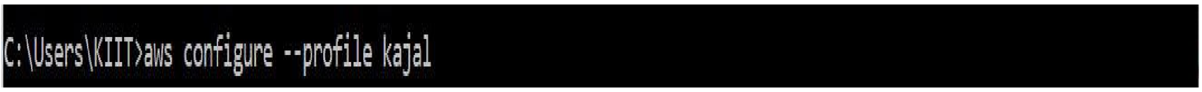
AWS AUTOMATION USING TERRAFORM:

Task: Have to create/launch Application using Terraform

1. Create the key and security group which allow the port 80.
2. Launch EC2 instance.
3. In this Ec2 instance use the key and security group which we have created in step 1.
4. Launch one Volume (EBS) and mount that volume into /var/www/html
5. Developer have uploded the code into github repo also the repo has some images.
6. Copy the github repo code into /var/www/html
7. Create S3 bucket, and copy/deploy the images from github repo into the s3 bucket and change the permission to public readable.

8 Create a Cloudfront using s3 bucket(which contains images) and use the Cloudfront URL to update in code in /var/www/html



```
C:\Users\KIIT>aws configure --profile kajal
```

```
provider "aws" {
     region = "ap-south-1"
     profile = "kajal"
}
data "aws_vpc" "selected" {
   default = true
}
```

/* Generate a private key and encode it as pem */

```
resource "tls_private_key" "example" {
   algorithm      = "RSA"
```

```
}

resource "local_file" "private-key" {
        content         = "tls_private_key.key.private_key_pem"
        filename = "mykey.pem"
        file_permission = 0400
}

resource "aws_key_pair" "key-pair" {
    key_name      = "mykey"
    public_key = tls_private_key.example.public_key_openssh
}
/* Create security group and allow HTTP, SSH and ICMP
protocols */
resource "aws_security_group" "allow_tls" {
    name                = "allow_tls1"
    description = "Allow_tls"
    vpc_id            = "${data.aws_vpc.selected.id}"

    ingress {
        description = "http"
        from_port      = 80
        to_port          = 80
        protocol        = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }
  ingress {
        description = "ssh"
        from_port      = 22
        to_port          = 22
        protocol        = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }
  ingress {
        description = "ping-icmp"
    from_port    = -1
    to_port      = -1
    protocol     = "icmp"
    cidr_blocks = ["0.0.0.0/0"]
  }
  egress {
    from_port    = 0
    to_port      = 0
    protocol     = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "allow_tls1"
  }
}
/* Launch an OS, attach the key generated and the security
groups created. SSH into the OS and install the webserver and
the required SDK's    */
resource "aws_instance" "web" {
   ami = "ami-0447a12f28fddb066"
   instance_type = "t2.micro"
   key_name = aws_key_pair.key-pair.key_name
   security_groups = [ aws_security_group.allow_tls.name ]

connection {
   type       = "ssh"
      user       = "ec2-user"
   private_key = tls_private_key.example.private_key_pem
      host       = aws_instance.web.public_ip
  }
provisioner "remote-exec" {
   inline = [
   "sudo yum install httpd php git -y",
   "sudo systemctl restart httpd",
     "sudo systemctl enable httpd",
     ]
}
tags = {
     Name = "lwos1"
     }
}
/* Fetch the availability zone of the instance and create an
EBS volume in the same zone. */
resource "aws_ebs_volume" "lw_ebs" {
   availability_zone = aws_instance.web.availability_zone
   size                          = 1
   tags = {
      Name = "lw_ebs"
   }
}

/* Attach the volume created to the instance */
resource "aws_volume_attachment" "ebs_att" {

   device_name = "/dev/sdh"
   volume_id      = "${aws_ebs_volume.lw_ebs.id}"
   instance_id = "${aws_instance.web.id}"
   force_detach = true
}
output "myos_ip" {
   value = aws_instance.web.public_ip
}

/* [Optional]*/
resource "null_resource" "nullocal2" {
     provisioner "local-exec" {
     command = "echo ${aws_instance.web.public_ip} >
publicip.txt"
     }
```

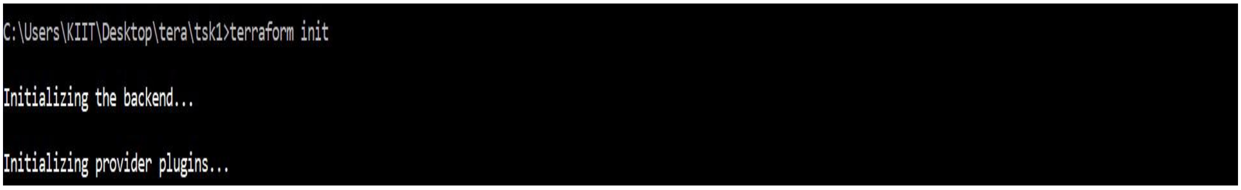
```
}
/* Provisioners are used to model specific actions on the local
machine or on a remote machine in order to prepare servers
or other infrastructure objects for service.*/
/* Provisioners need a null-resource that is a do-nothing
container for the actions taken by a provisioner. */

resource "null_resource" "nullremote3" {
/* Tells Terraform that EBS volume must be formatted,
mounted and store data only after the created volume has
been attched to the instance. */
depends_on = [
    aws_volume_attachment.ebs_att,
    ]

connection {
    type         = "ssh"
        user         = "ec2-user"
    private_key = tls_private_key.example.private_key_pem
        host         = aws_instance.web.public_ip
   }
provisioner "remote-exec" {
    inline = [
    "sudo mkfs.ext4 /dev/xvdh",
    "sudo mount /dev/xvdh /var/www/html",
    "sudo rm -rf /var/www/html/*",
    "sudo git clone
https://github.com/kajal1706043/multi_cloudTask1.git
/var/www/html"
    ]
    }
}
```

```
/*Tells Terraform that the IP address must be viewed onto the
browser only after the instance is completely ready with the
webpage. */
resource "null_resource" "nulllocal1" {

depends_on = [
    null_resource.nullremote3,
]

    provisioner "local-exec" {
        command = "start chrome
${aws_instance.web.public_ip}"
    }
}
/*Create an S3 bucket and grant public access to it */
resource "aws_s3_bucket" "b" {
   bucket = "tsk1bucket"
   acl       = "public-read"

   tags = {
      Name               = "mybucket"
   }
}

/* Deploy an image into the bucket from Github. */
resource "aws_s3_bucket_object" "deployimage" {
    bucket = aws_s3_bucket.b.bucket
    key = "cloudtask1.jpg"
    source = "git_image/Hybrid-Cloud.jpg"
    acl = "public-read"
}
```

