Bradley Chang

10.1g: Large Language Models..... 2
5. Document loading..... 2
6. Document searching..... 2
8. -..... 3
11. -..... 4
10.2g: CDN..... 6
6. Deployment..... 6
8. Update deployment..... 10
9. Latency measurements..... 10
16. Test groups..... 11
19. Test load balancer..... 11
20. Siege! (Part 1)..... 11
21. Siege! (Part 2)..... 12

10.1g: Large Language Models

5. Document loading

Explain what the transformer does to the HTML retrieved by the loader

Look for anything related to computer science and security.

Examine the document cleaning code. What kinds of characters are removed when the text is cleaned?

Spaces are removed, and \s+.

What size chunks is the content split into?

10000

What amount of overlap is there between chunks?

1000

How many documents are loaded at a time into the vector database?

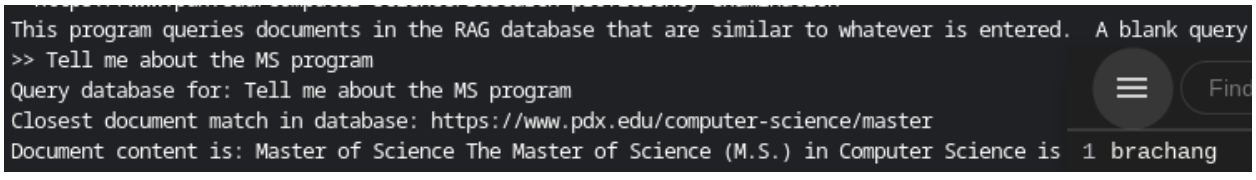
The amount of documents varies and depends upon this function:

```
Unset
def add_documents(vectorstore, chunks, n):
   for i in range(0, len(chunks), n):
       vectorstore.add_documents(chunks[i:i+n])
```

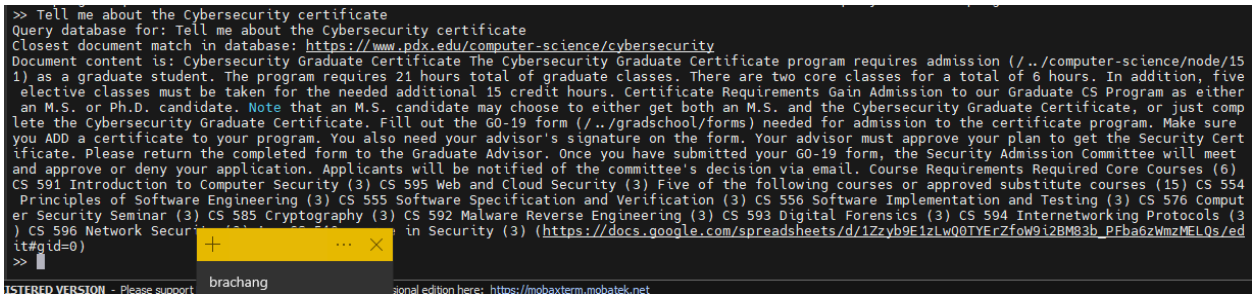
6. Document searching

Show the document URLs that are returned for the following queries:

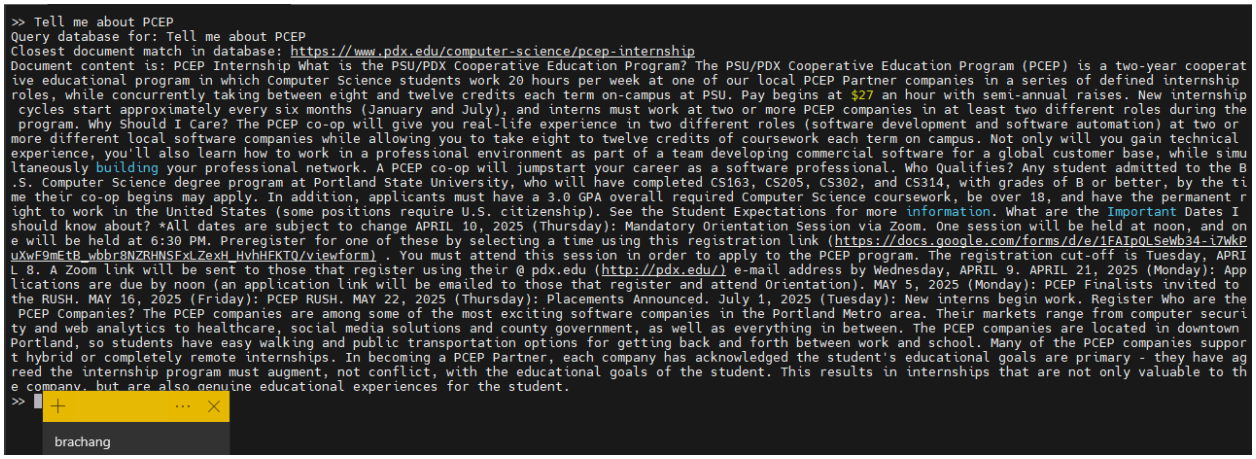
Tell me about the MS program



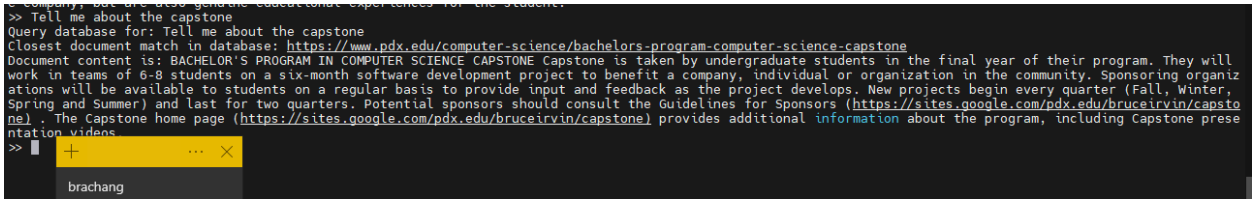
Tell me about the Cybersecurity certificate



Tell me about PCEP

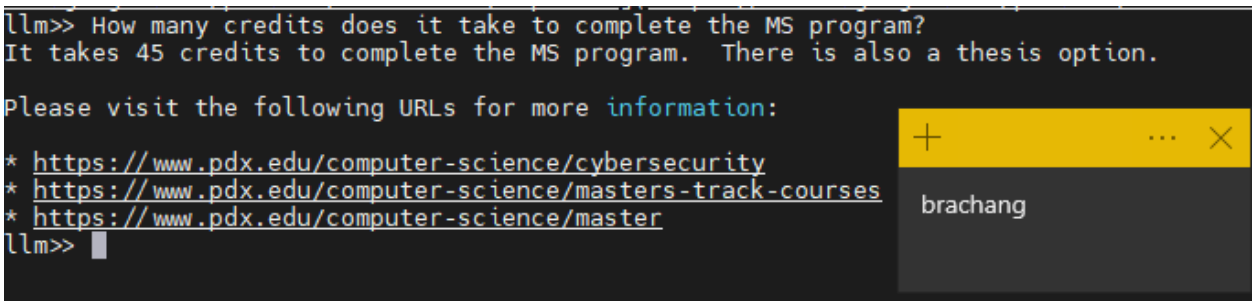


Tell me about the capstone

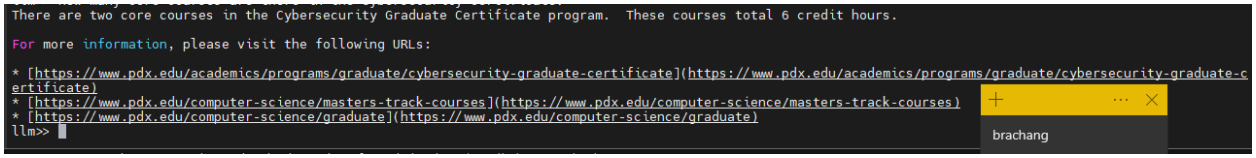


8. -

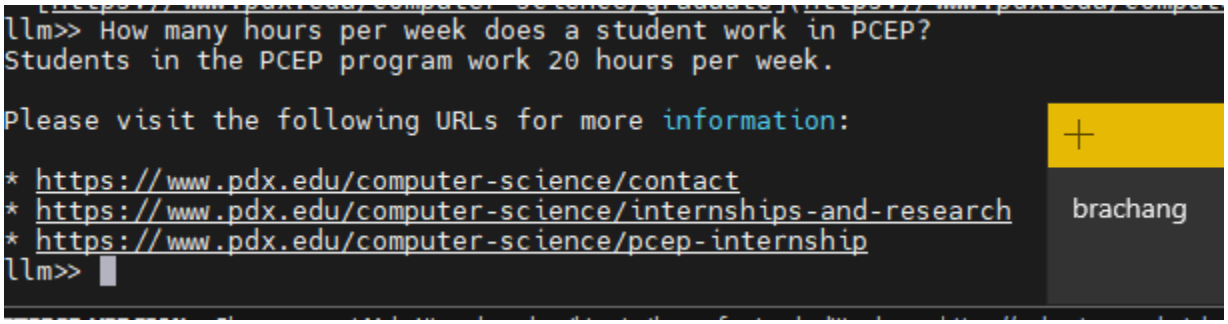
How many credits does it take to complete the MS program?



