ATLANTIC TECHNOLOGICAL UNIVERSITY DONEGAL

ASSIGNMENT COVER SHEET

To Be Completed By The Student

Lecturer’s Name: Ruth Lennon

Assessment Title: Assignment 1 Submission Date: 20th November 2022

Student’s Name: Chimezirim Ugochukwu Id. Number:\_L00170994\_

Course / Stage Msc DevOps

Subject/Module: IAC for DevOps Pipeline

Word Limit: Actual Word Count: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

I confirm that the work submitted has been produced solely through my own efforts.

Student’s signature: Date:

|  |
| --- |
| **Note**: **PENALTIES**   1. The total marks available for an assessment will be reduced by 15% for work submitted up to one week late. The total marks available are reduced by 30% for work up to two weeks late. 2. Assessment work received more than two weeks late, without prior approval by the lecturer will receive a mark of zero. 3. Marks awarded will be reduced by 10 % if submitted work is greater than 10% above or below the assigned word limit. 4. A further hard or electronic copy of your submitted work may be requested, and therefore you must keep a copy on disc. 5. Incidents of alleged plagiarism and cheating are dealt with in accordance with the Institute’s Assessment Regulations   **Plagiarism:** Presenting the ideas, words of someone else without proper acknowledgement. Refer to the Institutes’ procedures and guidelines for the assessment of learners. |

**GitHub Repository:** https://github.com/chimezirimugochukwu/CF-ASSIGNMENT.git

**CONCLUSION:**

When working on this project, nested stacks were not used because it removes the option of modularity in design when creating the templates.

This modularity gives the advantage of each stack to be able to be grouped according to its functions, which in turn makes the purpose of each templete easy to understand, and easy scalability of the resources.

A core principle in building any good IT infrastructure is applying the principle of least priviledge, in keeping with that a Jumpbox is used to authenticate into the application server.

In the production environment, when faced with the job of creating a Jumpbox there a variety of factors you need to consider;

1. What kind of service needs to be accessed: Ideally this should be mission-crtitcal systems such as Databases, Application Servers, or any sensitive system.
2. Who can access these services: This needs to be carefully thought so as not to give just anyone access administrator access to the Databses or sensitive systems.
3. Ingress source needs to be specified: Rules that govern which specific subnets, or vpn that will be allowed to establish a connection to the Jumbox needs to be stated
4. Authorization: The jumpbox is bes described as a single use machine and it is therefore best to limit who has access to it. It is best to enable logging on the Jumpbox as this tracks the activity and can help in troubleshooting, and audits.

The template design chosen for the project allows for easy implementation of cross-stack referencing using Outputs and the import function. This function is necessary as it enables us to design out templates compactly by just importing an already created resources. This is particulary useful when the output from one stack needs to be referenced/imported into multiple other stacks.

This project made use of an Elastic Ip address. AWS provieds its customers with the option of associating their resources with an Elastic Ip address which is almost free to use. This Elastic Ip ensures your infrastructure are reacheable over the internet and its public Ip address does not change in the event the EC2 instance shuts down or restarts.

This is highly effective in situations of failover, faster resolution of DNS queries. Elastic IPs also provides you the flexibility of not having a long term recurring expense of renewing your Static IP access.

NAT Gateways are used primarily to enable private instance establish connections to the internet, to stop connectoins originating from the internet directly to the Private subnets, and it facilitates the connection between Private subnets in the same VPC.

Nat Gateways are necessary to enable Web applications and Databases located in seperate subnets to communicate with each other, it enables Critical systems hosted in Private subnets get secutrity patches sent over the internet, download necessary trusted packages e.t.c.

It is best practice to create one NAT Gateway in each availability zone your subnets are located as NAT Gateways are highly available only within the Availability Zone.

Critical Resources like databases and other services which do not need communication from the outside world, It is always best practice to keep them behind a private subnet.

In keeping with industry best practice of security, it is a huge security risk to pass in plain text your Passwords or credentials when building out your clouformation templates. It is highly adviceable to use AWS Secrets manager or AWS Parameter store.

AWS Parameter store was used in this project as it provides a cost effective way of Encrypting confidential informations such as keypairs, passwords, and other secrets whic needs to be secured. It provides a versioning feature which allows you to track previous versions of your secrets in case you need to roll back to an earlier version. Parameter store also seamlessly integrates with cloudformation through the use of dynamic references in the cloud formation template.

Deploying your Database instance over multiple availability zones is an effective way of properly configuring your databases, as this helps to mitigate secutiry risks when your primary database comes under threat, it makes failing over to your standby database seamless. This practice also aids in data durability by distributing data backups across multiple availablility zones.

AWS knowing the advanages of multi avalaibility zone deployments strongly encourages its customers to implement it, by enforcing database deployment over at least two subnets in different availability zones, even when deploying databases in a single availability zone system.

This way it makes for a smooth conversion to a multi availaibility zone deployment when the need arises.

In a live environment some necessary measures could be taken ensure the system performs optimally, such as;

1. Creation of Auto scaling groups to optimize the infrasrtucture for peak periods and off peak periods.
2. Creation of Elastic Load Balancers to ensure to optimally split loads between available resources.
3. Replication of mission critical resources across multiple Availability zones to ensure a fault tolerant system and increase resiliency.

While working on this project several best practice regualtion were implemented with regards to Governance, Risk and Compliance.

From a Governance perspective, code readability, proper resource labeling, and appropraite allocation of resource permissions.

To mitigate security risks in this infrastructure appropraite security groups were created and accurately configured, the critical systems were accessed strictly only by a jump box, only necessary ports were exposed.

In an ideal production environment, various steps can be implemented to keep in line with compliance regulations like ISO 27001, GDPR, SOC2, PCI DSS, HIPAA;

1. Cloud trail logging would be implemented
2. S3 buckets for logs storage
3. Using AWS CloudFormation Guard for security static analysis.

**REFERENCES**

Amazon. (NA). *Amazon RDS FAQs*. [Online]. Amazon Web Service. Last Updated: NA. Available at: https://aws.amazon.com/rds/faqs/ [Accessed 17 November 2022].

Amazon. (NA). *Continuous Compliance Workflow for Infrastructure as Code: Part 1*. [Online]. Amazon Web Service. Last Updated: NA. Available at: https://aws.amazon.com/blogs/devops/continuous-compliance-workflow-for-infrastructure-as-code-part-1 [Accessed 17 November 2022].

Amazon. (NA). *AWS resource and property types reference*. [Online]. Amazon Web Service. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-template-resource-type-ref.html [Accessed 9 November 2022].

Amazon. (NA). *AWS::SecretsManager::Secret*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-secretsmanager-secret.ht [Accessed 9 November 2022].

Amazon. (NA). *AWS KMS keys*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#kms\_keysNA [Accessed 9 November 2022].

Amazon. (NA). *Get started with AWS Secrets Manager*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/secretsmanager/latest/userguide/getting-started.html [Accessed 9 November 2022].

