Date ____/____/____

MON TUE WED THU FRI SAT SUN

SANDEN
SANPAK ENGINEERING INDUSTRIES (PVT) LTD

Replay: use previously captured user response
use challange-response protocol

eavesdropping: gain password by observing user / read written password / keystroke logging.
use multiple factors
Static biometrics, use device auth.

continued ....

Trojan Horse: device prevends to be auth device to gain password

Denial of service: disable user auth by flooding host with requests
can be done to a selective user by exceeding log-in limit.
use multiple auth factors.

=> Database Security:

5.1: Need for database security

* Organizations need to provide access to databases but also prevent it from internal & external security threats.

* SQL is more complex than HTTPS requires full understanding of SQL to have effective security measures

Database security not up to pace

new features in DBMS has introduced new vulnerabilities & potential for misuse.

Database staff has limited knowledge of security & vice versa

In an enterprise, mixture of database platforms are used hence implementation of security is difficult.

Date ___/___/___

MON TUE WED THU FRI SAT SUN

SANDEN
SANPAK ENGINEERING INDUSTRIES (PVT) LTD.

5.2: Database Management Systems

* Database is a structured collection of data to be used by one or more applications.
* DBMS a suit of program to construct & maintain the database.
* Query language provides a uniform interface to database for users & applicants.
* DDL used to define database logical structure & procedural properties.
* DML provides a powerful set of tools for developers.
* DBMS uses authorization table to check if user has permission to run query on database.
* concurrent access table prevents conflicts when simultaneous conflicting commands are executed.
* Database's complexity & criticality generates security requirements beyond OS-based security mechanisms.
* OS only controls read/write access control whereas DBMS provides more operations like select/insert/update/delete. Hence requires seperate security mechanism.

5.3: Relational Databases

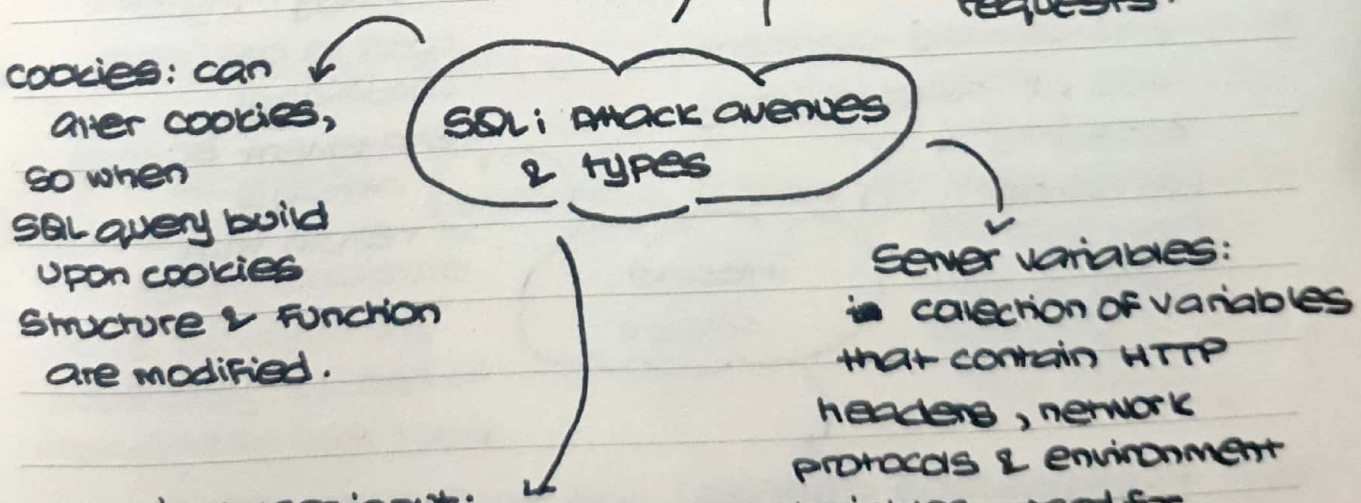
* A two dimensional file (rows & columns) is called a flat file.

* Rows refered as tuples.
* Columns refered as attributes.
* Primary key to uniquly identify each column.
* Foreign key - primary key of another table, used to create relations.
* View is a virtual table, formed on the basis of result of a query. Used for security purposes to restrict access so user can only view certain rows/columns.
* SQL used to define schema / manipulate / query data.

