Dwight Look College of
ENGINEERING
TEXAS A&M UNIVERSITY

Team 21: Load Balancing on an IoT Fog
Bi-Weekly Update 4

Sebastian Correa, Christopher Gonzales

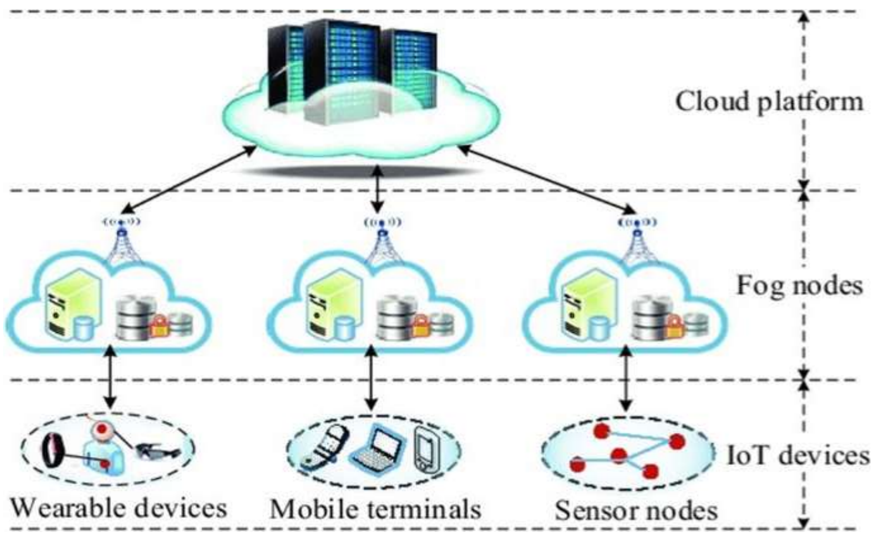
Sponsor: Swarnabha Roy

TA: Zhuwen Hu

Project Summary

Problem statement:

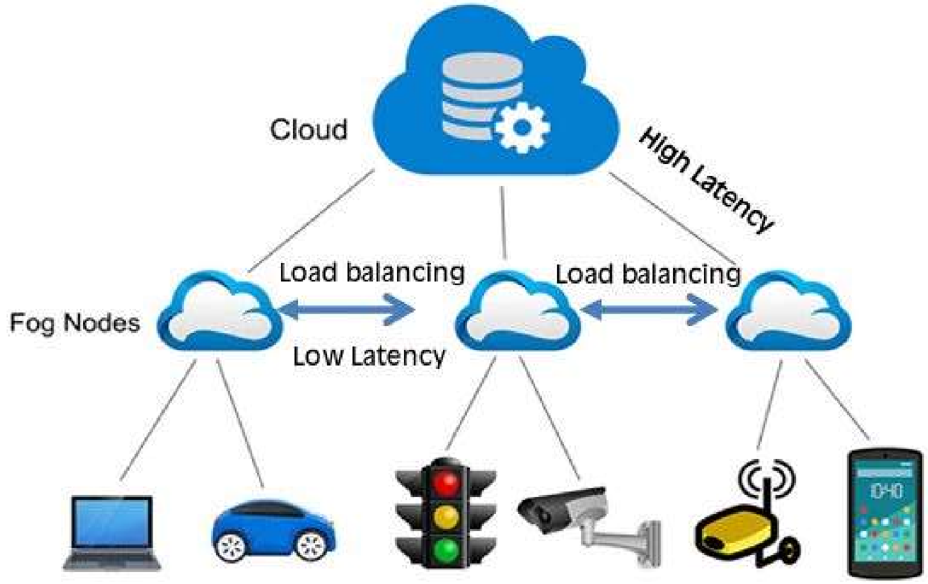
Load balancing only on the cloud is inefficient for applications, overloads virtual machines, and creates scalability issues.



