

Linux Command Mastery and Scripting Project

Done By:

Lana Alnimreen

Supervisor:

Dr. Motasem Aldiab

Abstract

This report introduces the practical application of Bash scripting. The report goes into two scripts: a Backup Script, which allows users to specify directory backups with compression and logging capabilities, and a System Health Check Script, which provides real-time monitoring of storage, RAM, running services, and system updates. These scripts aim to automate and monitor administrative processes in order to improve system dependability.

Contents

Abstract.....	2
List Of Figures	4
Introduction.....	5
Backup script	5
Flow Chart	6
Health script.....	9
Impact	11

List Of Figures

Figure 1: BackUP Script	6
Figure 2: Backup with compression.....	7
Figure 3: The Backup Directory	7
Figure 4: Log File	7
Figure 5: Backup without compression	8
Figure 6: The Backup Directory	8
Figure 7: Log File	8
Figure 8: Output of Health Script	9

Introduction

Bash scripting is an excellent approach to automate different types of tasks in a system. Developers can avoid doing repetitive tasks using bash scripting. Also, it supports variables, conditional statements, and loops just like programming languages. In this report I will discuss two scripts, the first one is a backup script that takes the paths of the directory to backup, and the destination where to save the backup. The second script is the health script, it displays system-health information such as storage usage, memory usage, running services and recent system updates.

Backup script

The Backup Script's aim is to produce a copy of user-specified files, ensuring that important data is safely moved to a backup place. Provides flexibility by allowing users to select which folders to backup. Saving storage space and maybe speeding up the backup by compressing the backup directory. Provides a traceable record of the backup process, which helps monitor and fix difficulties.

I use the WSL on Windows to complete this project and go through these steps:

Create a file via *touch* command then add the code in the file then use *chmod +x BackUp.sh* to make the file mode executable then Run the Script: *./BackUp.sh* Options that can be passed with the command:

```
./backup_script.sh /path/to/source1 /path/to/source2 /path/to/destination
```

- /path/to/source1: The paths to backup, it could be more than one space separated.
- /path/to/destination: The destination path where to save the backup.

Note:

If no source path or destination are provided to backup it will end the program and show message to user "Please provide at least one <source_directory1> and <destination_directory>"

Exception handling:

- Throwing an error if the source directory does not exist.
- Creating destination directory if it does not exist.
- If the selected compress format is not already installed, it will be installed.

Flow Chart

The flowchart below explains how the script work:

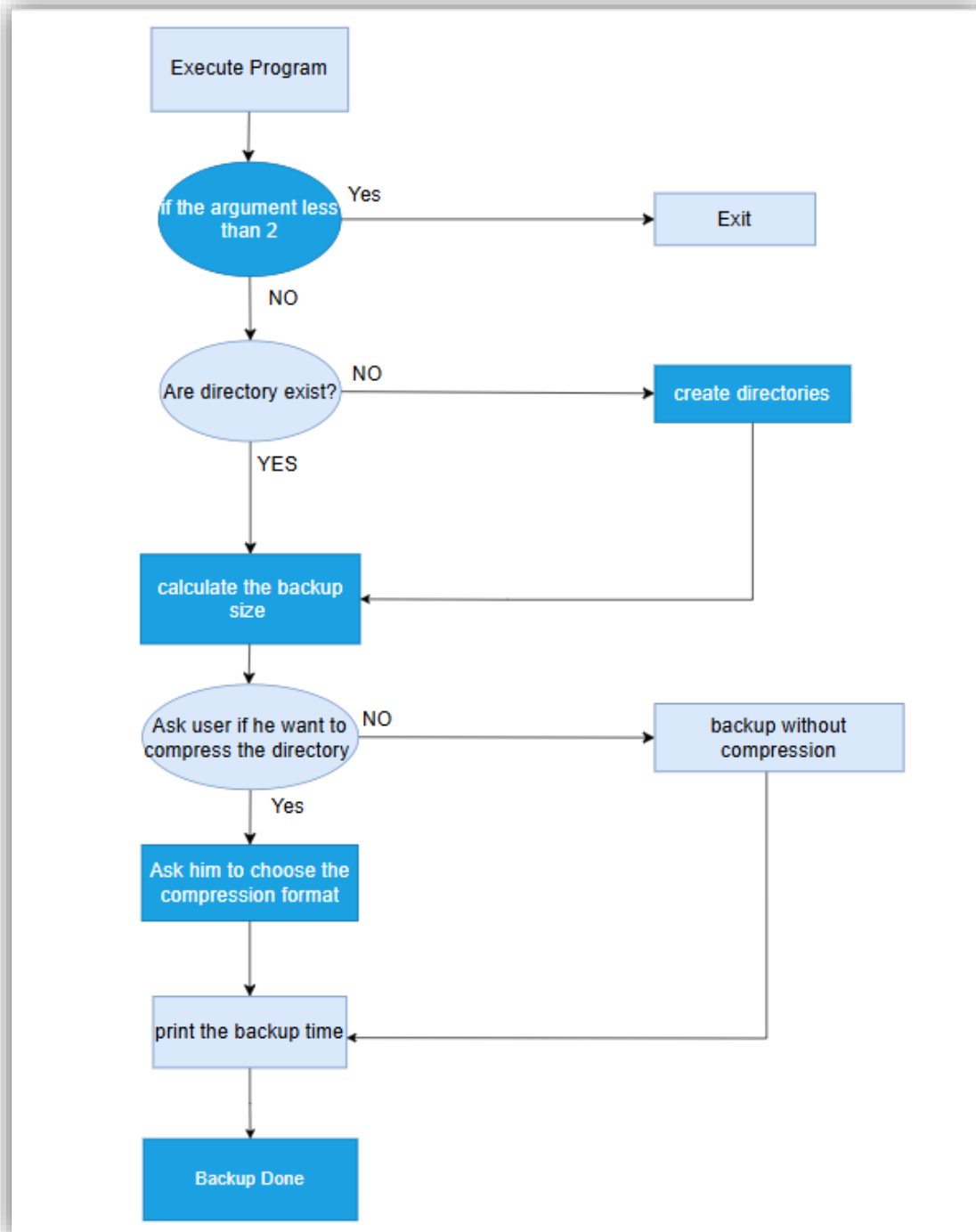
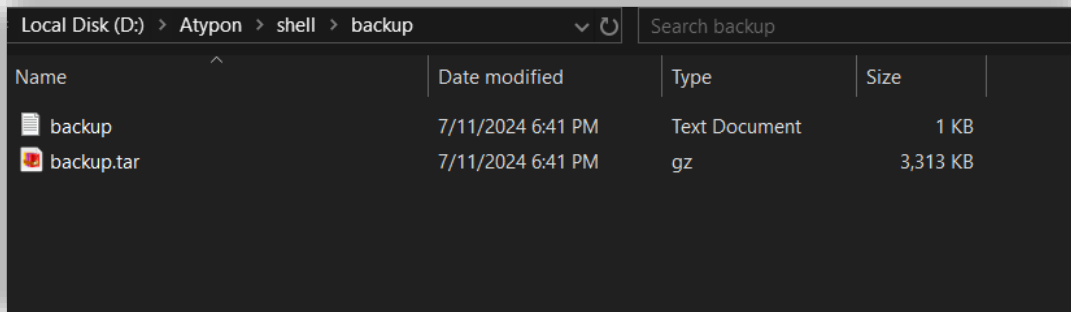


Figure 1: Backup Script

Here I choose to do Backup with compression using tar format option:

```
lana@DESKTOP-S0Q6K6J: /mnt/d/Atypon/shell
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$ clear
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$ ./BackUp.sh /mnt/d/Atypon/git /mnt/d/Atypon/karel /mnt/d/Atypon/Simulator /mnt/d/Atypon/shell/backup
2024-07-11 18:36:49 - Log file created at /mnt/d/Atypon/shell/backup/backup.log
2024-07-11 18:36:49 - Total size of the source directories to backup: 3.6M
Do you want to compress the backup file? (y/n): y
Choose from following formats: (tar, zip, rar):
tar
Installing /mnt/d/Atypon/git package...
[sudo] password for lana:
2024-07-11 18:37:07 - Starting the backup with tar compression
Backup Duration: 18 seconds
2024-07-11 18:37:07 - Backup script ended.
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$
```

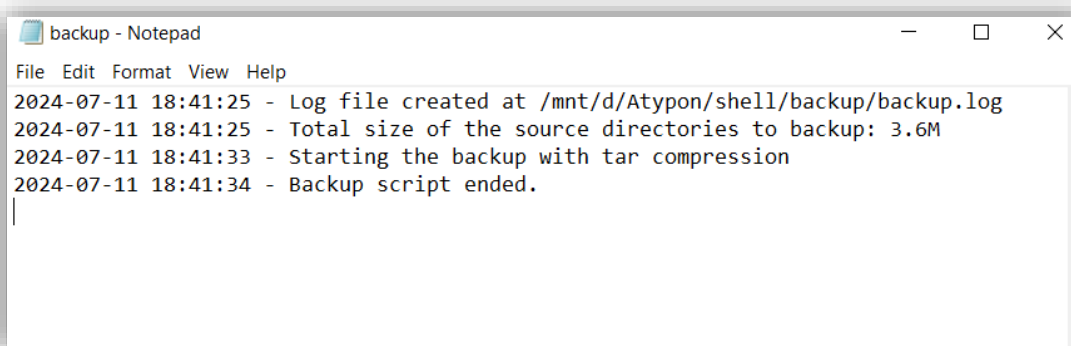
Figure 2: Backup with compression



The screenshot shows a Windows File Explorer window with the address bar set to 'Local Disk (D:) > Atypon > shell > backup'. The main area displays a table of files and folders.

Name	Date modified	Type	Size
backup	7/11/2024 6:41 PM	Text Document	1 KB
backup.tar	7/11/2024 6:41 PM	gz	3,313 KB

Figure 3: The Backup Directory



The screenshot shows a Notepad window titled 'backup - Notepad'. The text inside the window is as follows:

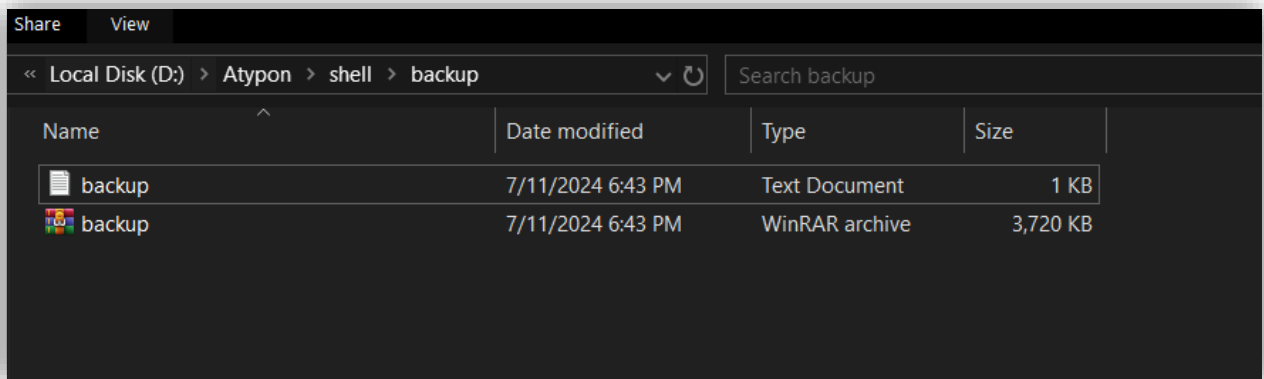
```
File Edit Format View Help
2024-07-11 18:41:25 - Log file created at /mnt/d/Atypon/shell/backup/backup.log
2024-07-11 18:41:25 - Total size of the source directories to backup: 3.6M
2024-07-11 18:41:33 - Starting the backup with tar compression
2024-07-11 18:41:34 - Backup script ended.
```

Figure 4: Log File

Here I just did backup without compression:

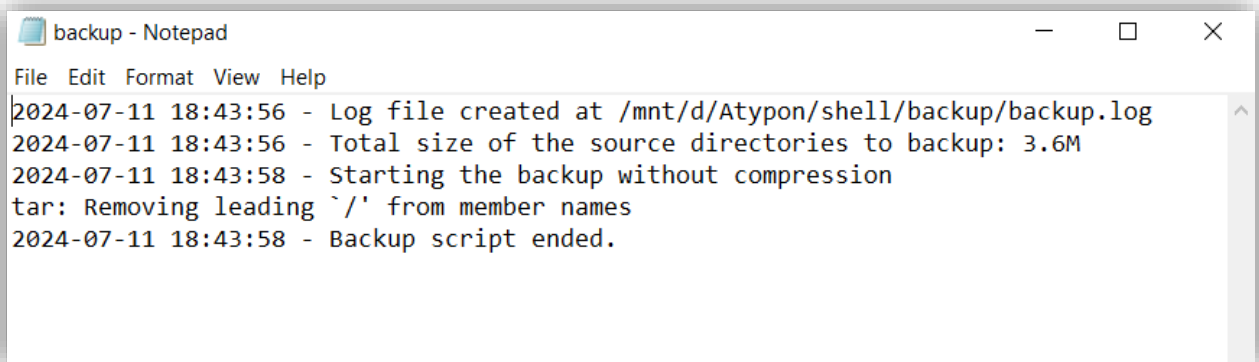
```
lana@DESKTOP-S0Q6K6J: /mnt/d/Atypon/shell
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$ clear
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$ ./BackUp.sh /mnt/d/Atypon/git /mnt/d/Atypon/karel /mnt/d/Atypon/Simulator /mnt/d/Atypon/shell/backup
2024-07-11 18:43:56 - Log file created at /mnt/d/Atypon/shell/backup/backup.log
2024-07-11 18:43:56 - Total size of the source directories to backup: 3.6M
Do you want to compress the backup file? (y/n): n
2024-07-11 18:43:58 - Starting the backup without compression
Backup Duration: 2 seconds
2024-07-11 18:43:58 - Backup script ended.
lana@DESKTOP-S0Q6K6J:/mnt/d/Atypon/shell$
```

Figure 5: Backup without compression



Name	Date modified	Type	Size
backup	7/11/2024 6:43 PM	Text Document	1 KB
backup	7/11/2024 6:43 PM	WinRAR archive	3,720 KB

Figure 6: The Backup Directory



```
backup - Notepad
File Edit Format View Help
2024-07-11 18:43:56 - Log file created at /mnt/d/Atypon/shell/backup/backup.log
2024-07-11 18:43:56 - Total size of the source directories to backup: 3.6M
2024-07-11 18:43:58 - Starting the backup without compression
tar: Removing leading `/' from member names
2024-07-11 18:43:58 - Backup script ended.
```

Figure 7: Log File

Health script

The purpose of a System Health Check Script is to provide an automated way to monitor the overall health and performance of a system. By regularly checking storage usage, memory usage, running services, and recent system updates. This script runs infinitely until we press "CTRL + C". It updates the above-mentioned information every two seconds.

```
lana@DESKTOP-S0Q6K6J: /mnt/d/Atypon/shell
lana@DESKTOP-S0Q6K6J: /mnt/d/Atypon/shell$ ./health.sh
System Health Report
=====
Disk Space Usage:
Filesystem      Size  Used Avail Use% Mounted on
rootfs          244G  154G   91G   63% /
none            244G  154G   91G   63% /dev
none            244G  154G   91G   63% /run
none            244G  154G   91G   63% /run/lock
none            244G  154G   91G   63% /run/shm
none            244G  154G   91G   63% /run/user
tmpfs           244G  154G   91G   63% /sys/fs/cgroup
C:\             244G  154G   91G   63% /mnt/c
D:\             203G   7.8G  196G    4% /mnt/d

Memory Usage:
             total      used      free   shared  buff/cache   available
Mem:         7.8G       5.6G       1.9G       17M       223M        2.0G
Swap:         24G       476M        23G

Free memory is 26% of total.
Free memory is at a healthy level.

Running Services:
[ ? ] apport
[ ? ] cryptdisks
[ ? ] hwclock.sh
[ + ] irqbalance
[ + ] iscsid
[ + ] lvm2-lvmetad
[ + ] lvm2-lvmpolld
[ + ] open-iscsi
[ ? ] plymouth
[ ? ] plymouth-log

Recent System Updates:
apparmor/bionic-updates,bionic-security 2.12-4ubuntu5.3 amd64 [upgradable from: 2.12-4ubuntu5.1]
base-files/bionic-updates 10.1ubuntu2.12 amd64 [upgradable from: 10.1ubuntu2.11]
distro-info-data/bionic-updates,bionic-security 0.37ubuntu0.18 all [upgradable from: 0.37ubuntu0.17]
iptables/bionic-updates 1.6.1-2ubuntu2.1 amd64 [upgradable from: 1.6.1-2ubuntu2]
libapparmor1/bionic-updates,bionic-security 2.12-4ubuntu5.3 amd64 [upgradable from: 2.12-4ubuntu5.1]
libip4tc0/bionic-updates 1.6.1-2ubuntu2.1 amd64 [upgradable from: 1.6.1-2ubuntu2]
libip6tc0/bionic-updates 1.6.1-2ubuntu2.1 amd64 [upgradable from: 1.6.1-2ubuntu2]
libiptc0/bionic-updates 1.6.1-2ubuntu2.1 amd64 [upgradable from: 1.6.1-2ubuntu2]
liblxc-common/bionic-updates 3.0.3-0ubuntu1~18.04.3 amd64 [upgradable from: 3.0.3-0ubuntu1~18.04.1]
liblxc1/bionic-updates 3.0.3-0ubuntu1~18.04.3 amd64 [upgradable from: 3.0.3-0ubuntu1~18.04.1]
```

Figure 8: Output of Health Script

Regarding storage, please note the following:

- **Filesystem:** The name of each filesystem or disk partition.
- **Size:** The total size of the filesystem.
- **Used:** The amount of space currently used.
- **Available:** The amount of space available for use.
- **Use%:** The percentage of the filesystem's total size that is currently used.
- **Mounted on:** The directory where the filesystem is mounted.

Regarding memory, please note the following:

Mem:

- **total:** The total amount of memory available.
- **used:** The amount of memory currently in use.
- **free:** The amount of memory that is currently not in use and is readily available.
- **shared:** The amount of memory used by shared resources, such as temporary file storage.
- **buff/cache:** The amount of memory used for buffers and caches.
- **available:** The amount of memory available for starting new applications, without swapping.

Swap:

Swap space is used when the RAM is full and the system needs more memory.

- **total:** The total amount of swap space available on the system.
- **used:** The amount of swap space currently in use.
- **free:** The amount of swap space that is free and available.

Running Services, please note the following:

- [+] indicates running services.
- [-] indicates stopped services.
- [?] indicates services for which the status is unknown.

Recent System Updates, please note that:

It prints the last 10 updates that occurred on the system. It works for both Ubuntu/Debian based systems, and Red Hat systems. Since it is done by checking for any “**apt**” or “**yum**” commands in the history file. **APT** (Advanced Package Tool) used by Debian-based distributions and **YUM** (Yellowdog Updater, Modified) used by Red Hat-based distributions are package managers used primarily in Linux distributions to manage software packages, installations, and updates.

Impact

This assignment was educational, showing Linux's strong ability. While my important research and learning, I've gained a better understanding for Linux. This experience taught me that Linux is a foundation for modern computer infrastructure, providing critical capabilities for effectively handling complex systems.