DATABASE SECURITY

- security requirements
- Realiability / Integrity
- sensitive database.
- Inference
- Multilevel database.

college database:-

NAME	BRANCH	CLASS	PLACE	FEES	LIBRARY
A	COMP	2	MUMBAI		
B	COMP	3	MUMBAI		
C	IT	3	DELHI		
D	COMP	3	PUNE		
E	IT	4	MUMBAI		
F	COMP	4	-		

There is a file with NAME, BRANCH, CLASS for every department say for library all these 3 + library field → a seperate file

∴ redundancy is there.

Database solves the redundancy problem

Consistency problem → with file system.

If any field changes it is to be changed in every file with which contains the data related to that field.

ADVANTAGES OF DATABASES OVER FILE SYSTEM.

1] DBMS solves the problem of redundancy.
2] DBMS - consistency is maintained.
3] Access control / Authentication : Access to authorized users.
4] shared access : every department gets view of DBMS.

SELECT QUERY :-

select Place = 'MUMBAI'

PROJECT QUERY :-

show Name where (place = "MUMBAI") ^ (CLASS = "2")

www.ErForum.Net

SECURITY REQUIREMENTS :-

- PHysical Integrity
- Logical Integrity
- Element Integrity
- Auditability
- Access Control
- Availability
- User PassWord (AuthenHication)

INTEGRITY :-

No modification, add, delete.

CONFIDENTIALITY :-

Unauthorized user should not get access in any case

AVAILABILITY :-

Authorized user should always get an access. that is he should not be denied to access in any case.

PHYSICAL INTEGRITY :-

Data is lost due to some pysical phenomenon eg :- ~~dete~~ deletion of data. power failure, Someone can reconstruct DB if it is destroyed through a catastrophe.

LOGICAL INTEGRITY :-

modification to the value of one field do not affect other field

No physical harm. only logical structure changes / damaged.

eg:- Branch is no longer logically interconnect with other entries.

ELEMENT INTEGRITY:- Data contained in each element are accurate.

Granularity → element value gets damaged

CONTROL FOR ELEMENT INTEGRITY:-

1] FIELD CHECKS:-
Elementry integrity control look for data is numeric / alphabetical.
check whether element is numeric / alphabeti
hook for range of element specified.

2] ACCESS LOG:-
maintains all activities performed on database.
∴ we can backtrack.

3] ~~con~~ CONTROL:-
Element wise assign some attribute
e.g. along with name some security value is assigned & one satisfying this security value has access to name attribute.

4 ~~AVAILABILITY~~:- AUDITABILITY:- ~~It~~ is possible to track who or what has accessed the elements in database.
log which maintains the information of updation, deletion of database.

ACCESS CONTROL:- A user is allowed to access only authorised data & different users can be restricted to different mode of access. (such as read/write)
During write updation not available.
security level.

AVAILABILITY :- User can access the database in general and all the data for which data is available ~~Write or updation not~~ During they are authorized.

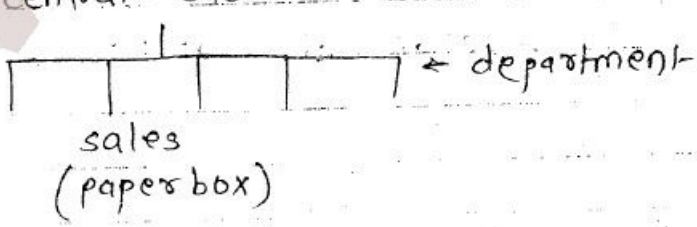
USER PASSWORD :- (Authentication).
Every user is truly identified both for the audit trail & for used during only log in process.
Permission access certain data.

eg :- difference bet^n user password & access control.
There are two different users i.e principle & clerk. both need password for log in. Principle has more access control than clerk.

RELIABILITY | INTEGRITY :- Can be viewed from 3 Dimension
① DB Integrity ② Element integrity
③ Element accuracy.

TWO PHASE UPDATE (intent / commit)
(F) / Make permanent changes.

