ĐẠI HỌC TÔN ĐỨC THẮNG
TON DUC THANG UNIVERSITY

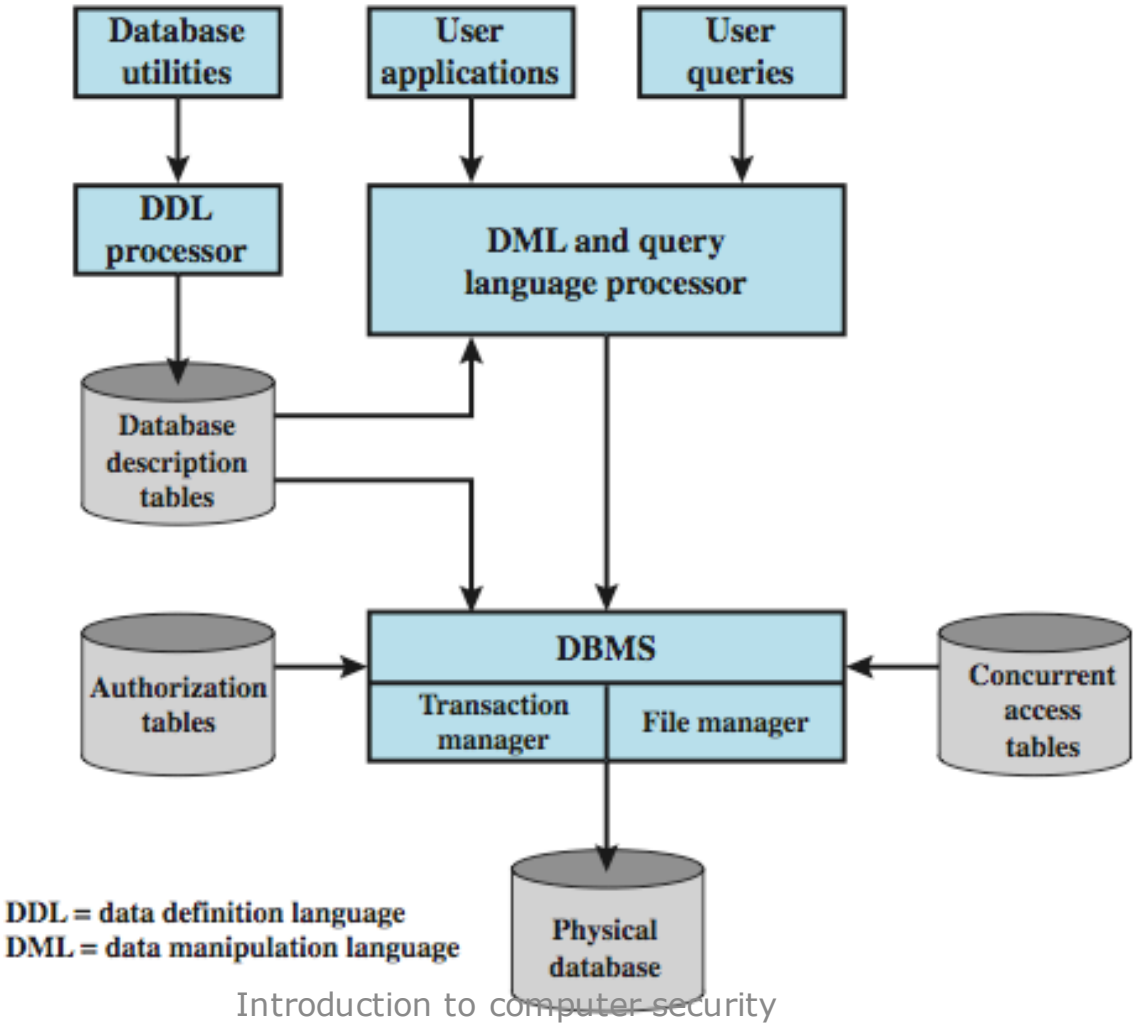
Chapter 8
Database and Cloud Security

Book Reading: Computer Security Principles and Practice (3ed), 2015, p.155-189

Database systems

- 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
- Query language: Provides a uniform interface to the database

Database Security



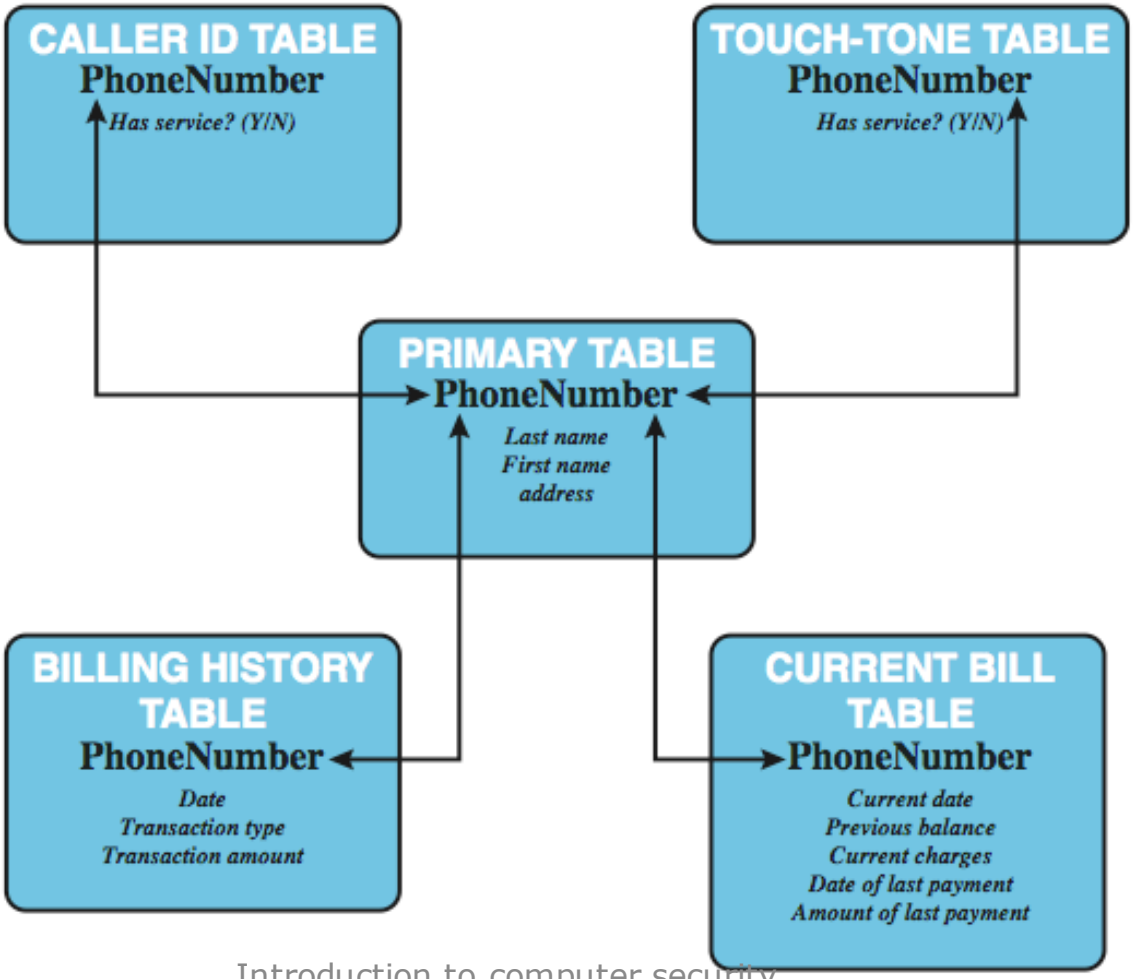
Relational Databases

- Constructed from tables of data
  - each column holds a particular type of data
  - each row contains a specific value these
  - ideally has one column where all values are unique, forming an identifier/key for that row
- Have multiple tables linked by identifiers
- Sse a query language to access data items meeting specified criteria

