MONASH University
Information Technology

FIT2093 INTRODUCTION TO CYBER SECURITY

MONASH University
Information Technology

FIT2093 INTRODUCTION TO CYBER SECURITY

Database Security

Unit Structure

- Introduction to security of
- Authentication
- Access Control
- Introduction to number theory
- Cryptography I
- Cryptography II
- Public key cryptography
- Integrity management
- Practical aspects of cyber security
- Hacking and countermeasures
- Database security
- IT risk management & Ethics and privacy

Outline

- **Security Issues in a relational database system**
- **SQL Injection Attacks**
- **Database Access Control**
- **Sensitive Data protection**
- **Inference threats**
- **Security Issues related to statistical databases**
- **Security issues related to cloud computing**

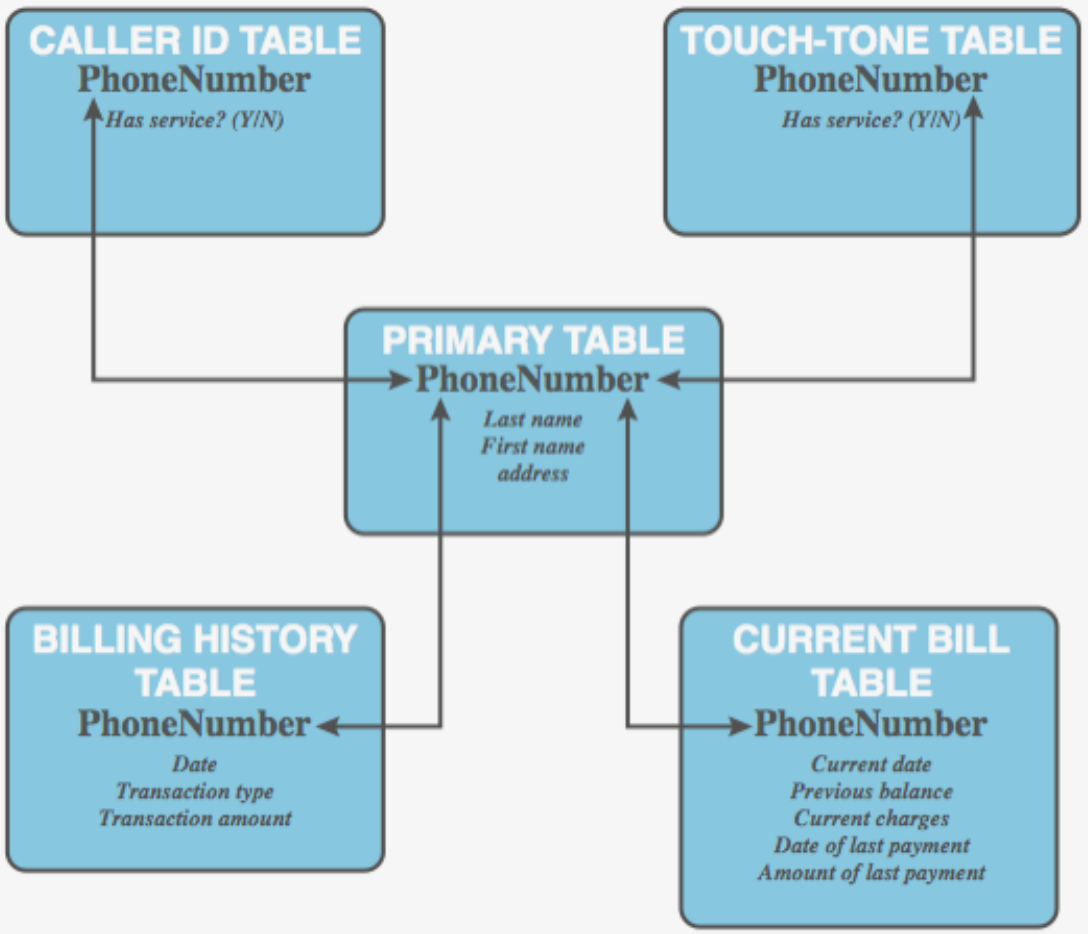
Databases

- **structured collection of data stored for use by one or more applications**
- **contains the relationships between data items and groups of data items**
- **can sometimes contain sensitive data that needs to be secured**
- **database management system (DBMS)**
  - suite of programs for constructing and maintaining the database
  - offers ad hoc query facilities to multiple users and applications

Relational Databases

- **table of data consisting of rows and columns**
  - each column holds a particular type of data
  - each row contains a specific value for each column
  - ideally has one column where all values are unique, forming an identifier/key for that row
- **enables the creation of multiple tables linked together by a unique identifier that is present in all tables**
- **use a relational query language to access the database**
  - allows the user to request data that fit a given set of criteria

Example Relational Database Model



Relational Database Elements

Department Table

Did	Dname	Dacctno
4	human resources	528221
8	education	202035
9	accounts	709257
13	public relations	755827
15	services	223945

primary key

Employee Table

Ename	Did	SalaryCode	Eid	Ephone
Robin	15	23	2345	6127092485
Neil	13	12	5088	6127092246
Jasmine	4	26	7712	6127099348
Cody	15	22	9664	6127093148
Holly	8	23	3054	6127092729
Robin	8	24	2976	6127091945
Smith	9	21	4490	6127099380

