CSCI 455: Principles of Computer Security

Set 04:
Data and Database Security, Part I.

Slide Sources

- Multiple sources including
  - Stallings & Brown textbook materials
    - Chapter 5
  - My slides from my CSCI 622 class
  - Other readings including NIST publications

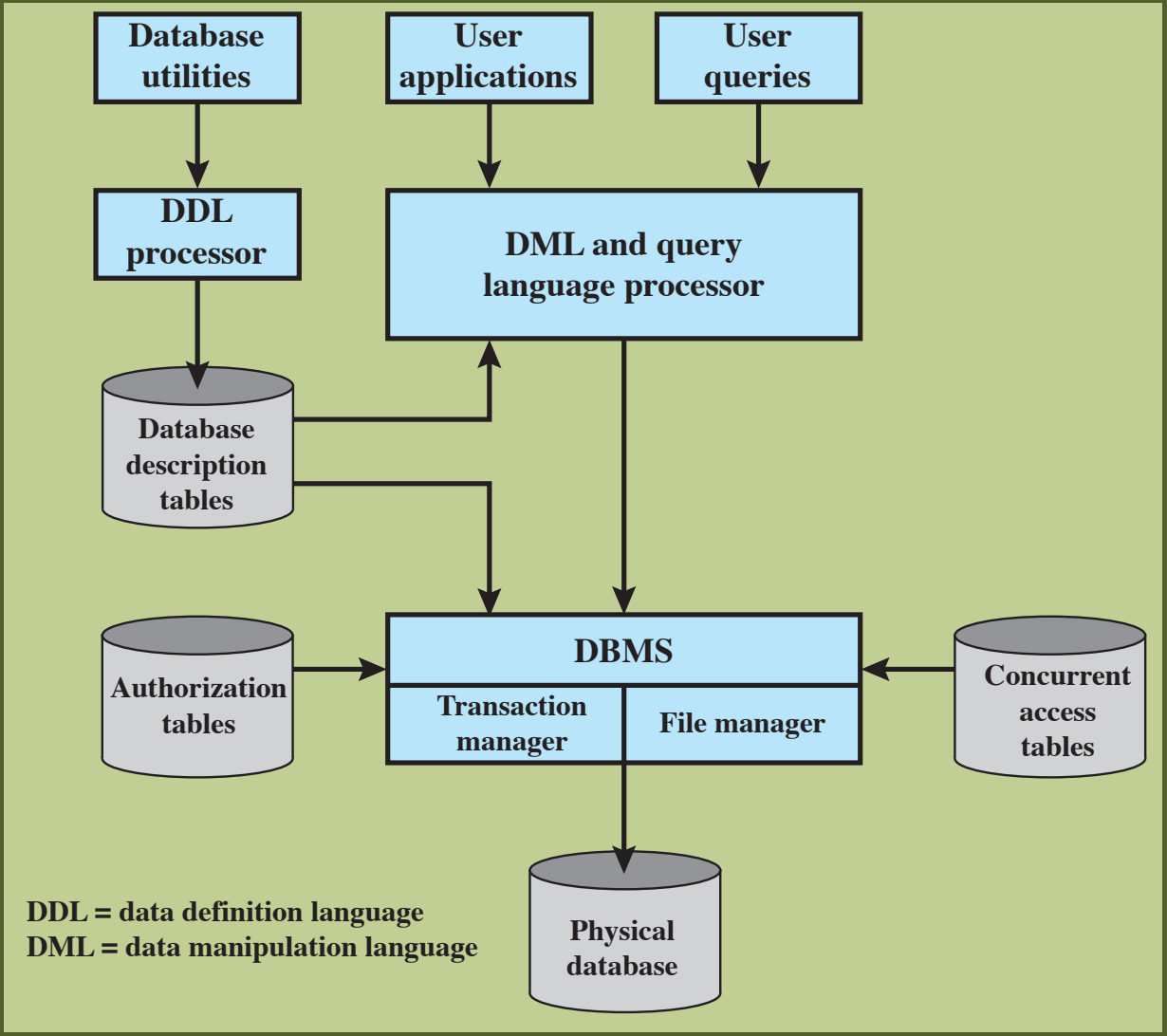
A Little Background on Database Systems

Databases and DBMS

- Structured collection of data for use by one or more applications
- Contains relationships between data items and groups of data items
- Can contain sensitive data to be secured
  - Query language
  - Provides a uniform interface to database
- Database management system (DBMS)
  - Suite of programs for constructing and maintaining the database
  - Offers ad hoc query facilities to multiple users and applications