Amazon. (NA). *Intrinsic function reference*. [Online]. Amazon Web Service. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-functio [Accessed 9 November 2022].

Amazon. (NA). *Outputs*. [Online]. Amazon Web Services. Last Updated: NA. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/outputs-section-structure.html [Accessed 17 November 2022].

Amazon. (NA). *Security best practices for AWS CloudFormation*. [Online]. Amazon Web Service. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/security-best-practices.html [Accessed 9 November 2022].

Amazon. (NA). *SSM parameter types*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/parameters-section-structure.html#aws [Accessed 9 November 2022].

Amazon. (NA). *SSM secure string parameters*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/dynamic-references.html#dynamic-refer [Accessed 9 November 2022].

Amazon. (NA). *Template anatomy*. [Online]. Amazon Web Service. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-anatomy.html [Accessed 9 November 2022].

Amazon. (NA). *Using dynamic references to specify template value*. [Online]. Amazon Web Services. Available at: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/dynamic-references.html [Accessed 9 November 2022].

Ashish, P. (2021). *AWS — Difference between Secrets Manager and Parameter Store (Systems Manager)*. [Online]. www.medium.com. Last Updated: 02 April, 2021. Available at: https://medium.com/awesome-cloud/aws-difference-between-secrets-manager-and-parameter-store-systems- [Accessed 17 November 2022].

Bitnami. (NA). *Find application credentials*. [Online]. bitnami. Available at: https://docs.bitnami.com/aws/faq/get-started/find-credentials/#administrator-username [Accessed 9 November 2022].

Borislav, H. (2022). *AWS SSH Permission denied (PublicKey) Error [Solved]*. [Online]. bobbyhadz. Last Updated: 25 July 2022. Available at: https://bobbyhadz.com/blog/aws-ssh-permission-denied-publickey [Accessed 9 November 2022].

Bridgecrew. (2022). *CloudFormation security 101: Four basic best practices for secure infrastructure as code*. [Online]. bridgecrew. Last Updated: 13 April 2022. Available at: https://bridgecrew.io/blog/basic-best-practices-guidelines-cloudformation-security/ [Accessed 9 November 2022].

Cloud, G. (2018). *AWS VPC - What is the difference between Internet Gateway NAT*. [Online]. www.edureka.co/. Last Updated: 24 April, 2018. Available at: https://www.edureka.co/community/2149/aws-vpc-what-is-the-difference-between-internet-gateway-nat [Accessed 17 November 2022].

Concordia University. (NA). *What is an ssh tunnel and how does it work?*. [Online]. concordia. Available at: https://www.concordia.ca/ginacody/aits/support/faq/ssh-tunnel.html [Accessed 9 November 2022].

Datapath. (2016). *Elastic IP, Static IP, Public IP. What’s the Difference?*. [Online]. www.medium.com. Last Updated: 23 December, 2016. Available at: https://medium.com/@datapath\_io/elastic-ip-static-ip-public-ip-whats-the-difference-8e36ac92b8e7 [Accessed 17 November 2022].

INAP. (2017). *AWS Availability How-To: Achieving Fault Tolerance and Redundancy in EC2*. [Online]. www.inap.com. Last Updated: 27 February, 2017. Available at: https://www.inap.com/blog/aws-fault-tolerance-redundancy-ec2/ [Accessed 17 November 2022].

Kevin, O. (NA). *Working with AWS CloudFormation - Part 2*. [Online]. www.dev.to. Last Updated: 13 January. Available at: https://dev.to/kevin\_odongo35/working-with-aws-cloudformation-part-2-2ak7 [Accessed 17 November 2022].

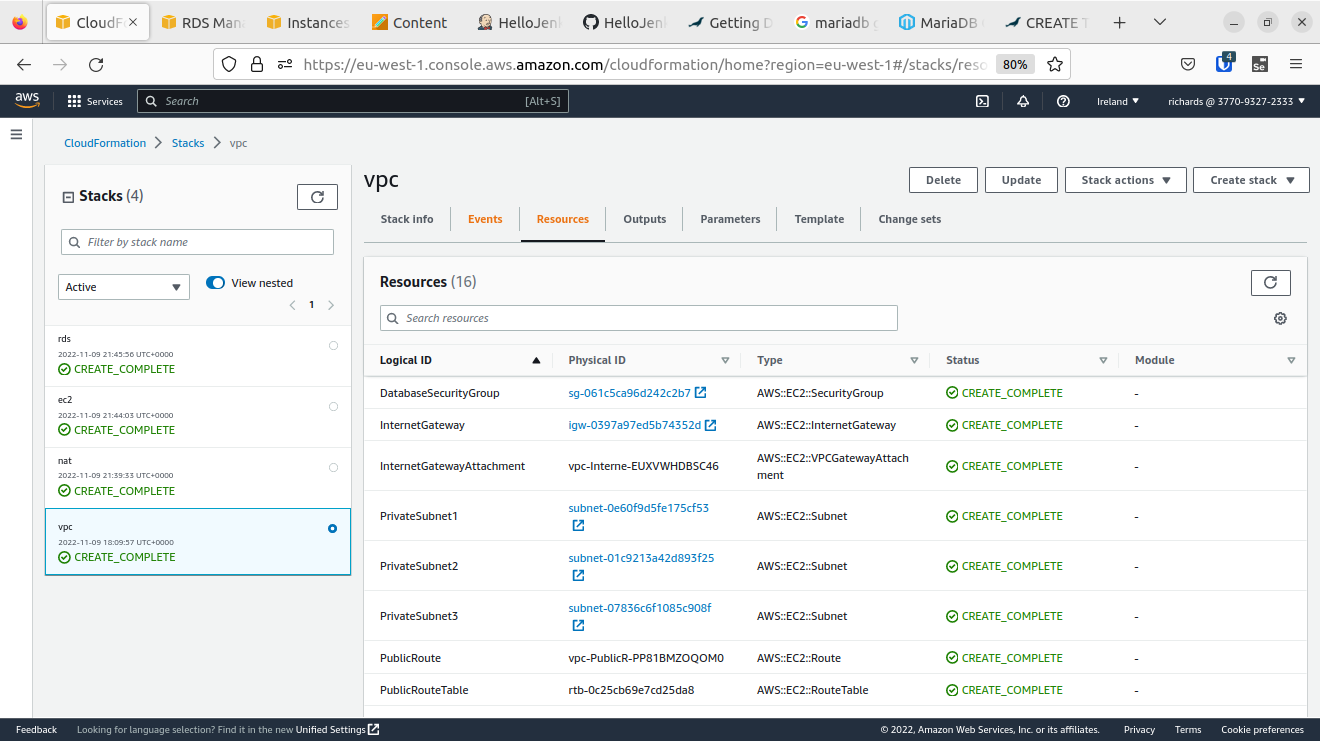
KTree. (2020). *AWS Internet Gateway VS AWS NAT Gateway Use cases*. [Online]. www.medium.com. Last Updated: 27 March, 2020. Available at: https://medium.com/@KTree\_Blog/aws-internet-gateway-vs-aws-nat-gateway-use-cases-d6e976f61782 [Accessed 17 November 2022].