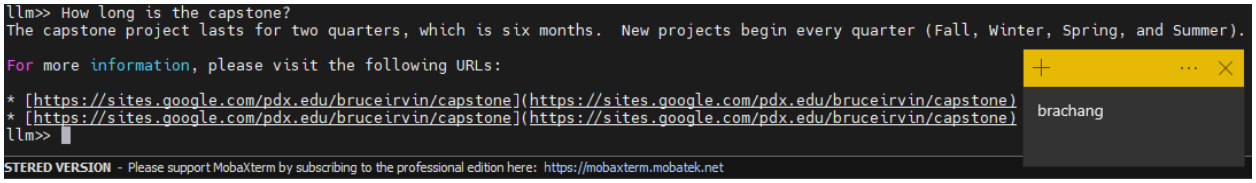
How many core courses are there in the Cybersecurity certificate?



How many hours per week does a student work in PCEP?



How long is the capstone?



11. -

Take a screenshot showing the results of execution

What is the current xkcd?

```
llm>> What is the current xkcd?


> Entering new AgentExecutor chain...
Thought: I should generate a plan to help with this query and then copy that plan exactly to the controller.
Action: api_planner
Action Input: Find the current xkcd comic.  This likely involves a single API call to retrieve the latest comic information.
Observation: 1. GET /info.0.json: This call retrieves the current comic and its metadata.  This fulfills the user's request.
Thought:I'm ready to execute the API call.
Action: api_controller
Action Input: 1. GET /info.0.json

> Entering new AgentExecutor chain...
Plan: 1. GET /info.0.json

Thought: I need to fetch the current comic information using the `/info.0.json` endpoint.  I'll use `requests_get` for this.

Action: requests_get

Action Input:
```json
{
  "url": "http://xkcd.com/info.0.json",
  "params": {},
  "output_instructions": "Extract the 'num', 'title', 'img', 'alt', 'transcript' fields."
}
```
Observation: ```json
{
  "num": 3062,
  "title": "Off By One",
  "img": "https://imgs.xkcd.com/comics/off_by_one.png",
  "alt": "It does come at the small cost of a LOT more off-by-40-or-50 errors.",
  "transcript": ""
}
```
Thought:Thought: I have successfully retrieved the comic information. I can now return the relevant details to the user.

Final Answer: Comic number 3062, titled "Off By One", with image URL "https://imgs.xkcd.com/comics/off_by_one.png", alt text "It does come at the small cost of
 a LOT more off-by-40-or-50 errors.", and an empty transcript.

> Finished chain.

Observation: Comic number 3062, titled "Off By One", with image URL "https://imgs.xkcd.com/comics/off_by_one.png", alt text "It does come at the small cost of
a LOT more off-by-40-or-50 errors.", and an empty transcript.
Thought:I am finished executing a plan and have the information the user asked for.
Final Answer: The current xkcd comic is number 3062, titled "Off By One".  The image URL is https://imgs.xkcd.com/comics/off_by_one.png, the alt text is "It do
es come at the small cost of a LOT more off-by-40-or-50 errors.", and there is no transcript.

> Finished chain.
{'input': 'What is the current xkcd?', 'output': 'The current xkcd comic is number 3062, titled "Off By One".  The image URL is https://imgs.xkcd.com/comics/of
f_by_one.png, the alt text is "It does come at the small cost of a LOT more off-by-40-or-50 errors.", and there is no transcript.'}
llm>>
```


What is the image link of the current xkcd?

```
llm>> What is the image link of the current xkcd?