From Stallings & Brown

DBMS Architecture



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 each row has a one column where all values are unique, forming a key for that row
- Enables creation of multiple tables linked together by a unique identifier that is present in all tables
- Relational query language to access database
  - Allows user to request data to fit a given criteria

Relational Database Elements

- Four levels (terminology)
  - Database
  - Relation/table/file
  - Tuple/row/record
  - Attribute/column/ field
- Primary key
  - Uniquely identifies a row
  - Consists of one or more column names
- Foreign key
  - Links one table to attributes in another
- View/virtual table
  - Result of query that returns selected rows and columns from one or more tables

Dog World—Relational Database

Dog

tag	name	bones	mom	age
12	Fido	4	45	4
45	Fifi	5	13	11
56	Scamp	565	82	1

DogHouse

address	color	year
13 Elm	red	1998
12 Oak	green	2002
42 Ash	white	2001

LivesIn

tag	address
12	12 Oak
45	12 Oak
23	42 Ash

Flea

id	name	dog
234	Jo	12
451	Bo	45
231	So	45

Structured Query Language (SQL)

- Standard)s
- Data Manipulation Languages (DML)
  - CRUD operations
    - Create data
      - Insert
    - Retrieve data
      - Select
    - Update data
      - Update
    - Delete data
      - Delete
- Data Definition Language (DML)
  - Create database structure
    - Tables, views,
    - Indexes, unique keys, …
    - Procedures, triggers, …
    - …
  - Modify structure
  - Drop structure

Data Manipulation Language (DML)

- Query languages
  - For accessing and manipulating data
  - SQL, the most widely used query language
- Two languages types
  - Procedural
    - User specifies what data is required and how to get those data
  - Nonprocedural
    - User specifies what data is required but not how to get the data

- Manipulation is CRUD
  - Create

```
insert into dog
(tag,name,age)
values (12, 'Lady', 5)
```

  - Retrieve

```
select name from dog
where age > 3
```

  - Update

```
update dog set age =
age + 1 where tag = 12
```

  - Delete (don't delete!)

```
delete from dog
where age > 1
```

Data Definition Language (DDL)

- Notation to specify database schema
  - Includes storage space, usage, indexes, keys
- DDL compiler
  - Generates tables stored in data dictionary
- Data dictionary
  - Contains metadata, i.e., data about data

```
CREATE TABLE dog (
        tag   integer,
        name  char(10),
        age   integer,
        mom   integer
)
```

```
ALTER TABLE dog
ADD dad integer
```

