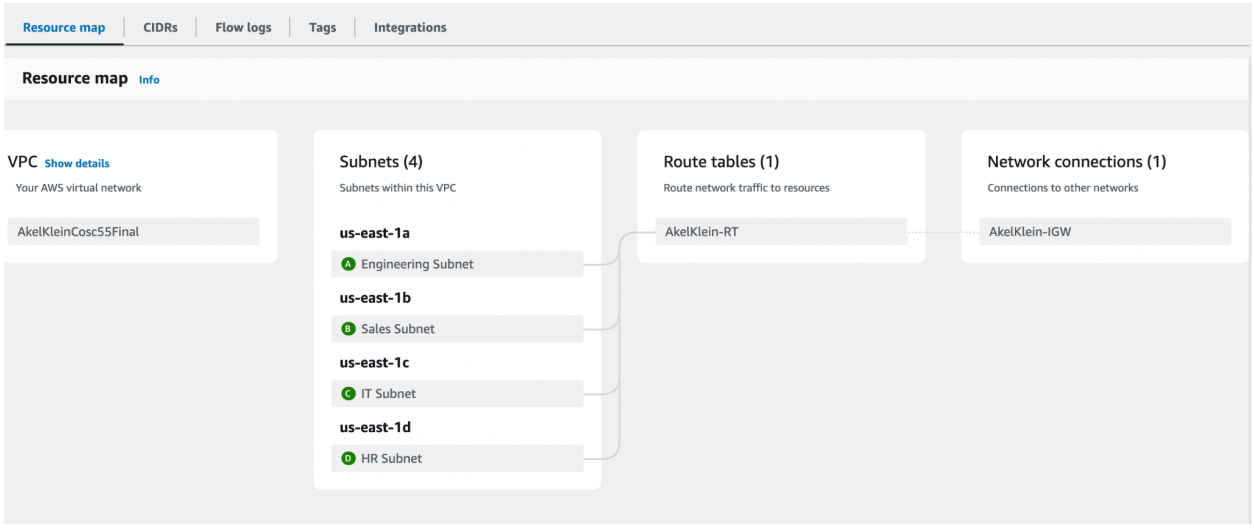
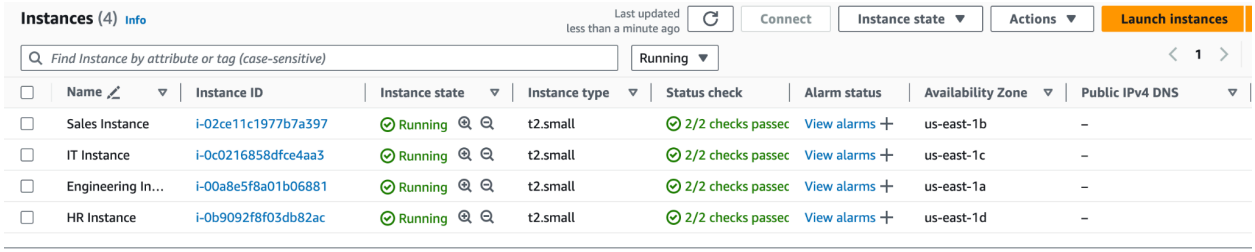
Task 1: Detailed Documentation of Cloud Security Solution

Process, Setup, and Configuration

1. Create VPC
2. Create 4 different subnets within VPC (each being from a separate availability zone to allow for use of ALB)
    a. Engineering Subnet
    b. Sales Subnet
    c. IT Subnet
    d. HR Subnet
3. Create IGW to allow for access to internet from VPC
4. Create Routing table to connect subnets