XYZ organisation → has some central stores



If some request for paper box
- Quantity of paper box 100
- Annual Budget 10,000
- Cost of paper box. 5,000

- Reorder Flags = 0

gather data, create dummy records, open files, lock the users, calculate final answer, gather the resources - update

INTENT PHASE :- No modification, No update collect info in term of in shadow reg.

If TRequest = 60
TQuantity = Quantity - TRequest
= 100 - 60
= 40
TBudget = Budget - cost
= 10,000 - 5,000
= 5,000
TReorder = 1

All these values are writter in shadow register.

If failure occurs during this phase repeate n no of times.

COMMIT PHASE :-

In commit phase, commit flag is set to 1 therefore
No going back to intent phase.
TQuantity = Quantity
TBudget = Budget
TReorder = Reorder

No updation takes place. final values are stored in database.

If failure occurs during this phase repeat any no of times as values are present in shadow register.

REDUNDANCY :- (Internal Consistency)

Parity code → even parity, odd parity, Huffman.

Monitors :- some kind of s/w or prog. that will be coming in front end for reliability checking in terms of range, state, transition.

- ✓ Range monitor ⇒ (Boundary / limit)
- ✓ state monitor ⇒ (president ⇒ organ)
- ✓ transition ⇒ commit phase & should be ~~Constraints~~ before modification.

SENSITIVE DATABASE :-

INHERENTLY SENSETIVE :-

sensitive means only few people will get an access to database.

eg :- Nuclear experiment,
Defence Dpt,
Missile position.

DECLARED SENSITIVE :-

Somebody declares the database to be sensitive.

eg :- ગુપ્તચર.

ACCESS DECISION:
1) Availability of data.
2) Assurance, authenticity.
3) Acceptability of Access.

* SOURCE SENSITIVE :-

Data is not important but source of data is important.

SENSETIVE W.R.T. PREVIOUSLY DISCLOSED INFO :-

eg:- Assume there is some location of gold mine. To locate it user enters longitude longitude, lattidutude.
Post is sensitive (declared) But longitude & lattitude is not.

TYPES OF DISCLOSURES :-

EXACT DATA :-
Not tolerable disclosure.

BOUNDS OF DATA :-
Attacker can get the bounds of data which is not acceptable.

$$L \leq z \leq H$$
(L: lower, H: Higher)
$$L < z < H/2$$

NEGATIVE ~~REASONS~~ :- RESULTS :-

eg:- Z is not Y

EXISTANCE :-
Existance of some attributes.
Long distance call.

PROBABLE VALUES: Predefined Range

INFERENCE :- Page 5.8 Table.

DIRECT ATTACKS :-

Drugs, Aid → sensitive
show Aid where Name = 'L'
1] d/B answers for this query then direct attack.

INDIRECT ATTACKS :-

1] sum ⇒ Total
show sum for (sex = 'm') ∧ (Hostel = 'H')
or (sex = 'F')

2] COUNT ⇒

3] TRACKER's ATTACK :- (uses logic / mathematic to comfort query)
- (Limited response suppression)

for N possible combinations go on tracking N-1 combinations so that Nth combination remains obvious answer.

Limited response supression :-
If query answer is in 1 or 2 like this then don't answer.

show count where (sex = F) ∧ (Race = 'c') ∧ (DORM = H)
a b c

↳ d/B refuse to answer because of limited Response suppression.
∴ Try (n-1) combination & for nth query get value.

$$\text{count}(a \wedge b \wedge c) = \text{count}(a) - \text{count}(a \wedge (b \wedge c))$$
$$= \text{count}(a) - \text{count}(a \wedge (\bar{b} \vee \bar{c}))$$
$$= 6 - 5$$
$$= 1$$

CONTROL AGAINST INFERENCE PROBLEM :-

	Holmes	Grey	Watt	Total
H	—	3	2	5
G	2	—	3	5
W	3	4	—	7
Total	5	7	5	17

1) **LIMITED RESPONSE SUPRESSION :-**

2) **RANGE / BOUND :-**
Instead of actual answer some range is given. if avg asked if its value is 5000 then range as 1000 to 7000.