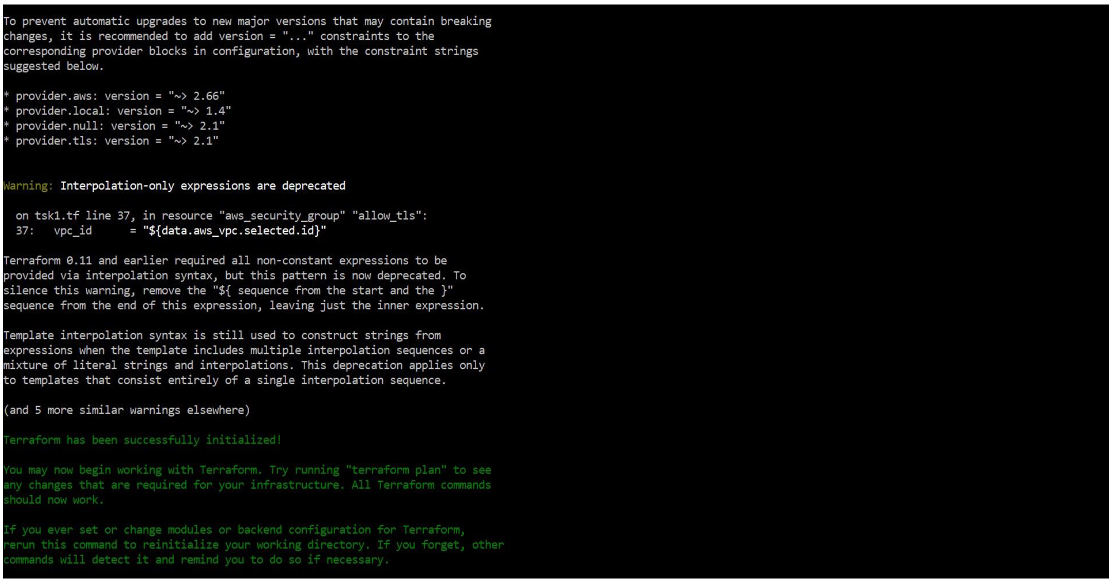
```
/* null-resources are the first to be executed by Terraform.
Thus, the image on github is first download onto the local
machine*/
resource "null_resource" "nulllocal4" {
provisioner "local-exec" {
     command = "git clone
https://github.com/kajal1706043/task1_s3.git git_image"
}
/* To remove the image from the local system when the
infrastructure is destroyed */
provisioner "local-exec" {
     when = destroy
     command = "rmdir /s /q git_image"
}
}
resource "aws_cloudfront_origin_access_identity"
"origin_access_identity" {
    comment = "Some comment"
}

#Create a CloudFront Distribution with the created S3 bucket
as Origin
locals {
    s3_origin_id = "myS3Origin"
    image-url =
"${aws_cloudfront_distribution.s3_distribution.domain_name}
/${aws_s3_bucket_object.deployimage.key}"
}

resource "aws_cloudfront_distribution" "s3_distribution" {
    origin {
        domain_name = aws_s3_bucket.b.bucket_domain_name
        origin_id      = "${local.s3_origin_id}"

      s3_origin_config {
            origin_access_identity =
"${aws_cloudfront_origin_access_identity.origin_access_identi
ty.cloudfront_access_identity_path}"
        }
}

default_cache_behavior {
        allowed_methods    = ["DELETE", "GET", "HEAD",
"OPTIONS", "PATCH", "POST", "PUT"]
        cached_methods      = ["GET", "HEAD"]
        target_origin_id = "${local.s3_origin_id}"

        forwarded_values {
            query_string = false

            cookies {
                forward = "none"
            }
        }
viewer_protocol_policy = "allow-all"
}
enabled                          = true
restrictions {
          geo_restriction {
                restriction_type = "none"
        }
}
viewer_certificate {
        cloudfront_default_certificate = true
    }
}
```

Commands:

- terraform init //to download the necessary plugins.

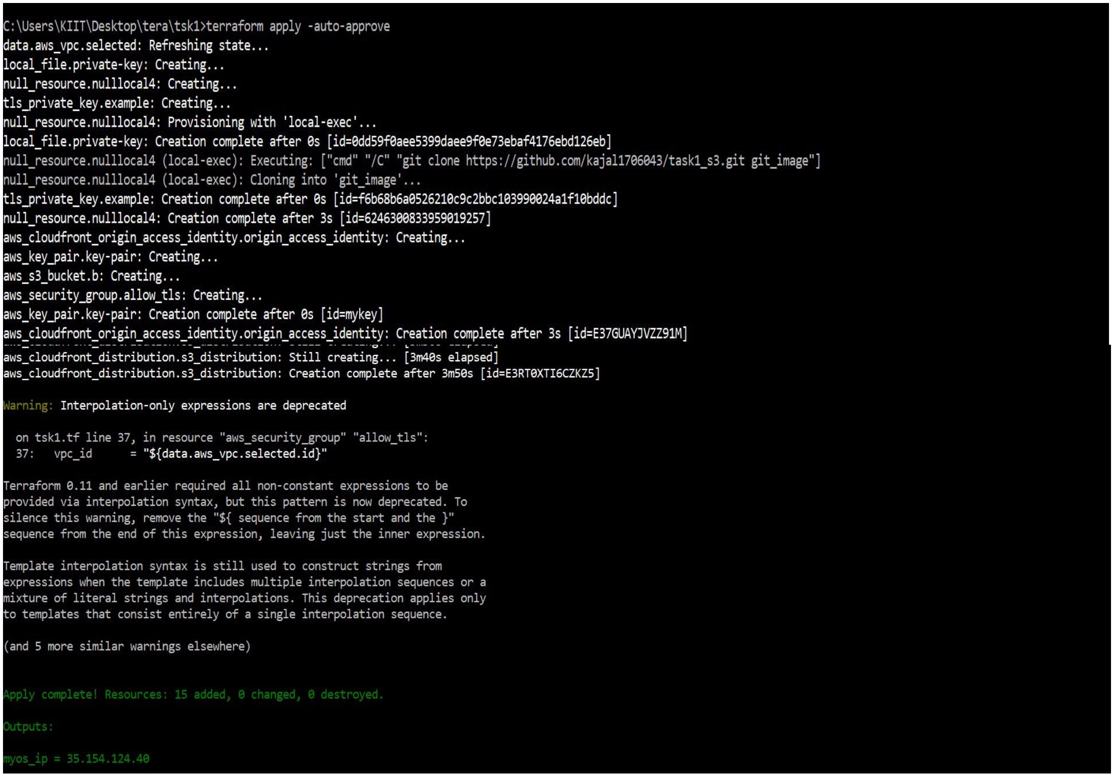


```
C:\Users\KIIT\Desktop\tera\tsk1>terraform init

Initializing the backend...

Initializing provider plugins...
```



