



LinkedIn profile: <https://www.linkedin.com/in/neel-darji-3a5a8661/>

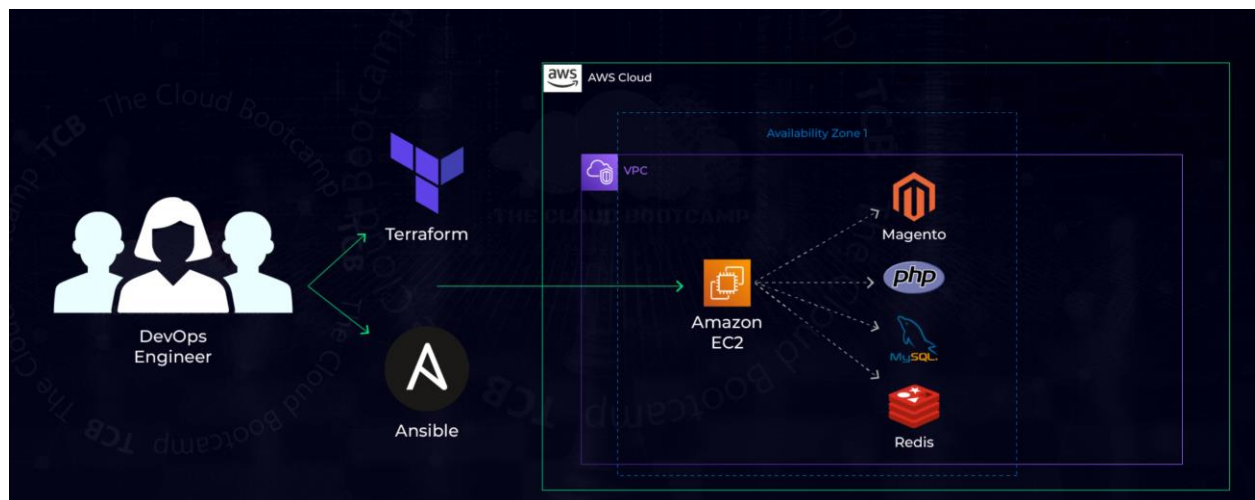
Portfolio: https://medium.com/@Neel_Darji



AWS – Terraform + Ansible Project

Implementation of an E-Commerce System on AWS in an automated way using Terraform and Ansible

Solution Architecture: Neel Darji



- **Project Definition:**

Implementation of an E-Commerce System on AWS in an automated way using Terraform and Ansible

- **Project Description:**

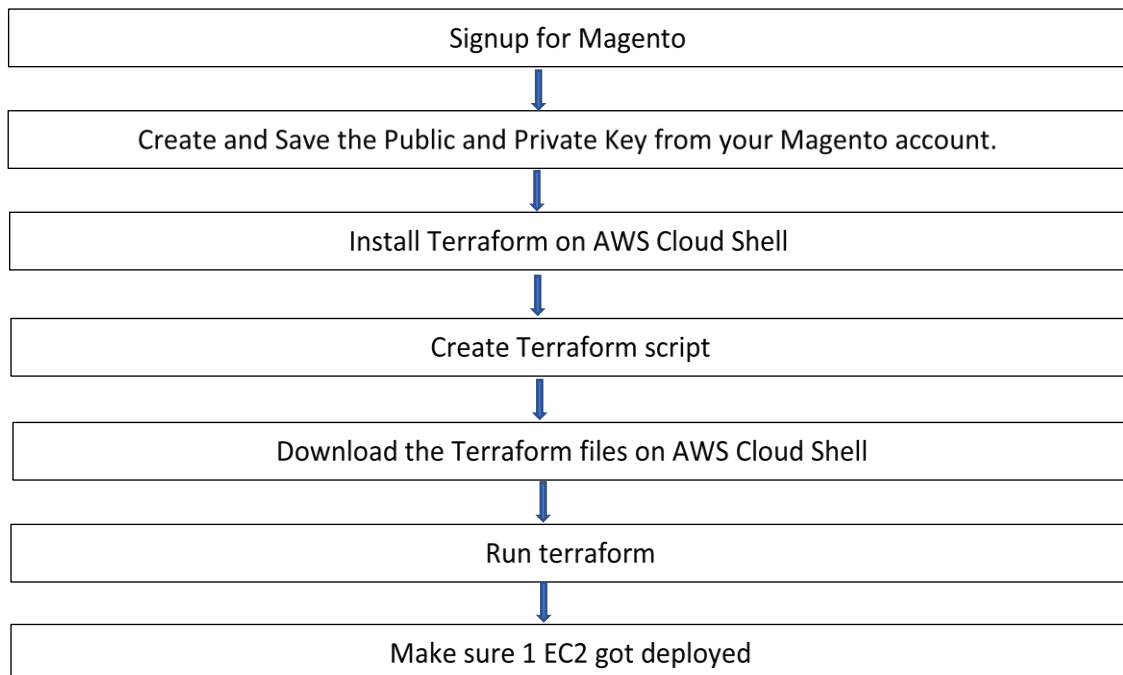
In another project based in a real-world scenario, I worked as Cloud Engineer using DevOps, where I created and implemented an e-Commerce MVP (Minimum Viable Product) on AWS in less than 2 hours and in an automated way using Terraform and Ansible (Infrastructure as Code – IaC).

I provisioned the infrastructure in an automated way using Terraform and Ansible to automate the configuration management process, software installation and package management of the EC2 instance. I also used Magento, PHP, MySQL, and Redis to complete this project.

