

# Project-03: blue-green

Create green deployment

```
root@galal-sre-02:~/nd087-c3-deployment-roulette-project/started
root@galal-sre-02:~/nd087-c3-deployment-roulette-project/started
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: green
    name: green
    namespace: udacity
spec:
  replicas: 3
  selector:
    matchLabels:
      app: green
  template:
    metadata:
      labels:
        app: green
    spec:
      containers:
        - name: nginx
          image: nginx
          ports:
            - containerPort: 80
          volumeMounts:
            - mountPath: /usr/share/nginx/html
              name: configmap-vol
      nodeSelector:
        kubernetes.io/os: linux
```

```
volumes:
  - name: configmap-vol
    configMap:
      name: green-config
```

## Creating blue-green.sh

```
vim blue-green.sh

#!/bin/bash

GREEN_PATH="starter/apps/blue-green/green.yml"

# Deploying
echo "[*] Deploying Green"
kubectl apply -f ${GREEN_PATH}

ATTEMPTS=0
ROLLOUT_STATUS_CMD="kubectl rollout status deployment/green -n i
until $ROLLOUT_STATUS_CMD || [ $ATTEMPTS -eq 60 ]; do
  $ROLLOUT_STATUS_CMD
  ATTEMPTS=$((attempts + 1))
  sleep 1
done
echo "[+] Green deployment successful!"

echo "[*] Waiting for green service to come up."
SERVICE_URL="curl $(k get svc | grep green-svc | awk '{print $4}
ATTEMPTS=0
until $SERVICE_URL 2> /dev/null || [ $ATTEMPTS -eq 300 ]; do
  echo "Service URL not available yet ..."
  sleep 2
done
```

```

if ${SERVICE_URL} 2> /dev/null; then
    echo "[+] URL is up"
else
    echo "[-] URL not initialized"
fi

```

Under terraform `infra/` create new terraform that will create k8s green service and green dns record

```

vim infra/green.tf
resource "kubernetes_service" "green" {
  metadata {
    name      = "green-svc"
    namespace = local.name
    annotations = {
      "service.beta.kubernetes.io/aws-load-balancer-type"
      "service.beta.kubernetes.io/aws-load-balancer-nlb-target-t
    }
  }
  spec {
    selector = {
      app = "green"
    }

    port {
      port          = 80
      target_port    = 80
    }

    type = "LoadBalancer"
  }

  depends_on = [
    module.project_eks
  ]
}

```

```

resource "aws_route53_record" "green" {
  zone_id = aws_route53_zone.private_dns.id
  name     = "blue-green"
  type     = "CNAME"
  ttl      = "5"

  weighted_routing_policy {
    weight = 1
  }

  set_identifier = "green"
  records        = [kubernetes_service.green.status.0.load_balancer_ip]
}

```

Runing script to deploy green and waiting for service URL

```

root@galal-sre-02:~/nd087-c3-deployment-roulette-project# ./blue-green.sh
[*] Deploying Green
deployment.apps/green created
Waiting for deployment "green" rollout to finish: 0 of 3 updated replicas are available...
Waiting for deployment "green" rollout to finish: 1 of 3 updated replicas are available...
Waiting for deployment "green" rollout to finish: 2 of 3 updated replicas are available...
deployment "green" successfully rolled out
[+] Green deployment successful!
[*] Waiting for green service to come up.
./blue-green.sh: line 19: k: command not found
Service URL not available yet ...
Service URL not available yet ...
Service URL not available yet ...

```

Getting running pods

```

root@galal-sre-02:~/nd087-c3-deployment-roulette-project# k get
NAME                                READY   STATUS    RESTARTS   AGE
blue-9b56b55cc-55x4c               1/1     Running   0           170r
blue-9b56b55cc-86z8c               1/1     Running   0           170r
blue-9b56b55cc-ttlhv               1/1     Running   0           168r
canary-v2-8dc775cd6-sc46l          1/1     Running   0           77m
canary-v2-8dc775cd6-wfmw7          1/1     Running   0           73m
canary-v2-8dc775cd6-z4rrw          1/1     Running   0           75m
green-568777db7-5npqt              1/1     Running   0           46s
green-568777db7-lcr4l              1/1     Running   0           46s
green-568777db7-rzd2w              1/1     Running   0           46s
hello-world-7b67f6885d-wttg5       1/1     Running   0           168r