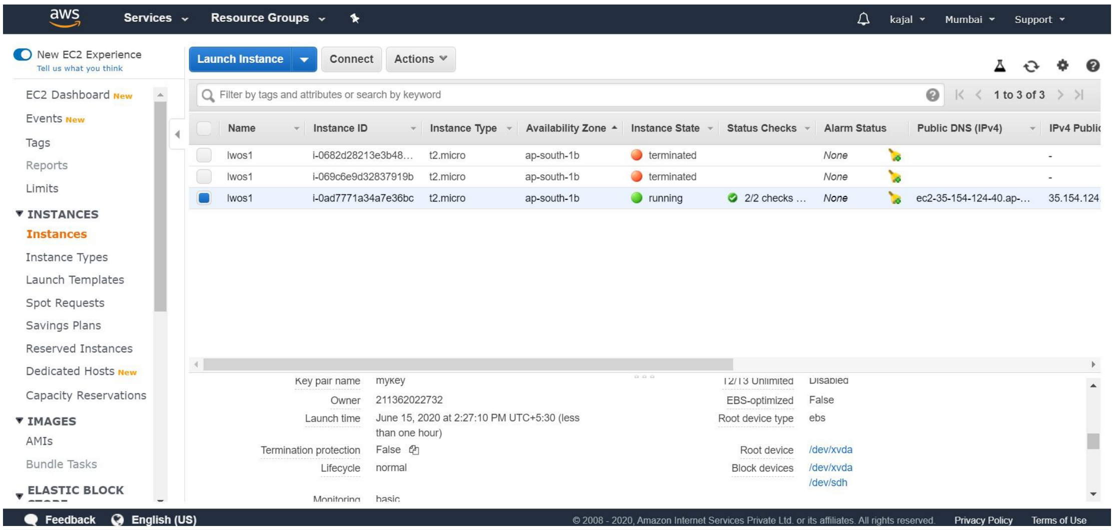
```
To prevent automatic upgrades to new major versions that may contain breaking
changes, it is recommended to add version = "..." constraints to the
corresponding provider blocks in configuration, with the constraint strings
suggested below.

* provider.aws: version = "~> 2.66"
* provider.local: version = "~> 1.4"
* provider.null: version = "~> 2.1"
* provider.tls: version = "~> 2.1"


Warning: Interpolation-only expressions are deprecated

  on tsk1.tf line 37, in resource "aws_security_group" "allow_tls":
  37:   vpc_id      = "${data.aws_vpc.selected.id}"

Terraform 0.11 and earlier required all non-constant expressions to be
provided via interpolation syntax, but this pattern is now deprecated. To
silence this warning, remove the "${ sequence from the start and the }"
sequence from the end of this expression, leaving just the inner expression.

Template interpolation syntax is still used to construct strings from
expressions when the template includes multiple interpolation sequences or a
mixture of literal strings and interpolations. This deprecation applies only
to templates that consist entirely of a single interpolation sequence.

(and 5 more similar warnings elsewhere)

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

- Terraform apply -auto-approve //to build the infrastructure.



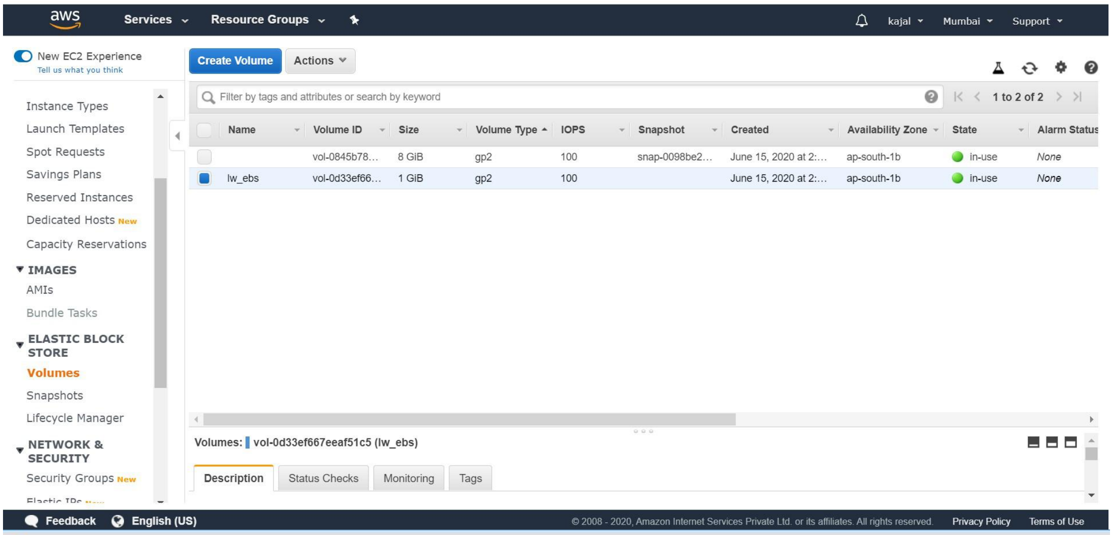
```
C:\Users\KIIT\Desktop\tera\tsk1>terraform apply -auto-approve
data.aws_vpc.selected: Refreshing state...
local_file.private-key: Creating...
null_resource.nulllocal4: Creating...
tls_private_key.example: Creating...
null_resource.nulllocal4: Provisioning with 'local-exec'...
local_file.private-key: Creation complete after 0s [id=0dd59f0aee5399daee9f0e73ebaf4176ebd126eb]
null_resource.nulllocal4 (local-exec): Executing: ["cmd" "/C" "git clone https://github.com/kajal1706043/task1_s3.git git_image"]
null_resource.nulllocal4 (local-exec): Cloning into 'git_image'...
tls_private_key.example: Creation complete after 0s [id=f6b68b6a0526210c9c2bbc103990024a1f10bddc]
null_resource.nulllocal4: Creation complete after 3s [id=6246300833959019257]
aws_cloudfront_origin_access_identity.origin_access_identity: Creating...
aws_key_pair.key-pair: Creating...
aws_s3_bucket.b: Creating...
aws_security_group.allow_tls: Creating...
aws_key_pair.key-pair: Creation complete after 0s [id=mykey]
aws_cloudfront_origin_access_identity.origin_access_identity: Creation complete after 3s [id=E37GUAYJVZZ91M]
aws_cloudfront_distribution.s3_distribution: Still creating... [3m40s elapsed]
aws_cloudfront_distribution.s3_distribution: Creation complete after 3m50s [id=E3RT0XTI6CZKZ5]

