This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Genesys Security Deployment Guide

Object-Based Access Control

3/15/2025

Object-Based Access Control

Contents

Object-Based Access Control implements user authorization by using permissions to define what each user can do to the objects to which he or she has access.

In general, any object for which permissions is not explicitly granted is forbidden.

Elementary Permissions

User authorization is provided by the combination of a set of elementary permissions, shown in the following table. This security mechanism implemented in Configuration Server allows the system administrator to define separately a level of access for any account with respect to any object.

Permission	Description
Read	Permission to read information and receive updates about the object.
Create	Permission to create objects in this folder.
Change	Permission to change the properties of the object. The Change permission is the same as allowing “Write” access.
Execute	Permission to perform a predefined action or set of actions with respect to the object. This is also required for a user to log in to a Graphical User Interface (GUI) application.
Delete	Permission to delete the object.
Read Permissions	Permission to read the access control settings for the object.
Change Permissions	Permission to change the access control settings for the object.
Read & Execute	• Permission to read information and receive updates about this object. • Permission to perform a predefined action or set of actions with respect to this object.
Propagate	For container objects (such as Tenants, Folders, Switches, IVRs, and Enumerators). The `Propagate` check box controls whether to propagate this set of elementary permissions to the child objects. By default, the check box is selected).

Access Privileges

The access privileges of authenticated user accounts define what the user can and cannot do within this application. The Execute permission is used to control access to applications, solutions, and other configuration objects. Without such permission, the user cannot work with a given application or execute control over a given object. Combinations of the Read, Create, Change, and Delete permissions define the level of access to configuration data. For example, users might have access to a real-time reporting solution but will get reports only about objects they have permission to read.

Access control for daemon applications is different from that for GUI applications. Access permissions for GUI applications are determined by the profile of the person who is currently logged in. Daemon applications do not have an explicit login procedure. Instead, their access permissions are determined by the permissions of the account with which they are associated: a personal account or the SYSTEM account. Any personal account registered as a Person object in the Configuration Database can be used as an account for any daemon application. By default, every daemon application is associated with a special account SYSTEM that has Read and Execute permissions for all objects in the Configuration Database except Access Groups.

Access Groups and Default Security Settings

Access Groups are groups of Person objects who must have the same set of permissions with respect to Configuration Database objects. By adding individuals to Access Groups—and then setting permissions for those groups—access control is greatly simplified.

Genesys offers these preconfigured Default Access Groups:

- Users: Members have Read and Execute permissions with respect to all objects except Access Groups.
- Administrators: Members have a full set of permissions with respect to all objects except the Super Administrators Access Group.
- Super Administrators: Members have a full set of permissions with respect to every object in the Configuration Database. No person is added to this group by default.

In addition, in a hierarchical multi-tenant configuration, Configuration Server creates these Default Access Groups for each new Tenant object:

- Users: Members have Read and Execute permissions with respect to all objects under this Tenant except Access Groups.
- Administrators: Members have a full set of permissions with respect to all objects under this Tenant.

> **Important**
>
> You cannot delete or rename Default Access Groups, although you can change their default privileges.

New Users

By default, Configuration Server considers a new user to be a member of the EVERYONE group. It does not assign that user to any Access Group when he or she is created. Likewise, the new user is not automatically assigned any permissions by default. In effect, the new user has no privileges, and cannot log in to any interface or use a daemon application. The new user must be explicitly added to appropriate Access Groups by an Administrator or by existing users with access rights to modify the user’s account. Refer to *Genesys Administrator 8.1 Help* for more information about adding a user to an Access Group.

By default, this behavior applies to all new users added by Configuration Server release 7.6 or later. Users created before release 7.6 keep their existing set of permissions and Access Group assignments. If you want new users to be added automatically to predefined Access Groups, as was the behavior prior to release 7.6, you must manually disable this feature by using the Configuration Server **no-default-access** configuration option.

For more information about this feature, including how it works and how to modify it, see No Default Access.

Master Account and Super Administrators

The Configuration Database contains a predefined user object, otherwise known as the *Master Account* or *Default User*. This account, named **default** and with a password of **password**, is not associated with any Access Group. The Master Account always exists in the system and has a full set of permissions with respect to all objects in the Configuration Database. You must use this account when you log in to the Configuration Layer for the first time after Configuration Database initialization.

