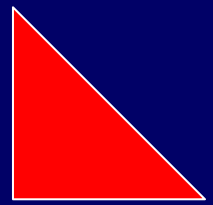
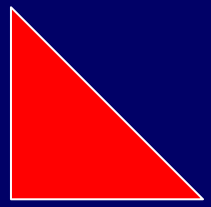


Security

Unit 5.2b

Security



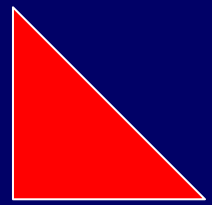
Database Security involves protection against:

- unauthorised disclosures
- alteration
- destruction

The protection which security gives is usually directed against two classes of user

- Stop people without database access from having any form of access.
- Stop people with database access from performing actions on the database
- which are not required to perform their duties.

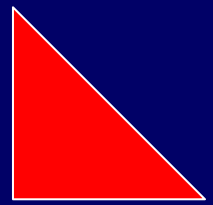
Security cont...



There are many aspects to security

- Legal, social and ethical aspects
- Physical controls
- Policy questions
- Operational problems
- Hardware controls
- Operating system security
- Database system security

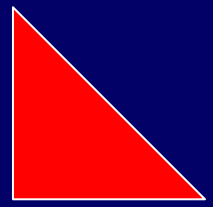
Granularity of DBMS Security



The unit of data used in specifying security in the database can be, for example;

- the entire database
- a set of relations
- individual relation
- a set of tuples in a relation
- individual tuple
- a set of attributes of all tuples
- an attribute of an individual tuple.

DBMS-level Protection



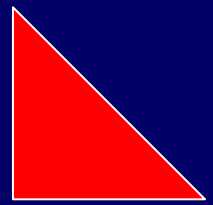
- Data encryption:

  Often it is hard to prevent people from copying the database and then hacking into the copy at another location. It is easier to simply make copying the data a useless activity by encrypting the data. This means that the data itself is unreadable unless you know a secret code. The encrypted data in combination with the secret key is needed to use the DBMS.

- Audit Trails:

  If someone does penetrate the DBMS, it is useful to find out how they did it and what was accessed or altered. Audit Trails can be set up selectively to minimise disk usage, identify system weaknesses, and finger naughty users.

User-level Security for SQL

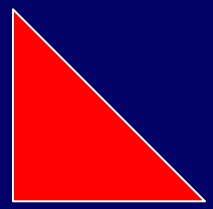


- Each user has certain access rights on certain objects.
- Different users may have different access rights on the same object.

In order to control the granularity of access rights, users can

- Have rights of access (authorisations) on a table
- Have rights of access on a view. Using views, access rights can be control horizontal and vertical subsets in a table, and on dynamically generated data from other tables.

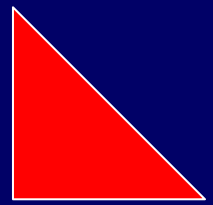
Naming Hierarchy



In a DBMS, there is a two layer approach to naming relations.

- The DBMS is made up of a number of 'databases'. The Database Administrator (DBA) has permission to create and delete databases, and to grant users access to databases.
- Each database is a flat name space. Users with the necessary permission can create tables and views in a database. Because it is a flat name space, all table names must be unique within a database. The DBMS helps users in this regard:
  - table and view names are prepended with the name of the user who created it.
  - the database login name is often taken as the username.

Naming Hierarchy cont...



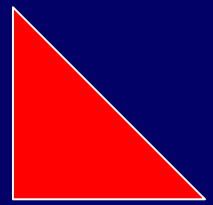
By way of an example, consider a table 'hello' created by a user jbloggs.

- The table will have the name jbloggs.hello
- The user jbloggs can access the table using the name 'hello'
- Other users must use the table's full name to access the table

The user jbloggs can control who has access to the table using the GRANT command.

If the DBA creates a table, and makes it available to PUBLIC, then no user needs to specify the full table name in order to access it.

The GRANT command



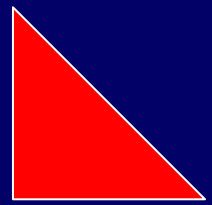
GRANT is used to grant privileges to users:

```
GRANT privileges ON tablename
TO { grantee ... }
[ WITH GRANT OPTION ]
```

Possible privileges are:

- SELECT - user can retrieve data
- UPDATE - user can modify existing data
- DELETE - user can remove data
- INSERT - user can insert new data
- REFERENCES - user can make references to the table

GRANT cont...



The WITH GRANT OPTION permits the specified user can grant privileges which that user possesses on that table to other users. This is a good way to permit other users to look after permissions for certain tables, such as allowing a manager to control access to a table for there subordinates.

grantee need not be a username or a set of usernames. It is permitted to specify PUBLIC, which means that the privileges are granted to everyone.

GRANT SELECT ON userlist TO PUBLIC;