Lervin, J. (NA). *AWS CloudFormation StackSets and Nested Stacks*. [Online]. www.tutorialsdojo.com. Last Updated: NA. Available at: https://tutorialsdojo.com/aws-cloudformation-stacksets-and-nested-stacks/ [Accessed 17 November 2022].

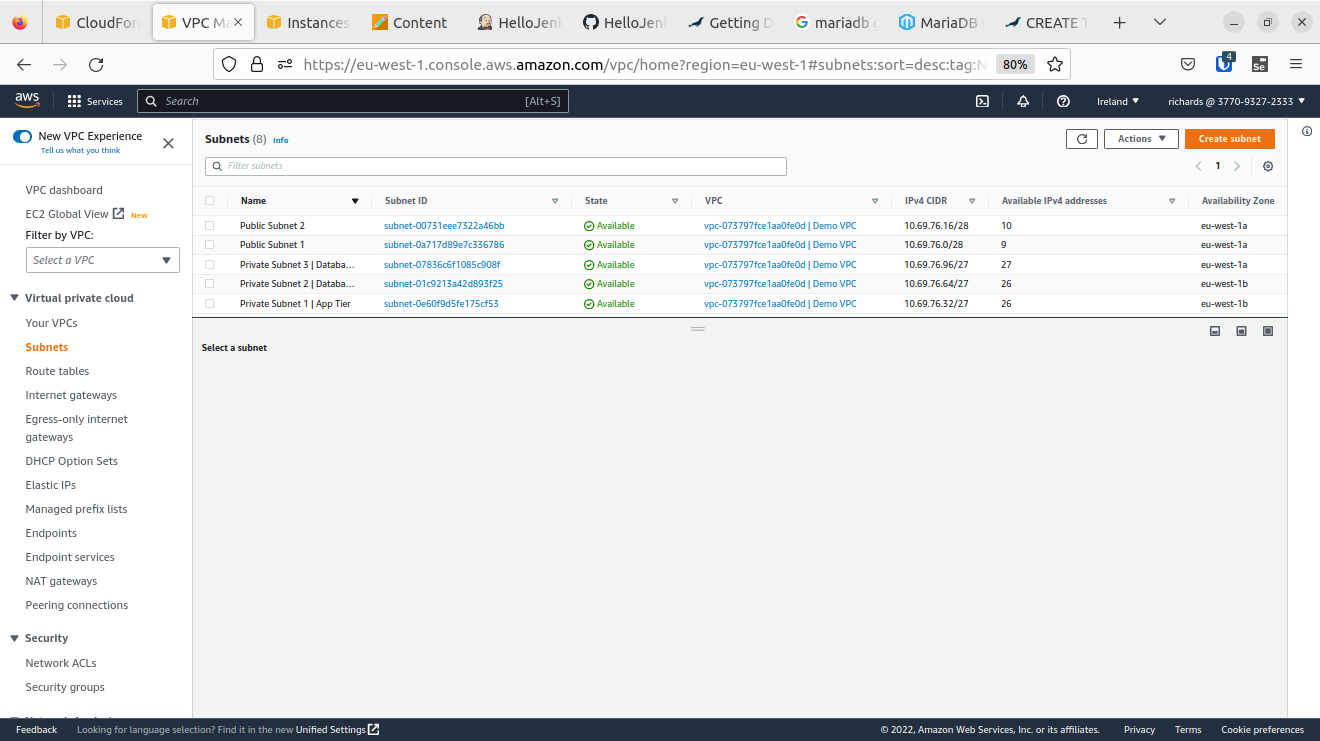
Manmohan, S B. (2021). *AWS VPC and Subnet CIDR calculation and allocation*. [Online]. medium.com. Last Updated: 2 July, 2021. Available at: https://medium.com/geekculture/aws-vpc-and-subnet-cidr-calculation-and-allocation-cfbe69050712 [Accessed 9 November 2022].

Neihaus, A. (2017). *AWS YAML sample CloudFormation template to create VPC*. [Online]. yobyot. Last Updated: 2 May, 2017. Available at: https://www.yobyot.com/aws/aws-cloudformation-example-with-jump-server/2017/05/02/ [Accessed 9 November 2022].

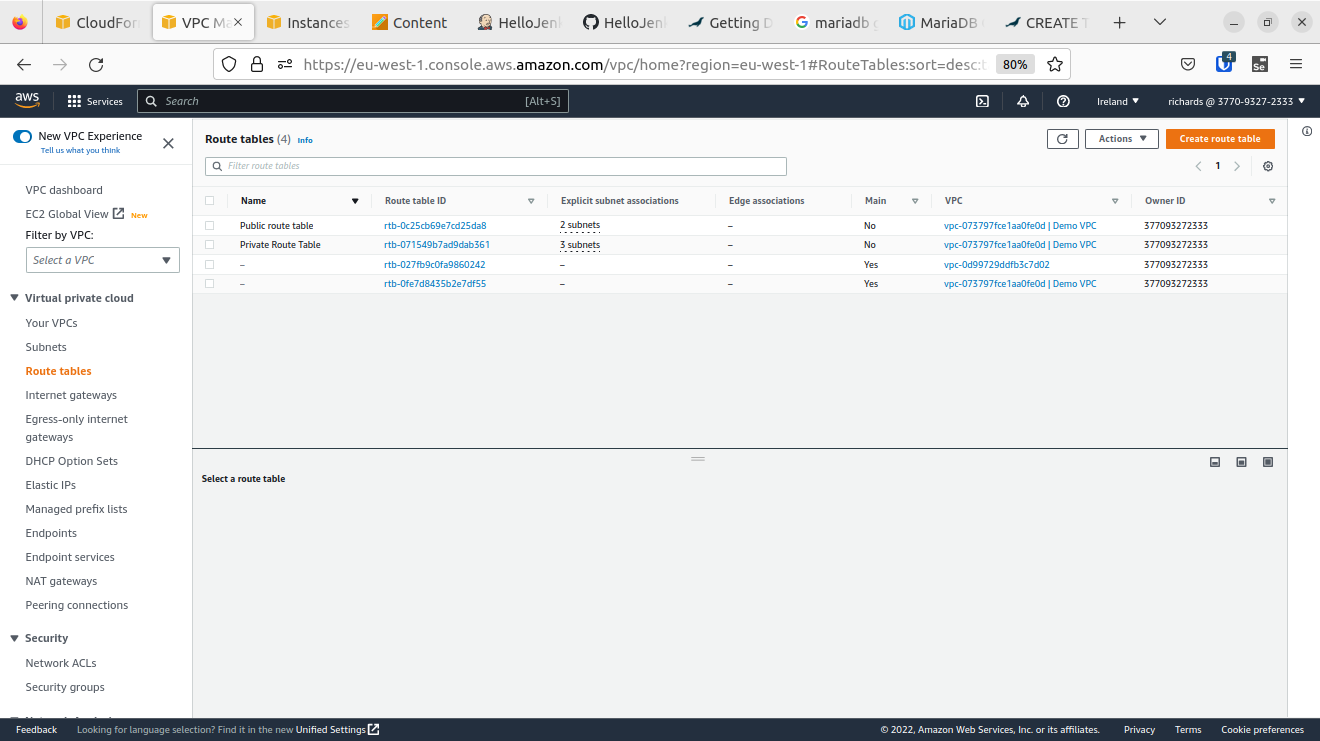
Raymond, P & Sander, V. (2021). *Protecting Critical Systems with Isolation and Jump Boxes*. [Online]. www.f5.com. Last Updated: 21 September, 2021. Available at: https://www.f5.com/labs/articles/cisotociso/protecting-critical-systems-with-isolation-and-jump-box [Accessed 17 November 2022].



