**Branch :- Computer Sci. & Engg. Class :- Final Year**

**Subject :-System and Software Security Lab manual Sem :- VII**

**Teacher Manual**

**PRACTICAL NO 9**

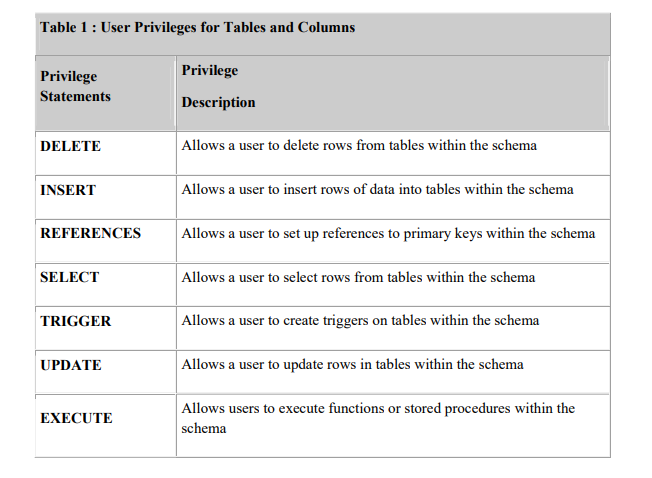
**AIM**:- Apply the security and privileges to a database.

**S/W REQUIRED:-** MySQL

Schemas are an integral part of security in PointBase. When creating a PointBase user, they do not have any access privileges to schemas or other data objects within the database. The PointBase RDBMS only permits the schema owner to grant privileges to the schema and data objects within the schema. The schema owner can grant privileges to the following data objects in the schema:

* Tables
* Columns
* SQL Procedures and Functions

Table 1 describes the privileges that the schema owner can grant users for tables and columns:



**Granting and Revoking Privileges**

When a PointBase database is first created the only user is the default user PUBLIC with a password of PUBLIC. The PUBLIC user owns the default PUBLIC schema. For security reasons, PointBase does not recommend using this schema to store sensitive data. Like any PointBase user, PUBLIC must be granted the appropriate privileges to access data objects in schema owned by other users.

The PUBLIC user can be used initially to create new users and new schema. The PUBLIC user will own any new schema that it creates unless otherwise specified during schema creation. New users are then able to create their own new schema and users, and grant appropriate privileges on schema that they own. All new users must be granted privileges to use the PUBLIC schema if this is required.

To grant the ability for a user to pass a privilege on to other users once granted, you must specify the optional WITH GRANT OPTION qualifier when granting the privilege.

**GRANT Statement Syntax**

GRANT privilege-list

ON object

TO user-list [ WITH GRANT OPTION ]

Use the GRANT statement to grant privileges on a data object. The following describes the GRANT statement syntax.

**Privilege-list Syntax**

privilege [ , privilege [ , privilege ]...] | ALL PRIVILEGES

**Privilege Syntax**

SELECT [ ( column-name [ , column-name ]...)]

| DELETE

| INSERT [ ( column-name [ , column-name ]...)]

| UPDATE [ ( column-name [ , column-name ]...)]

| REFERENCES [ ( column-name [ , column-name ]...)]

| TRIGGER [ ( column-name [ , column-name ]...)]

| EXECUTE

Usage Notes

* If you do not include one or more of these privileges in the GRANT statement, an error will be raised.
* If the optional column-names are not specified for the SELECT, INSERT, UPDATE, REFERENCES and TRIGGER privileges, the GRANT is applied to every column in the table to which the grant is applied.
* If you execute a GRANT statement that contains privileges that you don't have or for which you do not have the right to grant, then PointBase raises an error.

**Object Syntax**

[ TABLE ] table-name

|SPECIFIC routine\_type specific\_routine-name

|routine\_type routine\_name (parameter\_types\_list)

[ TRIGGER ] trigger-name

Usage Notes

If you grant a privilege on an SQL Function or Procedure, then the user can only

EXECUTE that SQL Function or Procedure. The user cannot access tables that the

SQL Function or Procedure uses.

**User-list Syntax**

user [ , user ]... [WITH GRANT OPTION] | PUBLIC

Usage Notes

If you do not specify WITH GRANT OPTION, the user cannot pass the same privilege on to others. However, if you do specify WITH GRANT OPTION, you have given the user permission to pass on the privilege to other users.

Granting a privilege to the user PUBLIC only grants the privilege to the default PointBase PUBLIC user and is not the same as granting a global privilege to all users.