> Entering new AgentExecutor chain...
Thought: I should generate a plan to help with this query and then copy that plan exactly to the controller.
Action: api_planner
Action Input: I need to find the right API calls to get the image link of the current xkcd comic.  Assuming an API exists with an endpoint to retrieve this inf
ormation.
Observation: Plan:
1. GET /info.0.json: This call retrieves information about the current comic, including its metadata.  We expect this metadata to contain the image link.
Thought:I'm ready to execute the API calls.
Action: api_controller
Action Input: 1. GET /info.0.json

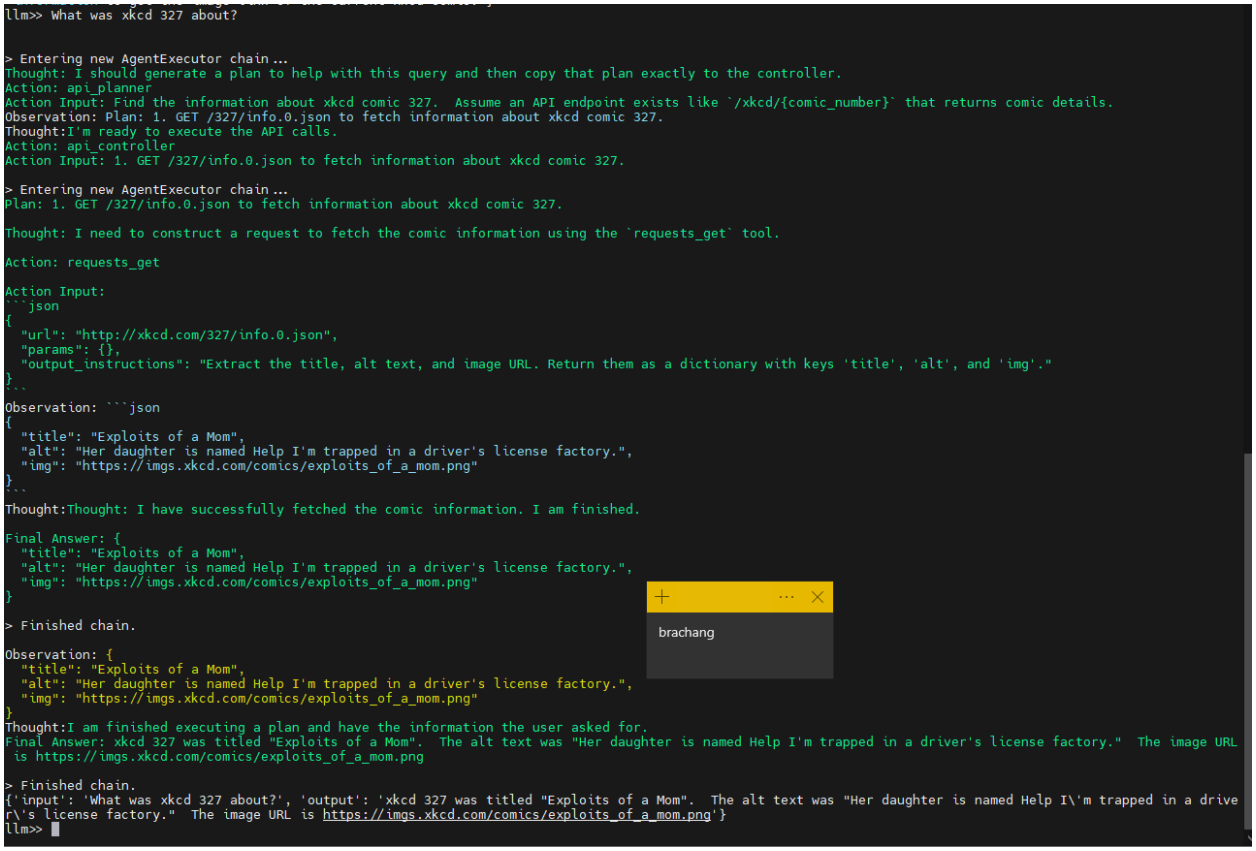
> Entering new AgentExecutor chain...
Plan: 1. GET /info.0.json

Thought: I need to fetch the current comic's information using the `/info.0.json` endpoint.  I'll use `requests_get` for this.