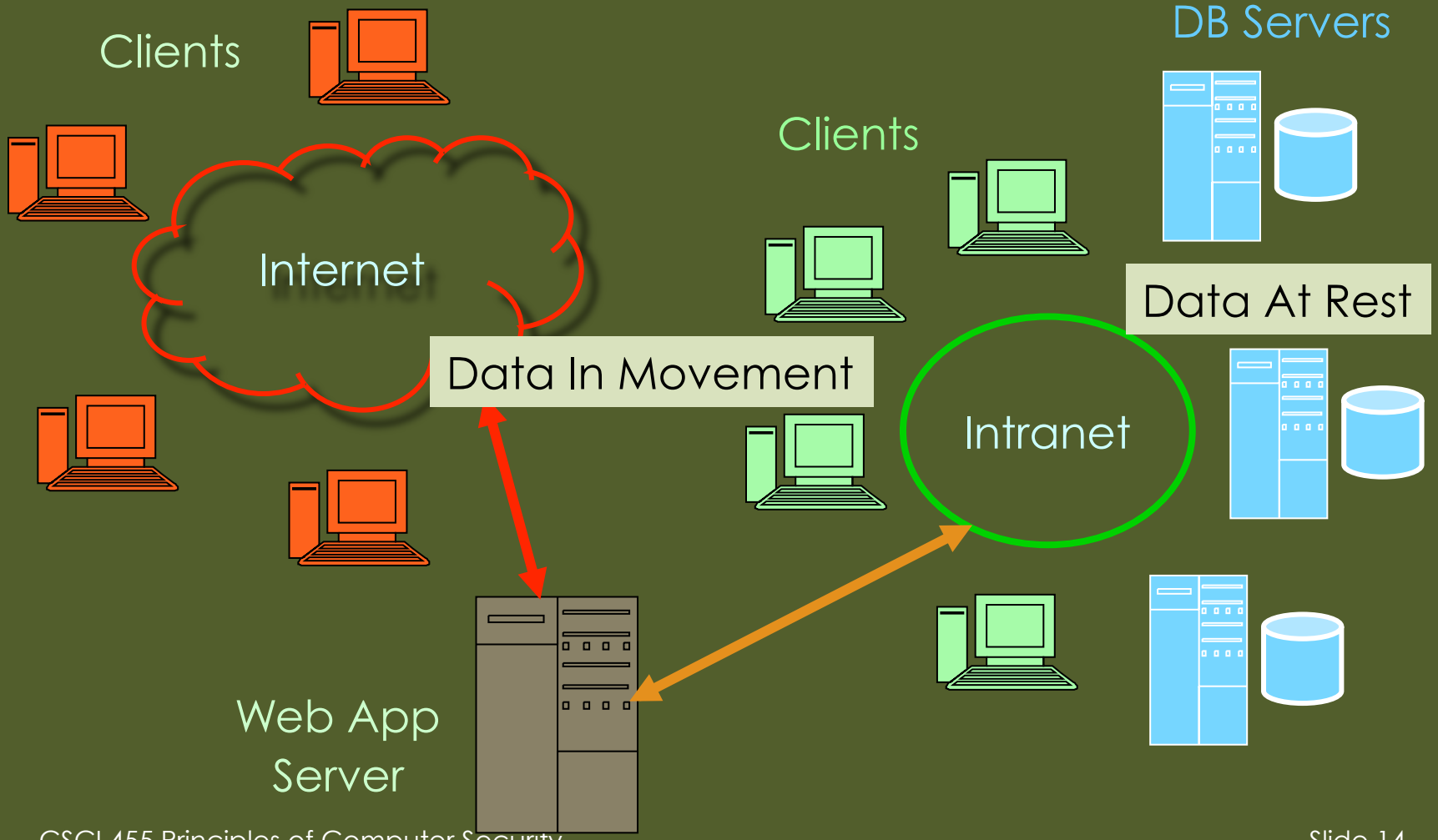
```
DROP TABLE dog
```

Data and Database Security.
Data Privacy

Not Merely Securing the Database

- Secure data
- Secure database
- Secure DBMS
- Secure applications
- Secure operating system
- Secure web servers
- Secure network environments
- Data privacy

Secure Software Systems: The Big Picture



Secure Databases

- Traditional database security topics and issues
  - Users, passwords
    - Default users/passwords
  - Privileges
    - Actions on data
  - Roles
    - Collections of system privileges
  - Grant/revoke
    - Giving privileges or roles to users
    - Removing privileges or roles from users
  - What else?

Secure DBMS

- Possible DBMS holes
  - For instance, Oracle security alerts
    - http://www.oracle.com/technetwork/topics/security/alerts-086861.html
  - Buffer overflow problems in DBMS code
  - Miscellaneous attacks
    - Denial of service, source code disclosure, …
- Continual DBMS patching needed
  - SQL Slammer Worm & Microsoft SQL Server

Secure Application Development

- Access DBMS via applications: SQL injection attacks
  - Keeping names/passwords in databases
  - Application code contains SQL statement
    - "SELECT * FROM users WHERE name = `" + username + "' AND password = '" + password + "'"
  - On success, one row returned, none on failure
  - Attacker changes query to
    - SELECT * FROM users WHERE name = 'anyname' AND password = 'anypassword' OR true
  - Discuss why this works or doesn't!
- Application security in enterprise environment such as J2EE, .NET, and SOA

Secure Operating System

- Interaction of DBMS with OS
  - Secure administrative accounts
  - Account policies
  - System versus user mode
  - Some Oracle files are SUID (root)
  - Other

Secure Web Server

- Interaction of DBMS and Web server
- Web server issues
  - Standard configuration has some potential problems
  - Ensure secure communication from web clients to web server
  - Limiting possible connections
  - Others…