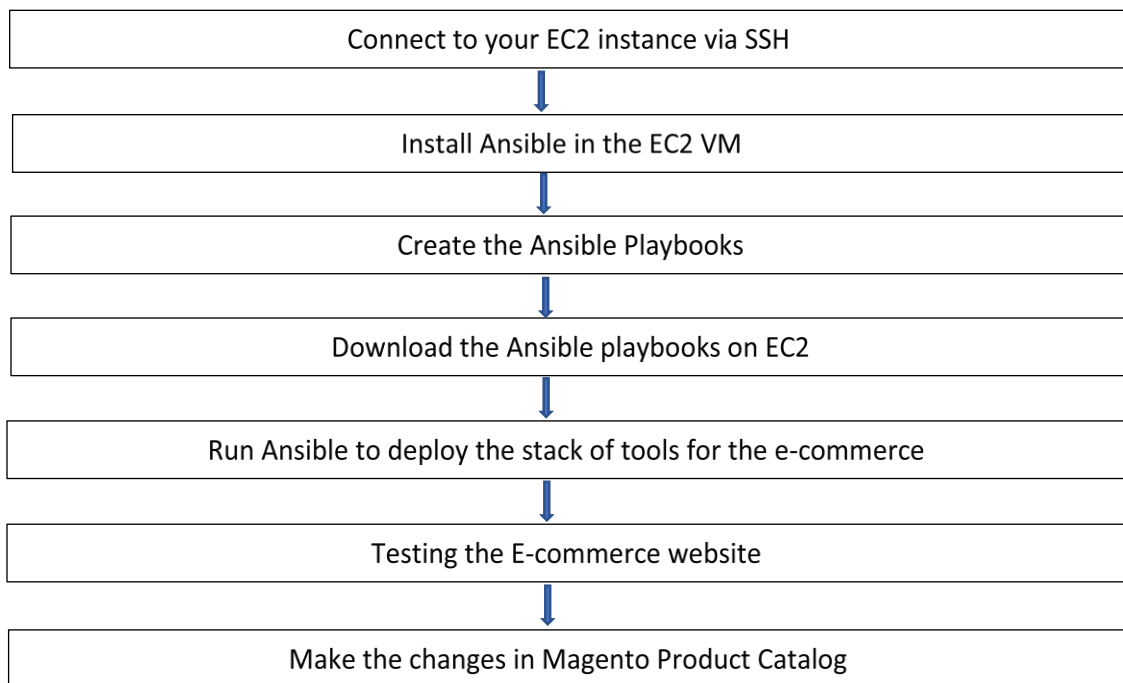
- **Technology used:**

- Terraform
- Ansible
- Amazon EC2
- Magento
- AWS VPC

❖ Project Implementation – Part A - **Terraform**



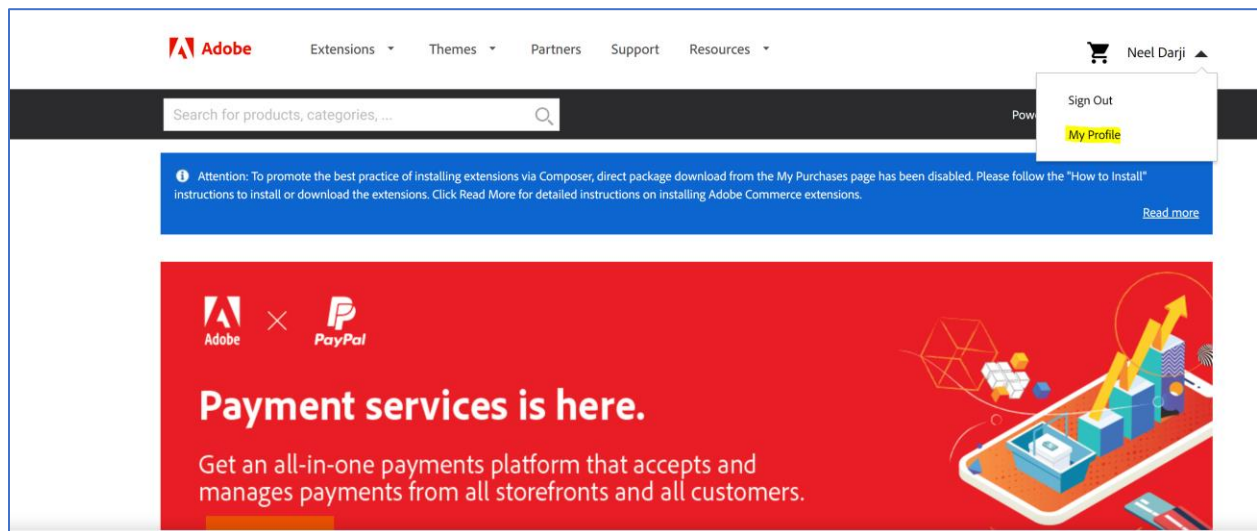
❖ Project Implementation – Part B - **Ansible**



○ **Step-1: Signup for Magento**

Create Magento free account on:

<https://marketplace.magento.com/>



○ **Step-2: Create and Save the Public and Private Key from your Magento account.**

The screenshot shows the Adobe Commerce Developer Portal. At the top, there's a navigation bar with the Adobe logo and links for Extensions, Themes, Partners, Support, and Resources. A search bar is present with the placeholder text "Search for products, categories, ...". Below the navigation bar, a blue banner contains an attention message about installing extensions via Composer. The main content area has three tabs: Marketplace, Adobe Commerce, and Developer Portal. Under the Adobe Commerce tab, there are three sections: My Products (with links for My Subscriptions and Access Keys), Payment (with links for Purchase History and Refunded Orders), and My Information (with a link for Profile Settings).