Action: requests_get

Action Input:
```json
{
  "url": "http://xkcd.com/info.0.json",
  "params": {},
  "output_instructions": "Extract the 'num', 'title', 'img', 'alt', 'transcript' fields."
}
```
Observation: ```json
{
  "num": 3062,
  "title": "Off By One",
  "img": "https://imgs.xkcd.com/comics/off_by_one.png",
  "alt": "It does come at the small cost of a LOT more off-by-40-or-50 errors.",
  "transcript": ""
}
```
Thought:Thought: I have successfully retrieved the comic information. I can now present the final answer to the user.

Final Answer: The current XKCD comic is number 3062, titled "Off By One".  The image URL is https://imgs.xkcd.com/comics/off_by_one.png, the alt text is "It do
es come at the small cost of a LOT more off-by-40-or-50 errors.", and the transcript is empty.

> Finished chain.

Observation: The current XKCD comic is number 3062, titled "Off By One".  The image URL is https://imgs.xkcd.com/comics/off_by_one.png, the alt text is "It doe
s come at the small cost of a LOT more off-by-40-or-50 errors.", and the transcript is empty.
Thought:I am finished executing a plan and have the information the user asked for.
Final Answer: The image link of the current xkcd comic is https://imgs.xkcd.com/comics/off_by_one.png

> Finished chain.
{'input': 'What is the image link of the current xkcd?', 'output': 'The image link of the current xkcd comic is https://imgs.xkcd.com/comics/off_by_one.png'}
llm>>
```

What was xkcd 327 about?



10.2g: CDN

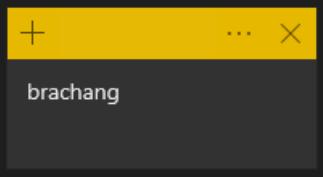
6. Deployment

Take a screenshot of the output to include in your lab notebook. How many networks, subnetworks, and VM instances have been created?

```
brachang@cloudshell:~/networking101 (cloud-chang-brachang)$ gcloud deployment-manager deployments create networking101 --config networking-lab.yaml
The fingerprint of the deployment is b'hjdWgE-1UTaH4b2cqfvYPw=='
Waiting for create [operation-1741911438489-630425c85797b-81ddbc89-54f13477]...done.
WARNING: Create operation operation-1741911438489-630425c85797b-81ddbc89-54f13477 completed with warnings:
---
code: EXTERNAL_API_WARNING
message: You are creating a global DNS VM. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: You are creating a global DNS VM. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: You are creating a global DNS VM. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: You are creating a global DNS VM. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: You are creating a global DNS VM. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
```

brachang

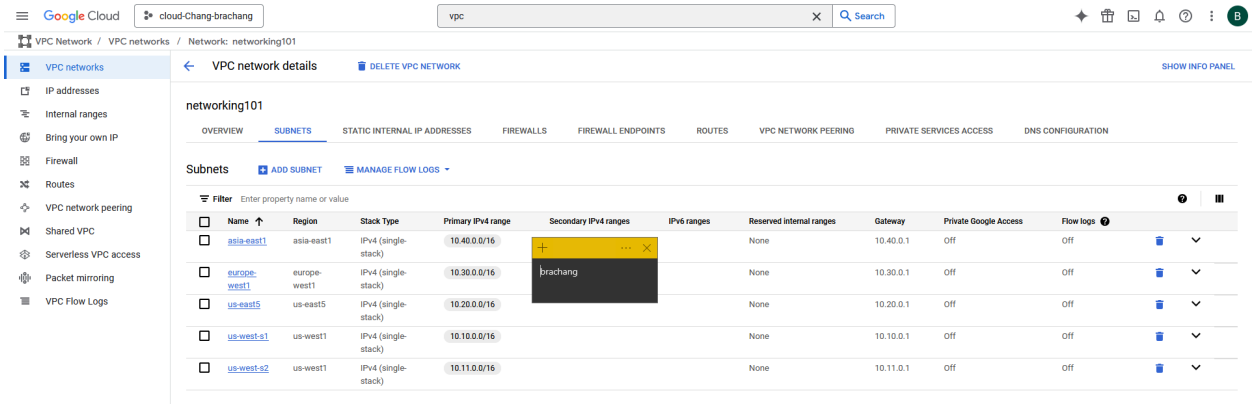
```
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns
---
code: EXTERNAL_API_WARNING
message: Your VM is using global DNS. VM instances using global DNS are vulnerable
  to cross-regional outages. To reduce the risk of widespread service disruption,
  use zonal DNS instead. Learn more at https://cloud.google.com/compute/docs/networking/zonal-dns

NAME: asia-east1
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: asia1-vm
TYPE: compute.v1.instance
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: e1-vm
TYPE: compute.v1.instance
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: eu1-vm
TYPE: compute.v1.instance
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: europe-west1
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: networking101
TYPE: compute.v1.network
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: us-east5
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: us-west-s1
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
```