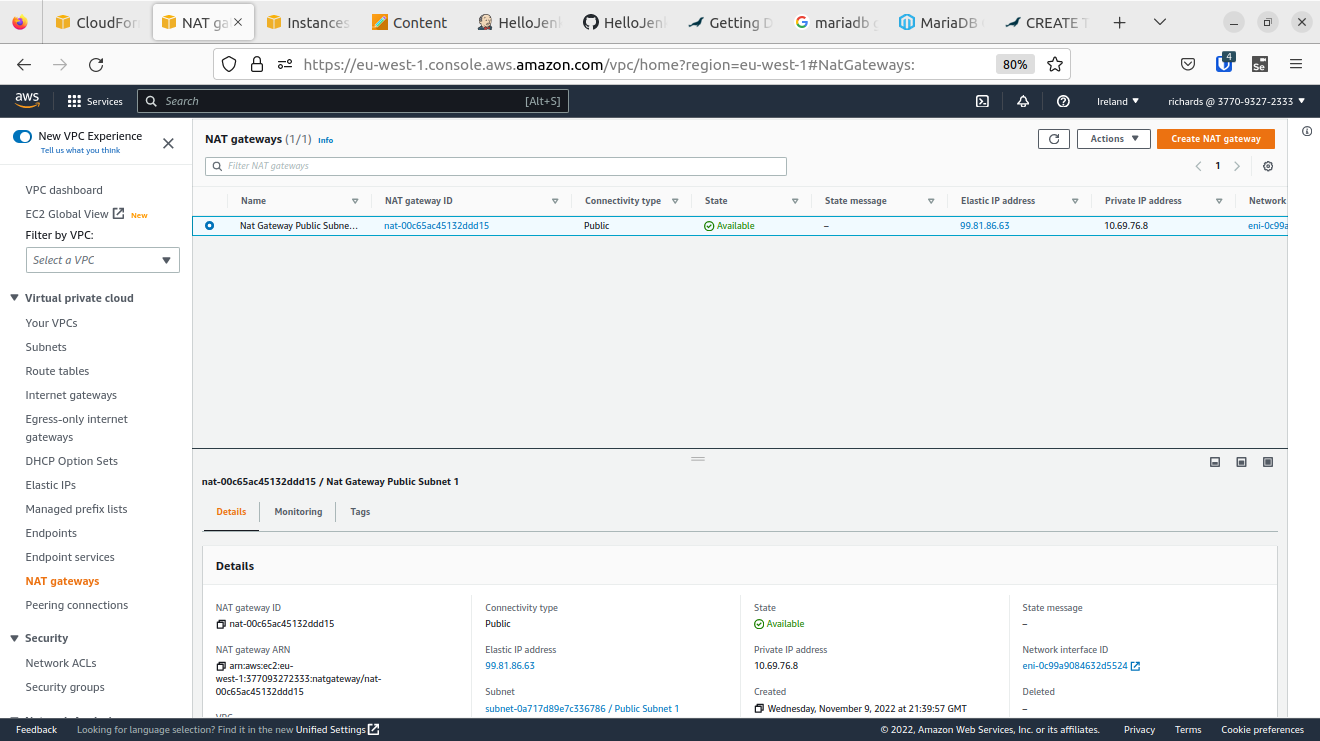
**APPENDICE 1**



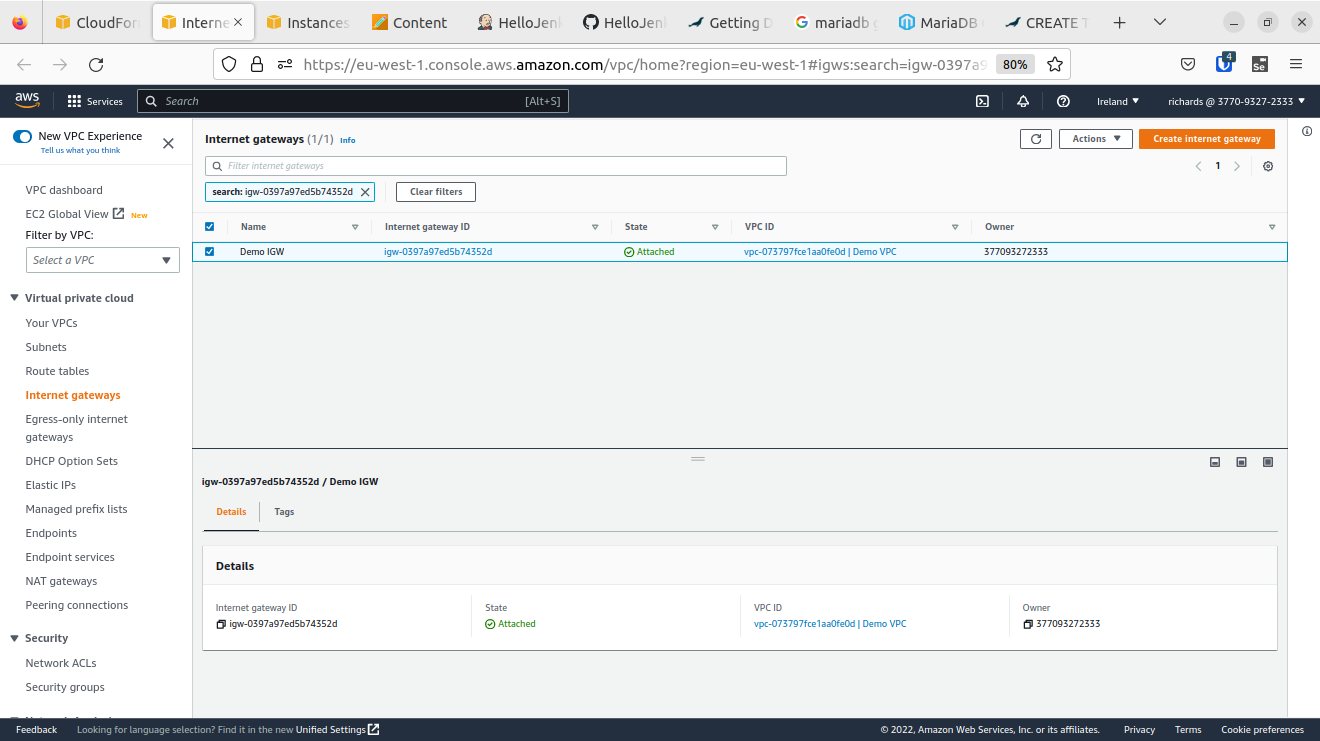
**APPENDICE 2**



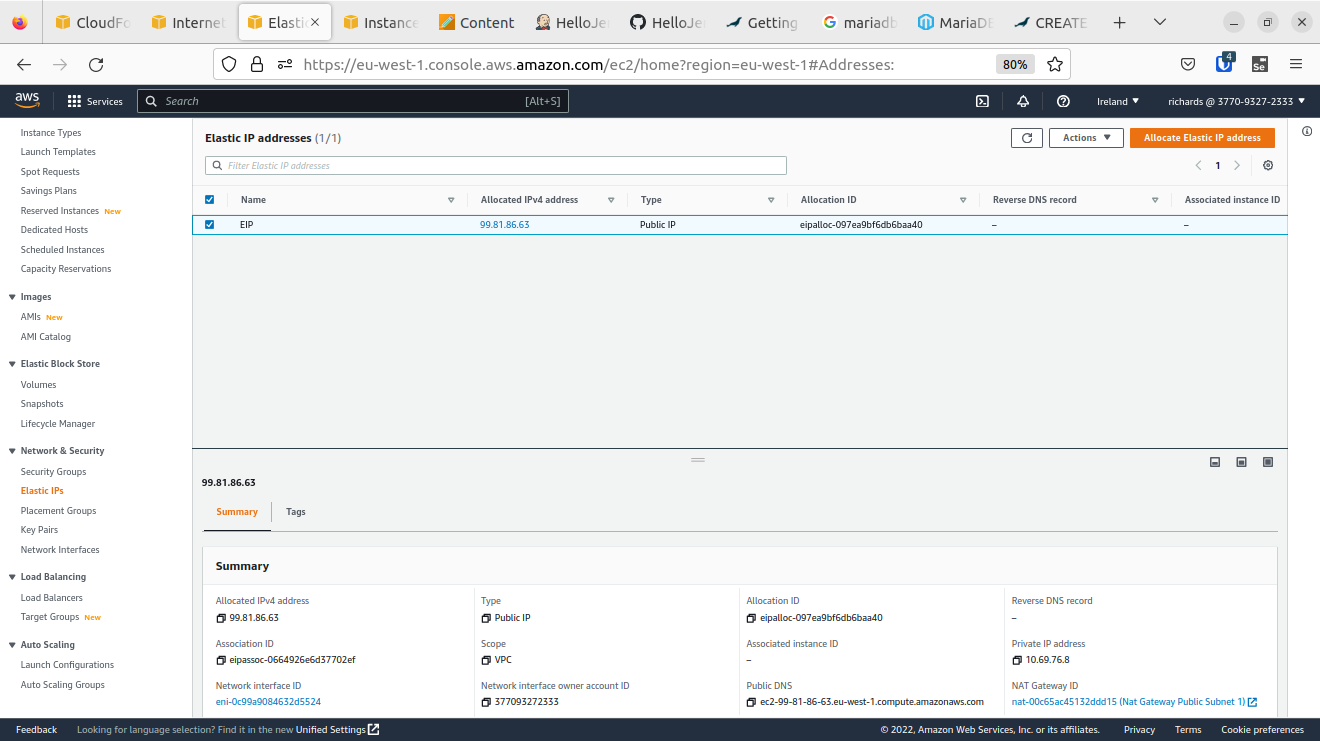
