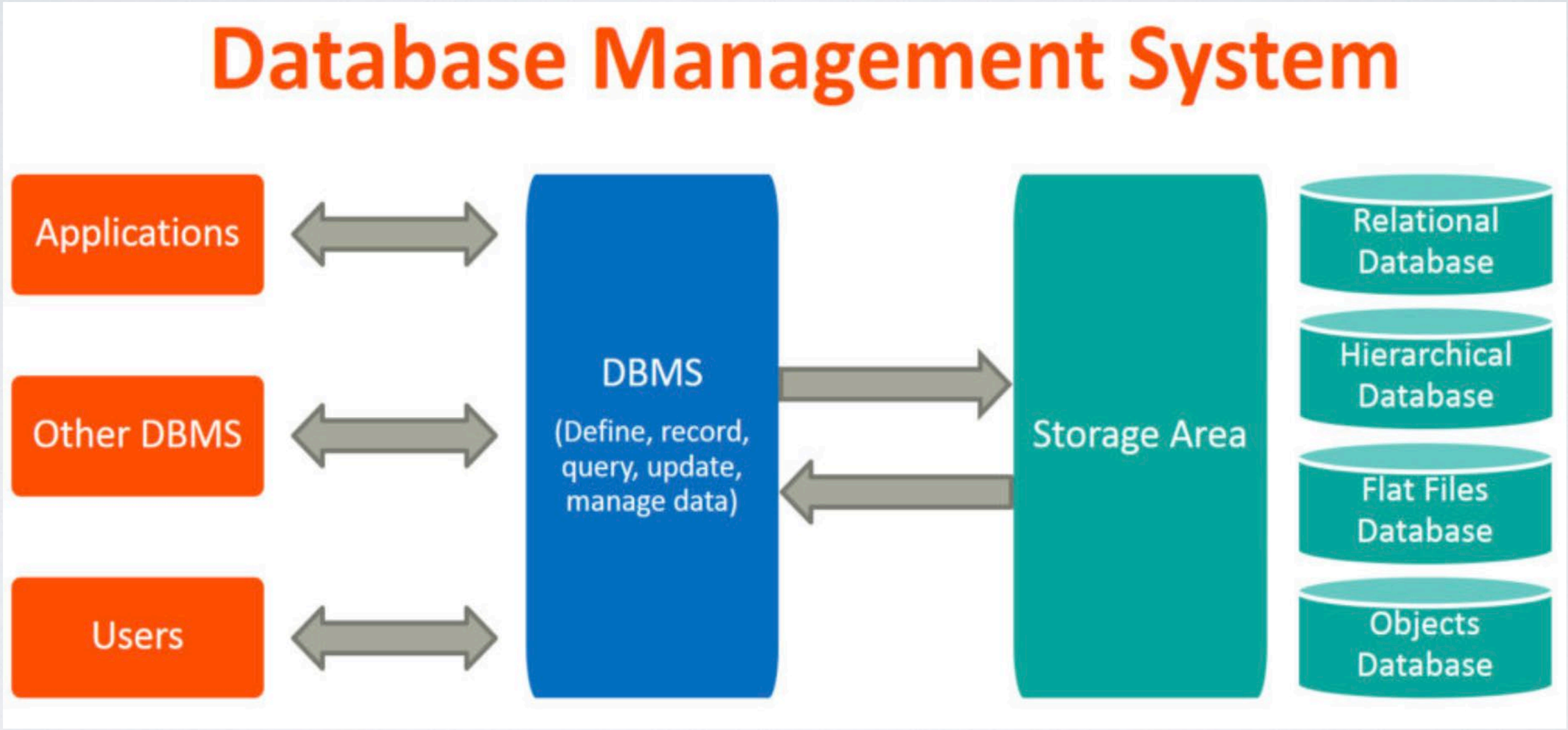
A payroll database in a company:

Individual salary is not disclosed

salaries are not modified

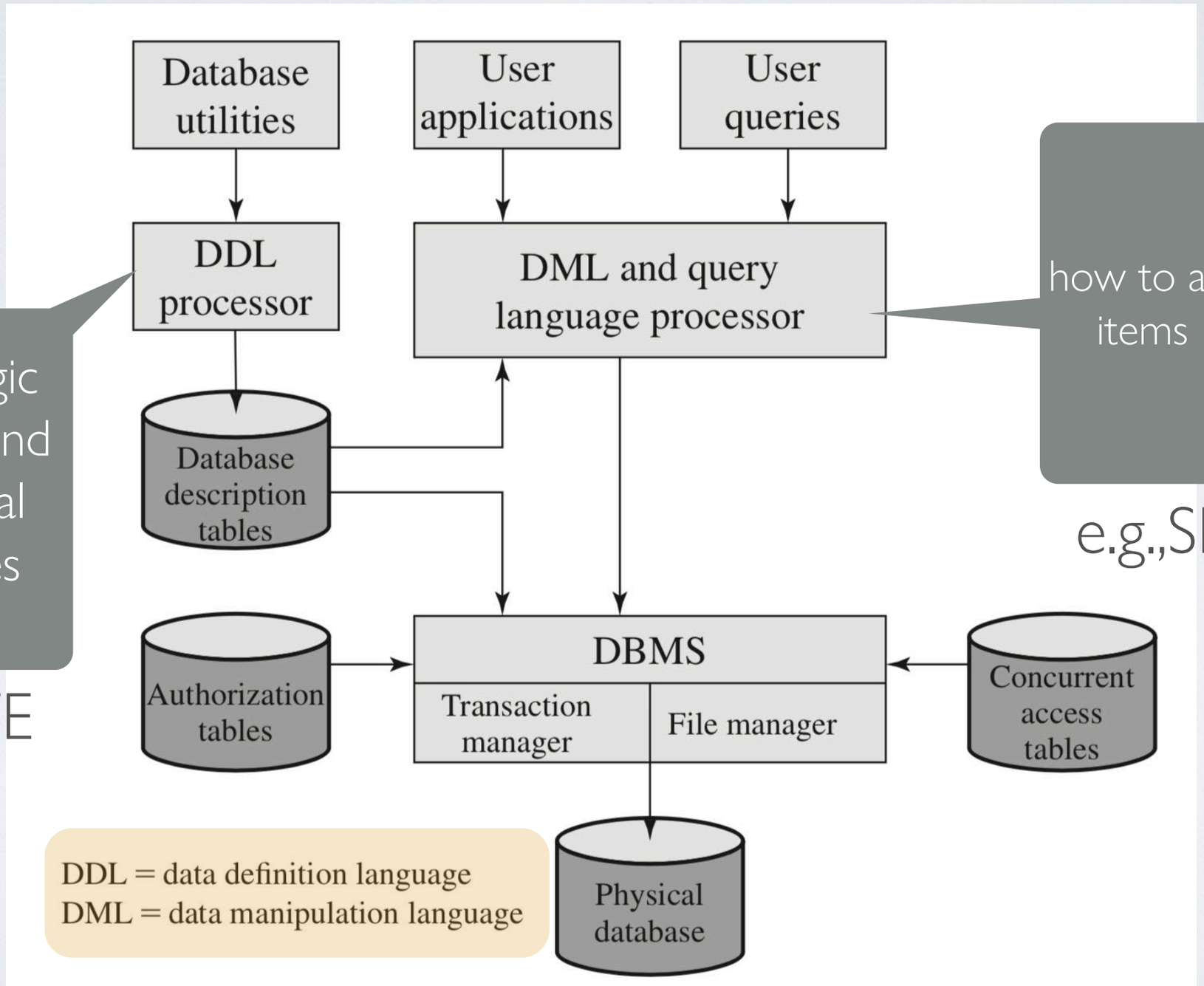
paychecks are printed in time

Database Management System: An Overview



Provides users and programers a systematic way to create, retrieve, update and manage data.