> **Important**
>
> - In addition to emergency situations, you still must use the Master Account for some specific administrative tasks, especially during migration. Refer to the description of the specific tasks throughout this and other documents, including the *Genesys Migration Guide*, to determine whether you need to use the Master Account, or whether you can use another account that has the required permissions.
> - Genesys recommends that you change the default name and password of the Master Account, store it securely, and use this account for only emergency purposes or whenever specifically required.

During one of your first working sessions, create non-agent accounts for everyone who needs full access to all objects and add these accounts to the Super Administrators group. By default, every member of the Super Administrators group has the same permissions as the Master Account.

EVERYONE Group

Think of the EVERYONE group as an Access Group that includes every user registered in the Configuration Database. You cannot delete or modify this group, which, by default, has no permissions set for any configuration objects.

Multiple Permissions

Multiple (and unequal) permissions can affect a User's access to an object. If a User belongs to multiple Access Groups and those Access Groups have different permissions for the object, the User gets the logical union of privileges from the set of access privileges with one exception: the No Access access privilege supersedes all others.

Examples

Assume that:

- User John is a member of Access Group A and Access Group B.
- Access Group A has Read-only access to the Host Friday, but Access Group B has Read/Write access to the Host Friday.

As a result, John has Read/Write access to the Host Friday.

To understand the exception to this rule, now assume that:

- User John joins Access Group C, which has No Access privileges to the Host Friday.

As a result, User John now has no access to the Host Friday.

Setting and Changing Permissions

Permissions are set and changed in Genesys Administrator on the **Permissions** tab of the appropriate object.

> Important
>
> Use caution when assigning permissions. Remember, the more complex the security system is, the more difficult it becomes to manage the data and the more it affects the performance of the Configuration Layer software.

Granting Permissions

To grant permissions, use the following steps. **[+] Show steps**

1. In Genesys Administrator, open the **Permissions** tab of the object for which permissions are to be granted.
2. Click **Add User** or **Add Access Group**, as applicable. A list of configured Users or Access Groups is displayed in a dialog box.
3. Select the User or Access Group to be granted permission.
4. Click **OK**. The dialog box closes, and the selected User or Access Group appears in the list on the **Permissions** tab, with default Read permission.
5. Click **Save** to save your configuration changes.

Modifying Permissions

To modify permissions, use the following steps: **[+] Show steps**

1. In Genesys Administrator, open the **Permissions** tab of the object for which you want to modify permissions.
2. Do one of the following:
   - Double-click on the name of the User or Access Group for whom you want to change permissions. A list of permissions appears in the **Access** dialog box; those permissions with a checkmark are currently assigned to that User or Access Group. Check those permissions that you want the User or Access Group to have. Clear the checkbox for those permissions that you do not want the User or Access Group to have.
   - Click the corresponding entry in the Access column and select an Access Level from the drop-down list. These Access Levels are pre-defined sets of permissions.
3. When you have checked the required permissions and cleared the permissions not required, do one of the following:
   - Click **OK** to save the changes. (If no changes were made, the **OK** button is not active.)
   - Click **Cancel** to save the permissions with no changes.
4. Click **Save** to save your configuration changes.

Removing Permissions

To remove permissions previously granted to a user or group of users, use the following steps: **[+] Show steps**

1. In Genesys Administrator, open the **Permissions** tab of the object from which the User or Access Group is to have permissions removed.
2. Select the User or Access Group.
3. Click **Remove**. The User or Access Group no longer appears on the `Permissions` tab for this object.

> **Important**
>
> This action does not remove the User or Access Group from the Configuration Database. It only removes the User or Access Group from the list of User or Access Group objects that have access to this particular object. To remove the Access Group or User from the Configuration Database, refer to instructions in *Genesys Administrator 8.1 Help*.

4. Click **Save** to save your configuration changes.

Changing Permissions Using Propagation

The **Propagate** check box in the properties of so-called container objects (such as Tenants, Folders, Switches, and IVRs) allows you to manage access permissions to both the container object and those objects that they contain—the so-called child objects—without affecting the permissions of other

Users or Access Groups.

When the **Propagate** check box is selected (the default setting) for a container object, any changes to permissions to the container object will be propagated to (that is, also made to) the permissions to each child object.