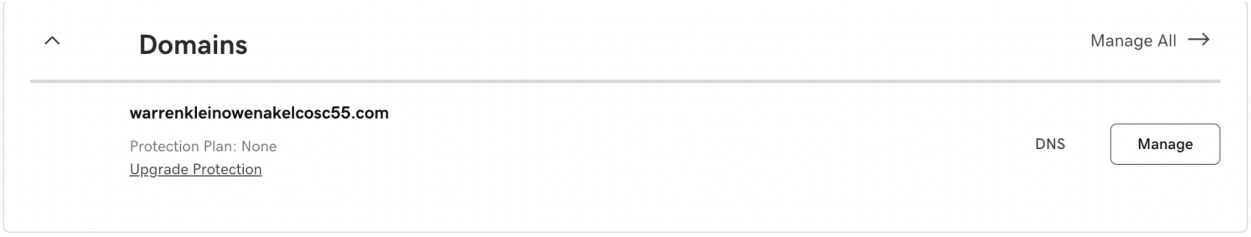


5. Initiate 4 instances → each connected to respective subnets

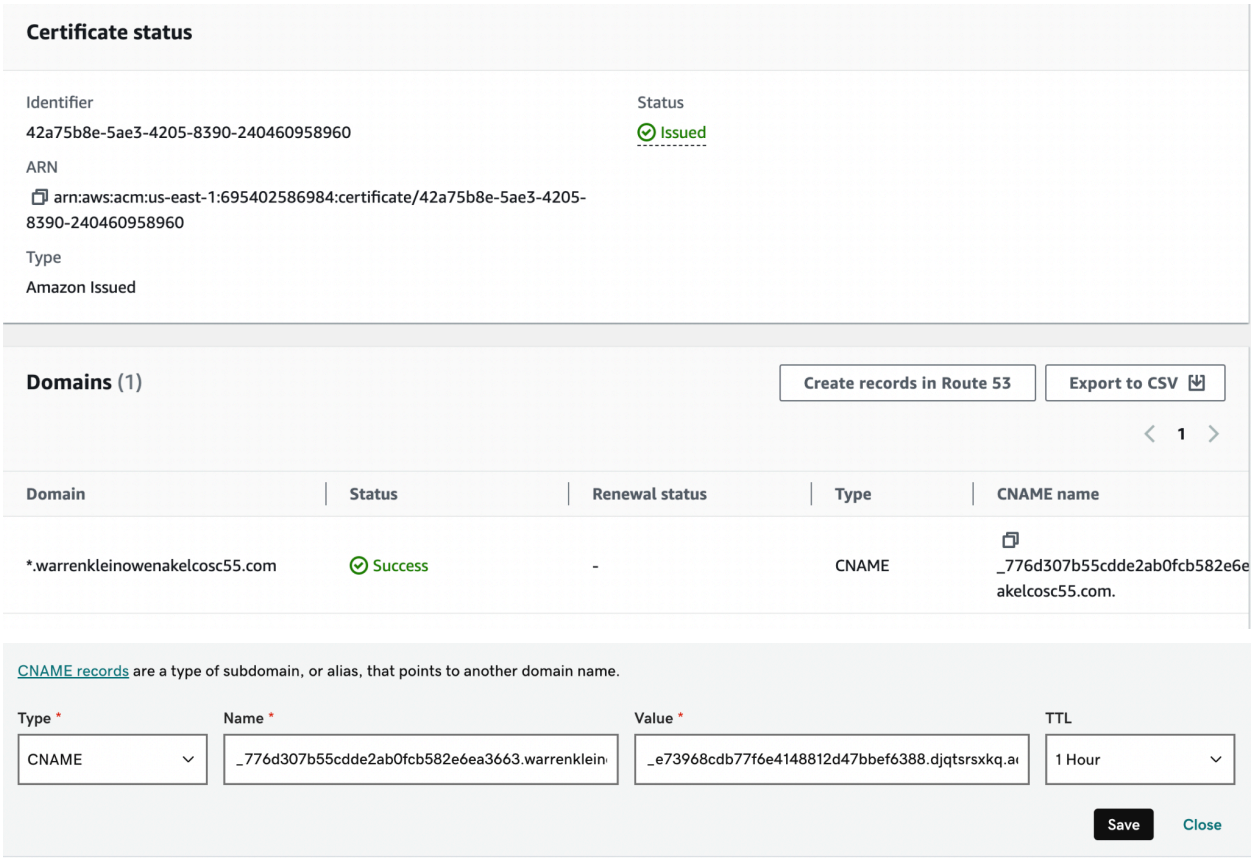


    a. Engineering Instance (Ubuntu t2.small)
    b. Sales Instance (Ubuntu t2.small)
    c. IT Instance (Ubuntu t2.small)
    d. HR Instance (Ubuntu t2.small)

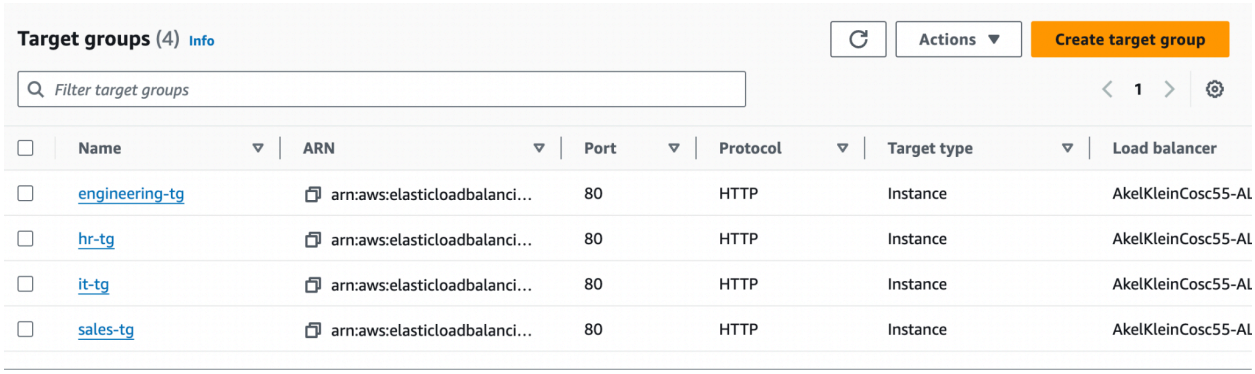
6. Register a GoDaddy Domain (to be used as central login console)



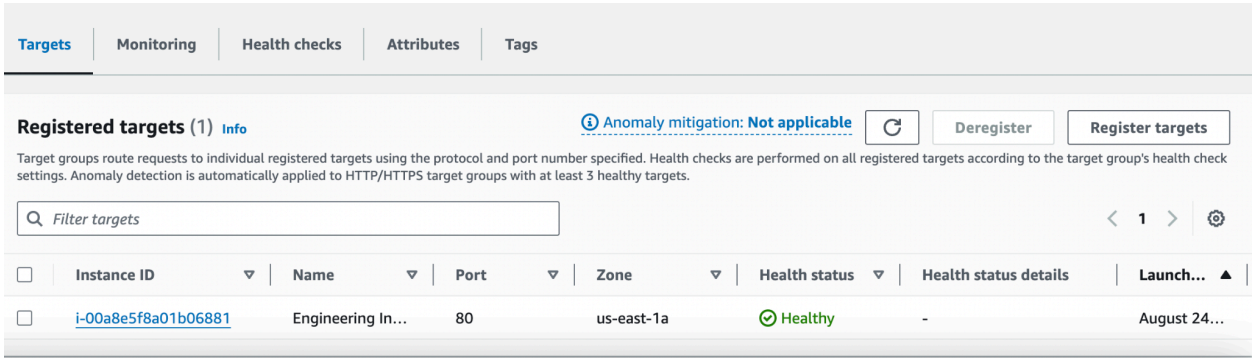
7. Create a wildcard SSL/TLS certificate in order to allow for encryption of subdomains (Source: https://medium.com/@samuelnnanna71/a-guide-to-obtaining-a-public-ssl-certificate-for-your-godaddy-domain-b6869c50f625)



8. Create Target Groups to be used for Application Load Balancer (Use HTTP protocol as cannot connect directly to port 443)