This screenshot shows the "Create A New Access Key" page. It features a "Magento 2" tab and a "Create A New Access Key" button. Below the button, there's a text block explaining that these keys can be used for authorized downloads of the Adobe Commerce 2.X (M2) platform, extensions, and themes. At the bottom, there's a table with the following structure:

Name	Access Keys	Status	Actions
------	-------------	--------	---------

The screenshot shows a modal dialog titled "Create new access keys". It has a close button (X) in the top right corner. Inside the dialog, there's a label "Keys name:" followed by a text input field containing the text "key01". At the bottom of the dialog, there are two buttons: "Cancel" and "OK". The background of the page is dimmed, showing the same "Create A New Access Key" page as in the previous screenshot.

✓

Changes were saved.

Magento 2

Create A New Access Key

These keys can be used for authorized downloads of the Adobe Commerce 2.X (M2) platform, extensions, and themes. Please see these instruction documents for installing [the platform](#) and [extensions or themes](#).

Name	Access Keys	Status	Actions
key01	Public Key: 5b8765381568bd8e785ddb109dd9f8b8 Copy Private Key: b9b288b25713369e19f7943847f3e003 Copy	Enabled	<div>Disable</div> <div>Delete</div>

Public Key: 5b8765381568bd8e785ddb109dd9f8b8

Private Key: **b9b288b25713369e19f7943847f3e003**

- **Step-3: Install Terraform on AWS Cloud Shell**
 - `sudo yum install -y yum-utils`
 - `sudo yum-config-manager --add-repo`
<https://rpm.releases.hashicorp.com/AmazonLinux/hashicorp.repo>
 - `sudo yum -y install terraform`

```

Package Arch Version Repository Size
-----
Installing:
yum-utils noarch 1.1.31-46.amzn2.0.1 amzn2-core 120 k
Installing for dependencies:
libxml2-python x86_64 2.9.1-6.amzn2.5.4 amzn2-core 246 k
python-charadet noarch 2.2.1-1.amzn2 amzn2-core 246 k
python-kitchen noarch 1.1.1-5.amzn2 amzn2-core 266 k
-----
Transaction Summary
-----
Install 1 Package (+3 Dependent packages)

Total download size: 859 k
Installed size: 4.2 M
Downloading packages:
(1/4): libxml2-python-2.9.1-6.amzn2.5.4.x86_64.rpm | 246 kB 00:00:00
(2/4): python-kitchen-1.1.1-5.amzn2.noarch.rpm | 266 kB 00:00:00
(3/4): yum-utils-1.1.31-46.amzn2.0.1.noarch.rpm | 120 kB 00:00:00
(4/4): python-charadet-2.2.1-1.amzn2.noarch.rpm | 227 kB 00:00:00
-----
Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : python-charadet-2.2.1-1.amzn2.noarch
Installing : python-kitchen-1.1.1-5.amzn2.noarch
Installing : libxml2-python-2.9.1-6.amzn2.5.4.x86_64
Installing : yum-utils-1.1.31-46.amzn2.0.1.noarch
Verifying : yum-utils-1.1.31-46.amzn2.0.1.noarch
Verifying : libxml2-python-2.9.1-6.amzn2.5.4.x86_64
Verifying : python-kitchen-1.1.1-5.amzn2.noarch
Verifying : python-charadet-2.2.1-1.amzn2.noarch
Installed:
yum-utils.noarch 0:1.1.31-46.amzn2.0.1
Dependency Installed:
libxml2-python.x86_64 0:2.9.1-6.amzn2.5.4 python-charadet.noarch 0:2.2.1-1.amzn2 python-kitchen.noarch 0:1.1.1-5.amzn2
Complete!
[scloud@ip-10-1-99-201 ~]$

```

```
[cloudshell-user@ip-10-1-99-201 ~]$ sudo yum-config-manager --add-repo https://rpm.releases.hashicorp.com/AmazonLinux/hashicorp.repo
Loaded plugins: ovl, priorities
adding repo from: https://rpm.releases.hashicorp.com/AmazonLinux/hashicorp.repo
grabbing file https://rpm.releases.hashicorp.com/AmazonLinux/hashicorp.repo to /etc/yum.repos.d/hashicorp.repo
repo saved to /etc/yum.repos.d/hashicorp.repo
[cloudshell-user@ip-10-1-99-201 ~]$
```

```
Dependencies Resolved

=====
Package                               Arch                               Version                               Repository                           Size
=====
Installing:
terraform                             x86_64                             1.2.4-1                             hashicorp                             13 M
Installing for dependencies:
openssl                               x86_64                             1:1.0.2k-24.amzn2.0.3               amzn2-core                             496 k
=====

Transaction Summary
-----
Install 1 Package (+1 dependent package)
Total download size: 13 M
Installed size: 62 M
Downloading packages:
(1/2): openssl-1.0.2k-24.amzn2.0.3.x86_64.rpm | 496 kB 00:00:00
warning: /var/cache/yum/x86_64/2/hashicorp/packages/terraform-1.2.4-1.x86_64.rpm: Header V4 RSA/SHA512 Signature, key ID a3219f7b: NOKEY
Public key for terraform-1.2.4-1.x86_64.rpm is not installed
(2/2): terraform-1.2.4-1.x86_64.rpm | 13 MB 00:00:00
-----
Total
Retrieving key from https://rpm.releases.hashicorp.com/gpg
Importing GPG key 8a32197b:
  Userid : "HashiCorp Security (HashiCorp Package Signing) <security+packaging@hashicorp.com>"
  Fingerprint: e88f 32e9 94d8 e8de a189 d270 d041 8c88 a321 9f7b
  From : https://rpm.releases.hashicorp.com/gpg
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : 1:openssl-1.0.2k-24.amzn2.0.3.x86_64 1/2
  Installing : terraform-1.2.4-1.x86_64 2/2
  Verifying : terraform-1.2.4-1.x86_64 1/2
  Verifying : 1:openssl-1.0.2k-24.amzn2.0.3.x86_64 2/2
Installed:
  terraform.x86_64 0:1.2.4-1
Dependency Installed:
  openssl.x86_64 1:1.0.2k-24.amzn2.0.3
Complete!
[cloudshell-user@ip-10-1-99-201 ~]$
```

○ Step-4: Create Terraform script

I am not sharing here, but in my previous posts, I have shared. You can take reference from there.

