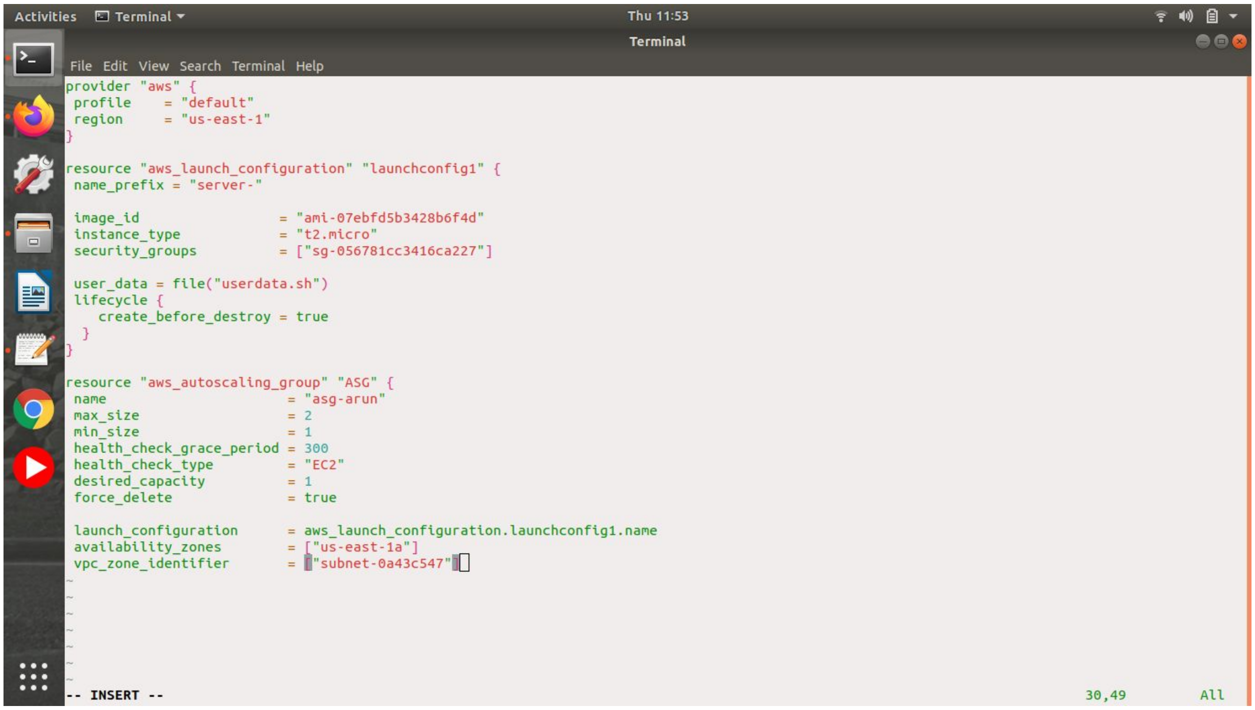
Assessment - 24

TERRAFORM-2

Trainee Name : Arun Parmar
Mentor Name : Ravi Kumar
College: UPES

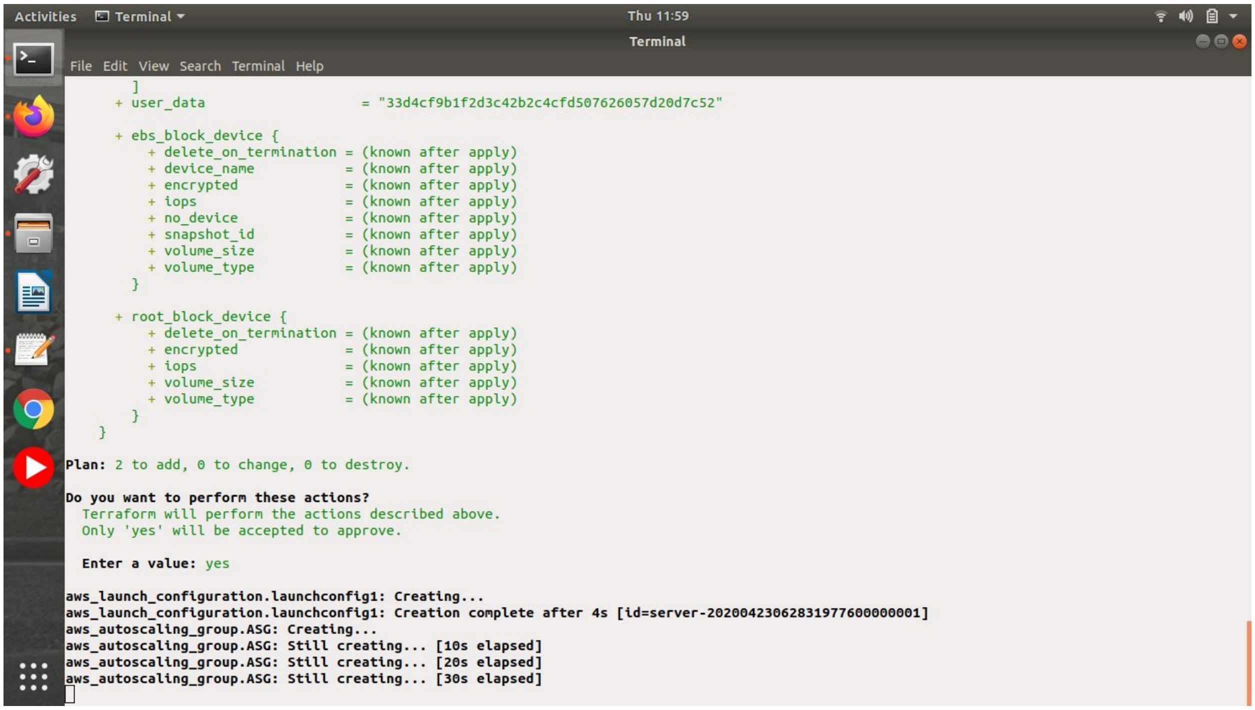
1. **Launch an ASG in AWS and do Rolling Deployment with change in User Data in LaunchConfig using terraform.**

- Create a main.tf file and a userdata.sh file with simple #!/bin/bash



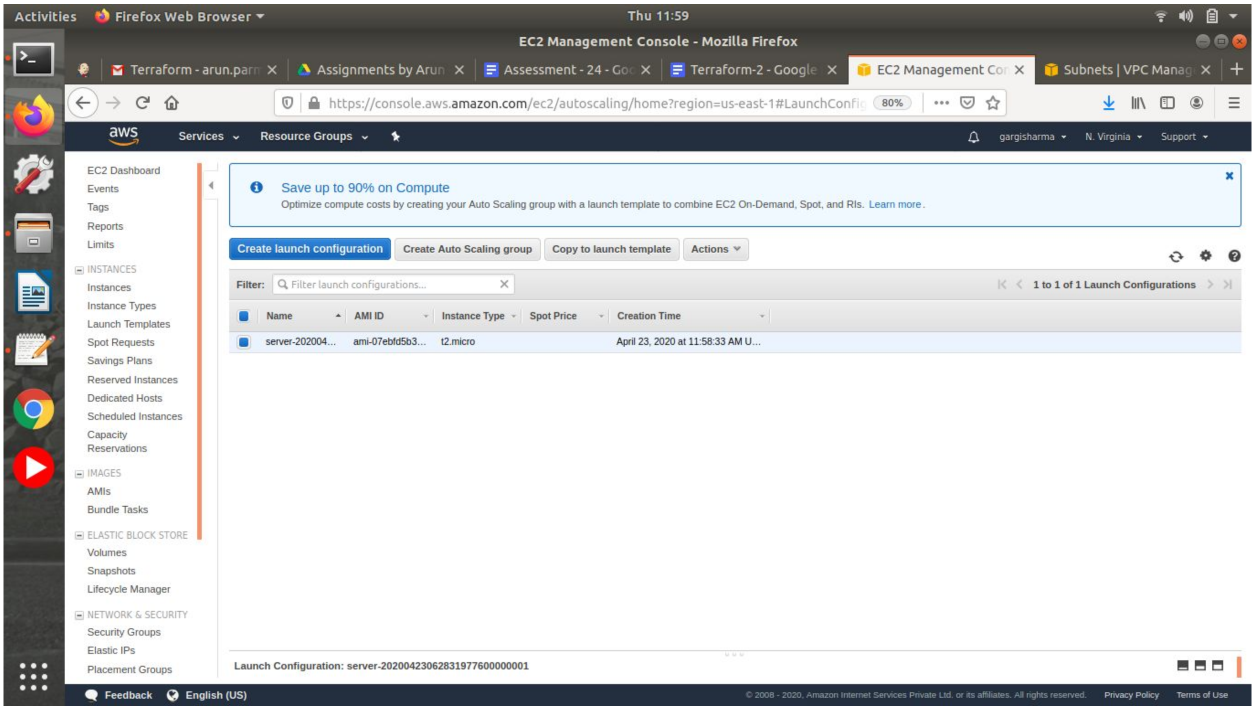
```
provider "aws" {
 profile    = "default"
 region     = "us-east-1"
}

resource "aws_launch_configuration" "launchconfig1" {
 name_prefix = "server-"

 image_id                   = "ami-07ebfd5b3428b6f4d"
 instance_type              = "t2.micro"
 security_groups            = ["sg-056781cc3416ca227"]

 user_data = file("userdata.sh")
 lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "ASG" {
 name                       = "asg-arun"
 max_size                   = 2
 min_size                   = 1
 health_check_grace_period  = 300
 health_check_type          = "EC2"
 desired_capacity           = 1
 force_delete               = true

 launch_configuration       = aws_launch_configuration.launchconfig1.name
 availability_zones         = ["us-east-1a"]
 vpc_zone_identifier        = ["subnet-0a43c547"]
~
~
~
~
~
~
~
-- INSERT --                                                                     30,49         All
```

- Terraform init > Terraform apply



```
        ]
      + user_data                   = "33d4cf9b1f2d3c42b2c4cfd507626057d20d7c52"

      + ebs_block_device {
          + delete_on_termination = (known after apply)
          + device_name           = (known after apply)
          + encrypted             = (known after apply)
          + iops                  = (known after apply)
          + no_device             = (known after apply)
          + snapshot_id           = (known after apply)
          + volume_size           = (known after apply)
          + volume_type           = (known after apply)
        }

      + root_block_device {
          + delete_on_termination = (known after apply)
          + encrypted             = (known after apply)
          + iops                  = (known after apply)
          + volume_size           = (known after apply)
          + volume_type           = (known after apply)
        }
    }

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_launch_configuration.launchconfig1: Creating...
aws_launch_configuration.launchconfig1: Creation complete after 4s [id=server-20200423062831977600000001]
aws_autoscaling_group.ASG: Creating...
aws_autoscaling_group.ASG: Still creating... [10s elapsed]
aws_autoscaling_group.ASG: Still creating... [20s elapsed]
aws_autoscaling_group.ASG: Still creating... [30s elapsed]
```