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

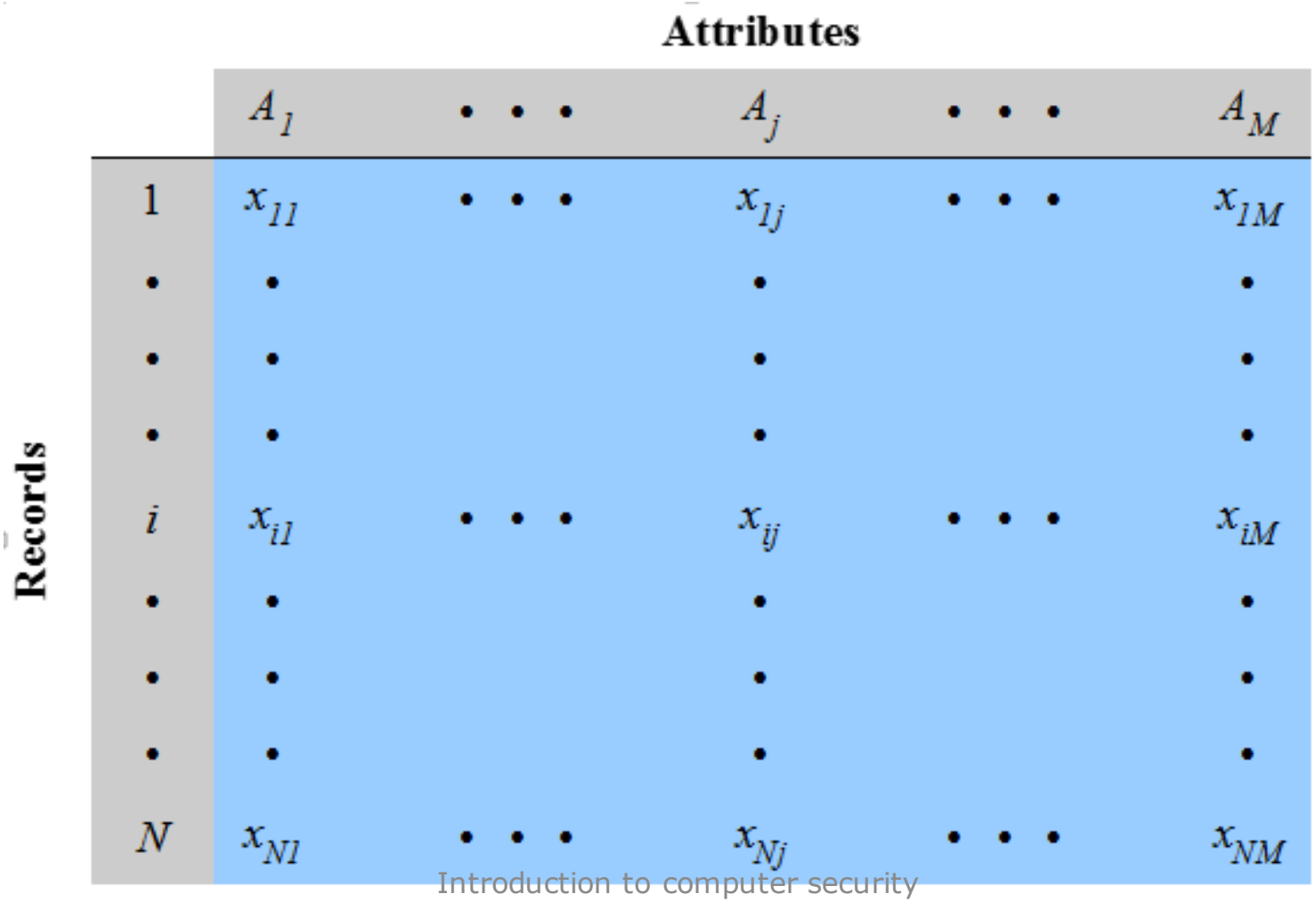
A relational database example



Relational database terms

- Relation/table/file
- Tuple/row/record
- Attribute/column/field
- Primary key: uniquely identifies a row
- Foreign key: links one table to attributes in another
- View/virtual table: Result of a query that returns selected rows and columns from one or more tables

Abstract view of a relation



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

Structured Query Language

- Structure Query Language (SQL)
  - originally developed by IBM in the mid-1970s
  - standardized language to define, manipulate, and query data in a relational database
  - several similar versions of ANSI/ISO standard

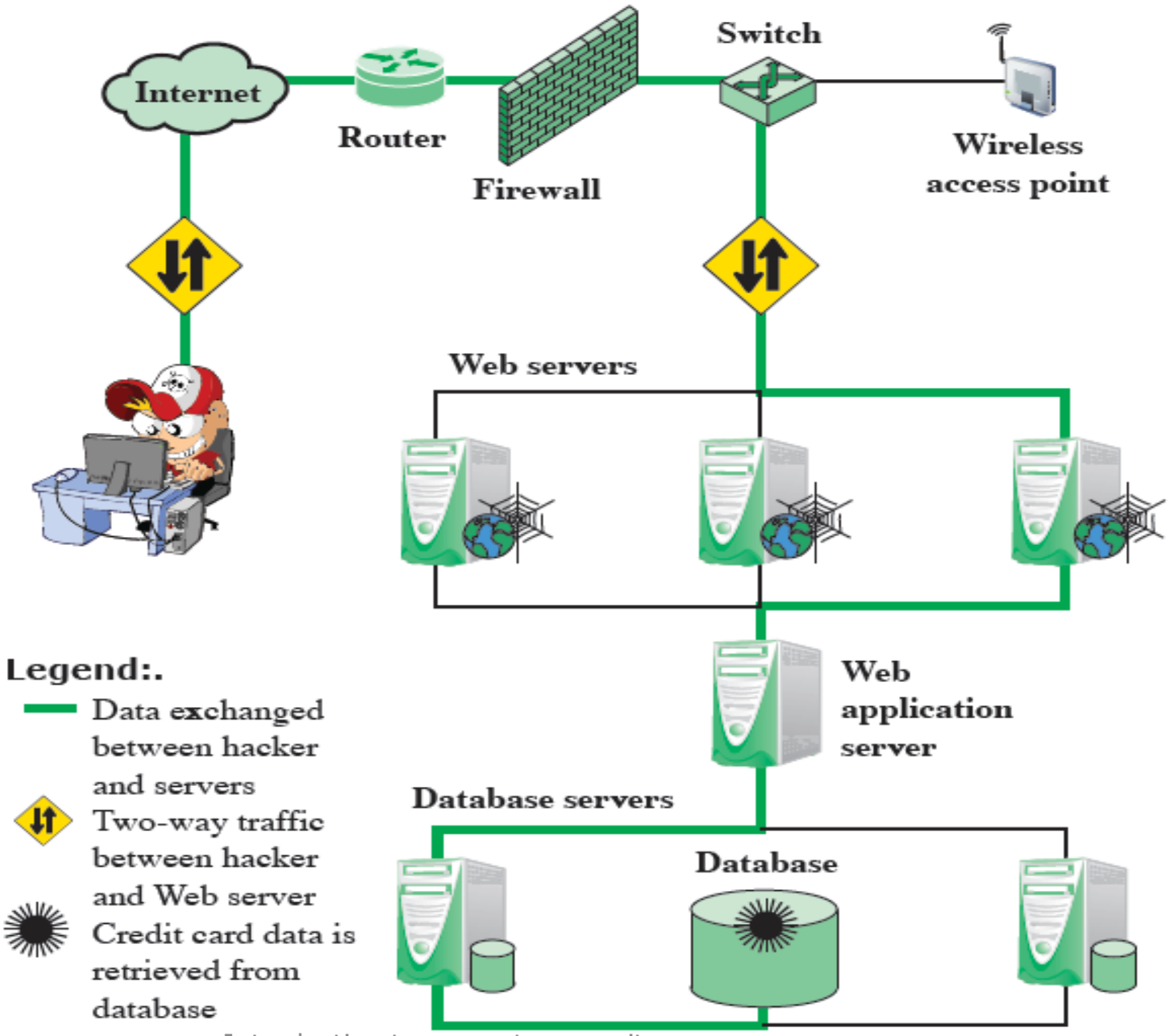
```
CREATE TABLE department (
          Did INTEGER PRIMARY KEY,
          Dname CHAR (30),
          Dacctno CHAR (6) )

CREATE TABLE employee (
          Ename CHAR (30),
          Did INTEGER,
          SalaryCode INTEGER,
          Eid INTEGER PRIMARY KEY,
          Ephone CHAR (10),
          FOREIGN KEY (Did) REFERENCES department (Did) )
```

```
CREATE VIEW newtable (Dname, Ename, Eid, Ephone)
AS SELECT D.Dname E.Ename, E.Eid, E.Ephone
FROM Department D Employee E
WHERE E.Did = D.Did
```

