BUILD A WEBSITE USING AWS LIGHTSAIL

(Mini-Project)

A project report submitted to the Srinivas University as partial fulfillment for the award of the degree of

Bachelor of Technology in Cloud Technology and Information Security

Submitted By

DIVYA SOMAPPA LAMANI

USN: 1SU19CI011

Under the Guidance of

Mr.Daniel Selvaraj

Professor

Department of Cloud Technology & Data Science

College of Engineering and Technology

SRINIVAS UNIVERSITY

Mukka, Mangalore – 574146

January 2022

BONAFIDE CERTIFICATE

This is to certify that this project report entitled "**BUILD A WEBSITE USING AWS LIGHTSAIL**" is submitted to Srinivas University College of Engineering and Technology, Mukka, is a bonafide record of work done by **DIVYA SOMAPPA LAMANI** under my supervision from 1ST of November 2022 to 28th of November 2022

Mrs. Daniel Selvaraj

Professor

Prof. Daniel Selvaraj

Head of Department

Department of Cloud Technology and Data Science

Srinivas University, Mukka

Date:

Place: Mukka

Abstract:

Amazon Lightsail provides easy-to-use cloud resources to get web applications or websites up and running in just a few clicks.With Amazon Lightsail, we launched a WordPress site on a virtual server. The virtual server will launch in minutes, with WordPress installed and all the benefits of running a server on AWS, including reliability and security. I also used Lightsail's load balancing feature to further optimize your WordPress site and accommodate for variations in traffic, providing a seamless experience for your visitors. And also Auto scaling feature to add new instances when traffic goes high.

Intruduction:

1. THE DOMAIN

Lightsail is a service offering from AWS, which bundles different services together for one price. All packages include DNS management, a static IP, EBS storage, and an EC2 instance (of the T2 types).Amazon Lightsail is a virtual private server (VPS) provider and is the easiest way to get started with AWS for developers, small businesses, students, and other users who need a solution to build and host their applications on cloud. Lightsail provides developers compute, storage, and networking capacity and capabilities to deploy and manage websites and web applications in the cloud. Lightsail includes everything you need to launch your project quickly – virtual machines, containers, databases, CDN, load balancers, DNS management etc. – for a low, predictable monthly price.

You can use Lightsail features to simply host static content, connect your content to an audience around the globe, or get your Windows Business server up and running. The Lightsail console guides you through the configuration process, and in many cases, has components already configured.
Lightsail offers virtual servers (instances) that are easy to set up and backed by the power and reliability of AWS. You can launch your website, web application, or project in minutes, and manage your instance from the intuitive Lightsail console or API.

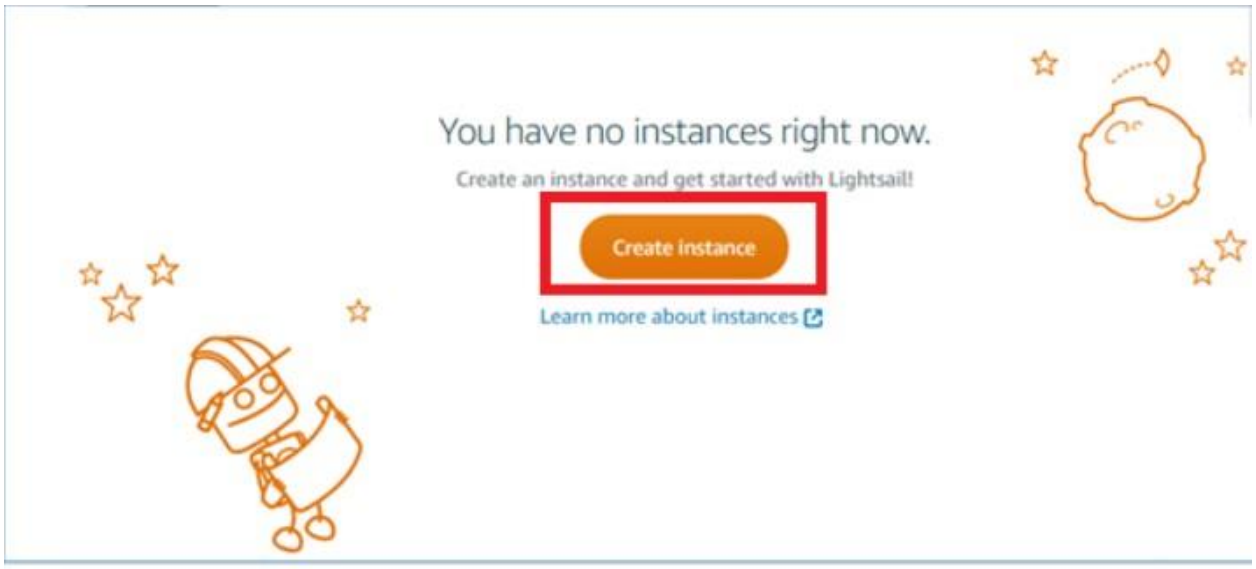
As a creating instance, Lightsail lets you click-to-launch a simple operating system (OS), a pre-configured application, or development stack—such as WordPress, Windows, Plesk, LAMP, Nginx, and more. Every Lightsail instance comes with a built-in firewall allowing you to allow or restrict traffic to your instances based on source IP, port, and protocol.