If you grant a privilege with the optional WITH GRANT OPTION and then grant the same privilege without this option (without first revoking the original privilege) the user retains the WITH GRANT OPTION.

Examples

The following statement grants the SELECT privilege on the CUSTOMER\_TBL

table to the user MARKETING\_MGR.

GRANT SELECT

ON customer\_tbl

TO marketing\_mgr;

The following GRANT statement allows the user FINANCIAL\_MGR to delete, insert and update rows from the DISCOUNT\_CODE\_TBL table; it also allows this user to grant the same privileges to others.

GRANT DELETE,INSERT,UPDATE

ON discount\_code\_tbl

TO financial\_mgr

WITH GRANT OPTION;

The following GRANT statement allows the user HR\_MGR to have ALL PRIVILEGES on the table SALES\_REP\_DATA\_TBL. However, the user HR\_MGR will only be granted privileges that the user granting the privileges has the right to grant. For example, if the user granting the privileges does not have the right to grant DELETE privileges, the HR\_MGR will not have the delete privilege.

GRANT ALL PRIVILEGES

ON sales\_rep\_data\_tbl

TO hr\_mgr

**REVOKE Statement Syntax**

REVOKE [ GRANT OPTION FOR ] privilege\_list

ON object

FROM user\_name [ RESTRICT | CASCADE ]

The REVOKE statement takes privileges away from users. The arguments are similar to the

GRANT statement. The major difference is the additional RESTRICT or CASCADE

keyword and the GRANT OPTION FOR clause. The following describes the optional clauses

GRANT OPTION FOR and RESTRICT or CASCADE.

NOTE: If none of the privileges that you are trying to revoke actually exist, an error is raised.

RESTRICT | CASCADE

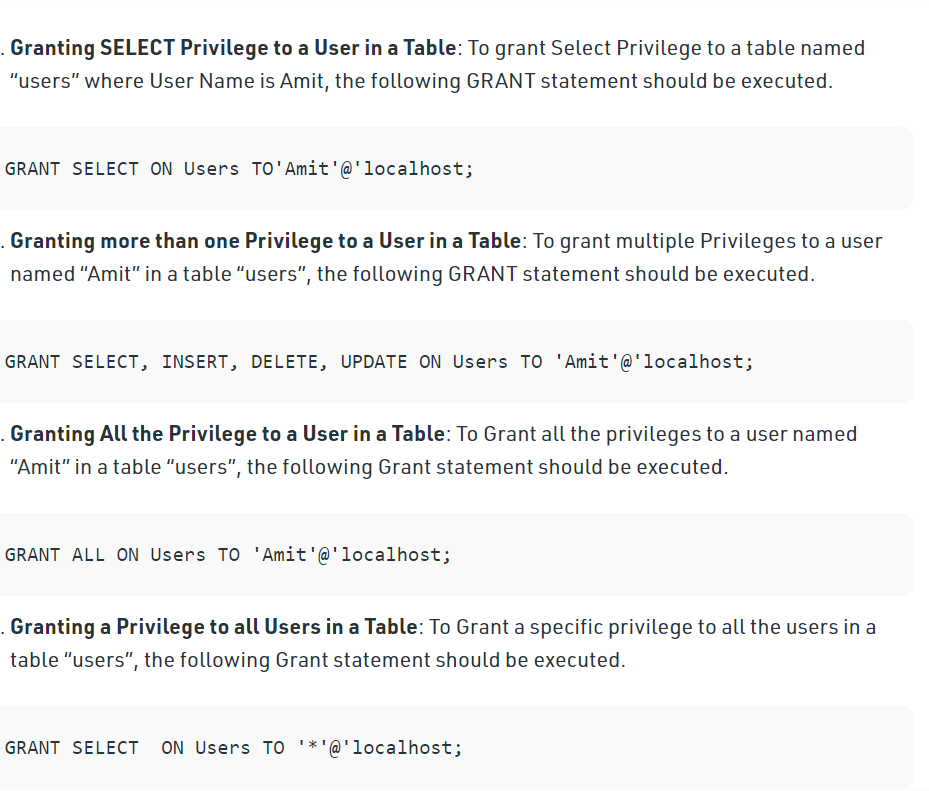
If you use RESTRICT keyword, the privilege will be revoked only from the specified user. If the specified user granted had the WITH GRANT OPTION and granted the same privilege to other users, they will retain the privilege.