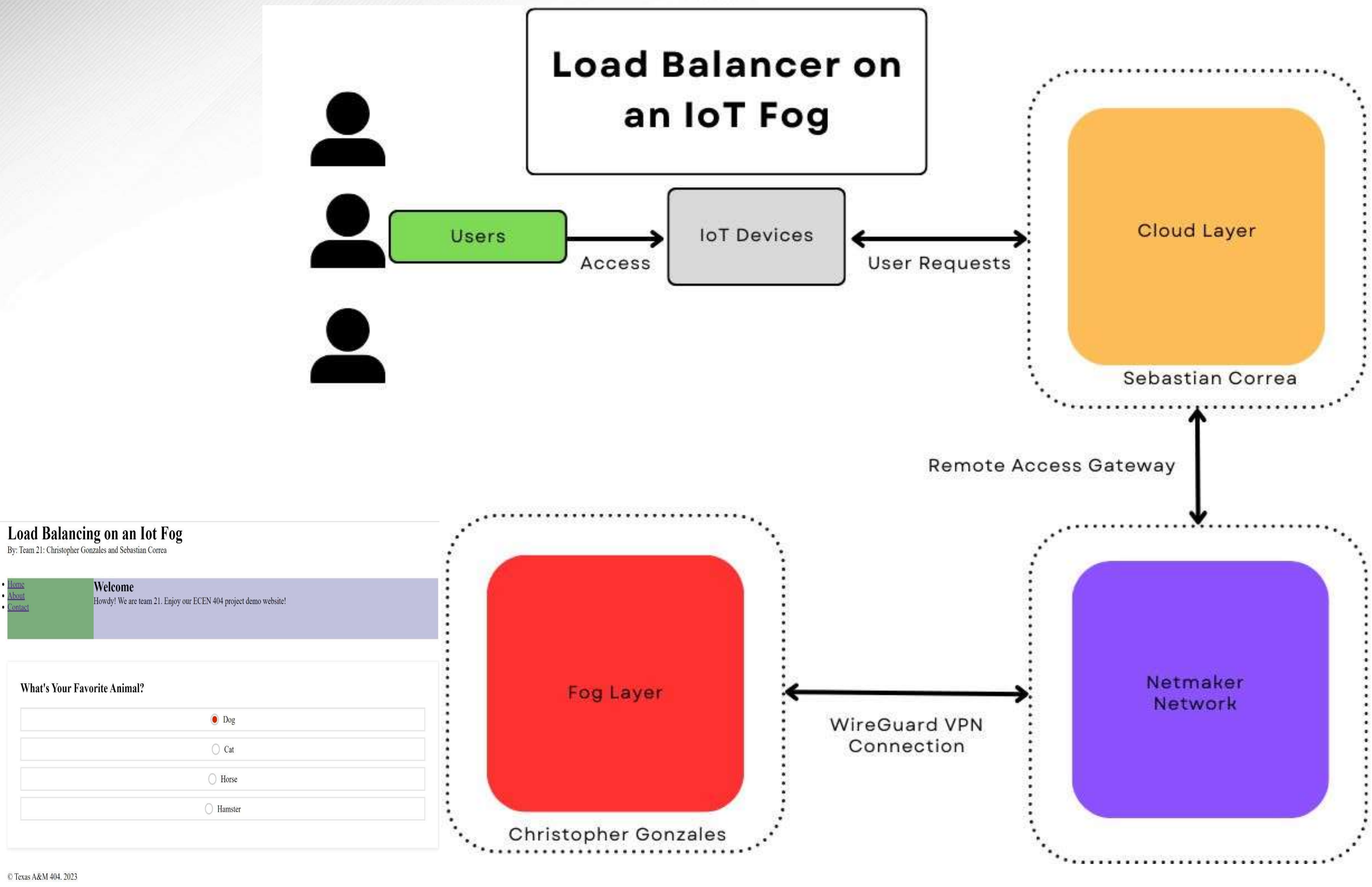
Project Summary

Load-Balancing Solution:

- ○ Takes input from IoT devices
- ○ Distributes the workload
- ○ Sends data to the cloud



Integrated Subsystems Diagram



Load Balancing on an Iot Fog

By: Team 21: Christopher Gonzales and Sebastian Correa

- Home
- About
- Contact

Welcome

Howdy! We are team 21. Enjoy our ECEN 404 project demo website!

What's Your Favorite Animal?