Implimentation:

Step 1:

Create a Lightsail Instance

Once you sign up with Amazon Lightsail, you can create your first Lightsail Instance.



2.Select Instance Image

3. Price plans

4.assign static IP

5.Deploying the load balancer and configuring SSL

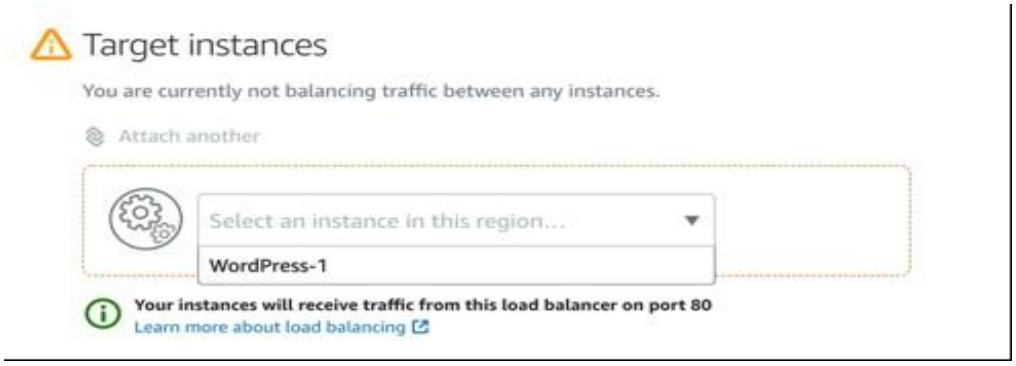
To deploy a Lightsail load balancer and configure it to support SSL, complete the following steps:

1. Open the Lightsail console.
2. From the menu, choose Networking.
3. Choose Create Load Balancer.
4. For Identify your load balancer, enter a name for your load balancer.
5. Choose Create Load Balancer.

The details page for your new load balancer opens. From here, add your initial WordPress server to the load balancer and configure SSL.

6. For Target instances, choose your WordPress server.

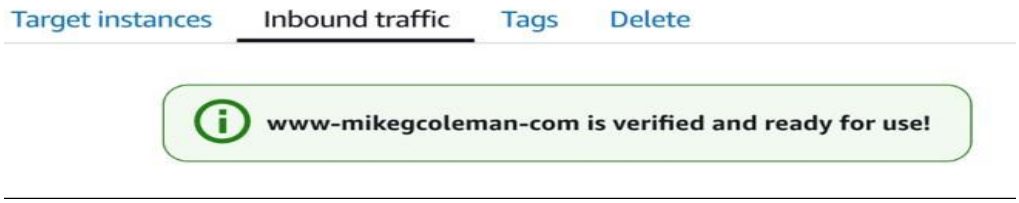
The following screenshot indicates that this post chooses the server WordPress-1.