- A launch configuration and auto scaling group will be created



Save up to 90% on Compute

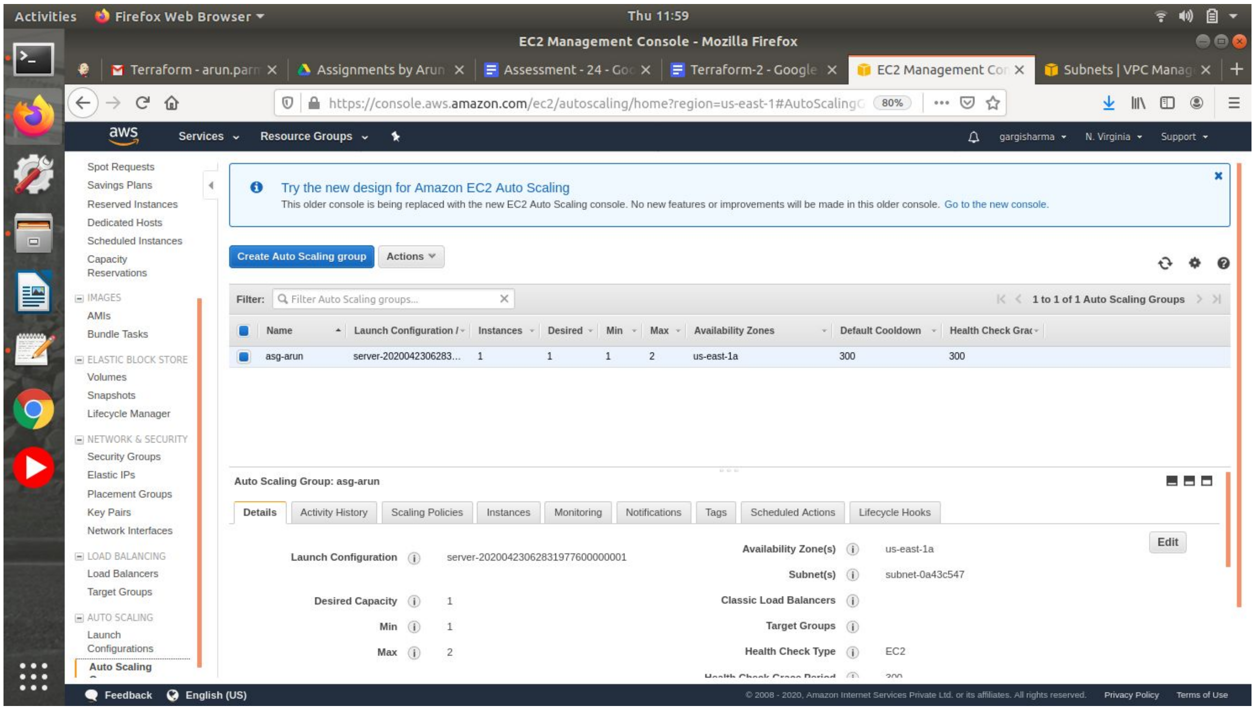
Optimize compute costs by creating your Auto Scaling group with a launch template to combine EC2 On-Demand, Spot, and RIs. Learn more.

1 to 1 of 1 Launch Configurations

Name	AMI ID	Instance Type	Spot Price	Creation Time
server-202004...	ami-07ebfd5b3...	t2.micro		April 23, 2020 at 11:58:33 AM U...

Launch Configuration: server-20200423062831977600000001

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



Try the new design for Amazon EC2 Auto Scaling

This older console is being replaced with the new EC2 Auto Scaling console. No new features or improvements will be made in this older console. Go to the new console.

1 to 1 of 1 Auto Scaling Groups

Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Check Grac
asg-arun	server-2020042306283...	1	1	1	2	us-east-1a	300	300

Auto Scaling Group: asg-arun

Launch Configuration: server-20200423062831977600000001

Desired Capacity: 1

Min: 1

Max: 2

Availability Zone(s): us-east-1a

Subnet(s): subnet-0a43c547

Classic Load Balancers:

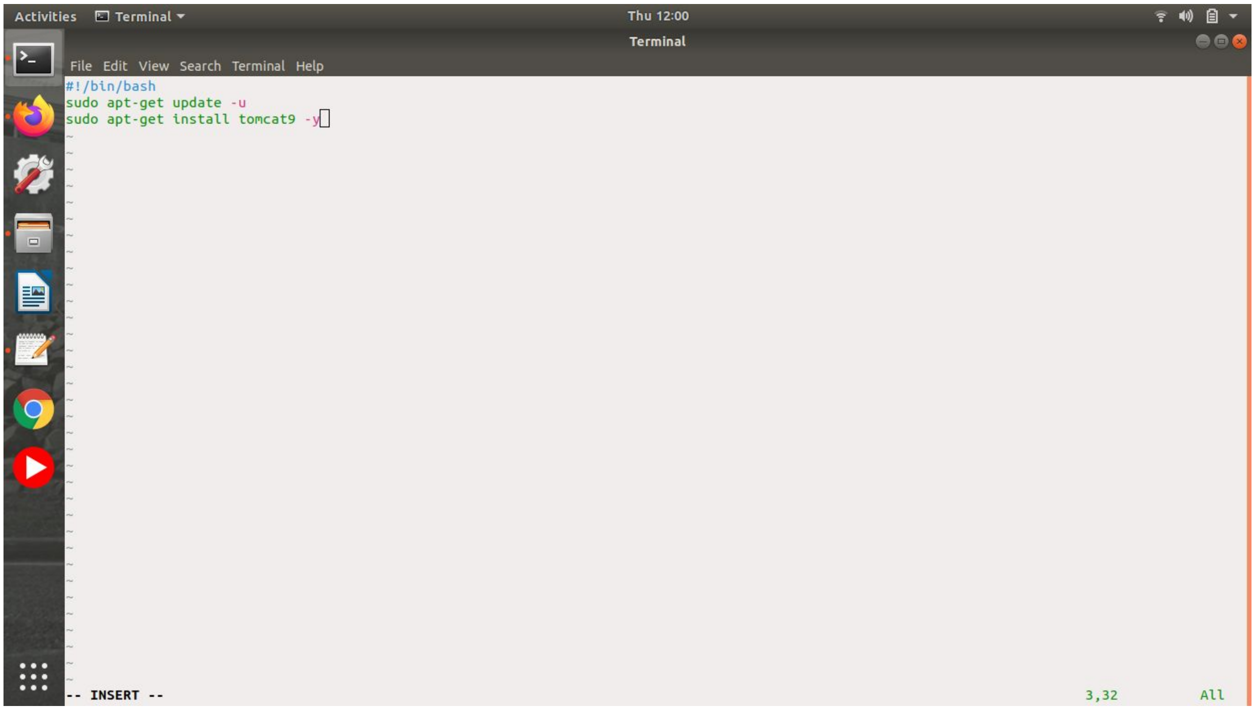
Target Groups:

Health Check Type: EC2

Health Check Grace Period: 300

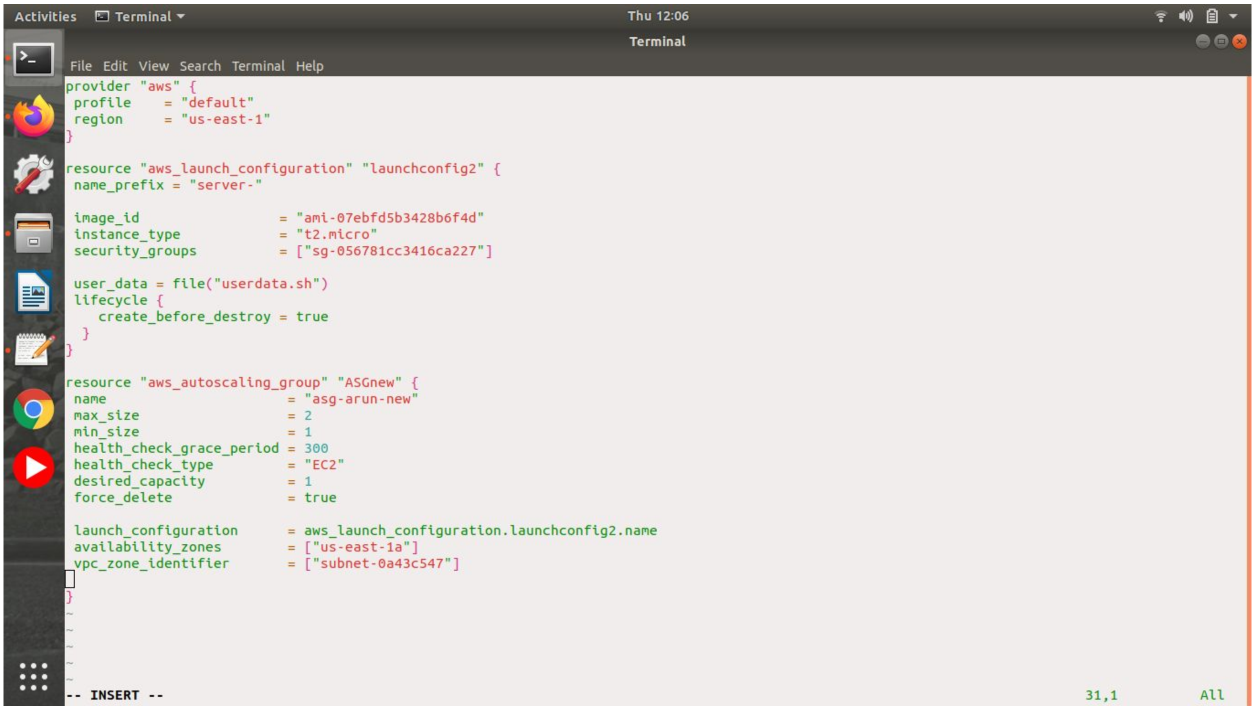
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

- Now change the userdata and name of ASG and Launch Conf



```
#!/bin/bash
sudo apt-get update -u
sudo apt-get install tomcat9 -y
```