○ Step-5: Download the Terraform files on AWS Cloud Shell.

We will create “final_project” directory inside which our Terraform code will run.

- mkdir final_project
- cd final_project
- wget https://tcb-bootcamps.s3.amazonaws.com/bootcamp-aws/en/final-project-terraform.zip
- unzip final-project-terraform.zip

us-east-1

```
[cloudshell-user@ip-10-1-99-201 ~]$ mkdir final_project
[cloudshell-user@ip-10-1-99-201 ~]$ cd final_project
[cloudshell-user@ip-10-1-99-201 final_project]$ ls -ltr
total 0
[cloudshell-user@ip-10-1-99-201 final_project]$
```

```
[cloudshell-user@ip-10-1-99-201 final_project]$ wget https://tcb-bootcamps.s3.amazonaws.com/bootcamp-aws/en/final-project-terraform.zip
--2022-07-09 22:42:34-- https://tcb-bootcamps.s3.amazonaws.com/bootcamp-aws/en/final-project-terraform.zip
Resolving tcb-bootcamps.s3.amazonaws.com (tcb-bootcamps.s3.amazonaws.com)... 52.217.232.97
Connecting to tcb-bootcamps.s3.amazonaws.com (tcb-bootcamps.s3.amazonaws.com)[52.217.232.97]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 948 [application/zip]
Saving to: 'final-project-terraform.zip'

100%[=====] 948 ---K/s in 0s

2022-07-09 22:42:34 (14.0 MB/s) - 'final-project-terraform.zip' saved [948/948]

[cloudshell-user@ip-10-1-99-201 final_project]$ unzip final-project-terraform.zip
Archive: final-project-terraform.zip
  creating: terraform/
  inflating: terraform/main.tf
  inflating: terraform/provider.tf
[cloudshell-user@ip-10-1-99-201 final_project]$ ls -ltr
total 0
drwxr-xr-x 2 cloudshell-user cloudshell-user 4096 Jun 17 15:56 terraform
-rw-rw-r-- 1 cloudshell-user cloudshell-user 948 Jul 7 00:22 final-project-terraform.zip
[cloudshell-user@ip-10-1-99-201 final_project]$
```

```
[cloudshell-user@ip-10-1-99-201 final_project]$ ls -ltr
total 8
drwxrwxr-x 2 cloudshell-user cloudshell-user 4096 Jun 17 15:56 terraform
-rw-rw-r-- 1 cloudshell-user cloudshell-user 948 Jul 7 00:22 final-project-terraform.zip
[cloudshell-user@ip-10-1-99-201 final_project]$ cd terraform/
[cloudshell-user@ip-10-1-99-201 terraform]$ ls -ltr
total 8
-rw-rw-r-- 1 cloudshell-user cloudshell-user 98 Nov 30 2021 provider.tf
-rw-rw-r-- 1 cloudshell-user cloudshell-user 1041 Jun 17 15:57 main.tf
[cloudshell-user@ip-10-1-99-201 terraform]$ vi main.tf
```

Vpc: vpc-0de6d21a52cfe5c3e

- **Step-6: Run Terraform.**
 - cd terraform
 - terraform init
 - terraform plan
 - terraform apply

```
+ setr          = raise
+ to_port       = 80
},
+ {
  + cidr_blocks  = [
    + "0.0.0.0/0",
  ]
  + description  = "SSH to EC2"
  + from_port    = 22
  + ipv6_cidr_blocks = []
  + prefix_list_ids = []
  + protocol     = "tcp"
  + security_groups = []
  + self         = false
  + to_port      = 22
},
]
+ name              = "allow_ssh_http"
+ name_prefix       = (known after apply)
+ owner_id          = (known after apply)
+ revoke_rules_on_delete = false
+ tags              = {
  + "Name" = "allow_ssh_http"
}
+ tags_all          = {
  + "Name" = "allow_ssh_http"
}
+ vpc_id            = "vpc-0de6d21a52cfe5c3e"
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

```
aws_security_group.allow_ssh_http: Creating...
aws_security_group.allow_ssh_http: Creation complete after 3s [id=sg-0f0c0cd933b1f5a98]
aws_instance.ecommerce1: Creating...
aws_instance.ecommerce1: Still creating... [10s elapsed]
aws_instance.ecommerce1: Creation complete after 12s [id=i-08587e78934b4a187]
```

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

```
[cloudshell-user@ip-10-1-153-191 terraform]$
```

- Step-7: Connect to your EC2 instance via SSH

