

## Blocking Adopted Authority Propagation

[System i Network Systems Management Newsletter](#)

Chuck Lundgren

Wed, 02/18/2004 (All day)

Adopted authority is an attribute of programs, service programs, and SQL packages that, when invoked, passes the authority of its owner to program users – in effect giving the program users their own authorities plus the authority of the program, service program, or SQL package's owner. The user keeps this augmented authority for as long as the program is in the invocation stack. Adopted authority is used to temporarily give users access for some specialized purpose to objects they normally can't reach.

When a program adopts its owner's authority, that authority also includes any special authorities the owner may have. However, if the owner is a member of a group, the group's authority is not included in an adopted authority.

There are two ways an iSeries program can adopted authority:

a. The program object can have its USRPRF attribute set to either \*USER or \*OWNER, where:

\*USER specifies that runtime authority will only be checked for the user profile currently running the job, and

\*OWNER specifies that if the user profile currently running the job does not have sufficient authority to a given action, then the authority of the user profile owning the program will be used.

b. If the program doesn't have the USRPRF attribute set to \*OWNER, it can inherit adopted authority from a program higher in the invocation stack, but only if its USEADPAUT attribute has been set to \*YES using the CHGPGM command. USEADPAUT is set to \*NO when the program is first created.

The above steps control how authority is adopted from a program higher in the invocation stack. However, there is no program attribute to control whether authority should be adopted at levels lower than the current invocation level; in other words, whether authority should be propagated. This means, for example, that if a program adopts authority from a powerful user profile and calls a program or command that makes a command line available to the user, that user will be able to run commands using the powerful user's authority.

There is a way to prevent authority propagation using the ILE MI built-in function '\_MODINVAU' (Modify invocation authority attribute), which is available to all ILE compilers. This built-in function is capable of setting the authority propagation attribute for the current invocation of a program without affecting the program object permanently.

Below is program CBX908, which shows how authority propagation blocking works:

```
Pgm

Dcl      &ModTplPpg   *Char   1    x'00'
Dcl      &ModTplBlk   *Char   1    x'01'

CallPrc  Prc( '_MODINVAU' )   Parm( &ModTplBlk )
Call     QCMD

CallPrc  Prc( '_MODINVAU' )   Parm( &ModTplPpg )
Call     QCMD

Endpgm:
EndPgm

Here's how to compile CBX908:

CrtClMod  Module( CBX908 )
```

```
SrcFile( QCLLESRC )
SrcMbr( CBX908 )
DbgView( *LIST )
CrtPgm      Pgm( CBX908 )
Module( CBX908 )
ActGrp( QILE )
UsrPrf( *OWNER )
ChgObjOwn  Obj( CBX908 )
ObjType( *PGM )
NewOwn( QSECOFR )
```

When run, CBX908 shows the following:

- a. Propagation of adopted authority is blocked and QCMD is called. Running the command WRKUSRPRF \*ALL will show you only the user profiles to which the executing user profile has authority. Press F3 to continue.
- b. Propagation of adopted authority is allowed and QCMD is called. Running the command WRKUSRPRF \*ALL again will now show you all user profiles, regardless of the authority that the executing user profile has to the listed user profiles. Press F3 to end the test program.

Please note that for this test to work, the user profile running this program must have LMTCPB(\*NO). Adopted authority does not override this user profile attribute.

The \_MODINVAU built-in function is documented here:

[http://publib.boulder.ibm.com/iserics/v5r1/ic2924/tstudio/tech\\_ref/mi/MODINVAU.htm](http://publib.boulder.ibm.com/iserics/v5r1/ic2924/tstudio/tech_ref/mi/MODINVAU.htm)

For a discussion of adopted authorities, read "[Security Auditing](#)" by Carol Woodbury at <http://www2.systeminetwork.com/article.cfm?ID=2542>.

The above sample program was written by Carsten Flensburg. For questions regarding this tip, contact Carsten at <mailto:flensburg@novasol.dk>.

**Source URL:** <http://iprodeveloper.com/database/blocking-adopted-authority-propagation>