5.4: SQL injection attacks

* SQL injection attack is one of the most prevelant & dangerous attack.
* Is designed to exploit nature of web application pages. When dynamic data is send/retrived from database server through API calls. Malicious SQL commands can be send.
* SQL injection can be used to extract bulk of data, populate data base with numerous records, execute arbitary OS commands, launch DOS or delete data.
* SQLi can take place when input not correctly filtered for string literal escape characters / not strongly typed.

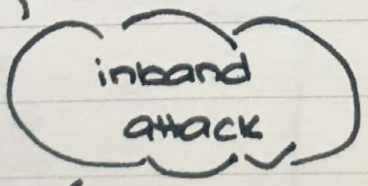
* Typical approach is to identify vulnerability & send malicious SQL query, hide it underneath traffic to be executed on database server. This will return desired results.
* SQLi works by prematurely terminating a string & appending a new command, and use -- to make it seem like a comment.
* Second-order injection: occur due to incomplete prevention mechanisms for SQLi can rely on data already present to trigger an attack.



* There are 3 main categories of attack types:
  - inband :- uses same communication channel for SQLi & retrieve results to be displayed on same webpage.
  - inferential :- no transfer of data. Attack sends particular requests to observe behaviour of database server & reconstruct information.
  - out of bound :- data retrieved using different channels. Used when limitations on data retrieval byt outbound connectivity of database server is relaxed.

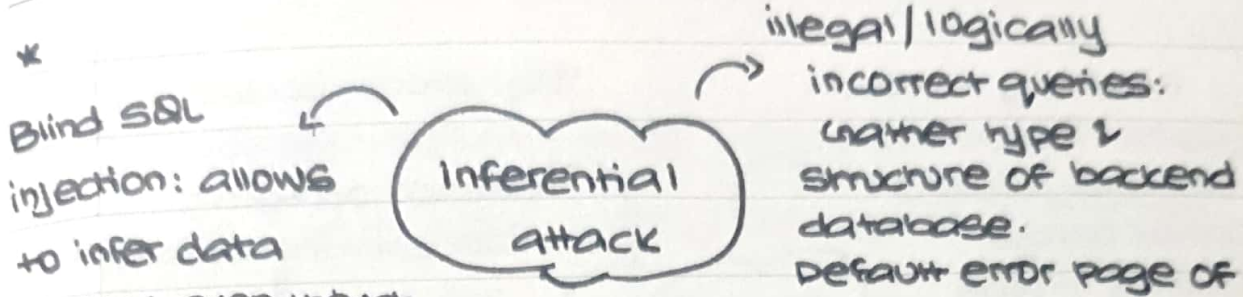
* end-of-line-comment: add -- after injecting code into a particular field to nullify it

tautology: inject code in one/more conditional statement so they always return true



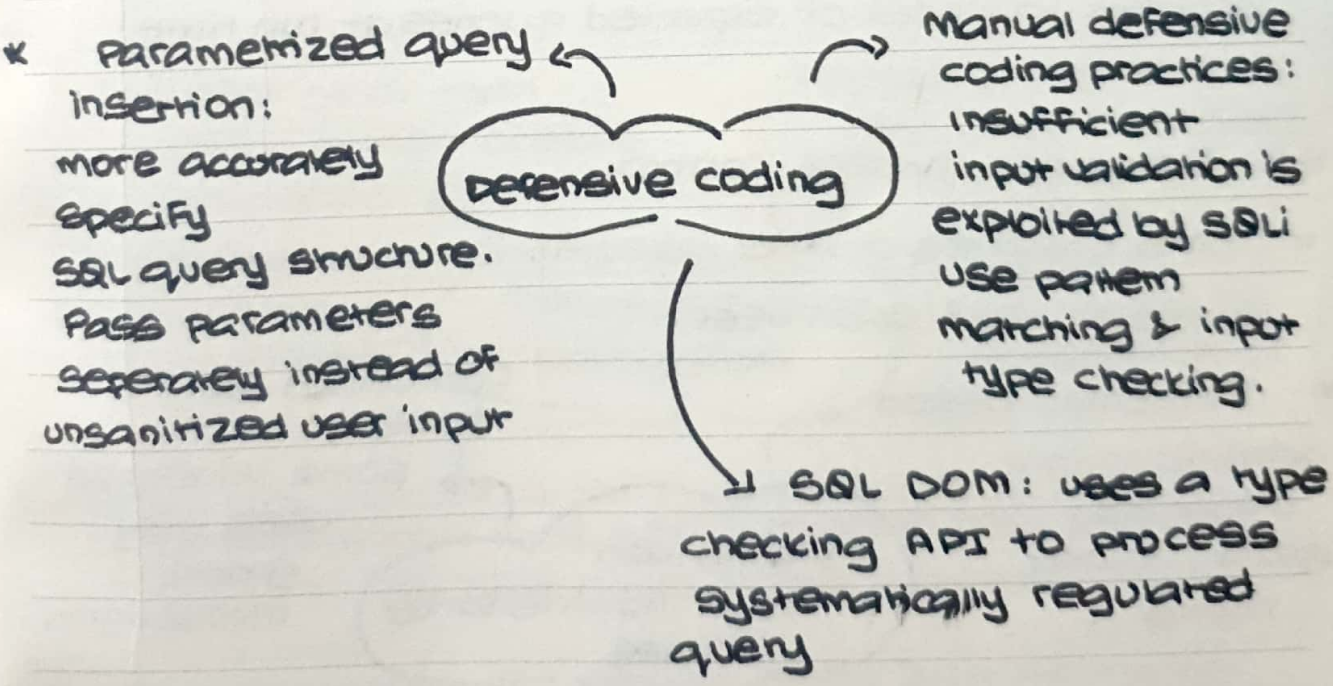
Piggybacked queries: add more query instead of the intended query. Relies on server configuration allowing different queries in a single stringe of code.