```
PROD+NDarji@Q91BBRV2 MINGW64 ~ (master)
$ cd Downloads/

PROD+NDarji@Q91BBRV2 MINGW64 ~/Downloads (master)
$ cd AWS
AWS/
AWS05/
AWSIAMIRL1013.logs.zip
AWSIAMIRLW0001.logs.zip
AWSIAMSGP1178-machine (1).config
AWSIAMSGP1178-machine.config
AWSIAMSGP1178.Logs.zip
AWSIAMSGP1178.Core.dll
AWSSDK.EC2.dll
AWSSDK.Route53.dll
AWSSDK.Route53Domains.dll
AWSSDK.Route53Resolver.dll
AWS_2/

PROD+NDarji@Q91BBRV2 MINGW64 ~/Downloads (master)
$ cd AWS05/

PROD+NDarji@Q91BBRV2 MINGW64 ~/Downloads/AWS05 (master)
$ ls
sshkey1.pem

PROD+NDarji@Q91BBRV2 MINGW64 ~/Downloads/AWS05 (master)
$ ssh ec2-user@44.202.204.136 -i sshkey1.pem
The authenticity of host '44.202.204.136 (44.202.204.136)' can't be established.
ED25519 key fingerprint is SHA256:Yw516FX0uMH004ptY2s1+OATtpa1N+kQ3HozTyeiv4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '44.202.204.136' (ED25519) to the list of known hosts.

  _ _ _ _ _
 _ _ _ _ _ ) Amazon Linux AMI
 _ _ _ _ _

https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
32 package(s) needed for security, out of 50 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-6-137 ~]$
```

- Step-8: Install Ansible in the EC2 VM

- sudo yum-config-manager --enable epel
- sudo yum install ansible -y

```
[ec2-user@ip-172-31-6-137 ~]$ sudo yum-config-manager --enable epel
Loaded plugins: priorities, update-motd, upgrade-helper
===== repo: epel =====
[epel]
async = True
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/latest
baseurl =
cache = 0
cachedir = /var/cache/yum/x86_64/latest/epel
check_config_file_age = True
compare_providers_priority = 80
cost = 1000
deltarpm_metadata_percentage = 100
deltarpm_percentage =
enabled = 1
enablegroups = True
exclude =
failovermethod = priority
ftp_disable_epsv = False
ggpcadir = /var/lib/yum/repos/x86_64/latest/epel/gpgcadir
gpgcakey =
gpgcheck = True
gpgdir = /var/lib/yum/repos/x86_64/latest/epel/gpgdir
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
hdrdir = /var/cache/yum/x86_64/latest/epel/headers
http_caching = all
includepkgs =
ip_resolve =
keepalive = True
keepcache = False
mddownloadpolicy = sqllite
mdpolicy = group:small
mediaid =
metadata_expire = 21600
metadata_expire_filter = read-only:present
metalink =
minrate = 0
mirrorlist = https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=x86_64
mirrorlist_expire = 86400
name = Extra Packages for Enterprise Linux 6 - x86_64
old_base_cache_dir =
password =
persistdir = /var/lib/yum/repos/x86_64/latest/epel
pkgdir = /var/cache/yum/x86_64/latest/epel/packages
priority = 99
proxy = False
```



```

Installing : python26-setuptools-36.2.7-1.35.amzn1.noarch 9/20
Installing : python26-babel-0.9.4-5.1.8.amzn1.noarch 10/20
Installing : python26-simplejson-3.6.5-1.12.amzn1.x86_64 11/20
Installing : python26-httplib2-0.18.1-1.13.amzn1.noarch 12/20
Installing : python26-PyYAML-3.10-3.10.amzn1.x86_64 13/20
Installing : python-crypto2.6-2.6.1-2.e16.x86_64 14/20
Installing : python26-pyasn1-0.1.7-2.9.amzn1.noarch 15/20
Installing : python-keyczar-0.71c-1.e16.noarch 16/20
Installing : python26-markupsafe-0.11-4.6.amzn1.x86_64 17/20
Installing : python-jinja2-2.6-2.6-3.e16.noarch 18/20
Installing : sshpass-1.06-1.e16.x86_64 19/20
Installing : ansible-2.6.20-1.e16.noarch 20/20
Verifying : python26-paramiko-1.15.1-2.7.amzn1.noarch 1/20
Verifying : python26-setuptools-36.2.7-1.35.amzn1.noarch 2/20
Verifying : python26-backports-1.0-3.14.amzn1.x86_64 3/20
Verifying : sshpass-1.06-1.e16.x86_64 4/20
Verifying : python-keyczar-0.71c-1.e16.noarch 5/20
Verifying : python26-six-1.8.0-1.23.amzn1.noarch 6/20
Verifying : python26-ecdsa-0.11-3.3.amzn1.noarch 7/20
Verifying : python26-babel-0.9.4-5.1.8.amzn1.noarch 8/20
Verifying : python26-simplejson-3.6.5-1.12.amzn1.x86_64 9/20
Verifying : python26-httplib2-0.18.1-1.13.amzn1.noarch 10/20
Verifying : python26-PyYAML-3.10-3.10.amzn1.x86_64 11/20
Verifying : python26-backports-ssl_match_hostname-3.4.0.2-1.12.amzn1.noarch 12/20
Verifying : python26-2.6.9-2.92.amzn1.x86_64 13/20
Verifying : python26-crypto-2.6.1-1.15.amzn1.x86_64 14/20
Verifying : ansible-2.6.20-1.e16.noarch 15/20
Verifying : python-jinja2-2.6-2.6-3.e16.noarch 16/20
Verifying : python-crypto2.6-2.6.1-2.e16.x86_64 17/20
Verifying : python26-pyasn1-0.1.7-2.9.amzn1.noarch 18/20
Verifying : python26-markupsafe-0.11-4.6.amzn1.x86_64 19/20
Verifying : python26-libs-2.6.9-2.92.amzn1.x86_64 20/20