● Dog

○ Cat

○ Horse

○ Hamster

© Texas A&M 404. 2023

Project Timeline

Test Pi's condition and choose new cloud service (completed 8/31/23)	Test Website on Pi and convert VM's to new cloud service (completed 9/14)	Re-establish the connection between the Pi and the cloud (to complete by 9/28)	Finalize Integration (to complete by 10/12)	Test system's capabilities (to complete by 11/2)	Validation (To complete by 11/23)	Demo and Report (To complete by 12/4)

Edge Node Subsystem

Christopher Gonzales

Accomplishments since last update 30 hrs of effort	Ongoing progress/problems and plans until the next presentation
● Reconfigured firewall rules ● Established RPI point-to-point connection ● Established Connection with master node VM ● RPI set as worker node in K3s Cluster ● Installed Wireguard VPN	● Test Fault Tolerance ● Test High Traffic of Container ● Bug Fix Website

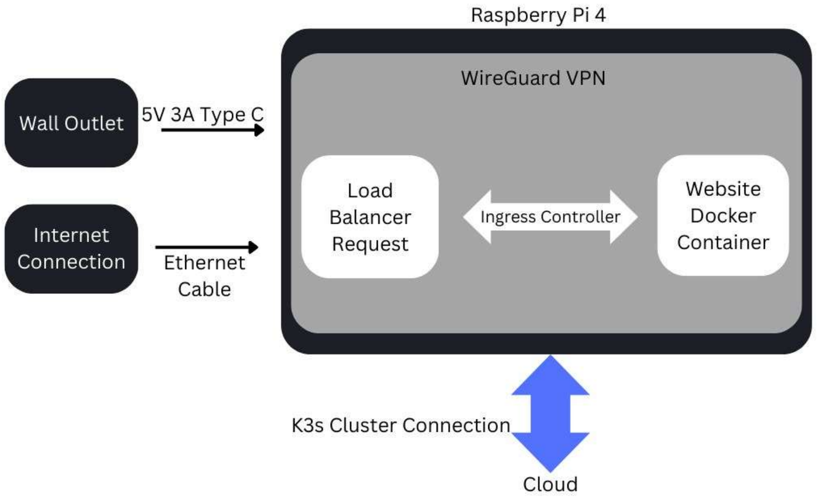
Edge Node Subsystem

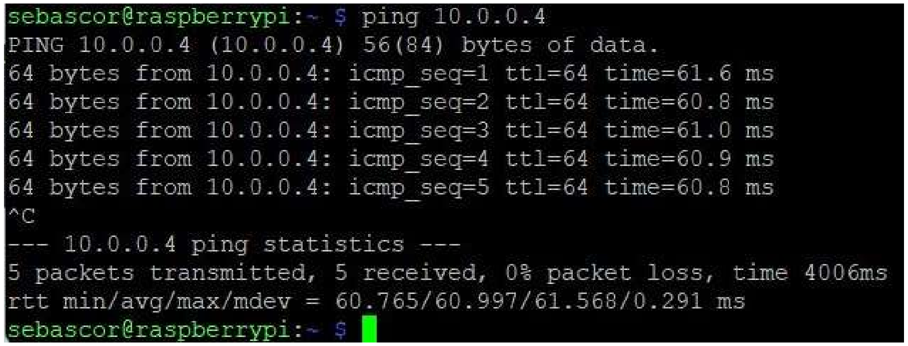
Christopher Gonzales

Function:

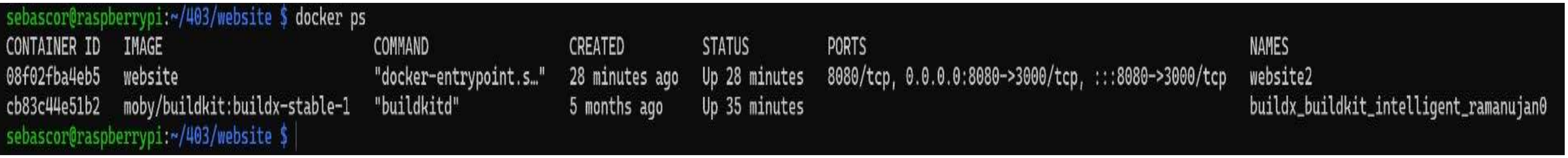
- Pi Takes Requests from cloud through k3s cluster
- Request is directed to website container pod
- Request directed back to cloud and user

- Website Docker Container
- Now able to connect to the master node IP





```
sebascor@raspberrypi:~ $ ping 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=61.6 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=60.8 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=61.0 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=64 time=60.9 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=64 time=60.8 ms
^C
--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 60.765/60.997/61.568/0.291 ms
sebascor@raspberrypi:~ $
```