Warning: Interpolation-only expressions are deprecated

  on tsk1.tf line 37, in resource "aws_security_group" "allow_tls":
  37:   vpc_id      = "${data.aws_vpc.selected.id}"

Terraform 0.11 and earlier required all non-constant expressions to be
provided via interpolation syntax, but this pattern is now deprecated. To
silence this warning, remove the "${ sequence from the start and the }"
sequence from the end of this expression, leaving just the inner expression.

Template interpolation syntax is still used to construct strings from
expressions when the template includes multiple interpolation sequences or a
mixture of literal strings and interpolations. This deprecation applies only
to templates that consist entirely of a single interpolation sequence.

(and 5 more similar warnings elsewhere)


Apply complete! Resources: 15 added, 0 changed, 0 destroyed.

Outputs:

myos_ip = 35.154.124.40
```

The infrastructure is built:



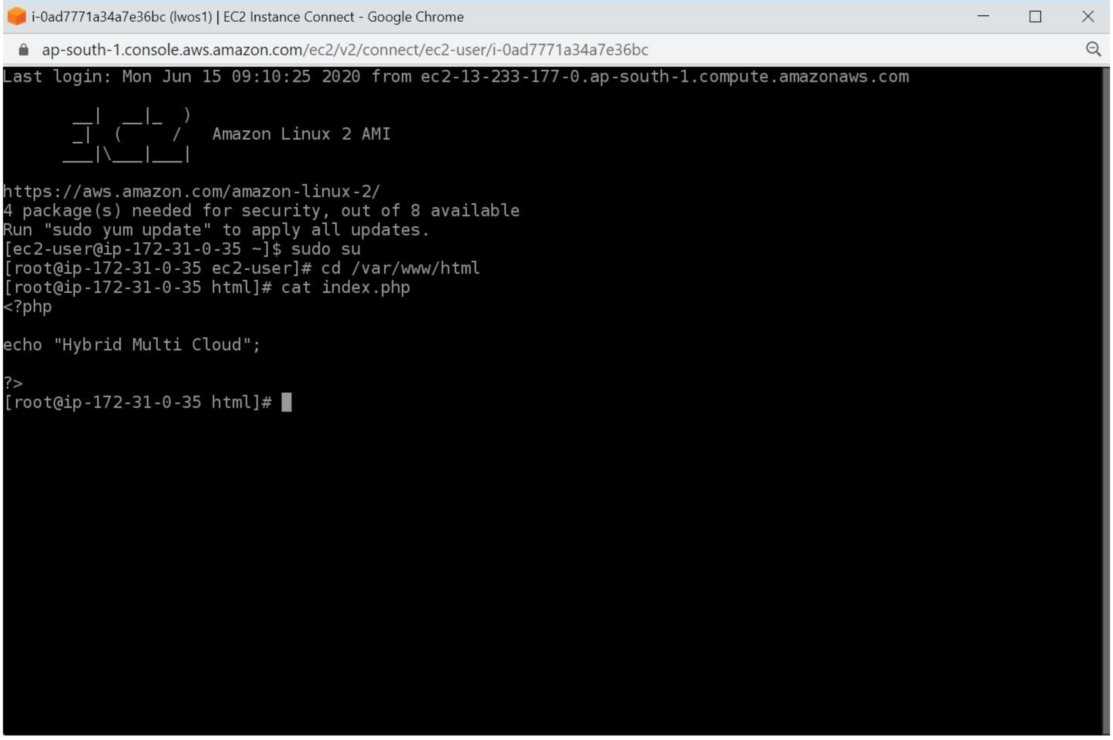
	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public
□	lwos1	i-0682d28213e3b48…	t2.micro	ap-south-1b	● terminated		None		-
□	lwos1	i-069c6e9d32837919b	t2.micro	ap-south-1b	● terminated		None		-
■	lwos1	i-0ad7771a34a7e36bc	t2.micro	ap-south-1b	● running	✓ 2/2 checks …	None	ec2-35-154-124-40.ap-…	35.154.124

Key pair name	mykey	T2/T3 Unlimited	Disabled
Owner	211362022732	EBS-optimized	False
Launch time	June 15, 2020 at 2:27:10 PM UTC+5:30 (less than one hour)	Root device type	ebs
Termination protection	False	Root device	/dev/xvda
Lifecycle	normal	Block devices	/dev/xvda /dev/sdh
Monitoring	basic		

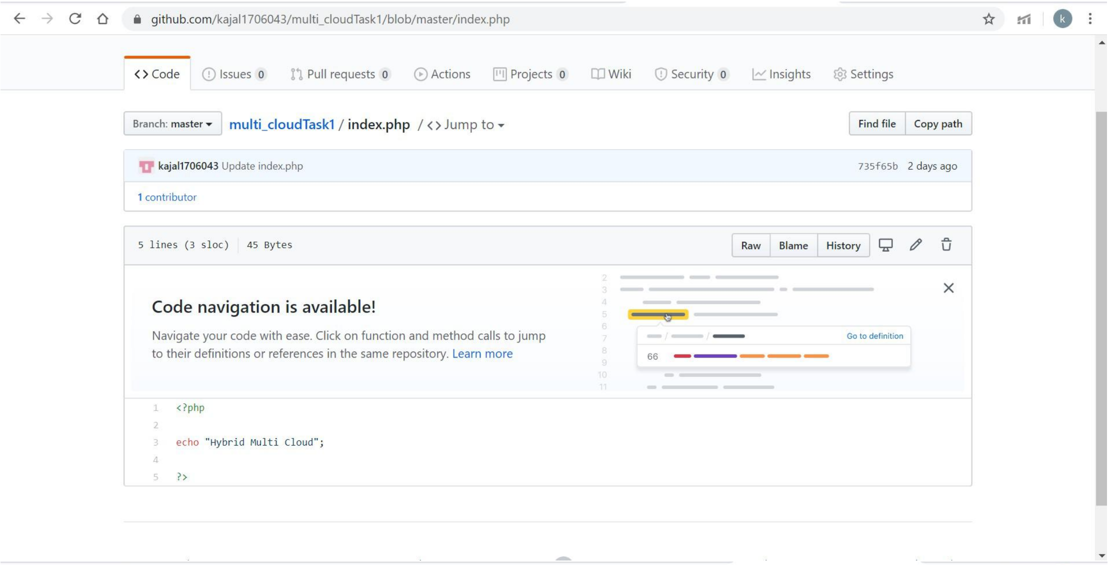


	Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created	Availability Zone	State	Alarm Status
□		vol-0845b78…	8 GiB	gp2	100	snap-0098be2…	June 15, 2020 at 2:…	ap-south-1b	● in-use	None
■	lw_ebs	vol-0d33ef66…	1 GiB	gp2	100		June 15, 2020 at 2:…	ap-south-1b	● in-use	None

Volumes: vol-0d33ef667eeaf51c5 (lw_ebs)


i-0ad7771a34a7e36bc (lwos1) | EC2 Instance Connect - Google Chrome
ap-south-1.console.aws.amazon.com/ec2/v2/connect/ec2-user/i-0ad7771a34a7e36bc