Database Management System: An Overview



defines logic structure and procedural properties

e.g.,CREATE

how to access data items

e.g.,SELECT

Relational Databases

First name	Last name	Employee ID	Department ID	etc.
Annie	Oakley	340955	6	etc.
Noam	Chomsky	409102	9	etc.
Marvin	Hagler	268003	6	etc.
William	Cody	550254	9	etc.
Walt	Disney	027851	6	etc.

Department ID	Department name	Department VP emp ID	Department cost centre	etc.
6	Collections	711203	7684	etc.
9	Graphic Arts	488030	2417	etc.

Tables of data, similar to a spreadsheets

Relational Databases

First name	Last name	Employee ID	Department ID	etc.
Annie	Oakley	340955	6	etc.
Noam	Chomsky	409102	9	etc.
Marvin	Hagler	268003	6	etc.
William	Cody	550254	9	etc.
Walt	Disney	027851	6	etc.

Department ID	Department name	Department VP emp ID	etc.
6	Collections	711203	etc.
9	Graphic Arts	488030	etc.

Multiple tables can be tied together by unique identifiers

Structured Query Language (SQL)

Create, retrieve, update …. database

Structured Query Language (SQL)

CREATE TABLE EMPLOYEE:
First Name CHAR (30)
Last Name CHAR (30)
Employee ID CHAR (10)
Department ID INTEGER
Etc CHAR (30)

First name	Last name	Employee ID	Department ID	etc.
Annie	Oakley	340955	6	etc.
Noam	Chomsky	409102	9	etc.
Marvin	Hagler	268003	6	etc.
William	Cody	550254	9	etc.
Walt	Disney	027851	6	etc.

