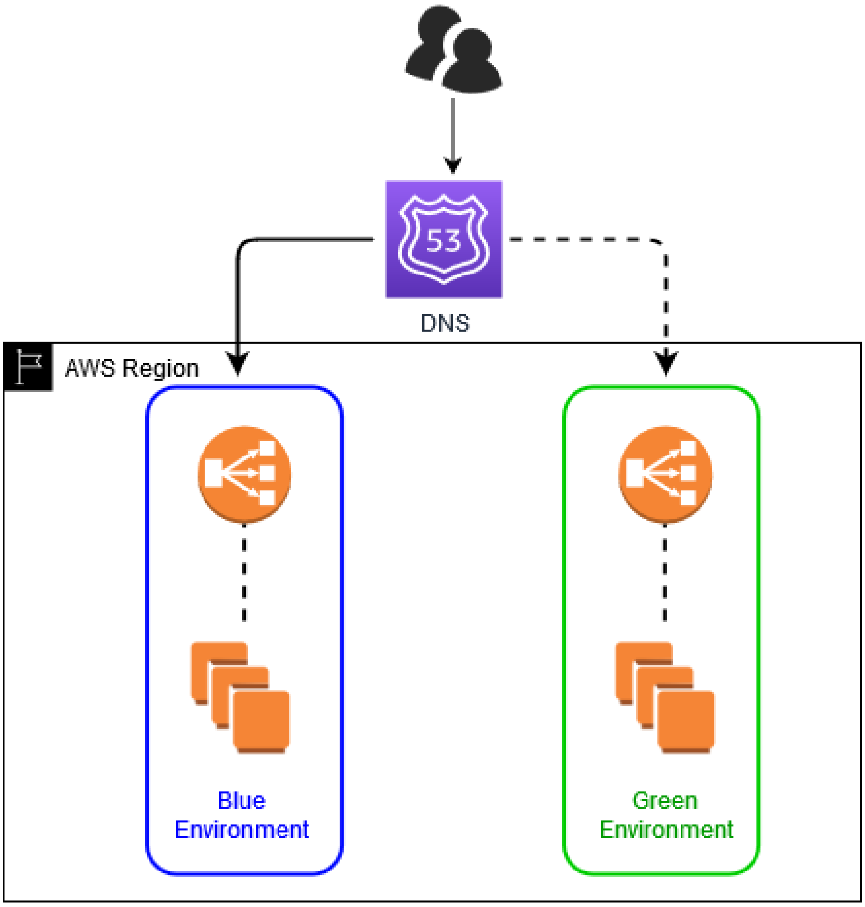
JON BONSO AND KENNETH SAMONTE

AWS CERTIFIED DEVOPS ENGINEER PROFESSIONAL

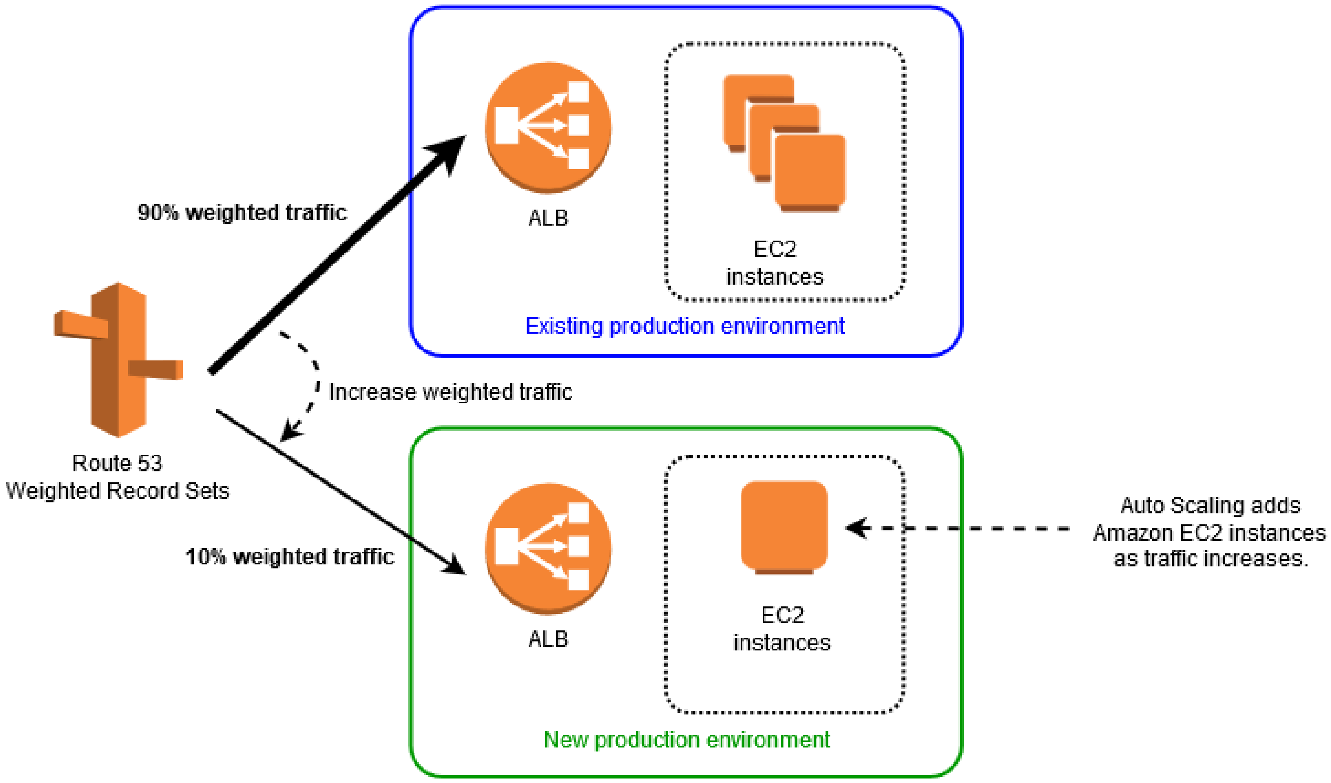
Tutorials Dojo
Study Guide and Cheat Sheets

