Working with Files and Directories in Linux

Introduction

Linux offers powerful command-line tools for managing files and directories. Unlike Windows, where GUI tools like Explorer are commonly used, Linux allows efficient and fast file handling through terminal commands.

Benefits of Terminal Usage

- Faster access and editing of files
- Interactive editing with or without editors like `vim` or `nano`
- Regex support for selective file modifications
- Command chaining, redirection, and batch editing

Basic Operations

Creating Files and Directories

```
# Create an empty file
touch info.txt

# Create a directory
mkdir Storage

# Create nested directories
mkdir -p Storage/local/user/documents

# View directory structure
tree .
```

Output:

```
.
├── info.txt
└── Storage
    └── local
        └── user
            └── documents
```

Creating Files in Nested Directories

```
touch ./Storage/local/user/userinfo.txt

# Updated tree
tree .
```

Output:

```
.
├── info.txt
└── Storage
    └── local
        └── user
            ├── documents
            └── userinfo.txt
```

Moving and Renaming Files

```
# Rename info.txt to information.txt
mv info.txt information.txt

# Create readme.txt
touch readme.txt

# Move files to Storage/
mv information.txt readme.txt Storage/
```

Updated tree:

```
.
└── Storage
    ├── information.txt
    ├── local
    │   └── user
    │       ├── documents
    │       └── userinfo.txt
    └── readme.txt
```

Copying Files

```
# Copy readme.txt to local/
cp Storage/readme.txt Storage/local/
```

Final tree:

```
.
└── Storage
```

```
      ├── information.txt
      ├── local
      │   ├── readme.txt
      │   └── user
      │       ├── documents
      │       └── userinfo.txt
      └── readme.txt
```

Advanced File Handling

- **Redirection**: Redirect input/output using `>` and `>>`
- **Text Editors**: Use `vim`, `nano`, etc., for interactive editing

More advanced techniques will be covered later.

Author: Aizakku04@htb

Topic: Linux File and Directory Management