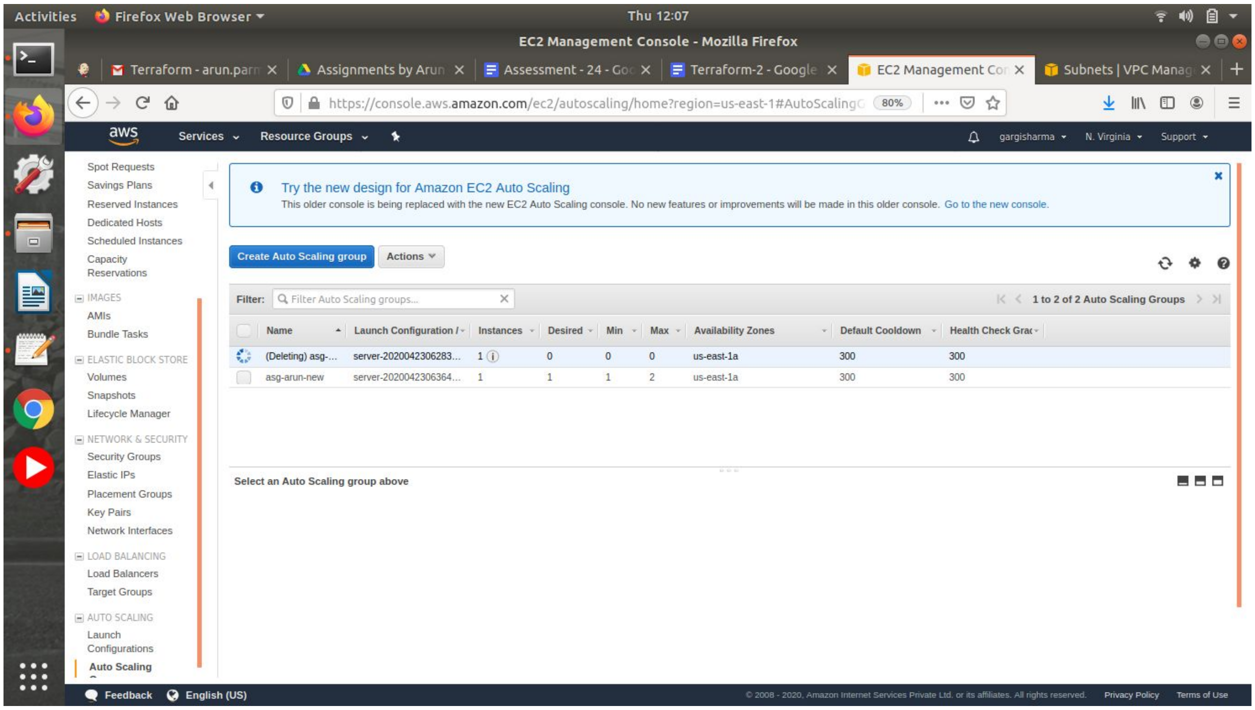
```
-- INSERT --                                                   3,32          All
```



```
provider "aws" {
 profile     = "default"
 region      = "us-east-1"
}

resource "aws_launch_configuration" "launchconfig2" {
 name_prefix = "server-"

 image_id                   = "ami-07ebfd5b3428b6f4d"
 instance_type              = "t2.micro"
 security_groups            = ["sg-056781cc3416ca227"]

 user_data = file("userdata.sh")
 lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "ASGnew" {
 name                        = "asg-arun-new"
 max_size                    = 2
 min_size                    = 1
 health_check_grace_period = 300
 health_check_type           = "EC2"
 desired_capacity            = 1
 force_delete                = true

 launch_configuration        = aws_launch_configuration.launchconfig2.name
 availability_zones          = ["us-east-1a"]
 vpc_zone_identifier         = ["subnet-0a43c547"]

}
```

```
-- INSERT --                                                   31,1          All
```

- Now you will first a new auto scaling group is created than the old is deleted and at last the old launch config is deleted



Try the new design for Amazon EC2 Auto Scaling

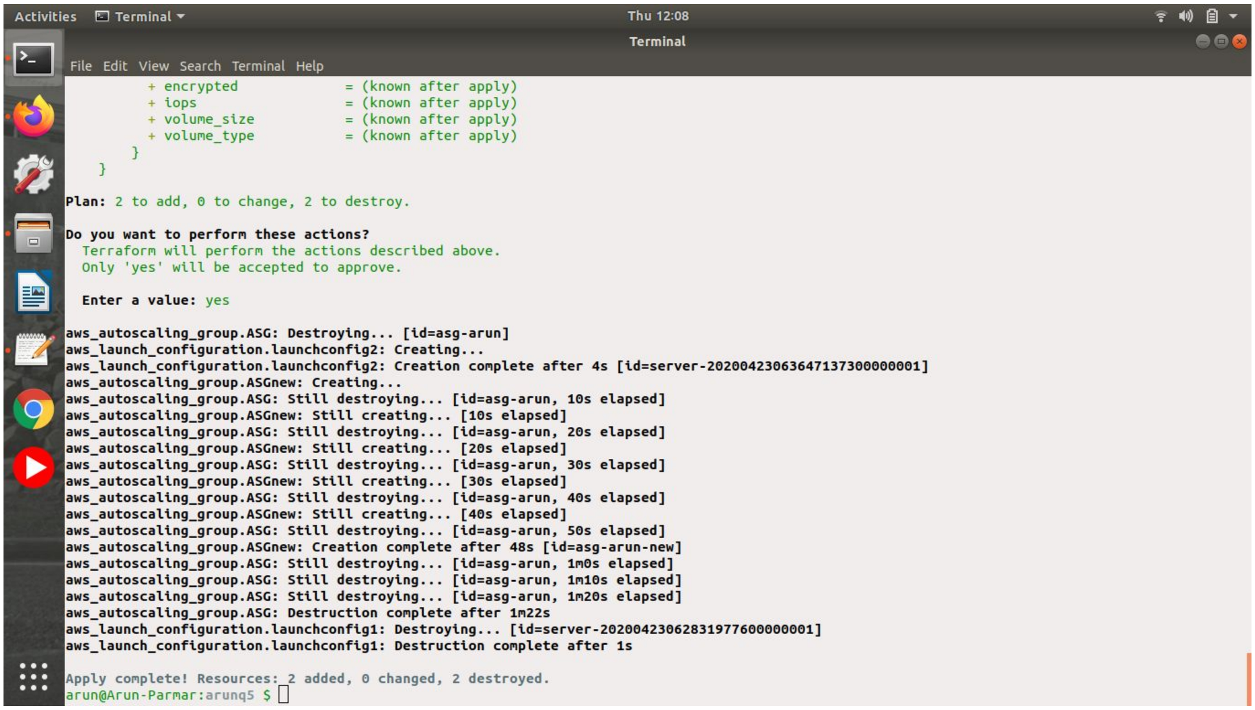
This older console is being replaced with the new EC2 Auto Scaling console. No new features or improvements will be made in this older console. Go to the new console.

1 to 2 of 2 Auto Scaling Groups

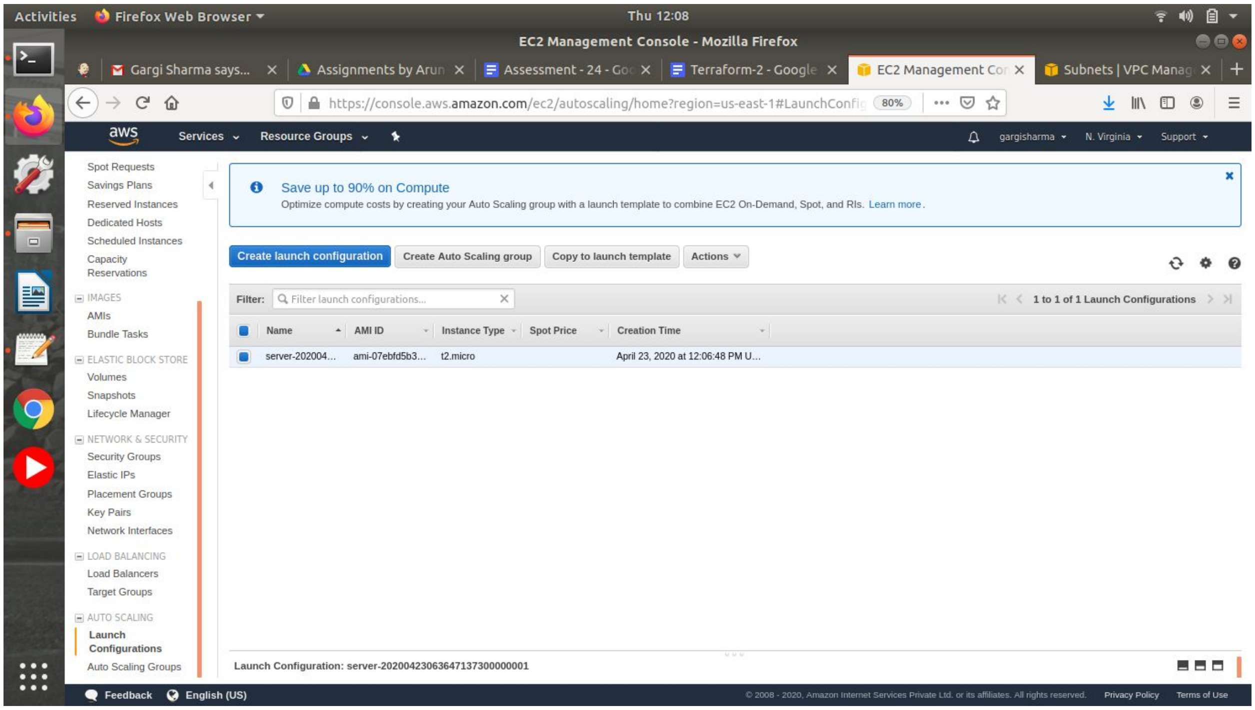
Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Check Grac
(Deleting) asg-...	server-2020042306283...	1	0	0	0	us-east-1a	300	300
asg-arun-new	server-2020042306364...	1	1	1	2	us-east-1a	300	300

Select an Auto Scaling group above

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



```
        + encrypted             = (known after apply)
        + iops                  = (known after apply)
        + volume_size           = (known after apply)
        + volume_type           = (known after apply)
      }
  }

Plan: 2 to add, 0 to change, 2 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_autoscaling_group.ASG: Destroying... [id=asg-arun]
aws_launch_configuration.launchconfig2: Creating...
aws_launch_configuration.launchconfig2: Creation complete after 4s [id=server-20200423063647137300000001]
aws_autoscaling_group.ASGnew: Creating...
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 10s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [10s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 20s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [20s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 30s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [30s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 40s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [40s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 50s elapsed]
aws_autoscaling_group.ASGnew: Creation complete after 48s [id=asg-arun-new]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m0s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m10s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m20s elapsed]
aws_autoscaling_group.ASG: Destruction complete after 1m22s
aws_launch_configuration.launchconfig1: Destroying... [id=server-20200423062831977600000001]
aws_launch_configuration.launchconfig1: Destruction complete after 1s

Apply complete! Resources: 2 added, 0 changed, 2 destroyed.
arun@Arun-Parmar:arunq5 $
```



