Data and object security

Object Security

Object security controls what content users see within ThoughtSpot. Objects are tables, columns in tables, worksheets, pinboards, and saved answers.

Users gain access to objects when an object owner shares access with them. Owners can share with individual users or with entire groups, giving access to everyone within that group. Objects may be shared with edit or view-only options. A user can automatically share objects with anyone else in the groups to which they belong. This has implications on setting up privileges, and on applying row-level security.

Permissive Security Mode

The default Permissive Security mode of ThoughtSpot means that when someone shares an object with you, you can see all the data it uses, regardless of explicit permissions to the parent object data. You can see a shared pinboard without having access to its underlying worksheet or table.

Advanced Security Mode

ThoughtSpot's Advanced Security mode is opposite of the default permissive mode. Unless the user has explicit permissions to the entire stack of parent objects, they cannot see the data in the child object. For example, in a shared pinboard, you can see data only if you have explicit permissions to the relevant columns of the parent worksheet. Similarly, you can only see the data in a worksheet to which you have access if you have explicit permissions to its parent table object.

Work with your ThoughtSpot support team to enable the Advanced Security Mode on the relevant clusters.

Row level security (RLS)

Row level security controls what data a user can see in each shared piece of content. Even if a user has access to a worksheet, they can only see rows from the tables they have permission to see.

RLS applies at the table level, so it automatically extends to all worksheets, saved answers, and pinboards based on that table, every time. Also, in queries where there are tables with table filters, all joins are always enforced to avoid accidentally allowing users access to data they shouldn't see.

RLS requires three things:

- A table filter with a column (possibly in a joined table) that can be used to determine who can see a row, such as account id or tenant id.
- A group that can be associated with the row of data by name. For example, if the column is `account_id` and has values of `1`, `2`, `3`, users can be assigned to groups `group_1`, `group_2`, `group_3` and

then only see their data.

- Users must be assigned to the group. If they are not assigned to a group that has access, they do not see any data.

Administrative users can always see all rows of data because RLS does not apply to them.

RLS supports a hierarchy of groups, which makes it possible to grant access to some users across multiple groups.

Keep in mind that users within a group can share with one another. If you put everyone in your organization into the same group for RLS, they can share with anyone in the company.

Column level security (CLS)

Column level security lets users see certain columns in a table, but not other columns. This can be accomplished by sharing a limited set of columns in a table with specific users or groups.

Because someone can share with anyone in the same group, they can potentially share restricted columns. For example, if a *Human Resources* repository has a column with salary information, and it appears in a worksheet, any *Human Resources* group member could create an answer with visible salary information and mistakenly share with someone outside of *Human Resources*. That 'outside' person now has access to the salary information. In such cases, we recommend that you work with your ThoughtSpot support team to enable the Advanced Security Mode on the relevant clusters.

System privileges

System privileges refer to what a user can do in ThoughtSpot. For example, can they upload or download data or share with all users. These privileges are defined on a group level and inherit downwards. So, if Group A had child groups Group B and Group C, then any privilege given to Group A is also available to Group B and Group C. What this often means is that separate sets of groups are required to manage privileges.