5/8/22, 12:46 PM AWS Proton Workshop



(3)

AWS Proton Workshop > Module 4: Multi-account deployments > Create the services

Create the services

(i) All of the microservices builds will be identical to the ones in your **beta** environment. This is because both, beta and prod, are instances of the same service, and thus share the same repository id and pipeline (that listens for changes from the master branch). Alternatively, you could create separate services with different code repository branches, or create services using a service template that do not include a service pipeline

Let's provision the frontend service in our multi-svc production environment:

```
1 sh -c "~/environment/proton-workshop-helper add-instance --service_name frontend"
```



Please, provide the following values when prompted:

Input	Value
iliput	vatue
Service Instance Name	frontend-prod
Template Name	lb-fargate-service
Environment Name	multi-svc- <your_account_id></your_account_id>
Port	3000
Desired Count	3
Task Size	medium
Scope	public
lmage	<don't_change></don't_change>
Environment Variables	CRYSTAL_URL=http://crystal.protonworkshop.prod.local:3000/crystal NODEJS_URL=http://nodejs.protonworkshop.prod.local:3000
Service Backends	crystal;nodejs

The proton-workshop-helper will a execute similar command to the following example:

```
aws proton update-service \
1
2
      --region ${AWS_REGION} \
3
      --name "frontend-prod" \
      --spec "proton: ServiceSpec
4
5
               instances:
                 - name: 'frontend-beta'
6
                   environment: 'multi-svc-beta'
8
                     port: \"${Proton::CURRENT_VAL}\"
9
                     desired_count: \"${Proton::CURRENT_VAL}\"
10
                     task_size: \"${Proton::CURRENT_VAL}\"
11
                     scope: \"${Proton::CURRENT_VAL}\"
12
                     image: \"${Proton::CURRENT_VAL}\"
13
14
                     env_vars: \"${Proton::CURRENT_VAL}\"
```

Report Workshop Studio bug

© 2008 - 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy policy Terms of use

5/8/22, 12:46 PM AWS Proton Workshop

 \equiv

```
- name: 'frontend-prod'
16
                  environment: 'multi-svc-<your_account_id>'
17
18
                  spec:
                    port: 3000
19
20
                    desired_count: 3
                    task_size: medium
21
                    scope: public
22
23
                    image: public.ecr.aws/z9d2n7e1/nginx:1.19.5
24
                    env_vars: >
                          CRYSTAL_URL=http://crystal.protonworkshop.prod.local:3000/crystal;
25
                          NODEJS_URL=http://nodejs.protonworkshop.prod.local:3000" \
26
```

It is time to repeat the same process with crystal and nodejs.

	Input	Value	
	Service Instance Name	nodejs-prod	
	Template Name	lb-fargate-service	
	Environment Name	multi-svc- <your_account_id></your_account_id>	
	Port	3000	
	Desired Count	3	
	Task Size	medium	
	Scope	private	
	Image	<don't_change></don't_change>	
	Environment Variables	<leave_empty></leave_empty>	
	Service Backends	<leave_empty></leave_empty>	
e the proton-works	shop-helper:		

Previous

Next

1