Save up to 90% on Compute

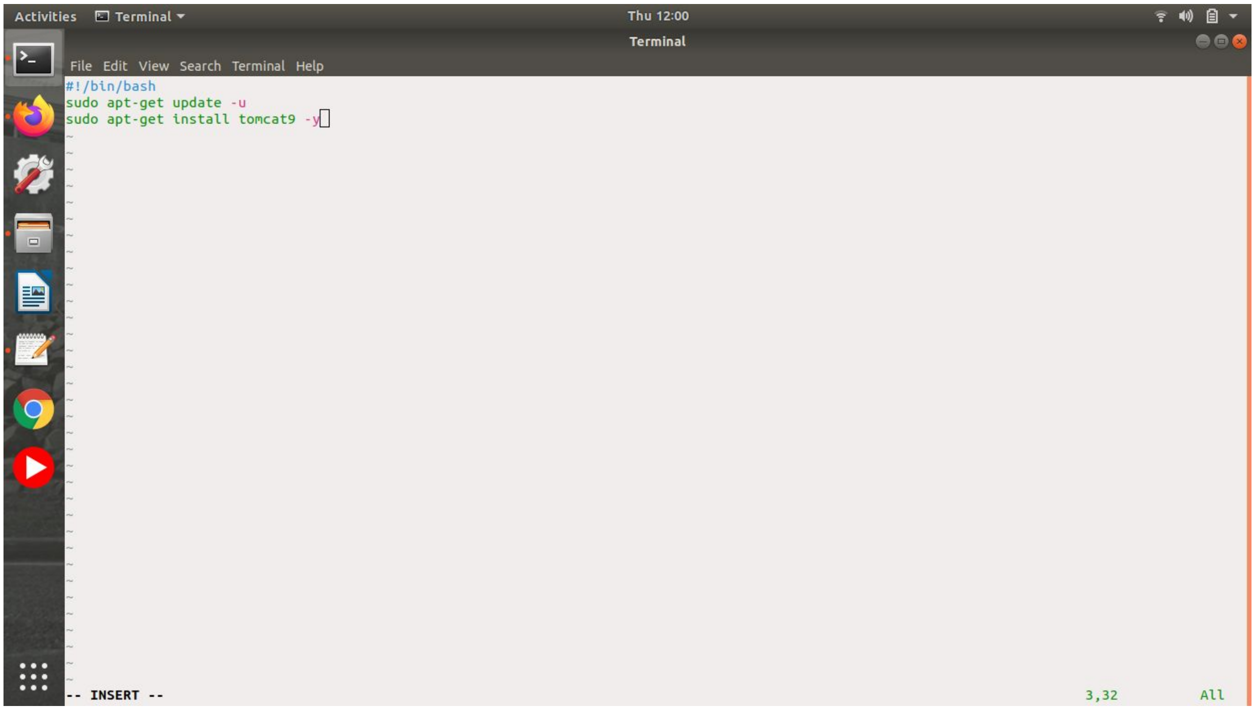
Optimize compute costs by creating your Auto Scaling group with a launch template to combine EC2 On-Demand, Spot, and RIs. Learn more.

Name	AMI ID	Instance Type	Spot Price	Creation Time
server-202004...	ami-07ebfd5b3...	t2.micro		April 23, 2020 at 12:06:48 PM U...

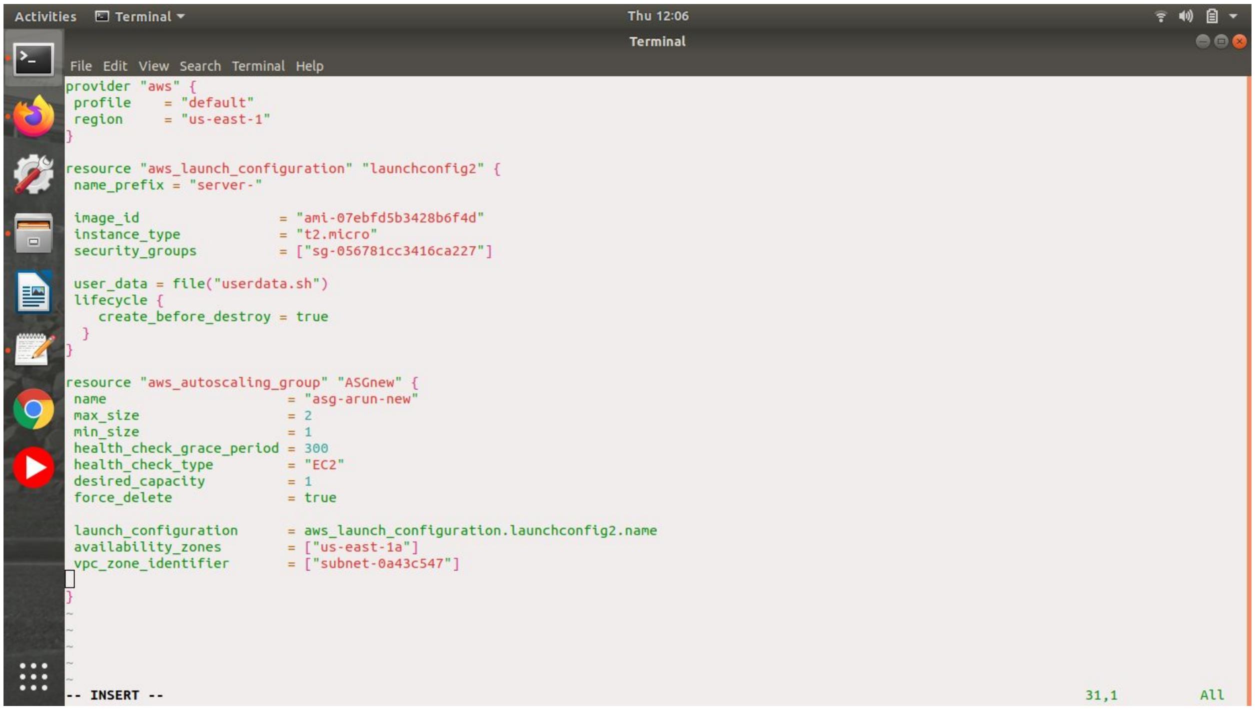
Launch Configuration: server-20200423063647137300000001

2. Deploy a sample nginx/tomcat/react service on it.

- Edit userdata and main.tf

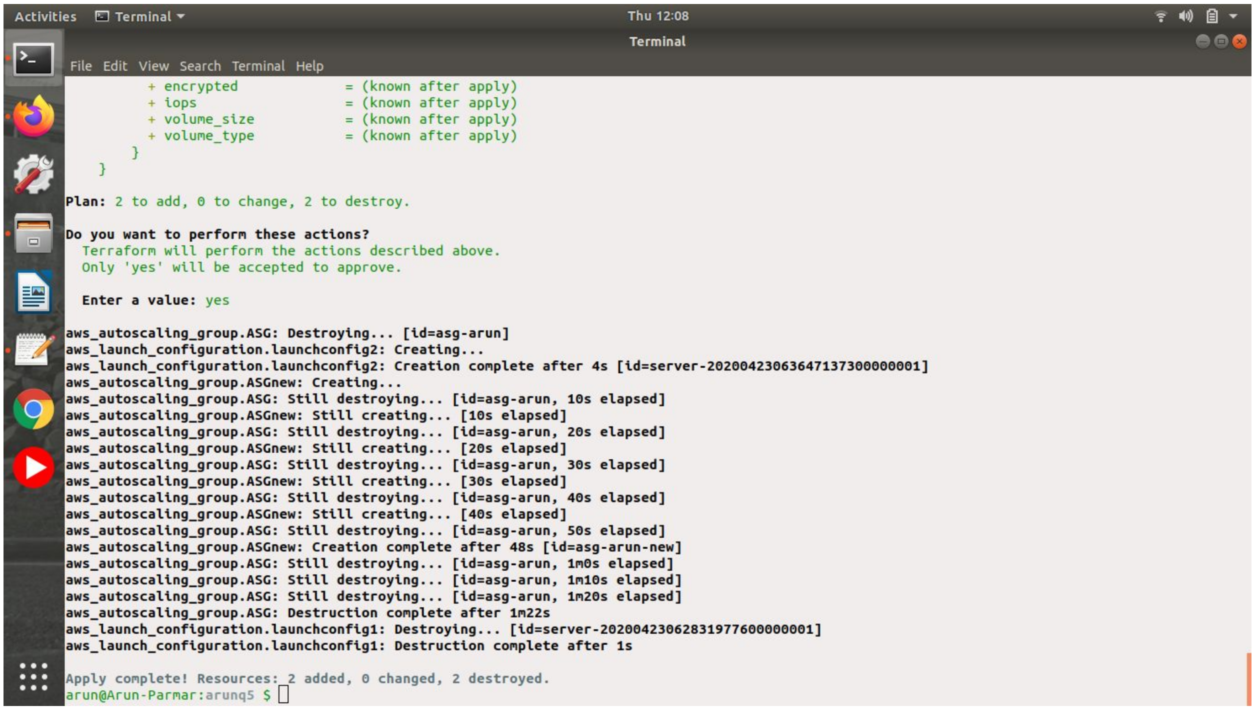


```
#!/bin/bash
sudo apt-get update -u
sudo apt-get install tomcat9 -y
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                              3,32          All
```

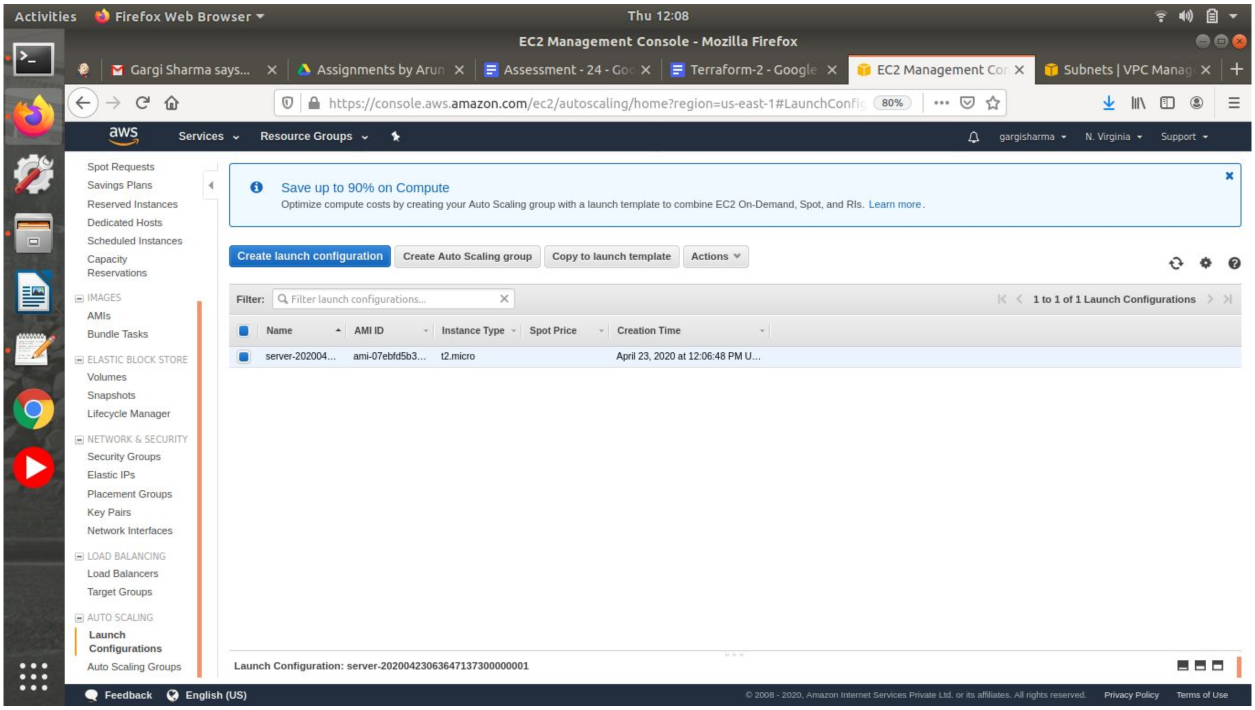


```
provider "aws" {
 profile    = "default"
 region     = "us-east-1"
}

resource "aws_launch_configuration" "launchconfig2" {
 name_prefix = "server-"

 image_id                  = "ami-07ebfd5b3428b6f4d"
 instance_type             = "t2.micro"
 security_groups           = ["sg-056781cc3416ca227"]

 user_data = file("userdata.sh")
 lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "ASGnew" {
 name                       = "asg-arun-new"
 max_size                   = 2
 min_size                   = 1
 health_check_grace_period  = 300
 health_check_type          = "EC2"
 desired_capacity           = 1
 force_delete               = true

 launch_configuration       = aws_launch_configuration.launchconfig2.name
 availability_zones         = ["us-east-1a"]
 vpc_zone_identifier        = ["subnet-0a43c547"]

}
~
~
~
~
~
-- INSERT --                                                              31,1          All
```