SQL injection attacks

- One of the most prevalent and dangerous network-based security threats
- Sends malicious SQL commands to the database server
- 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


Internet
Router
Firewall
Switch
Wireless access point
Web servers
Web application server
Database servers
Database
Legend:.
Data exchanged between hacker and servers
Two-way traffic between hacker and Web server
Credit card data is retrieved from database

Sample SQL injection

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

SELECT fname

FROM student

where fname is ‘`user prompt`’;

User: John’; DROP table Course;--

Sample SQL injection: tautology

$query= “
SELECT info FROM user WHERE name =
`$_GET[“name”]’ AND pwd = `GET[“pwd”]`
”;

Attacker enters: ` OR 1=1 –-

In-band attacks

- **Tautology**: This form of attack injects code in one or more conditional statements so that they always evaluate to true
- **End-of-line comment**: After injecting code into a particular field, legitimate code that follows are nullified through usage of end of line comments
- **Piggybacked queries**: The attacker adds additional queries beyond the intended query, piggy-backing the attack on top of a legitimate request

Database Access Control

- DBMS provide access control for database
- assume have authenticated user
- DBMS provides specific access rights to portions of the database
  - e.g. create, insert, delete, update, read, write
  - to entire database, tables, selected rows or columns
  - possibly dependent on contents of a table entry
- can support a range of policies:
  - centralized administration
  - ownership-based administration
  - decentralized administration

Inferential attack (gathering info)

- There is no actual transfer of data, but the attacker is able to reconstruct the information by sending particular requests and observing the resulting behavior of the Website/database server
  - Illegal/logically incorrect queries: lets an attacker gather important information about the type and structure of the backend database of a Web application

Out-band attack

- This can be used when there are limitations on information retrieval, but outbound connectivity from the database server is lax

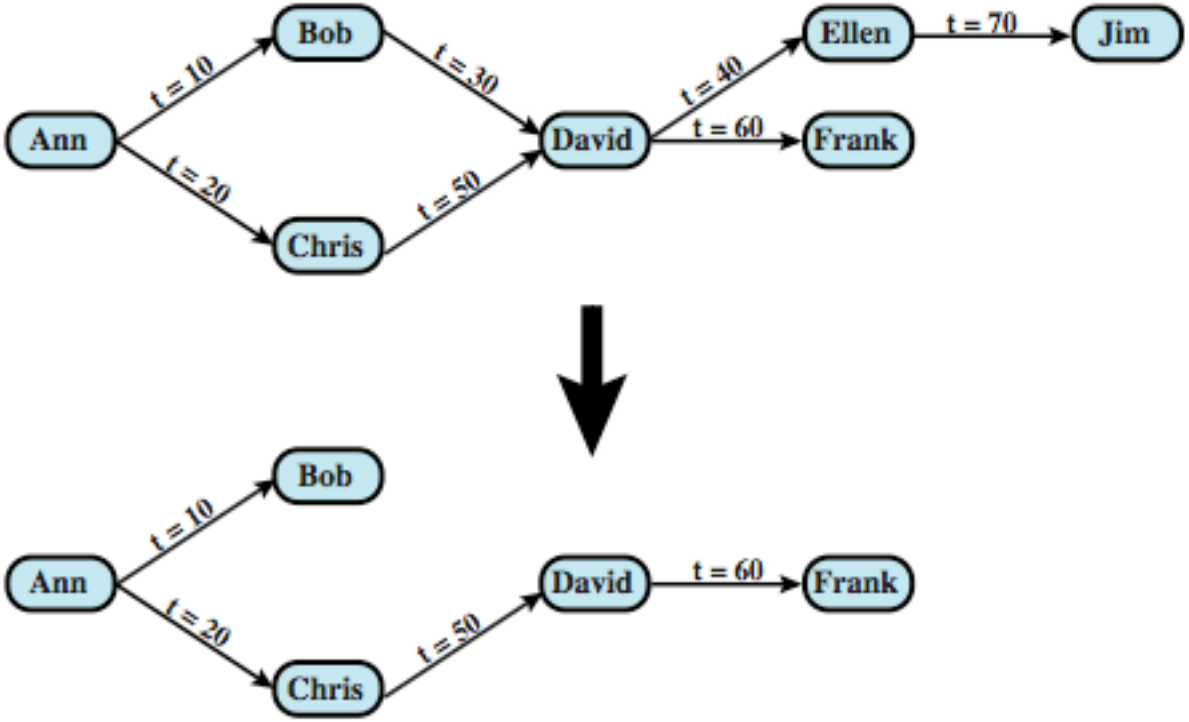
SQLi countermeasures

- Defensive coding: stronger data validation
- Detection
  - Signature based
  - Anomaly based
  - Code analysis
- Runtime prevention: Check queries at runtime to see if they conform to a model of expected queries

SQL Access Controls

- If the user has access to the entire database or just portions of it
- Two commands:
  - `GRANT {privileges | role} [ON table] TO {user | role | PUBLIC} [IDENTIFIED BY password] [WITH GRANT OPTION]`
    - e.g. GRANT SELECT ON ANY TABLE TO john
  - `REVOKE {privileges | role} [ON table] FROM {user | role | PUBLIC}`
    - e.g. REVOKE SELECT ON ANY TABLE FROM john
  - `WITH GRANT OPTION: whether grantee can grant "GRANT" option to other users`
- Typical access rights are:
  - `SELECT, INSERT, UPDATE, DELETE, REFERENCES`