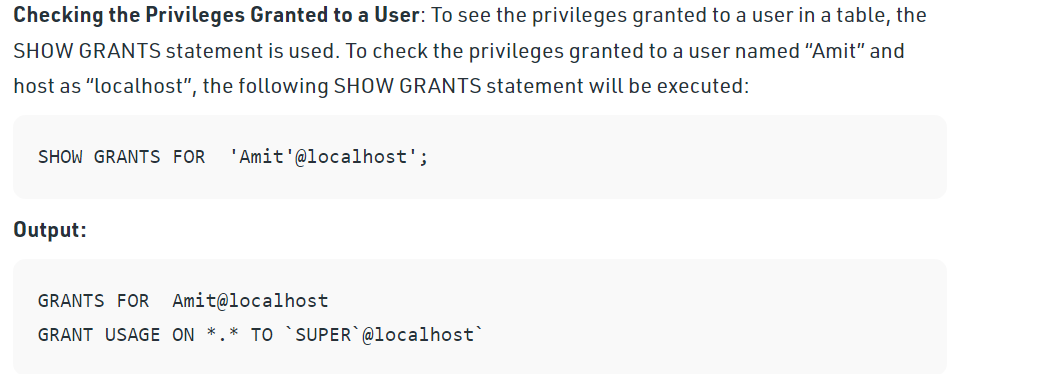
If you use CASCADE, it will revoke the privilege and any dependent privileges as a result of your grant. A dependent privilege is one that could exist, if you granted the privilege that you're trying to revoke, which is what you are trying to achieve as a result of your REVOKE statement.

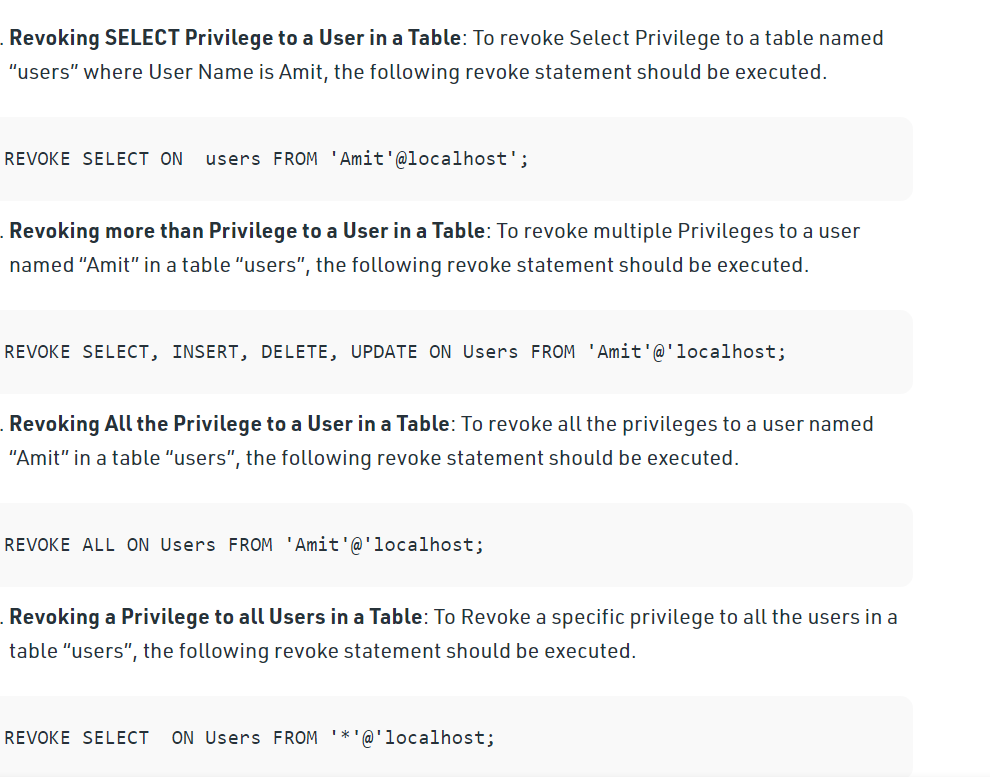
If the optional RESTRICT or CASCADE keywords are not used, PointBase uses RESTRICT by default.

GRANT OPTION FOR If he optional GRANT OPTION FOR clause is used, the WITH GRANT OPTION right is revoked. The actual privilege itself is not revoked. the GRANT OPTION is revoked.

CASCADE and RESTRICT may be used in the same way as a normal REVOKE statement.







**CONCLUSION:** Thuswe have applied various security and privileges to the database