Al-Aqsa University
Faculty of Computers and IT
Engineering and Networks Dept.

جامعة الأقصى
كلية الحاسبات وتكنولوجيا المعلومات
قسم الهندسة والشبكات

Operating Systems for Intelligent Systems (ENGS2306)

Final Project Assignment

Automating the IoT Monitoring Hub Deployment

Presented to: Eng. Mohammed Alhams

Student name: Marah Amjad Qandeel
Student number: 2320226461

Semester: Second Semester 2024/2025

Project Final:

Automating the IoT Monitoring Hub Deployment

Question

Write a Bash script (deploy_hub.sh) that fully automates the setup of a secure, functional web-based IoT Monitoring Hub
on a fresh Ubuntu system. The script must perform the following tasks:

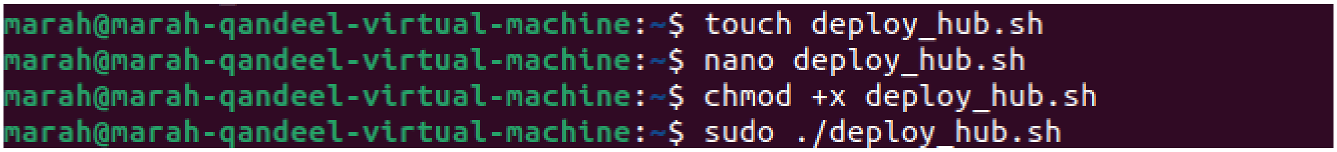
1. Error Handling: exit immediately if any command fails.
2. User & Group Management:
   - Create a group named hub_admins.
   - Create a user named iot_admin.
   - Add the iot_admin user to the hub_admins group.
3. Directory & Permissions Setup:
   - Create /srv/iot_hub directory.
   - Set ownership to iot_admin:hub_admins.
   - Set permissions to 775.
4. Software Installation:
   - Update apt package list.
   - Install nginx web server without prompts.
5. Firewall Configuration:
   - Deny all incoming traffic by default.
   - Allow incoming SSH (22) and HTTP (80).
   - Enable ufw non-interactively.
6. Content Deployment:
   - Create index.html with: <h1>IoT Hub Dashboard - System Normal</h1>.
   - Configure nginx to serve /srv/iot_hub instead of default path.
7. Service Management:
   - Enable nginx at boot.
   - Start/restart nginx.
8. Final Verification:
   - Check if nginx is running.
   - Print green SUCCESS if running, red ERROR if not.

Bash: Deploy_hub.sh

```
#!/bin/bash
#set -e #1-to ensure his immediate exit if any anything fails

#2-user and group mangment
groupadd -f hub_admins
id -u iot_admin &> /dev/null || useradd -m -g hub_admins iot_admin
#3-directory & permissions setup

mkdir -p /srv/iot_hub
chown iot_admin:hub_admins /srv/iot_hub
chmod 755 /srv/iot_hub

#4-softwear installation

apt update
apt install nginx -y

#5-firewall configuration

ufw default deny incoming
ufw default allow outgoing
ufw allow ssh
ufw allow http
echo 'y'|ufw enable
#6-Content Deployment
 echo "<h1> Iot HubDashboard_ System Normal</h1>" | tee /srv/iot_hub/index.html
chown iot_admin:hub_admins /srv/iot_hub/index.html
 chmod 644 /srv/iot_hub/index.html
sed -i 's|/var/www/html|/srv/iot_hub|g' /etc/nginx/sites-available/default
#7-service management
systemctl start nginx
systemctl status nginx
systemctl enable nginx
#8-final verification
 if pgrep nginx &> /dev/null;then
 echo -e"[32mSUCCESS:IoT Hub deployment completed successfully.e[0m"
else
echo -e"[31mERROR: IoT Hub deployment failed.e[0m"
fi
```

IN THE TRIMANL:

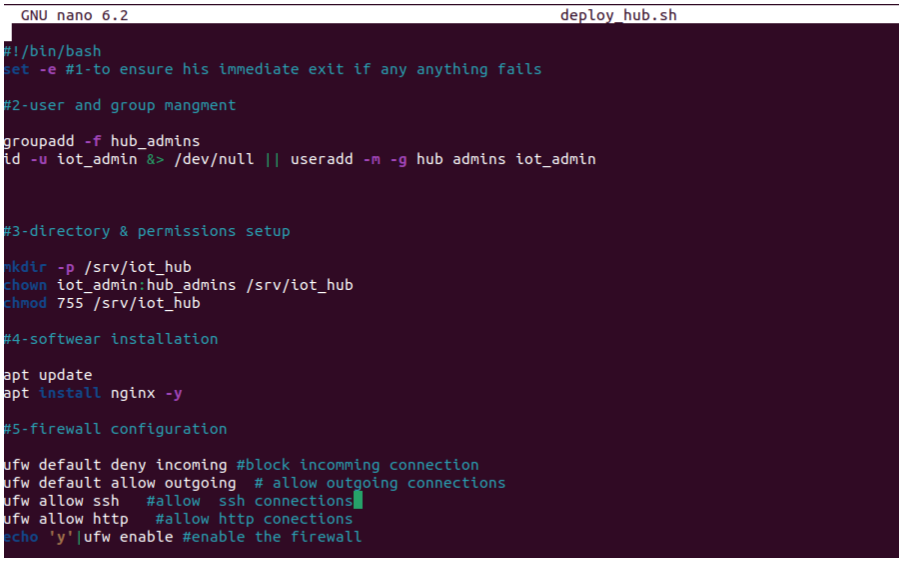
1-Create file deploy_hub.sh



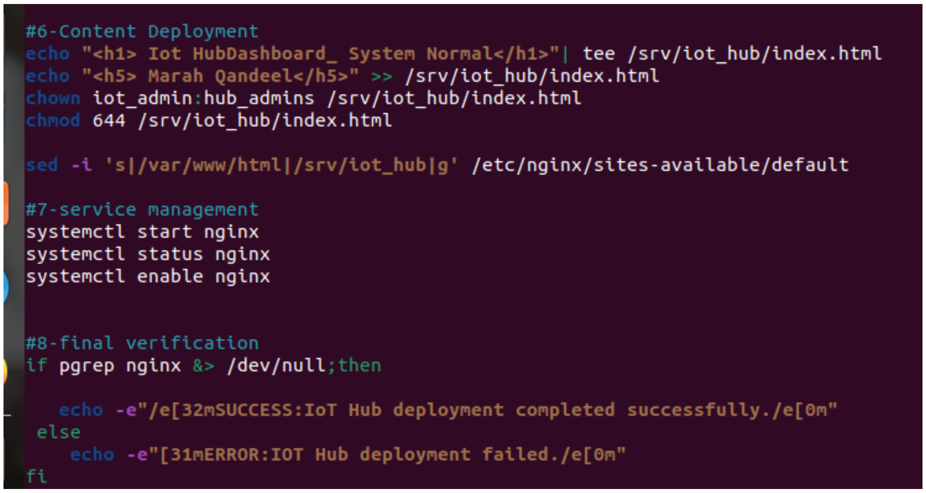
```
marah@marah-qandeel-virtual-machine:~$ touch deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ nano deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ chmod +x deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ sudo ./deploy_hub.sh
```

Create an empty file named deploy_hub.sh using touch
Nano :A text editor that runs directly in the command line. I copied bash into it and saved it using ctrl+o
then enter, then ctrl+x to exit.
Then I changed the file permissions using chmod to execute.
Sudo to execute the file as root instead of typing sudo for each command inside the file.

