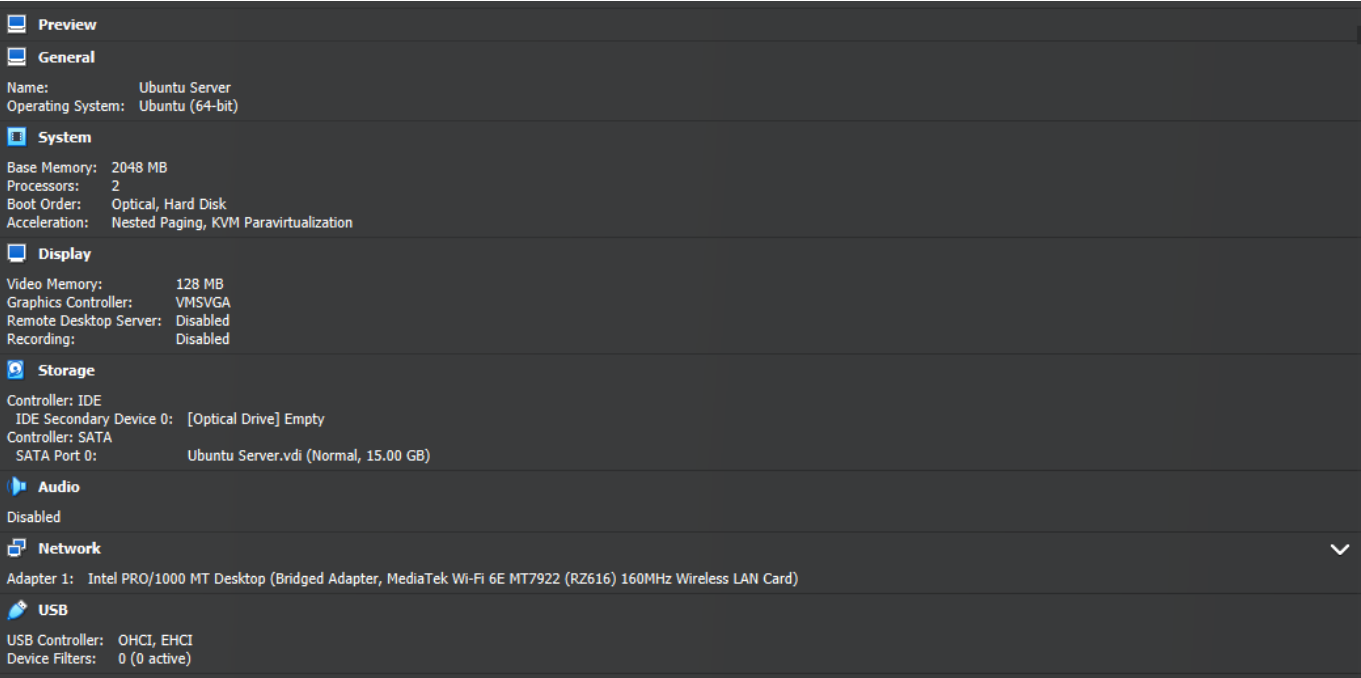


Final Project Deliverable 2

Question 1: What are the server hardware specifications(virtual machine settings)?



Question 2: What is the Ubuntu server login screen



Question 3: What is the IP address of your ubuntu server virtual machine

The IP address of my server virtual machine is 192.168.1.152. This can be found once you are logged into the server next to IPV4 address for enp0s3

```
PS C:\Users\Owner> ssh servernoob@192.168.1.152
servernoob@192.168.1.152's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-105-generic x86_64)

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

System information as of Wed Apr 24 04:41:32 AM UTC 2024

System load:  0.0          Processes:            120
Usage of /:   48.1% of 9.75GB Users logged in:      0
Memory usage: 11%         IPv4 address for enp0s3: 192.168.1.152
Swap usage:   0%

 * 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.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Wed Apr 24 04:41:32 2024
servernoob@projectserver:~$ |
```

Question 4: How do you enable the Ubuntu Firewall?

You can enable the ubuntu firewall by running the command `sudo ufw enable`

```
servernoob@projectserver:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
servernoob@projectserver:~$
```

Question 5: How do you check if the Ubuntu Firewall is running?