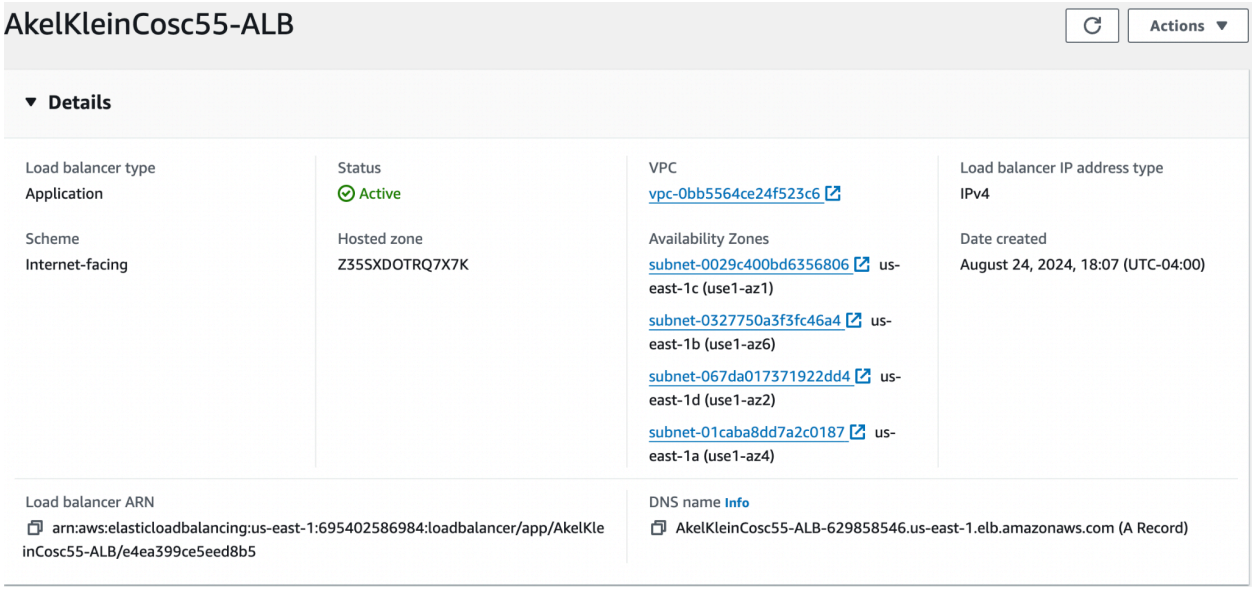


9. Link each Target Group to an EC2 instance

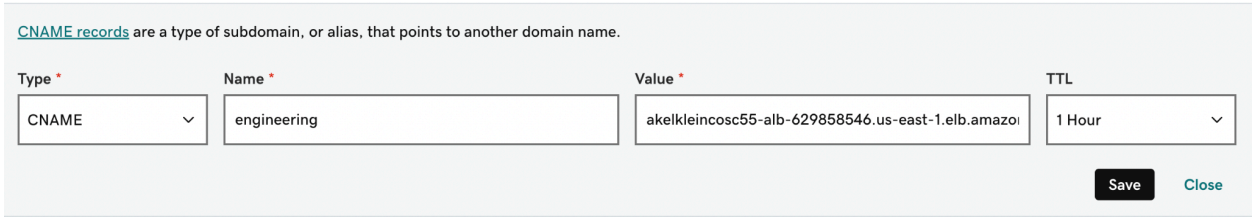


10. Create Application Load Balancer (Purpose: to distribute incoming HTTP and HTTPS traffic across multiple targets, providing high availability, scalability, and advanced routing capabilities for web applications deployed on aws) (Source: easydeploy.io) → Ensure that ALB is using the correct VPC and availability zones of all subnets are selected →