3) **SAMPLED RESULT :-** (Percentage)
30%, 40% instead of exact ans give percentage for sample result.

3+) **PERTURBATION :-**
Add some error i.e. if avg result is 5000 but d/B will return value 5050.

5) **QUERY ANALYSIS :-**
Used to solve / control trackers attack dat aggregation ~~blem~~ problem or data mining.

MULTILEVEL DATABASE :-

(Multiple Grades of sensitivity)
(Two levels :- sensitive / Non sensitive)

NAME	DEPT	SALARY	PHONE
A	Training	30,000	28519
B	sales	25,000	24219
C	Personal	20,000	25220
D	Personal	50,000	27231
E	sales	20,000	29131

pple
⇓
Database
⇓
sensitive database
⇓
Multilevel

Person D gets 50000 (more than all)
Why ?
We don't want to give info about this field
∴ reveal that field (Hide) as Top secret ∴ It won't be counted for any result so that (salary) ~~for~~ column as salary is sensitive.

eg:- Defence Budget — Top secret
Missile - 200 crores
stationary, salary, paper } secret

PROPOSALS FOR MULTILEVEL DATABASE

SEPERATION:- (based on sensitivity level)

PARTITIONING :-
Each attribute is given some value as Top secret, secret, gather the attributes of same value and create a seperate database for each value.

User having 'common' access; can access attributes with common value only.

Disadv :- Redundancy.

ENCRYPTION :-
The d/B is encrypted so that ever if accidently attacked, attacker gets access of it Nothing can be reveled from d/B.

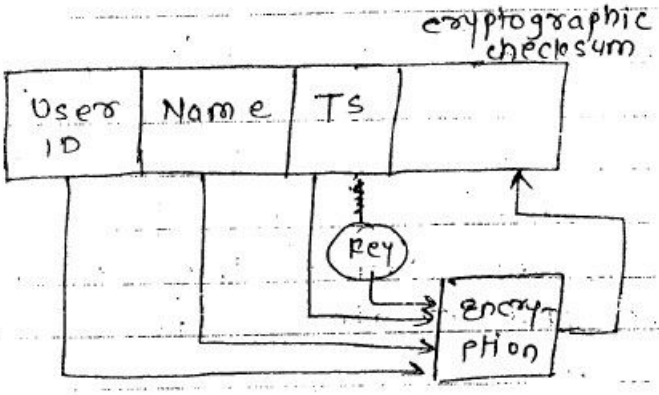
INTEGRITY LOCK :- (proposed by us Air force)
code Named - spray painting.

Item	security level	cryptographic checksum
Name	TS	

checksum :- instead of calculating even/odd parity use checksum method.

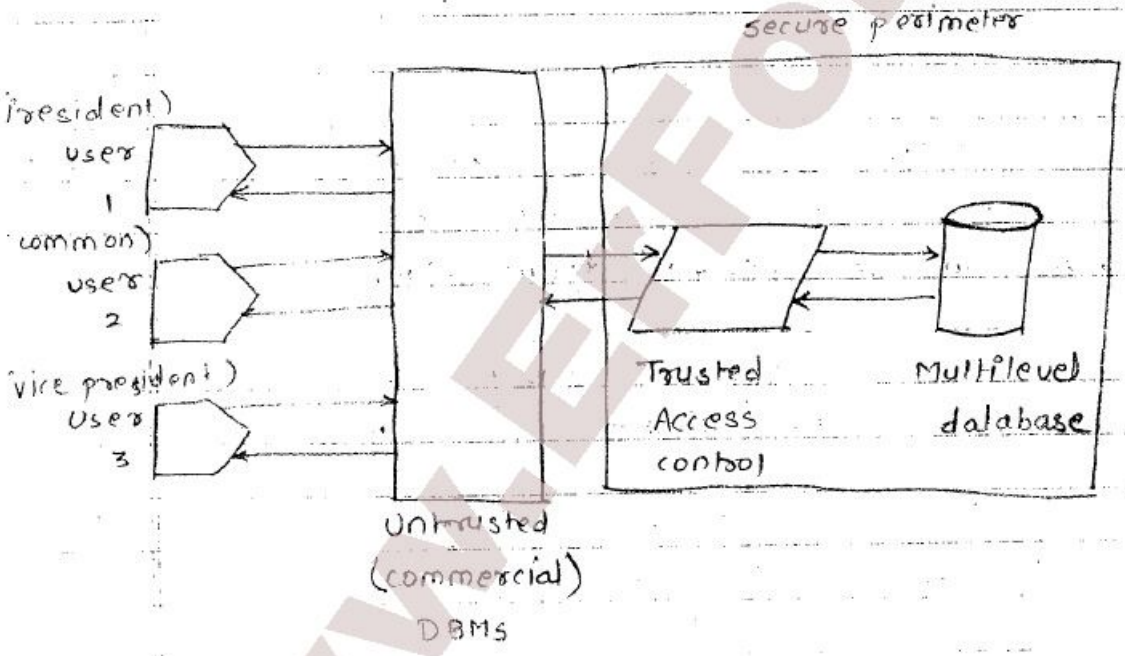
As there ~~are~~ may many persons having same name user ID is introduced.

