

diff-pro by Thanura Rukshan

Tool Documentation

Overview

diff-pro is a powerful shell tool that allows for comparing files and directories in various scenarios. It supports local and remote comparisons, checks for differences in permissions, ownership, and content, and generates detailed comparison reports. The tool is particularly useful for system administrators and developers who need to compare configurations or source files between different locations, both locally and across remote servers.

Features

- **Compare Local Files/Directories:** Compare files or directories on the same system.
- **Compare Remote Files/Directories:** Compare files or directories on two separate servers over SSH.
- **Permission Comparison:** Optionally check for differences in file/directory permissions.
- **Ownership Comparison:** Optionally check for differences in ownership (user/group).
- **Content Comparison:** Optionally check for content differences, showing the side-by-side diff.
- **Generates Detailed Reports:** Outputs the comparison result to a specified report file.

Prerequisites

- **SSH Access:** For remote comparisons, ensure SSH is enabled on the remote servers and you have the necessary credentials (username/password).
- **sshpass:** For automated SSH login in non-interactive environments, sshpass must be installed on your local machine.

Installation

1. Copy the script to a directory of your choice, e.g., /usr/local/bin/:

```
sudo cp diff-pro /usr/local/bin/
```

```
sudo chmod +x /usr/local/bin/diff-pro
```
2. Ensure that the script is executable and available in your \$PATH.

Usage

Basic Command Format

```
diff-pro /path/to/dir1/or/file1 /path/to/dir2/or/file2 --report-name <report_name> [-p][-o][-c][-user1 <username>][--user2 <username>][--pw1 <password>][--pw2 <password>]
```

Parameters

- **/path/to/dir1/or/file1**: Path to the first directory or file (local or remote).
- **/path/to/dir2/or/file2**: Path to the second directory or file (local or remote).
- **--report-name <report_name>**: Name of the report file (without extension). The report will be saved as <report_name>.txt.
- **-p**: (Optional) Check for permission differences between the two files or directories.
- **-o**: (Optional) Check for ownership differences (user/group) between the two files or directories.
- **-c**: (Optional) Check for content differences between the two files or directories. This option performs a side-by-side diff of the files' content.
- **--user1 <username>**: (Optional, required for remote comparisons) SSH username for accessing the first remote server.
- **--user2 <username>**: (Optional, required for remote comparisons) SSH username for accessing the second remote server.
- **--pw1 <password>**: (Optional, required for remote comparisons) SSH password for the first remote server.
- **--pw2 <password>**: (Optional, required for remote comparisons) SSH password for the second remote server.

Example Commands

Compare Two Local Files:

```
diff-pro /path/to/file1.txt /path/to/file2.txt --report-name "file_comparison" -p -o -c
```

This compares two files locally, checking permissions, ownership, and content, and generates a report named file_comparison.txt.

Compare Two Local Directories:

```
diff-pro /path/to/dir1 /path/to/dir2 --report-name "dir_comparison" -p -o -c
```

This compares two local directories, checking for permission, ownership, and content differences.

Compare Files on Two Remote Servers:

```
diff-pro 192.168.0.1:/path/to/file1.txt 192.168.0.2:/path/to/file2.txt --report-name  
"remote_comparison" --user1 "username" --user2 "username" --pw1 "password1" --pw2 "password2"  
-p -o -c
```

This compares two files on two different remote servers using SSH, checking for permission, ownership, and content differences, and generates a report named remote_comparison.txt.

Compare Directories on Two Remote Servers:

```
diff-pro 192.168.0.1:/path/to/dir1 192.168.0.2:/path/to/dir2 --report-name
"remote_dir_comparison" --user1 "username" --user2 "username" --pw1 "password1" --pw2
"password2" -p -o -c
```

This compares two directories on different remote servers using SSH.

Report Format

The generated report is a text file containing detailed comparison results, including:

- Date and time of comparison.
- Paths of the compared files or directories.
- Permission, ownership, and content differences (if applicable).
- Missing files (if any) between directories.

Example Output:

Comparison Report - Thu Nov 7 2024

Path 1: /path/to/file1.txt

Path 2: /path/to/file2.txt

=====

Permissions differ: 644 vs 755

Ownership differs: user1:group1 vs user2:group2

Content differs:

file1.txt: Line 1

file2.txt: Line 2

Remote File Access

When comparing files on remote servers, the script uses SSH to fetch the files or directories from the remote system. Ensure SSH access is enabled and the necessary credentials (username/password) are provided for both servers.

The tool supports two types of remote comparisons:

1. **Remote File Comparison:** Compare individual files on two remote servers.

2. **Remote Directory Comparison:** Compare entire directories across remote servers.

Requirements

1. **SSH Access:** SSH access to the remote servers.
2. **sshpass:** For automated SSH login (used by the script for non-interactive login).
3. **Bash:** The script requires a Bash environment.

Troubleshooting

1. **Missing sshpass:** If the tool fails due to missing sshpass, install it on your system:
`sudo apt-get install sshpass` # On Ubuntu/Debian
2. **Permission Denied:** Ensure that you have the necessary permissions to read the files and directories on both the local and remote systems.
3. **Invalid SSH Credentials:** Ensure the SSH credentials are correct for the remote servers.
4. **Missing Files:** If files are missing between directories, they will be listed in the report under "File missing in".