7. Choose Attach.

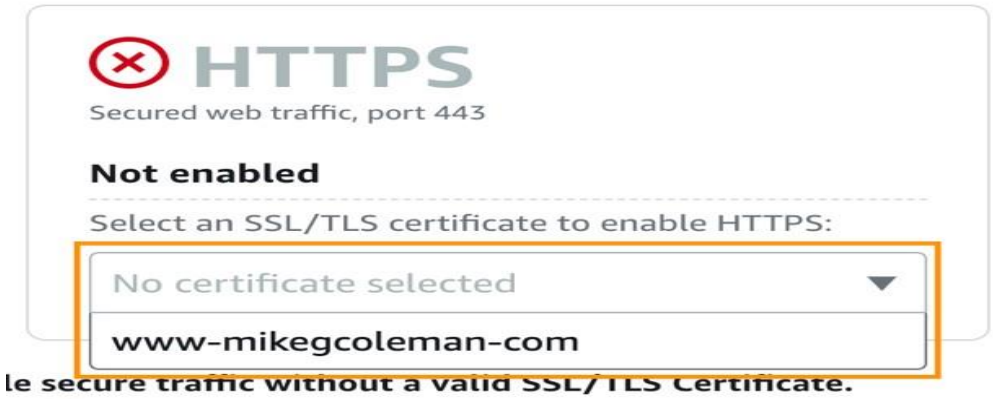
It can take a few seconds for your instance to attach to the load balancer and the Health Check to report as Passed. See the following screenshot of the Health Check status.

8. From the menu, choose Inbound traffic.
9. Under Certificates, choose Create certificate.
10. For PRIMARY DOMAIN, enter the domain name that you want to use to reach your WordPress site.



25. In the HTTPS box, select your certificate from the drop-down menu.

The following screenshot shows the HTTPS box.



le secure traffic without a valid SSL/TLS certificate.
ns

26. **C**opy the DNS name for your load balancer.

The following screenshot shows the example DNS name.

27. Return to the Lightsail DNS console and follow steps 13 through 23 as a guide in creating a CNAME record that maps your website address to the load balancer's DNS name.

    Use the subdomain you chose for your WordPress server

    The following screenshot shows the CNAME record details.

Scaling your WordPress servers

With your WordPress server fully configured, the last step is to create additional instances and place them behind the load balancer so that if one of your WordPress servers fails, your site is still reachable. An added benefit is that your site is more scalable because there are additional servers to handle incoming requests.

Complete the following steps:

1. On the Lightsail console, choose the name of your WordPress server.
2. Choose Snapshots.
3. For instance snapshot, enter the name of your snapshot.

This post uses the name WordPress-version-1. See the following screenshot of your snapshot details.

---

4. Choose Create snapshot.

It can take a few minutes for the snapshot creation process to finish.

5. Click the three-dot menu icon to the right of your snapshot name and choose Create new instance.

The following screenshot shows the Recent snapshots section.

To provide the highest level of redundancy, deploy each of your WordPress servers into a different Availability Zone within the same region. By default, the first server was placed in zone A; place the subsequent servers in two different zones (B and C would be good choices). For more information, see Regions and
Availability Zones.

6. For Instance location, choose Change AWS Region and Availability Zone.
7. Choose Change your Availability Zone.
8. Choose an Availability Zone you have not used previously.

The following screenshot shows the Availability Zones to choose from.

9. Give your instance a new name.

This post names the instance WordPress-2.

10. Choose Create Instance.

You should have at least two WordPress server instances to provide a minimum degree of redundancy. To add more, create additional instances by following steps 1–10.

Return to the Lightsail console home page, and wait for your instances to report back Running.

Adding your instances to the load balancer

Lightsail load balancers offer the following features:

- HTTPS encryption — By default, Lightsail load balancers handle unencrypted (HTTP) traffic requests through port 80. Activate HTTPS encryption by attaching a validated Lightsail SSL/TLS certificate to your load balancer. This allows your load balancer to handle encrypted (HTTPS) traffic requests through port 443. For more information, see SSL/TLS certificates in Amazon Lightsail.

