Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'setfacl.1'

$ man setfacl.1

SETFACL(1)   Access Control Lists   SETFACL(1)

NAME

setfacl - set file access control lists

SYNOPSIS

setfacl [-bkndRLPvh] [{-m|-x} acl_spec] [{-M|-X} acl_file] file ...

setfacl --restore=file

DESCRIPTION

This utility sets Access Control Lists (ACLs) of files and directories.

On the command line, a sequence of commands is followed by a sequence of files (which in turn can be followed by another sequence of com? mands, ...).

The -m and -x options expect an ACL on the command line. Multiple ACL entries are separated by comma characters (`,'). The -M and -X options read an ACL from a file or from standard input. The ACL entry format is described in Section ACL ENTRIES.

The --set and --set-file options set the ACL of a file or a directory. The previous ACL is replaced. ACL entries for this operation must in? clude permissions.

The -m (--modify) and -M (--modify-file) options modify the ACL of a file or directory. ACL entries for this operation must include permissions.

The -x (--remove) and -X (--remove-file) options remove ACL entries. It is not an error to remove an entry which does not exist. Only ACL entries without the perms field are accepted as parameters, unless POSIXLY_CORRECT is defined.

When reading from files using the -M and -X options, setfacl accepts the output getfacl produces. There is at most one ACL entry per line. After a Pound sign (`#'), everything up to the end of the line is treated as a comment.

If setfacl is used on a file system which does not support ACLs, setfacl operates on the file mode permission bits. If the ACL does not fit completely in the permission bits, setfacl modifies the file mode permission bits to reflect the ACL as closely as possible, writes an error message to standard error, and returns with an exit status greater than 0.

PERMISSIONS

The file owner and processes capable of CAP_FOWNER are granted the right to modify ACLs of a file. This is analogous to the permissions required for accessing the file mode. (On current Linux systems, root is the only user with the CAP_FOWNER capability.)

OPTIONS

-b, --remove-all

Remove all extended ACL entries. The base ACL entries of the owner, group and others are retained.

-k, --remove-default

Remove the Default ACL. If no Default ACL exists, no warnings are issued.

-n, --no-mask

Do not recalculate the effective rights mask. The default behavior of setfacl is to recalculate the ACL mask entry, unless a mask entry was explicitly given. The mask entry is set to the union of

all permissions of the owning group, and all named user and group
entries. (These are exactly the entries affected by the mask en?
try).

--mask

Do recalculate the effective rights mask, even if an ACL mask entry
was explicitly given. (See the -n option.)

-d, --default

All operations apply to the Default ACL. Regular ACL entries in the
input set are promoted to Default ACL entries. Default ACL entries
in the input set are discarded. (A warning is issued if that hap?
pens).

--restore=file

Restore a permission backup created by `getfacl -R' or similar. All
permissions of a complete directory subtree are restored using this
mechanism. If the input contains owner comments or group comments,
setfacl attempts to restore the owner and owning group. If the in?
put contains flags comments (which define the setuid, setgid, and
sticky bits), setfacl sets those three bits accordingly; otherwise,
it clears them. This option cannot be mixed with other options ex?
cept `--test'.

--test

Test mode. Instead of changing the ACLs of any files, the resulting
ACLs are listed.

-R, --recursive

Apply operations to all files and directories recursively. This op?
tion cannot be mixed with `--restore'.

-L, --logical

Logical walk, follow symbolic links to directories. The default be?
havior is to follow symbolic link arguments, and skip symbolic
links encountered in subdirectories. Only effective in combination
with -R. This option cannot be mixed with `--restore'.

-P, --physical

Physical walk, do not follow symbolic links to directories.   This also skips symbolic link arguments. Only effective in combination with -R. This option cannot be mixed with `--restore'.

-v, --version

Print the version of setfacl and exit.

-h, --help

Print help explaining the command line options.

-- End of command line options. All remaining parameters are inter?preted as file names, even if they start with a dash.