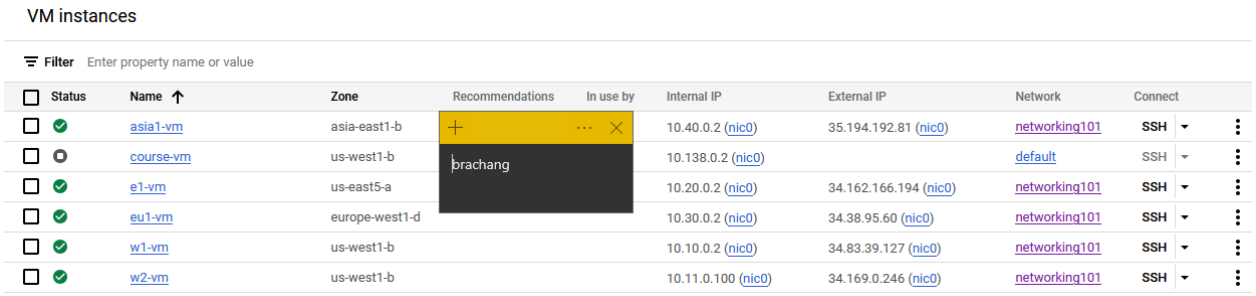
```
NAME: us-west-s1
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: us-west-s2
TYPE: compute.v1.subnetwork
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: w1-vm
TYPE: compute.v1.instance
STATE: COMPLETED
ERRORS: []
INTENT:

NAME: w2-vm
TYPE: compute.v1.instance
STATE: COMPLETED
ERRORS: []
INTENT:
brachang@cloudshell:~/networking101 (cloud-chang-brachang)$
```

brachang

Visit the web console for VPC network and show the network and the subnetworks that have been created. Validate that it has created the infrastructure in the initial figure. Note the lack of firewall rules that have been created.



Visit the web console for Compute Engine and show all VMs that have been created, their internal IP addresses and the subnetworks they have been instantiated on. Validate that it has created the infrastructure shown in the initial figure.

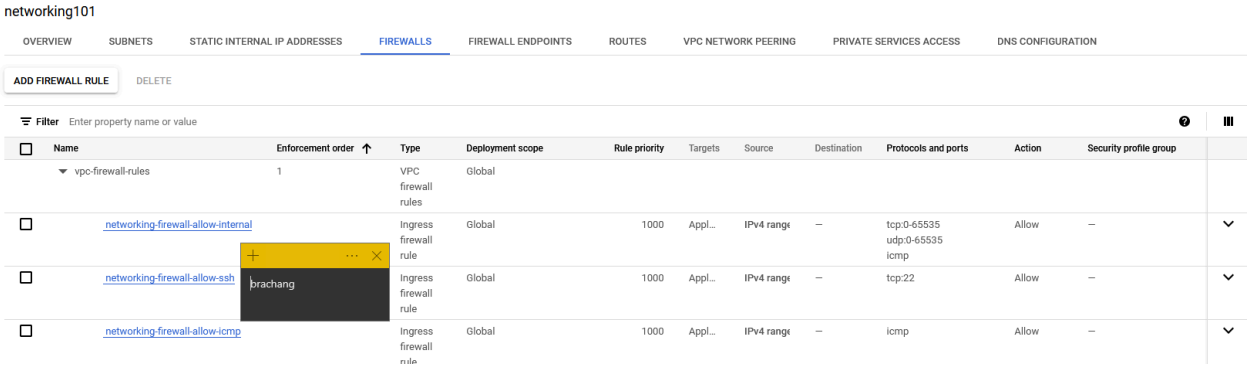


Click on the ssh button for one of the VMs and attempt to connect. Did it succeed?

No

8. Update deployment

Take a screenshot that indicates the new rules have been deployed



9. Latency measurements

Given this, fill in the table with the measured latencies between the 6 pairs and include it in your lab notebook. Use the shortest latency measured for each pair.

Location pair	Ideal latency	Measured latency
us-west1 us-east5	~45 ms	64.649 ms
us-west1 europe-west1	~93 ms	134.249 ms
us-west1 asia-east1	~114 ms	116.611ms
us-east5 europe-west1	~76 ms	87.559 ms
us-east5 asia-east1	~141 ms	241.821 ms
europe-west1 asia-east1	~110 ms	261.807 ms

16. Test groups

Are the instances in the same availability zone or in different ones?