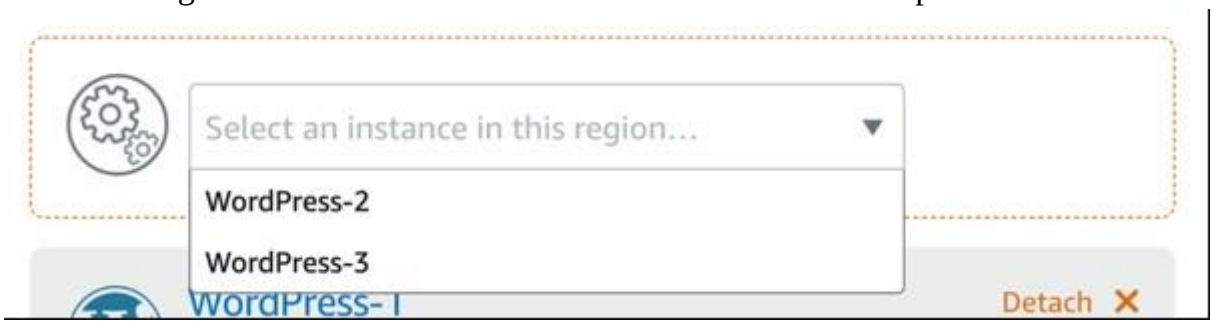
  The following features are available after you activate HTTPS encryption on your load balancer:

  - HTTP to HTTPS redirection — Activate HTTP to HTTPS redirection to automatically redirect HTTP requests to an HTTPS encrypted connection. For more information, see Configuring HTTP to HTTPS redirection for your Amazon Lightsail load balancer.
  - TLS security policies — Configure a TLS security policy on your load balancer. For more information, see Configuring TLS security policies on your Amazon Lightsail load balancers.
- Health checking — By default, health checks are performed on the attached instances at the root of the web application that is running on them. The health checks monitor the health of the instances so that the load balancer can send requests only to the healthy instances. For more information, see Health checking for a Lightsail load balancer.
- Session persistence — Configure session persistence if you're storing session information locally in your website visitors' browsers. For example, you might be running a Magento e-commerce application with a shopping cart on your load-balanced Lightsail instances. If your website visitors add items to their shopping carts, and then end their sessions, when they come back, the shopping cart items will still be there if you configured session persistence. For more information, see Enable session persistence for Amazon Lightsail load balancers.

Now that you have your additional WordPress instances created, add them to the load balancer. This is the same process you followed previously to add the first instance:

1. On the Lightsail console, choose Networking.
2. Choose the load balancer you previously created.
3. Choose Attach another.
4. From the drop-down menu, select the name of your instance.

The following screenshot shows the available instances on the drop-down menu.



5. Choose Attach.
6. Repeat steps 3–5 for any additional instances.

Block public access for buckets in Amazon Lightsail

Amazon Simple Storage Service (Amazon S3) is an object storage service on which customers can store and protect data. The Amazon Lightsail object storage service is built on Amazon S3 technology. Amazon S3 offers account-level block public access, which you can use to limit public access to all S3 buckets in an . Account-level block public access can make all S3 buckets private, regardless of existing individual bucket and object permissions.

When allowing or denying public access, Lightsail object storage buckets take into account the following:

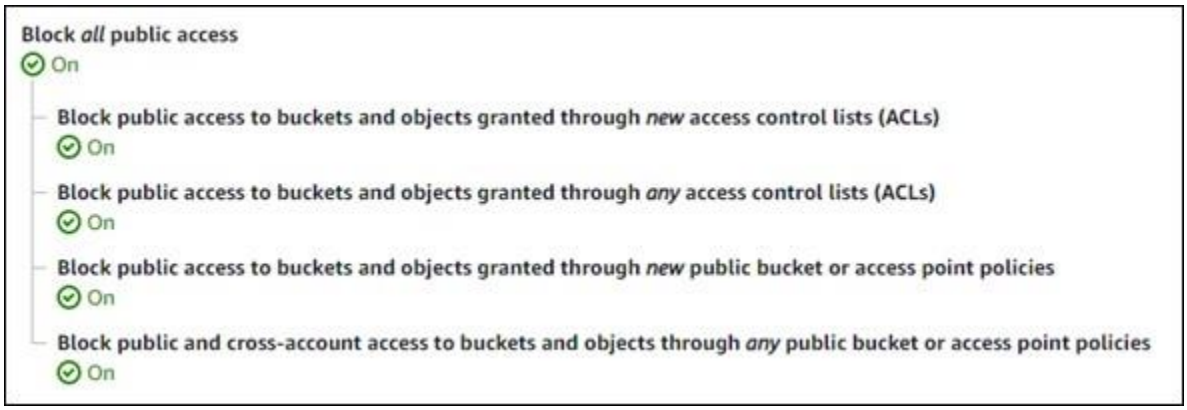
- Lightsail bucket access permissions. For more information see Understanding bucket permissions in Amazon Lightsail.
- Amazon S3 account-level block public access configurations, which override the Lightsail bucket access permissions.

If you turn on account-level Block all public access in Amazon S3, your public Lightsail buckets and objects become private and are no longer publicly accessible.

Configuring block public access settings for your account

You can use the Amazon S3 console, (), SDKs, and REST API to configure block public access settings. You can access the account-level block public access feature in the navigation pane of the Amazon S3 console as shown in the following example.

The Amazon S3 console offers settings to block all public access, block public access granted through new or any access control lists, and block public access to buckets and objects granted through new or any public bucket or access point policies.



You can turn each setting On or Off in the Amazon S3 console. In the API, the corresponding setting is TRUE (On) or FALSE Off).

- Block *all* public access — Turn on this setting to block all public access to your S3 buckets, Lightsail buckets, and their corresponding objects. This setting incorporates all of the following settings. When you turn on this setting, only you (the bucket owner) and authorized users are allowed to access your buckets and their objects. You can only turn this setting on in the Amazon S3 console. It is not available in the Amazon S3 API, or SDKs.
    - Block public access to buckets and objects granted through *new* access control lists (ACLs) — Turn on this setting to block putting public ACLs on buckets and objects. This setting does not impact existing ACLs. Therefore, an object that already has a public ACL remains public. This setting also has no impact on objects that are public due to a bucket access permission being set to All objects are public and read-only. This setting is labeled as `BlockPublicAcls` in the Amazon S3 API.

Uploading files to a bucket in Amazon Lightsail

When you upload a file to your bucket in the Amazon Lightsail object storage service, it is stored as an object. Objects consist of the file data and metadata that describe the object. You can have any number of objects in a bucket.

You can upload any file type—images, backups, data, movies—into a bucket. The maximum file size that you can upload by using the Lightsail console is 2 GB. To upload a larger file, use the Lightsail API, AWS Command Line Interface (AWS CLI), or AWS SDKs.

Lightsail offers the following options depending on the size of the file you want to upload:

- Upload an object up to 2 GB in size using the Lightsail Console — With the Lightsail console, you can upload a single object up to 2 GB in size.

- Upload an object up to 5 GB in size with a single operation using the AWS SDKs, REST API, or AWS CLI — With a single PUT operation, you can upload a single object up to 5 GB in size.
- Upload an object in parts using the AWS SDKs, REST API, or AWS CLI — Using the multipart upload API, you can upload a single large object, of 5 MB to 5 TB in size. The multipart upload API is designed to improve the upload experience for larger objects. You can upload an object in parts. These object parts can be uploaded independently, in any order, and in parallel.

Upload files to a bucket using the Lightsail console

Complete the following procedure to upload files and directories using the Lightsail console.

1. Sign in to the Lightsail console.
2. On the Lightsail home page, choose the Storage tab.
3. Choose the name of the bucket that you want to upload files and folders into.
4. In the Objects tab, perform one of the following actions:
    - Drag and drop files and folders to the Objects page.
    - Choose Upload, and choose File to upload an individual file, or Directory to upload a folder and all of its contents.

    An Upload successful message is displayed when the upload completes.