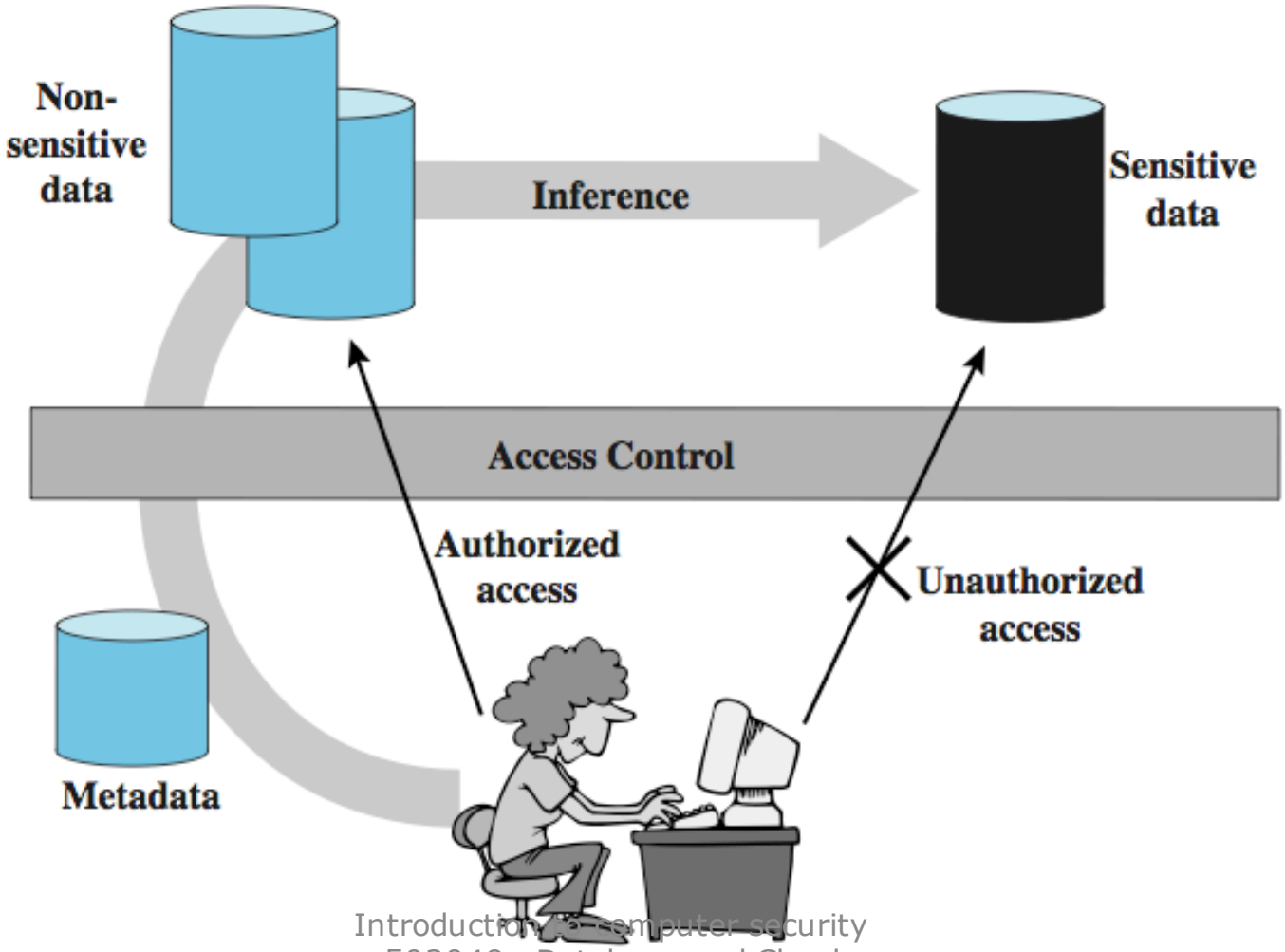
Cascading Authorizations



Role-Based Access Control

- Role-based access control work well for DBMS
  - eases admin burden, improves security
- Categories of database users:
  - application owner
  - end user
  - administrator
- DB RBAC must manage roles and their users

Inference


Non-
sensitive
data
Inference
Sensitive
data
Access Control
Authorized
access
Unauthorized
access
Metadata

Inference Example

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 obtained from combining query answers

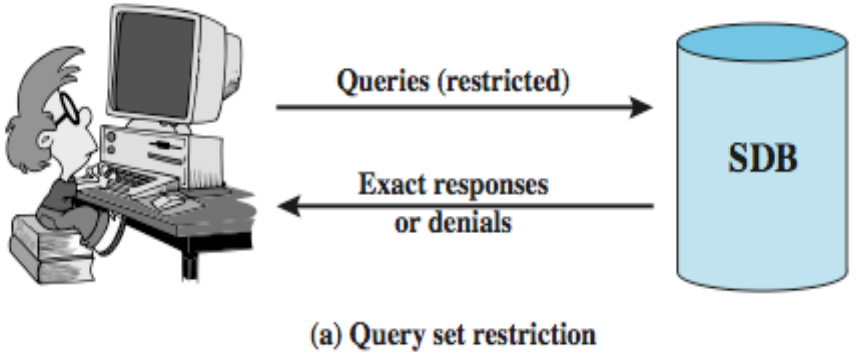
Inference Countermeasures

- Inference detection at database design
  - alter database structure or access controls
- Inference detection at query time
  - by monitoring and altering or rejecting queries

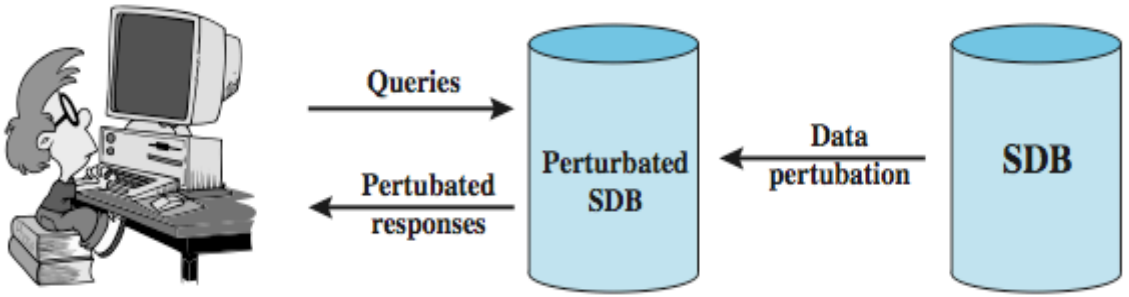
Statistical Databases

- Provides data of a statistical nature
  - e.g. counts, averages
- Two types:
  - pure statistical database
  - ordinary database with statistical access
    - some users have normal access, others statistical
- Access control objective to allow statistical use without revealing individual entries
- Security problem is one of inference