*

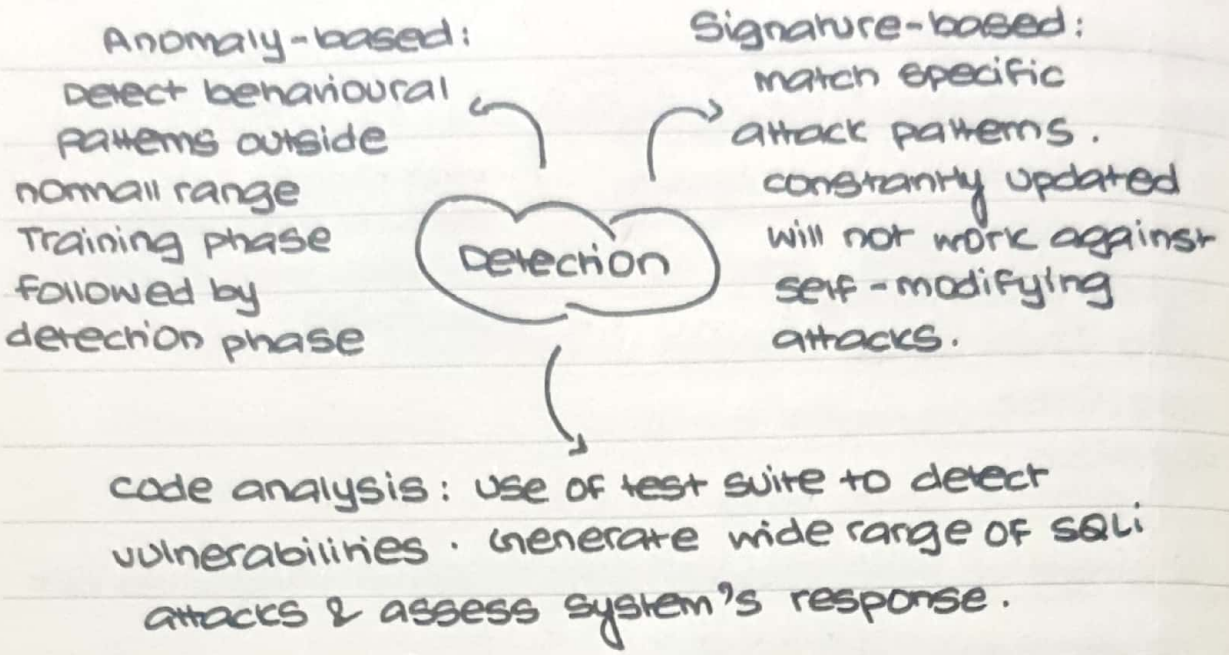


* A single countermeasure is insufficient. Integrated set of techniques are used.

* Countermeasures have 3 categories. Defensive coding, detection & run-time prevention.