```
Last login: Mon Jun 15 09:10:25 2020 from ec2-13-233-177-0.ap-south-1.compute.amazonaws.com

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 8 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-0-35 ~]$ sudo su
[root@ip-172-31-0-35 ec2-user]# cd /var/www/html
[root@ip-172-31-0-35 html]# cat index.php
<?php

echo "Hybrid Multi Cloud";

?>
[root@ip-172-31-0-35 html]# 
```



github.com/kajal1706043/multi_cloudTask1/blob/master/index.php

multi_cloudTask1 / index.php / <> Jump to

kajal1706043 Update index.php 735f65b 2 days ago

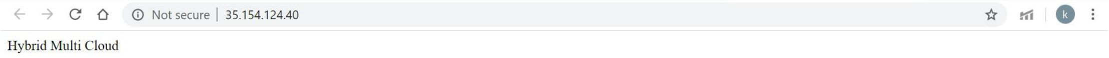
1 contributor

5 lines (3 sloc) | 45 Bytes

Code navigation is available!

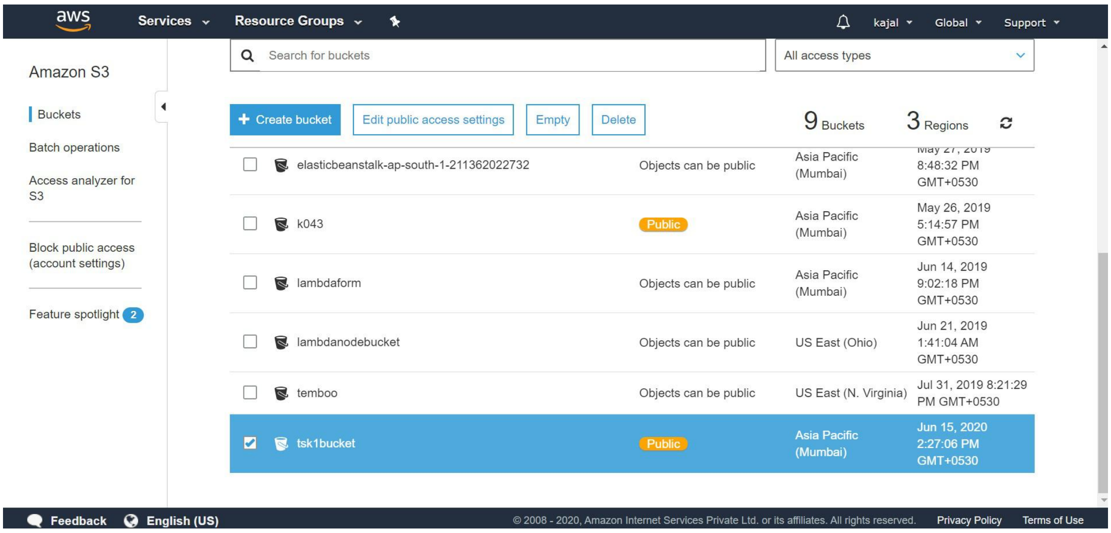
Navigate your code with ease. Click on function and method calls to jump to their definitions or references in the same repository. Learn more

```php
<?php

echo "Hybrid Multi Cloud";

