

CS4375-13948

Diego Jared Avina

djavinaesco@miners.utep.edu

## HW 1: Introduction to xv6

### Task 1. Boot xv6 and explore utilities:

For this part of the assignment, I decided to go with a tool named multipass; a tool used to generate cloud-style ubuntu virtual machines. Since multipass does not come with a UI, I decided to download one, and use Microsoft Remote Desktop in order to run ubuntu.

Multipass:

```
base ~
multipass shell robust-guppy

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-84-generic aarch64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

System information as of Fri Sep 22 10:49:52 MDT 2023

System load:          0.240234375
Usage of /:            26.8% of 24.05GB
Memory usage:         7%
Swap usage:           0%
Processes:            174
Users logged in:      0
IPv4 address for enp0s1: 192.168.64.3
IPv6 address for enp0s1: fd59:5c2f:ce05:494a:5054:ff:fe37:128b

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
  just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

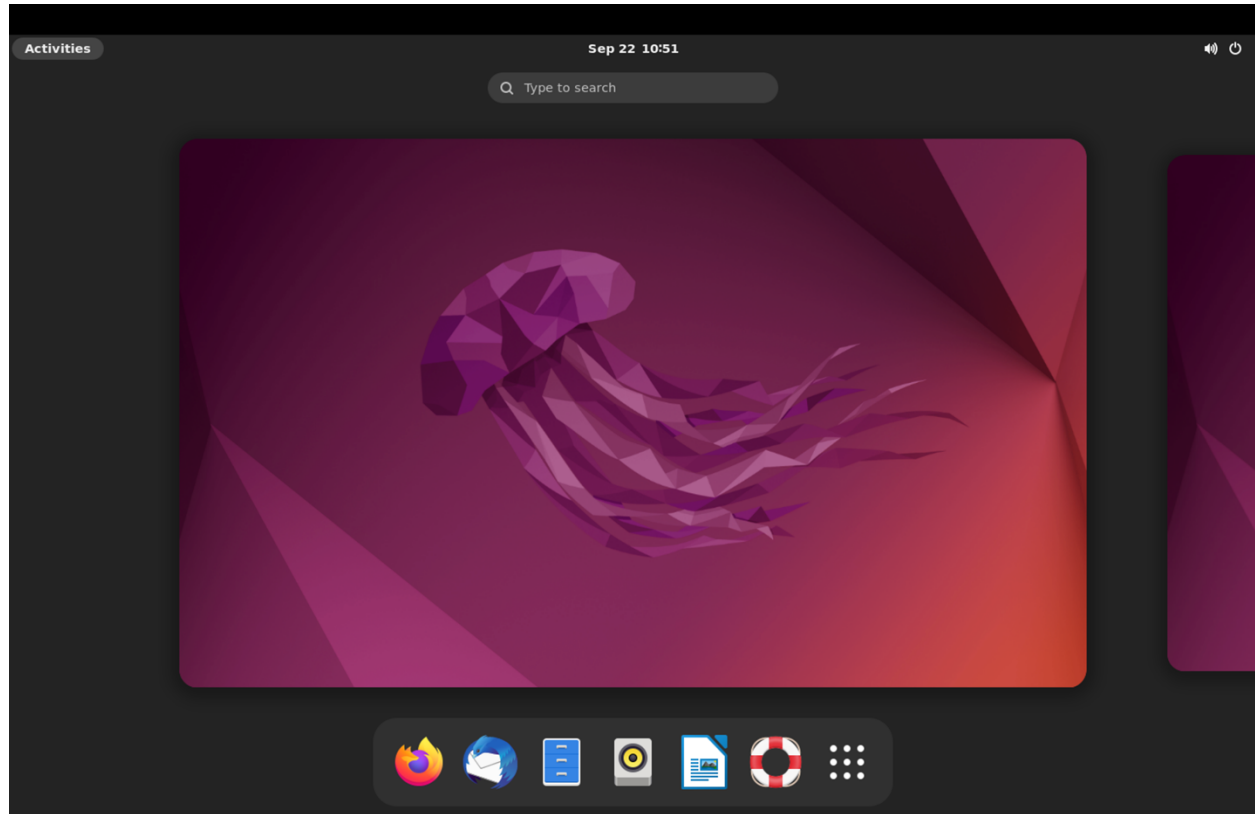
Expanded Security Maintenance for Applications is not enabled.

21 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

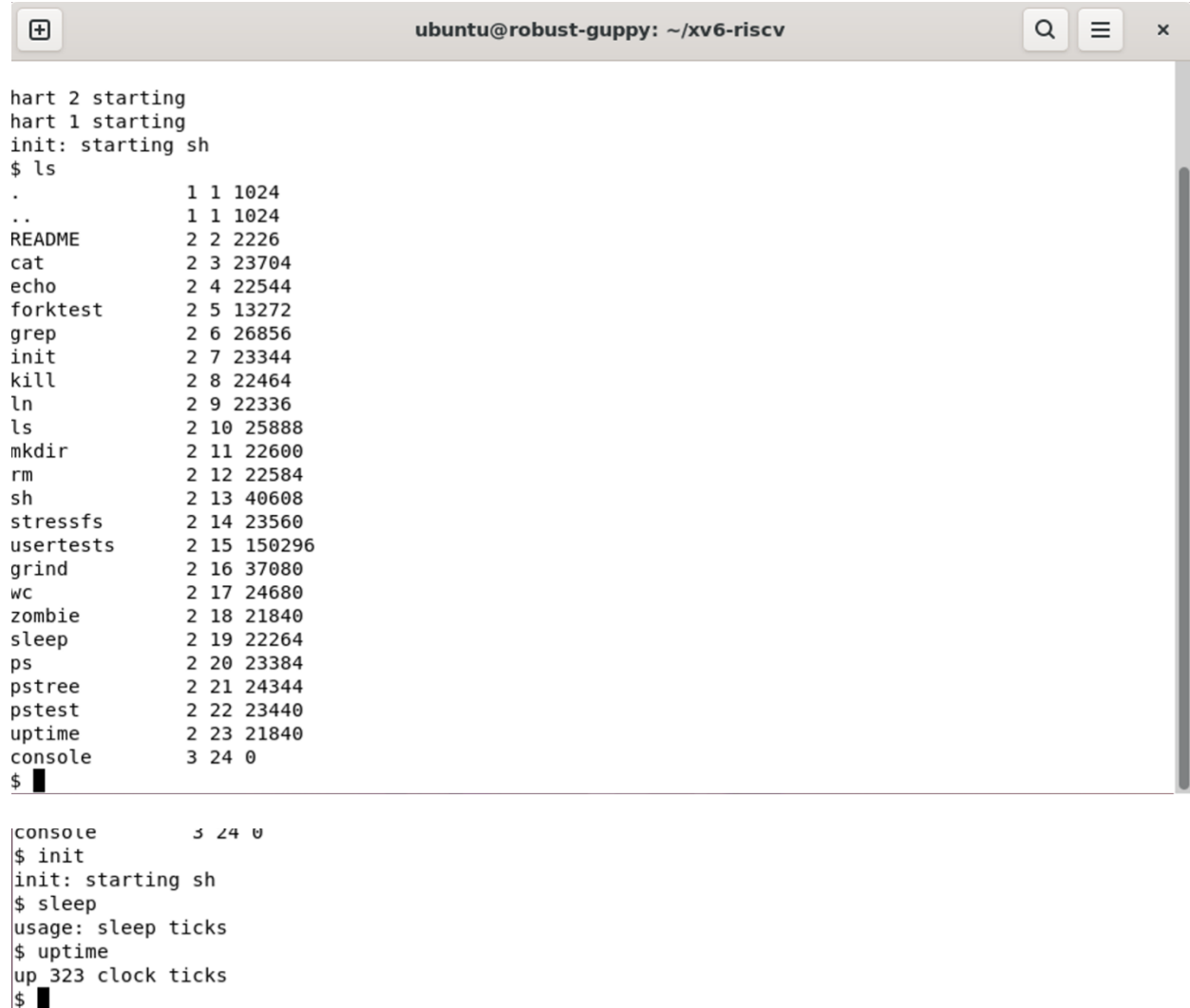
7 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Fri Sep 22 10:30:51 2023 from 192.168.64.1
ubuntu@robust-guppy:~$
```