SENSETIVITY LOCK :-



DESIGN OF MULTILEVEL DATABASE (ECQ) :-

(1) INTEGRITY LOCK :-



For each item attach a prooper security table.

If user 2 wants to access salary of BOB(TS)

query :

show NAME where salary ≥ 50,000

this query is forwarded to trusted controller by untrusted commercial d/B

manager only after user authentication · supply to d/B : d/B gives ans all values satisfying query given to trusted Access controller from d/B.

Now trusted Access controller decides which one out of resulted values is to be given to untrusted manager.

e.g:- Total 15 entry from DB to trusted

5 → TS , 5 → C , 5 - S.

Now trusted check authority of user if user has authority of common access the only 5 values with common access given to user 2.

Problem:-

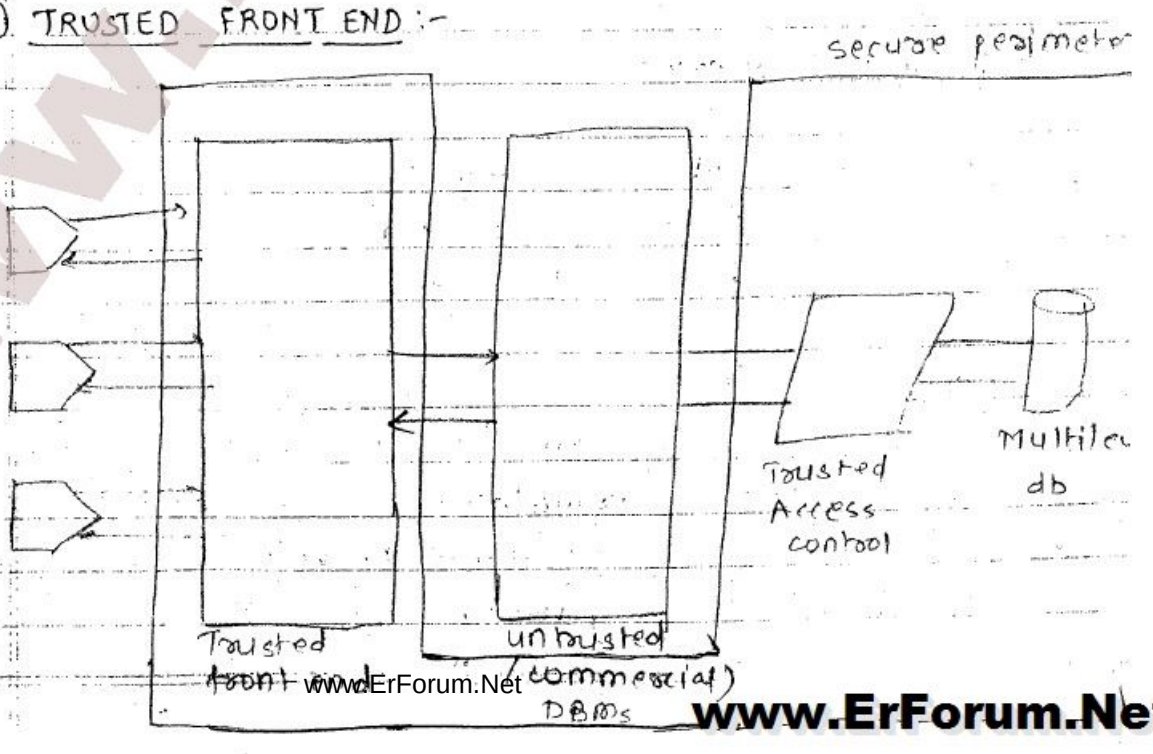
large amount of data unnecessarily is brought to trusted access control.