**APPENDICE 3**



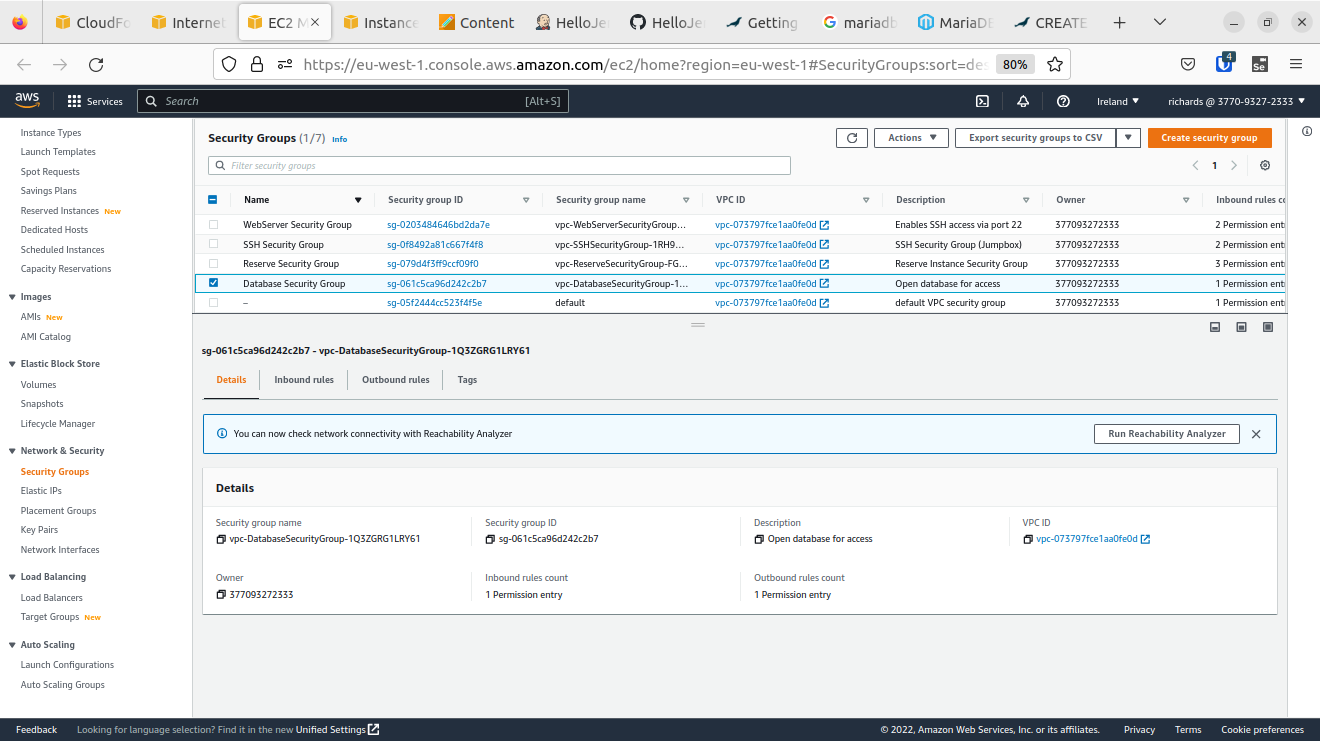
**APPENDICE 4**



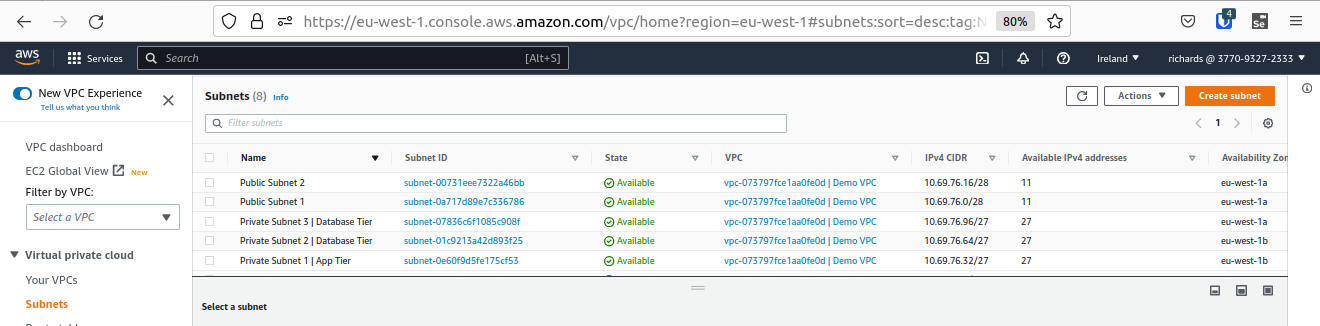
**APPENDICE 5**



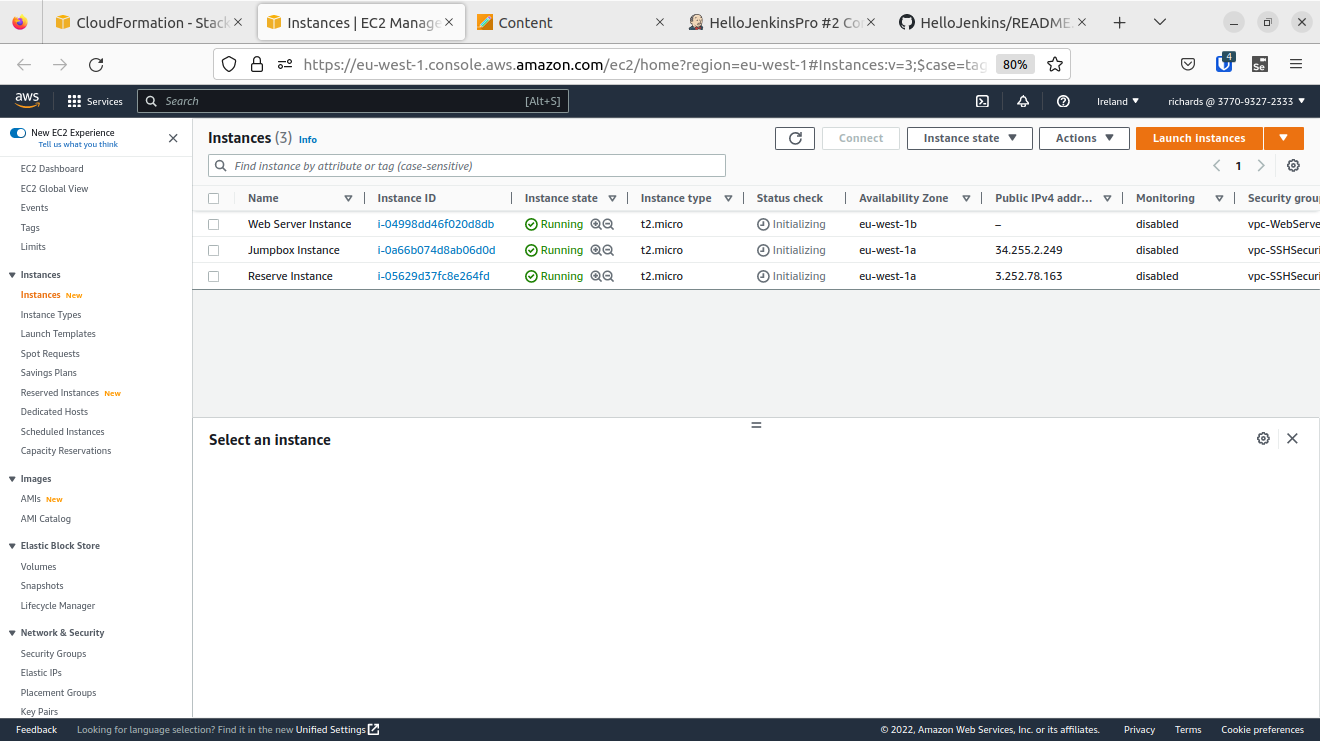
