

Chipper on AWS and Terraform

- [Chipper on AWS and Terraform](#)
 - [References](#)
 - [Misc](#)
 - [AWS CLI Setup](#)
 - [Create Environment](#)
 - [Prep Terraform's Remote State in S3](#)
 - [Miscellaneous Commands](#)
 - [Find AMI ID for use with Terraform](#)
 - [Locate Recently Published Image for CentOS](#)

References

- <https://www.terraform.io/docs/providers/aws/index.html>

Misc

- Default Region: `us-east-2` .

AWS CLI Setup

- **Ubuntu/Linux** is assumed.

1. Install `aws-cli` snap package (They keep fairly old version in the `stable` channel):

```
sudo snap install aws-cli --classic --channel=edge
```

2. Install `aws-cli-authenticator` :

```
sudo sh -c 'curl -o /usr/local/bin/aws-iam-authenticator https://amazon-eks.s3-us-w
```



Create Environment

There are two separate **Terraform** folders used. *terraform-bootstrap-remote-state* is used to create the **S3** bucket/policies for storing **Terraform**'s remote state. Then, create the **Chipper** environment from `terraform` folder. This will provision the **VPC** and **EKS** cluster and **EC2** nodes. The `main.tf` files in each project need small edits due to variable interpolation limitations.

First, export environment variables:

```
export AWS_ACCESS_KEY_ID='ID_HERE' && \  
export AWS_SECRET_ACCESS_KEY='KEY_HERE' && \  
export AWS_DEFAULT_REGION='us-east-2'
```

Prep Terraform's Remote State in S3

Prep Terraform's Remote State in S3. Working directory: `./aws/terraform-bootstrap-remote-state/`

Create VPC / EKS Cluster. Working directory: `./aws/terraform/`

Run these commands in the two directories above, chronologically:

1. Update modules and create a plan:

```
terraform init && terraform plan -out="planfile" -detailed-exitcode
```

2. Execute the plan to build:

```
terraform apply "planfile"
```

3. Destroy what was built (Be warned!):

```
terraform init && terraform plan -destroy -out="planfile" -detailed-exitcode  
terraform apply "planfile"
```

Miscellaneous Commands

Find AMI ID for use with Terraform

- [Link](#)

Locate Recently Published Image for CentOS

Note: the region is specified in the command.

```
aws ec2 describe-images --owners aws-marketplace --filters Name=product-code,Values=
```