Basic Blue/Green Deployment using Route 53

Blue/green deployment on the AWS platform provides a safer way to upgrade production software. This deployment usually involves two environments, the production environment (blue) and the new updated environment (green).



Once the new version is deployed on the green environment, you can validate the new software before going live. Then, you start shifting traffic away from the blue environment and sending it to the green one. Normally, you'd use Route 53 weighted routing policy because it gives you an easy way to push incremental traffic to the green environment or revert traffic back to the blue environment in case of issues. If you want to, you can switch the traffic immediately by updating the production Route 53 record to point to the green endpoint. Users will not see that you changed the endpoint since from their perspective, the production URL is the same.



You can also shift a small portion (like 10%) of traffic on the Green environment by using a weighted routing policy on Route 53. This way, you can test live traffic on the new environment, analyze the new logs, and then you can easily revert to the original environment if you find any problems. This process is also called a canary deployment.

Source:
https://aws.amazon.com/blogs/startups/upgrades-without-tears-part-2-bluegreen-deployment-step-by-step-on-aws/

Amazon Route 53 Routing Policies

Most large organizations often have complex network structures to support their hybrid cloud architecture, distributed systems, and global users. They have several on-premises networks integrated with AWS Direct Connect or AWS VPN to connect to their AWS cloud resources across multiple Availability Zones and AWS Regions. To ensure business continuity, companies implement a disaster recovery plan that fails over the production traffic from the primary environment to the disaster recovery (DR) site.

Amazon Route 53 is a global Domain Name System (DNS) service that allows you to route traffic across various AWS Regions and external systems outside of AWS. It provides a variety of routing policies that you can implement to meet your required use cases and automatically monitor the state and performance of your applications, servers, and other resources using health checks. You can combine two or more routing policies to comply with your company's strict RTO and RPO requirements. It helps simplify the process of setting up an active-passive or active-active failover for your disaster recovery plan by intelligently routing traffic from your primary resources to the secondary resources based on the rules you specify.

Your globally distributed resources can either be considered active or passive. It's active if it accepts live production traffic and passive if it is just on standby, which will only be activated during a failover event. You can set up an active-active failover to improve your systems' fault tolerance and performance. By having several active environments, you can ensure the high availability and resiliency of your global applications. To set up an active-active failover, you can use a single or a combination of routing policies such as latency, geolocation, geoproximity, and others to configure Route 53 to respond to a DNS query using any healthy record.