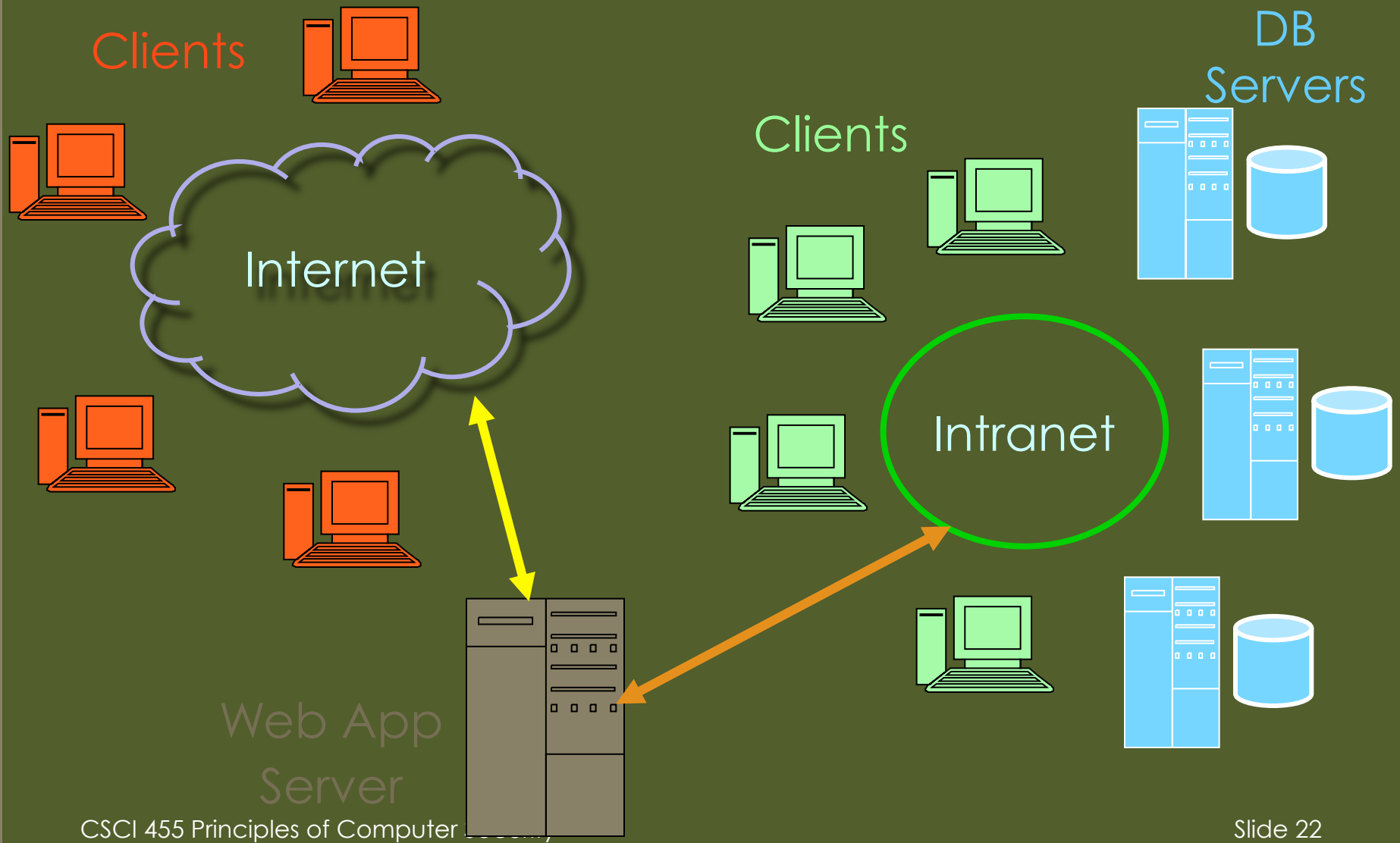
Secure Network

- Interaction of DBMS and Network
  - Authentication, integrity, and encryption
  - Oracle server generally behind firewall
    - Separating DBMS and web servers
  - Other network issues
    - Sniffing and spoofing and …

Miscellaneous Issues

- Auditing policy
  - Developing comprehensive audit system for database activity tracking
- See Ben Natan for other topics
- See Litchfield for specific Oracle issues

Security: The Big Picture



Data Privacy

- Overlaps with security, especially confidentiality
- Dramatic increase in scale of information collected and stored
  - Motivated by law enforcement, national security, economic incentives
- Individuals have become increasingly aware of access and use of personal information and private details about their lives
- Concerns about extent of privacy compromise have led to a variety of legal and technical approaches to reinforcing privacy rights

Security Specification within SQL Databases

This is a very small part of the overall data security space!