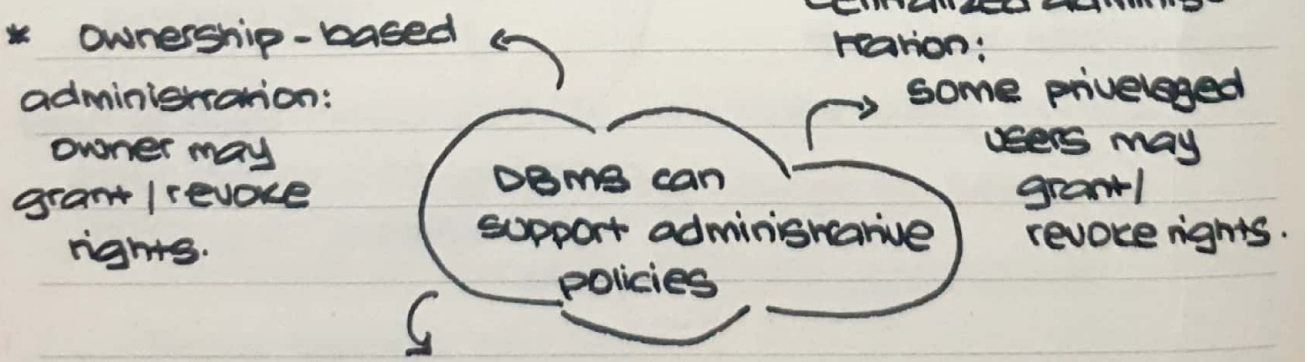
*



* Run-time prevention: techniques to check queries conform to model of expected queries at run time.

5.5: Database Access Control

* DBMS operates on the assumption computer has authenticated each user.

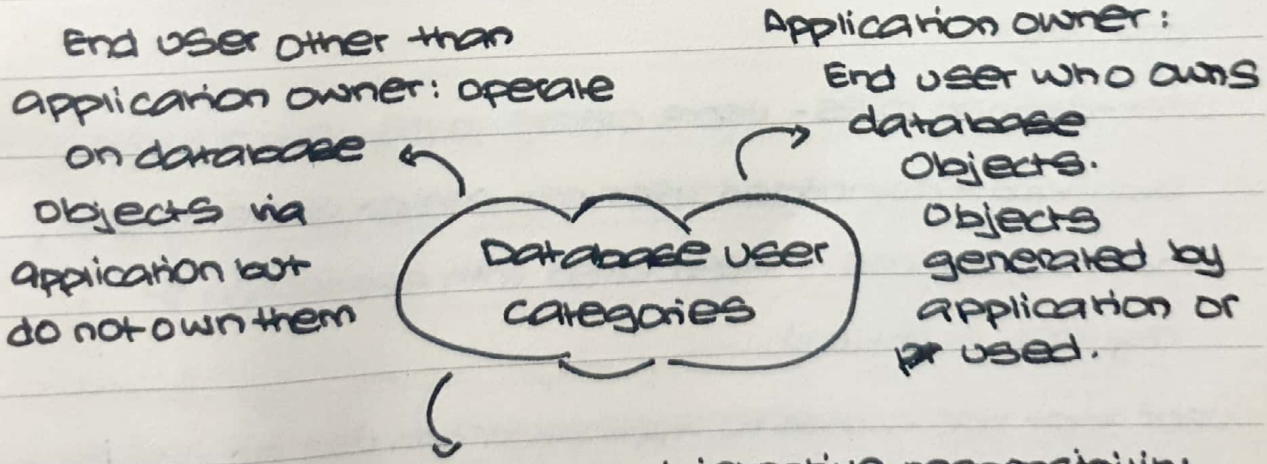


Date ..... / ..... / .....

MON TUE WED THU FRI SAT SUN

SANDEN
SANPAK ENGINEERING INDUSTRIES (PVT) LTD.

* Access rights can be entire table / databases / selected rows / columns.
* SQL provides two commands GRANT / REVOKE. can be use to give access rights / assign a role to an user.
* Ranges of access rights include: Select, Insert, Update, Delete, Reference - define foreign keys in another table.
* If an user revokes the access rights all cascaded access ~~rig~~ granted are also revoked.
* Database systems support dozens of applications unlike a file system.. Hence we use RBAC.
*