?>
```



Not secure | 35.154.124.40

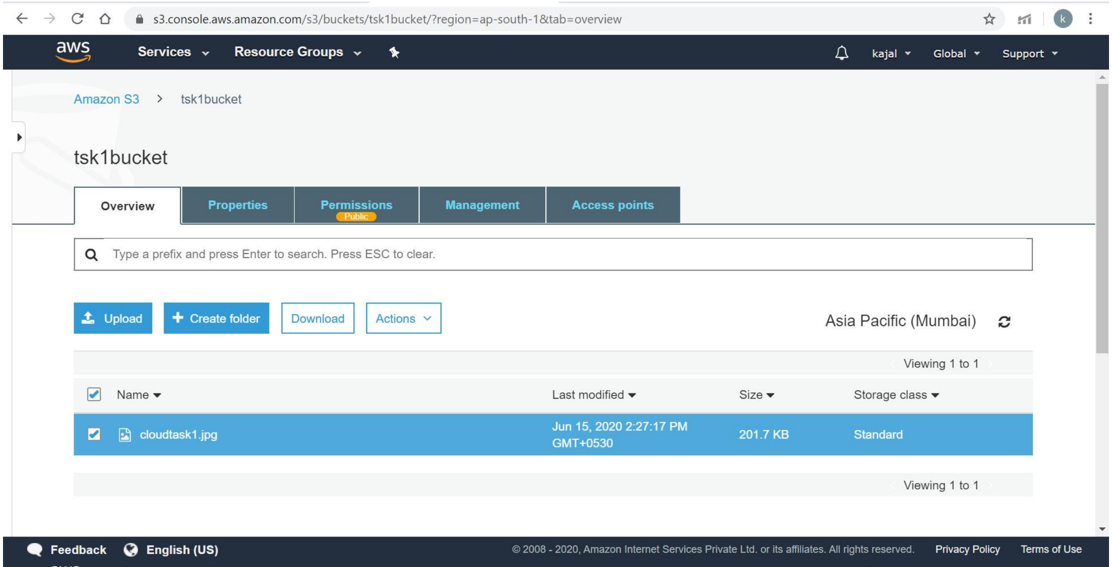
Hybrid Multi Cloud



9 Buckets 3 Regions

□	elasticbeanstalk-ap-south-1-211362022732	Objects can be public	Asia Pacific (Mumbai)	May 27, 2019 8:48:32 PM GMT+0530
□	k043	Public	Asia Pacific (Mumbai)	May 26, 2019 5:14:57 PM GMT+0530
□	lambdaform	Objects can be public	Asia Pacific (Mumbai)	Jun 14, 2019 9:02:18 PM GMT+0530
□	lambdanodebucket	Objects can be public	US East (Ohio)	Jun 21, 2019 1:41:04 AM GMT+0530
□	temboo	Objects can be public	US East (N. Virginia)	Jul 31, 2019 8:21:29 PM GMT+0530
☑	tsk1bucket	Public	Asia Pacific (Mumbai)	Jun 15, 2020 2:27:06 PM GMT+0530

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.



s3.console.aws.amazon.com/s3/buckets/tsk1bucket/?region=ap-south-1&tab=overview

Amazon S3 > tsk1bucket

tsk1bucket

Type a prefix and press Enter to search. Press ESC to clear.

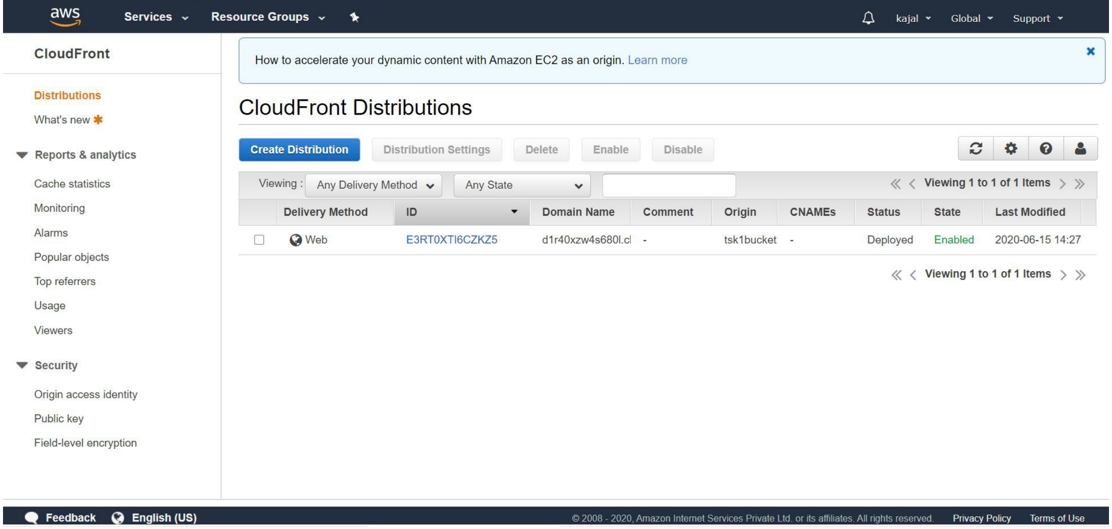
Asia Pacific (Mumbai)

Viewing 1 to 1

☑	Name	Last modified	Size	Storage class
☑	cloudtask1.jpg	Jun 15, 2020 2:27:17 PM GMT+0530	201.7 KB	Standard

Viewing 1 to 1

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use



How to accelerate your dynamic content with Amazon EC2 as an origin. Learn more

CloudFront Distributions

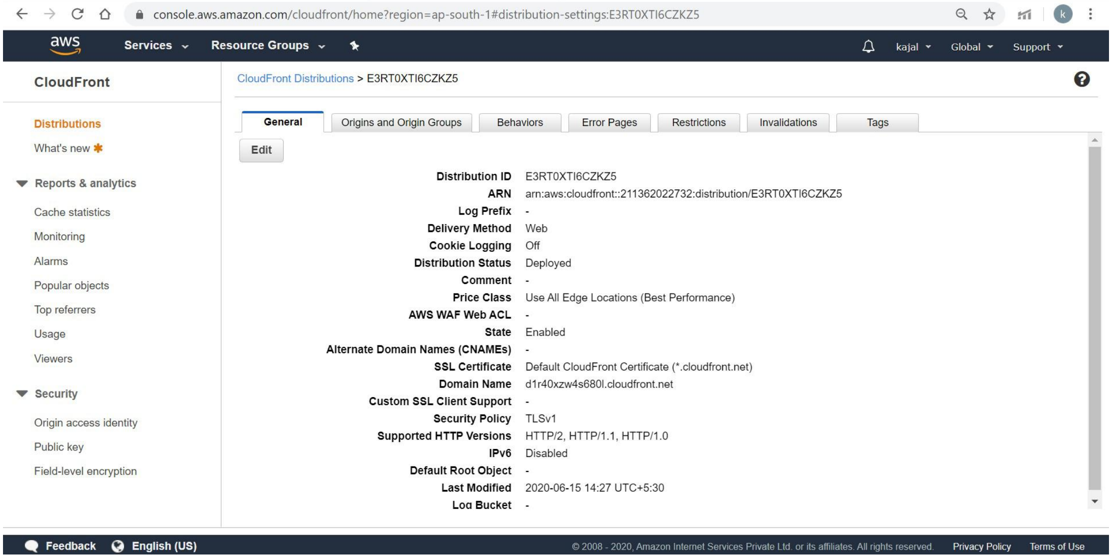
Viewing : Any Delivery Method Any State

Viewing 1 to 1 of 1 Items

	Delivery Method	ID	Domain Name	Comment	Origin	CNAMEs	Status	State	Last Modified
□	Web	E3RT0XTI6CZKZ5	d1r40xzw4s680l.cl	-	tsk1bucket	-	Deployed	Enabled	2020-06-15 14:27

Viewing 1 to 1 of 1 Items

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

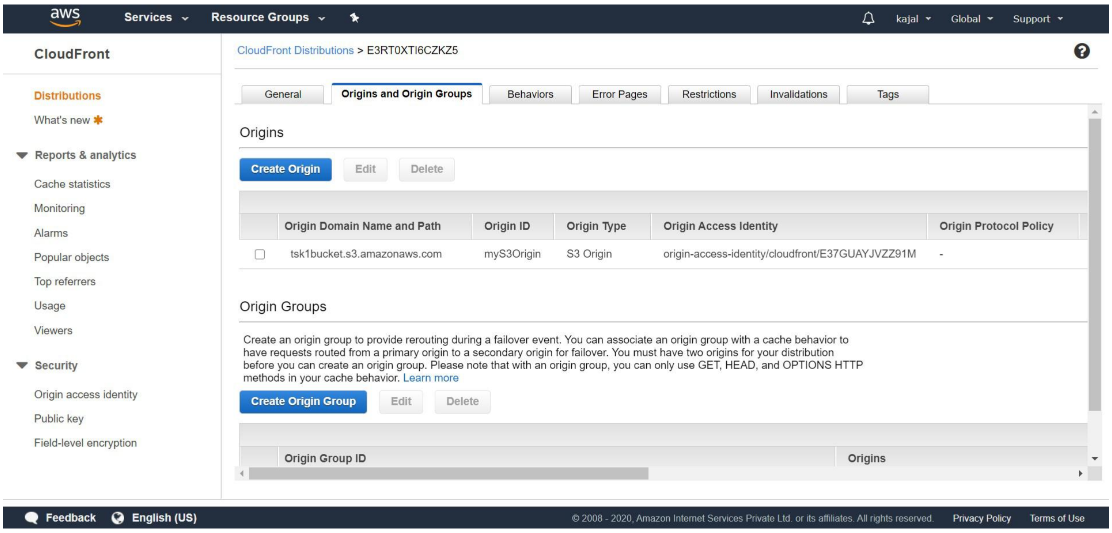