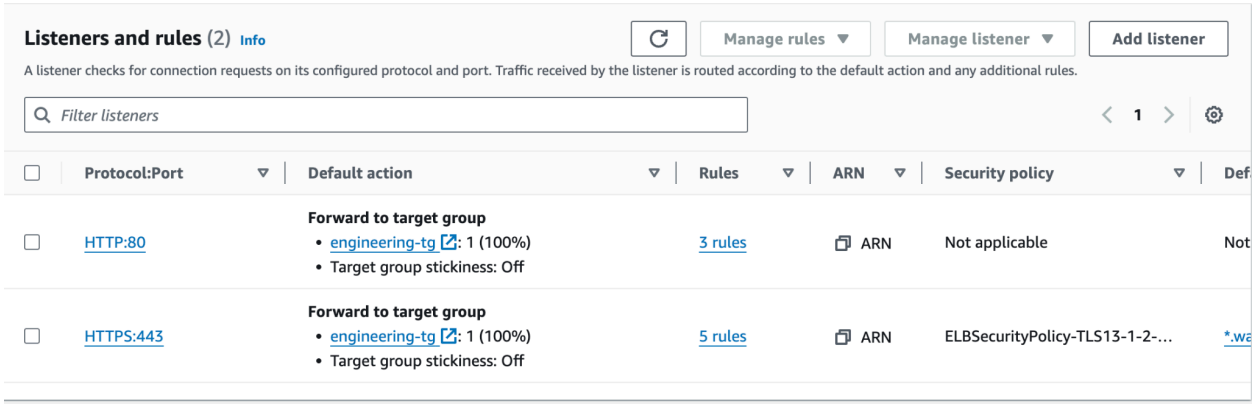
see below



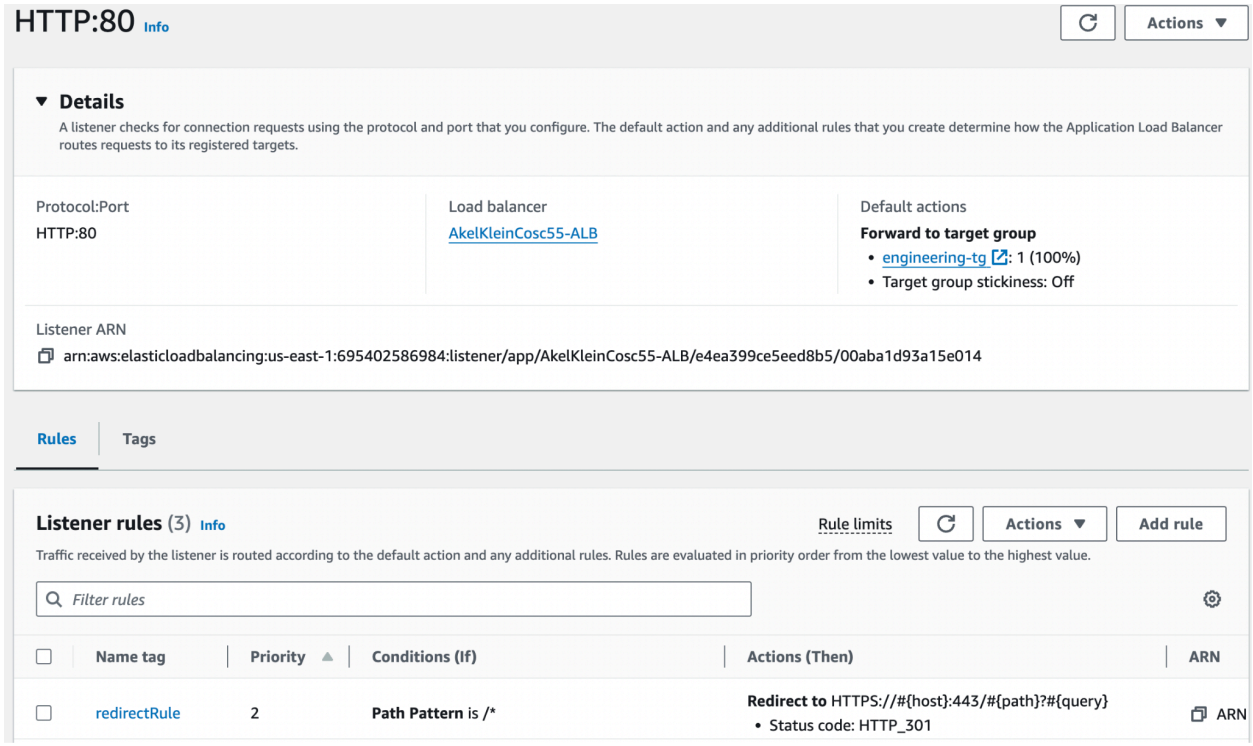
11. Using GoDaddy.com create subdomains such that subdomains of root can be redirected to EC2 instances through the ALB (CNAME: name = desired subdomain, value = ALB)
→ see example below of engineering subdomain creation
(source: https://montanawong.medium.com/how-to-point-your-custom-domain-to-an-aws-load-balancer-51dc2eb6d84c)



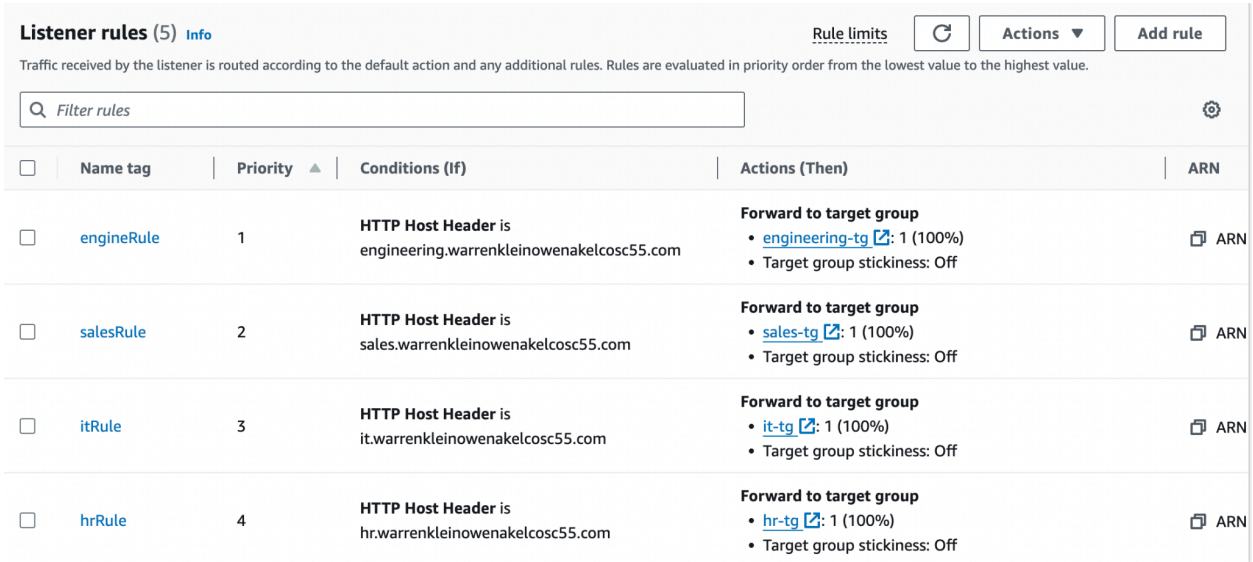
12. Go back to the ALB and create two listeners, 1 for port 80 (HTTP) and 1 for port 443 (HTTPS)