Installed:
ansible.noarch 0:2.6.20-1.e16

Dependency Installed:
python-crypto2.6.x86_64 0:2.6.1-2.e16 python-jinja2-26.noarch 0:2.6-3.e16 python-keyczar.noarch 0:0.71c-1.e16
python26.x86_64 0:2.6.9-2.92.amzn1 python26-PyYAML.x86_64 0:3.10-3.10.amzn1 python26-babel.noarch 0:0.9.4-5.1.8.amzn1
python26-backports.x86_64 0:1.0-3.14.amzn1 python26-backports-ssl_match_hostname.noarch 0:3.4.0.2-1.12.amzn1 python26-crypto.x86_64 0:2.6.1-1.15.amzn1
python26-ecdsa.noarch 0:0.11-3.3.amzn1 python26-httplib2.noarch 0:0.18.1-1.13.amzn1 python26-libs.x86_64 0:2.6.9-2.92.amzn1
python26-markupsafe.x86_64 0:0.11-4.6.amzn1 python26-paramiko.noarch 0:1.15.1-2.7.amzn1 python26-pyasn1.noarch 0:0.1.7-2.9.amzn1
python26-setuptools.noarch 0:36.2.7-1.35.amzn1 python26-simplejson.x86_64 0:3.6.5-1.12.amzn1 python26-six.noarch 0:1.8.0-1.23.amzn1
sshpas.x86_64 0:1.06-1.e16

Complete!
[ec2-user@ip-172-31-6-137 ~]$

```

- **Step-9: Create Ansible Playbooks.** (I am not sharing here, but you can refer my previous posts in which I have shown how to create Ansible Playbooks)
- **Step-10: Download the Ansible playbooks**
 - `wget https://tcb-bootcamps.s3.amazonaws.com/bootcamp-aws/en/final-project-ansible-magento2.zip`
 - `unzip final-project-ansible-magento2.zip`

```

extracting: ansible-magento2/.git/refs/remotes/origin/HEAD
creating: ansible-magento2/.git/refs/tags/
inflating: ansible-magento2/ansible-magento2.yml
extracting: ansible-magento2/ansible.cfg
creating: ansible-magento2/group_vars/
inflating: ansible-magento2/group_vars/all.yml
extracting: ansible-magento2/hosts.yml
inflating: ansible-magento2/LICENSE
inflating: ansible-magento2/README.md
creating: ansible-magento2/roles/
inflating: ansible-magento2/roles/.DS_Store
creating: ansible-magento2/roles/common/
creating: ansible-magento2/roles/common/tasks/
inflating: ansible-magento2/roles/common/tasks/main.yml
creating: ansible-magento2/roles/composer/
creating: ansible-magento2/roles/composer/tasks/
inflating: ansible-magento2/roles/composer/tasks/main.yml
creating: ansible-magento2/roles/httpd/
inflating: ansible-magento2/roles/httpd/.DS_Store
creating: ansible-magento2/roles/httpd/handlers/
inflating: ansible-magento2/roles/httpd/handlers/main.yml
creating: ansible-magento2/roles/httpd/tasks/
inflating: ansible-magento2/roles/httpd/tasks/main.yml
creating: ansible-magento2/roles/httpd/templates/
inflating: ansible-magento2/roles/httpd/templates/httpd.conf
extracting: ansible-magento2/roles/httpd/templates/ports.conf
inflating: ansible-magento2/roles/httpd/templates/vhost.conf
creating: ansible-magento2/roles/magento/
creating: ansible-magento2/roles/magento/tasks/
inflating: ansible-magento2/roles/magento/tasks/main.yml
creating: ansible-magento2/roles/mysql/
inflating: ansible-magento2/roles/mysql/.DS_Store
creating: ansible-magento2/roles/mysql/handlers/
inflating: ansible-magento2/roles/mysql/handlers/main.yml
creating: ansible-magento2/roles/mysql/tasks/
inflating: ansible-magento2/roles/mysql/tasks/main.yml
creating: ansible-magento2/roles/mysql/templates/
inflating: ansible-magento2/roles/mysql/templates/my.cnf.j2
creating: ansible-magento2/roles/php/
creating: ansible-magento2/roles/php/handlers/
inflating: ansible-magento2/roles/php/handlers/main.yml
creating: ansible-magento2/roles/php/tasks/
inflating: ansible-magento2/roles/php/tasks/main.yml
creating: ansible-magento2/roles/redis/
creating: ansible-magento2/roles/redis/tasks/
inflating: ansible-magento2/roles/redis/tasks/main.yml
[ec2-user@ip-172-31-6-137 ~]$

```

```

--
# Magento Domain
magento_domain: http://44.202.204.136/

# This is used for the httpd vhost configuration and folders creation.
server_hostname: 44.202.204.136

# Magento Repo API Key and Secret Key
repo_api_key: 5b8765381568bd8e785ddb109dd9f8b8
repo_secret_key: b9b288b25713369e19f7943847f3e003

##### DO NOT CHANGE BELOW LINES #####

# Which version of Magento to deploy: http://www.magentocommerce.com/download
# You can also see release notes here: https://devdocs.magento.com/guides/v2.2/release-notes/bk-release-notes.html
magento_version: 2.3.0 #

# These are the Magento database settings
magento_db_host: localhost
magento_db_name: magento
magento_db_user: magento
magento_db_password: SomeDBPassword

