Managing Special Permissions

Andrew Mallett
Linux Author and Trainer

@theurbanpenguin www.theurbanpenguin.com

Examining the File Mode

Special Permissions

User Permissions

Group Permissions

Others Permissions

Overview

In this module we introduce:

- Special permissions, the 4th block
- Sticky bit to control deletions
- SGID bit to control group ownership
- SUID and SGID on executables
- Linux capabilities

Standard Permissions

Managing the File Mode

Andrew Mallett

```
$ stat -c %a /etc/hosts
0644
```

12 Bit Permissions

The Linux file mode has 12 bits that are used for permissions, 4 blocks of three bits starting with special, user, group and others

Special Permission

The first block of three bits are the special permissions

4

SUID

Used on programs to run as the user owner during execution

2

SGID

On directories, new files are assigned the group owner from the directory

1

Sticky Bit

With this set users can only delete files they own from shared directories

```
$ mkdir -p ~/perms/dir{1..4}

$ chmod -v 1777 ~/perms/dir1 # Sticky bit set

$ chmod -v 2777 ~/perms/dir2 # SGID bit set

$ chmod -v 3777 ~/perms/dir3 # Both the sticky bit and SGID bit set

$ chmod -v 1770 ~/perms/dir4 # Sticky bit is set but no permissions to others

$ ls -l ~/perms
```

Setting Special Directory Permissions

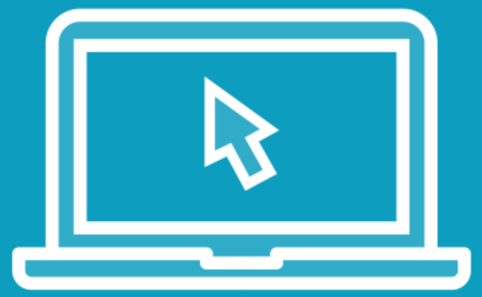
It is useful to be able to list files with certain permission set and helps us with security audit. For the moment we will become used to setting the permissions and list the top-level directory

```
$ find ~/perms/ -type d -perm /g=s,o=t # List dirs where either SGID or Sticky bit set

$ find ~/perms/ -type d -perm -g=s,o=t # List dirs where both SGID and Sticky bit set

$ find ~/perms/ -type d -perm /o=t # List dirs where Sticky bit set

$ find ~/perms/ -type d -perm /o=tw # List dirs where Sticky bit set or world writable
```

Finding Special Directory Permissions

The find command in Linux is very flexible and practice is always useful

Demo



Using the ls command we can list permissions on directories that include special permissions

- ls -l can list permissions
- locating files with special permissions using find

Collaboration Directory Permissions

Sticky Bit:

Users will need the write permission to the directory to add content. This also means that they can delete files. Setting the sticky bit on directory allows them to only delete the files they own

SGID Bit:

Normally, a new file will be group owned by the user's current primary group (gid). Where access to others is restricted, perhaps via the umask, it is preferably that all new files in a directory are group owned by the group specified in the directory. The SGID bit makes this happen

Demo

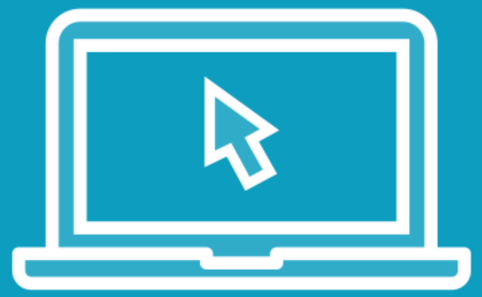
First, we demonstrate why we need the sticky bit and see how it is set on our shared directory

- The sticky-bit prevents users deleting files they do not own

Demo

For this demonstration, we will demonstrate the SGID bit on the directory and observe the difference

Demo



Practical use of the umask and SGID permissions for and Apache Web Server deployment

Special Permissions on Executables

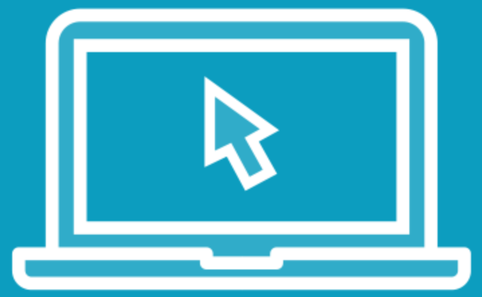
SUID:

Where the SUID bit is set on and executable, the program runs in the execution context of the user owner of the file. In this way, the /usr/bin/passwd file can access the /etc/shadow file as root

SGID Bit:

Similar to the SUID bit, the execution group context takes on the group owner of the file. It is normal that the tty a user is connected to has write permissions granted to the tty group. The write / bsd-write program will habe the SGID bit sit and owned by tty group

Demo



Investigating SUID/SGID permissions o executables

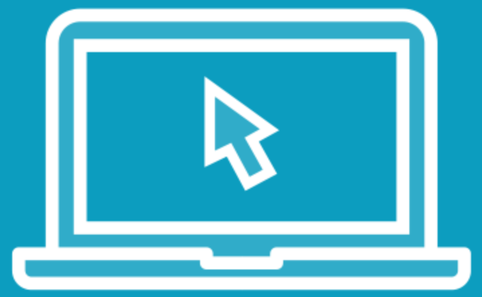
Linux Capabilities - Avoid Special Permissions

man 7 capabilities:
Check the available capabilities

getcap:
Read capabilities from a file

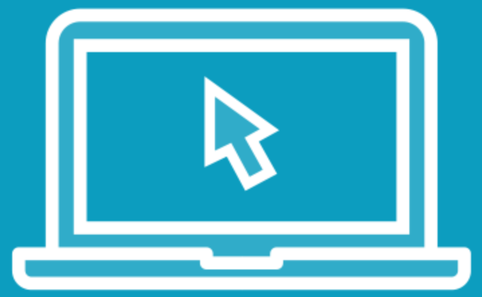
setcap:
Set capabilities on a file

Demo



Investigating Linux capabilities

Demo



Investigating Linux capabilities

Summary

Sticky Bit: o+t : Ensure users can delete only their own files

SGID: g+s : On a directory, ensures new files are created with the group ownership of the directory

SUID/SGID: When set on executables they change the execution context

Capabilities: Adding a capability can allow the required activity but only the required activity

Extending the File Mode with ACLS