

File and Directory Permissions Explained

What You Will Learn

- Symbolic permissions
- Numeric / octal permissions
- File versus directory permissions
- Changing permissions
- Working with groups
- File creation mask

Permissions

```
$ ls -l
```

```
-rw-rw-r-- 1 jason users 10400 Sep 27 08:52 sales.data
```

Symbol

Type

-

Regular file

d

Directory

l

Symbolic link

Permissions

```
$ ls -l
```

```
-rw-rw-r-- 1 jason users 10400 Sep 27 08:52 sales.data
```

Symbol

Permission

r

Read

w

Write

x

Execute

Permissions - Files vs Directories

Permission

File

Directory

Read (r)

Allows a file to be read.

Allows file names in the directory to be read.

Write (w)

Allows a file to be modified.

Allows entries to be modified within the directory.

Execute (x)

Allows the execution of a file.

Allows access to contents and metadata for entries.

Permission Categories

Symbol	Category
u	User
g	Group
o	Other
a	All

Groups

- Every user is in at least one group.
- Users can belong to many groups.
- Groups are used to organize users.
- The `groups` command displays a user's groups.
- You can also use `id -Gn`.

```
[jason@linuxsvr ~]$ groups
```

```
jason sales
```

```
[jason@linuxsvr ~]$ id -Gn
```

```
jason sales
```

```
[jason@linuxsvr ~]$ groups pat
```

```
pat : finance newyork
```

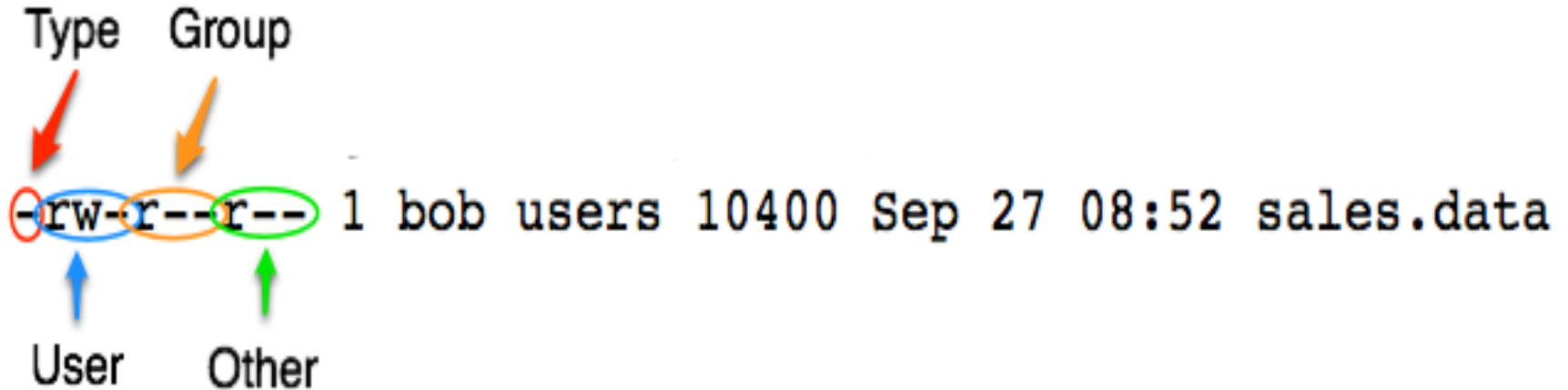
```
[jason@linuxsvr ~]$ groups adminuser
```

```
adminuser : adminuser
```

```
[jason@linuxsvr ~]$
```


Secret Decoder Ring

Type Group

The diagram shows a file permission string '-rw-r--r--' with several annotations. A red arrow points to the first character '-' with the label 'Type' above it. A blue oval encircles the 'rw' part, with a blue arrow pointing to it from the label 'User' below. An orange arrow points to the first 'r' of the second triplet 'r--' with the label 'Group' above it. An orange oval encircles this 'r'. A green arrow points to the first 'r' of the third triplet 'r--' with the label 'Other' below. A green oval encircles this 'r'. The full permission string is followed by the file details: '1 bob users 10400 Sep 27 08:52 sales.data'.

`-rw-r--r-- 1 bob users 10400 Sep 27 08:52 sales.data`

User Other

Changing Permissions

Item	Meaning
chmod	Change mode command
ugo	User category user, group, other, all
+-=	Add, subtract, or set permissions
rwX	Read, Write, Execute

Numeric Based Permissions

r	w	x	
0	0	0	Value for off
1	1	1	Binary value for on
4	2	1	Base 10 value for on

Octal	Binary	String	Description
0	0	---	No permissions
1	1	--x	Execute only
2	10	-w-	Write only
3	11	-wx	Write and execute (2+1)
4	100	r--	Read only
5	101	r-x	Read and execute (4+1)
6	110	rw-	Read and write (4+2)
7	111	rwX	Read, write, and execute (4+2+1)

Order Has Meaning

	U	G	O
Symbolic	rwX	r-x	r--
Binary	111	101	100
Decimal	7	5	4

Commonly Used Permissions

Symbolic	Octal
----------	-------

<code>-rwx-----</code>	700
------------------------	-----

<code>-rwxr-xr-x</code>	755
-------------------------	-----

<code>-rw-rw-r--</code>	664
-------------------------	-----

<code>-rw-rw----</code>	660
-------------------------	-----

<code>-rw-r--r--</code>	644
-------------------------	-----

Working with Groups

- New files belong to your primary group.
- The `chgrp` command changes the group.

Directory Permissions Revisited

- Permissions on a directory can effect the files in the directory.
- If the file permissions look correct, start checking directory permissions.
- Work your way up to the root.

File Creation Mask

- File creation mask determines default permissions.
- If no mask were used permissions would be:
 - 777 for directories
 - 666 for files

The `umask` Command

```
umask [-S] [mode]
```

- Sets the file creation mask to mode, if given.
- Use -S to for symbolic notation.

	Directory	File
Base Permission	777	666
Subtract Umask	-022	-022
Creations Permission	755	644

	Directory	File
Base Permission	777	666
Subtract Umask	-002	-002
Creations Permission	775	664

Octal Subtraction Is an Estimation

	Directory	File
Base Permission	777	666
Subtract Umask	-007	-007
Creations Permission	770	660 *

Common umask modes

- 022
- 002
- 077
- 007

Octal	Binary	Dir Perms	File Perms
0	0	rwX	rw-
1	1	rw-	rw-
2	10	r-X	r--
3	11	r--	r--
4	100	-WX	-W-
5	101	-W-	-W-
6	110	--X	---
7	111	---	---

Special Modes

- `umask 0022` is the same as `umask 022`
- `chmod 0644` is the same as `chmod 644`
- The special modes are:
 - `setuid`
 - `setgid`
 - `sticky`

Summary

- Symbolic permissions
- Numeric / octal permissions
- File versus directory permissions
- Changing permissions
- Working with groups
- File creation mask