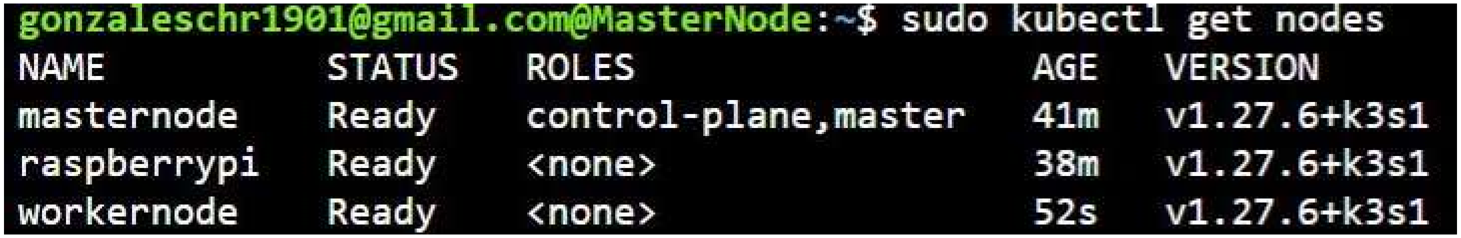


```
sebascor@raspberrypi:~/403/website $ docker ps
CONTAINER ID   IMAGE                           COMMAND                  CREATED          STATUS          PORTS                                               NAMES
08f02fba4eb5   website                         "docker-entrypoint.s…"   28 minutes ago   Up 28 minutes   8080/tcp, 0.0.0.0:8080->3000/tcp, :::8080->3000/tcp   website2
cb83c44e51b2   moby/buildkit:buildx-stable-1   "buildkitd"              5 months ago     Up 35 minutes                                                       buildx_buildkit_intelligent_ramanujan0
sebascor@raspberrypi:~/403/website $
```

Cloud Integration

Sebastian Correa

Accomplishments since last update 60 hrs of effort	Ongoing progress/problems and plans until the next presentation
● Integrated the Raspberry Pi ● Created Point-to-Point Connection ● Converted the Pi into a gateway device ● Created a new Azure account	● Creating a Rancher Dashboard ● Monitoring load balancing ● Testing traffic transmission



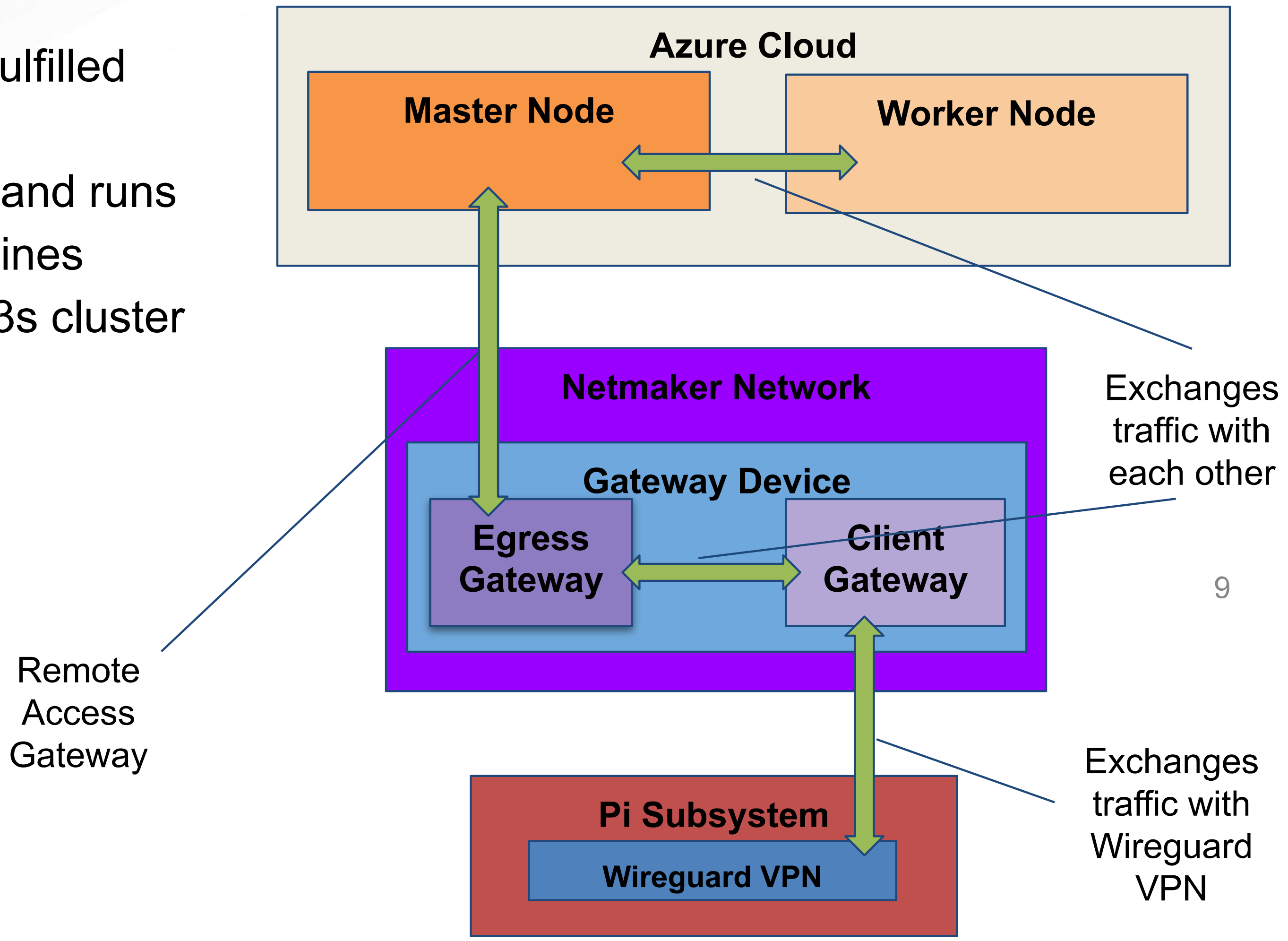
```
gonzaleschr1901@gmail.com@MasterNode:~$ sudo kubectl get nodes
NAME          STATUS    ROLES                   AGE    VERSION
masternode    Ready     control-plane,master    41m    v1.27.6+k3s1
raspberrypi   Ready     <none>                  38m    v1.27.6+k3s1
workernode    Ready     <none>                  52s    v1.27.6+k3s1
```