Writing in the script



```
  GNU nano 6.2                                         deploy_hub.sh

#!/bin/bash
set -e #1-to ensure his immediate exit if any anything fails

#2-user and group mangment

groupadd -f hub_admins
id -u iot_admin &> /dev/null || useradd -m -g hub admins iot_admin



#3-directory & permissions setup

mkdir -p /srv/iot_hub
chown iot_admin:hub_admins /srv/iot_hub
chmod 755 /srv/iot_hub

#4-softwear installation

apt update
apt install nginx -y

#5-firewall configuration

ufw default deny incoming #block incomming connection
ufw default allow outgoing  # allow outgoing connections
ufw allow ssh   #allow  ssh connections
ufw allow http   #allow http conections
echo 'y'|ufw enable #enable the firewall
```



```
#6-Content Deployment
echo "<h1> Iot HubDashboard_ System Normal</h1>"| tee /srv/iot_hub/index.html
echo "<h5> Marah Qandeel</h5>" >> /srv/iot_hub/index.html
chown iot_admin:hub_admins /srv/iot_hub/index.html
chmod 644 /srv/iot_hub/index.html

sed -i 's|/var/www/html|/srv/iot_hub|g' /etc/nginx/sites-available/default

#7-service management
systemctl start nginx
systemctl status nginx
systemctl enable nginx


#8-final verification
if pgrep nginx &> /dev/null;then

   echo -e"/e[32mSUCCESS:IoT Hub deployment completed successfully./e[0m"
 else
    echo -e"[31mERROR:IOT Hub deployment failed./e[0m"
fi
```

4.Explaining the Script

1-Error handing

After I typed #/bing/bash,I started with set _e a to stop the script execution
so that it tells the shell to exit immediately in case of any error

2-user_group

I used groupadd to create a new group.
-f even if the group already exists, no error is displayed and execution continues.
The second line checks whether the iot_admin user already exists. The id -u command searches for the user, and all its output (normal output or errors) is redirected to /dev/null for hiding.
A new user is created using useradd
-m -g creates a local directory and adds the user to the created group.

3-Directory and Permissions:

Creates /srv/iot_hub as a .-p folder. If it doesn't exist, it's created without error.
Changes the new owner (user iot_admin) and the new group (group hub_admins).
Changes permissions to 775, meaning the owner and group have full access, and others can only read and execute.

4-softwear installation:

Using apt update, the list of things I can download is updated.
Apt install-y automatically downloads specific software without asking.
I downloaded nginx, an open-source web server, through it.
A program installed on the server displays web pages to users, such as HTML, CSS, etc.

5- Firewall configuration:

First, I block all incoming traffic (reject incoming traffic) and allow outgoing traffic. Then, I allow SSH so it can access the machine remotely (server management). I also allow HTTP connections to the web server because the machine hosts a website. Finally, I enable the firewall automatically without manual input.

6. Content Deployment:

I created an index.html file with the title "IoT Hub Dashboard - System Normal." It creates a simple HTML web page with a title and a text message.

It sets the owner and permissions for the index.html file.

It allows the web server (Nginx) to read and view the file, but prevents other users from modifying its content.

It modifies the Nginx settings to point to the new folder instead of the default folder by sed -i

7. **Service Management**:

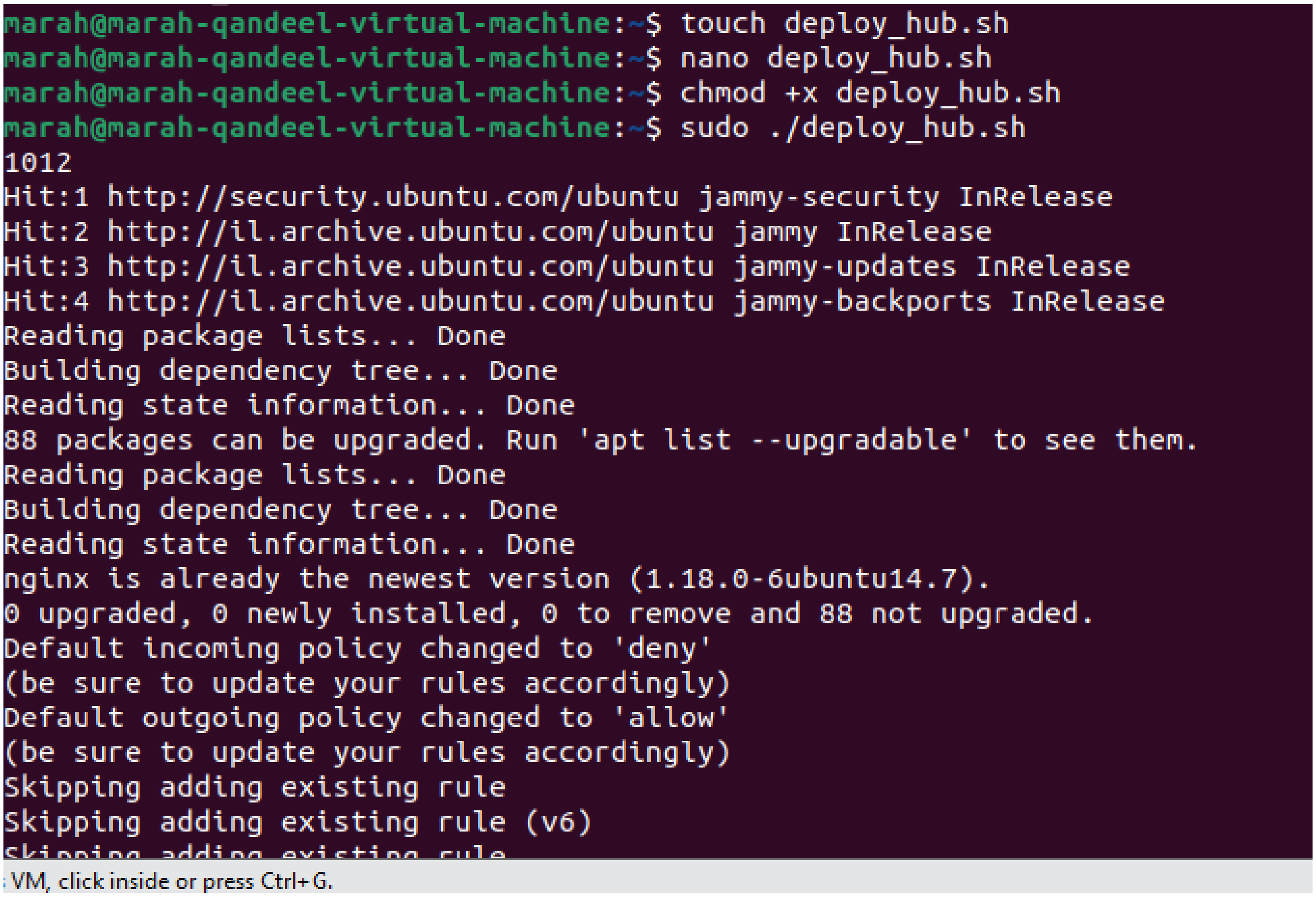
by systemctl (daemon) used to control the Nginx service's lifecycle, ensuring it is running correctly and will start automatically after a system reboot.

8. **Final Verification:**

To verify that the Nginx server is running after completing all the steps, a green success message is displayed. If it's running, a red error message is displayed, making it easier for the user to determine whether the deployment was successful.

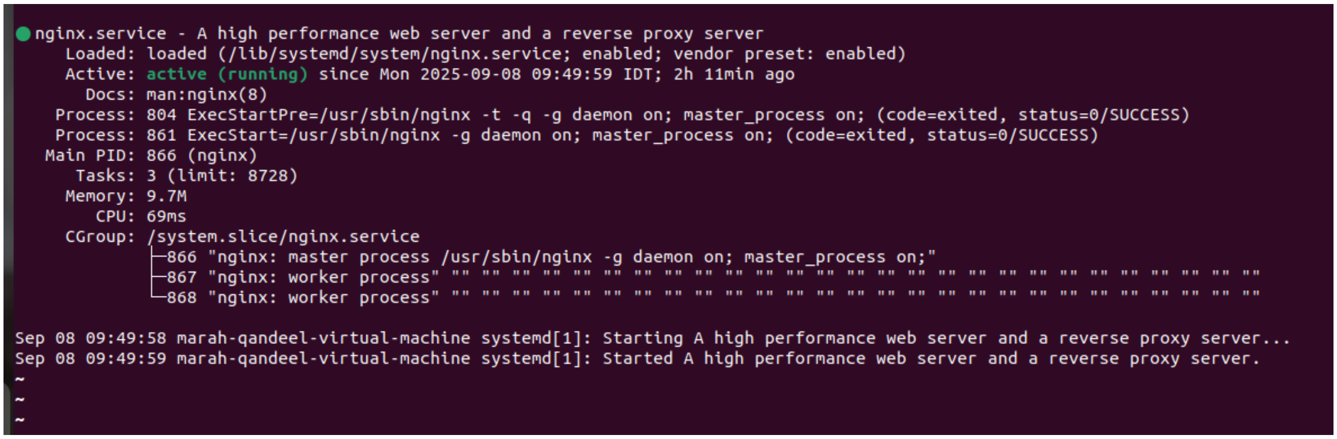
Also use curl http://localhost to verify the webpage.

