Configure your Lightsail instances for load balancing

Last updated: November 29, 2017

Before you attach instances to your Lightsail load balancer, you need to evaluate your application's configuration. For example, load balancers often work better when the data tier is separated from the rest of the application. This topic tells you about each Lightsail instance and makes recommendations about whether to load balance (or *horizontally scale*) and how to best configure your application.

General guidelines: Applications that use a database

For Lightsail applications that use a database, we recommend that you separate the database instance from the rest of your application, so that you only have one database instance. The main reason is that you want to avoid writing data to more than one database. If you don't create a single database instance, then the data will be written to the database on whichever instance the user happens to hit.

WordPress

Horizontally scale? Yes, for either a WordPress blog or website.

Configuration recommendations before using a Lightsail load balancer

- Separate your database so that every WordPress instance running behind the load balancer is storing and retrieving information from the same place. If you need more performance from your database, you can replicate or change the processing power or memory independently of your web server.

- Use a WordPress plugin that lets you store your files in Amazon S3. This gives you a centralized place for your content or images, rather than keeping separate copies on each target instance. That way, if you edit your content or change an image, the updates get picked up from the centralized store and your users see the same content, regardless of which instance they hit.

  Find WordPress Amazon S3 plugins

See WordPress on AWS.

Node.js

Horizontally scale? Yes, with some considerations.

Configuration recommendations before using a Lightsail load balancer

- In Lightsail, the Node.js stack powered by Bitnami contains Node.js, Apache, Redis (an in-memory database), and Python. Depending on the application you're deploying, you can load balance across a few servers. However, you would need to configure a load balancer to balance the traffic among all the web servers and move Redis to another server.
- Split the Redis server to another server to communicate with all the instances. Add a database server, if necessary.
- One of the primary use cases for Redis is to cache data locally so you don't have to constantly hit the central database. We recommend that you enable session persistence to leverage the performance

improvement from Redis.
- You can also have a shared Redis node, so you can also share a node or use a local cache on each machine using session persistence.
- Consider including the `mod_proxy_balancer` in the Apache server, if you want to deploy a load balancer using Apache.

For more information, see Scaling Node.js applications.

Magento

**Horizontally scale? ** Yes.

Configuration recommendations before using a Lightsail load balancer

- You can use an AWS reference deployment of Magento that uses additional components, such as an Amazon RDS database: Magento on the AWS Cloud: Quick Start Reference Deployment.
- Be sure to enable session persistence. Magento uses a shopping cart, and this helps ensure that customers who make multiple visits across more than one session will retain items in their shopping carts when they return for a new session. Learn more about session persistence

GitLab

Horizontally scale? Yes, with considerations.

Configuration recommendations before using a Lightsail load balancer

You must have the following:

- A Redis node running and ready to use
- A shared network storage server (NFS)
- A centralized database (MySQL or PostgreSQL) for the application. See the general guidelines about databases, above.

For more information, see the following articles on the GitLab website.

- High Availability
- Configuring GitLab for HA
- High Availability on AWS

Note
The shared network storage server (NFS) referred to in the above links is not currently available with the GitLab blueprint.

Drupal

Horizontally scale? Yes. Drupal has an official document describing how to horizontally scale your application: Server Scaling.

Configuration recommendations before using a Lightsail load balancer

You must set up a Drupal module to synchronize files among different instances. The Drupal website features several modules, but they may be more suitable for prototyping as opposed to production use.

Use a module that lets you store your files in Amazon S3. This gives you a centralized place for your files, rather than keeping separate copies on each target instance. That way, if you edit your files, the updates get picked up from the centralized store and your users see the same files, regardless of which instance they hit.

- Amazon S3 File System
- Content Synchronization

For more information, see Horizontal and Vertical Scaling or Scaling Drupal horizontally and in cloud.

LAMP stack

Horizontally scale? Yes.

Configuration recommendations before using a Lightsail load balancer

- You should create a database on a separate instance. All the instances behind the load balancer should point to this separate database instance so they store and retrieve information from the same place.
- Depending on the application that you want to deploy, think about how to share the file system (NFS, Lightsail block storage disks, or Amazon S3 storage).

MEAN stack

Horizontally scale? Yes.

Configuration recommendations before using a Lightsail load balancer

Move MongoDB to another machine and configure a mechanism to share the root document among the Lightsail instances.

Redmine

Horizontally scale? Yes.

Configuration recommendations before using a Lightsail load balancer

- Get the redmine_s3 plugin to store the attachments on Amazon S3 instead of on the local file system.
- Separate the database to a different instance.

Nginx

Horizontally scale? Yes.

You can have one or more Lightsail instances running Nginx and attached to a Lightsail load balancer. For more information, see Scaling Web Applications with NGINX, Part 1: Load Balancing.

Joomla!

Horizontally scale? Yes, with considerations.

Configuration recommendations before using a Lightsail load balancer

Although there is no official documentation on the Joomla website, there are some discussions on their community forums. Some users managed to horizontally scale their Joomla instances having a cluster with the following configuration:

- A Lightsail load balancer configured to enable session persistence. Learn more about session persistence
- Several Lightsail instances running Joomla attached to the load balancer with the document root of Joomla! synchronized. You can do this using tools like rsync, having an NFS server that is in charge of synchronizing the content among all Lightsail instances, or sharing files using AWS.
- Several database servers configured with a replication cluster.
- The same cache system configured in each Lightsail instance. There are some useful extensions, such as JotCache.