

Security

Built-in Slide ACL Implementation

A short introduction (copied, more or less unchanged, from the mailing list) on how Webdav ACLs work:

The ACL spec doesn't say anything about the relative 'strengths' of rights, the only thing that matters is the order within the ACL.

The way it works is this:

- 1) you want to figure out if a user can perform some action (for example: writing to resource X)
- 2) You look at each ACE on resource X *in order*. The order of evaluation is: direct permissions on the resource (in their explicit ordering within the ACL), then inherited permissions from the parent, then inherited permissions from the grandparent, etc.
- 3) If this permission allows you to perform the appropriate action, then the access control checks pass. Don't look at the rest of the ACEs
- 4) If this permission *denies* the appropriate permission, the access control check fails. Don't look at the rest of the ACEs.
- 5) If this permission does neither, continue to the next ACE. This is common - a 'read' permission, for example, doesn't allow writes, but doesn't deny them either.

So the default behaviour (I think you can plug in alternative implementations, by the way, to answer your original question - but that would likely make it incompatible with the ACL spec) will be that, in your example, things 'just work'. However, this depends on exactly how you've set up your permissions.

Specifically, you said "user has read-only access as a member of one group". There are two ways you could set that up. One would be to say "this group has read access" (and say nothing at all about write access!), the other would be to say "this group has read access AND this group explicitly does not have write access", using two ACEs (a grant and a deny). This latter form would not do what you want, so you should avoid it.

JAAS

See <http://jakarta.apache.org/slide/howto-jaas.html>

Also, if you wish to use an external mechanism for authentication and authorization, you may also want to look at [AutoCreateUser](#).

Custom