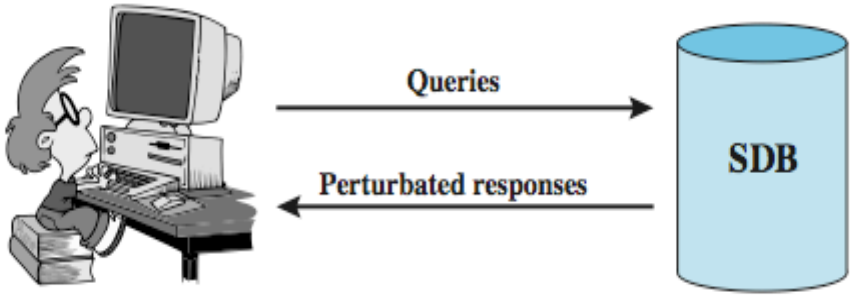
Protecting Against Inference



(a) Query set restriction



(b) Data perturbation



(c) Output perturbation

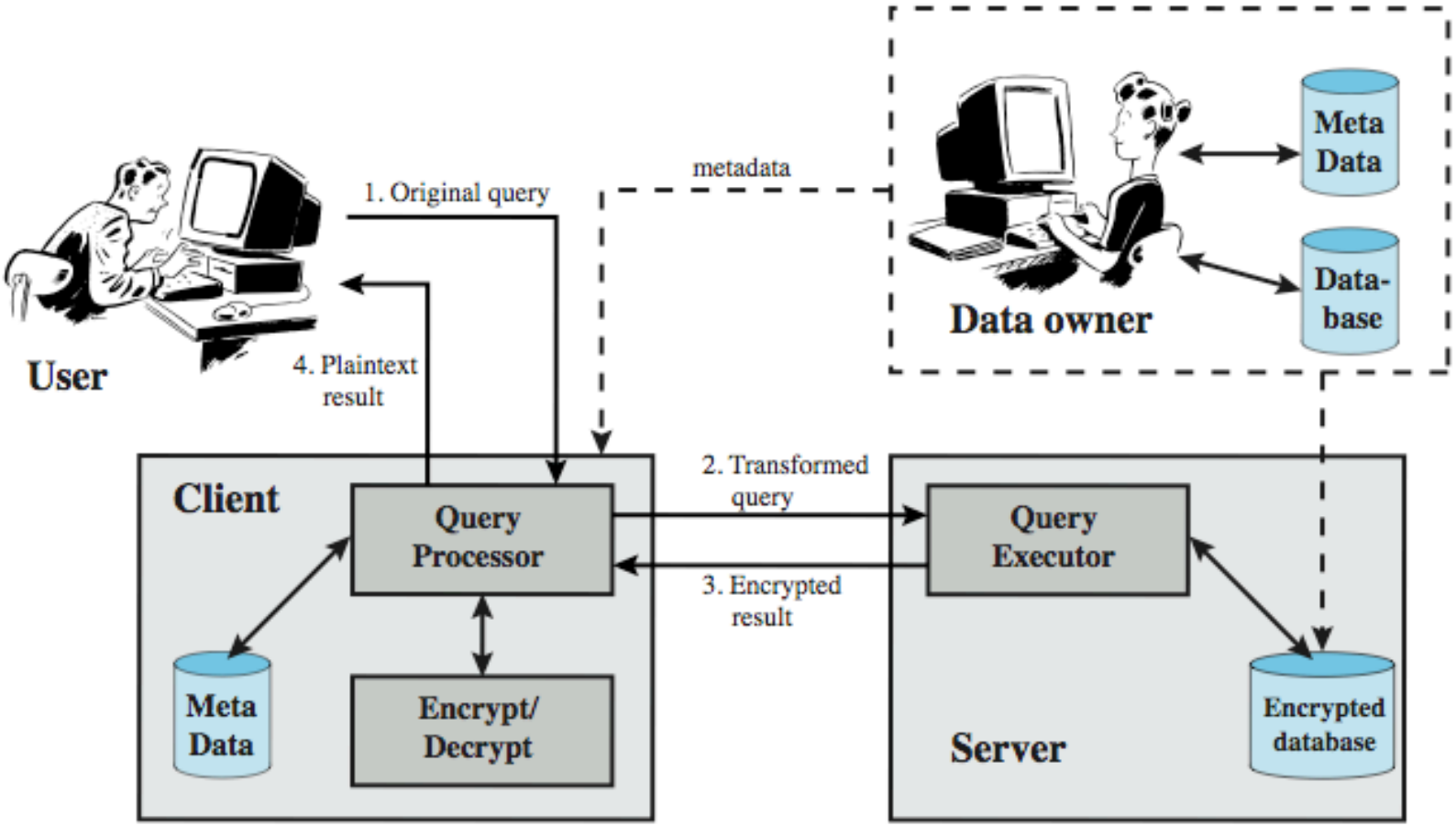
Perturbation

- Add noise to statistics generated from data
  - will result in differences in statistics
- Data perturbation techniques
  - data swapping
  - generate statistics from probability distribution
- Output perturbation techniques
  - random-sample query
  - statistic adjustment
- Must minimize loss of accuracy in results

Database Encryption

- Databases typical a valuable info resource
  - protected by multiple layers of security: firewalls, authentication, O/S access control systems, DB access control systems, and database encryption
- Can encrypt
  - entire database - very inflexible and inefficient
  - individual fields - simple but inflexible
  - records (rows) or columns (attributes) - best
    - also need attribute indexes to help data retrieval
- Varying trade-offs

