



APIs by Example: Override Group Profile

[System iNetwork](#)

System iNEWS Staff

Thu, 12/16/2004 (All day)

The Integrated File System APIs include some powerful security functions that form the basis for this installment of APIs by Example. In this article, I will present the Override Group Profile (OVRGRPPRF) command.

The purpose of the OVRGRPPRF command is to provide an easy, controllable way to grant temporary access to objects that, due to security procedures or audit requirements, someone might not normally have. For example, a programmer might be given temporary access to a file to fix a problem with the system.

Providing this authorization by means of a group profile offers some advantages over alternative methods because the true identity of the user actually running the job is preserved. The job's current user profile remains the same. Only the primary group is replaced.

Another advantage of this approach is the ability to exclude a user profile that inherits a group profile's *ALLOBJ special authority from accessing certain objects using the regular object authorization facilities.

A user profile that, itself, has *ALLOBJ authority is able to access any object on the system. When *ALLOBJ special authority is detected in the current user profile, the system's object authorization lookup algorithm grants access immediately, bypassing all other authority checks.

Since private authorities to an object are evaluated prior to the group authority, a user profile having *EXCLUDE private authority to an object will not be able to access that object, even if the user's group has *ALLOBJ special authority. The authorization lookup process never reaches the step where group authority is checked because it terminates as soon as it finds that the private authority is excluded.

Therefore, you will be able to prevent a given user profile from accessing a command or object, even if the group has *ALLOBJ authority, by setting the user's authority to *EXCLUDE.

The following three APIs provide the foundation of this utility:

getgrnam -- Get Group Information using Group Name. This API accepts a group profile name and returns, among other information, the group profile's group ID. The group ID is an integer value that provides a unique identification of a group profile. It will be used by the qsyssetgid API to set the primary group profile of the current job to the one specified on the OVRGRPPRF command.

qsyssetgid -- Set Effective Group ID. Sets the current job's primary group. A job can have both a real and an effective group ID, similar in concept to profile swapping where a job can have both an original and a current user profile. The real group ID is the original group profile at job creation time. The effective group ID is the currently active group profile, and consequently can be different from the real group ID.

getegid -- Get Effective Group ID. This API returns the currently effective group ID and is used by this utility to save the group ID so that it can be restored later.

This utility also uses the _MODINVAU (Modify Invocation Authority Attributes) MI built-in. This is used because *ALLOBJ special authority is required to run the qsyssetgid API. The _MODINVAU built-in is called to prevent that special authority from propagating to subsequent invocation levels.

Here's what the OVRGRPPRF command prompt looks like:

```
= = = = =
```

Override Group Profile (OVRGRPPRF)

Type choices, press Enter.

Group profile	Name
Authorization code	Character value
Reason	

= = = = =

A short explanation of the command's parameters follows. Please refer to the command's help text if you need further details.

Group Profile: The group profile that the current job will have its primary group profile temporarily changed to.

Authorization code: An authorization code provided by the security officer to enable the group profile override to take place.

NOTE: In the next installment of APIs by Example, two commands will be provided to add and manage authorization codes. For now, specifying the special value *NONE will allow the OVRGRPPRF command to be run. If, for security reasons, you do not want to have this option available, please remove the following condition from the CBX128V program source prior to compiling the module and creating the program.

```
--> If PxAutCod <> '*NONE';
      ExSr ChkVldLst;
--> EndIf;
```

Reason: A brief explanation of the reason for the group profile override. This information will also be recorded in an audit journal entry.

The OVRGRPPRF command deposits audit journal entries to system audit journal QAUDJRN, documenting both initiation and termination of the command. The audit entries will have journal code 'U'. The entry type for group override initiation is 'A2' and then entry type for override termination is 'A3'. The data structures JrnEntA2 and JrnEntA3 in the program CBX128 documents the layout of the two journal entry formats. If audit journal QAUDJRN does not exist, the command will not be allowed to run.

To store and verify the command's authorization code, a validation list object is created when the command is built (see below). The OVRGRPPRF command will verify that this validation list object is owned by user profile QSECOFR and returns an error if this is not the case.