In different zones

List all availability zones that your servers show up in for your lab notebook.

us-east5-a

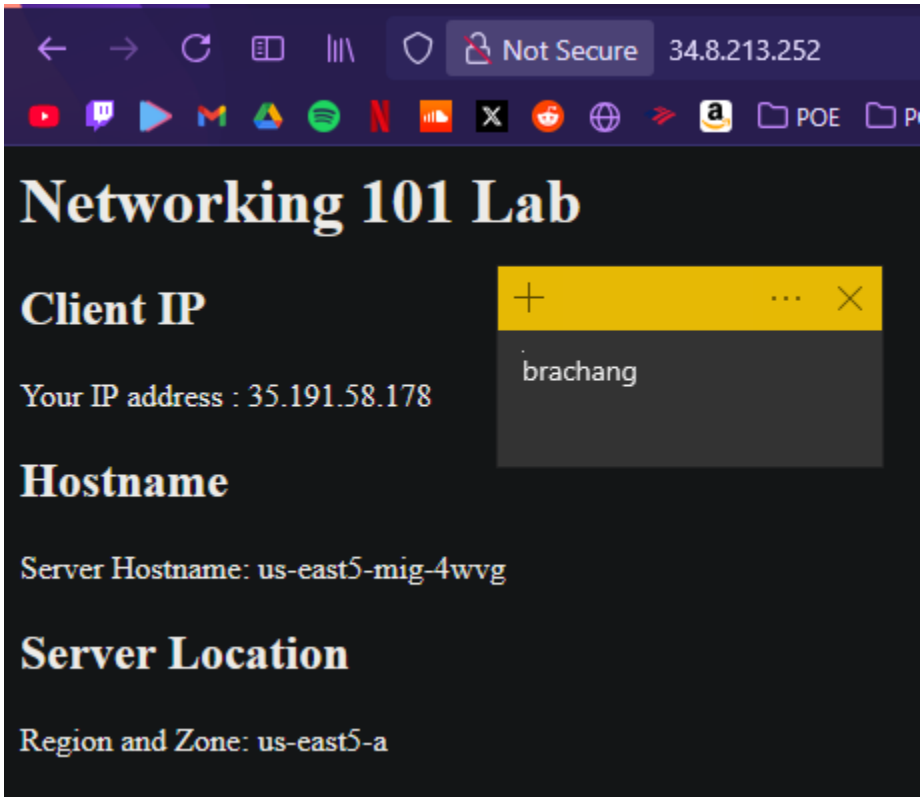
europe-west1-b

europe-west1-d

europe-west1-c

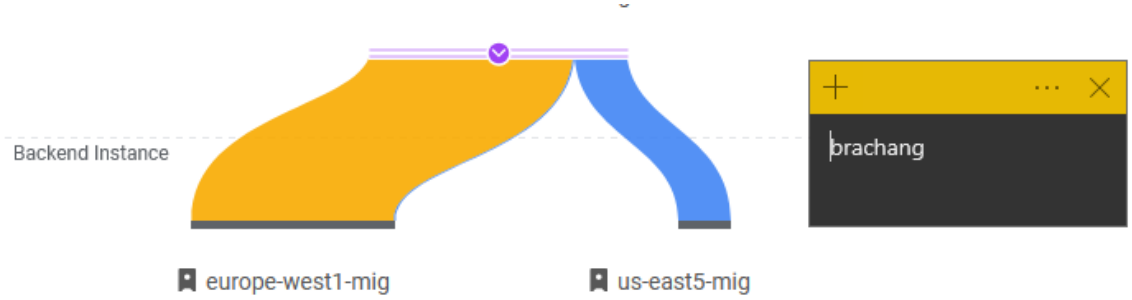
19. Test load balancer

Show a screenshot of the page that is returned.

