How do permissions work in AFS?

How do permissions work in AFS?

On this page:

Overview
Displaying current permissions
Setting permissions
Inheritance
A note about `system:anyuser` permissions
Special groups: system:htaccess.mit
More information

Just need to share files or change permissions?
This document is intended to provide detailed information about how permissions work in AFS. It is recommended reading for new Athena users, or for anyone taking over maintenance of a group or department's locker. However, if you are in a hurry, you may wish to view one of the following instead:

- How do I change permissions in AFS?
- How can I easily share files with other users on Athena?

Overview

Permissions to access files and directories are set in the form of access control lists, or acl's. Unlike NFS, where individual files may be readable and writable by everyone, a group of people, or only you, AFS only allows you to set permissions for entire directories at a time. However, these permissions may be set for any number of people or groups, and each one may be completely different.

There are seven types of permissions that can be given to users for a directory:

Right	Enables users to:
r	read the contents of files in the directory
l	list the names of files in the directory
i	insert files into the directory
d	delete files from the directory
w	write (or modify) files into the directory
k	lock (or modify the write-mode bit) of files in the directory
a	administer or change the acl of the directory

As owner of the directory, administrative (a) access is implied. Other than that, no rights above imply any others; for example, a user with "write" permission is not automatically given "insert" permission as well.

Displaying current permissions

Acl's are examined and modified using the "fs" command. To determine what permissions a directory has, type the command:

```
joeuser@athena:~$ fs la directory
```

Note: Be sure to fill in the correct information for *directory*.

at your prompt, where "directory" is the name of the directory to be examined. (To see the current directory, this can be left out.) In Joeuser's

sample directory, for example, this will return an acl that might look something like this:

```
joeuser@athena:~$ fs la sample


Access list for sample is
Normal rights:
  system:anyuser rl
  system:expunge ld
  joeuser rlidwka
  janeuser rliw
```

The list contains pairs of users or groups, and their respective permissions on that directory. In this example, because joeuser is the owner of the directory, he has all permission to do anything to the directory. Janeuser has permission to read, lookup, insert, and write files. "system:anyuser" corresponds to anyone, so anybody can read and lookup files. "system:expunge" has "list" and "delete" permissions. This is the automated expunger, which purges files marked for deletion with the "delete" command.

Setting permissions

To set permissions, use the "sa" argument with "fs" in the following format:

```
joeuser@athena:~$ fs sa directory user permissions
```

Note: Be sure to fill in the correct information for *directory* and *user* and *permissions*.

or

```
joeuser@athena:~$ fs sa directory system:group permissions
```

Note: Be sure to fill in the correct information for *directory* and *group* and *permissions*.

"directory" corresponds to the directory for which rights are being set, the "user" or "group" means the person or group for whom you are setting these rights, and "permissions" is the list of rights to be given. For example, in order for Joeuser to give Jimuser access to read, lookup, and delete files in his sample directory, he would type:

```
joeuser@athena:~$ fs sa sample jimuser rld
```

To set permissions on the current directory, you can use a single dot (.) in place of the directory name.

Groups that are prefixed with system: correspond to the moira list of the same name, as long as the moira list has the group bit set.

There are also common aliases for groups of rights, that can be referred to as follows:

Alias	Expands to
read	rl
write	rlidwk
all	rlidwka
none	nothing (used to remove access)

You are encouraged to use the aliases whenever possible to avoid problems. For example, if you intend to give someone permission to write files in your directory, and give them "rlw" permissions, instead of "rlidwk", they will only be able to edit existing files. They will not be able to create new files (including emacs tilde (~) files), nor will they be able to use version control systems (such as RCS) that rely on changing the "write" mode bit of a file.

Inheritance

When you create a new directory in AFS, it inherits the permissions of its parent directory. However, when you change the permissions on a directory of AFS, it does **not** automatically change permissions for all subdirectories. For more information on changing AFS permissions on all subdirectories of a directory, see How do I change AFS permissions for all subdirectories of a directory (recursively)?

A note about `system:anyuser` permissions

In order for a directory to be viewable over the web, all parent directories must have "list" permissions for system:anyuser.

You should **NEVER** assign "write" permissions to `system:anyuser`, because your directory will immediately be abused by spammers and others with malicious intent, and you will likely lose data. If you want everyone at MIT to have permission to edit files in a certain directory, you may assign "write" permissions to `system:authuser` instead, which will restrict permissions to users who have Athena accounts. However, the best choice is always to assign permissions only to those who need them.

Special groups: system:htaccess.mit

Note that system:htaccess.mit works like system:anyuser as far as the web.mit.edu and www.mit.edu servers are concerned - the web server can't tell the difference between that acl and something that it sees due to it being system:anyuser acl'd.

More information

For more details about fs, you can type "fs help" or "fs help <topic>" at the athena% prompt.

Also, please see our other answers:

- How do I change AFS permissions for all subdirectories of a directory (recursively)?
- How can I easily share files with other users on Athena?
- How do I change permissions in AFS?