**APPENDICE 6**



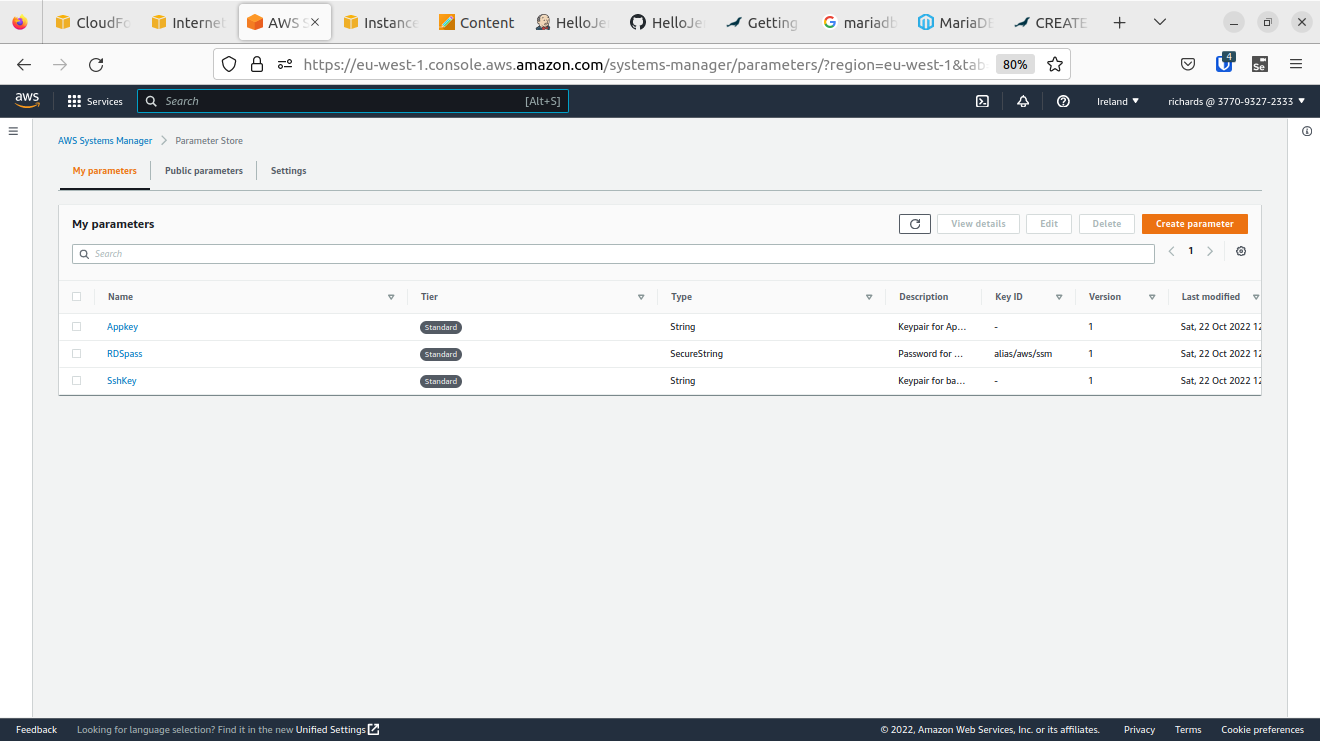
**APPENDICE 7**



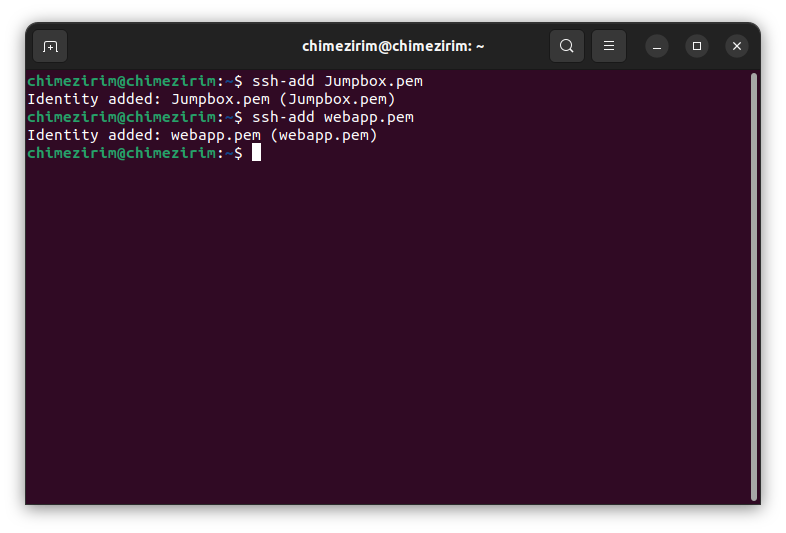
