

# NTFS SECURITY & SECURITY PITFALLS

By Mark E. Donaldson

## NTFS PERMISSIONS OVERVIEW

Windows NT provides security features to prevent unauthorized access to data, programs, printers, etc. It complies with the government Class C2 standard, which basically requires:

- Logon protection, where each user has a unique name and password.
- Access protection, to limit who can access a file or run a program or use other system resources.
- Audit capability, to make a record of anyone who has tried to access something they are not allowed to access.
- Memory protection, in which memory (RAM, not disk) is reinitialized when a process finishes using it.

The second and third points are the subject of this article.

## How Security is Assigned

All Windows NT system resources are treated as "objects". This includes all files and programs. Thus, they can all be processed by the Windows NT object manager, all in the same way, which makes the code much simpler and more reliable. Every time users attempt to access an object, they must pass through a Windows NT security gateway, which is the object manager. Since security is applied through the Master File Table, this applies only to files and programs on NTFS partitions. The FAT file system does not support access protection. When an object is created, it is immediately assigned its security descriptors. The primary purpose of these descriptors is to list the protections for the object. This list is called the Access Control List (ACL). The ACL includes a series of entries, called Access Control Entries (ACE). Each ACE contains a security ID (the name of one user or group) and the permissions which that ID has in regard to this object. Usually the creator of the object is the owner, who can specify the protections for that object. There are five built-in groups: Administrators, Power Users, Users, Backup Operators, and Guests. Additional user groups can be created by Administrators and Power Users. In addition, there are the individual users. Any of these groups or users can have an ACE created for it and inserted in the ACL of an object. ***There are three criteria, which the security manager uses to assign ACLs to new objects. In descending order of priority, they are:***

1. If the creator of the object specifies an ACL for the object, then that ACL is used.
2. If no ACL is specified, and the object has a name, then the security system checks the directory to which the object belongs. If the ACL for that directory contains ACEs which are marked "inherit", then those ACEs are used to make the ACL for the new object.
3. If the creator did not specify an ACL and the object does not have a name, then the creator's default ACL is attached to the new object.

## How Security is Used

When a user attempts to access an object, the user is assigned an object handle. The security system uses the security reference monitor to check whether this user is allowed to access the object. It does this by checking the ACEs in the object's ACL until it finds one that matches the user name or any group of which the user is a member. If the access is allowed, the handle is granted; otherwise access is denied. If security auditing is enabled for that object, an entry is made in the audit log, recording the object and the security ID of the user who attempted access. An alarm can also be sounded or displayed on the security administrator's monitor. Note that the security reference monitor

# NTFS SECURITY & SECURITY PITFALLS

By Mark E. Donaldson

stops its check at the first ACE that fits the user. It is possible for the user to fit more than one ACE (for example, username ROCKY and group USERS). The ROCKY ACE may allow access, while the USERS ACE denies access. Whether Rocky gets access depends on which ACE comes first in the ACL. This feature can be useful. In the example above, you can exclude everyone in the group USERS, except for selected users, such as ROCKY.

## How to Enable Security

The details on enabling security can be found by starting from your desktop and clicking Start, then Help. Type the key word or phrase (given below). The key word will highlight in the index, with several subtopics below it. Double-click the subtopic you want. Each key phrase below is followed by its subtopics.

- access permissions
- DCOM applications
- files, directories
- inheritance
- printers
- RAS
- shared folders
- permissions
- DCOM applications
- files
- directories
- security auditing
- DDE shares
- Domains
- events
- files, read-only
- logs, Event Viewer
- ownership
- policies
- user accounts

## NTFS SECURITY PITFALLS

Over the years we have come across certain problems which you can run into through incorrect use of these security features. This article covers the most important ones, and how to avoid them.

### Omitted Permissions

When an NTFS partition is formatted, it is assigned the Access Control Entry (ACE) "Everyone – Full Control" which, as the name implies, allows everyone full access to everything. Naturally, when one starts assigning specific security to partitions, folders and files, this is the first thing to be revised. A very common error is to then add specific permissions while neglecting to add "System – Full Control" and "Administrators – Full Control". In almost all cases, these are both necessary for the system to function properly. Without "System – Full Control", many system applications such as anti-virus programs and defragmenters cannot access all of the files. Without "Administrators – Full Control", the system administrators must be members of many specific groups in order to properly manage the system.