SQL Retrieve

SELECT First Name
from EMPLOYEE:
Where Employee ID = 409102

First name	Last name	Employee ID	Department ID	etc.
Noam		409102		

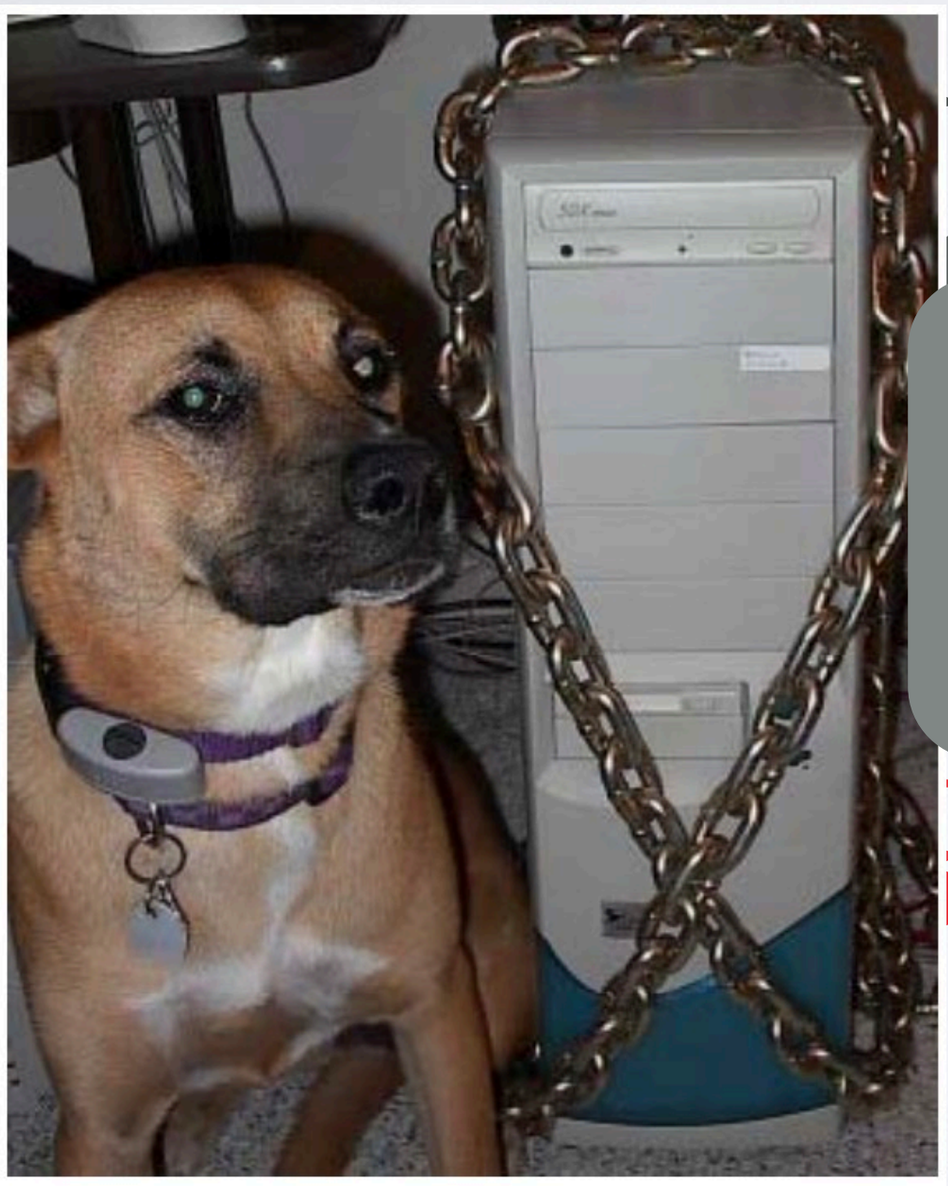
Table & View

View is a virtual table:
the result of a query that returns selected
rows and columns from one or more tables

```
CREATE VIEW [Brazil Customers] AS
SELECT CustomerName, ContactName
FROM Customers
WHERE Country = 'Brazil';
```

Database Access Control

A d ol system distinguishes
dif ncluding create, insert
A
t of to selected rows of
ithin a table.



Assuming proper user authentication mechanism is on place

Discretionary Access Control

Access matrix is one common method

		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix

SQL Access Definition

- Objectes to be protected are tables and views
- Privileges include: select, update, insert, delete, drop etc

View/Content Based Authorization

Suppose we want to authorize user Ann to access only employees whose salary is lower than 20k

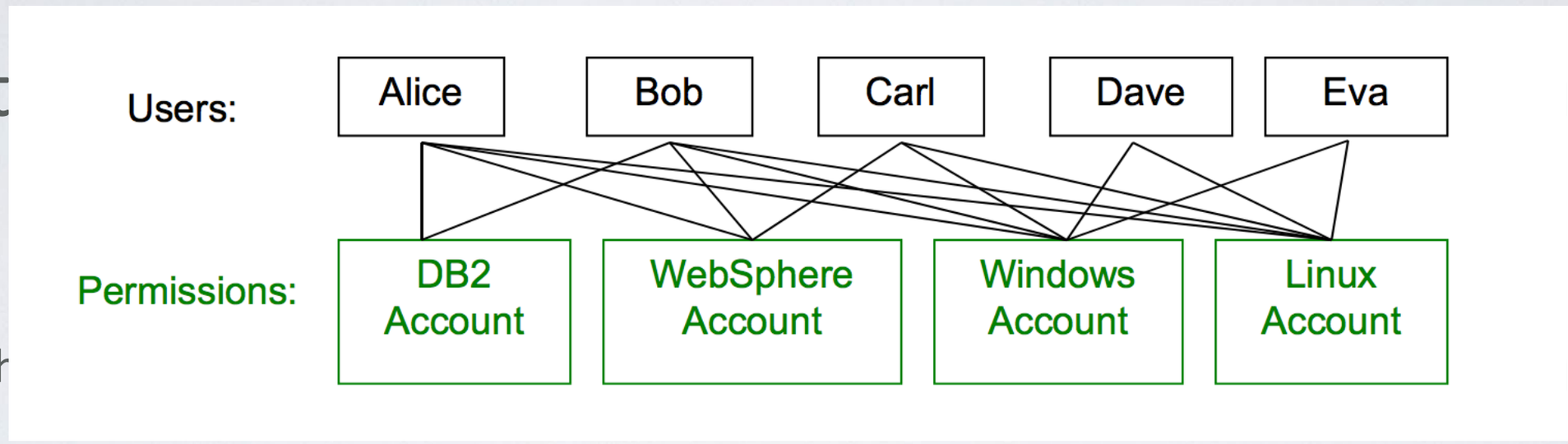
```
CREATE VIEW Vemp AS
SELECT * FROM Employee
WHERE Salary < 20000;

GRANT Select ON Vemp TO Ann
```