foreign key — primary key

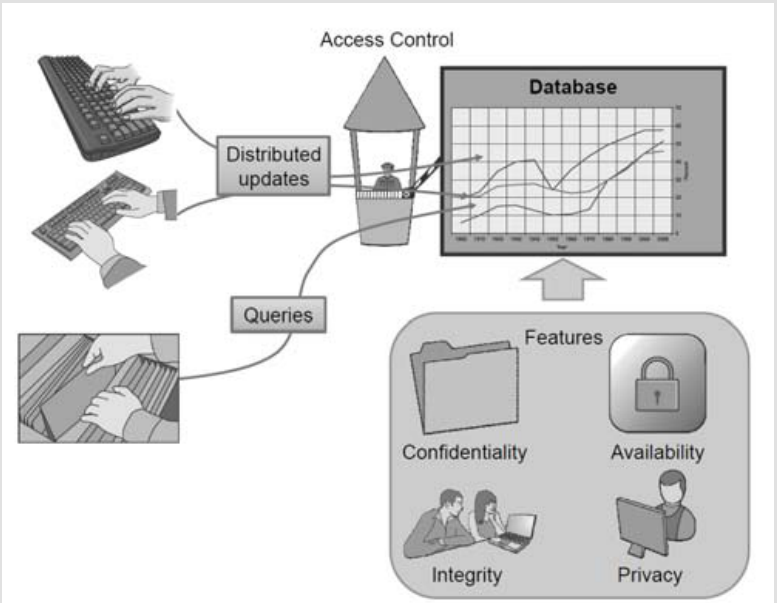
(a) Two tables in a relational database

Dname	Ename	Eid	Ephone
human resources	Jasmine	7712	6127099348
education	Holly	3054	6127092729
education	Robin	2976	6127091945
accounts	Smith	4490	6127099380
public relations	Neil	5088	6127092246
services	Robin	2345	6127092485
services	Cody	9664	6127093148

(b) A view derived from the database

The Need for Database Security

- **Because databases play such an important role in storing large amounts of potentially valuable information, they are often the target of attacks by malicious parties seeking to gain access to this data; hence, we need good ways to secure them.**



Security Requirements

- **Physical database integrity.**
  - The data are immune to physical problems
- **Logical database integrity**
  - The structure of the database is preserved
- **Element integrity**
  - The data contained in each element are accurate
- **Auditability**
  - possible to track who has accessed the elements
- **Access control**
- **User authentication**
- **Availability**

SQL Injection Attacks (SQLi)

- **One of the most prevalent and dangerous network-based security threats**
- **Designed to exploit the nature of Web application pages**
- **Sends malicious SQL commands to the database server**
- **Most common attack goal is bulk extraction of data**
- **Depending on the environment SQL injection can also be exploited to:**
  - Modify or delete data
  - Execute arbitrary operating system commands
  - Launch denial-of-service (DoS) attacks

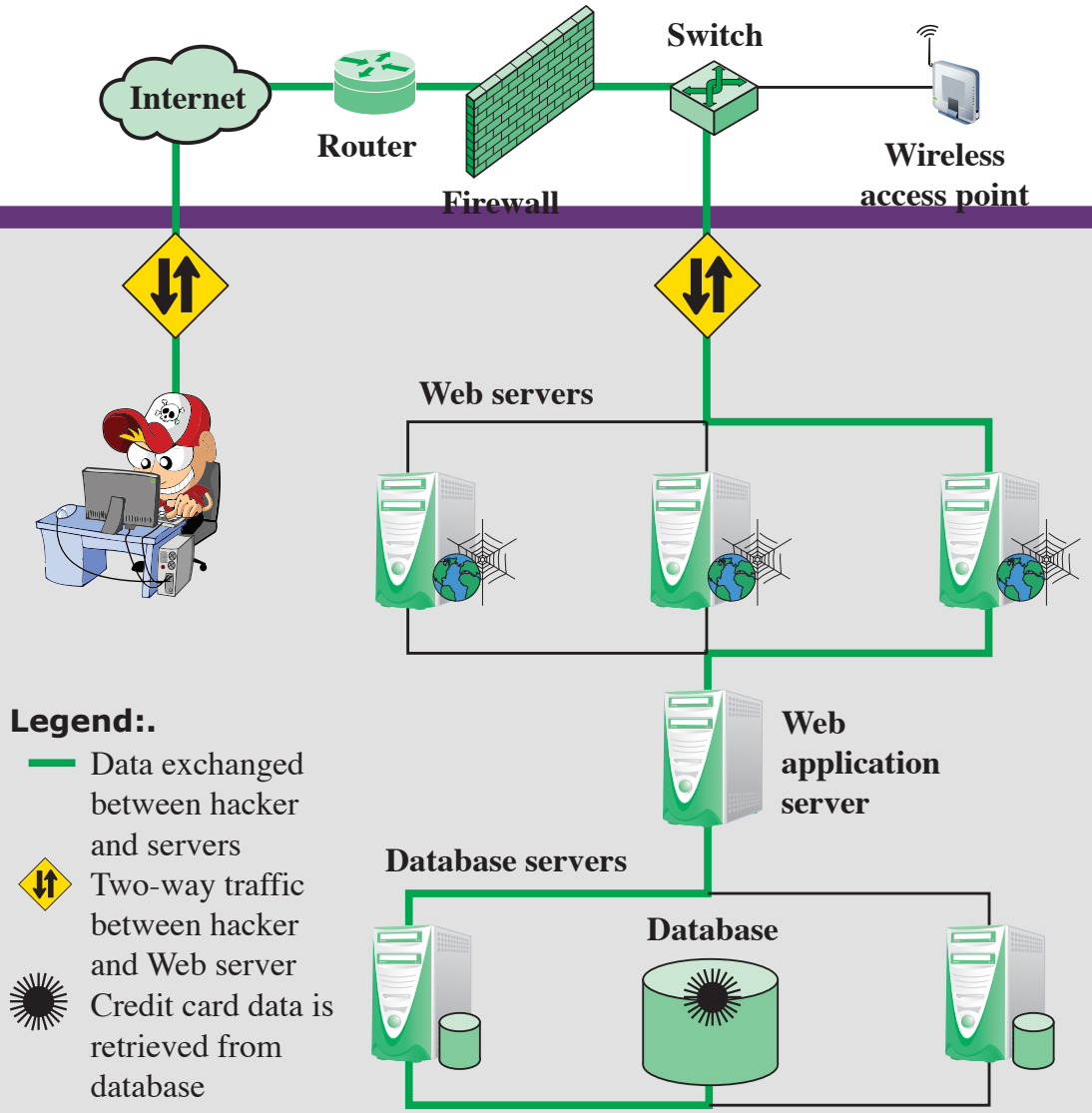


Figure 5.5 Typical SQL Injection Attack

Injection Technique

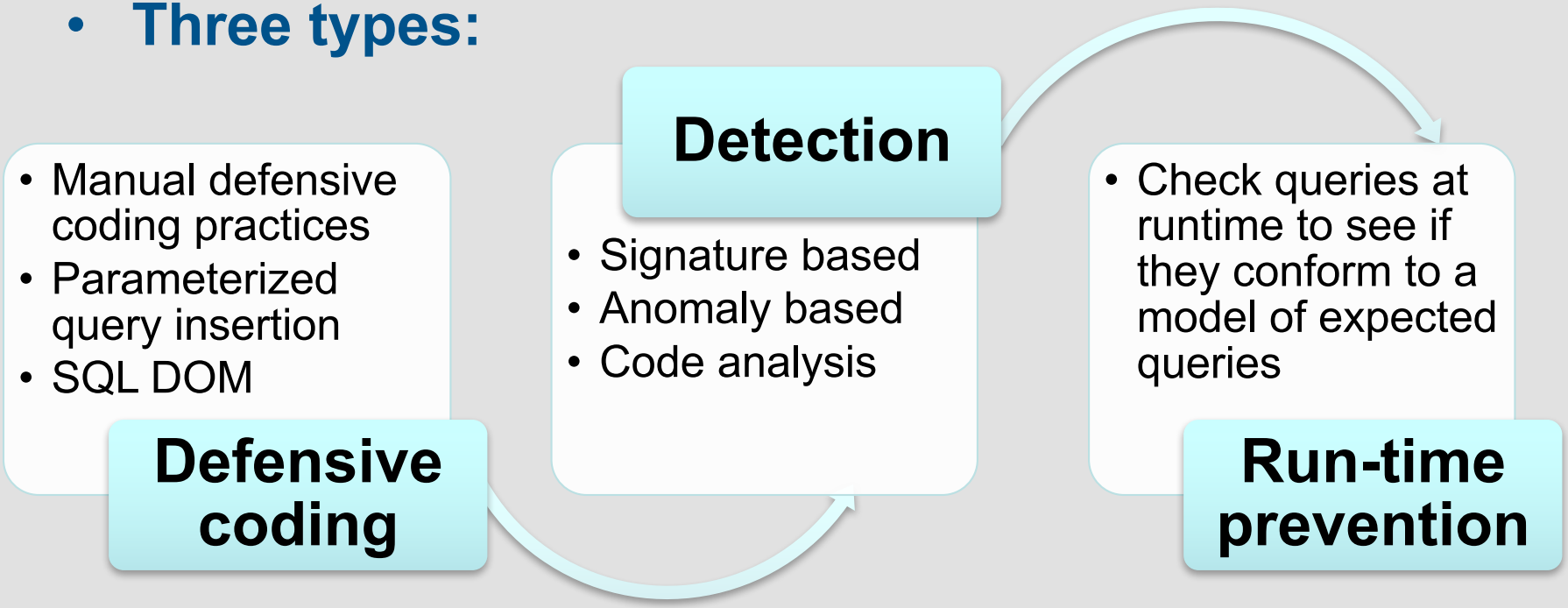
The SQLi attack typically works by prematurely terminating a text string and appending a new command

Because the inserted command may have additional strings appended to it before it is executed the attacker terminates the injected string with a comment mark “- -”

Subsequent text is ignored at execution time

SQLi Countermeasures

- **Three types:**



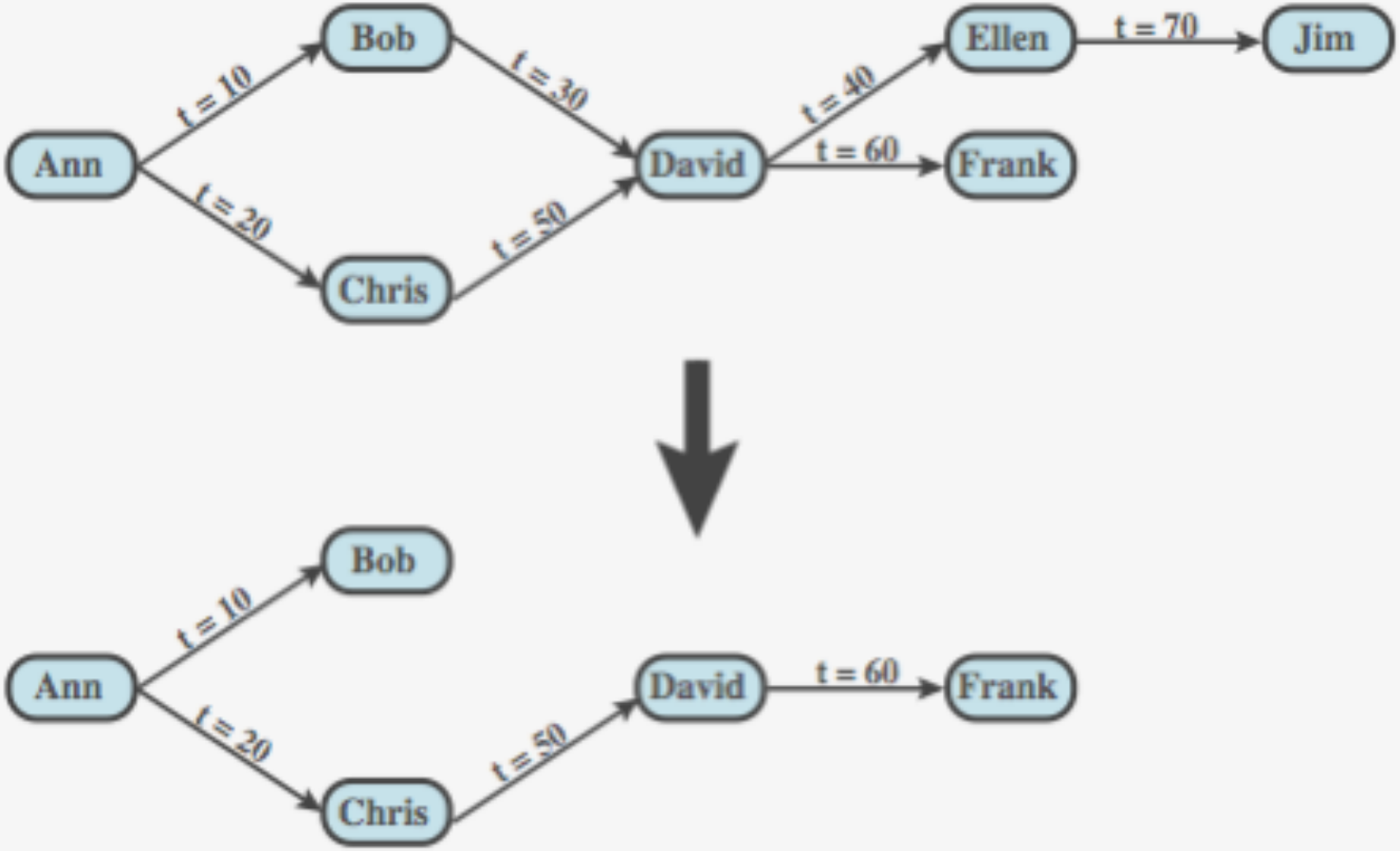
Database Access Control

- **database access control system determines:**
  - **if the user has access to the entire database or just portions of it**
  - **what access rights the user has (create, insert, delete, update, read, write)**
  - **Commercial systems provide DAC or RBAC**
- **can support a range of administrative policies**
  - **centralized administration**
    - **small number of privileged users may grant and revoke access rights**
  - **ownership-based administration**
    - **the creator of a table may grant and revoke access rights to the table**
  - **decentralized administration**
    - **the owner of the table may grant and revoke authorization rights to other users, allowing them to grant and revoke access rights to the table**

Privilege Delegation & Revocation

- **In addition to being able to grant certain privileges to other users, table owners can also allow other users to grant privileges for those tables**
- **The propagation of privileges in a database can be visualized using a diagram, where nodes represent users and directed edges represent granted privileges.**
- **A user, Alice, who has granted privileges to another, Bob, can opt to revoke those privileges at a later time**

Cascading Authorizations



Role-Based Access Control

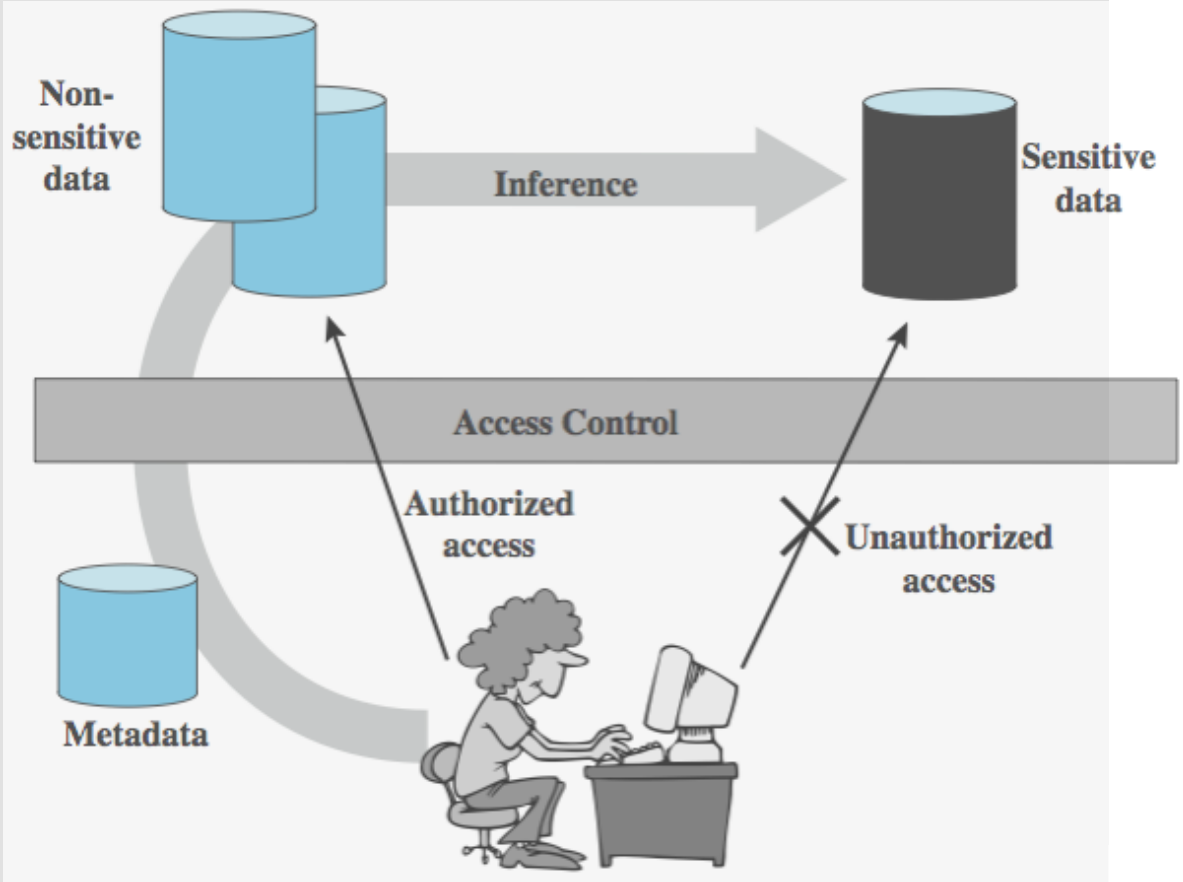
- **role-based access control eases administrative burden, improves security**
- **In a DAC environment, database users broad categories:**
  - **application owner**
    - > an end user who **owns database objects** as part of an application
  - **end user**
    - > an end user **who operates on database** objects via a particular application but does not own any of the database objects
  - **administrator**
    - > user who has **administrative responsibility** for part or all of the database
- **a database RBAC needs to provide the following capabilities:**
  - > create and delete roles
  - > define permissions for a role
  - > assign and cancel assignment of users to roles

Sensitive Data

- **sensitive data should not be made public**
- **stored in a way that protects**
  - the privacy of users and
    - > careful to consider the privacy impacts of publishing or granting access
    - > If released to the public (for research) then all identifying information be removed or changed to **masking values** be used
  - any confidentiality requirements for sensitive data
    - > should instead be stored as the output of a cryptographic function
    - > For encrypted data, decryption key known only by authorized users, but not stored in the database itself

Inference

- **the process of performing queries and deducing unauthorized information from the legitimate responses received**
- **inference channel**
  - is the information transfer path by which unauthorized data is obtained



Inference Example

- **Analyzing functional dependencies**
- **Merging views with same constraints**

Name	Position	Salary ($)	Department	Dept. Manager
Andy	senior	43,000	strip	Cathy
Calvin	junior	35,000	strip	Cathy
Cathy	senior	48,000	strip	Cathy
Dennis	junior	38,000	panel	Herman
Herman	senior	55,000	panel	Herman
Ziggy	senior	67,000	panel	Herman

(a) Employee table

Position	Salary ($)
senior	43, 000
junior	35,000
senior	48,000

Name	Department
Andy	strip
Calvin	strip
Cathy	strip

(b) Two views

Name	Position	Salary ($)	Department
Andy	senior	43,000	strip
Calvin	junior	35,000	strip
Cathy	senior	48,000	strip

(c) Table derived from combining query answers

Protecting Against Inference Attacks

- **To protect a database from inference attacks, the following techniques can be used prior to making the database public.**
  - **Cell suppression**.
    > some of the cells in a database are removed and left blank in the published version.
  - **Generalization.**
    > some values in a published database are replaced with more general values.
  - **Noise addition**.
    > values in a published database have random values added to them, so that the noise across all records for the same attribute averages out to zero.

Example

Num	Age1	Age2
11	49.3	53.6
18	46.9	63.2
20	49.3	49.8
35	43.6	46.5
42	46.4	
44	47.5	

(a)

Num	Age1	Age2
11	49.3	
18	46.9	63.2
20	49.3	
35		
42	46.4	
44	47.5	

(b)

Num	Age1	Age2
11	45–50	50–60
18	45–50	60–75
20	45–50	45–50
35	40–45	45–50
42	45–50	
44	45–50	

(c)

Num	Age1	Age2
11	47.7	55.2
18	49.2	64.3
20	51.6	52.8
35	42.3	47.3
42	47.1	
44	48.0	

(d)

Figure 10.4: Obfuscation techniques for protecting the privacy of individuals included in a public database: (a) A table with individual names removed. (b) A table anonymized using cell suppression. (c) A table anonymized using generalization. (d) A table anonymized using noise.

Inference Countermeasures

- **inference detection at database design**
  - alter database structure or access controls
- **inference detection at query time**
  - by monitoring and altering or rejecting queries
- **need some inference detection algorithm**
  - a difficult problem
  - cf. employee-salary example
  - Subject of on-going research

Statistical Databases

- **provides data of a statistical nature such as counts, averages**
- **two types:**
  - pure statistical database
  - ordinary database with statistical access
    - contains individual entries
    - uses DAC, MAC, and RBAC
- **access control objective is to provide users with the needed information without compromising the confidentiality of the database**
- **security problem is one of inference**

Statistical Database Example

(a) Database with statistical access with $N = 13$ students

Name	Sex	Major	Class	SAT	GP
Allen	Female	CS	1980	600	3.4
Baker	Female	EE	1980	520	2.5
Cook	Male	EE	1978	630	3.5
Davis	Female	CS	1978	800	4.0
Evans	Male	Bio	1979	500	2.2
Frank	Male	EE	1981	580	3.0
Good	Male	CS	1978	700	3.8
Hall	Female	Psy	1979	580	2.8
Iles	Male	CS	1981	600	3.2
Jones	Female	Bio	1979	750	3.8
Kline	Female	Psy	1981	500	2.5
Lane	Male	EE	1978	600	3.0
Moore	Male	CS	1979	650	3.5

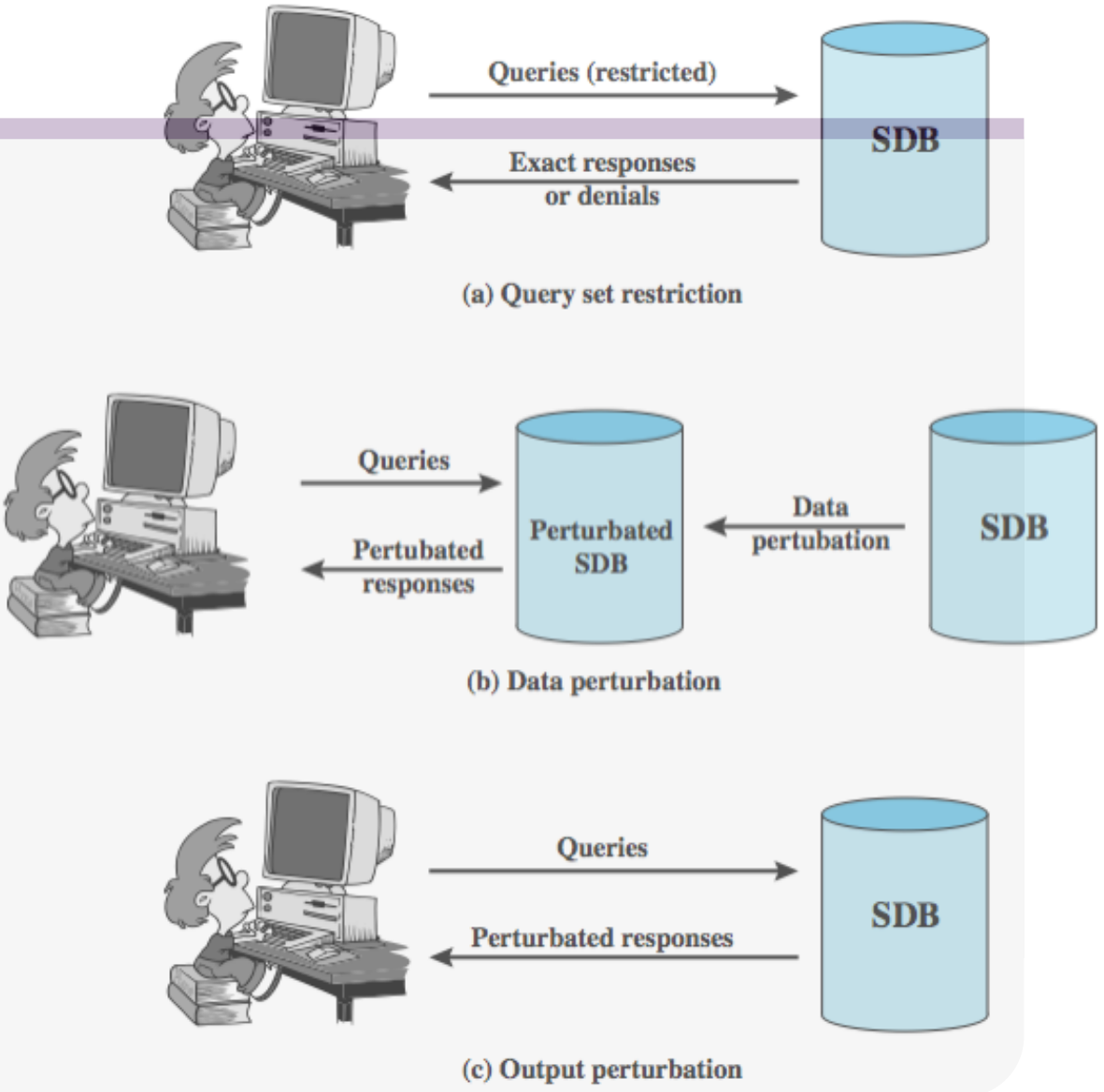
(b) Attribute values and counts

Attribute A_j	Possible Values	$\lvert A_j \rvert$
Sex	Male, Female	2
Major	Bio, CS, EE, Psy, ...	50
Class	1978, 1979, 1980, 1981	4
SAT	310, 320, 330, ..., 790, 800	50
GP	0.0, 0.1, 0.2, ..., 3.9, 4.0	41

Statistical Database Security

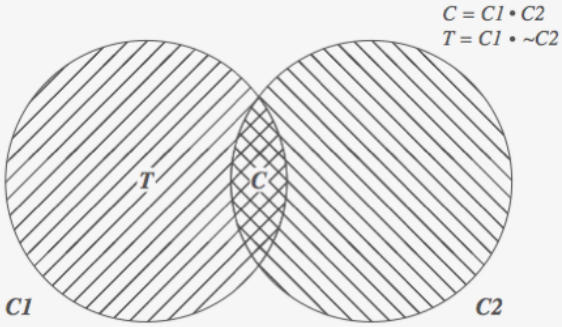
- **use a characteristic formula C**
  - a logical formula over the values of attributes
  - e.g. `(`*`Sex`*`=Male) AND ((`*`Major`*`=CS) OR (`*`Major`*`=EE))`
- **query set X(*C*) of characteristic formula *C*, is the set of records matching C**
- **a statistical query is a query that produces a value calculated over a query set**
  - E.g. Statistics are count, sum, average, median, max, min.

Protecting Against Inference



Tracker Attacks

- **Query size restriction ➔ k ≤ |X(C)| ≤ N-k**
- **divide queries into parts**
  - C = C1.C2
  - count(C) = count(C1) - count (C1. ~*C2)*
- **combination is called a tracker**
- **each part acceptable query size**
- **overlap is desired result**



Other Query Restrictions

- **query set overlap control**
  - limit overlap between new & previous queries
  - has problems and overheads
- **partitioning**
  - cluster records into a number of mutually exclusive groups
  - query the statistical properties of each group as a whole
- **query denial and information leakage**
  - denials can leak information
  - to counter must track queries from user

Perturbation

- **add noise to statistics generated from data**
- **data perturbation techniques**
  - data can be modified to produce statistics that cannot be used to infer values for individual records
- **output perturbation techniques**
  - random-sample query
  - system generates statistics that are modified from those that the original database would provide
- **goal is to minimize the differences between original results and perturbed results**
- **main challenge is to determine the average size of the error to be used**

Data Perturbation Techniques: Data Swapping

Table 5.6 Example of Data Swapping

Record	D			D'		
	Sex	Major	GP	Sex	Major	GP
1	Female	Bio	4.0	Male	Bio	4.0
2	Female	CS	3.0	Male	CS	3.0
3	Female	EE	3.0	Male	EE	3.0
4	Female	Psy	4.0	Male	Psy	4.0
5	Male	Bio	3.0	Female	Bio	3.0
6	Male	CS	4.0	Female	CS	4.0
7	Male	EE	4.0	Female	EE	4.0
8	Male	Psy	3.0	Female	Psy	3.0

Database Encryption

- **databases typically a valuable info resource**
  - protected by multiple layers of security: firewalls, authentication, O/S access control systems, DB access control systems, and database encryption
- **encryption is often implemented with particularly sensitive data**
  - can be applied to the entire database at the record level, the attribute level, or level of the individual field
- **disadvantages of encryption:**
  - key management
  - inflexibility

Database Encryption

- **Data owner** – organization that produces data
- **User** – human entity that presents queries
- **Client** – frontend that transforms user queries into queries on the encrypted data
- **Server** – an organization that receives the encrypted data from a data owner and makes them available for distribution to clients

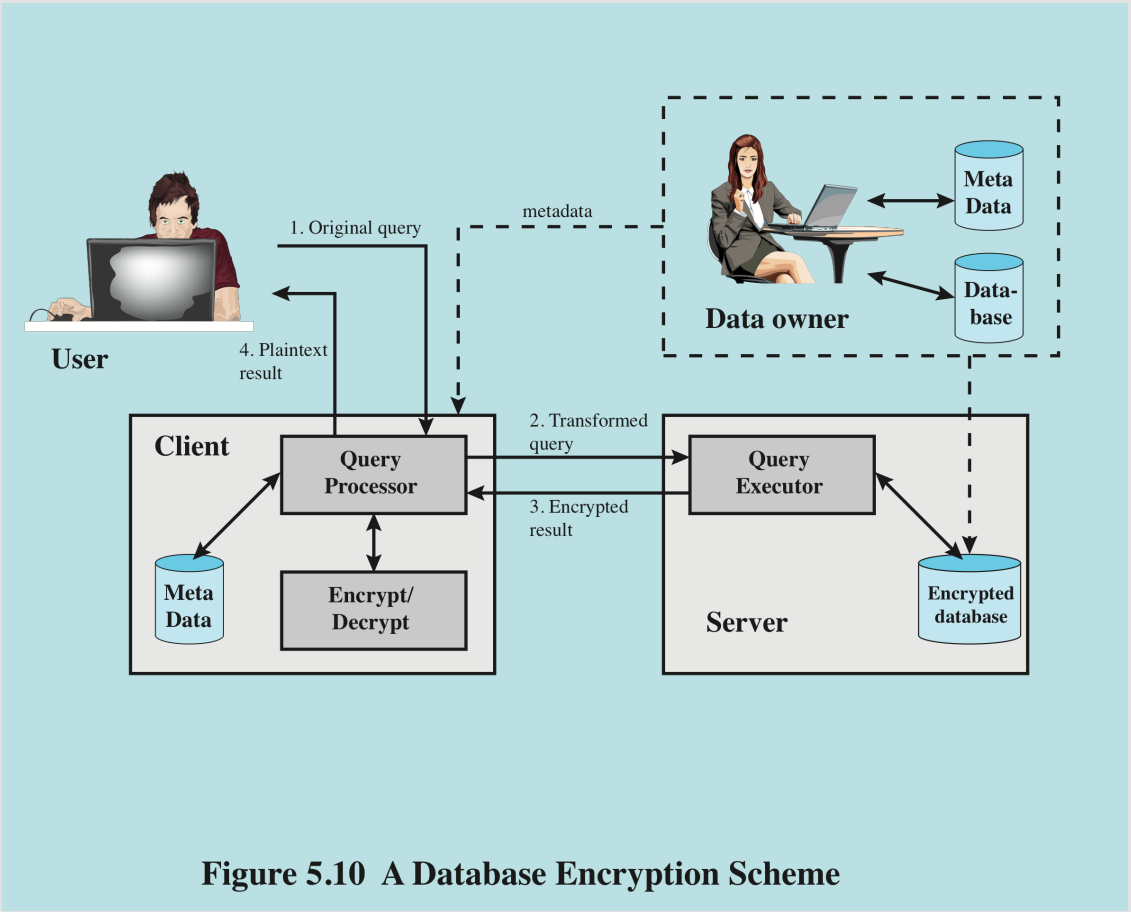


Figure 5.10 A Database Encryption Scheme

Table 5.7 Encrypted Database Example

(a) Employee Table

eid	ename	salary	addr	did
23	Tom	70K	Maple	45
860	Mary	60K	Main	83
320	John	50K	River	50
875	Jerry	55K	Hopewell	92

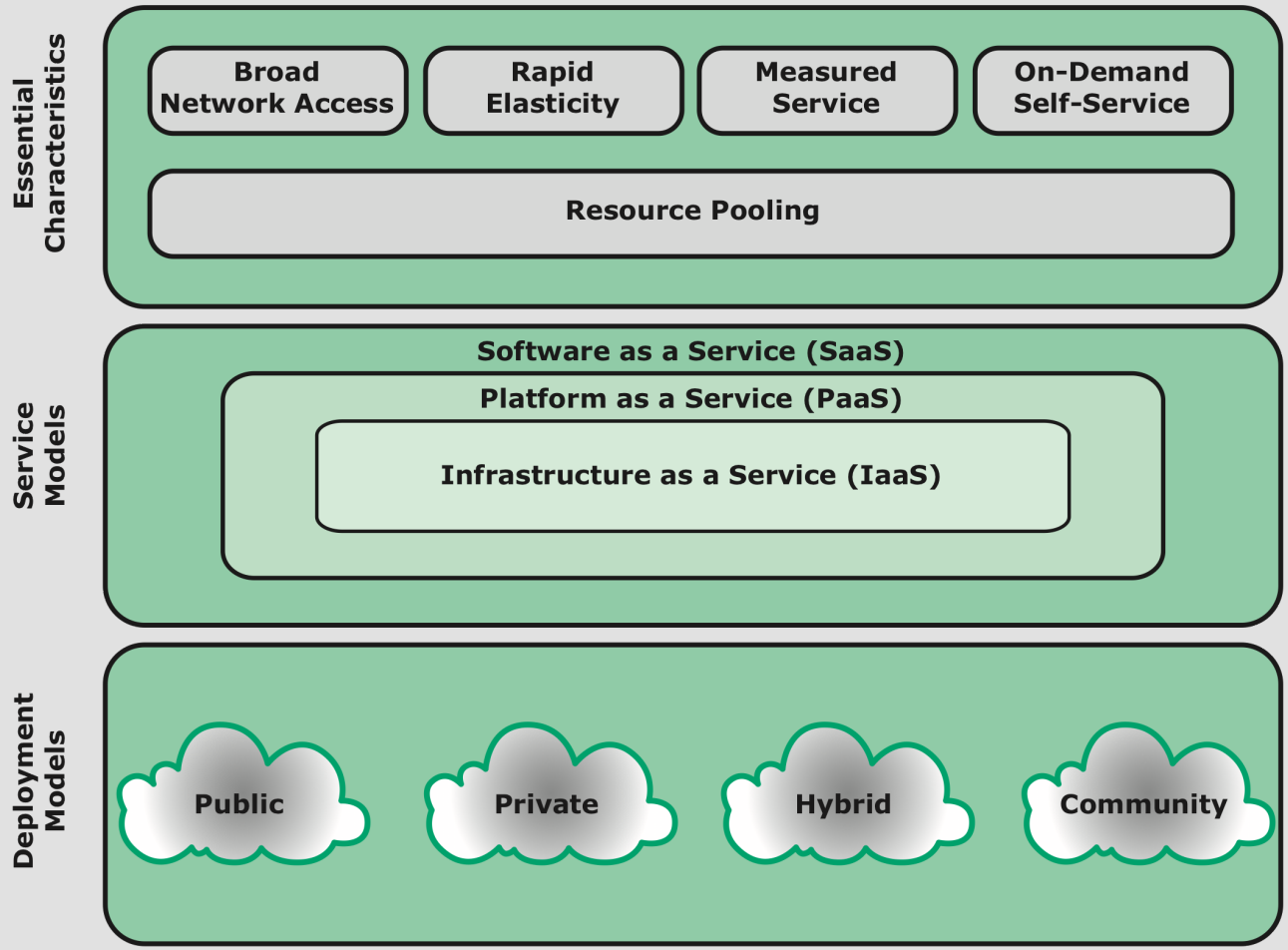
(b) Encrypted Employee Table with Indexes

E(k, B)	I(eid)	I(ename)	I(salary)	I(addr)	I(did)
1100110011001011…	1	10	3	7	4
0111000111001010…	5	7	2	7	8
1100010010001101…	2	5	1	9	5
0011010011111101…	5	5	2	4	9

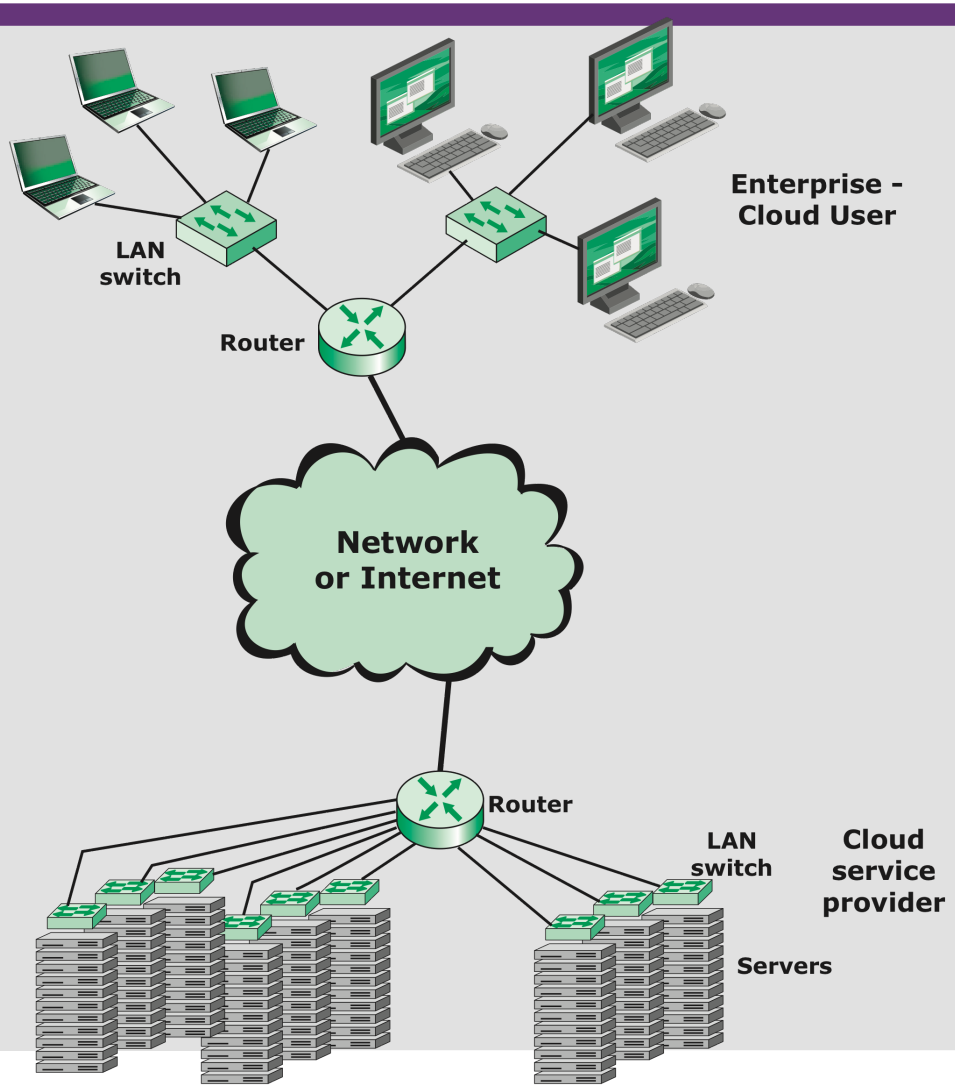
Cloud Security

- **NIST defines cloud computing as follows [MELL11]:**
- **"A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models."**

Cloud Computing Elements



Cloud Computing Context



Cloud Security Risks

- abuse and nefarious use of cloud computing
- insecure interfaces and APIs
- malicious insiders
- shared technology issues
- data loss or leakage
- account or service hijacking
- unknown risk profile

Data Protection in the Cloud

- **multi-instance model**
  - provides a unique DBMS running on a virtual machine instance for each cloud subscriber
  - gives the subscriber complete control over administrative tasks related to security
- **multi-tenant model**
  - provides a predefined environment for the cloud subscriber that is shared with other tenants typically through tagging data with a subscriber identifier
  - gives the appearance of exclusive use of the instance but relies on the cloud provider to establish and maintain a secure database environment

Summary

- **Security Issues in a relational databases**
- **SQL Injection Attacks**
- **database access control issues**
  - Privilege delegation/revocation, role-based
- **inference**
- **statistical database security issues**
- **database encryption**
- **Cloud computing security & data protection**

Further Reading

- **Chapter 5 of the textbook: *Computer Security: Principles and Practice" by William Stallings & Lawrie Brown,* Prentice Hall, 2015**

- **Acknowledgement: part of the materials presented in the slides was developed with the help of Instructor's Manual and other resources made available by the author of the textbook.**