Access Control Lists in the Security Descriptor

An ACL is an ordered list of ACEs (Access Control Entries) that define the access attributes that apply to an object and its properties. Each ACE identifies a security principal (user or group account) and specifies a set of access rights that are allowed, denied, or audited for that security principal.

The security descriptor for an object can contain two ACLs:

- A DACL that identifies the users and groups that are allowed or denied access
- An SACL that controls how access is audited

The following table shows the data structure for an ACL:

ACL data structure

ACE Field	Description
ACL Size	Number of bytes, allocated for the ACL. This includes the ACL header, ACEs, and remaining free space in the buffer. The size of an ACL varies with the number and size of its ACEs. The maximum size of an ACL is 64K, or approximately 1,820 ACEs, depending on the size of the ACEs. However, for performance reasons, the maximum size is not practical.
ACL revision	Revision number for the ACL's data structure. The structure of an ACL is the same for all revisions, but the structure of ACEs in the ACL can vary
ACE count	Number of ACEs in the ACL. A value of zero means the ACL has no ACEs — it is empty; therefore, access-checking can stop. Note that an empty DACL differs from a NULL DACL. An empty DACL gives no access to anyone, but a NULL DACL gives unconditional access to everyone, and should therefore be avoided.
ACE[1]	An ordered list containing zero or more ACEs.During an access check, ACEs are processed in the order in which they are listed.
ACE[...]	An ordered list containing zero or more ACEs.During an access check, ACEs are processed in the order in which they are listed.

Previous | NTFS Permissions | Next

- NTFS Permissions
  - Setting Permissions
  - File and Folder Basic Permissions
  - File and Folder Advanced Permissions
  - Effective Permissions
  - Changing Ownership of Files and Folders
  - Moving and Copying Protected Files
  - Troubleshooting Access to Files and Shared Folders
  - Permissions for Other Objects
  - User Rights vs. NTFS Permissions
  - Share Permissions vs. NTFS Permissions
  - Explicit vs. Inherited Permissions
  - Allow vs. Deny Permissions
  - Permission Precedence
  - Combining Shared Folder Permissions and NTFS Permissions
  - Sharing and Adding Permissions
  - Backing up and Restoring NTFS Permissions on a Specified Volume
  - Off-line Access to Shared Folders (Caching)
- Metafile $Secure
  - A Brief History of NTFS
  - Structure of $Secure File
  - NTFS Security Descriptor
  - ✓ Access Control Lists in the Security Descriptor
  - How the System Uses ACLs
  - Access Control Entries
  - Security Identifier
- Appendix. Script to Backup or Restore

NTFS Permissions

- Glossary

Data Recovery

UNFORMAT
Active@ UNERASER
Active@ UNDELETE
Active@ File Recovery
Active@ Partition Recovery
Active@ Password Changer

Disk Utilities

Active@ Boot Disk(Live CD)
Active@ Partition Manager
Active@ Hard Disk Monitor
NTFS Reader for DOS
Active@ Disk Editor
NTFS Recovery toolkit

Data Security

Active@ KillDisk
Active@ ERASER
Active@ ZDelete
Active@ ZDelete Network

Data Backup

Active@ Disk Image

Data CD/DVD

Active@ ISO File Manager
Active@ ISO Burner
Active@ Data Burner
Active@ DVD Eraser

Be with us Follow us Look for us Contact us

NTFS.com by Active@ Data Recovery Software ©1998-2016