# Module-3: Route 53 Assignment - 3

You have been asked to:

- 1.Use the Route 53 Hosted Zone created in the Assignment
- 2.Route the traffic to an EC2 instance with an Apache web server running in it using it's IP address

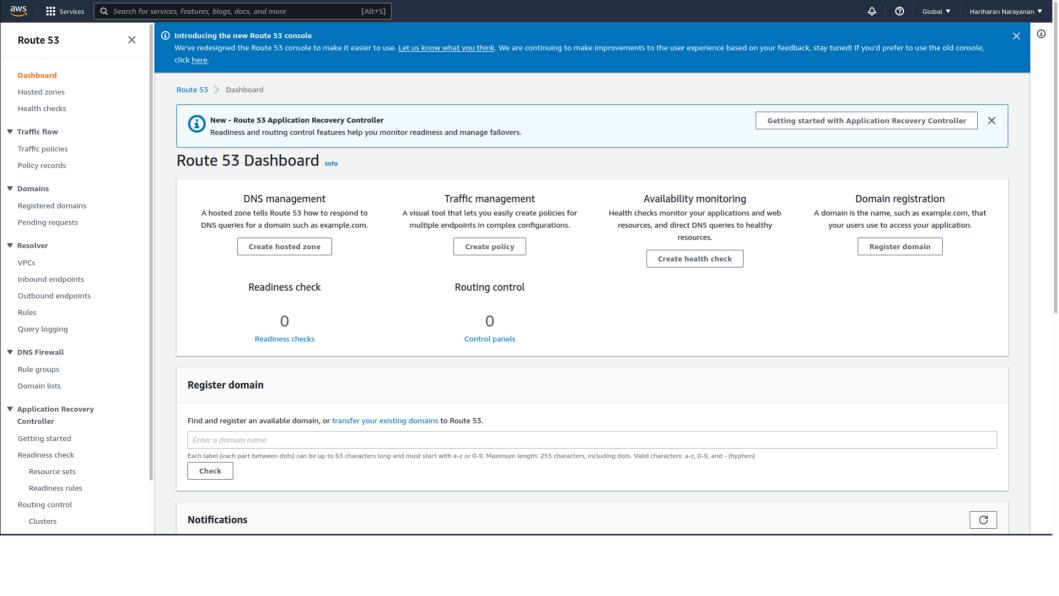
# 1. Create an EC2 instance with Apache web server running in it

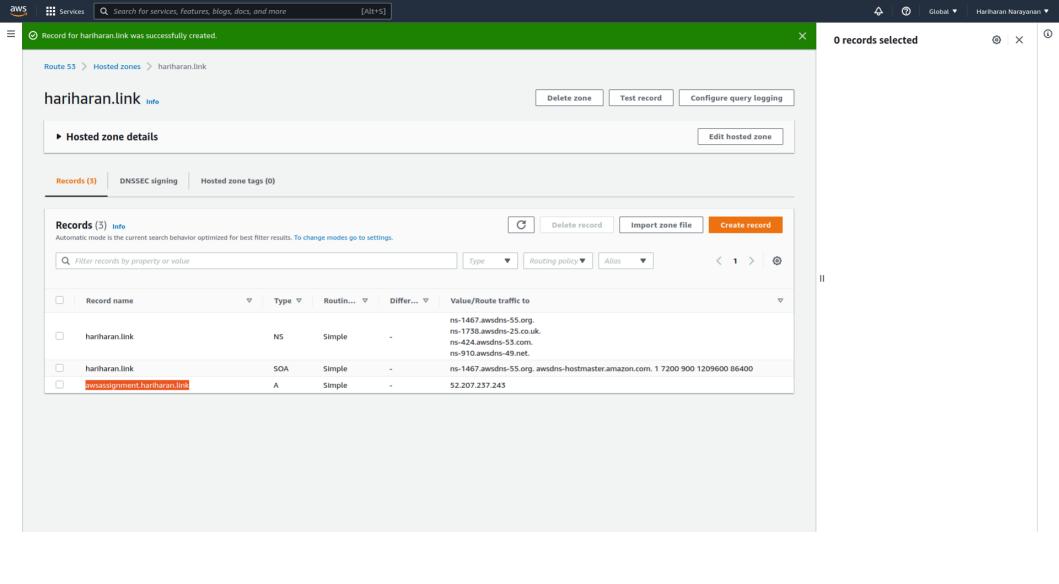
- The steps for this were already provided in answer to module 2 assignment 2
  - Follow the same steps to create an EC2 instance named \*\*module\_2\_assignment\_3\_ec2\_vm1\*\* and install and configure apache2 in it.
  - Verify from a browser that typing the dns name of the EC2 instance in the address bar will take to the \*\*Apache2 Ubuntu Default Page\*\*.

- 2. Use Route 53 to register a DNS name called \_hariharan.link\_
- Open the AWS Route 53 console -<a href="https://console.aws.amazon.com/route53">https://console.aws.amazon.com/route53</a>
- In "Route 53 Dashboard", enter
  \*\*hariharan.link\*\* in the text box under "Register Domain"
- Finish the purchase of this domain.

# 3. Use Route 53 to route traffic

- Open the AWS Route 53 console <a href="https://console.aws.amazon.com/route53">https://console.aws.amazon.com/route53</a>
- Navigate to \*\*Hosted zones\*\* and select \*\*hariharan.link\*\*. Click on \*\*View details\*\*
- In the "hariharan.link" page, click on \*\*Create record\*\*
- In the "Quick create record" page...
  - In the field "Record name", enter \*\*awsassignment\*\* so that the full DNS name reads as \*\*awsassignment.hariharan.link\*\*
  - In the field "Routing policy", choose \*\*Simple routing\*\*
  - In the field "Value", paste the value of \*\*IPv4 Public IP\*\* copied from the EC2 console of the EC2 instance created above. Finally click on \*\*Create records\*\*
- In the "hariharan.link" page, verify that a record for \_awsassignment.hariharan.link\_ is now added
- Finally, open the link http://awsassignment.hariharan.link in a browser to verify that the \*\*Apache2 Ubuntu Default Page\*\* is opened. This confirms that the DNS record in Route 53 created correctly.







# **Apache2 Ubuntu Default Page**

# It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

## **Configuration Overview**

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is fully documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the manual if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

/etc/apache2/ |-- apache2.conf `-- ports.conf I-- mods-enabled 1-- \*.load -- \*.conf |-- conf-enabled `-- \*.conf I-- sites-enabled `-- \*.conf

- apache2.conf is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- · ports.conf is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the mods-enabled/, conf-enabled/ and sites-enabled/ directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- . They are activated by symlinking available configuration files from their respective \*-available/ counterparts. These should be managed by using our helpers a2enmod, a2dismod, a2ensite, a2dissite, and a2enconf, a2disconf. See their respective man pages for detailed information.
- . The binary is called apache2. Due to the use of environment variables, in the default configuration, apache2 needs to be started/stopped with /etc/init.d/apache2 or apache2ctl. Calling /usr/bin/apache2 directly will not work with the default configuration.