SQL Grant Statement

- grant statement is used to confer authorization

  grant <privilege list>

  on <relation or view name> to <user list>

  - <user list> is …
    - a user-id
    - public, which means all valid users
    - a role (more on this later)
- Granting a privilege on a view does not imply granting any privileges on underlying relations
- Grantor of a privilege must already hold privilege on specified item (or be the DBA)

Privileges in SQL

- select, insert, update, delete

  grant select on branch to ‘abc1234’, ‘zyx9876’
- References
  - Ability to declare foreign keys
- All
  - Ability to give all privileges
- with grant option
  - Allow user to pass on privilege
  - grant select on branch to U1 with grant option

Roles

- Specify common privileges once (role)
  - Assign roles to a class of users
  - Roles assigned to users or even to other roles
  - Supported from SQL:1999
- Example

  create role teller

  create role manager

  grant select on branch to teller
  grant update (balance) on account to teller
  grant all privileges on account to manager
  grant teller to manager
  grant teller to tigger, pooh, kanga
  grant manager to Christopher

Revoking Authorization in SQL

- The revoke statement

  revoke<privilege list> on <relation or view name> from <user list> [restrict | cascade]

  - revoke select on branch from U1, U2 cascade
- Revoking user privilege may cascade to others
  - Cascading revoke
  - This can be restricted if appropriate
- Revoking is intricate and interesting!
  - Explicit versus implicit privileges
  - Singly and doubly granted privileges
  - Timing issues

Limitations of SQL Authorization

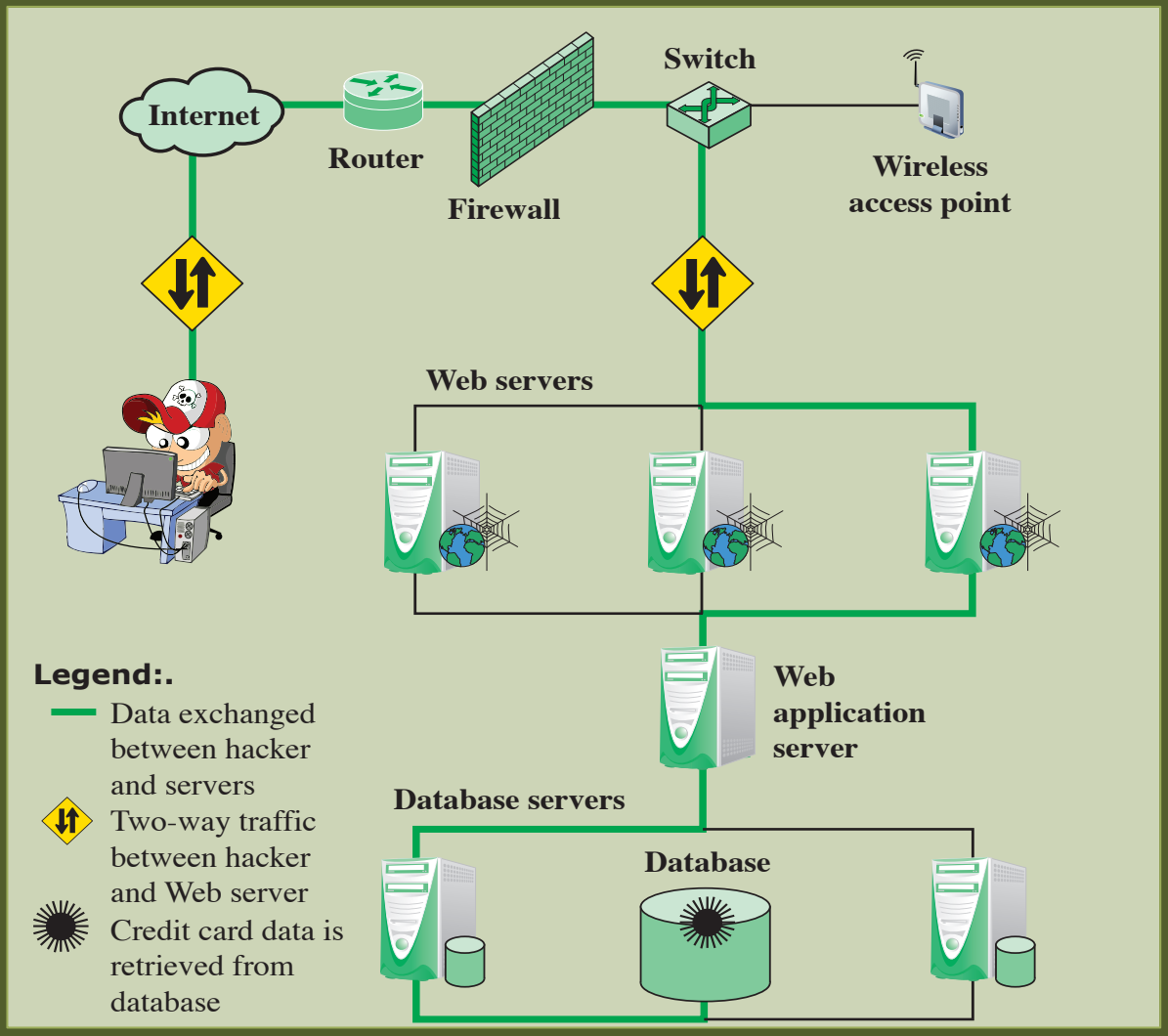
- SQL does not support tuple level authorization
  - Cannot restrict students to see only (the tuples storing) their own grades
- All application (e.g., a web application) end-users may be mapped to a single DBMS user
- Above tasks of authorization falls on the application program, with no support from SQL
  - Authorization must be done in application, and is dispersed all over an application code
  - Checking for absence of authorization loopholes is difficult because it requires reading large amounts of application code

