

Lab 4: Bash scripting

Ahmed Baha Eddine Alimi

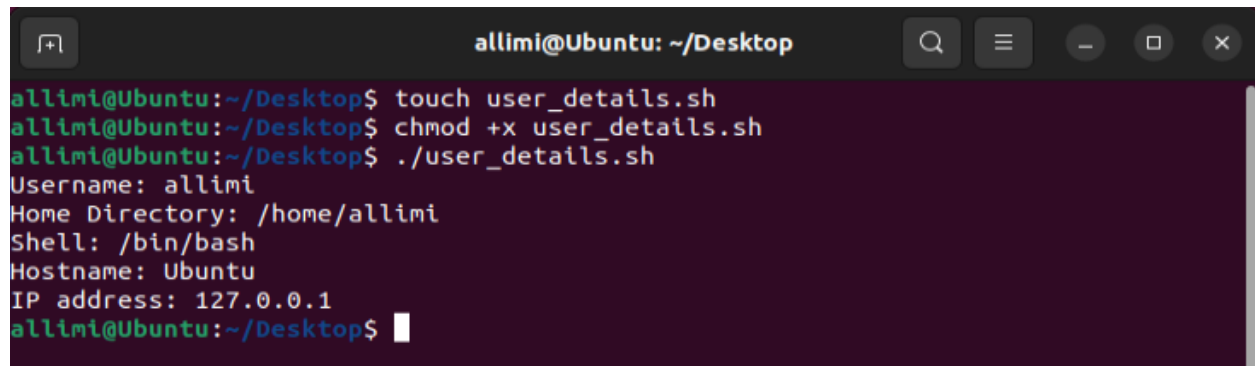
Assignment Report

SD-01

I. Task 1 (user_details.sh file) :

1. Write a bash script that displays the following details of the logged-in user from the environment variables:

- Login username
- Home directory
- Shell
- The hostname of the system
- The script should extract the IP address of the system from the ifconfig or ip command. Save the IP address to the ipaddress variable and display it as output.

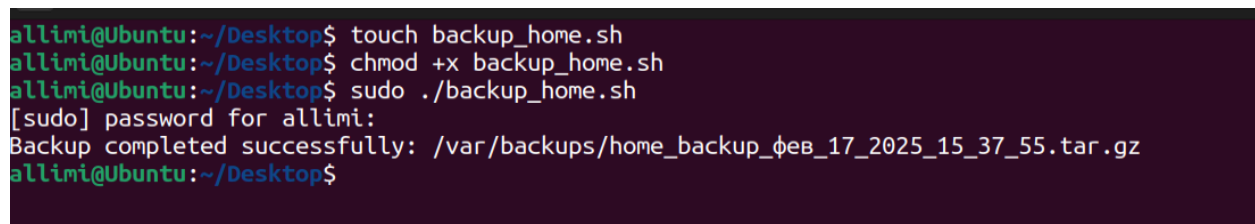
A terminal window titled 'allimi@Ubuntu: ~/Desktop' showing the execution of a bash script. The user 'allimi' runs 'touch user_details.sh', 'chmod +x user_details.sh', and then './user_details.sh'. The script outputs the following details: Username: allimi, Home Directory: /home/allimi, Shell: /bin/bash, Hostname: Ubuntu, and IP address: 127.0.0.1.

```
allimi@Ubuntu:~/Desktop$ touch user_details.sh
allimi@Ubuntu:~/Desktop$ chmod +x user_details.sh
allimi@Ubuntu:~/Desktop$ ./user_details.sh
Username: allimi
Home Directory: /home/allimi
Shell: /bin/bash
Hostname: Ubuntu
IP address: 127.0.0.1
allimi@Ubuntu:~/Desktop$
```

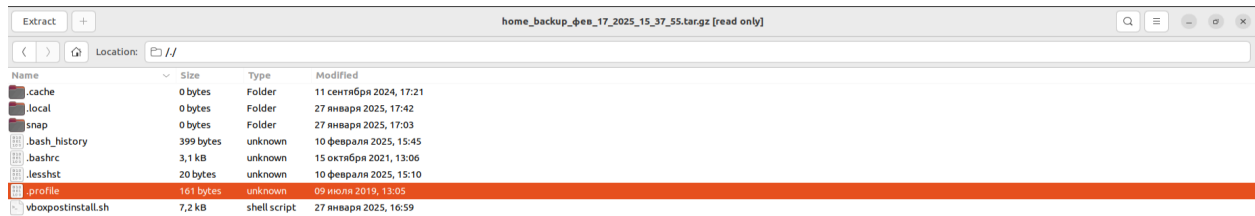
II. Task 2 (backup_home.sh file):

1. Backups are important in system administration. Create a script that will backup your home directory.

- The backup file should be compressed to tar.gz.
- All files and directory permissions should be preserved in the backup.
- The backup destination directory is /var/backups/
- The script should create the destination directory if it doesn't already exist.
- The backup file name should take the format
home_backup_month_day_year_hour_minute_second.tar.gz.
For example home_backup_Feb_18_2023_02_30_02.tar.gz

A terminal window titled 'allimi@Ubuntu: ~/Desktop' showing the execution of a backup script. The user 'allimi' runs 'touch backup_home.sh', 'chmod +x backup_home.sh', and 'sudo ./backup_home.sh'. The script prompts for a password and then outputs: 'Backup completed successfully: /var/backups/home_backup_feb_17_2025_15_37_55.tar.gz'.

```
allimi@Ubuntu:~/Desktop$ touch backup_home.sh
allimi@Ubuntu:~/Desktop$ chmod +x backup_home.sh
allimi@Ubuntu:~/Desktop$ sudo ./backup_home.sh
[sudo] password for allimi:
Backup completed successfully: /var/backups/home_backup_feb_17_2025_15_37_55.tar.gz
allimi@Ubuntu:~/Desktop$
```



III. Task 3 (`system_artifacts.sh` file):

1. Write a bash script that checks various artifacts on the system.

The script mainly checks for system information, and OS components. Your script should do the following:

- Print the OS kernel name and kernel version.
- Print the system architecture.
- Print all currently logged in users (show the date or time which the users logged in, and show the command line of the users' current process).
- Verify that EFI is enabled and print the relevant output.
- List all connected block devices (Bonus: Identify the devices that have the GPT partition by adding an * to them in the output).
- List the first boot device on your system. This should be done according to the boot order in the NVRAM.

```
allimi@Ubuntu: ~/Desktop
allimi@Ubuntu:~/Desktop$ touch system_artifacts.sh
allimi@Ubuntu:~/Desktop$ chmod +x system_artifacts.sh
allimi@Ubuntu:~/Desktop$ sudo ./system_artifacts.sh
[sudo] password for allimi:
Kernel Name: Linux
Kernel Version: 6.8.0-52-generic
System Architecture: x86_64
Logged-in Users:
allimi  tty2          2025-02-17 15:34 00:13      1349 (tty2)
allimi  pts/1          2025-02-17 15:47 .          4521
EFI is enabled.
Connected Block Devices:
NAME      SIZE TYPE MOUNTPOINT
loop0     4K  loop /snap/bare/5
loop1    74,3M loop /snap/core22/1612
loop2    73,9M loop /snap/core22/1748
loop3   271,2M loop /snap/firefox/4848
loop4   505,1M loop /snap/gnome-42-2204/176
loop5   91,7M loop /snap/gtk-common-themes/1535
loop6   12,9M loop /snap/snap-store/1113
loop7   12,2M loop /snap/snap-store/1216
loop8   38,8M loop /snap/snapd/21759
loop9   44,4M loop /snap/snapd/23545
loop10  500K loop /snap/snapd-desktop-integration/178
loop11  568K loop /snap/snapd-desktop-integration/253
sda      25G  disk
├─sda1    1M  part
├─sda2   513M part /boot/efi
└─sda3   24,5G part /
sr0     1024M rom
First Boot Device:
BootCurrent: 0004
BootOrder: 0004,0000,0001,0002,0003
Boot0004* ubuntu
allimi@Ubuntu:~/Desktop$
```

IV. Bonus Task 4 (scan_bash_files.sh file):

1. Write a bash script that scans the entire system for files that contain the string `"/bin/bash"`. The script should print only the matches that the currently logged in user has execute permission on.

```
allini@ubuntu:~/Desktop$ touch scan_bash_files.sh
allini@ubuntu:~/Desktop$ chmod +x scan_bash_files.sh
allini@ubuntu:~/Desktop$ sudo ./scan_bash_files.sh
Files containing '/bin/bash' with execute permission for root:
/etc/cron.daily/update-resolv-conf
/etc/pdnl/xsession
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/hw/hw_get_das_info.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/hw/hw_get_ifconfig.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/hw/hw_get_dhcp_info.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/net/vmlinux_test.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/net/vmlinux_test.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/test_unshare_from_different_cu.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/stat_all_metrics.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/stat_metrics_values.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/stat_test_output.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/stat_csv_output.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/test_perf_data_converter_json.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/dammon.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/test_task_analyzer.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/test_json_output.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/stat_json_output.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/tests/shell/test_data_symbol.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/check-headers.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/perf/perf-archive.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/certs/print-cert-tls-bash.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/firmware/fw_load.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/firmware/fw_fallback.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/firmware/fw_filesystem.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/firmware/fw_run_tests.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/nvme/run_hugelbfs_test.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/fpu/run_test_fpu.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/efivarfs/efivarfs.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/AsmTest/initial.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/cgroup/test_stress.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/cgroup/test_cpnet_prs.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/cgroup/vlth_stress.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/rt/rt_loopback.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/p9/p9-mockup.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/ptp/phc.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/ndisc_unsolicited_na_test.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/rf_route_leaking.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/leaking_derecopy_bx.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/rtnetlink.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/cmg_time.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/userspace_pm.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/pm_netlink.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/mptcp_join.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/mptcp_connect.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/mptcp_listen_files.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/mptcp/mptcp_socket.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/rf_strict_mode_test.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/altnames.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/rtnetlink_ack_lat.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/rfb_rule_tests.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/openvsw/openvswlitch.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/cmg_ipv6.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/test_vxlan_vni_filtering.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/psock_and.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/test_bridge_neigh_suppress.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/traceroute.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/icmp.sh
/usr/src/linux-hwe-6.8-headers-6.8.0-51/tools/testing/selftests/net/hur/hur_ping.sh
```