**Access control list**

From Wikipedia, the free encyclopedia

This article is about the computer permissions list. For the ligament, see [Anterior cruciate ligament](https://en.wikipedia.org/wiki/Anterior_cruciate_ligament).

An **access control list** (**ACL**), with respect to a [computer](https://en.wikipedia.org/wiki/Computer) [file system](https://en.wikipedia.org/wiki/File_system), is a list of [permissions](https://en.wikipedia.org/wiki/File_system_permissions) attached to an [object](https://en.wikipedia.org/wiki/Computer_file). An ACL specifies which users or system processes are granted access to objects, as well as what operations are allowed on given objects.[[1]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-1) Each entry in a typical ACL specifies a subject and an operation. For instance, if a file object has an ACL that contains (Alice: read,write; Bob: read), this would give Alice permission to read and write the file and Bob to only read it.

**Contents**

* [1 Implementations](https://en.wikipedia.org/wiki/Access_control_list#Implementations)
  + [1.1 Filesystem ACLs](https://en.wikipedia.org/wiki/Access_control_list#Filesystem_ACLs)
  + [1.2 Networking ACLs](https://en.wikipedia.org/wiki/Access_control_list#Networking_ACLs)
  + [1.3 SQL implementations](https://en.wikipedia.org/wiki/Access_control_list#SQL_implementations)
* [2 Comparing with RBAC](https://en.wikipedia.org/wiki/Access_control_list#Comparing_with_RBAC)
* [3 See also](https://en.wikipedia.org/wiki/Access_control_list#See_also)
* [4 References](https://en.wikipedia.org/wiki/Access_control_list#References)
* [5 Further reading](https://en.wikipedia.org/wiki/Access_control_list#Further_reading)

**Implementations**

Many kinds of systems implement ACLs, or have a historical implementation.

**Filesystem ACLs**

A [filesystem](https://en.wikipedia.org/wiki/Filesystem) ACL is a data structure (usually a table) containing entries that specify individual user or group rights to specific system objects such as programs, processes, or files. These entries are known as access-control entries (ACEs) in the [Microsoft Windows NT](https://en.wikipedia.org/wiki/Microsoft_Windows_NT),[[2]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-2) [OpenVMS](https://en.wikipedia.org/wiki/OpenVMS), [Unix-like](https://en.wikipedia.org/wiki/Unix-like), and [Mac OS X](https://en.wikipedia.org/wiki/Mac_OS_X) [operating systems](https://en.wikipedia.org/wiki/Operating_system). Each accessible object contains an identifier to its ACL. The privileges or permissions determine specific access rights, such as whether a user can read from, write to, or [execute](https://en.wikipedia.org/wiki/Execution_(computing)) an object. In some implementations, an ACE can control whether or not a user, or group of users, may alter the ACL on an object.

Most of the Unix and Unix-like operating systems (e.g. [Linux](https://en.wikipedia.org/wiki/Linux),[[3]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-3) [BSD](https://en.wikipedia.org/wiki/BSD), or [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system))) support POSIX.1e ACLs, based on an early [POSIX](https://en.wikipedia.org/wiki/POSIX) draft that was withdrawn in 1997. Many of them, for example [AIX](https://en.wikipedia.org/wiki/AIX), [FreeBSD](https://en.wikipedia.org/wiki/FreeBSD),[[4]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-4) [Mac OS X](https://en.wikipedia.org/wiki/Mac_OS_X) beginning with version 10.4 ("[Tiger](https://en.wikipedia.org/wiki/Mac_OS_X_Tiger)"), or [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)) with [ZFS](https://en.wikipedia.org/wiki/ZFS) filesystem,[[5]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-5) support [NFSv4](https://en.wikipedia.org/wiki/NFSv4) ACLs, which are part of the NFSv4 standard. There are two experimental implementations of NFSv4 ACLs for Linux: NFSv4 ACLs support for [Ext3](https://en.wikipedia.org/wiki/Ext3) filesystem[[6]](https://en.wikipedia.org/wiki/Access_control_list" \l "cite_note-6) and the more recent [Richacls](https://en.wikipedia.org/wiki/Richacls),[[7]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-7) which brings NFSv4 ACLs support for [Ext4](https://en.wikipedia.org/wiki/Ext4) filesystem.

[PRIMOS](https://en.wikipedia.org/wiki/PRIMOS) featured ACLs at least as early as 1984.[[8]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-8)

In the 1990s the ACL and [RBAC](https://en.wikipedia.org/wiki/Role-based_access_control) models were extensively tested[[*by whom?*](https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Words_to_watch#Unsupported_attributions)] and used to administer file permissions.

**Networking ACLs**

On some types of proprietary computer-hardware (in particular [routers](https://en.wikipedia.org/wiki/Router_(computing)) and [switches](https://en.wikipedia.org/wiki/Network_switch)), an access control list provides rules that are applied to [port numbers](https://en.wikipedia.org/wiki/TCP_and_UDP_port) or [IP addresses](https://en.wikipedia.org/wiki/Ip_address) that are available on a [host](https://en.wikipedia.org/wiki/Server_(computing)) or other [layer 3](https://en.wikipedia.org/wiki/Network_Layer), each with a list of hosts and/or networks permitted to use the service. Although it is additionally possible to configure access control lists based on network domain names, this is a questionable idea because individual [TCP](https://en.wikipedia.org/wiki/Transmission_Control_Protocol), [UDP](https://en.wikipedia.org/wiki/User_Datagram_Protocol), and [ICMP](https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol) headers do not contain domain names. Consequently, the device enforcing the access control list must separately [resolve names](https://en.wikipedia.org/wiki/Name_resolution_(computer_systems)) to numeric addresses. This presents an additional attack surface for an attacker who is seeking to compromise security of the system which the access control list is protecting. Both individual [servers](https://en.wikipedia.org/wiki/Server_(computing)) as well as [routers](https://en.wikipedia.org/wiki/Router_(computing)) can have network ACLs. Access control lists can generally be configured to control both inbound and outbound traffic, and in this context they are similar to [firewalls](https://en.wikipedia.org/wiki/Firewall_(networking)). Like firewalls, ACLs could be subject to security regulations and standards such as [PCI DSS](https://en.wikipedia.org/wiki/PCI_DSS).

**SQL implementations**

ACL algorithms have been ported to [SQL](https://en.wikipedia.org/wiki/SQL) and to [relational database systems](https://en.wikipedia.org/wiki/Relational_database_management_system). Many "modern" (2000s and 2010s) [SQL](https://en.wikipedia.org/wiki/SQL)-based systems, like [enterprise resource planning](https://en.wikipedia.org/wiki/Enterprise_resource_planning) and [content management](https://en.wikipedia.org/wiki/Content_management_system) systems, have used ACL models in their administration modules.

**Comparing with RBAC**

The main alternative to the ACL model is the [role-based access control](https://en.wikipedia.org/wiki/Role-based_access_control) (RBAC) model. A "minimal RBAC model", *RBACm*, can be compared with an ACL mechanism, *ACLg*, where only groups are permitted as entries in the ACL. Barkley (1997)[[9]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-9) showed that *RBACm* and *ACLg* are equivalent.

In modern SQL implementations, ACL also manage groups and inheritance in a hierarchy of groups. So "modern ACLs" can express all that RBAC express, and are notably powerful (compared to "old ACLs") in their ability to express access control policy in terms of the way in which administrators view organizations.

For data interchange, and for "high level comparisons", ACL data can be translated to [XACML](https://en.wikipedia.org/wiki/XACML).[[10]](https://en.wikipedia.org/wiki/Access_control_list#cite_note-10)