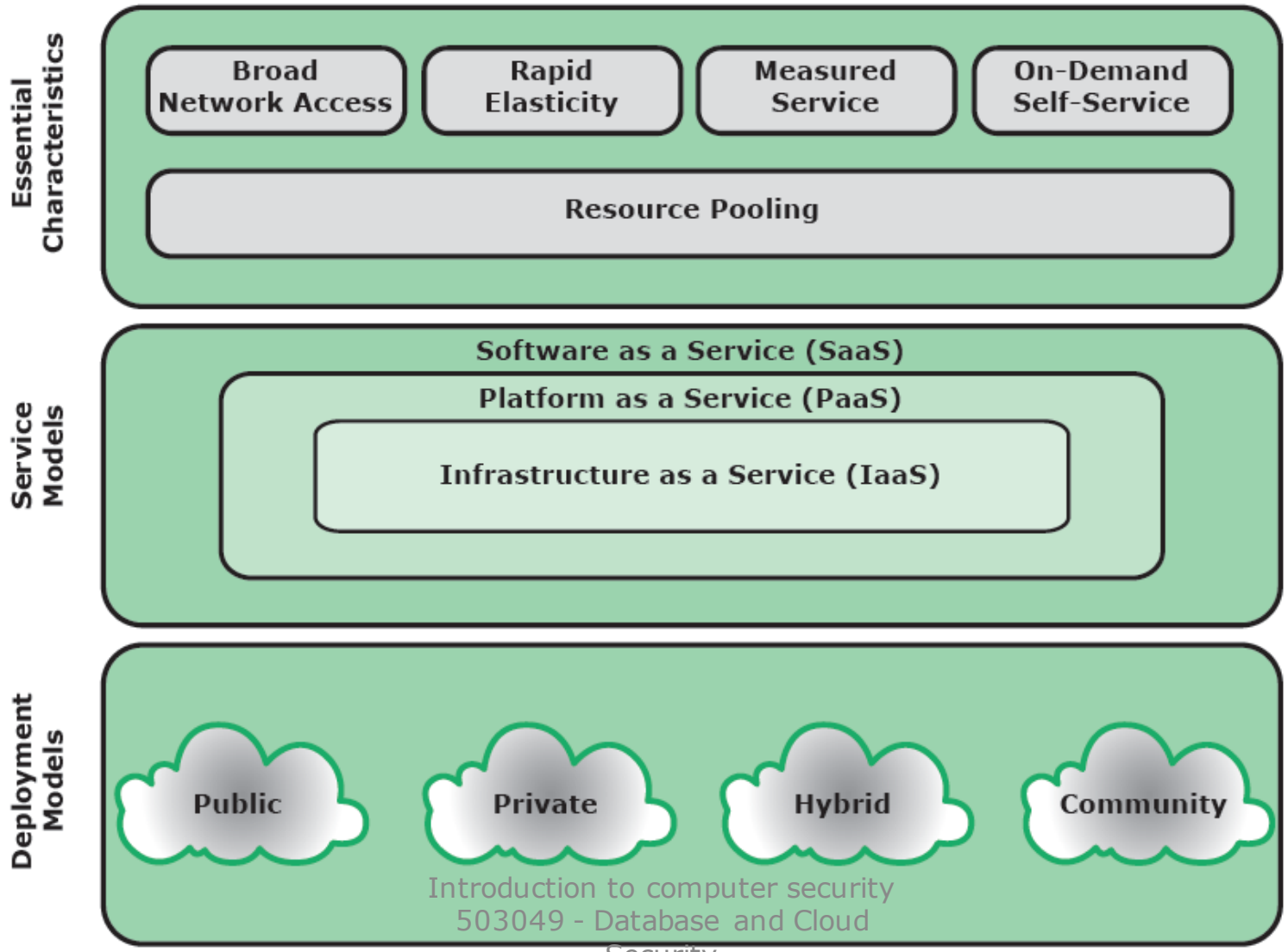
Database Encryption



Cloud computing

- An increasing trend in many organizations to move a substantial portion (or all) of their IT to cloud computing
- NIST SP-800-145 defines cloud computing as: "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. …"

Cloud computing elements



Essential characteristics: Broad network access

- Capabilities available over a network
- Accessed thru standard mechanisms
- Use by heterogeneous client platforms (desktops, laptops, tablets, cell phones)

Essential characteristics: Rapid elasticity

- The ability to expand/reduce services to specific requirements
- Large numbers of servers during the holidays
- Smaller number of resources during off-peak periods
- Release a resource when a task is completed

Essential characteristics: Measured service

- Cloud system automatically
  - Control and optimize resource use by leveraging a metering capability appropriate to the type of service
  - Storage, processing, bandwidth, active users

Essential characteristics: On-demand service

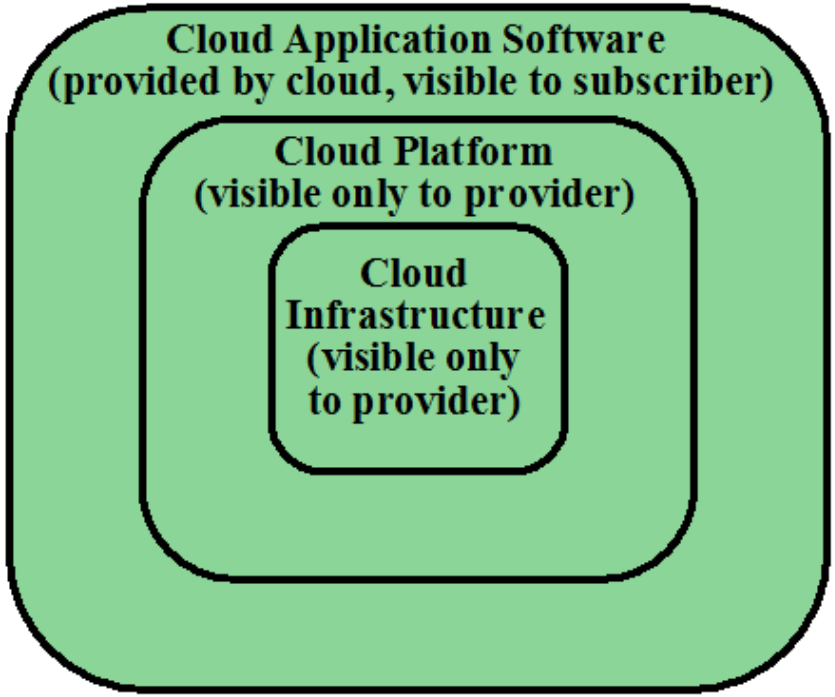
- A consumer can unilaterally provision computing capabilities without requiring human interaction
  - Server time, network storage
  - Resources wont have to a permanent part of the client's IT infrastructure

Essential characteristics: Resource pooling

- As a consequence of the above
- The providers resources are pooled to serve multiple consumers using a multi-tenant model
- Multiple resources assigned and re-assigned to clients
  - Storage, processing power, bandwidth …
- Consumers need not to know the physical location of the service

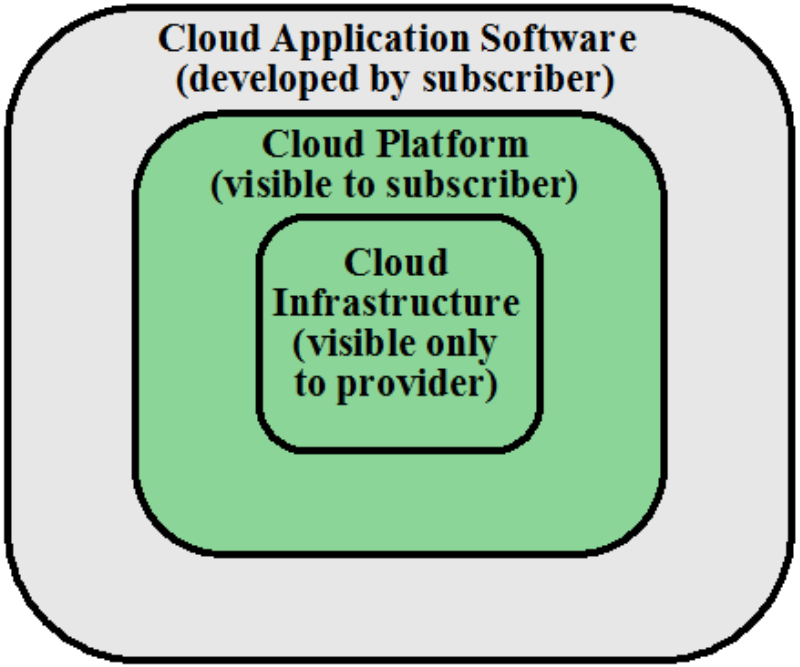
Service model: SaaS

- provides service to customers in the form of application software
- Clients need not to install
- Clients can access application via various platforms thru a simple interface (often a browser)