### Excess Groups

You may think it's no big deal to make administrators members of all groups, but there are several drawbacks. First, there is the extra work of adding the group membership to each administrator's

# NTFS SECURITY & SECURITY PITFALLS

By Mark E. Donaldson

account. Second, there is the danger of missing one; if you attempt some global action such as defragmenting, the action will not be performed on any files which belong to a group which the administrator is not a member of, and you will not know those files were skipped. Third, if you add too many group memberships, you may run into buffer overflows, where in some circumstances only the first 256 or 512 characters of the list of groups is used.

There is a fourth drawback that isn't very obvious: If a given group, such as Users, is specifically denied access to a file or folder, then the administrator will be denied access if he is a member of that group. Yes, even if you have "Administrators – Full Control" set. This is because Windows NT and 2000 first search the Access Control List (ACL) for ACEs that deny access, and they stop the search as soon as they find one ACE that fits the user.

Adding "Administrators – Full Control" to every file and folder on the system is a simple and foolproof action. If this is done, there is no need to add any other group membership to any administrator's account. The only situation I have come across where this is not acceptable is in certain financial institutions; they recognize that you cannot prevent an administrator from accessing any file, but they can require setting an ACE, which gets recorded in the security log.

Also note that each Domain.Admin group is by default a member of each local administrator group.

## Editing Permissions

Most people change existing permissions by using NT Explorer. You right click the folder or file, then click Properties, Security, Permissions. From this window you can make any change you wish, if you are an administrator. ***The error here, when modifying a folder, is to check the "Replace Permissions on Subdirectories" box or to fail to uncheck the "Replace Permissions on Existing Files" box.*** If these boxes are checked, the permissions as they appear in the window are copied to all existing files and folders below the one you are in, erasing whatever permissions were already there. In other words, any security settings in existence at the time is lost, overwritten by the new settings, and you have to start over.

This is fine if that is what you want, but there is also a way to make specific changes without altering the existing permissions: Use the CACLS program.

## Using CALCS

If you want to add or remove permissions without destroying the existing ones, you need to use the CACLS command line interface executed from the root of the partition:

```
D:\>cacls/?
```

Displays or modifies access control lists (ACLs) of files:

```
CACLS filename [/T] [/E] [/C] [/G user:perm] [/R user [...]]  
[ /P user:perm [...]] [/D user [...]]
```

filename	Displays ACLs.
/T	Changes ACLs of specified files in the current directory & subdirectories.
/E	Edit ACL instead of replacing it.
/C	Continue on access denied errors.
/G user:perm	Grant specified user access rights.
Perm can be:	R Read
	C Change (write)
	F Full control

# NTFS SECURITY & SECURITY PITFALLS

By Mark E. Donaldson

/R user                    Revoke specified user's access rights (only valid with /E).  
/P user:perm              Replace specified user's access rights.  
                            Perm can be:    N None  
    R Read  
    C Change (write)  
    F Full control  
/D user                    Deny specified user access.

Wildcards can be used to specify more than one file in a command. You can specify more than one user in a command.

SPECIAL NOTE: You must be at the root directory level of the partition in question. Use this command:

## CD drive\_letter:

Now the command line:

## CACLS \* /e /t /g SYSTEM:F

SPECIAL NOTE: If you see this message: "Unable to perform a security operation on an object which has no associated security" you are executing this from a FAT partition. ACLs are only used on NTFS partitions.

The /e switch tells the CACLS command to EDIT the ACLs rather than REPLACE the existing permissions, and the /t switch tells it to apply the edit to subdirectories. Any number of ACCOUNT:PERM sets may follow the GRANT (/g) switch. As you can see from the above listing, there is additional flexibility built into the CACLS command - its only limitation is the extent of selections for PERM values.

You may also need to add SYSTEM to the drive itself. Do that through Explorer with these steps:

- Start EXPLORER
- Right click the partition in question
- Click PROPERTIES
- Click the SECURITY tab
- Click the PERMISSIONS button
- If SYSTEM is not listed, click ADD and select SYSTEM
- Highlight SYSTEM
- Set TYPE OF ACCESS to FULL CONTROL
- **Clear the REPLACE PERMISSIONS ON EXISTING FILES check box**
- Click OK

## Summary

When setting permissions, keep it simple. There is no advantage to using complex security schemes, especially since they may leave open security holes simply because the scheme is too complex to grasp as a whole.