Use propagation when you want to set identical permissions for a user to a container object and all its child objects. For example, if you are setting up a new user or Access Group, and that user or group is to have identical permissions to a container object and all the objects that it contains, you have to add permissions for that user or groups only once—in the container object.

If you want to change the permissions to the container object without changing those of the child objects, clear the **Propagate** check box before changing the object’s access permissions.

The setting of the **Propagate** check box (checked or unchecked) is saved between propagations. This enables you to ensure that subsequent changes to permissions settings are consistently propagated or not.

If you want to set permissions for only the child objects without changing those of the parent object, set the child permissions as required. If the consistently check box in the parent is checked for the users whose permissions were changed, any changes for the child will last only until the next propagation. However, if you then change permissions for another user at the parent level, the resulting propagation will not overwrite the earlier manual change to the first user.

Changing Permissions Recursively

If the **Propagate** and **Replace permissions recursively** check boxes are selected for a container object, all permission settings for its child objects are removed and replaced with all permission settings configured for the parent object. Recursion is basically propagation on a clean slate—removing any access rights to the child objects for any users and groups except those propagated from the parent object.

The **Replace permissions recursively** check box is unchecked by default, and must be selected explicitly each time that you want to propagate recursively.

Hierarchical Multi-Tenant Environments

Generally, permissions function in a hierarchical multi-tenant environment in the same way as they do in an enterprise environment. However, there are some exceptions. This section identifies the issues related to using object permissions in a hierarchical multi-tenant environment, and provides workarounds where available.

Accessing Tenants and Objects in Other Tenants

By default, and with one exception, users in one tenant cannot create another tenant, nor can they access any objects in another tenant. Generally, the only exception to this situation is that the Default User (using the Master Account) and members of the SuperAdministrators Access Group can create new tenants and access objects in other tenants.

The details of default behavior in a hierarchical multi-tenant environment, and recommendations to work around the limitations imposed by that default behavior, are given in the following sections.

Creating New Tenants

A new Tenant object can be created only by the Default User or a user who is a member of the Super Administrators Access Group.

When a tenant is created, permissions to it are granted to the following Access Groups, as follows:

- Environment/default (the Default user)—Full control
- Super Administrators (from the Environment Tenant)—Full control
- SYSTEM (from the Environment Tenant)—Read & Execute (RX)
- [new Tenant]\Administrators—Read & Execute (RX)
- [new Tenant]\Users—Read & Execute (RX)

Resolution
Add users as necessary to the Super Administrators group to enable them to create tenants. Refer to *Genesys Administrator 8.1 Help* for instructions about adding users to Access Groups.

Providing Users Access to Objects in Other Tenants

By default, a new user is not granted access to any objects. As in an enterprise environment, each new user must explicitly be granted permissions and/or added to an Access Group with permissions, to access any objects. See No Default Access for New Users for more information.

To log in to an Application, a user must have at least Read & Execute permissions for that Application. After he or she is logged in, the user can access only those objects in his or own Tenant; he or she cannot access any objects in another Tenant.

Resolution
To gain access to objects in another Tenant, the user must be granted permissions to those other objects by one of the following:

- the creator of the other Tenant
- another member of the Super Administrators Access Group

Providing Users in Parent Tenant Access to Objects in Child Tenants

A user in a parent tenant has no default access to the objects in the child tenants.

> **Tip**
> To work around this limitation, do one (or both) of the following:
>
> - Explicitly grant at least Read access to all child tenants.
> - Explicitly add the user to one of the two built-in Access Groups in each child

> tenant—Administrators or Users.

Voice Platform Solution Limitation

When a hierarchical multi-tenant configuration is used with the Voice Platform Solution in a managed server setting, a major limitation arises when creating Tenants and Direct Inward Dialing (DID) numbers. In essence, this limitation forces the system owner to create and maintain all Tenants and DIDs for all tenants.

In the Voice Platform Solution, DIDs must be unique across the entire system. The software is designed to validate this uniqueness when DIDs are created. This requires that the user who inputting this information must have at least Read access to all DID objects in all Tenants, and therefore access to the Tenants themselves. However, in a managed server environment, it is highly unlikely that the Service Provider wants one tenant to see, or even know about, other tenants. Therefore, the only user that could input this information would be a member of the Super Administrators Access Group, namely, the system owner. The current model of access permissions does not permit any workaround to this situation at this time.