

第三集：文件权限与管理

课程目标

本课程将教你如何使用 Bash 命令管理文件的权限与所有权。通过学习本节的命令，你将能够更好地控制文件的安全性，并有效管理系统资源。

1. chmod：更改文件权限

- **功能：**用于修改文件的读（r）、写（w）、执行（x）权限。
- **数字表示法：** rwx 权限可以用数字表示，例如 755 表示所有者有读、写、执行权限，组和其他用户有读、执行权限。

示例：

- 将文件 `script.sh` 设置为所有者可读写执行，组和其他用户可读可执行：

```
chmod 755 script.sh
```

- 授予 `file.txt` 只读权限给所有用户：

```
chmod 444 file.txt
```

script.sh 文件示例：

```
#!/bin/bash
# 这是一个简单的 Bash 脚本

echo "Hello, World!"
echo "当前用户是: $(whoami)"
echo "当前目录是: $(pwd)"
```

- 解释：

- `#!/bin/bash`：声明该脚本由 Bash 解释器执行。
- `echo "Hello, World!"`：输出 "Hello, World!" 到终端。
- `$(whoami)`：运行 `whoami` 命令来显示当前用户。
- `$(pwd)`：显示当前工作目录。

2. chown：更改文件所有者

- 功能：更改文件的所有者和所属组。

示例：

- 将文件 `example.txt` 的所有者改为 `user1`，所属组改为 `group1`：

```
chown user1:group1 example.txt
```

3. ps 和 top：查看当前进程

- `ps`：列出当前运行的进程。
- `top`：实时显示系统资源使用情况和进程列表。

示例：

- 查看当前用户的所有进程：

```
ps aux
```

- 实时监控进程和资源占用：

```
top
```

4. 实操任务

- **任务 1：**使用 `chmod` 命令为 `script.sh` 文件授予可执行权限。

```
chmod +x script.sh
```

- **任务 2：**运行 `script.sh` 文件，查看输出结果。

```
./script.sh
```

示例输出：

```
Hello, World!  
当前用户是：user1  
当前目录是：/home/user1
```

- **任务 3：**使用 `chown` 命令更改 `file.txt` 的所有者为 `user2`，所属组为 `staff`。

```
chown user2:staff file.txt
```

- **任务 4：**使用 `ps` 查看当前系统的所有进程，并使用 `top` 实时监控系统资源。

小结

在本集课程中，我们学习了如何使用 `chmod` 和 `chown` 命令管理文件的权限和所有者，还学习了如何使用 `ps` 和 `top` 查看当前运行的进程。通过实操示例，我们更好地掌握了这些命令对于文件管理和系统资源监控的重要性。

下节预告

在下集课程中，我们将学习 Bash 脚本的基础，包括如何创建和运行 Bash 脚本，以及其基本语法，如变量和条件语句的使用。