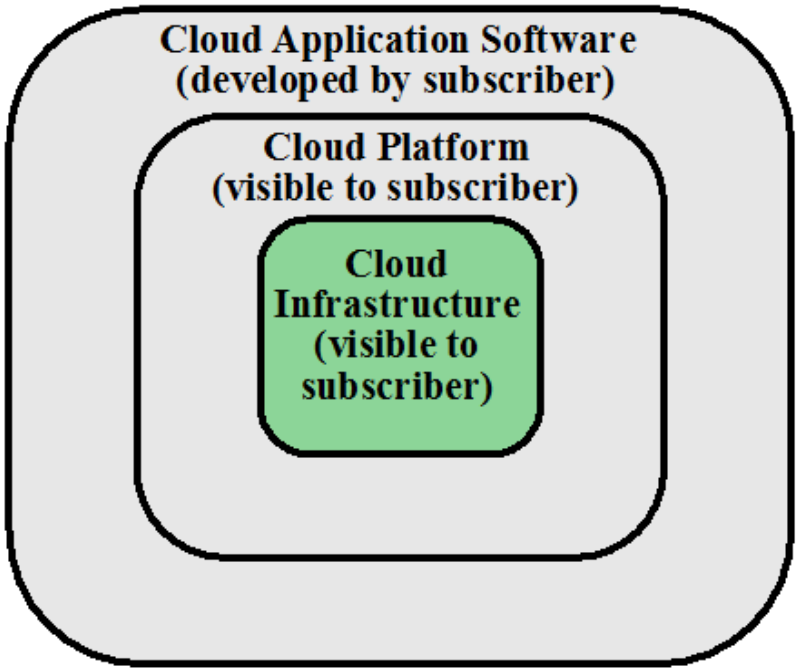
Service model: PaaS

- Provides service to customers in the form of a platform on the which the customer apps can run
- Clients deploy on the cloud
- PaaS provides useful building block/tools



Service model: IaaS

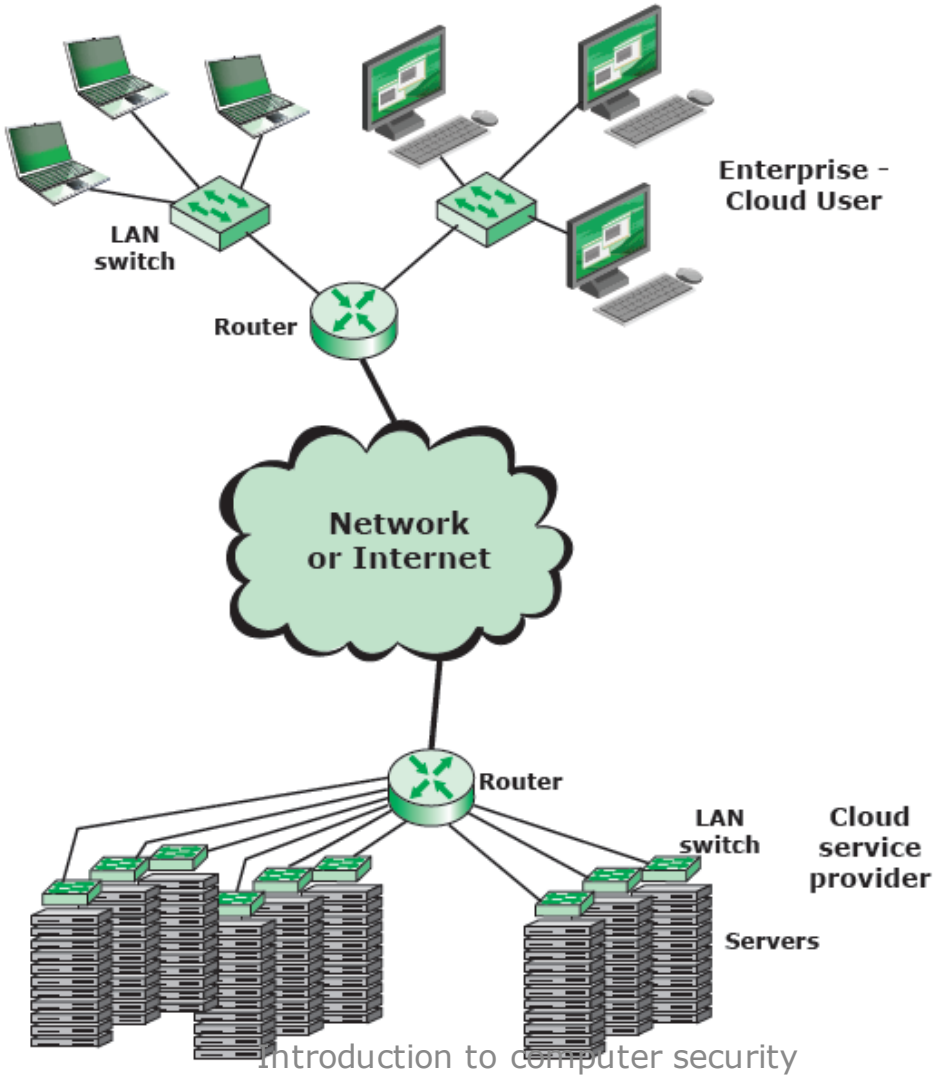
- Provides clients access to the underlying cloud infrastructure
- VM and other abstracted hardware, OS, APIs
- Amazon Elastic Computing (EC2) and Windows Azure