```

Terraform create new k8s green-svc and create DNS record

```

root@galal-sre-02:~/nd087-c3-deployment-roulette-project/terraform
...
Enter a value: yes

```

```

kubernetes_service.green: Creating...
kubernetes_service.green: Creation complete after 5s [id=udacity
aws_route53_record.green: Creating...
aws_route53_record.green: Still creating... [10s elapsed]
aws_route53_record.green: Creation complete after 13s [id=Z0764

```

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

```

account_id = "204340383769"
caller_arn = "arn:aws:iam::204340383769:user/admin"
caller_user = "AIDAS7E5C2AMSWFGYAWC2"

```

Output of blue-green.sh

Service URL not available yet ...

```
<html>
```

```
<h1>This is version GREEN</h1>
```

```
</html>
```

```
<html>
```

```
<h1>This is version GREEN</h1>
```

```
</html>
```

```
[+] URL is up
```

Show service

```
root@galal-sre-02:~/nd087-c3-deployment-roulette-project# k get
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP
blue-svc	LoadBalancer	172.20.119.148	a551d01ab67f84b3aa...
canary-svc	ClusterIP	172.20.88.145	<none>
green-svc	LoadBalancer	172.20.82.216	ad12dcaad748b4dc09...
hello-world	LoadBalancer	172.20.84.109	a1d199afdae1643339...

From Route53 blue-green.udacityproject

Route 53 > Hosted zones > udacityproject

Private udacityproject info Delete zone Test record Configure query logging

► Hosted zone details Edit hosted zone

Records (4) Hosted zone tags (2)

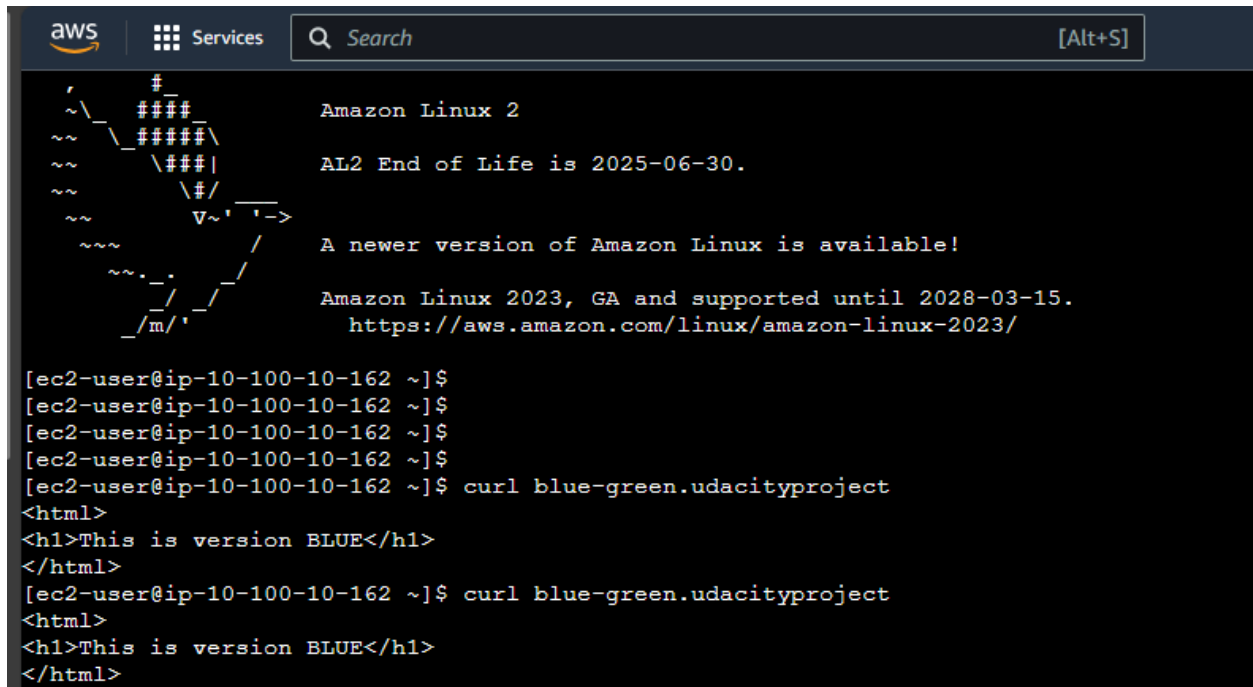
Records (4) info Refresh Delete record Import zone file Create record

Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

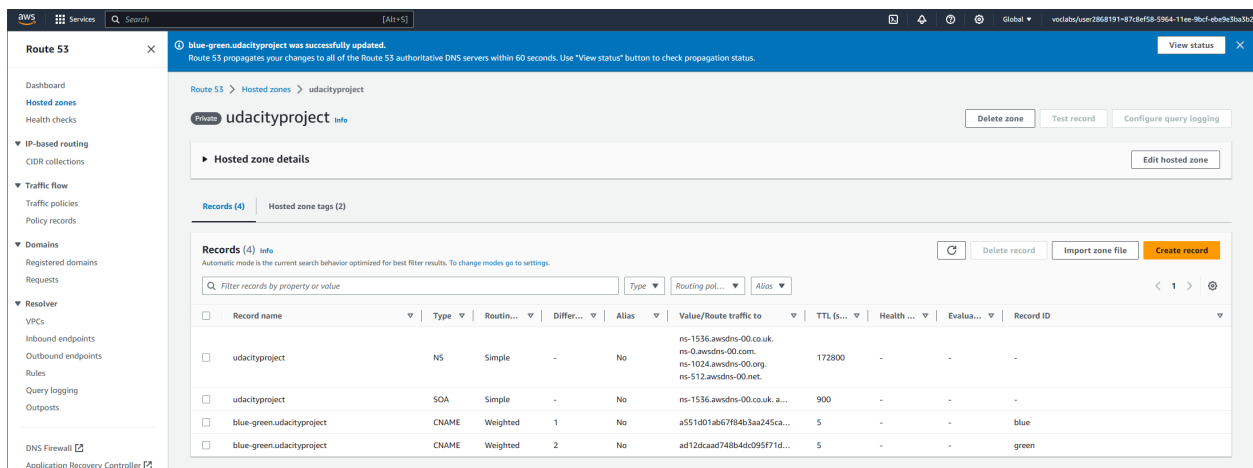
Filter records by property or value Type ▼ Routing pol... ▼ Alias ▼ < 1 > ⚙