- If the file name parameter is a single dash, setfacl reads a list of files from standard input.

ACL ENTRIES

The setfacl utility recognizes the following ACL entry formats (blanks inserted for clarity):

[d[efault]:] [u[ser]:]uid [:perms]

Permissions of a named user. Permissions of the file owner if uid is empty.

[d[efault]:] g[roup]:gid [:perms]

Permissions of a named group. Permissions of the owning group if gid is empty.

[d[efault]:] m[ask][:] [:perms]

Effective rights mask

[d[efault]:] o[ther][:] [:perms]

Permissions of others.

Whitespace between delimiter characters and non-delimiter characters is ignored.

Proper ACL entries including permissions are used in modify and set op?erations. (options -m, -M, --set and --set-file). Entries without the perms field are used for deletion of entries (options -x and -X).

For uid and gid you can specify either a name or a number. Character literals may be specified with a backslash followed by the 3-digit oc?tal digits corresponding to the ASCII code for the character (e.g., \101 for 'A'). If the name contains a literal backslash followed by 3 digits, the backslash must be escaped (i.e., \\).

The perms field is a combination of characters that indicate the read (r), write (w), execute (x) permissions. Dash characters in the perms field (-) are ignored. The character X stands for the execute permis? sion if the file is a directory or already has execute permission for some user. Alternatively, the perms field can define the permissions numerically, as a bit-wise combination of read (4), write (2), and exe? cute (1). Zero perms fields or perms fields that only consist of dashes indicate no permissions.

AUTOMATICALLY CREATED ENTRIES

Initially, files and directories contain only the three base ACL en? tries for the owner, the group, and others. There are some rules that need to be satisfied in order for an ACL to be valid:

- The three base entries cannot be removed. There must be exactly one entry of each of these base entry types.
- Whenever an ACL contains named user entries or named group objects, it must also contain an effective rights mask.
- Whenever an ACL contains any Default ACL entries, the three Default ACL base entries (default owner, default group, and default others) must also exist.
- Whenever a Default ACL contains named user entries or named group objects, it must also contain a default effective rights mask.

To help the user ensure these rules, setfacl creates entries from ex? isting entries under the following conditions:

- If an ACL contains named user or named group entries, and no mask entry exists, a mask entry containing the same permissions as the group entry is created. Unless the -n option is given, the permis? sions of the mask entry are further adjusted to include the union of all permissions affected by the mask entry. (See the -n option description).
- If a Default ACL entry is created, and the Default ACL contains no owner, owning group, or others entry, a copy of the ACL owner, own? ing group, or others entry is added to the Default ACL.
- If a Default ACL contains named user entries or named group en?

tries, and no mask entry exists, a mask entry containing the same
permissions as the default Default ACL's group entry is added. Un?
less the -n option is given, the permissions of the mask entry are
further adjusted to include the union of all permissions affected
by the mask entry. (See the -n option description).

EXAMPLES

Granting an additional user read access

```
setfacl -m u:lisa:r file
```

Revoking write access from all groups and all named users (using the
effective rights mask)

```
setfacl -m m::rx file
```

Removing a named group entry from a file's ACL

```
setfacl -x g:staff file
```

Copying the ACL of one file to another

```
getfacl file1 | setfacl --set-file=- file2
```

Copying the access ACL into the Default ACL

```
getfacl --access dir | setfacl -d -M- dir
```

CONFORMANCE TO POSIX 1003.1e DRAFT STANDARD 17

If the environment variable POSIXLY_CORRECT is defined, the default be?
havior of setfacl changes as follows: All non-standard options are dis?
abled. The ``default:'' prefix is disabled. The -x and -X options
also accept permission fields (and ignore them).

AUTHOR

Andreas Gruenbacher, <andreas.gruenbacher@gmail.com>.

Please send your bug reports, suggested features and comments to the
above address.

SEE ALSO

getfacl(1), chmod(1), umask(1), acl(5)

May 2000   ACL File Utilities   SETFACL(1)