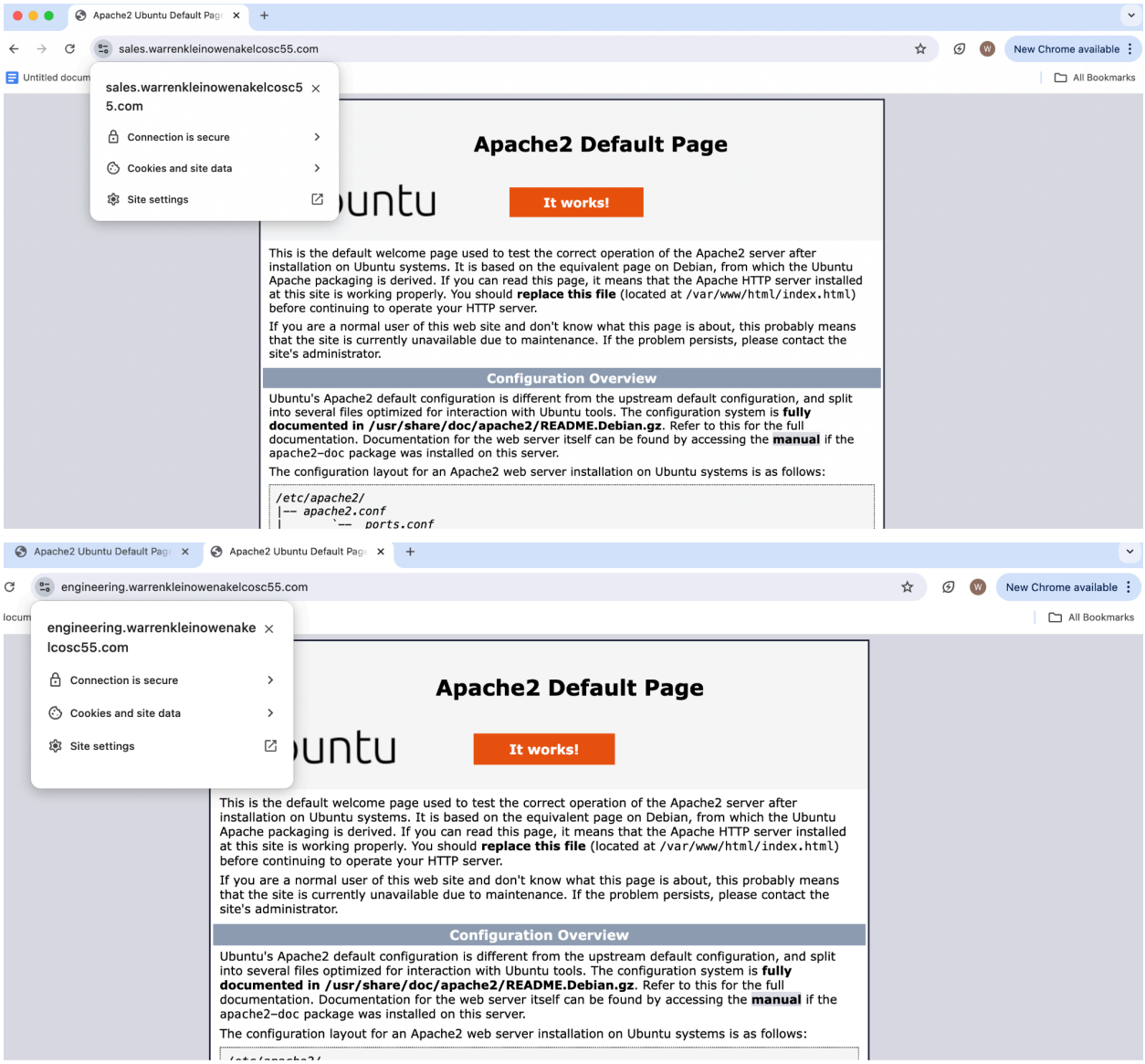
13. Rules for Port 80: redirect all traffic to HTTPS → allow for secure connection (for condition select **Path:** enter /* to allow for all, and the action select Redirect to HTTPS, make this rule the highest priority)

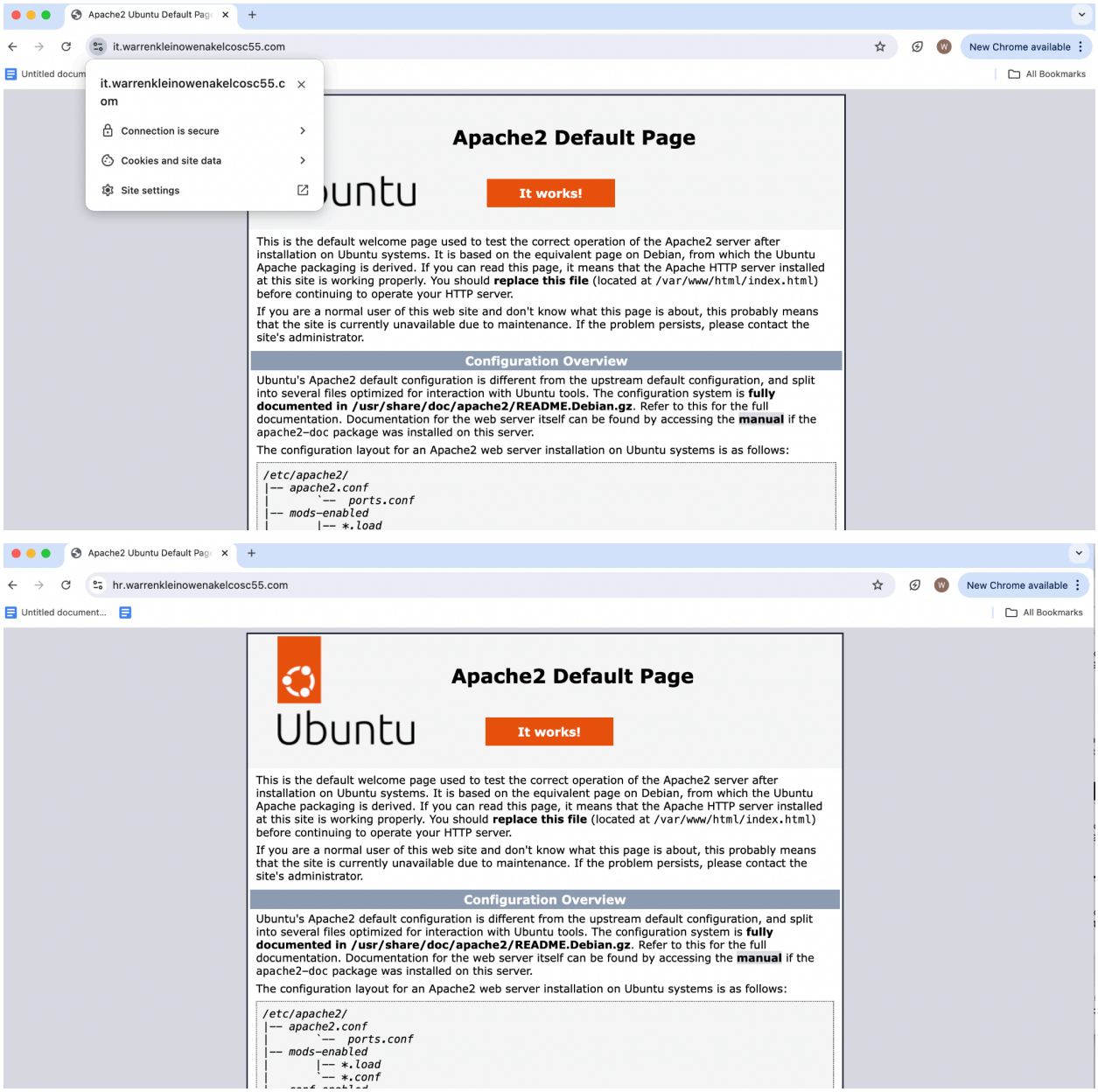


14. Rules for Port 443, for each of the target groups, set conditions for hostname to redirect to the various subdomains: ex sales.warrenkleinowenakelcosc.55 → aligns with subdomain created on GoDaddy, connect the rule to the target group created above → as a result the traffic from HTTP will be redirected to HTTPS as per the rule above which will in turn create a secure connection for each of the subdomains