More Generally

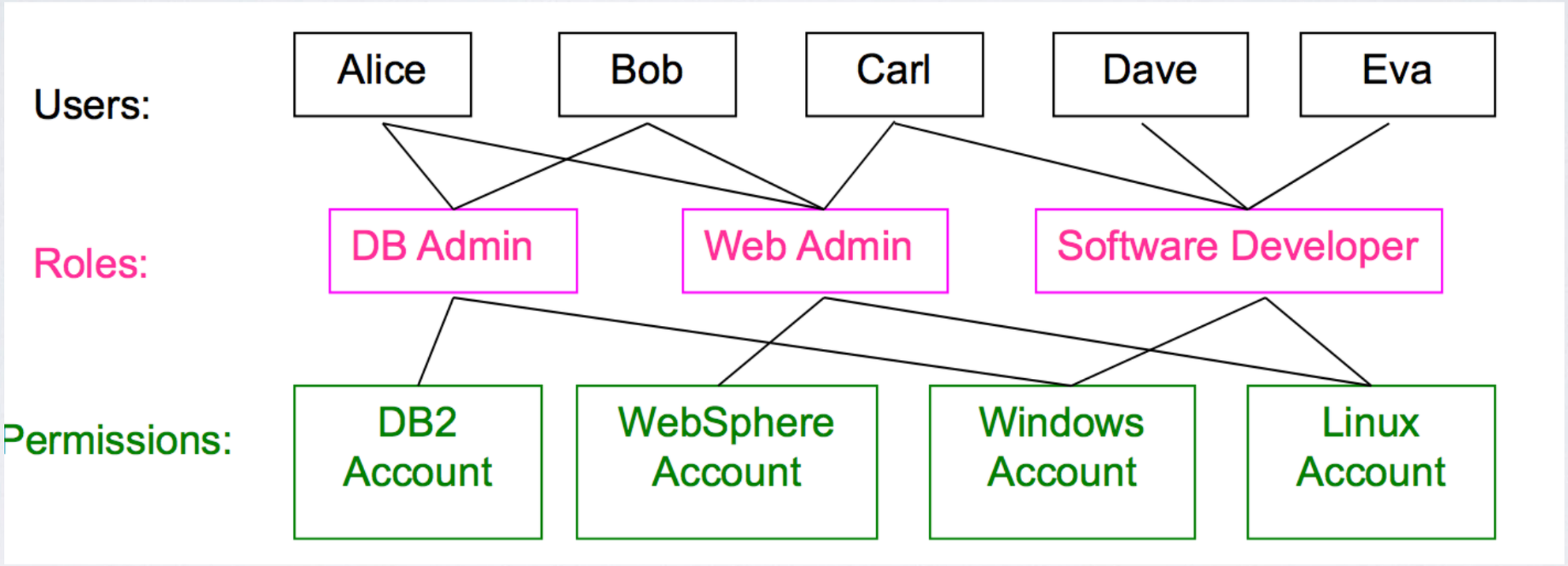
A dat tions

Each privileges



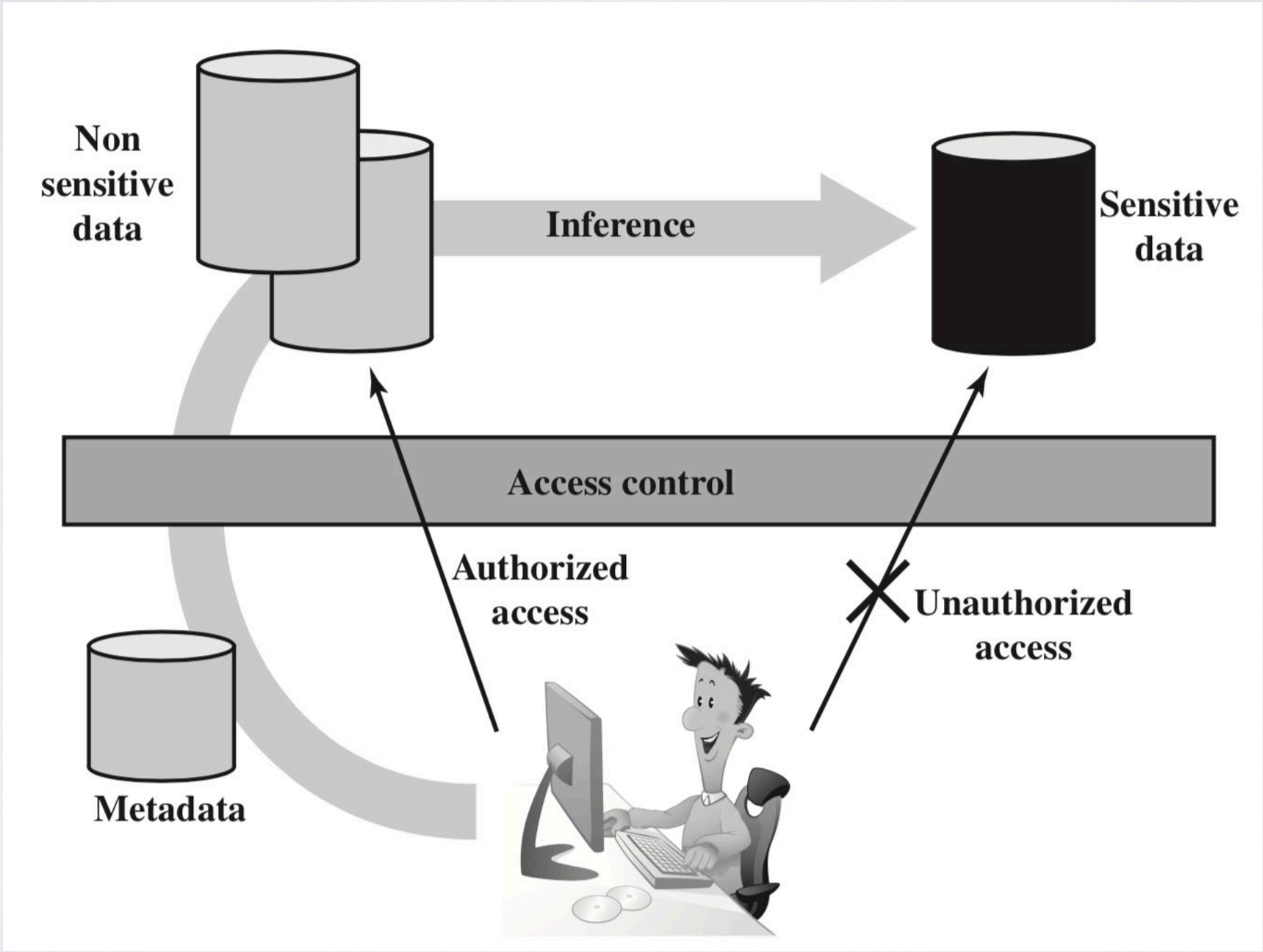
Role Based Access Control on Database

Organizations work on roles



Assign users to roles requires less tech skills than assigning permissions to roles

Inference Attacks



Inference Attacks

Two methods

make dependent queries

Merge views

Vulnerability Example

Name	Sex	Programme	Units	Grade Ave
Alma	F	MBA	8	63
Bill	M	CS	15	58
Carol	F	CS	16	70
Don	M	MIS	22	75
Errol	M	CS	8	66
Flora	F	MIS	16	81
Gala	F	MBA	23	68
Homer	M	CS	7	50
Igor	M	MIS	21	70