# You shouldn't need to change this. Not being used at the moment.
mysql_port: 3306

# Magento Installation
# Magento Admin info
magento_admin_first: Admin
magento_admin_last: Surname
magento_admin_email: admin@email.com
magento_admin_user: Admin
magento_admin_pass: Strong123Password#

magento_admin_location: securelocation

# Timezone can be found here: https://gist.github.com/CajuCLC/0a92e071c331103d0defa7374e92df0a
magento_timezone: America/New_York

# Currency codes can be found here: https://gist.github.com/CajuCLC/063fd8b91a1748d5b473d2c062f38bd1
magento_currency: USD

# Language to be used
magento_language: en_US

```

- **Step-11: Run Ansible to deploy the stack of tools for the e-commerce**
 - `cd ..`
 - `ansible-playbook -i hosts.yml ansible-magento2.yml -k -vvv --become`

```

[ec2-user@ip-172-31-6-137 ansible-magento2]$ ansible-playbook -i hosts.yml ansible-magento2.yml -k -vvv --become
ansible-playbook 2.6.20
  config file = /home/ec2-user/ansible-magento2/ansible.cfg
  configured module search path = [u'/home/ec2-user/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.6/site-packages/ansible
  executable location = /usr/bin/ansible-playbook
  python version = 2.6.9 (unknown, Aug 4 2020, 23:47:46) [gcc 4.8.5 20150623 (Red Hat 4.8.5-28)]
Using /home/ec2-user/ansible-magento2/ansible.cfg as config file
SSH password:
Parsed /home/ec2-user/ansible-magento2/hosts.yml inventory source with ini plugin

PLAYBOOK: ansible-magento2.yml *****
1 plays in ansible-magento2.yml

PLAY [Install Apache, PHP + Magento and/or Varnish] *****

TASK [Gathering Facts] *****
task path: /home/ec2-user/ansible-magento2/ansible-magento2.yml:6
<localhost> ESTABLISH LOCAL CONNECTION FOR USER: ec2-user
<localhost> EXEC /bin/sh -c 'echo ~ec2-user && sleep 0'
<localhost> EXEC /bin/sh -c 'umask 77 && mkdir -p "" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336 "" && echo ansible-tmp-1657476988.34-213639159093336="" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336 "" ) && sleep 0'
Using module file /usr/lib/python2.6/site-packages/ansible/modules/system/setup.py
<localhost> PUT /home/ec2-user/.ansible/tmp/ansible-local-8903twdYCz/tmpPjivlp To /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336/setup.py
<localhost> EXEC /bin/sh -c 'chmod u+x /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336/ /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336/setup.py && sleep 0'
<localhost> EXEC /bin/sh -c 'sudo -H -S -n -u root /bin/sh -c ""'echo BECOME-SUCCESS-stazlghoepdipvqulgwpcywcokxes; /usr/bin/python /home/ec2-user/.ansible/tmp/ansible-tmp-1657476988.34-213639159093336/setup.py ""' && sleep 0'
ok: [localhost]
META: ran handlers

TASK [Common : Run yum update first] *****
task path: /home/ec2-user/ansible-magento2/roles/common/tasks/main.yml:2
<localhost> ESTABLISH LOCAL CONNECTION FOR USER: ec2-user
<localhost> EXEC /bin/sh -c 'echo ~ec2-user && sleep 0'
<localhost> EXEC /bin/sh -c 'umask 77 && mkdir -p "" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916 "" && echo ansible-tmp-1657476989.36-192390981924916="" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916 "" ) && sleep 0'
Using module file /usr/lib/python2.6/site-packages/ansible/modules/packaging/os/yum.py
<localhost> PUT /home/ec2-user/.ansible/tmp/ansible-local-8903twdYCz/tmpPjivlp To /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916/yum.py
<localhost> EXEC /bin/sh -c 'chmod u+x /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916/ /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916/yum.py && sleep 0'
<localhost> EXEC /bin/sh -c 'sudo -H -S -n -u root /bin/sh -c ""'echo BECOME-SUCCESS-wobjgnwxhegtszqxqfxxmgwqrwrrku; /usr/bin/python /home/ec2-user/.ansible/tmp/ansible-tmp-1657476989.36-192390981924916/yum.py ""' && sleep 0'

```

```

        runlevel: default,
        "sleep": null,
        "state": "restarted"
    },
    "name": "httpd",
    "state": "started"
}

RUNNING HANDLER [mysql : restart mysql] *****
<localhost> ESTABLISH LOCAL CONNECTION FOR USER: ec2-user
<localhost> EXEC /bin/sh -c 'echo ~ec2-user && sleep 0'
<localhost> EXEC /bin/sh -c '( umask 77 && mkdir -p "" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601 "" && echo ansible-tmp-1657477589.41-59139986714601="" echo /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601 "" ) && sleep 0'
Using module file /usr/lib/python2.6/site-packages/ansible/modules/system/service.py
<localhost> PUT /home/ec2-user/.ansible/tmp/ansible-local-8903tdwycz/tmpP3ivlp TO /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601/service.py
<localhost> EXEC /bin/sh -c 'chmod u+x /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601/ /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601/service.py && sleep 0'
<localhost> EXEC /bin/sh -c 'sudo -H -S -n -u root /bin/sh -c '""""echo BECOME-SUCCESS-vuxmodeccuynzgqzwumjgyqfwyibzrlq: /usr/bin/python /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601/service.py"""" && sleep 0'
<localhost> EXEC /bin/sh -c 'rm -f -r /home/ec2-user/.ansible/tmp/ansible-tmp-1657477589.41-59139986714601/ > /dev/null 2>&1 && sleep 0'
changed: [localhost] => {
    "changed": true,
    "invocation": {
        "module_args": {
            "arguments": "",
            "enabled": null,
            "name": "mysqld",
            "pattern": null,
            "runlevel": "default",
            "sleep": null,
            "state": "restarted"
        }
    },
    "name": "mysqld",
    "state": "started"
}
META: ran handlers
META: ran handlers