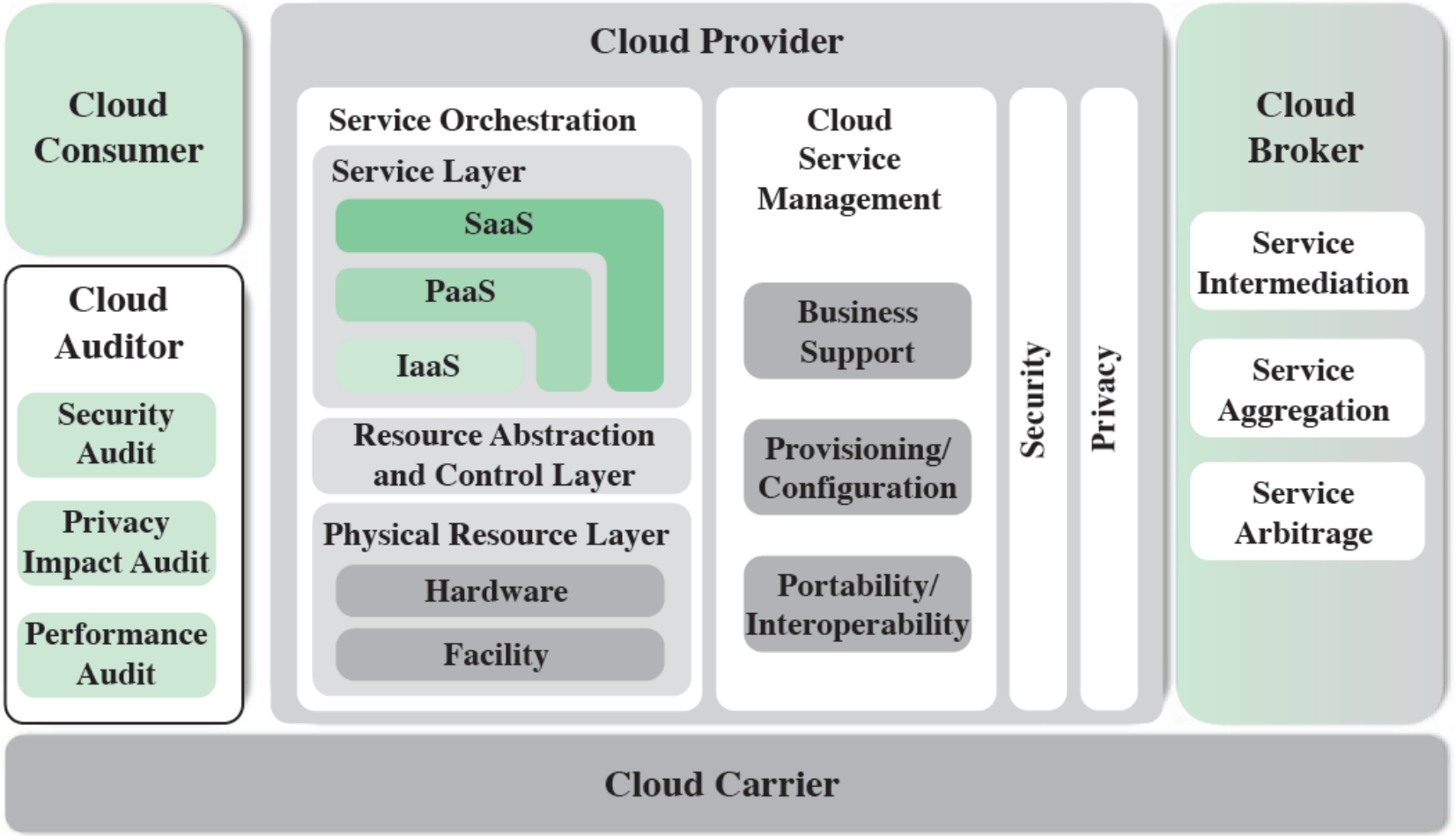
Deployment models

- **Public**: the cloud resources/services are available to the general public
- **Private**: The cloud infrastructure is solely for an organization
- **Community**: the cloud infrastructure is for a specific community and is shared by several organizations
- **Hybrid**: a composition two or more of the above

A typical cloud service context



NIST's reference architecture (conceptual model)



NIST's reference architecture (conceptual model)

- **Cloud customer**: A person or organization that uses or is interested in cloud providers services
- **Cloud provider** (CP): a person or organization that makes cloud services available
- **Cloud auditor**: A party that conducts independent assessment of sources (e.g., security)
- **Cloud broker**: A party that manages use, performance,, negotiations
- **Cloud carrier**: An intermediator that provides connectivity and service transport

Cloud provider (CP)

- For SaaS: CP deploys, configures, maintains, updates app software
- For PaaS: CP manages the computing infrastructure for the platform (middleware, database, OS, programming languages, tools)
- For IaaS: CP acquires physical computing resources: servers, network, storage, …

Cloud security risks

- **Abuse and nefarious use of cloud computing**
  - Relatively easy to register for the services
  - Many cloud services offer free trials
  - Enables hackers to get inside to conduct attacks (spamming, DoS, …)
  - The burden on CP to provide protection
- Countermeasures
  1. Stricter/restrict registration
  2. Enhanced credit card monitoring
  3. Comprehensive monitoring (traffic)

Cloud security risks

- **Insecure interfaces of APIs**
  - CPs expose a set of APIs for customers apps
  - Security depends on APIs: must be secure (from authentication to encryption)
- Countermeasures
  1. Analyzing the security of APIs
  2. Ensuring strong authentication and access control
  3. Understanding the dependency chair between the APIs

Cloud security risks

- **Malicious insiders**
  - Client organization relinquishes many (or all) of its IT and gives to the CP
  - A grave concern: malicious insider activity
- Countermeasures (by client)
  1. Comprehensive supplier assessment
  2. Specify human resource requirements as part of legal contract
  3. Require transparency

Cloud security risks

- **Shared technology issues**
  - IaaS services are delivered in a scalable way by a shared infrastructure (CPU caches, GPUs, …)
  - Probably not designed for multi-tenant architecture
  - Vulnerable to attacks (insider and outsiders)
- Countermeasures
  1. Implement best security practices during implementation, deployment, configuration
  2. Monitor environment for unauthorized activities

Cloud security risks

- **Data loss or leakage**
  - Cloud storage may be the only backup storage
  - Could be devastating for many clients if data is lost
- Countermeasures
  1. Implement strong API access control
  2. Encrypt and protect integrity when in transit
  3. Analyze data protection at design and run time

Cloud security risks

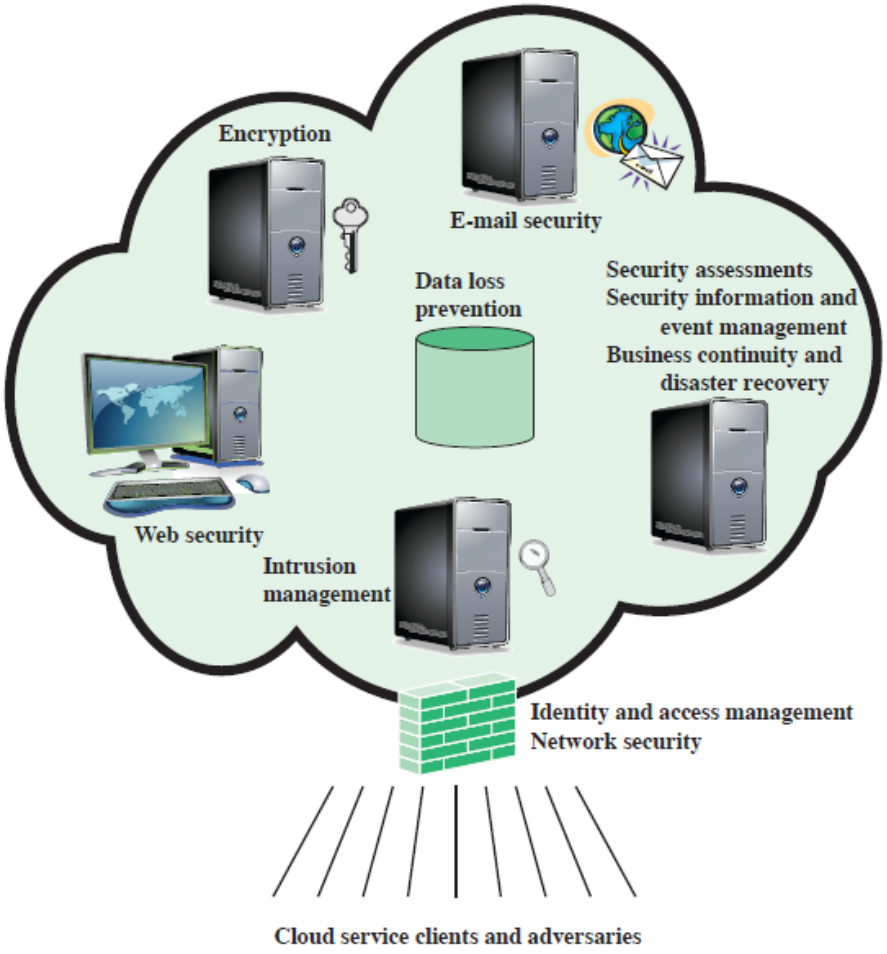
- **Account or service hijacking**
  - Usually with stolen credentials
  - Compromise confidentiality, integrity and availability
- Countermeasures
  1. Prohibit sharing of credentials between users
  2. Leverage strong multi-factory authentication
  3. Employ proactive monitoring

Data protection in the cloud

- Data must be secured while at transit, in use, or at rest
  - Client can employ encryption for transit
  - Client can encrypt data for storage (rest) but can also be CP's responsibility (key management)

Cloud security as a service

- Identify management
- Data loss prevention
- Web security
- E-mail security
- Encryption
- Business continuity
- Network security
- …



ĐẠI HỌC TÔN ĐỨC THẮNG
TON DUC THANG UNIVERSITY

Summary

- Introduced databases and DBMS
- Relational databases
- Database access control issues
  - SQL, role-based
- Inference
- Statistical database security issues
- Database encryption
- Cloud security