Running Script



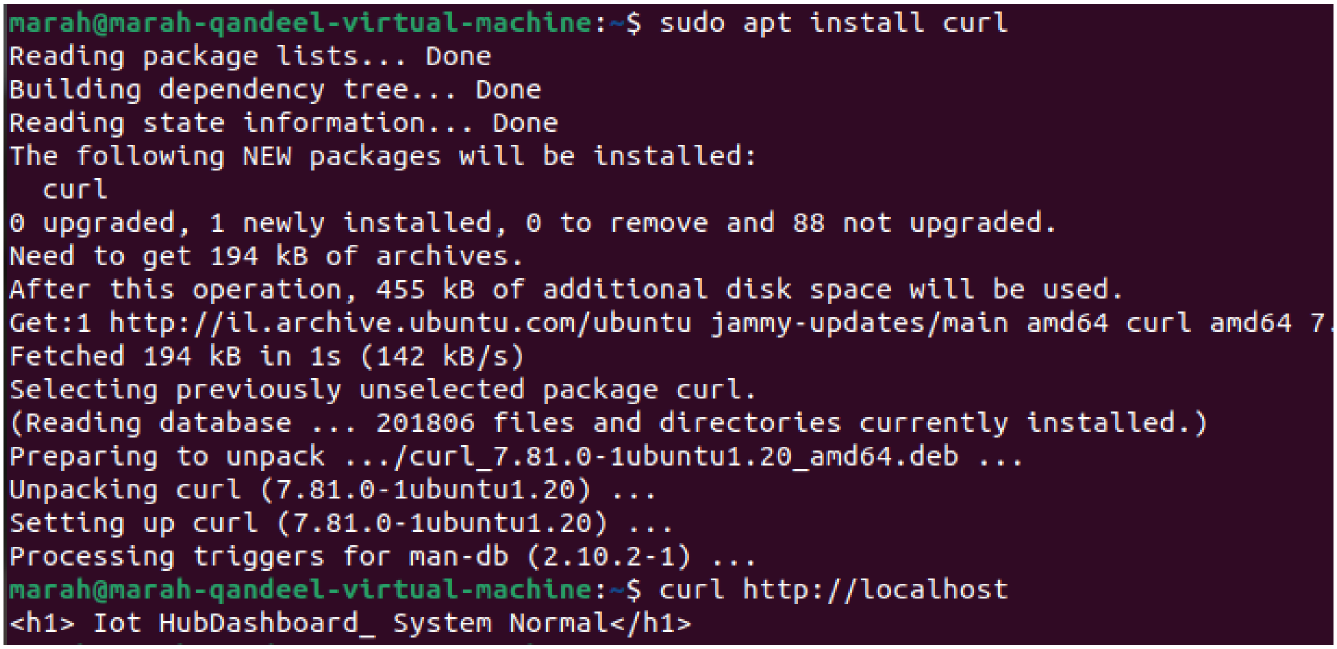
```
marah@marah-qandeel-virtual-machine:~$ touch deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ nano deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ chmod +x deploy_hub.sh
marah@marah-qandeel-virtual-machine:~$ sudo ./deploy_hub.sh
1012
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://il.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://il.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://il.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
88 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.18.0-6ubuntu14.7).
0 upgraded, 0 newly installed, 0 to remove and 88 not upgraded.
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
Skipping adding existing rule
Skipping adding existing rule (v6)
Skipping adding existing rule
```

VM, click inside or press Ctrl+G.