PLAY RECAP *****
localhost : ok=42  changed=40  unreachable=0  failed=0


[ec2-user@ip-172-31-6-137 ansible-magento2]$ |

```

○ Step-12: Testing the E-commerce website

Default welcome msg!
Sign In
or
Create an Account





Home Page

CMS homepage content goes here.

Search Terms
Privacy and Cookie Policy
Orders and Returns
Contact Us
Advanced Search

Step-13: Make the changes in Magento Product Catalog



Welcome, please sign in


* Username

* Password

[Forgot your password?](#)

Sign in

Copyright © 2022 Magento Commerce Inc. All rights reserved.



DASHBOARD

SALES

CATALOG

CUSTOMERS

MARKETING

CONTENT

REPORTS

STORES

SYSTEM

FIND PARTNERS & EXTENSIONS

Content

×

Elements

Pages

Blocks

Widgets

Design

Configuration

Themes

Schedule

on

Filters

Default View

Columns

20 per page 1 of 1

	Store	Store View	Action	Theme Name
			Edit	-- No Theme --
			Edit	-- No Theme --
	Main Website Store	Default Store View	Edit	-- No Theme --

Commerce Inc. All rights reserved.

Magento [Report](#)

Design Configuration

Filters

Default View

Columns

3 records found 20 per page 1 of 1

Default	Website	Store	Store View	Action	Theme Name
Global				Edit	-- No Theme --
Global	Main Website			Edit	-- No Theme --
Global	Main Website	Main Website Store	Default Store View	Edit	-- No Theme --


Copyright © 2022 Magento Commerce Inc. All rights reserved.

Magento ver. 2.3.0 [Report an Issue](#)

Default Store View

← BackSave and ContinueSave Configuration

Other Settings

HTML Head 

Favicon Icon

Upload

Not all browsers support all these formats!

Default Page Title

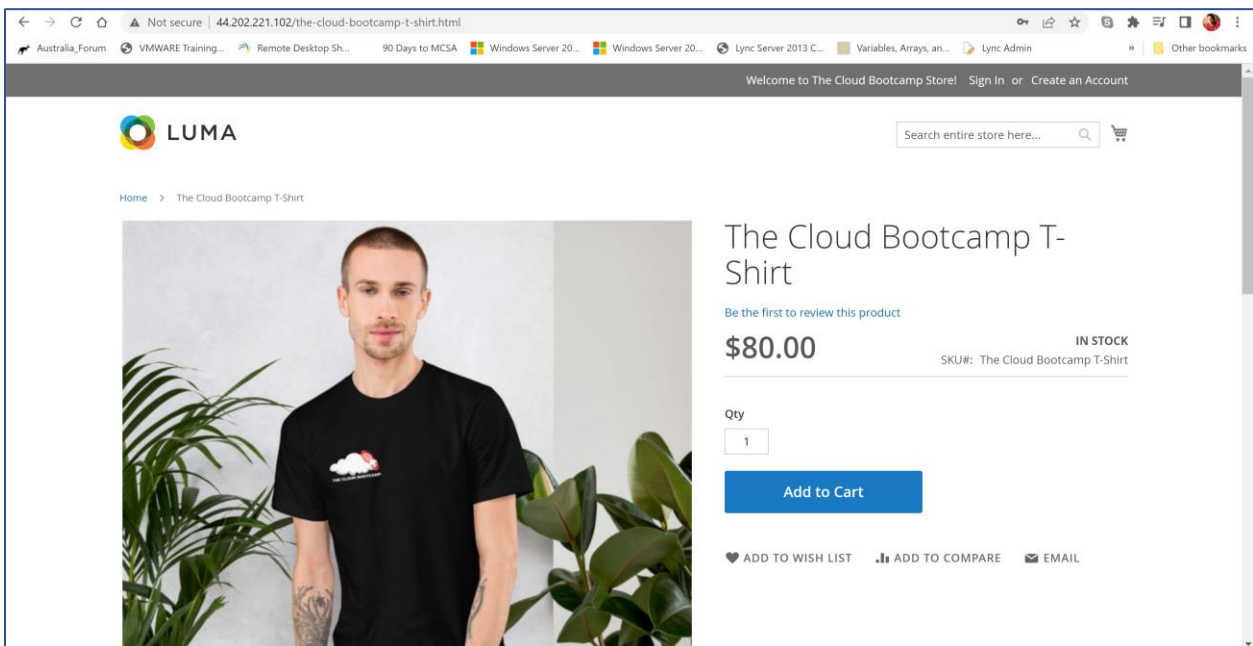
The Cloud Bootcamp Store

↔ Use Default Value

Page Title Prefix

Page Title Suffix

Default Meta Description



We can do more customization here:

-- Content > Configuration > HTML Head > Edit (Default Store View)

--- Default page title: The Clodu Bootcamp Store

--- Header > Logo image: The Cloud Bootcamp logo from images

--- Header > Welcome text: Welcome to The Cloud Bootcamp Store!

--- Cache Refresh (Flush it)

-- Catalog > Products > Add product > The Cloud Bootcamp T-Shirt

--- Price: 80

--- Quantity: 100

--- Images And Videos > Add images

--- Save

-- Content > Pages > Edit Home Pages

--- Click Content > Erase content

--- Insert Widget > Widget type: Catalog New Products List > Insert Widget