- Terraform init > Terraform apply



```
          + encrypted             = (known after apply)
          + iops                  = (known after apply)
          + volume_size           = (known after apply)
          + volume_type           = (known after apply)
        }
    }

Plan: 2 to add, 0 to change, 2 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_autoscaling_group.ASG: Destroying... [id=asg-arun]
aws_launch_configuration.launchconfig2: Creating...
aws_launch_configuration.launchconfig2: Creation complete after 4s [id=server-20200423063647137300000001]
aws_autoscaling_group.ASGnew: Creating...
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 10s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [10s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 20s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [20s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 30s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [30s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 40s elapsed]
aws_autoscaling_group.ASGnew: Still creating... [40s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 50s elapsed]
aws_autoscaling_group.ASGnew: Creation complete after 48s [id=asg-arun-new]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m0s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m10s elapsed]
aws_autoscaling_group.ASG: Still destroying... [id=asg-arun, 1m20s elapsed]
aws_autoscaling_group.ASG: Destruction complete after 1m22s
aws_launch_configuration.launchconfig1: Destroying... [id=server-20200423062831977600000001]
aws_launch_configuration.launchconfig1: Destruction complete after 1s

Apply complete! Resources: 2 added, 0 changed, 2 destroyed.
arun@Arun-Parmar:arunq5 $ 
```



https://console.aws.amazon.com/ec2/autoscaling/home?region=us-east-1#LaunchConfig

Save up to 90% on Compute

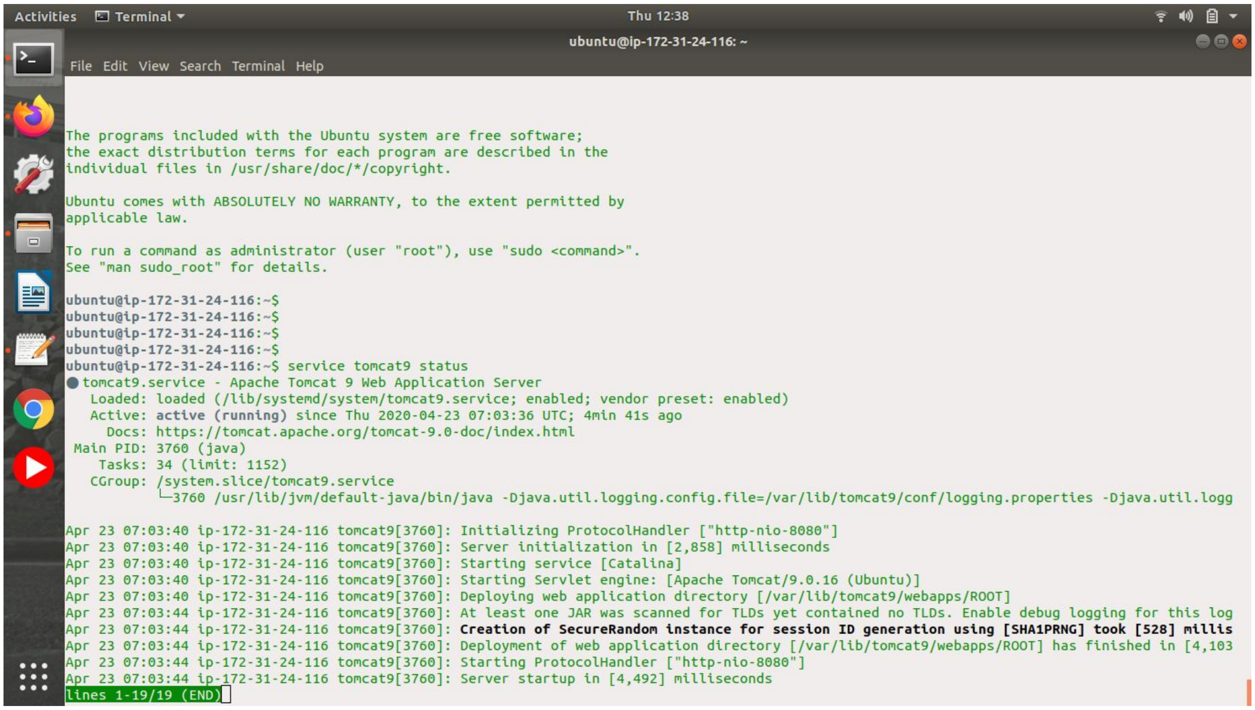
Optimize compute costs by creating your Auto Scaling group with a launch template to combine EC2 On-Demand, Spot, and RIs. Learn more.

1 to 1 of 1 Launch Configurations

Name	AMI ID	Instance Type	Spot Price	Creation Time
server-202004...	ami-07ebfd5b3...	t2.micro		April 23, 2020 at 12:06:48 PM U...

Launch Configuration: server-20200423063647137300000001

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

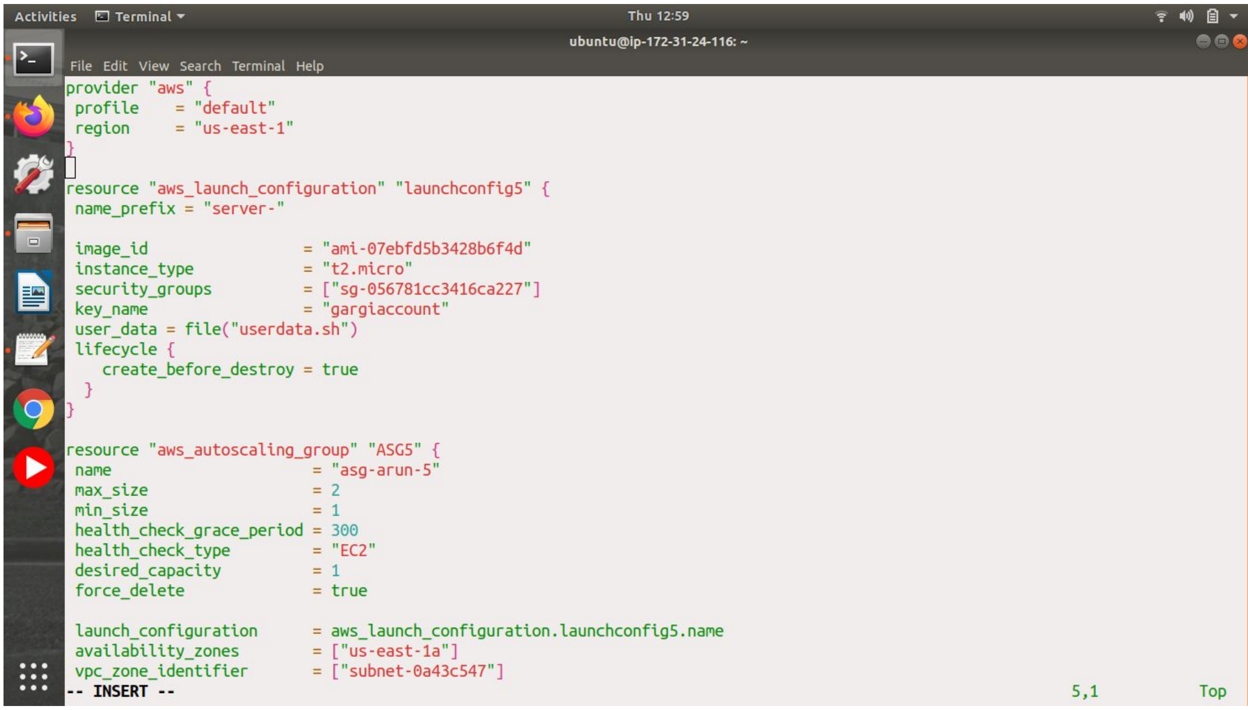


Activities
Terminal
Thu 12:38
ubuntu@ip-172-31-24-116: ~
File Edit View Search Terminal Help