Statistics allowed

Grade not allowed

Making Dependent Queries

Name	Sex	Programme	Units	Grade Ave
Alma	F	MBA	8	63
Bill	M	CS	15	58
Carol	F	CS	16	70
Don	M	MIS	22	75
Errol	M	CS	8	66
Flora	F	MIS	16	81
Gala	F	MBA	23	68
Homer	M	CS	7	50
Igor	M	MIS	21	70

Q2: SELECT Avg (Grade Ave) FROM Students WHERE Sex ='F' AND Programme = 'CS'

Q1: SELECT Count (*) FROM Students WHERE Sex ='F' AND Programme = 'CS'

Merge Views

Item	Availability	Cost ($)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware
Cake pan	online only	12.99	housewares
Shower/tub cleaner	in-store/online	11.99	housewares
Rolling pin	in-store/online	10.99	housewares

(a) Inventory table

Users are not allowed to access relation between item - cost

Allow others

Merge Views

Item	Availability	Cost ($)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware
Cake pan	online only	12.99	housewares
Shower/tub cleaner	in-store/online	11.99	housewares
Rolling pin	in-store/online	10.99	housewares

(a) Inventory table

```
CREATE view V1 AS
SELECT Availability, Cost
FROM Inventory
WHERE Department = "hardware"
```

```
CREATE view V2 AS
SELECT Item, Department
FROM Inventory
WHERE Department = "hardware"
```

Merge Views

Availability	Cost ($)
in-store/online	7.99
online only	5.49
in-store/online	104.99

Item	Department
Shelf support	hardware
Lid support	hardware
Decorative chain	hardware

```
CREATE view V1 AS
SELECT Availability, Cost
FROM Inventory
WHERE Department = "hardware"
```

```
CREATE view V2 AS
SELECT Item, Department
FROM Inventory
WHERE Department = "hardware"
```

Defending Inference Attacks

detection during DB AC design:

Example1 — further restrict Allowed queries

modify DB structure: removing data dependences, splitting databases etc

Example2 — get a separate table for item-price and restrict access

detection at query time

SQL Injection Attacks

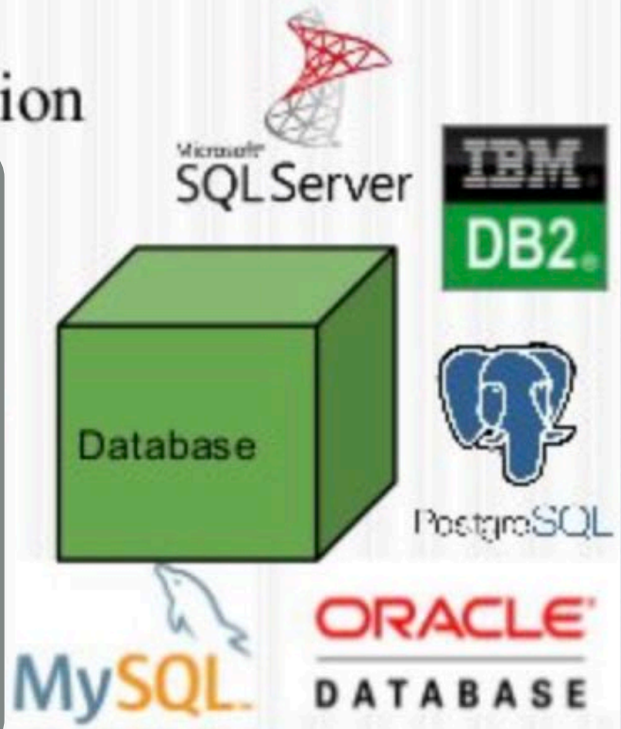
"A SQL injection attack consists of insertion or "injection" of a SQL query via the input data from the client to the application."