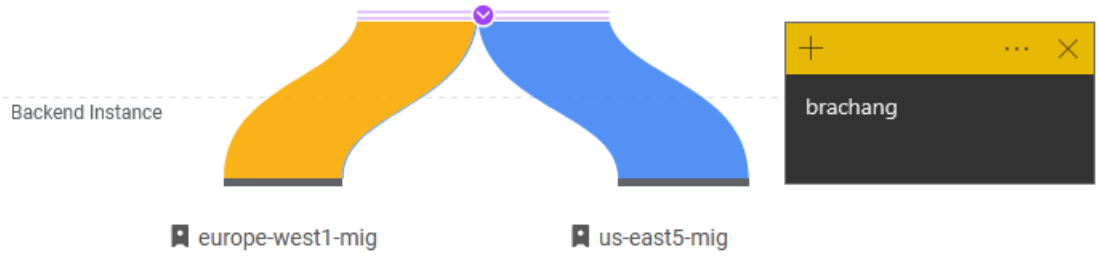


20. Siege! (Part 1)

Take a screenshot of the initial traffic distribution

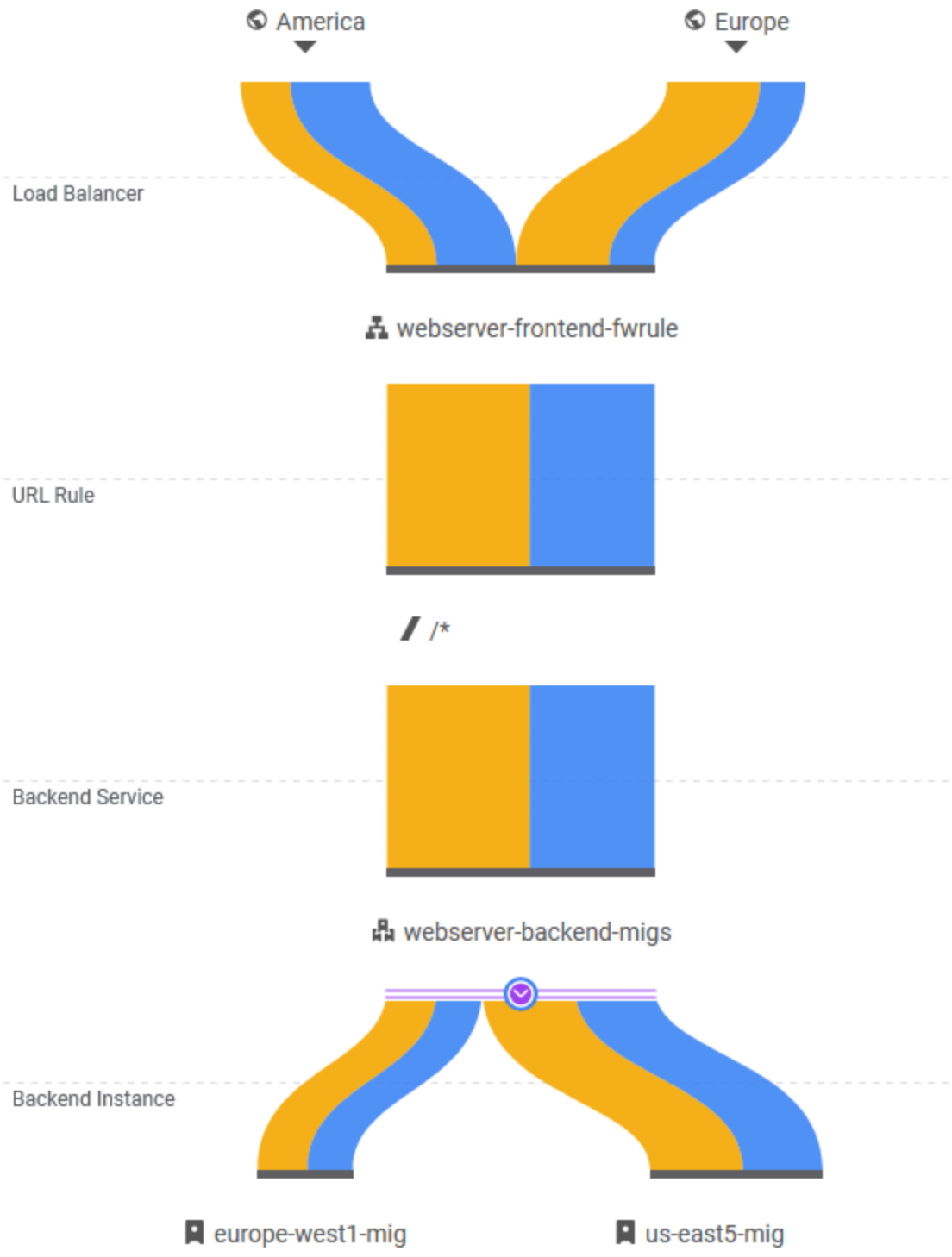


Take a screenshot of the UI as additional instances are brought up and show that the traffic distribution shifts



21. Siege! (Part 2)

Show a screenshot of the final traffic distribution.


America
Europe
Load Balancer
webserver-frontend-fwrule
URL Rule
/*
Backend Service
webserver-backend-migs
Backend Instance
europe-west1-mig
us-east5-mig