Many of the access and integrity checks are performed in the command's validity checking program. To ensure that this program has been run correctly, it issues a profile token that is subsequently verified by the command's processing program. The profile token generation also offers the potential to add a password parameter to the command to verify the current user's identity.

To learn more about profile tokens, refer to the following article from the Programming Tips Newsletter of April 29, 2004: <http://www2.systeminetwork.com/article.cfm?ID=18522>

Running the following command

```
OVRGRPPRF GRPPRF( QSECOFR )
          AUTCOD( *NONE )
          REASON( 'Test OVRGRPPRF command' )
```

will present the Command Entry panel, from which point on your job's primary group profile will be QSECOFR. Once the tasks prompting the use of the OVRGRPPRF command have been completed, press F3 from the Command Entry panel, and your original primary group profile will be restored.

By design, the OVRGRPPRF command is only intended to be run from the command line, and the command is therefore restricted to this environment only.

The OVRGRPPRF command includes the following sources:

CBX128 -- Command processing program that performs the group profile override.

CBX128V -- Validity checking program performing integrity and access control.

CBX128H -- Command help text panel group.

CBX128X -- Command definition source member.

CBX128M -- Creates and configures all command objects.

Once all of the source members listed above have been copied to their default source files and the program CBX128M has been compiled, you can call CBX128M to create all necessary command objects. Specify the source file library as the only parameter. All the command objects will be created in that library as well.

```
Call    Pgm( CBX128M )    Parm( ' ' )
```

Please note that to successfully run the above program, the user profile performing the call must have *ALLOBJ special authority.

Please also note that the primary objective of the OVRGRPPRF command is to demonstrate practical use of the IFS security APIs. Before creating and installing the command on a production system, you should carefully and thoroughly test the utility to ensure that it meets your security requirements and guidelines.

This article demonstrates the following APIs:

Get Effective Group ID (getegid) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/getegid.htm>

Get Group Information using Group Name (getgrnam) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/getgrnam.htm>

Set Effective Group ID (qsysetegid) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsyseteg.htm>

Retrieve Job Information (QUSRJOBI) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qusrjobi.htm>

Retrieve Object Information (QUSROBJD) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qusrobsd.htm>

Send Program Message (QMHSNDPM) API:

<http://publib.boulder.ibm.com/iseres/v5r2/ic2924/info/apis/QMHSNDPM.htm>

Send Journal Entry (QJOSJRNE) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/QJOSJRNE.htm>

Find Validation List Entry (QsyFindValidationLstEntry) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/QSYFIVLE.htm>

Find Validation List Entry Attributes (QsyFindValidationLstEntry Attrs) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/QSYFIVLA.htm>

Verify Validation List Entry (QsyVerifyValidationLstEntry) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/QSYVFLVLE.htm>

Add Validation List Entry (QsyAddValidationLstEntry) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsyavle.htm>

Generate Profile Token (QsyGenPrfTkn) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsygenpt.htm>

Check Profile Token User (QsyChkPrfTknUser) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsychktu.htm>

Get Profile Token Time Out (QsyGetPrfTknTimeOut) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsygetpt.htm>

Remove Profile Token (QsyRemovePrfTkn) API:

<http://as400bks.rochester.ibm.com/iseres/v5r2/ic2924/info/apis/qsyrptkn.htm>

The following MI built-in function is demonstrated in this article: _MODINVAU -- Modify Invocation Authority Attributes http://publib.boulder.ibm.com/iseres/v5r1/ic2924/tstudio/tech_ref/mi/MODINVAU.htm

The following function from the ILE C runtime library is demonstrated in this article:

strerror – Set Pointer to Run-Time Error Message

<http://publib.boulder.ibm.com/iseres/v5r2/ic2924/books/c415607107.htm#HDRSTRERRO>

You can retrieve the source code for this utility from

<http://www2.systeminetwork.com/noderesources/code/clubtechcode/OvrGrpPrf.zip>.

The above article was written by Carsten Flensburg. If you have any questions, you can contact Carsten at <mailto:flensburg@novasol.dk>.

Source URL: <http://iprodeveloper.com/rpg-programming/apis-example-override-group-profile>