Some Attacks on Database Systems

SQL Injection Attacks (SQLi)

- Common network-based security threat
- Exploits basic web application design
- Sends hacked SQL to database server
- Most common attack goal is bulk data extraction
- Depending on SQL environment, injection can also be exploited
  - Modify or delete data
  - Execute arbitrary operating system commands
  - Launch denial-of-service (DoS) attacks

SQL Injection Attack: Anatomy



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 Attack Avenues

- User input attack
  - Inject SQL commands by crafting suitable user input
- Server variables' attack
  - Forge HTTP values and network headers to exploit vulnerability by placing data directly into the headers
- Second-order injection attack
  - Malicious user relies on existing system data to trigger SQL injection attack, so when attack occurs, the input is modified from within system itself
- Cookies' attack
  - Application server modifies SQL query using cookie's content
- Physical user input attack
  - Apply user input to construct an attack outside realm of web requests

Inband Attacks

- Uses same communication channel for injecting SQL code and retrieving results
  - Retrieved data presented directly in application Web page
- Tautology attacks
  - 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 nullified through usage of end of line comments
- Piggybacked queries
  - Attacker adds additional queries beyond the intended query, piggy-backing attack on top of a legitimate request

Inferential Attack

- No actual transfer of data, but attacker can reconstruct information by sending particular requests and observing the resulting behavior of website/database server
- Illegal/logically incorrect queries
  - This attack lets an attacker gather important information about the type and structure of the backend database of a Web application
  - The attack is considered a preliminary, information-gathering step for other attacks
- Blind SQL injection
  - Allows attackers to infer the data present in a database system even when the system is sufficiently secure to not display any erroneous information back to attacker

Out-of-Band Attack

- Data are retrieved using a different channel
- Can be used when there are limitations on information retrieval, but outbound connectivity from the database server is lax

SQLi Countermeasures

- Defensive coding
  - Manual defensive coding practices
  - Parameterized query insertion
  - SQL DOM
- Detection
  - Signature based
  - Anomaly based
  - Code analysis
- Run-time prevention
  - Check queries at runtime to see if they conform to a model of expected queries

Database Encryption

- Database typically the most valuable information resource
- Protected by multiple layers of security
- Firewalls, authentication, general access control systems, DB access control systems, database encryption
- Encryption becomes last line of database security defense
  - Can be applied to the entire database, at the record level, the attribute level, or level of the individual field
- Disadvantages to encryption:
  - Key management
- Authorized users must have access to the decryption key for the data for which they have access
  - Inflexibility
- When part or all of the database is encrypted it becomes more difficult to perform record searching

Secure Data Miscellany

Methods for Inference Handling

- Access restriction
- Query set restriction
- Microaggregation
- Data perturbation
- Output perturbation
- Random sampling

and

- Auditing

Security versus Protection

- Security policy
  - What is needed
- Protection mechanism
  - Hardware/firmware/software that implement security policies
- Good idea to separate policy and mechanism
  - Good
    - Access control matrices used in DAC
  - Not so good
    - BLP, which intertwines protection & security

Security versus Trust

- Trusted system
  - One that meets intended security requirements, is of high enough quality, and justifies user confidence
  - Trust perceived by user, not developer or designer, even if one is not able to evaluate trust directly

Secure	Trusted
Either-or: Either is or is not secure	Graded: Degrees of trustworthiness
Property of presenter	Property of receiver or user
Asserted based on product characteristics	Judged based on evidence and analysis
Absolute: not qualified as to how used, where, when, or by whom	Relative: viewed in context of use
A goal	A characteristic

More Data and Database Security Topics

- Auditing
- MLS Databases
- App design with database systems
- Intrusion detection
- Cloud data management