console.aws.amazon.com/cloudfront/home?region=ap-south-1#distribution-settings:E3RT0XTI6CZKZ5

CloudFront Distributions > E3RT0XTI6CZKZ5

Distribution ID	E3RT0XTI6CZKZ5
ARN	arn:aws:cloudfront::211362022732:distribution/E3RT0XTI6CZKZ5
Log Prefix	-
Delivery Method	Web
Cookie Logging	Off
Distribution Status	Deployed
Comment	-
Price Class	Use All Edge Locations (Best Performance)
AWS WAF Web ACL	-
State	Enabled
Alternate Domain Names (CNAMEs)	-
SSL Certificate	Default CloudFront Certificate (*.cloudfront.net)
Domain Name	d1r40xzw4s680l.cloudfront.net
Custom SSL Client Support	-
Security Policy	TLSv1
Supported HTTP Versions	HTTP/2, HTTP/1.1, HTTP/1.0
IPv6	Disabled
Default Root Object	-
Last Modified	2020-06-15 14:27 UTC+5:30
Log Bucket	-

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use



CloudFront Distributions > E3RT0XTI6CZKZ5

Origins

	Origin Domain Name and Path	Origin ID	Origin Type	Origin Access Identity	Origin Protocol Policy
□	tsk1bucket.s3.amazonaws.com	myS3Origin	S3 Origin	origin-access-identity/cloudfront/E37GUAYJVZZ91M	-

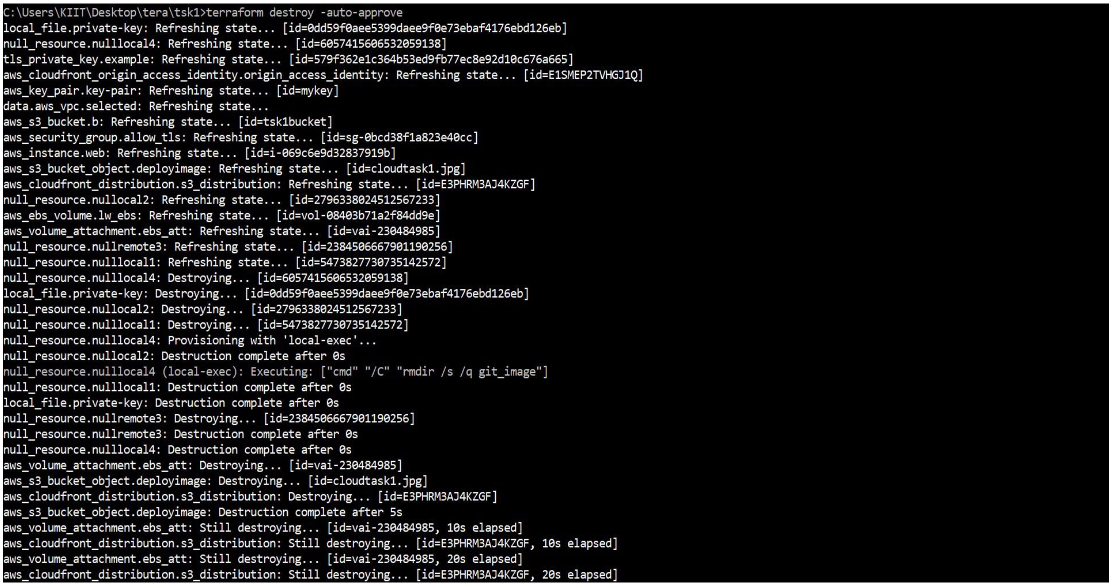
Origin Groups

Create an origin group to provide rerouting during a failover event. You can associate an origin group with a cache behavior to have requests routed from a primary origin to a secondary origin for failover. You must have two origins for your distribution before you can create an origin group. Please note that with an origin group, you can only use GET, HEAD, and OPTIONS HTTP methods in your cache behavior. Learn more

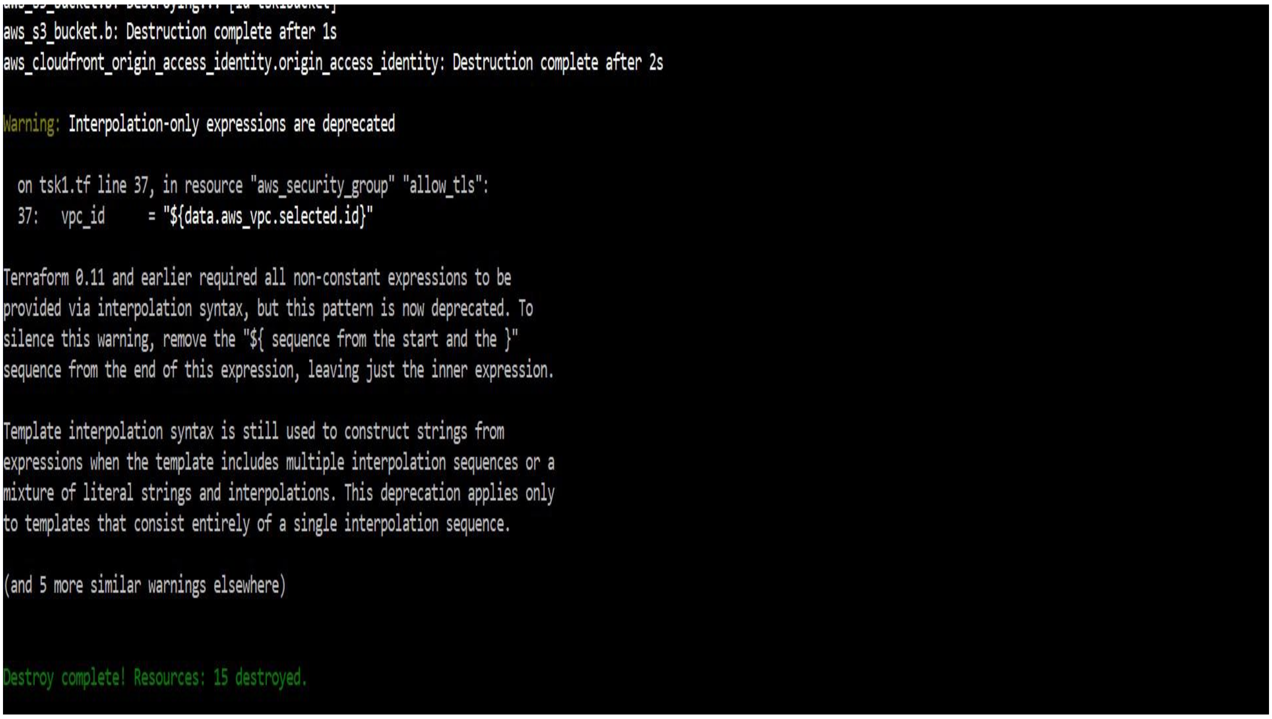
Origin Group ID	Origins

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

- To destroy the infrastructure:



```
C:\Users\KIIT\Desktop\tera\tsk1>terraform destroy -auto-approve
local_file.private-key: Refreshing state... [id=0dd59f0aee5399daee9f0e73ebaf4176ebd126eb]
null_resource.nulllocal4: Refreshing state... [id=6057415606532059138]
tls_private_key.example: Refreshing state... [id=579f362e1c364b53ed9fb77ec8e92d10c676a665]
aws_cloudfront_origin_access_identity.origin_access_identity: Refreshing state... [id=E1SMEP2TVHGJ1Q]
aws_key_pair.key-pair: Refreshing state... [id=mykey]
data.aws_vpc.selected: Refreshing state...
aws_s3_bucket.b: Refreshing state... [id=tsk1bucket]
aws_security_group.allow_tls: Refreshing state... [id=sg-0bcd38f1a823e40cc]
aws_instance.web: Refreshing state... [id=i-069c6e9d32837919b]
aws_s3_bucket_object.deployimage: Refreshing state... [id=cloudtask1.jpg]
aws_cloudfront_distribution.s3_distribution: Refreshing state... [id=E3PHRM3AJ4KZGF]
null_resource.nullocal2: Refreshing state... [id=2796338024512567233]
aws_ebs_volume.lw_ebs: Refreshing state... [id=vol-08403b71a2f84dd9e]
aws_volume_attachment.ebs_att: Refreshing state... [id=vai-230484985]
null_resource.nullremote3: Refreshing state... [id=2384506667901190256]
null_resource.nulllocal1: Refreshing state... [id=5473827730735142572]
null_resource.nulllocal4: Destroying... [id=6057415606532059138]
local_file.private-key: Destroying... [id=0dd59f0aee5399daee9f0e73ebaf4176ebd126eb]
null_resource.nullocal2: Destroying... [id=2796338024512567233]
null_resource.nulllocal1: Destroying... [id=5473827730735142572]
null_resource.nulllocal4: Provisioning with 'local-exec'...
null_resource.nullocal2: Destruction complete after 0s
null_resource.nulllocal4 (local-exec): Executing: ["cmd" "/C" "rmdir /s /q git_image"]
null_resource.nulllocal1: Destruction complete after 0s
local_file.private-key: Destruction complete after 0s
null_resource.nullremote3: Destroying... [id=2384506667901190256]
null_resource.nullremote3: Destruction complete after 0s
null_resource.nulllocal4: Destruction complete after 0s
aws_volume_attachment.ebs_att: Destroying... [id=vai-230484985]
aws_s3_bucket_object.deployimage: Destroying... [id=cloudtask1.jpg]
aws_cloudfront_distribution.s3_distribution: Destroying... [id=E3PHRM3AJ4KZGF]
aws_s3_bucket_object.deployimage: Destruction complete after 5s
aws_volume_attachment.ebs_att: Still destroying... [id=vai-230484985, 10s elapsed]
aws_cloudfront_distribution.s3_distribution: Still destroying... [id=E3PHRM3AJ4KZGF, 10s elapsed]
aws_volume_attachment.ebs_att: Still destroying... [id=vai-230484985, 20s elapsed]
aws_cloudfront_distribution.s3_distribution: Still destroying... [id=E3PHRM3AJ4KZGF, 20s elapsed]
```



```
aws_s3_bucket.b: Destroying... [id=tsk1bucket]
aws_s3_bucket.b: Destruction complete after 1s
aws_cloudfront_origin_access_identity.origin_access_identity: Destruction complete after 2s

Warning: Interpolation-only expressions are deprecated

  on tsk1.tf line 37, in resource "aws_security_group" "allow_tls":
  37:   vpc_id      = "${data.aws_vpc.selected.id}"

Terraform 0.11 and earlier required all non-constant expressions to be
provided via interpolation syntax, but this pattern is now deprecated. To
silence this warning, remove the "${ sequence from the start and the }"
sequence from the end of this expression, leaving just the inner expression.

Template interpolation syntax is still used to construct strings from
expressions when the template includes multiple interpolation sequences or a
mixture of literal strings and interpolations. This deprecation applies only
to templates that consist entirely of a single interpolation sequence.

(and 5 more similar warnings elsewhere)


Destroy complete! Resources: 15 destroyed.
```