**Creating Publicly Accessible RDS with CloudFormation**

AWS [CloudFormation](https://aws.amazon.com/cloudformation/) let us create AWS resources with JSON or YAML files. With CloudFormation, you can create and update your AWS infrastructure by code.

In the previous post, we discuss how we can create publicly available RDS ([How to Make RDS in Private Subnet Accessible From the Internet](https://www.mydatahack.com/how-to-make-rds-in-private-subnet-accessible-from-the-internet/)). In this post, let’s create a CloudFormation template for the public RDS stack.

We are going to use YAML file for the template because it is easier to manage than JSON. In fact, writing CloudFormation template in JSON is much harder as you need to worry about curly brackets and quotations.

**What to create**

From the [previous post](https://www.mydatahack.com/how-to-make-rds-in-private-subnet-accessible-from-the-internet/), we are creating the first design. This one has more resources than the second one and would be more interesting to code.

**Resources**

List of resources we are creating with CloudFormation is below.

* VPC
* Public Subnets
* Private Subnets
* RDS
* Network Load Balancer
* Load Balancer Target Group
* Route Tables
* Security Groups

**Limitations on Target Group RDS IP Address Mapping**

The vanilla CloudFormation does not support getting IP address of the RDS instance. The GetAtt function only returns the endpoint address with Endpoint.Address. For Network Load Balancer, the target group has to be an IP address. Therefore, the mapping of RDS IP address to the target group cannot be done with just using the simple CloudFormation template (there are workarounds you can do by using SDK or custom resources). For this post, let’s keep it simple and accept this as a limitation.

Once it creates all the resources, you need to do nslookup to obtain the IP address of RDS from the endpoint, then create target with the IP and port in target group.

**Using DNS**

If you have your own domain name, you can create a hosted zone and map the Load Balancer as Alias as a record set. In this way, you do not need to change the client connection details every time the stack gets updated or recreated.

**Parameter File**

At the moment, CloudFormation does not support YAML as an external parameter file format. The parameter file is set up as below and will be referenced in the main template. You can fill CIDR ranges for VPC and each subnet.

[cc lang="text" tab\_size="4" lines="-1"

[/cc]

**Execution**

Once the template is ready, we can run the command below from the folder where you keep your template and parameter files.

[cc lang="text" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

[/cc]

**Template**

Here is the actual template. Fill the CIDR range and try running it in your AWS environment. Using CloudFormation does not cost you. But, the resources created will cost. The example only uses the resources from free-tier (the first 12 months one). Even if you ran out of the free tier credit, the cost should be minimal. Have a go!

[cc lang="yaml" tab\_size="4" lines="-1"

[/cc]

--- Ref

You can check how this is created by CloudFormation from the next post ([Creating Publicly Accessible RDS with CloudFormation](https://www.mydatahack.com/creating-publicly-accessible-rds-with-cloudformation/)).