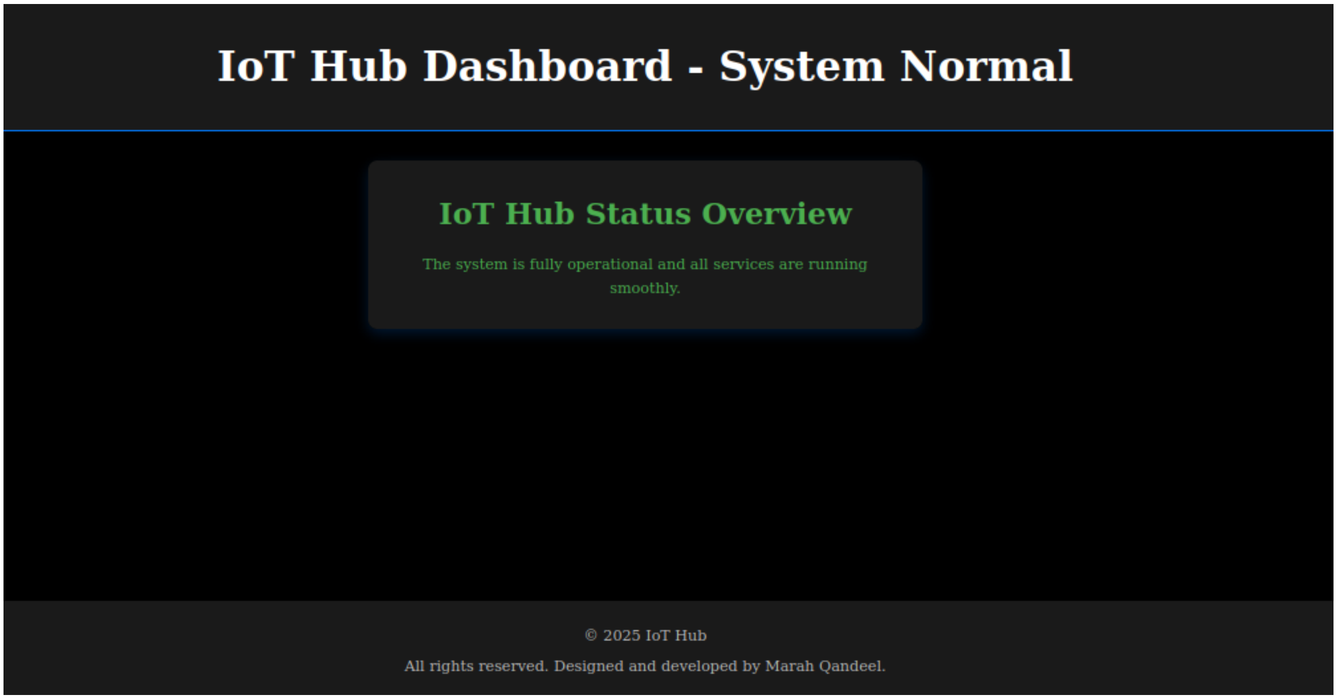


```
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2025-09-08 09:49:59 IDT; 2h 11min ago
       Docs: man:nginx(8)
    Process: 804 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 861 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 866 (nginx)
      Tasks: 3 (limit: 8728)
     Memory: 9.7M
        CPU: 69ms
     CGroup: /system.slice/nginx.service
             ├─866 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─867 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
             └─868 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""

Sep 08 09:49:58 marah-qandeel-virtual-machine systemd[1]: Starting A high performance web server and a reverse proxy server...
Sep 08 09:49:59 marah-qandeel-virtual-machine systemd[1]: Started A high performance web server and a reverse proxy server.
~
~
~
```

curl http://localhost shows the webpage content



```
marah@marah-qandeel-virtual-machine:~$ sudo apt install curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 88 not upgraded.
Need to get 194 kB of archives.
After this operation, 455 kB of additional disk space will be used.
Get:1 http://il.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.
Fetched 194 kB in 1s (142 kB/s)
Selecting previously unselected package curl.
(Reading database ... 201806 files and directories currently installed.)
Preparing to unpack .../curl_7.81.0-1ubuntu1.20_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.20) ...
Setting up curl (7.81.0-1ubuntu1.20) ...
Processing triggers for man-db (2.10.2-1) ...
marah@marah-qandeel-virtual-machine:~$ curl http://localhost
<h1> Iot HubDashboard_ System Normal</h1>
```



IoT Hub Dashboard - System Normal

IoT Hub Status Overview

The system is fully operational and all services are running smoothly.

© 2025 IoT Hub

All rights reserved. Designed and developed by Marah Qandeel.