Below are the different types of Amazon Route 53 routing policies that you can use in your architecture:

- **Simple -** This routing policy is commonly used for a single resource that performs a straight-forward function for your domain records. For example, you can use this policy to route traffic from tutorialsdojo.com apex domain to an NGINX web server running on an Amazon EC2 instance.
- **Failover** – As the name implies, you can use this policy to set up an active-passive failover for your network architecture.
- **Geolocation** – Amazon Route 53 can detect the geographical location where the DNS queries originated. This routing policy lets you choose the specific resources that serve incoming traffic based on your users' geographic location. Say, you might want all user traffic from North America routed to an Application Load Balancer in the Singapore region. It works by mapping the IP addresses to geographical areas using the Extension Mechanisms for DNS version 0 (EDNS0).
- **Geoproximity** – This one is similar to the Geolocation routing policy except that it uses the Traffic Flow feature of Route 53 and has an added capability of shifting more or less traffic to your AWS services in one geographical location using a bias. It concentrates on the proximity of the resource in a given geographic area rather than its exact location.

- **Latency** – You can improve the application performance for the benefit of your global users by serving their requests from the AWS Region that provides the lowest latency. This routing policy is suitable for organizations that have resources in multiple AWS Regions.
- **Multivalue Answer** – Unlike the *Simple* routing policy, this type can route traffic to numerous resources in response to DNS queries with up to eight active records selected randomly. This policy is perfect if you are configuring an active-active failover for your network.
- **Weighted** – This policy allows you to route traffic to multiple resources in proportions that you specify. It acts as a load balancer that routes requests to a record based on the relative percentage of traffic or weight that you specified.

To monitor the system status or health, you can use Amazon Route 53 health checks to properly execute automated tasks to ensure the availability of your system. Health checks can also track the status of another health check or an Amazon CloudWatch Alarm.

Sources:
https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html
https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/disaster-recovery-resiliency.html
https://tools.ietf.org/html/rfc2671

Amazon Route 53

- A highly available and scalable Domain Name System (DNS) web service used for domain registration, DNS routing, and health checking.

Routing Internet Traffic to your Website or Web Application

- Use the Route 53 console to register a domain name and configure Route 53 to route internet traffic to your website or web application.
- After you register your domain name, Route 53 automatically creates a **public hosted zone** that has the same name as the domain.
- To route traffic to your resources, you create **records**, also known as *resource record sets*, in your hosted zone.
- You can create special Route 53 records, called **alias records**, that route traffic to S3 buckets, CloudFront distributions, and other AWS resources.
- Each record includes information about how you want to route traffic for your domain, such as:
    - Name - name of the record corresponds with the domain name or subdomain name that you want Route 53 to route traffic for.
    - Type - determines the type of resource that you want traffic to be routed to.
    - Value

Route 53 Health Checks

- Create a health check and specify values that define how you want the health check to work, such as:
    - The IP address or domain name of the endpoint that you want Route 53 to monitor.
    - The protocol that you want Route 53 to use to perform the check: HTTP, HTTPS, or TCP.
    - The **request interval** you want Route 53 to send a request to the endpoint.
    - How many consecutive times the endpoint must fail to respond to requests before Route 53 considers it unhealthy. This is the **failure threshold**.
- You can configure a health check to check the health of one or more other health checks.
- You can configure a health check to check the status of a CloudWatch alarm so that you can be notified on the basis of a broad range of criteria.

Know the following Concepts

- Domain Registration Concepts - domain name, domain registrar, domain registry, domain reseller, top-level domain
- DNS Concepts
    - **Alias record** - a type of record that you can create to route traffic to AWS resources.

  - **Hosted zone** - a container for records, which includes information about how to route traffic for a domain and all of its subdomains.
  - **Name servers** - servers in the DNS that help to translate domain names into the IP addresses that computers use to communicate with one another.
  - **Record** (DNS record) - an object in a hosted zone that you use to define how you want to route traffic for the domain or a subdomain.
  - **Routing policy -** policy on how to redirect users based on configured routing policy
  - **Subdomain** - name below the zone apex. Example: portal.tutorialsdojo.com
  - Time to live (TTL) - time that the DNS record is cached by querying servers.
- Health Checking Concepts
  - **DNS failover** - a method for routing traffic away from unhealthy resources and to healthy resources.
  - Endpoint - the URL or endpoint on which the health check will be performed.
  - Health check - the metric on which to determine if an endpoint is healthy or not.

Records

- Create records in a hosted zone. Records define where you want to route traffic for each domain name or subdomain name. The name of each record in a hosted zone must end with the name of the hosted zone.
- Alias Records
  - Route 53 **alias records** provide a Route 53–specific extension to DNS functionality. Alias records let you route traffic to selected AWS resources. They also let you route traffic from one record in a hosted zone to another record.
  - You can create an alias record at the top node of a DNS namespace, also known as the zone apex.
- CNAME Record
  - You cannot create an alias record at the top node (zone apex) of a DNS namespace using a CNAME record.