<input type="checkbox"/>	Record ... ▼	Type ▼	Routin... ▼	Differ... ▼	Alias ▼	Value/Route traffic to ▼	TTL (s... ▼	Health ... ▼	Evalua... ▼	Record ID ▼
<input type="checkbox"/>	udacitypr...	NS	Simple	-	No	ns-1536.awsdns-00.co.uk. ns-0.awsdns-00.com. ns-1024.awsdns-00.org. ns-512.awsdns-00.net.	172800	-	-	-
<input type="checkbox"/>	udacitypr...	SOA	Simple	-	No	ns-1536.awsdns-00.co.uk. a...	900	-	-	-
<input type="checkbox"/>	blue-gree...	CNAME	Weighted	2	No	a551d01ab67f84b3aa245ca...	5	-	-	blue
<input type="checkbox"/>	blue-gree...	CNAME	Weighted	1	No	ad12dcaad748b4dc095f71d...	5	-	-	green

## Curl from instance



I changed dns weight for blue  $\Rightarrow$  1 and green  $\Rightarrow$  2, so all traffics will go to green



## Testing

```
aws Services Search [Alt+S]
[ec2-user@ip-10-100-10-162 ~]$ curl blue-green.udacityproject
<html>
<h1>This is version GREEN</h1>
</html>
[ec2-user@ip-10-100-10-162 ~]$ curl blue-green.udacityproject
<html>
<h1>This is version GREEN</h1>
</html>
[ec2-user@ip-10-100-10-162 ~]$
```

Now I can destroy blue env

```
root@galal-sre-02:~/nd087-c3-deployment-roulette-project# k delete deployment.apps "blue"
deployment.apps "blue" deleted
root@galal-sre-02:~/nd087-c3-deployment-roulette-project# k get pods
NAME                                READY   STATUS    RESTARTS   AGE
canary-v2-8dc775cd6-sc46l           1/1     Running   0           103r
canary-v2-8dc775cd6-wfmw7           1/1     Running   0           100r
canary-v2-8dc775cd6-z4rrw           1/1     Running   0           102r
green-568777db7-5npqt               1/1     Running   0           27m
green-568777db7-lcr4l               1/1     Running   0           27m
green-568777db7-rzd2w               1/1     Running   0           27m
hello-world-7b67f6885d-wttg5        1/1     Running   0           3h1!
```