

# CSE 498U/598U – Computer Security

Class 31: Friday, November 12

---

## Reminders

- Homework
  - Homework 4 Nov 19 @ 5 PM
  - Project Status Report Nov 24 @ 5 PM
  - Project Reviews Nov 24 @ 5 PM
- Reading / Web Browsing
  - Google: Database security

---

## Schedule

Friday  
    Help – projects  
    Database security  
Monday  
    Database security

---

## Lecture Outline

Example code – sniffing + spoofing  
Database overview

---

## Key Points

- What is the difference between a schema and a query?
- What is a join query?
- What is inference?

## Lecture Notes – CSE 498U/598U - Computer Security

Lecturer: Dr. Aaron Striegel [striegel@nd.edu](mailto:striegel@nd.edu)

Topic: Sniffing / Spoofing  
Databases

### Sniffing

libpcap library

Example: tcpdump, Ethereal

Promiscuous mode

Man on the side vs. man in the middle

### Spoofing

Raw socket

UDP vs. TCP

Review of TCP sequence numbers

Where to start?

### Databases

Poll: How many have used a database?

Database Data + Rules (hierarchy/relationship)

DB Admin

DBMS Database Management System

Front end for the DB

Examples of DBs:

Access, Oracle, MySQL, etc.

Record Related group of data

Field Individual elements of data within a record

Schema Logical structure of the database

Subschema Restriction to only parts of the DB

Attribute Name of a column

Relation Set of columns

Query Retrieve information from a database that satisfies the request

SQL: SELECT NAME FROM CSE498U WHERE ZIP=46530

Logical operators

AND, OR, etc.

Join Query Using a common shared element, join two listings together

Example: Table of people with zipcodes

## Table of airports for individual zip codes

Show me the airport for each person

### DB vs. File System

Shared Access	Common, centralized set of data
Minimal redundancy	Do not have to collect own data
Data consistency	Change affects all users of the data value
Data integrity	Protect against accidental or malicious changes
Controlled access	Only authorized users can see files

Caveat: Security is now harder

### Security Requirements – DB

Physical DB integrity	Immune to physical issues (power, etc.)
Logical DB integrity	Structure of DB must be preserved
Element integrity	Each element is accurate
Auditability	Track who has modified the data
Access control	User-level read/write control
User authentication	Must ID each user
Availability	Can use the DB

### Example Scenarios

Citibank	Millions of credit card transactions
Voting Precinct	Thousands of votes
On-line forum	Hundreds of comments

### Physical issues

System crash in the middle of a transaction  
Disk goes bad, processor locks up

### Logical Integrity

Integrity of the user  
Access control      Biba-style control  
Verify the DB  
Field checks  
Change log

### Auditability

Use it to recover from integrity violations  
Can use it to monitor knowledge passed to the user

### Inference

Obtain data values from others  
Example:      Sherlock Holmes for DBs

Two-phase update

Intent phase

Gather the appropriate information

Commit phase

Write, mark a commit flag

Similar to the threading/coherency issue

Make the write atomic

Example:

Paper clips in the office