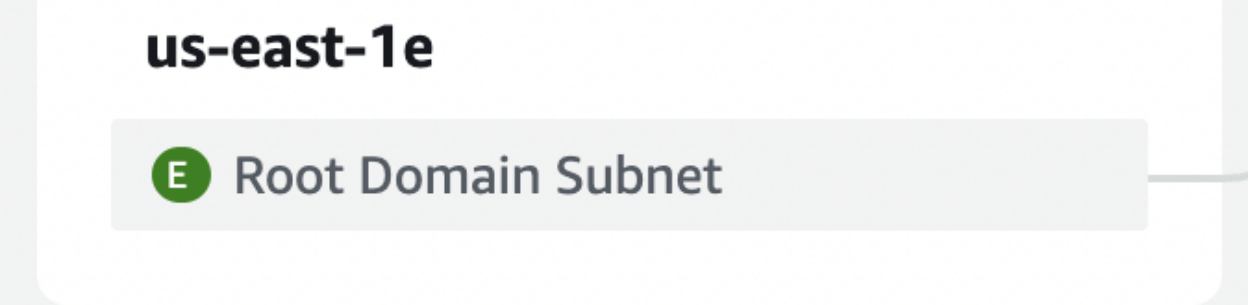
15. Look up website and verify that connection is secure



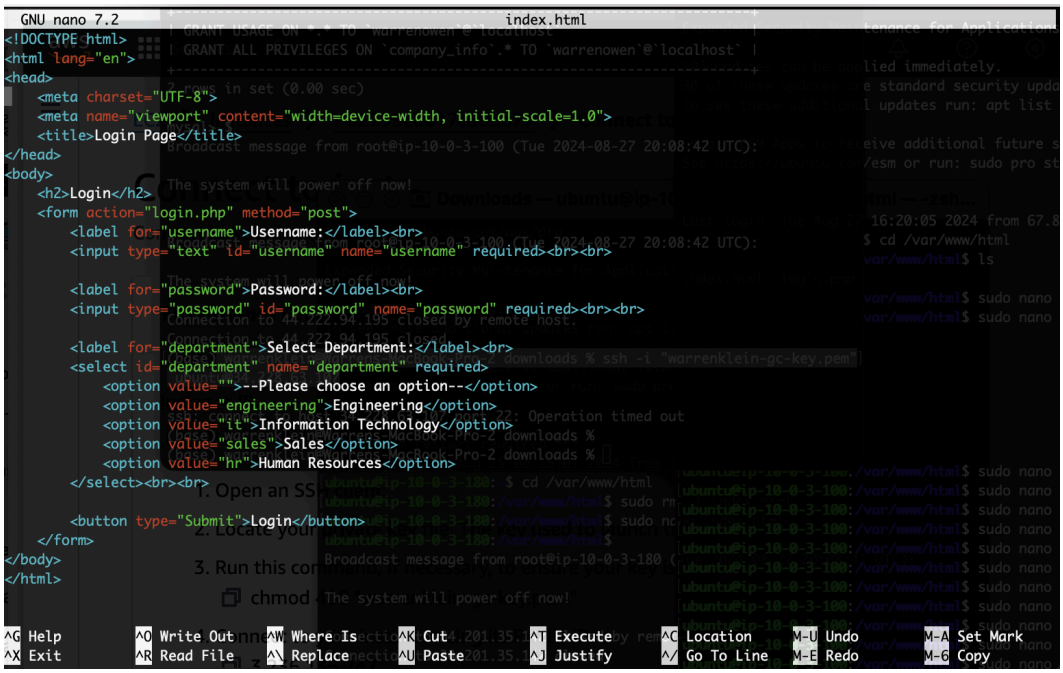


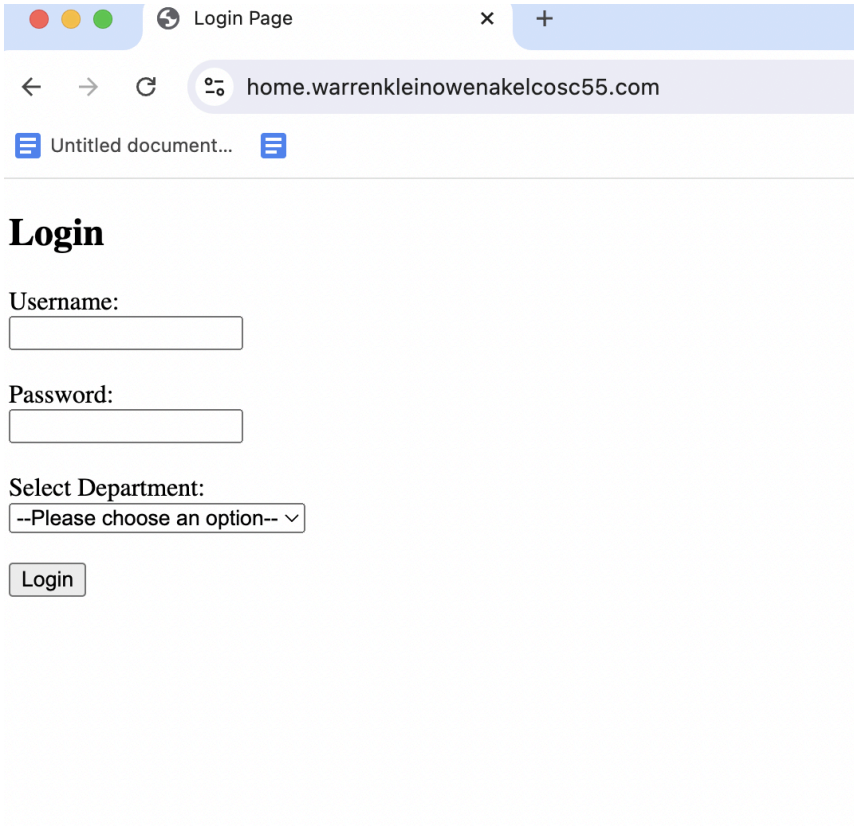
16. Create a new subnet labeled root / home to be used as the homepage to redirect traffic based off login credentials → will forward to secured pages above (IT, HR, Engineering,