**APPENDICE 8**



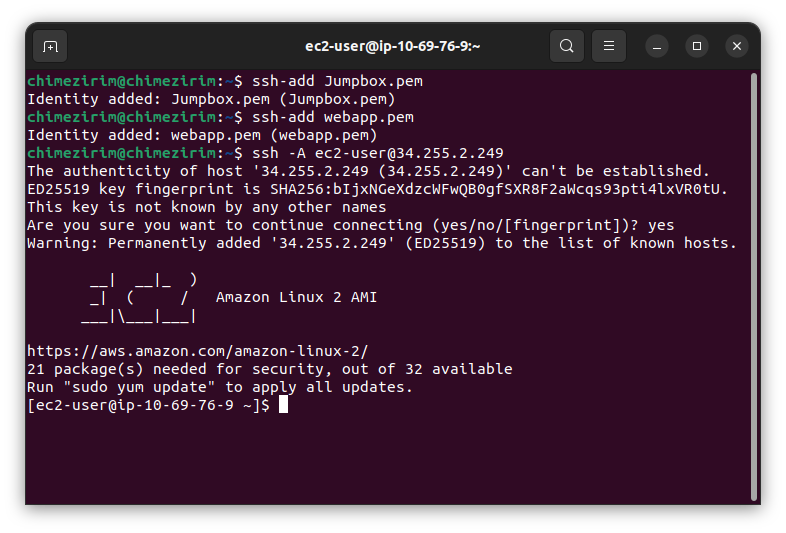
**APPENDICE 9**



**APPENDICE 10**



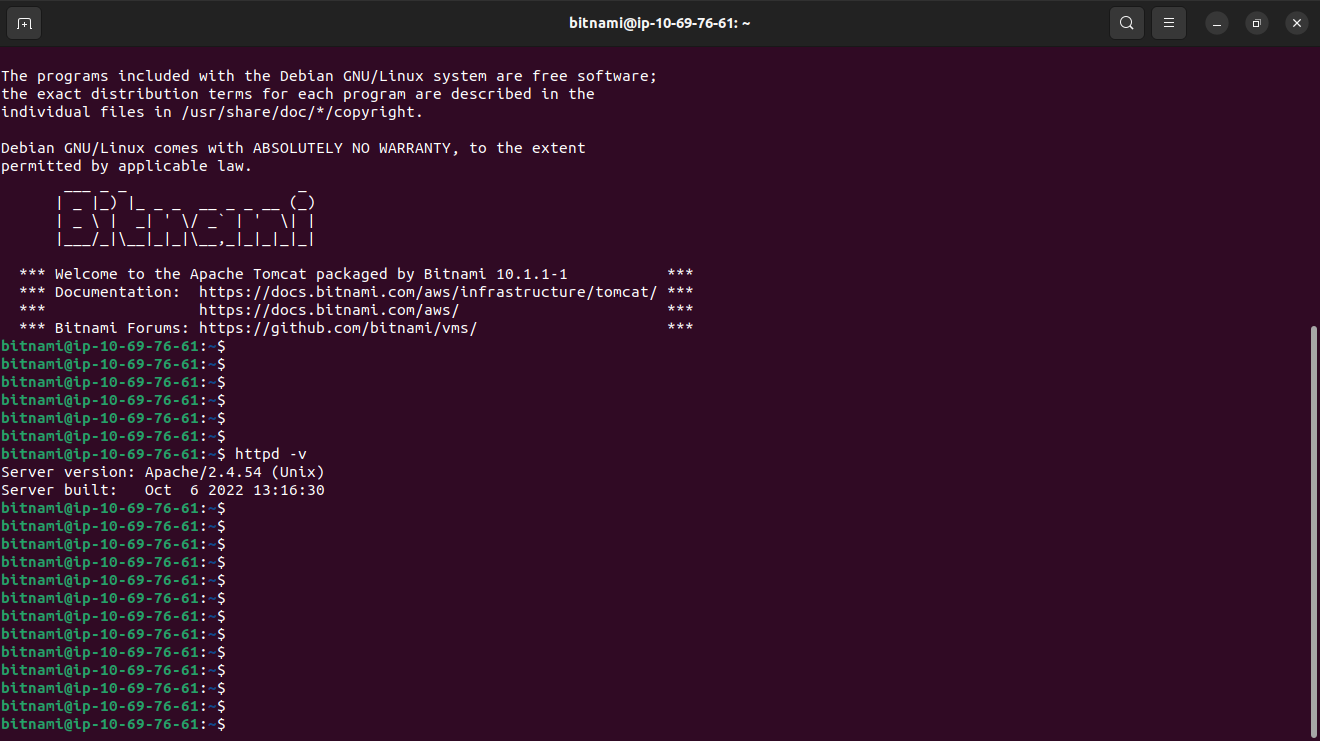
**APPENDICE 11**



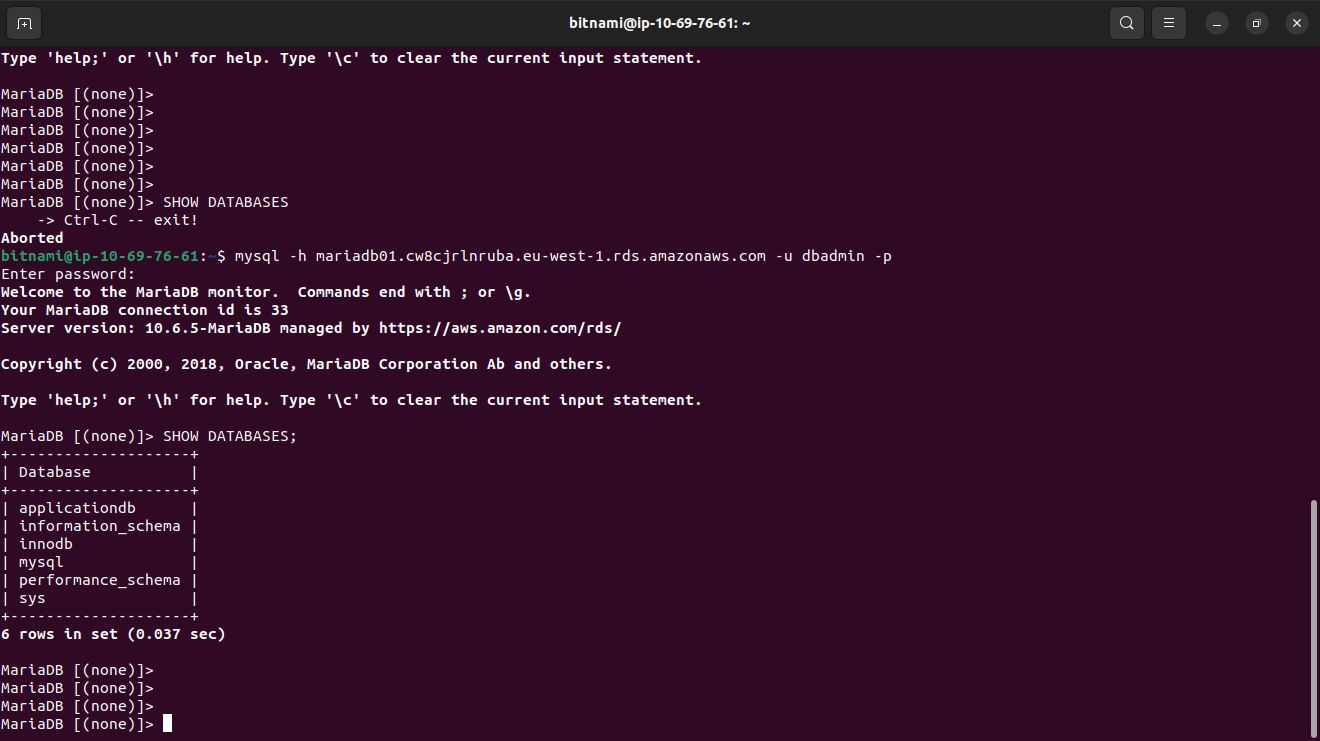
