Deploy a High-Availability Web App using CloudFormation

Meets Specifications

Dear student,
I enjoyed reviewing your project. You did a great job in designing an infrastructure to host a web application with high availability.
Keep up the good job!
Congratulations!

The Basics

The more the better, but an exaggerated number of parameters can be messy (say, 10 or more). 1 or 0 is definitely lacking.

great that you have to parameter json files, you made the code more decoupled and reusable, see the documentation https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/parameters-section-structure.html

This is the mandatory section of the script, we are looking for a LoadBalancer, Launch Configuration, AutoScaling group a health check, security groups and a Listener and Target Group.

This is optional, but it would be nice to have a URL here with the Load Balancer DNS Name and “http” in front of it .

You did a great job here. see the documentation https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-elasticloadbalancingv2-loadbalancer.html

If the student provides a URL to verify his work is running properly, it will be a page that says “it works! Udagram, Udacity”

	Just a reminder to take down your stack to avoid extra charges on your account.

Load Balancer

	The auto-scaling group needs to have a property that associates it with a target group. The Load Balancer will have a Listener rule associated with the same target group
	See the code details. A discussion about https://stackoverflow.com/questions/48529074/how-is-target-groups-different-from-auto-scaling-groups-in-aws

	Port 80 should be used in Security groups, health checks and listeners associated with the load balancer
	port 80 is in all of them, which uses tcp protocol for http requests, a very useful port

Auto-Scaling

	Students should be using PRIV-NET (private subnets) for their auto-scaling instances

	The machine should have 10 GB or more of disk and should be a t3.small or better.
	You chose an appropriate EC2 instance type for high availability web application.

	There shouldn't be a 'keyname' property in the launch config
	See this link for details on how to ssh to instances in private sunbets. https://serverfault.com/questions/643647/how-to-ssh-to-ec2-instance-in-vpc-private-subnet-via-nat-server

Bonus

	Any values in the output section are a bonus
	see the documentation https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/outputs-section-structure.html

	Any resource of type AWS:EC2::Instance, optional, but nice to have.
	Architecture with bastion for more security, see the documentation https://docs.aws.amazon.com/quickstart/latest/linux-bastion/architecture.html

DOWNLOAD PROJECT

9 CODE REVIEW COMMENTS ›

RETURN TO PATH