Sales)



17. Follow same steps as before with connecting to GoDaddy (apologies for misnamed subnet), EC2 instance, target groups, routing table, target group, and link to ALB with listener rules (makes secure)
18. Develop HTML script to be used as login screen for home domain (see login credentials and desired department)



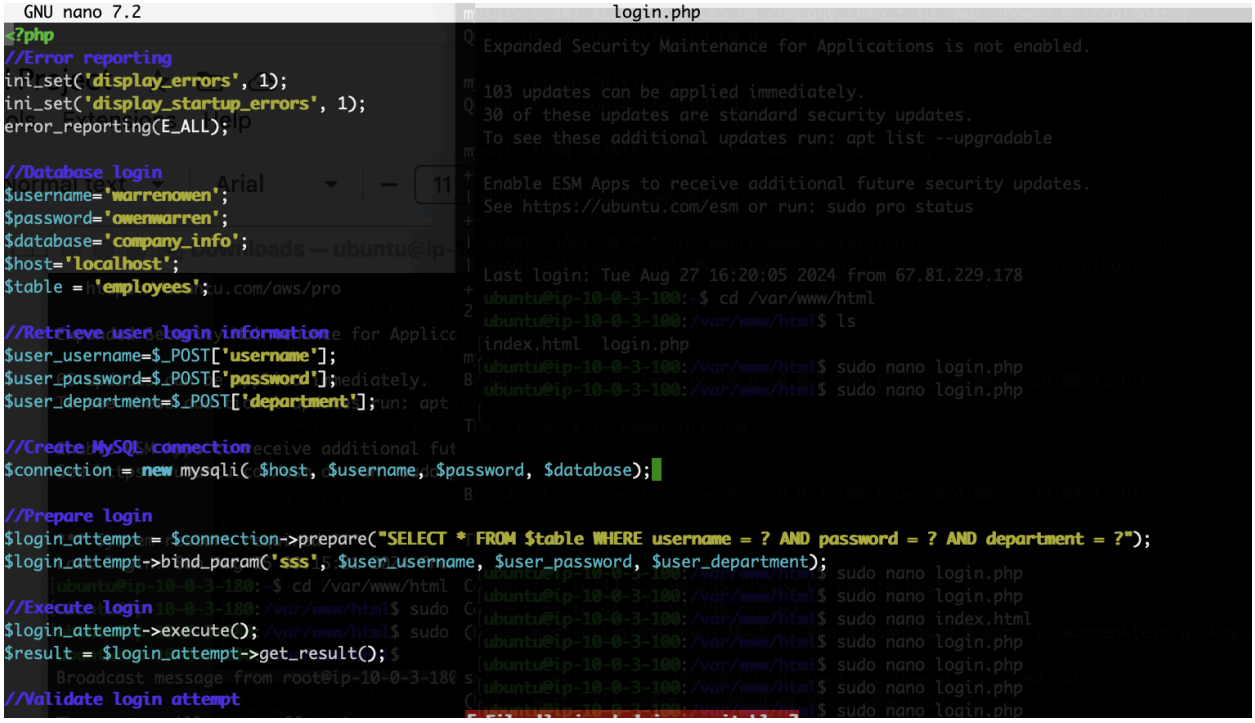


19. Connect PHP to MySQL to be used to authenticate login credentials and grant all privileges to newly created database

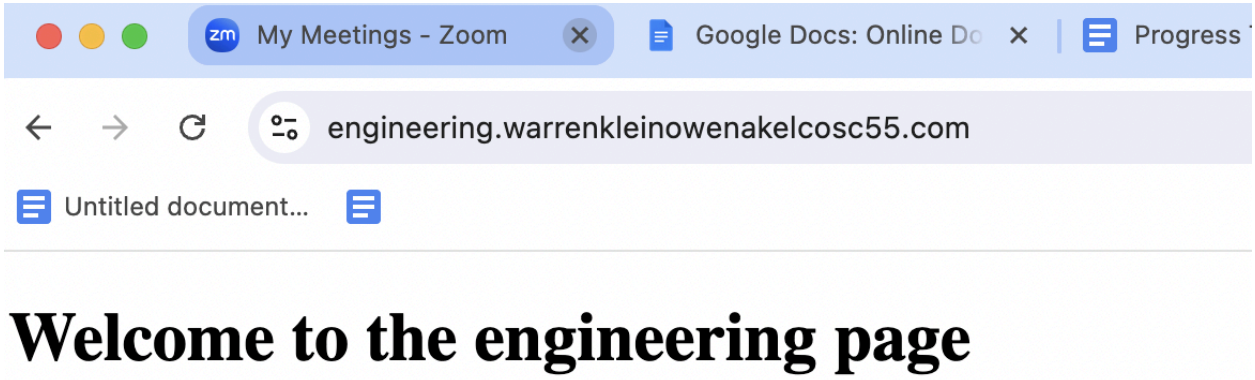
```
mysql> CREATE DATABASE company_info;
Query OK, 1 row affected (0.01 sec)

mysql> USE company_info;
Database changed
mysql> CREATE TABLE employees (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     username VARCHAR(50) NOT NULL,
    ->     password VARCHAR(255) NOT NULL,
    ->     department VARCHAR(50) NOT NULL
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO employees (username, password, department) VALUES
    -> ('Warren', 'dog', 'engineering'),
    -> ('Owen', 'cat', 'hr'),
    -> ('Bennett', 'bug', 'it'),
    -> ('William', 'monkey', 'sales');
Query OK, 4 rows affected (0.02 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM employees;
+----+----------+----------+-------------+
| id | username | password | department  |
+----+----------+----------+-------------+
|  1 | Warren   | dog      | engineering |
|  2 | Owen     | cat      | hr          |
|  3 | Bennett  | bug      | it          |
|  4 | William  | monkey   | sales       |
+----+----------+----------+-------------+
```

