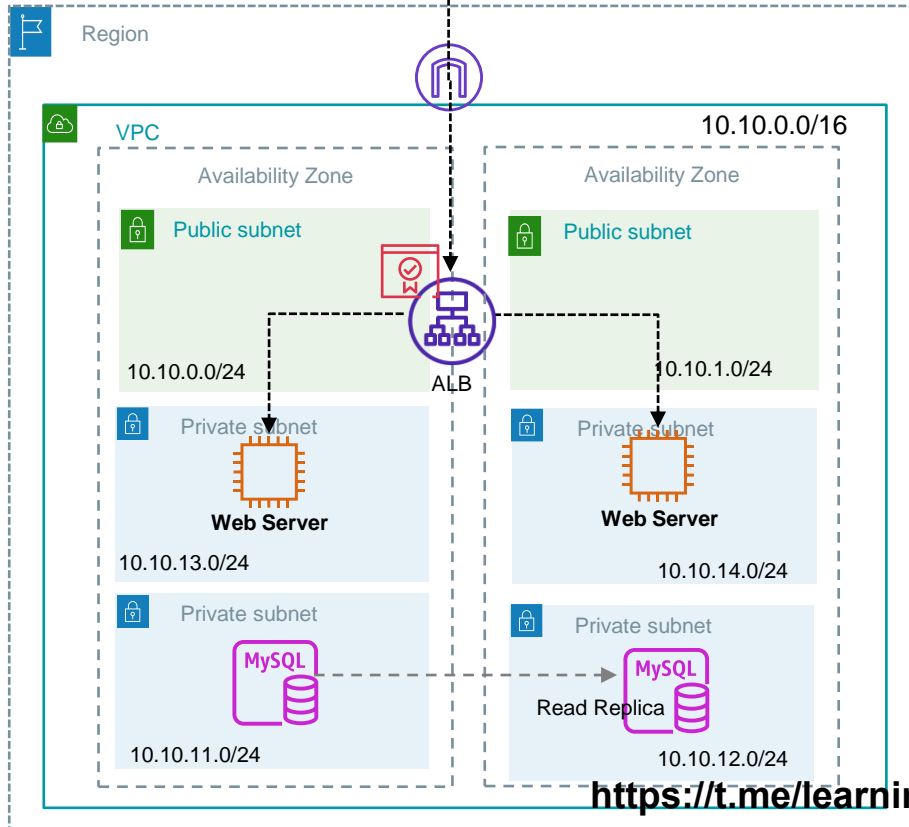
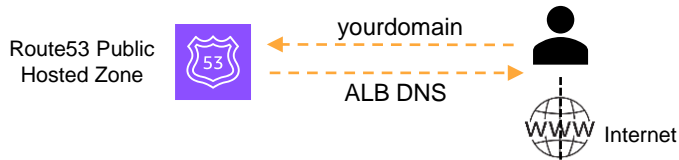


Assignment (Part 2):

Make your website highly available and secure

Using Application Load Balancer with HTTPs listener

<https://t.me/learningnets>



<https://t.me/learningnets>

High level steps

- 1 Stop the webserver and create an AMI. You need AMI because we will now launch webserver in Private subnets.
- 2 Create 2 additional Private subnets for Webserver.
- 3 Launch 2 webserver in respective Private subnets and create ALB across 2 Public subnets as shown. Configure ALB & Target group to sent HTTP traffic to Webserver.
- 4 Modify Database setting and make it Multi-AZ.
- 5 Modify Route53 DNS to point to ALB DNS. Access website using your <http://domainname/corp.php>
- 6 Update ALB listener to HTTPS. Create a new TLS certificate for your domain in ACM and associate with HTTPS listener. See if you need to change anything else for HTTPS.
- 7 Access website using your <https://domain name/corp.php>.

Detailed steps not provided – Try it on your own!

- Hope you can implement part 2 of the exercise on your own.
- Refer troubleshooting steps on next page if you are stuck.

Test your Multi-AZ setup

- Stop one of your webserver and verify that there is no impact on the web application
- For testing database failover – Reboot database instance with failover option and verify that DB points to secondary database.
- Refer this article for how to reboot database:

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_RebootInstance.html

If you could deploy https multi-az website with custom domain,

Congratulations, Really well done !!

Do not forget to perform clean-up for AWS resources that you created.

<https://t.me/learningnets>

Troubleshooting

- If you face problem creating EC2 AMIs or HTTPS configuration, refer to ALB section exercises
- For changing single AZ database to multi-AZ database refer:
<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.MultiAZSingleStandby.html#Concepts.MultiAZ.Migrating.Convert>
- If your setup does not work, possibly web server is not running or web server is not able to communicate with the database. In that case:
 - a. Launch a temporary Jump host EC2 instance in Public subnet and connect to it over SSH. From there, connect to any of the webserver via this jump host over SSH. Check if webserver is running on port 80. Use commands like:
`systemctl status httpd.service`.
 - b. Check public and private subnet routes
 - c. Check security groups for Web server, Database and ALB.

If issue persists, write down in the lecture comment or Q&A or send email to awswithchetan@gmail.com

<https://t.me/learningnets>

Clean-up

It's very important that you delete all your AWS resources after you are done with your exercise to avoid any AWS usage cost.

1. Delete the database:
 - a. Go to RDS console -> Select your database -> Actions -> Delete
 - b. Uncheck Create final snapshot and check the "I Acknowledge .."
 - c. Delete
2. Terminate EC2 instance:
 - a. Go to EC2 console -> Select your EC2 instance -> Instance State -> Terminate
3. Delete ALB Target group and ALB:
 - a. Go to EC2 console -> Application Load Balancer -> Target group -> Delete
 - b. Go to EC2 console -> Application Load Balancer -> Select your ALB -> Delete
4. Delete VPC
 - a. Wait for database and EC2 to be deleted completely
 - b. Go to VPC console -> Select your VPC -> Actions -> Delete VPC
5. Delete Route 53 Public hosted zone (if you are not using it for any other purpose)