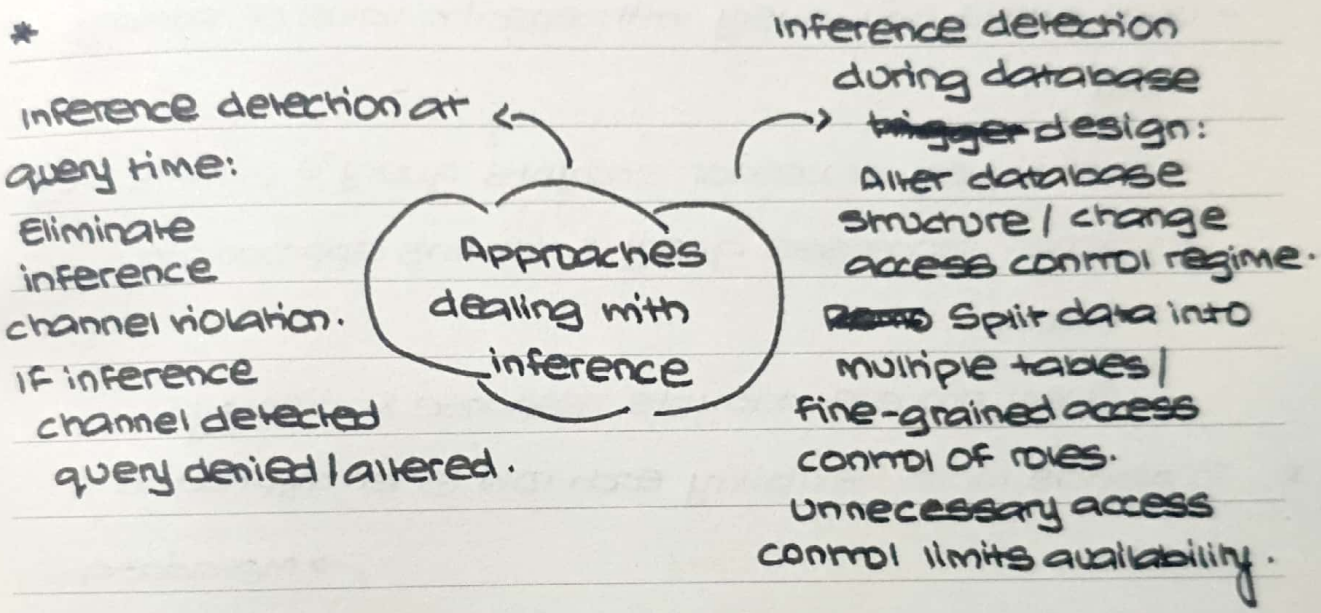
* Administrators can assign users administrative based roles.

* RBAC needs to provide:
  - Create & delete roles.
  - Define permissions for a role.
  - Assign & cancel assignment of users to roles.
* Administrator cannot add/delete/modify fixed roles. They can only add/remove users as members to fixed roles.
* Fixed server roles - defined at server level are independent of any user. Have different permissions and intend to distribute administrative responsibilities.
* Fixed database roles - at the level of individual database. Some roles designed to help DBA to distribute administrative privileges while others for an end user.
* User-defined roles - users create roles. Are 2 types Standard:- authorized user can assign users to roles, and Applications:- associated with application & requires password.
* User that has access to application can use application role.

5.6: Inference

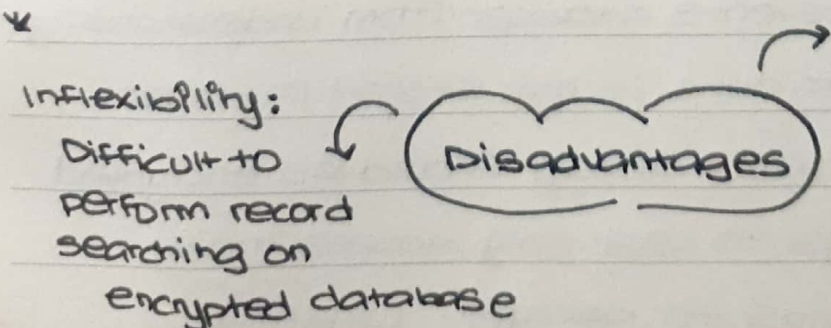
* Performing authorized queries & deducing unauthorized information from legitimate responses.

* Inference problem happens when set of individual items is more sensitive / can be used to infer sensitive data.
* Metadata is correlation / dependencies between data items.
* Information transfer path of unauthorized data is called inference channel.
*



5.7: Database encryption

* Encryption is last line of defense in database security.
*



* Encryption can be applied at record level, attribute level (column) / individual field.
* DBMS requires skilled personnel to maintain, update, protect from disaster & secure database.
* To user encrypted database: (Not flexible method).
  - user sends SQL query with specific value of primary key.
  - client query processor encrypts query's primary key.
  - server processes query & returns appropriate record.
  - Query process decrypts response & displays.
* To provide more flexibility each row is encrypted as a block. → metadata.
* Mapping function is stored at clients & data owner's.
* Each data is divided in a range and each range is assigned an attribute.
* Indexing scheme prevents attacker from understanding data because metadata is not stored on server.
* Different chunks of database should be encrypted with different keys so user only access those which they have keys to decrypt. (RBAC).