You can check if the firewall is running by using the command `sudo ufw status`

```
servernoob@projectserver:~$ sudo ufw status
Status: active

To Action From
--
Apache ALLOW Anywhere
OpenSSH ALLOW Anywhere
Apache (v6) ALLOW Anywhere (v6)
OpenSSH (v6) ALLOW Anywhere (v6)

servernoob@projectserver:~$
```

Question 6: How do you disable Ubuntu Firewall

You can disable the firewall by using the same command used to enable but opposite

```
servernoob@projectserver:~$ sudo ufw disable
Firewall stopped and disabled on system startup
servernoob@projectserver:~$ |
```

Question 7: How do you add apache to the firewall?

You can add apache to the firewall by running the following command `sudo ufw allow 'Apache'`

```
servernoob@projectserver: ~ x Windows PowerShell x + v
servernoob@projectserver:~$ sudo ufw allow 'Apache'
Rule added
Rule added (v6)
servernoob@projectserver:~$ sudo ufw status
Status: active

To Action From
--
OpenSSH ALLOW Anywhere
Apache ALLOW Anywhere
OpenSSH (v6) ALLOW Anywhere (v6)
Apache (v6) ALLOW Anywhere (v6)

servernoob@projectserver:~$
```

Question 8: What is the command you used to install Apache?

The command used to install Apache is `sudo apt install apache2 -y`

```
servernoob@projectserver:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.52-1ubuntu4.9).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
servernoob@projectserver:~$ |
```

Question 9: What is the command you use to check if Apache is running

The command you would use to see if Apache is running is `sudo systemctl status apache2 --no-pager`

```
servernoob@projectserver:~$ systemctl status apache2 --no-pager
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-04-24 05:09:17 UTC; 6min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 2146 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 2150 (apache2)
    Tasks: 55 (limit: 2220)
   Memory: 4.8M
      CPU: 17ms
   CGroup: /system.slice/apache2.service
           └─2150 /usr/sbin/apache2 -k start
             └─2151 /usr/sbin/apache2 -k start
               └─2152 /usr/sbin/apache2 -k start

Apr 24 05:09:17 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:09:17 projectserver systemd[1]: Started The Apache HTTP Server.
servernoob@projectserver:~$
```

Question 10: What is the command you use to stop Apache?

The command you would use to stop Apache is `sudo systemctl stop apache2 --no-pager`

```
servernoob@projectserver:~$ systemctl stop apache2 --no-pager
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to stop 'apache2.service'.
Authenticating as: servernoob
Password:
==== AUTHENTICATION COMPLETE ====
servernoob@projectserver:~$ systemctl status apache2 --no-pager
○ apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Wed 2024-04-24 05:20:12 UTC; 4s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 2387 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Process: 2466 ExecStop=/usr/sbin/apachectl graceful-stop (code=exited, status=0/SUCCESS)
  Main PID: 2391 (code=exited, status=0/SUCCESS)
      CPU: 18ms

Apr 24 05:19:20 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:19:20 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 05:20:12 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 05:20:12 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 05:20:12 projectserver systemd[1]: Stopped The Apache HTTP Server.
servernoob@projectserver:~$
```

Question 11: What is the command you would use to restart Apache?

The command you would use to restart Apache would be `sudo systemctl restart apache2 --no-pager`

```
servernoob@projectserver:~$ sudo systemctl restart apache2 --no-pager
[sudo] password for servernoob:
servernoob@projectserver:~$
```

Question 12: What is the command used to test Apache configuration?

The command used to test for Apache configuration is `sudo apachectl -t`

```
servernoob@projectserver:~$ sudo apachectl -t
Syntax OK
servernoob@projectserver:~$
```

Question 13: What is the command used to check the installed version of Apache?

The command you would use to check for the installed version of Apache is `sudo apache2 -v`

```
servernoob@projectserver:~$ sudo apache2 -v
Server version: Apache/2.4.52 (Ubuntu)
Server built:   2024-04-10T17:45:18
servernoob@projectserver:~$
```

Question 14: What are the most common commands to troubleshoot Apache errors?

The most common commands used to troubleshoot Apache errors are `systemctl`, `journalctl`, and `apachectl`

Descriptions

- Systemctl is the first common command and is used to control and interact with Linux services via the `systemd` service manager.
- Journalctl is the second common command and is used to query and view the logs that are generated by `systemd`.
- Apachectl is the third common command and is used to check Apache's configuration.