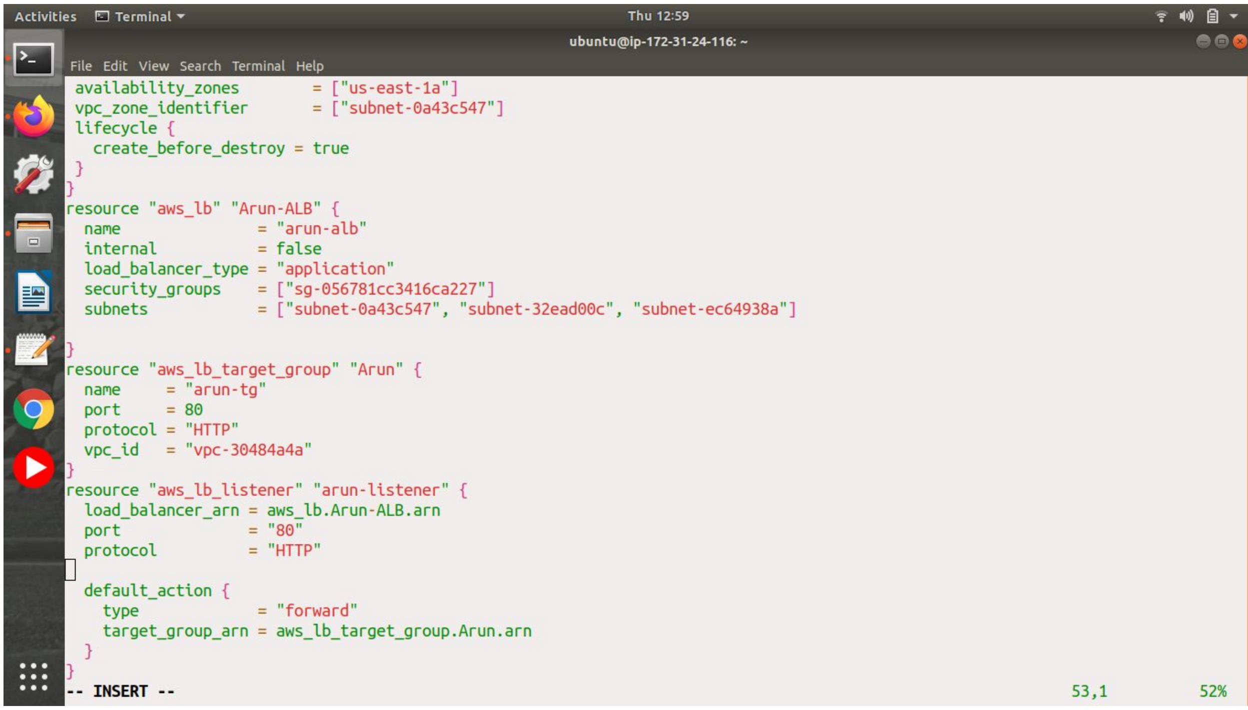
```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-24-116:~$
ubuntu@ip-172-31-24-116:~$
ubuntu@ip-172-31-24-116:~$
ubuntu@ip-172-31-24-116:~$
ubuntu@ip-172-31-24-116:~$ service tomcat9 status
● tomcat9.service - Apache Tomcat 9 Web Application Server
   Loaded: loaded (/lib/systemd/system/tomcat9.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2020-04-23 07:03:36 UTC; 4min 41s ago
     Docs: https://tomcat.apache.org/tomcat-9.0-doc/index.html
 Main PID: 3760 (java)
    Tasks: 34 (limit: 1152)
   CGroup: /system.slice/tomcat9.service
           └─3760 /usr/lib/jvm/default-java/bin/java -Djava.util.logging.config.file=/var/lib/tomcat9/conf/logging.properties -Djava.util.logg

Apr 23 07:03:40 ip-172-31-24-116 tomcat9[3760]: Initializing ProtocolHandler ["http-nio-8080"]
Apr 23 07:03:40 ip-172-31-24-116 tomcat9[3760]: Server initialization in [2,858] milliseconds
Apr 23 07:03:40 ip-172-31-24-116 tomcat9[3760]: Starting service [Catalina]
Apr 23 07:03:40 ip-172-31-24-116 tomcat9[3760]: Starting Servlet engine: [Apache Tomcat/9.0.16 (Ubuntu)]
Apr 23 07:03:40 ip-172-31-24-116 tomcat9[3760]: Deploying web application directory [/var/lib/tomcat9/webapps/ROOT]
Apr 23 07:03:44 ip-172-31-24-116 tomcat9[3760]: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this log
Apr 23 07:03:44 ip-172-31-24-116 tomcat9[3760]: Creation of SecureRandom instance for session ID generation using [SHA1PRNG] took [528] millis
Apr 23 07:03:44 ip-172-31-24-116 tomcat9[3760]: Deployment of web application directory [/var/lib/tomcat9/webapps/ROOT] has finished in [4,103
Apr 23 07:03:44 ip-172-31-24-116 tomcat9[3760]: Starting ProtocolHandler ["http-nio-8080"]
Apr 23 07:03:44 ip-172-31-24-116 tomcat9[3760]: Server startup in [4,492] milliseconds
lines 1-19/19 (END)
```

3. Attach a LB and create R53 endpoint pointing to lab, service should be accessible from the endpoint.

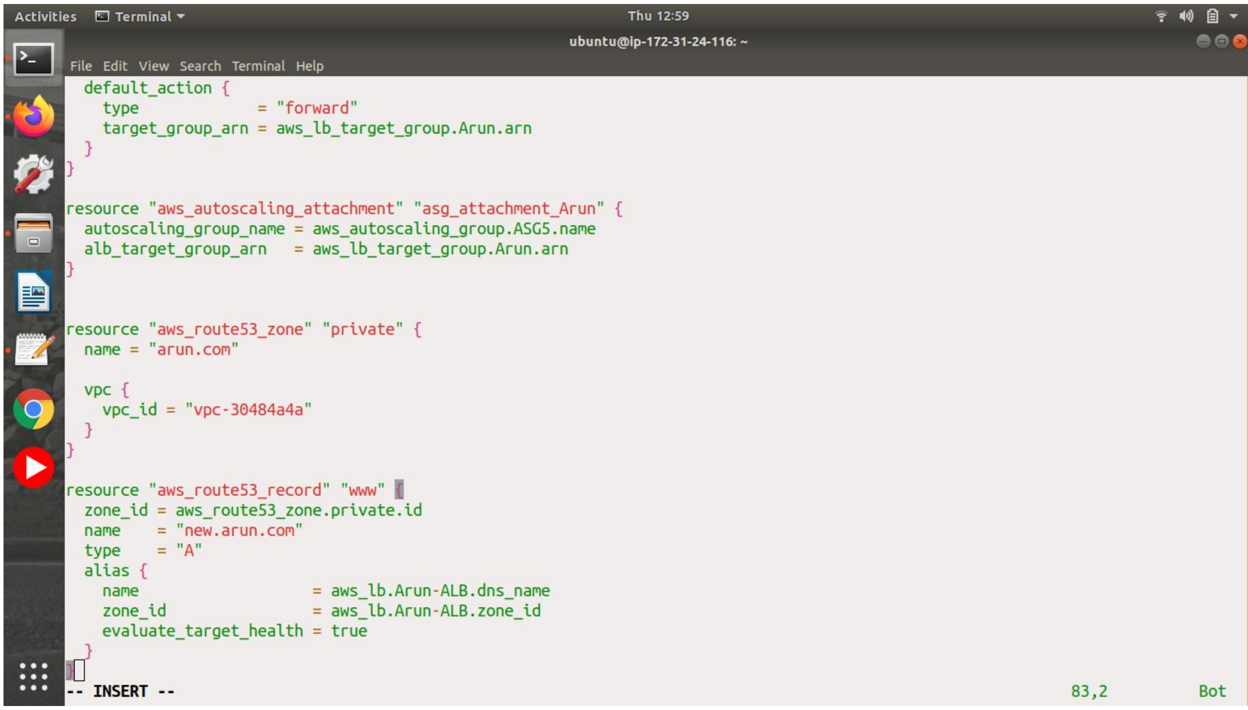
- Edit main.tf


Activities
Terminal
Thu 12:59
ubuntu@ip-172-31-24-116: ~
File Edit View Search Terminal Help

```
provider "aws" {
 profile    = "default"
 region     = "us-east-1"
}

resource "aws_launch_configuration" "launchconfig5" {
 name_prefix = "server-"

 image_id                = "ami-07ebfd5b3428b6f4d"
 instance_type           = "t2.micro"
 security_groups         = ["sg-056781cc3416ca227"]
 key_name                = "gargiaccount"
 user_data = file("userdata.sh")
 lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "ASG5" {
 name                      = "asg-arun-5"
 max_size                  = 2
 min_size                  = 1
 health_check_grace_period = 300
 health_check_type         = "EC2"
 desired_capacity          = 1
 force_delete              = true

 launch_configuration      = aws_launch_configuration.launchconfig5.name
 availability_zones        = ["us-east-1a"]
 vpc_zone_identifier       = ["subnet-0a43c547"]
-- INSERT --                                                                   5,1           Top
```

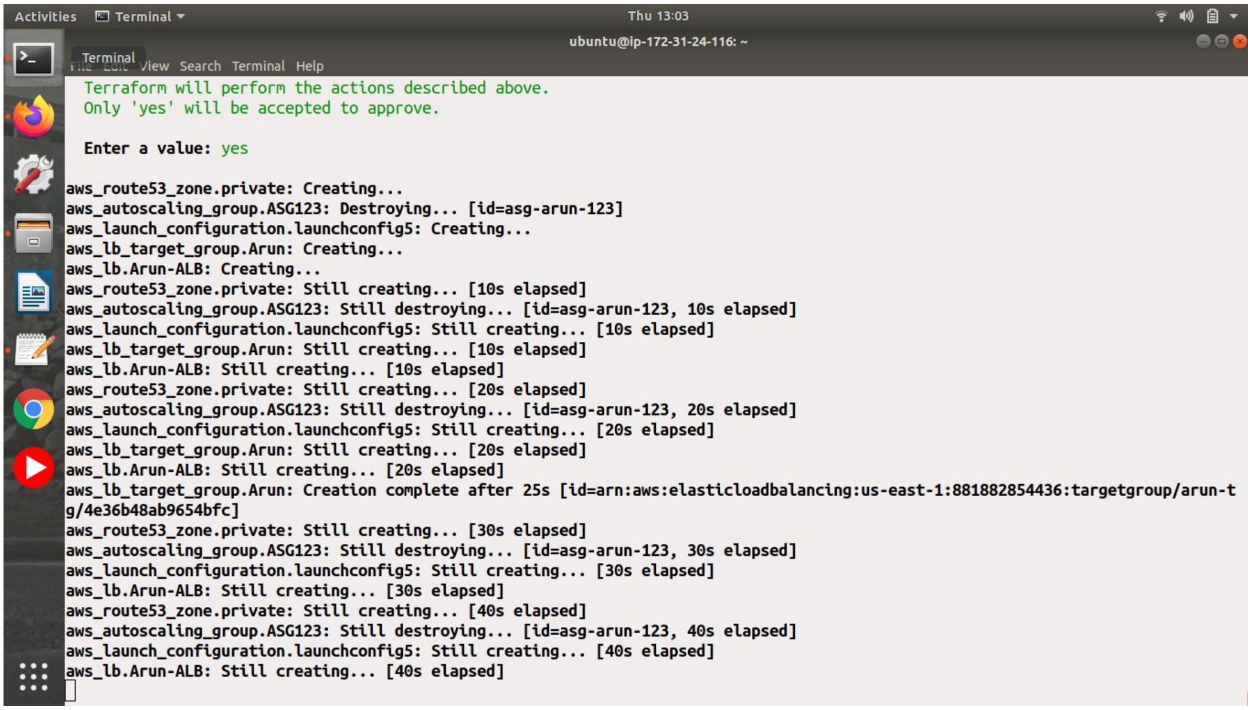

Activities
Terminal
Thu 12:59
ubuntu@ip-172-31-24-116: ~
File Edit View Search Terminal Help

```
 availability_zones        = ["us-east-1a"]
 vpc_zone_identifier       = ["subnet-0a43c547"]
 lifecycle {
   create_before_destroy = true
 }
}
resource "aws_lb" "Arun-ALB" {
  name               = "arun-alb"
  internal           = false
  load_balancer_type = "application"
  security_groups    = ["sg-056781cc3416ca227"]
  subnets            = ["subnet-0a43c547", "subnet-32ead00c", "subnet-ec64938a"]

}
resource "aws_lb_target_group" "Arun" {
  name     = "arun-tg"
  port     = 80
  protocol = "HTTP"
  vpc_id   = "vpc-30484a4a"
}
resource "aws_lb_listener" "arun-listener" {
  load_balancer_arn = aws_lb.Arun-ALB.arn
  port              = "80"
  protocol          = "HTTP"

  default_action {
    type             = "forward"
    target_group_arn = aws_lb_target_group.Arun.arn
  }
}
-- INSERT --                                                                   53,1          52%
```


Activities
Terminal
Thu 12:59
ubuntu@ip-172-31-24-116: ~
File Edit View Search Terminal Help

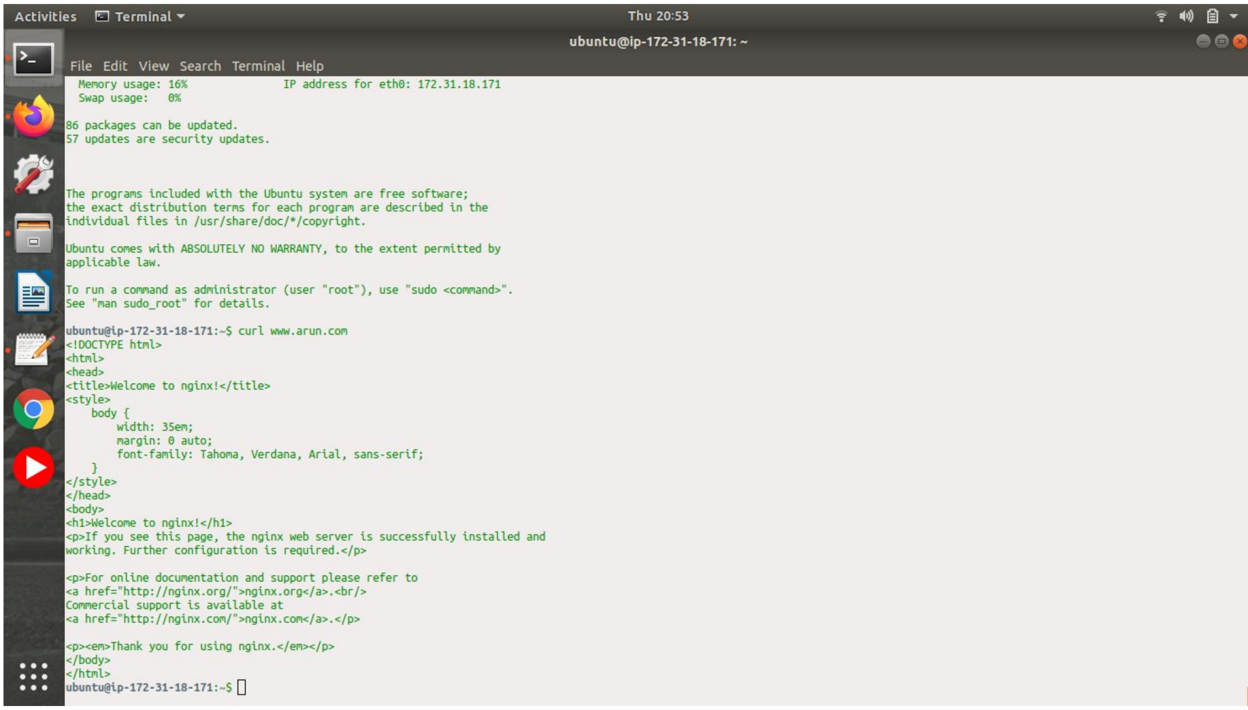
```
  default_action {
    type             = "forward"
    target_group_arn = aws_lb_target_group.Arun.arn
  }
}

resource "aws_autoscaling_attachment" "asg_attachment_Arun" {
  autoscaling_group_name = aws_autoscaling_group.ASG5.name
  alb_target_group_arn   = aws_lb_target_group.Arun.arn
}

resource "aws_route53_zone" "private" {
  name = "arun.com"

  vpc {
    vpc_id = "vpc-30484a4a"
  }
}

resource "aws_route53_record" "www" {
  zone_id = aws_route53_zone.private.id
  name    = "new.arun.com"
  type    = "A"
  alias {
    name                   = aws_lb.Arun-ALB.dns_name
    zone_id                = aws_lb.Arun-ALB.zone_id
    evaluate_target_health = true
  }
}
-- INSERT --                                                                83,2          Bot
```

- Terraform init > terraform apply



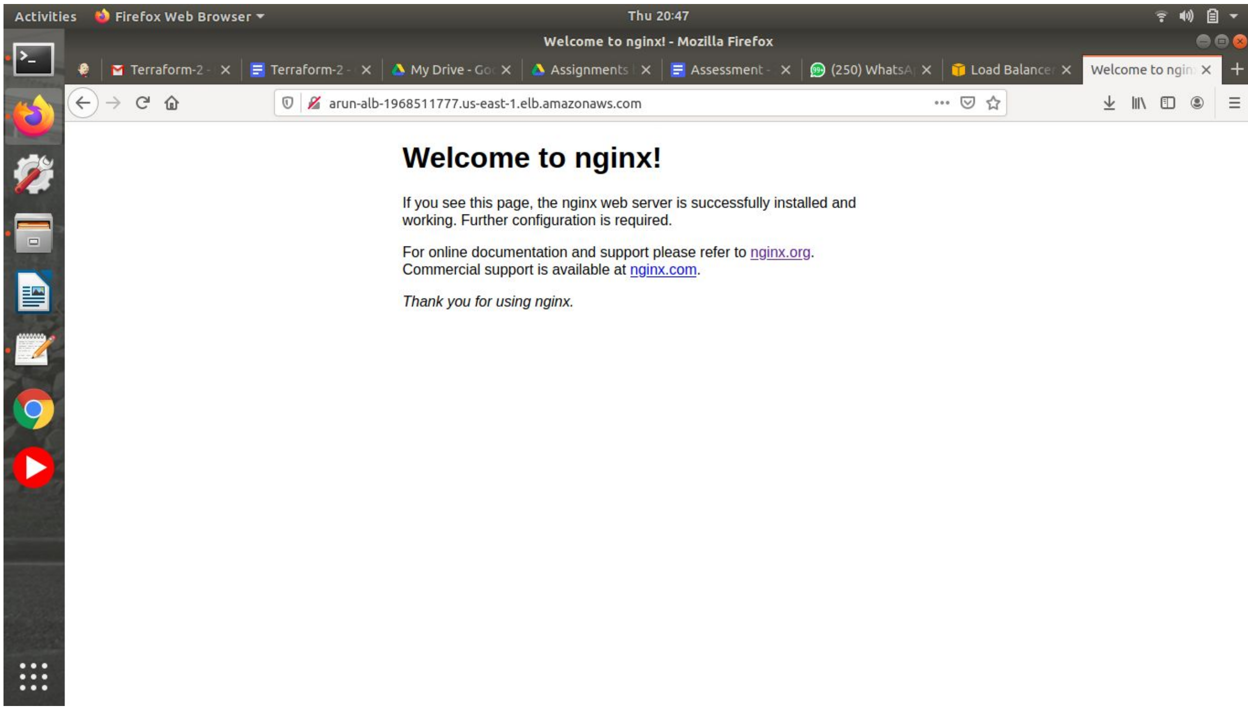
```
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_route53_zone.private: Creating...
aws_autoscaling_group.ASG123: Destroying... [id=asg-arun-123]
aws_launch_configuration.launchconfig5: Creating...
aws_lb_target_group.Arun: Creating...
aws_lb.Arun-ALB: Creating...
aws_route53_zone.private: Still creating... [10s elapsed]
aws_autoscaling_group.ASG123: Still destroying... [id=asg-arun-123, 10s elapsed]
aws_launch_configuration.launchconfig5: Still creating... [10s elapsed]
aws_lb_target_group.Arun: Still creating... [10s elapsed]
aws_lb.Arun-ALB: Still creating... [10s elapsed]
aws_route53_zone.private: Still creating... [20s elapsed]
aws_autoscaling_group.ASG123: Still destroying... [id=asg-arun-123, 20s elapsed]
aws_launch_configuration.launchconfig5: Still creating... [20s elapsed]
aws_lb_target_group.Arun: Still creating... [20s elapsed]
aws_lb.Arun-ALB: Still creating... [20s elapsed]
aws_lb_target_group.Arun: Creation complete after 25s [id=arn:aws:elasticloadbalancing:us-east-1:881882854436:targetgroup/arun-t
g/4e36b48ab9654bfc]
aws_route53_zone.private: Still creating... [30s elapsed]
aws_autoscaling_group.ASG123: Still destroying... [id=asg-arun-123, 30s elapsed]
aws_launch_configuration.launchconfig5: Still creating... [30s elapsed]
aws_lb.Arun-ALB: Still creating... [30s elapsed]
aws_route53_zone.private: Still creating... [40s elapsed]
aws_autoscaling_group.ASG123: Still destroying... [id=asg-arun-123, 40s elapsed]
aws_launch_configuration.launchconfig5: Still creating... [40s elapsed]
aws_lb.Arun-ALB: Still creating... [40s elapsed]
```

Ssh in instance and try the hostname


Activities
Terminal
Thu 20:53
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

```
  Memory usage: 16%               IP address for eth0: 172.31.18.171
  Swap usage:   0%

86 packages can be updated.
57 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-18-171:~$ curl www.arun.com
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
ubuntu@ip-172-31-18-171:~$ 
```

Hit on dns of alb



Welcome to nginx!

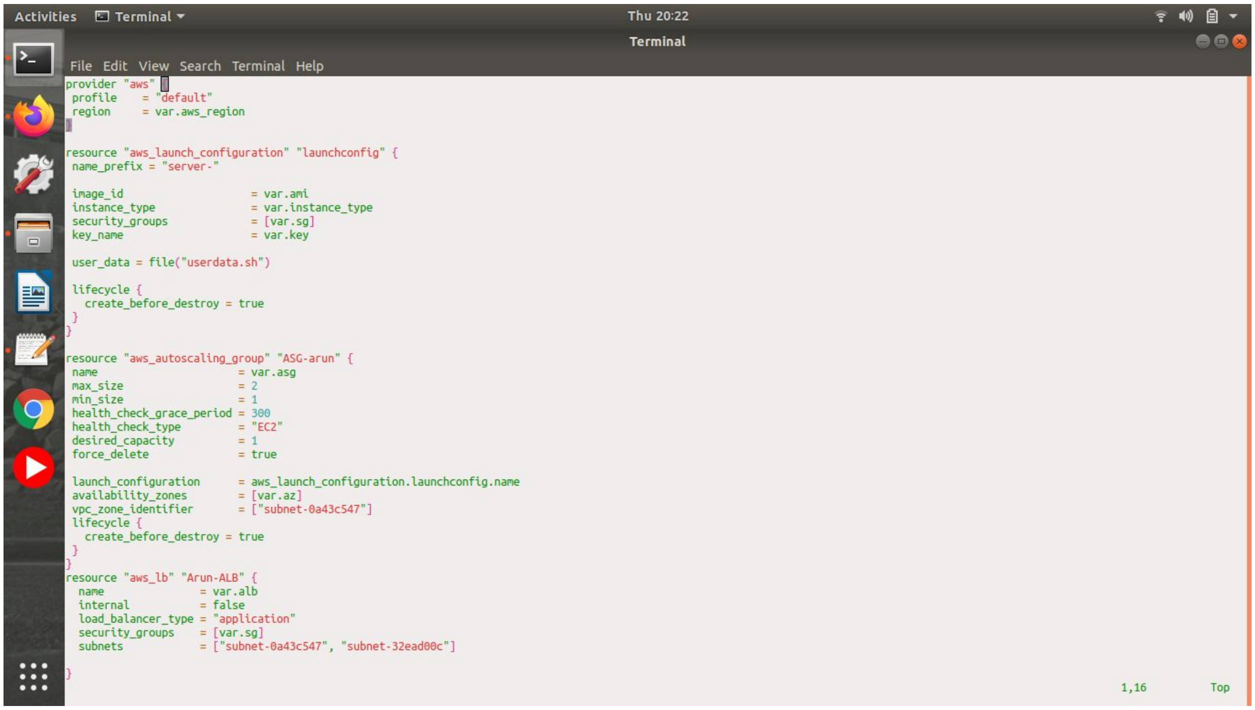
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