Cloud Subsystem

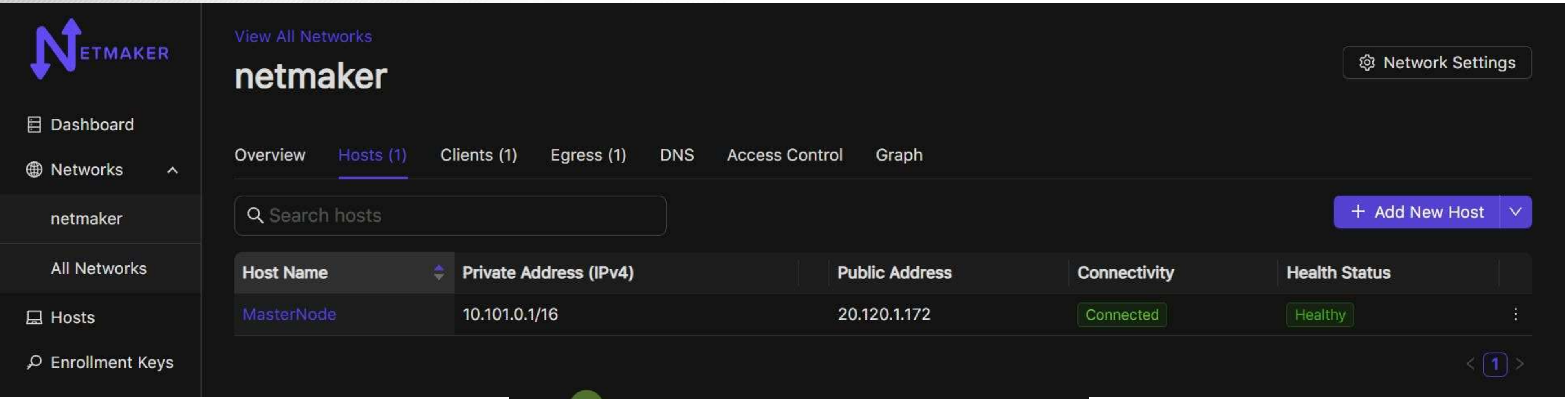
Sebastian Correa

Function:

- Takes in unfulfilled requests
- Stores data and runs virtual machines
- Hosts the K3s cluster

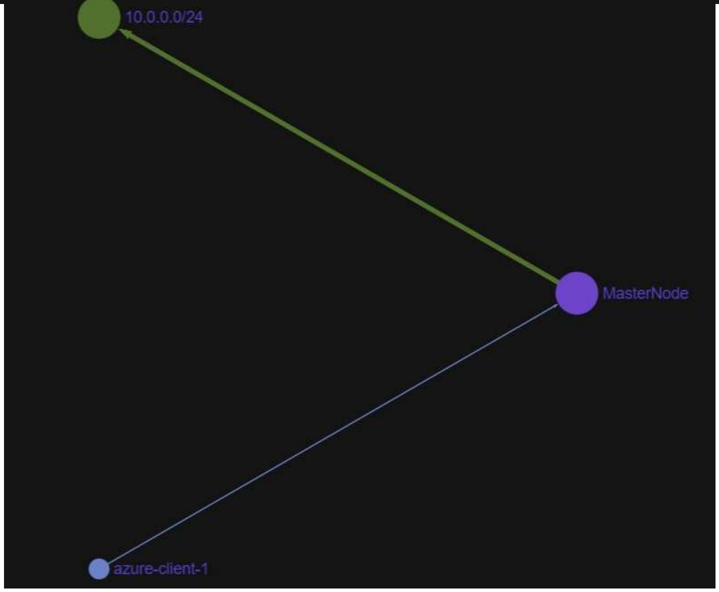


Netmaker Dashboard



netmaker

Host Name	Private Address (IPv4)	Public Address	Connectivity	Health Status
MasterNode	10.101.0.1/16	20.120.1.172	Connected	Healthy



Execution plan

Work	End Date	Owner	Status	Completion Date
Check pi condition	8/24	Christopher	Complete	8/24
Choose new cloud service	8/24	Sebastian	Complete	8/24
Convert all virtual machines to Microsoft Azure	9/7	Sebastian	complete	9/11
Build website	9/7	Christopher	complete	9/7
Install K3s on VMs	9/14	Sebastian	complete	9/14
Test website	9/14	Christopher	complete	9/11
Add more features to the website	9/21	Christopher	Ongoing	9/18
Create connections between VMs	9/21	Sebastian	Complete	9/21
Connect the Pi with the Cloud	9/28	Sebastian	Complete	10/1
Convert Pi connection into worker node	10/5	Sebastian	Ongoing	10/15
Configure containerization in Pi	10/5	Christopher	Complete	10/5
Finalize Integration	10/12	Christopher	Complete	10/15

Execution Plan

Work	End Date	Owner	Status	Completion Date
Test Traffic transmission	10/19	Sebastian	Incomplete	N/A
Demonstrate movement of loads	10/26	Sebastian	Incomplete	N/A
Test Fault Tolerance	11/2	Christopher	Incomplete	N/A
Bug fix the VM’s	11/2	Sebastian	Incomplete	N/A
Bug fix the website	11/2	Christopher	Incomplete	N/A
Configure cloud validation	11/9	Sebastian	Incomplete	N/A
Configure pi validation	11/9	Christopher	Incomplete	N/A
Finalize validation	11/23	Christopher	Incomplete	N/A

Validation plan

Task	Specification	Summary	Result	Owner
Cloud Response Time	<500ms	Amount of time it takes the Cloud to respond to Load Balancer	60-62 ms	Sebastian
Edge Device Runtime	<500ms	Amount of time it takes the Edge Device to respond to Load Balancer	60-62 ms	Christopher
Edge and Cloud Transmission Time	<500ms	Amount of time it takes for the edge device and cloud to respond to one another	60-62 ms	Sebastian
Reading Traffic	<500ms	Amount of time it takes the K3s cluster to read the incoming traffic	Ongoing	Sebastian
Minimum Number of Test Cases	50	50 test cases, or traffic data, being sent to our system for testing	Ongoing	Christopher
Minimum Number of applications to run	1	Our application we’re going to use for testing	Ongoing	Christopher
Load Balancing Test	40%-60%	Ensure that the load is distributed equally between rpi website container and VM container	Ongoing	Sebastian
Failover Test	<500 ms	Shut down one edge node and ensure traffic is redirected to working node within seconds	Ongoing	Christopher

Questions?