Examples:

```

servernoob@projectserver:~$ sudo systemctl status apache2.service -l --no-pager
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-04-24 05:39:05 UTC; 6min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 2560 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 2564 (apache2)
    Tasks: 55 (limit: 2220)
   Memory: 4.8M
      CPU: 16ms
   CGroup: /system.slice/apache2.service
           └─2564 /usr/sbin/apache2 -k start
             └─2565 /usr/sbin/apache2 -k start
               └─2566 /usr/sbin/apache2 -k start

Apr 24 05:39:05 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:39:05 projectserver systemd[1]: Started The Apache HTTP Server.
servernoob@projectserver:~$

```

```

servernoob@projectserver:~$ sudo journalctl -u apache2.service --since today --no-pager
Apr 24 03:23:56 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 03:23:56 projectserver apachectl[1520]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Apr 24 03:23:56 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 03:33:30 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 03:33:30 projectserver apachectl[2133]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Apr 24 03:33:30 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 03:33:30 projectserver systemd[1]: Stopped The Apache HTTP Server.
-- Boot 1aacad18bc2418782e02102aad3c3cc --
Apr 24 03:41:10 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 03:41:10 projectserver apachectl[1963]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Apr 24 03:41:10 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 03:46:57 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 03:46:57 projectserver apachectl[1199]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Apr 24 03:46:57 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 03:46:57 projectserver systemd[1]: Stopped The Apache HTTP Server.
-- Boot 08d6be08a43743299797255e227bfa4b --
Apr 24 03:47:44 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 03:47:44 projectserver apachectl[1727]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Apr 24 03:47:44 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 04:06:59 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 04:06:59 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 04:06:59 projectserver systemd[1]: Stopped The Apache HTTP Server.
Apr 24 04:06:59 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 04:06:59 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 04:09:58 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 04:09:58 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 04:09:58 projectserver systemd[1]: Stopped The Apache HTTP Server.
-- Boot 6e5a2fc90dc5446d88c7b50e0e0d1b78 --
Apr 24 04:14:26 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 04:14:26 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 04:39:51 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 04:39:51 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 04:39:51 projectserver systemd[1]: Stopped The Apache HTTP Server.
-- Boot 22b3c0cd5d8124722b642210d2aeb78 --
Apr 24 04:40:17 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 04:40:17 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 04:41:43 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 04:41:43 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 04:41:43 projectserver systemd[1]: Stopped The Apache HTTP Server.
-- Boot 687cc24871694588b080663dc7c85609 --
Apr 24 04:42:03 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 04:42:03 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 05:08:52 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 05:08:52 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 05:08:52 projectserver systemd[1]: Stopped The Apache HTTP Server.
Apr 24 05:09:17 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:09:17 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 05:18:21 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 05:18:21 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 05:18:21 projectserver systemd[1]: Stopped The Apache HTTP Server.
Apr 24 05:19:20 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:19:20 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 05:20:12 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 05:20:12 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 05:20:12 projectserver systemd[1]: Stopped The Apache HTTP Server.
Apr 24 05:21:15 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:21:15 projectserver systemd[1]: Started The Apache HTTP Server.
Apr 24 05:39:05 projectserver systemd[1]: Stopping The Apache HTTP Server...
Apr 24 05:39:05 projectserver systemd[1]: apache2.service: Deactivated successfully.
Apr 24 05:39:05 projectserver systemd[1]: Stopped The Apache HTTP Server.
Apr 24 05:39:05 projectserver systemd[1]: Starting The Apache HTTP Server...
Apr 24 05:39:05 projectserver systemd[1]: Started The Apache HTTP Server.
servernoob@projectserver:~$

```

```

servernoob@projectserver:~$ sudo apachectl configtest
Syntax OK
servernoob@projectserver:~$

```

Question 15: Which are Apache log Files, and what are they used for?

Apache log Files are a helpful resource for troubleshooting. they are used to indicate a specific error with helpful diagnostic messages and error codes.

```
servernoob@projectserver:~$ sudo tail -n 2 /var/log/apache2/error.log
[Wed Apr 24 05:39:05.889738 2024] [mpm_event:notice] [pid 2564:tid 140375633004416] AH00489: Apache/2.4.52 (Ubuntu) configured -- resuming normal operations
[Wed Apr 24 05:39:05.889776 2024] [core:notice] [pid 2564:tid 140375633004416] AH00094: Command line: '/usr/sbin/apache2'
servernoob@projectserver:~$
```