

## 100 Useful Scripts for Manual Tasks in Linux

**Follow – Krishan Bhatt** 

Linux users often find themselves repeating certain tasks. Scripting these tasks can save time and reduce errors. Below are 100 scripts organized into categories to help you manage files, monitor system performance, and more.

### File Management

1. List all files in a directory:

```
#!/bin/bash
```

```
ls -al
```

2. Find and delete files older than 30 days:

```
#!/bin/bash
```

```
find /path/to/directory -type f -mtime +30 -exec rm {} \;
```

3. Copy a directory and its contents:

```
#!/bin/bash
```

```
cp -r /source/directory /destination/directory
```

4. Rename multiple files by adding a prefix:

```
#!/bin/bash
```

```
for file in *.txt; do mv "$file" "prefix_$file"; done
```

5. Rename multiple files by changing extension:

```
#!/bin/bash
```

```
for file in *.jpeg; do mv "$file" "${file%.jpeg}.jpg"; done
```

6. Count the number of files in a directory:

```
#!/bin/bash
```

```
ls -1 | wc -l
```

7. Create a backup of a directory:

```
#!/bin/bash
```

```
tar -czvf backup.tar.gz /path/to/directory
```

8. Extract a tar.gz archive:

```
#!/bin/bash
```

```
tar -xvzf archive.tar.gz
```

9. Find and replace text in multiple files:

```
#!/bin/bash
```

```
find /path/to/directory -type f -name "*.txt" -exec sed -i 's/old_text/new_text/g' {} \;
```