**APPENDICE 12**



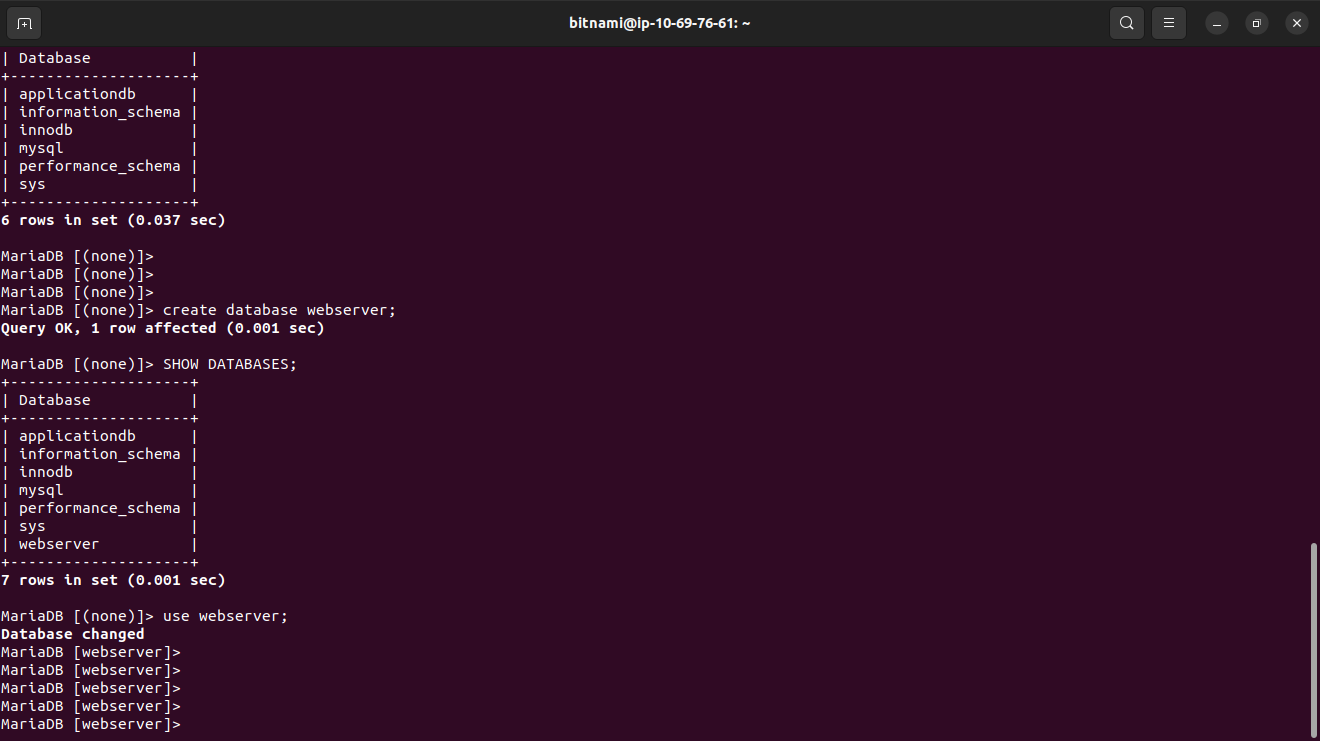
**APPENDICE 13**



**APPENDICE 14**



**APPENDICE 15**



**APPENDICE 16**