## Ubuntu running using Microsoft Remote Desktop



After following the instructions for task 1, these were the results:



```

hart 2 starting
hart 1 starting
init: starting sh
$ ls
.          1 1 1024
..         1 1 1024
README    2 2 2226
cat        2 3 23704
echo       2 4 22544
forktest   2 5 13272
grep       2 6 26856
init       2 7 23344
kill       2 8 22464
ln         2 9 22336
ls         2 10 25888
mkdir      2 11 22600
rm         2 12 22584
sh         2 13 40608
stressfs   2 14 23560
usertests  2 15 150296
grind      2 16 37080
wc         2 17 24680
zombie     2 18 21840
sleep      2 19 22264
ps         2 20 23384
pstree     2 21 24344
pstest     2 22 23440
uptime     2 23 21840
console    3 24 0
$ █

console    3 24 0
$ init
init: starting sh
$ sleep
usage: sleep ticks
$ uptime
up 323 clock ticks
$ █

```

Task 2. Implement the uptime utility:

For task 2 we were told that we needed to implement the uptime utility. At first I was going through some confusion, but after reading the assignment I noticed that I needed to have the **uptime.c** file inside my users folder. I then created a file, inside of it, I firstly included my **user.h** file and made a simple print statement in which I called uptime. The picture above shows the results of uptime.