

a. / aws...

Q Type to search

>_

+ ▾

<> Code

Issues 1

Pull requests

Actions

Projects

Security

Insights

This repository has been archived by the owner on Apr 24, 2018. It is now read-only.

aws-refarch-magento

Public archive

forked from

joaoasrosa/quickstart-magento

master ▾

chrs

Go to file

t

Go to

<> About ▾

...

This branch is 17 commits ahead of, 23 commits behind joaoasrosa/quickstart-magento:master .

hyandell

493 commits 7 years ago

145 Commits

...

ci

Update tem...

7 years ago

imag...

Update lau...

7 years ago

scripts

Install Ama...

7 years ago

sub...

Add linux b...

7 years ago

temp...

Update rea...

7 years ago

.gitm...

add quickst...

7 years ago

LICE...

Initial commit

7 years ago

LICE...

Create LICE...

8 years ago

NOTI...

Update noti...

7 years ago

REA...

Update RE...

6 years ago

ARCHIVED

Readme

Apache-2.0, Apache-2.0 licenses found

Code of conduct

Security policy

Activity

Custom properties

31 stars

39 watching

126 forks

Report repository

Releases

No releases published

Packages

No packages published

Languages

Shell 100.0%

Archived in favour of <https://github.com/aws-quickstart/quickstart-magento>

README

Code of conduct

Apache-2.0 license

Apache-2.0 license

Security

Hosting Magento on AWS

https://github.com/amazon-archives/aws-refarch-magento

Page 1 of 3

This reference architecture provides a set of CloudFormation templates to deploy Magento Community Edition on the AWS Cloud.

Magento is an open-source content management system for e-commerce websites. This automated deployment builds a cluster that runs Magento along with optional sample data, which lets you experiment with custom themes and view the web store.

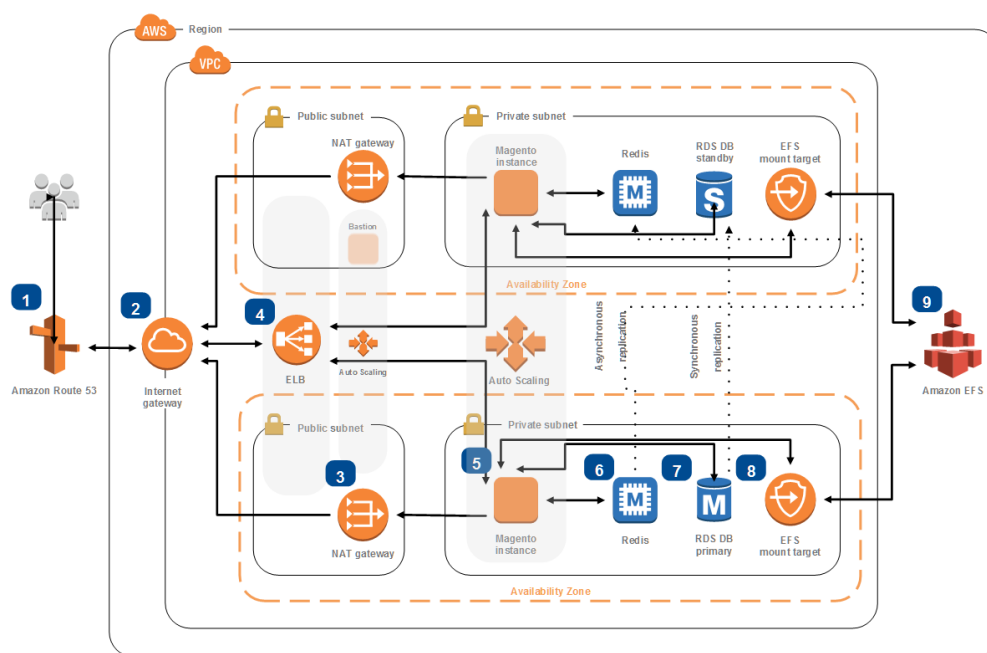
You can launch this CloudFormation stack, using in your own account into: -

- A new VPC - 
- An existing VPC - 

Overview

Magento CE Hosting Running Magento Community Edition (CE) on AWS

Magento Community Edition (CE) is a flexible, open-source commerce platform for developers and small businesses. This reference architecture simplifies the complexity of deploying a scalable and highly available Magento CE commerce platform on AWS.



- 1 Amazon Route 53 provides DNS configuration and routes traffic to Elastic Load Balancing (ELB) endpoints.
- 2 An Internet gateway allows communication between instances in your VPC and the internet.
- 3 NAT gateways in each public subnet enable Amazon EC2 instances in private subnets to access the internet.
- 4 Use an ELB Load Balancer to distribute web traffic across an Auto Scaling group of Amazon EC2 instances in multiple Availability Zones.
- 5 Run your Magento commerce site using an Auto Scaling group of Amazon EC2 instances. Install the latest versions of Magento CE, Nginx web server, and PHP 7. Then, build an Amazon Machine Image (AMI) that the Auto Scaling group launch configuration can use to launch new instances in the group.
- 6 If database access patterns are read-heavy, consider using a caching layer like Amazon ElastiCache for Redis in front of the database layer to cache frequently accessed data.
- 7 Simplify your database administration by running your database layer in Amazon RDS using either Aurora or MySQL.
- 8 Amazon EC2 instances access the shared Magento data in an Amazon EFS file system using mount targets in each Availability Zone in your VPC.
- 9 Use an Amazon EFS network file system so that Magento instances can access your shared, unstructured Magento data such as images, media files etc.



© 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Reference Architectures

This reference architecture deploys Magento using AWS CloudFormation templates and offers two options: you can build a new AWS infrastructure for your Magento stack, or deploy Magento into your existing AWS infrastructure. The deployment uses MySQL on Amazon RDS for database operations, Amazon EFS for shared storage between EC2 instances, and an Amazon ElastiCache cluster with the Redis cache engine to improve application load times. The Magento stack deployed using the templates provided here comes with Amazon Pay plugin, pre-installed.

You can use the AWS CloudFormation templates included here to deploy a fully configured Magento infrastructure in your AWS account. This reference architecture automates the following:

- Deploying Magento Community Edition into a new VPC
- Deploying Magento Community Edition into an existing VPC

You can also use the AWS CloudFormation templates as a starting point for your own implementation.

For architectural details, best practices, step-by-step instructions, and customization options, see the [Magento Quick Start](#)