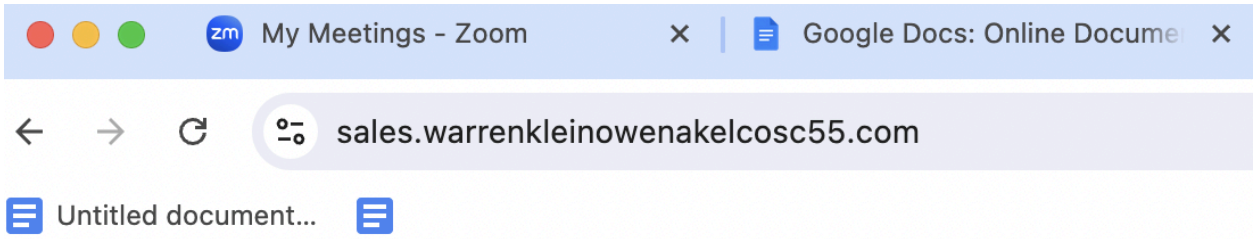
```
mysql> GRANT ALL PRIVILEGES ON company_info.* TO 'warrenowen'@'localhost' IDENTIFIED BY 'owenwarren';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to
use near 'IDENTIFIED BY 'owenwarren'' at line 1
mysql> GRANT ALL PRIVILEGES ON company_info.* TO 'warrenowen'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> SHOW GRANTS FOR 'warrenowen'@'localhost';
+-----------------------------------------------------------------------+
| Grants for warrenowen@localhost                                       |
+-----------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `warrenowen`@`localhost`                        |
| GRANT ALL PRIVILEGES ON `company_info`.* TO `warrenowen`@`localhost` |
+-----------------------------------------------------------------------+
2 rows in set (0.00 sec)
```

20. Develop PHP script that accesses database, authenticates credentials and redirects to web pages

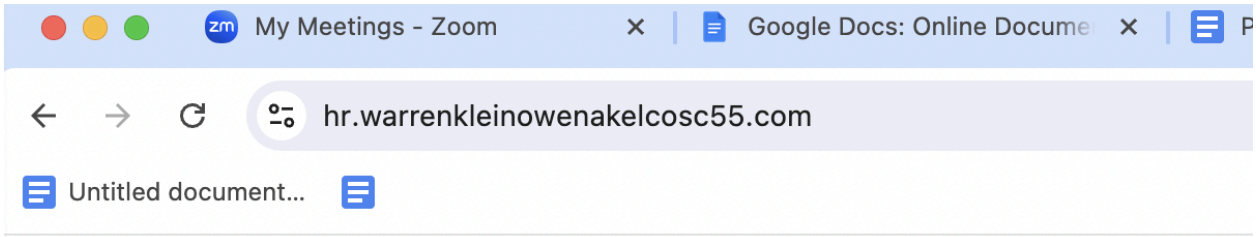


21. Create sample webpages for each of the different departments that the login page redirects to

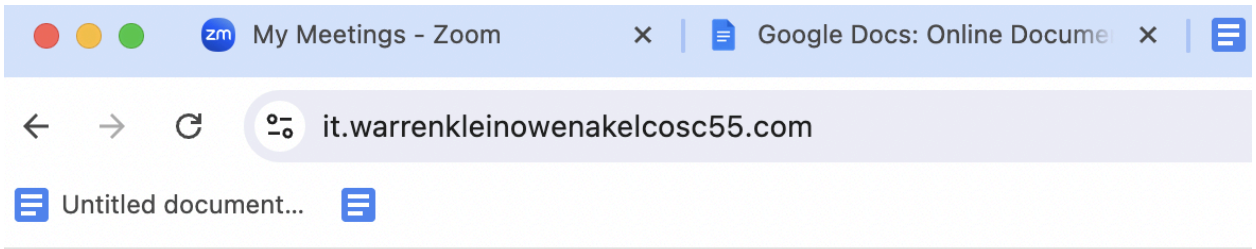




Welcome to the sales page



Welcome to the HR page



Welcome to the IT page

22. Video Demo: https://drive.google.com/file/d/1rqTRogNbPhHJdHp7OnDiwT76MguqLSmK/view?usp=drive_link

Setbacks and Challenges with initial setup:

- **Domain Registration and DNS Configuration:**
  - **Initial Setup Issues:** Faced difficulties with registering a domain through AWS Route 53, leading to the decision to use GoDaddy for domain registration.
  - **DNS Propagation Delay:** Experienced delays in DNS propagation when configuring subdomains with GoDaddy, which affected the accessibility of your subdomains.
  - **CNAME vs. A Record Confusion:** Encountered uncertainty about whether to use CNAME records or A records, particularly for the root domain and its integration with the AWS ALB.
- **ALB Listener and Rule Configuration:**
  - **Host Header Rule Configuration:** Initially struggled with configuring host header rules to route traffic from subdomains to the correct target groups through the ALB.
  - **Redirection Rule Setup:** Faced challenges in setting up HTTP to HTTPS redirection, including determining the appropriate condition and ensuring the redirection rule was prioritized correctly.
- **Health Check Failures:**
  - **Unhealthy Target Groups:** Encountered issues with target groups showing as unhealthy, likely due to incorrect health check paths, protocols, or ports, which needed troubleshooting and adjustment.
- **HTTPS and SSL/TLS Setup:**
  - **Wildcard SSL/TLS Certificate Application:** Successfully applied a wildcard SSL/TLS certificate but faced difficulties in ensuring that all HTTP traffic was correctly redirected to HTTPS, securing all subdomains.

Reasons for using ALB and horizontal integration of the departments:

Isolation of Services:

- Each department (e.g., Engineering, HR, IT) operates independently on its own EC2 instance, reducing the risk of one department's issues affecting others.

Customized Resource Allocation:

- Different departments can have EC2 instances tailored to their specific needs, optimizing performance and cost efficiency.

Simplified Management and Scaling:

- By isolating departments on separate instances, you can scale each department's resources independently based on demand.

Enhanced Security and Compliance:

- Keeping departments on separate instances allows for more granular security policies and easier compliance with data protection regulations.

Simplified Troubleshooting and Maintenance:

- Issues within one department’s application can be isolated and addressed without impacting the entire organization, leading to more efficient maintenance.

Brainstorming/Troubleshooting: