PowerTech
your security expert

ARTICLE

Our System Administrator did *WHAT?*

By Robin Tatam

One of the greatest challenges that an organization faces when securing an IBM i environment is protecting the system from the very people who are charged with its care: programmers, administrators, and security officers. While these power users often need access to restricted objects and commands, they rarely need that level of access 24 hours a day, and definitely not without accountability.

Fortunately, IBM i lets you audit events in a secure repository for forensic analysis and reporting. If you haven't seen these controls, we offer an event auditing white paper on our Web site at www.powertech.com, and we conduct frequent Webinars on auditing.

Setting Up Auditing

The first step to auditing in IBM i is to create a security audit journal to collect event information. Starting with V5R3, IBM provided the Change Security Auditing (CHGSECAUD) CL command, which creates the security audit journal QAUDJRN, and creates and associates the first journal receiver. By prompting the command, you can override the default journal receiver name and location. You also can designate the initial settings for two auditing system values: QAUDCTL is the auditing on/off switch; QAUDLVL designates which events will be recorded for all users.

The QAUDJRN journal must reside in the QSYS library. I recommend that you create a separate secure library to contain the journal receiver. Since the system generates many receivers over time, a secure library means easier receiver management than using the default receiver library of QGPL. If auditing is currently active, you can create a fresh journal receiver in the new library and use the CHGSECAUD command to direct subsequent events to it. You also can move previous audit journal receivers into the new library.

The PowerTech Group, Inc.
www.powertech.com
TEL USA: 253.872.7788
TOLL FREE: 800.915.7700

Once you have established an auditing infrastructure, you can begin to audit events, including:

- System events
- User activities
- Object accesses

User Auditing

You can view current audit settings using normal user profile commands, but to change the configuration, you must use the Change User Auditing (CHGUSRAUD) CL command. (You must have *AUDIT special authority to use the CHGUSRAUD command, which supports the separation of duty between auditors and administrators.)

PowerTech recommends that activities should be audited for any user with command line permission. Users have command line permission if their profile has a Limit capabilities setting of *NO or *PARTIAL. Network interfaces, such as FTP and DDM, permit commands to be executed and may not adhere to a limited capabilities restriction. You should control and audit these interfaces using an exit point solution, such as PowerTech's Network Security because network data access is not normally auditable.

There are sixteen categories of events that can be audited for users in IBM i 6.1:

- *AUTFAIL Authority Failures
- *CMD Commands
- *CREATE Object Creations
- *DELETE Object Deletions
- *JOBDTA* *Actions Affecting Jobs*
- *NETCMN* *Network Communications*
- *OBJMGT Object Management
- *OPTICAL Optical Drive Operations
- *PGMADP Program Adoptions
- *PGMFAIL Program Failure
- *PRTDTA Print Data
- *SAVRST Save and Restore Operations
- *SECURITY* *Security Operations*
- *SERVICE Service Functions
- *SPLFDTA Spooled File Functions
- *SYSMGT System Management

(The italicized categories support subset values to restrict the volume of events that may be generated.)

The QAUDLVL system value supports all but one of these categories to audit the associated activities for all users. The value of *CMD is unique to individual user auditing. It records the commands that a user runs via the command line and through the execution of program code.

The Operating System Challenge

So far I have discussed how to manually configure the operating system to audit system events and user activities. Normally, we only audit users who can enter system commands. This leaves the issues of how to control administrators and how to report on the large volumes of audit data that is generated.

Removing command line access for administrators or programmers is not a viable option, but you should audit their activities. In addition, restrict each profile to have only the special authorities and private authorities that are needed throughout the day and remove unnecessary capabilities.

Solution 1: Adopting Authority

When a system activity, such as resetting a password, requires a special authority, consider using a custom program. For example, a password reset program could adopt the authority of a profile with Security Administrator (*SECADM) authority and prevent access to other profile settings. This permits a user to perform a restricted function without needing security administrator rights. Of course, any program that adopts authority for sensitive activities should first verify that the user is eligible to perform the activity.

Authority adoption is cumulative, meaning that the program owner's authority is added to the run-time user's own authority. The more programs in the call stack, the more authority the user may potentially receive. If the run-time user already has the necessary authority (perhaps through *ALLOBJ special authority), there is no way to reduce the capabilities. It is not very feasible to have a program for every function that a programmer or administrator might need to perform.

Solution 2: Profile Switching

Profile switching is another solution to the issue of temporary authority. IBM provides programming APIs to allow a job that was started under one profile to assume the authority of another profile mid-process. PowerTech Authority Broker, a powerful solution built around these APIs, is designed to address the issue of controlling powerful users. Authority Broker lets you remove the special authorities and private authorities from a user's normal profile and provide elevated access on demand when it is required. There are numerous advantages to profile switching using Authority Broker:

- The user does not have to manage another user profile and password
- Detailed auditing points to the user that started the job
- Notification events provide messaging of use to interested parties
- Comprehensive reporting (including a .csv file export) allows you to analyze your data
- The "Firecall" feature allows particular switch profiles to be used only after activation by an administrator

There are many other powerful application features, including the ability to use exit programs to perform custom processing as part of a switch, and the ability to use switch profiles in your own application programs. A popular use is to reduce a user's authority before they perform a function like STRSQL or Query.

We will always need administrators and security officers. But, we must ensure that they do not become our biggest vulnerability. Overly powerful users are a common issue on IBM i that you want to address in your security plans. Balance an auditor's desire to strip these capabilities from profiles with the fact that users occasionally need alternate privileges to perform their job. Strong auditing and notification functionality prevents a powerful user from "running wild" and provides accountability.

For more information on PowerTech Authority Broker, or for a free security assessment of your system (including a review of user capabilities), visit **www.powertech.com/mcp/**.

Robin Tatam *is the Director of Security Technologies for PowerTech, the leading provider of security solutions and services for IBM i servers. Robin was co-author of the Redbook* IBM System i Security: Protecting i5/OS Data with Encryption *and is a frequent speaker and writer on security topics. Contact Robin at* ***robin.tatam@powertech.com****.*