

OPEN-SOURCE EBOOK

++101 LINUX COMMANDS



A dark, moody photograph of a man with glasses looking at computer monitors displaying code.

BOBBY ILIEV

Table of Contents

101 Linux commands Open-source eBook

This is an open-source eBook with 101 Linux commands that everyone should know. No matter if you are a DevOps/SysOps engineer, developer, or just a Linux enthusiast, you will most likely have to use the terminal at some point in your career.

Permission and Ownership

Setting Ownership After Creation

```
# Create directory and set ownership
mkdir myapp
sudo chown user:group myapp

# Create with specific permissions and ownership
sudo mkdir -m 755 /opt/myapp
sudo chown user:group /opt/myapp
```

Creating Directories for Different Users

```
# Create user-specific directories
sudo mkdir -p /home/newuser/{Documents,Downloads,Pictures}
sudo chown -R newuser:newuser /home/newuser
sudo chmod 755 /home/newuser
```

SELinux Context

Setting SELinux Context

```
# Create directory with specific SELinux context
mkdir -Z user_home_t user_data

# Create directory and set context afterward
mkdir secure_data
sudo semanage fcontext -a -t httpd_exec_t
"/path/to/secure_data(.*?)?"
sudo restorecon -R /path/to/secure_data
```

Troubleshooting

Common Issues and Solutions

```
# Permission denied  
sudo mkdir /restricted/path # Use sudo for system directories  
  
# Parent directory doesn't exist  
mkdir -p path/to深深/directory # Use -p flag  
  
# Directory already exists  
mkdir -p existing_dir # -p prevents error if directory exists  
  
# Invalid characters in name  
mkdir "valid_name" # Use quotes or escape special characters
```

Debugging Directory Creation

```
# Check available space before creating  
df -h .  
  
# Verify parent directory permissions  
ls -ld parent_directory  
  
# Check if directory was created successfully  
if [ -d "newdir" ]; then  
    echo "Directory created successfully"  
    ls -ld newdir  
fi
```

Options Reference

| Option | Long Form | Description |
|------------------------|----------------------------|---|
| <code>-m MODE</code> | <code>--mode=MODE</code> | Set file mode (permissions) for created directories |
| <code>-p</code> | <code>--parents</code> | Create parent directories as needed, no error if existing |
| <code>-v</code> | <code>--verbose</code> | Print a message for each created directory |
| <code>-Z CTX</code> | <code>--context=CTX</code> | Set SELinux security context |
| <code>--help</code> | <code>-</code> | Display help message and exit |
| <code>--version</code> | <code>-</code> | Output version information and exit |

Best Practices

Directory Naming Conventions

```
# Use descriptive names
mkdir user_documents    # Good
mkdir stuff              # Poor

# Use consistent naming patterns
mkdir project_2024_01
mkdir project_2024_02

# Avoid spaces and special characters
mkdir my-project        # Good
mkdir "my project"      # Works but can cause issues
```

Organization Strategies

```
# Date-based organization
mkdir -p archives/$(date +%Y)/{01..12}

# Project-based organization
mkdir -p projects/{active,completed,archived}

# User-based organization
mkdir -p users/{admins,developers, testers}
```

Automation and Scripting

```
#!/bin/bash
# Script to create standard project structure

PROJECT_NAME="$1"
if [ -z "$PROJECT_NAME" ]; then
    echo "Usage: $0 <project_name>"
    exit 1
fi

echo "Creating project structure for: $PROJECT_NAME"
mkdir -p "$PROJECT_NAME"/{
    src/{main,test},
    docs/{api,user},
    config/{dev,prod,test},
    scripts/{build,deploy},
    data/{input,output,temp}
}

echo "Project structure created successfully!"
tree "$PROJECT_NAME"
```

| Short Flag | Description |
|-------------------|--|
| -h | print the heading of the columns displayed |
| -b | print the time of last system boot |

The `free` command

The `free` command in Linux/Unix is used to show memory (RAM/SWAP) information.

Usage

Show memory usage

Action: --- Output the memory usage - available and used, as well as swap

Details: --- The values are shown in kibibytes by default.

Command:

```
| free
```

This is a sample from "101 Linux Commands" by Bobby Iliev and the Hacktoberfest community.

For more information, [Click here](#).