- https://www.owasp.org/index.php/SQL_Injection

Application Users via client programs

User inject
crafted SQL
manipulate

Most current Web sites have dynamic content asking for info & transferred to and from backend DB



Typical SQL Injection Attacks

- Hacker finds vulnerable slitty in a Web application and injects an SQL command, which will be accepted by the firewall
- The web server receives the malicious code and sends to Web application server
- The web application server receives the malicious code and sends to the database server
- The database server executes the malicious code, and returns data

Vulnerability Examples

```
var Shipcity;
ShipCity = Request.form (“ShipCity”);
var sql = “select * from OrdersTable where ShipCity = ‘” +
ShipCity + “’”;
```

Waiting for user to enter a city name to execute an SQL command

However

```
var Shipcity;
ShipCity = Request.form
var sql = "select * from                    = '" +
ShipCity + "'";
```

Ignores potential subsequent text using the comment mark

Hacker inputs: Redmond'; DROP table OrdersTable - -

This results in the following SQL query:

```
SELECT * FROM OrdersTable WHERE ShipCity =
'Redmond'; DROP table OrdersTable--
```

Countermeasures

Defensive coding in the scripts/backend

Input validation,
e.g., type checking

Use prepared SQL

Access control: least privilege

Database Security
Case Study

- ABC Healthcare is a large healthcare organization that stores sensitive patient information, including medical records, personal details, and billing information, in its database.

- Recently, the organization faced a security breach where unauthorized access was gained to their database, resulting in the compromise of patient data.

- This breach has raised concerns about the effectiveness of their database security measures and the potential impact on patient privacy and regulatory compliance.

Possible Issues

1. **Unauthorized Access:** The breach highlighted vulnerabilities in the database access controls, allowing unauthorized users to gain access to sensitive data.

2. **Data Exposure:** Patient data, including medical records and personal information, was exposed, raising concerns about privacy breaches and potential identity theft.

3. **Compliance Violations:** The breach may have violated regulatory requirements, such as HIPAA (Health Insurance Portability and Accountability Act) in the United States, leading to potential legal consequences and financial penalties.

4. **Data Integrity:** There are concerns about the integrity of the data stored in the database, as unauthorized access could have led to tampering or manipulation of patient records.

5. **Lack of Encryption:** Data stored in the database may not have been adequately encrypted, increasing the risk of data exposure in the event of a breach or unauthorized access.

Solutions

Implement Role-Based Access Control (RBAC): ABC Healthcare should implement RBAC to restrict access to the database based on user roles and responsibilities. This ensures that only authorized users have access to specific data and functionalities within the database.

Enhance Authentication Mechanisms: Strengthen authentication mechanisms by implementing multi-factor authentication (MFA) to verify the identity of users attempting to access the database. This adds an extra layer of security and reduces the risk of unauthorized access.

Encrypt Sensitive Data: Utilize encryption techniques, such as Transparent Data Encryption (TDE), to encrypt sensitive data stored in the database. Encryption protects data at rest and in transit, making it unreadable to unauthorized users even if the database is compromised.

Implement Database Activity Monitoring (DAM): Deploy DAM solutions to monitor database activity in real-time and detect suspicious or unauthorized access attempts. DAM tools can alert administrators to potential security incidents and help mitigate risks proactively.

Regular Security Audits and Penetration Testing: Conduct regular security audits and penetration testing to identify vulnerabilities in the database environment. By proactively identifying and addressing security weaknesses, ABC Healthcare can strengthen their database security posture and reduce the risk of data breaches.

Employee Training and Awareness: Provide comprehensive training to employees on database security best practices, including data handling procedures, password management, and security protocols. Increasing employee awareness of security risks and responsibilities can help prevent security breaches caused by human error or negligence.

Data Loss Prevention (DLP) Measures: Implement DLP solutions to prevent unauthorized data disclosure and leakage. DLP technologies can monitor and control the movement of sensitive data within the database and across network boundaries, ensuring compliance with regulatory requirements and protecting patient privacy.