10. Create a directory structure:

```
#!/bin/bash
```

```
mkdir -p /parent/child/subchild
```

## System Monitoring

11. Check disk usage:

```
#!/bin/bash
```

```
df -h
```

12. Check memory usage:

```
#!/bin/bash
```

```
free -m
```

13. Monitor CPU usage:

```
#!/bin/bash
```

```
top
```

14. List top 10 memory-consuming processes:

```
#!/bin/bash
```

```
ps aux --sort=-%mem | head -n 10
```

15. Monitor network traffic:

```
#!/bin/bash
```

```
iftop
```

16. Check open ports:

```
#!/bin/bash
```

```
netstat -tuln
```

17. Show current users:

```
#!/bin/bash
```

```
who
```

18. Display system uptime:

```
#!/bin/bash
```

```
uptime
```

19. Check system load averages:

```
#!/bin/bash
```

```
uptime | awk '{print $8,$9,$10}'
```

20. List all installed packages:

```
#!/bin/bash
```

```
dpkg -l
```

Networking

21. Ping a host to check connectivity:

```
#!/bin/bash
```

```
ping -c 4 example.com
```

22. Get the IP address of a domain:

```
#!/bin/bash
```

```
nslookup example.com
```

23. Show current network interfaces:

```
#!/bin/bash
```

```
ifconfig
```

24. Download a file from the internet:

```
#!/bin/bash
```

```
wget http://example.com/file.zip
```

25. Check the status of a web server:

```
#!/bin/bash
```

```
curl -Is http://example.com | head -n 1
```

26. Get public IP address:

```
#!/bin/bash
```

```
curl ifconfig.me
```

27. Scan for open ports on a server:

```
#!/bin/bash  
nmap example.com
```

28. Test connection to a specific port:

```
#!/bin/bash  
nc -zv example.com 80
```

29. Flush DNS cache:

```
#!/bin/bash  
sudo systemd-resolve --flush-caches
```

30. Restart network service:

```
#!/bin/bash  
sudo systemctl restart networking
```

User Management

31. Add a new user:

```
#!/bin/bash
```

```
sudo adduser newuser
```

32. Delete a user:

```
#!/bin/bash
```

```
sudo deluser newuser
```

33. List all users:

```
#!/bin/bash
```

```
cut -d: -f1 /etc/passwd
```

34. Change a user's password:

```
#!/bin/bash
```

```
sudo passwd username
```

35. Add a user to a group:

```
#!/bin/bash
```



```
sudo usermod -aG groupname username
```

36. Remove a user from a group:

```
#!/bin/bash
```

```
sudo deluser username groupname
```

37. Show groups a user belongs to:

```
#!/bin/bash
```

```
groups username
```

38. List all groups:

```
#!/bin/bash
```

```
cut -d: -f1 /etc/group
```

39. Create a new group:

```
#!/bin/bash
```

```
sudo groupadd groupname
```

40. Delete a group:

```
#!/bin/bash
```

```
sudo groupdel groupname
```

## Process Management

41. List all running processes:

```
#!/bin/bash
```

```
ps aux
```

42. Kill a process by PID:

```
#!/bin/bash
```

```
kill PID
```

43. Kill a process by name:

```
#!/bin/bash
```

```
pkill processname
```

44. Show top 10 CPU-consuming processes:

```
#!/bin/bash
```

```
ps aux --sort=-%cpu | head -n 10
```

45. Start a process in the background:

```
#!/bin/bash
```

```
command &
```

46. Show process tree:

```
#!/bin/bash
```

```
pstree
```

47. Check if a process is running:

```
#!/bin/bash
```

```
pgrep processname
```

48. Pause a process:

```
#!/bin/bash
```

```
kill -STOP PID
```

49. Resume a paused process:

```
#!/bin/bash
```

```
kill -CONT PID
```

50. Change process priority (nice value):

```
#!/bin/bash
```

```
renice -n 10 -p PID
```

## Disk Management

51. Mount a disk:

```
#!/bin/bash
```

```
sudo mount /dev/sdX1 /mnt
```

52. Unmount a disk:

```
#!/bin/bash
```

```
sudo umount /mnt
```

53. Check disk space usage:

```
#!/bin/bash
```

```
df -h
```

54. Check disk inode usage:

```
#!/bin/bash
```

```
df -i
```

55. List all disks and partitions:

```
#!/bin/bash
```

```
lsblk
```

56. Format a disk partition to ext4:

```
#!/bin/bash
```

```
sudo mkfs.ext4 /dev/sdX1
```

57. Create a swap file:

```
#!/bin/bash
```

```
sudo fallocate -l 1G /swapfile
```

```
sudo chmod 600 /swapfile
```

```
sudo mkswap /swapfile
```

```
sudo swapon /swapfile
```

58. Disable a swap file:

```
#!/bin/bash
```

```
sudo swapoff /swapfile
```

59. Add swap file to fstab for automatic mounting:

```
#!/bin/bash
```

```
echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
```

60. Check swap usage:

```
#!/bin/bash
```

```
swapon --show
```

Disk Management

61. Create an ISO image from a directory:

```
#!/bin/bash
```

```
mkisofs -o image.iso /path/to/directory
```

62. Burn an ISO image to a CD/DVD:

```
#!/bin/bash
```

```
wodim -v dev=/dev/cdrom image.iso
```

63. Check disk for errors:

```
#!/bin/bash
```

```
sudo fsck /dev/sdX1
```

64. Resize a partition:

```
#!/bin/bash
```

```
sudo resize2fs /dev/sdX1
```

65. Create a new partition using fdisk:

```
#!/bin/bash
```

```
sudo fdisk /dev/sdX
```

66. Show disk usage of each directory in current path:

```
#!/bin/bash
```

```
du -sh *
```

67. Check SMART status of a disk:

```
#!/bin/bash
```

```
sudo smartctl -a /dev/sdX
```

68. Securely erase a disk:

```
#!/bin/bash
```

```
sudo dd if=/dev/zero of=/dev/sdX bs=1M
```

69. Create a disk image with dd:

```
#!/bin/bash
```

```
sudo dd if=/dev/sdX of=/path/to/image.img bs=1M
```

70. Restore a disk image with dd:

```
#!/bin/bash
```

```
sudo dd if=/path/to/image.img of=/dev/sdX bs=1M
```



## Backup and Restore

71. Create a full system backup:

```
#!/bin/bash
```

```
sudo tar -cvpzf /backup/full-backup.tar.gz --exclude=/backup --one-file-system /
```