∴ filtering.

1 way filtering (from query user → d/B) not from (d/B to trusted)

→ Potential covert channel
→ filtering in only 1 way.

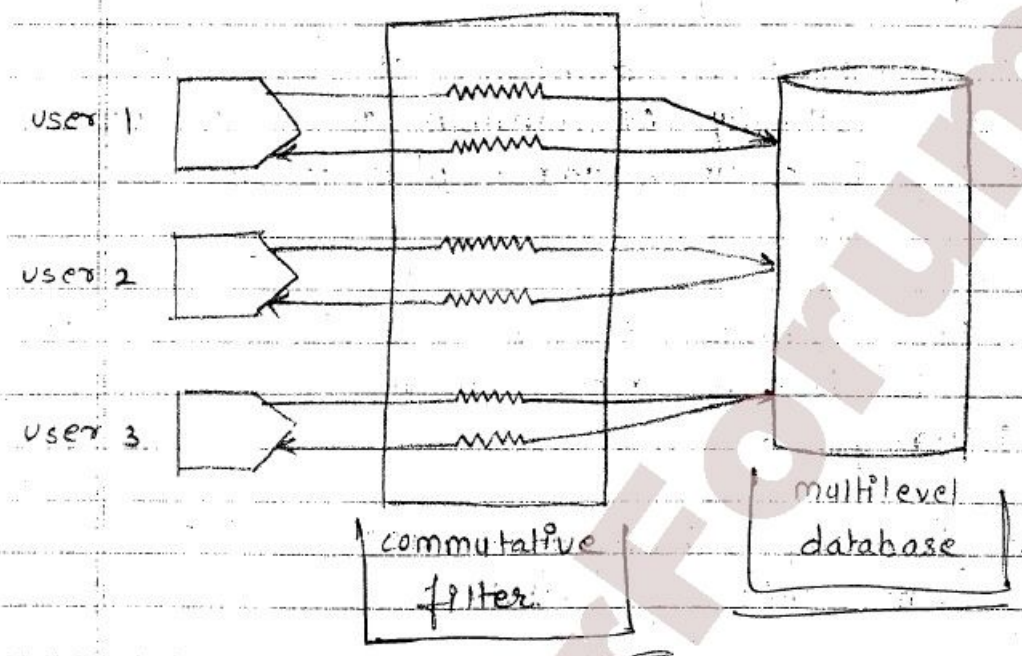
② TRUSTED FRONT END :-



covert problem is solved in trusted front end.

→ No direct contact between user & commercial d/B manager.

(8) COMMUTATIVE FILTER (2 way filter)



User query :-

show NAME where salary ≥ 50,000

Reformatted query :-

show NAME where salary ≥ 50,000
∧ (Name_priv ≥ user_priv)
∧ (salary_priv ≥ user_priv)

2 way filtering
1 user d/B ⇒ reformat query
db ← user → granularity (depending on granularity value is given to user)

Granularity :-

Database
- Entire - schema
- Partial - subschema

attribute - Record / Element

Consider a column

1] If you select entire column value ⇒ one type of granularity if some of those value ⇒ another type of granularity.

www.ErForum.Net
www.ErForum.Net