72. Backup MySQL databases:

```
#!/bin/bash
```

```
mysqldump -u username -p password --all-databases > all-databases.sql
```

73. Restore MySQL databases:

```
#!/bin/bash
```

```
mysql -u username -p password < all-databases.sql
```

74. Create incremental backups using rsync:

```
#!/bin/bash
```

```
rsync -av --delete /source/directory /backup/directory
```

75. Create a backup of PostgreSQL databases:

```
#!/bin/bash
```

```
pg_dumpall -U username > all-databases.sql
```

76. Restore PostgreSQL databases:

```
#!/bin/bash
```

```
psql -U username -f all-databases.sql
```

77. Create a backup of home directory:

```
#!/bin/bash
```

```
tar -cvpzf /backup/home-backup.tar.gz /home/username
```

78. Backup a directory to a remote server using scp:

```
#!/bin/bash
```

```
scp -r /path/to/directory user@remote:/path/to/backup
```

79. Backup using rsnapshot:

```
#!/bin/bash
```

```
sudo rsnapshot daily
```

80. Restore a specific file from a tar archive:

```
#!/bin/bash
```

```
tar -xvzf backup.tar.gz path/to/file
```

## Security

81. Generate an SSH key pair:

```
#!/bin/bash
```

```
ssh-keygen -t rsa -b 4096 -C "your_email@example.com"
```

82. Add a public key to the authorized\_keys file:

```
#!/bin/bash
```

```
cat ~/.ssh/id_rsa.pub | ssh user@remote "mkdir -p ~/.ssh && cat >>
~/.ssh/authorized_keys"
```

83. Change file permissions:

```
#!/bin/bash
```

```
chmod 755 /path/to/file
```

84. Change file ownership:

```
#!/bin/bash
```

```
chown user:group /path/to/file
```

85. Encrypt a file using GPG:

```
#!/bin/bash
```

```
gpg -c file.txt
```

86. Decrypt a file using GPG:

```
#!/bin/bash
```

```
gpg file.txt.gpg
```

87. Set up a firewall using UFW:

```
#!/bin/bash
```

```
sudo ufw enable
```

```
sudo ufw allow ssh
```

88. Check UFW status and rules:

```
#!/bin/bash
```

`sudo ufw status verbose`

89. Securely delete a file:

```
#!/bin/bash
```

```
shred -u file.txt
```

90. List all sudo users:

```
#!/bin/bash
```

```
grep '^sudo:.*$' /etc/group | cut -d: -f4
```

## Miscellaneous

91. Download a file with curl:

```
#!/bin/bash
```

```
curl -O http://example.com/file.zip
```

92. Create a symbolic link:

```
#!/bin/bash
```

```
ln -s /path/to/original /path/to/link
```

93. Schedule a task with cron:

```
#!/bin/bash
```

```
echo "0 5 * * * /path/to/script.sh" | crontab -
```

94. List all cron jobs:

```
#!/bin/bash
```

```
crontab -l
```

95. Show system information:

```
#!/bin/bash
```

```
uname -a
```

96. Display detailed system information:

```
#!/bin/bash
```

```
lshw
```

97. Update package list and upgrade all packages:

```
#!/bin/bash
```

```
sudo apt-get update && sudo apt-get upgrade -y
```

98. Install a package:

```
#!/bin/bash
```

```
sudo apt-get install package-name
```

99. Remove a package:

```
#!/bin/bash
```

```
sudo apt-get remove package-name
```

100. Clean up unused packages:

```
bash  #!/bin/bash  sudo apt-get autoremove -y  sudo apt-get clean
```

These scripts cover a wide range of tasks and can be tailored to suit your specific needs. Using them can greatly enhance your efficiency and productivity when working with Linux systems.