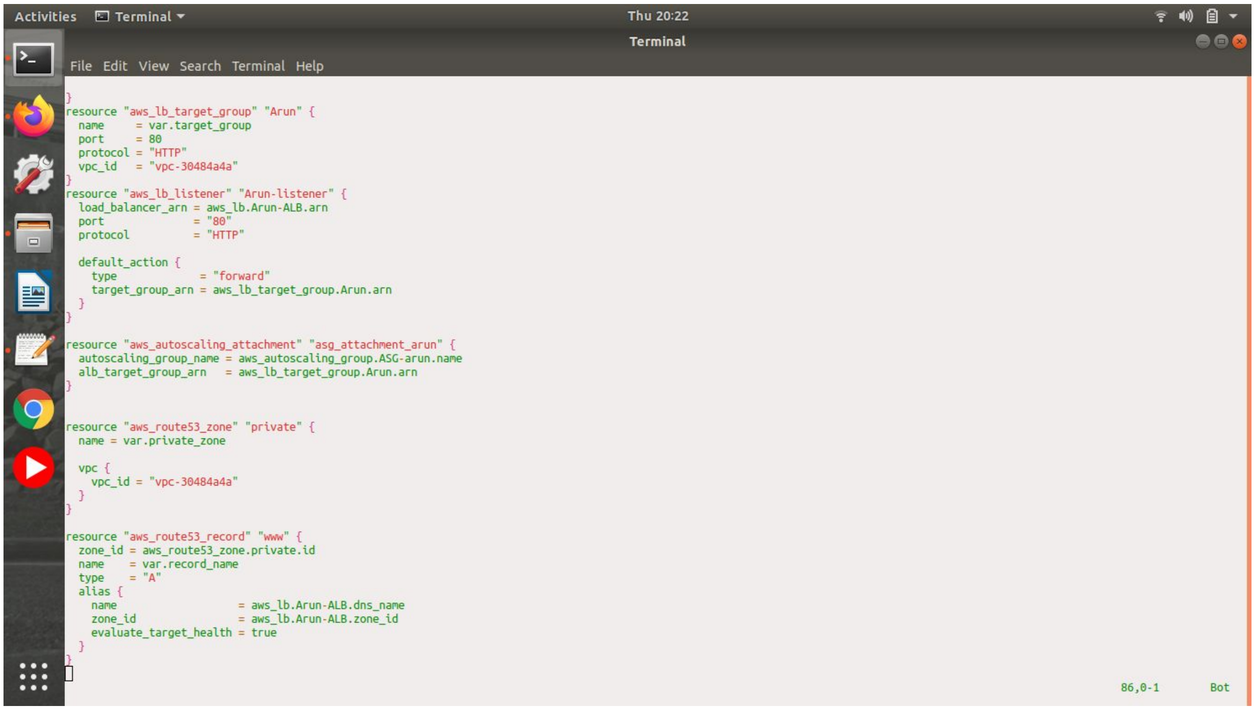
Thank you for using nginx.

4. Variablize all parameters and pass values as env.tfvars file

- Edit main.tf

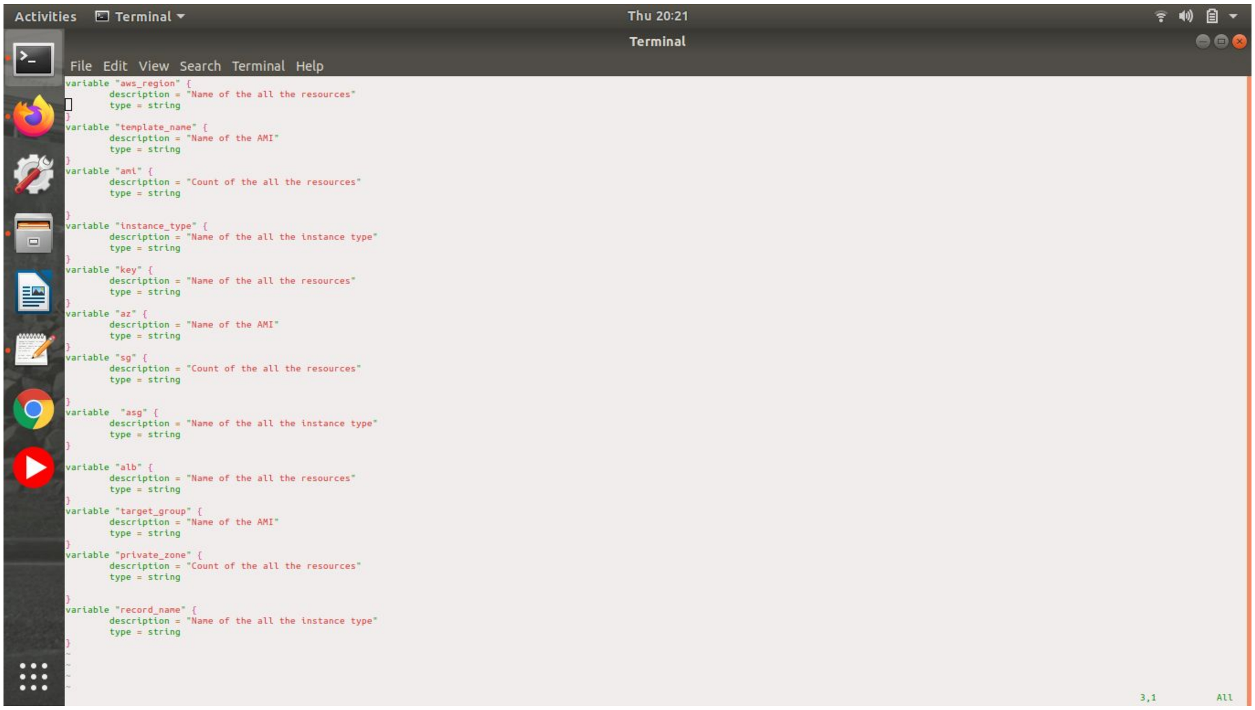


```
provider "aws" {
 profile    = "default"
 region     = var.aws_region
}

resource "aws_launch_configuration" "launchconfig" {
 name_prefix = "server-"

 image_id                    = var.ami
 instance_type               = var.instance_type
 security_groups             = [var.sg]
 key_name                    = var.key

 user_data = file("userdata.sh")

 lifecycle {
   create_before_destroy = true
 }
}

resource "aws_autoscaling_group" "ASG-arun" {
 name                      = var.asg
 max_size                  = 2
 min_size                  = 1
 health_check_grace_period = 300
 health_check_type         = "EC2"
 desired_capacity          = 1
 force_delete              = true

 launch_configuration      = aws_launch_configuration.launchconfig.name
 availability_zones        = [var.az]
 vpc_zone_identifier       = ["subnet-0a43c547"]
 lifecycle {
   create_before_destroy = true
 }
}
resource "aws_lb" "Arun-ALB" {
  name               = var.alb
  internal           = false
  load_balancer_type = "application"
  security_groups    = [var.sg]
  subnets            = ["subnet-0a43c547", "subnet-32ead00c"]

}
                                                                        1,16          Top
```



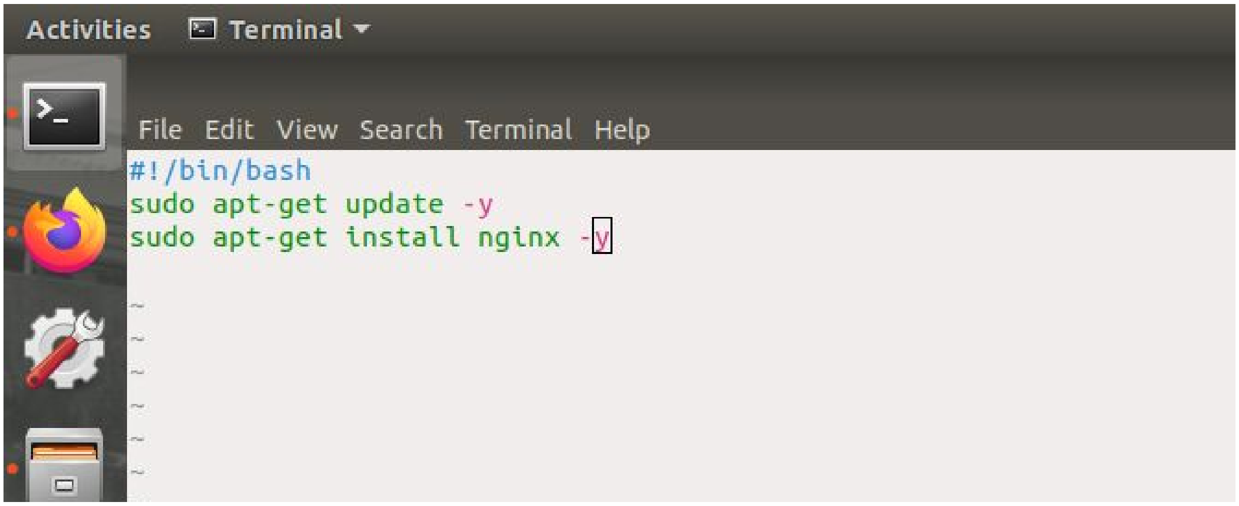
```

}
resource "aws_lb_target_group" "Arun" {
  name     = var.target_group
  port     = 80
  protocol = "HTTP"
  vpc_id   = "vpc-30484a4a"
}
resource "aws_lb_listener" "Arun-listener" {
  load_balancer_arn = aws_lb.Arun-ALB.arn
  port              = "80"
  protocol          = "HTTP"

  default_action {
    type             = "forward"
    target_group_arn = aws_lb_target_group.Arun.arn
  }
}

resource "aws_autoscaling_attachment" "asg_attachment_arun" {
  autoscaling_group_name = aws_autoscaling_group.ASG-arun.name
  alb_target_group_arn   = aws_lb_target_group.Arun.arn
}


resource "aws_route53_zone" "private" {
  name = var.private_zone

  vpc {
    vpc_id = "vpc-30484a4a"
  }
}

resource "aws_route53_record" "www" {
  zone_id = aws_route53_zone.private.id
  name    = var.record_name
  type    = "A"
  alias {
    name                   = aws_lb.Arun-ALB.dns_name
    zone_id                = aws_lb.Arun-ALB.zone_id
    evaluate_target_health = true
  }
}

                                                                        86,0-1        Bot
```

- Edit variables.tf



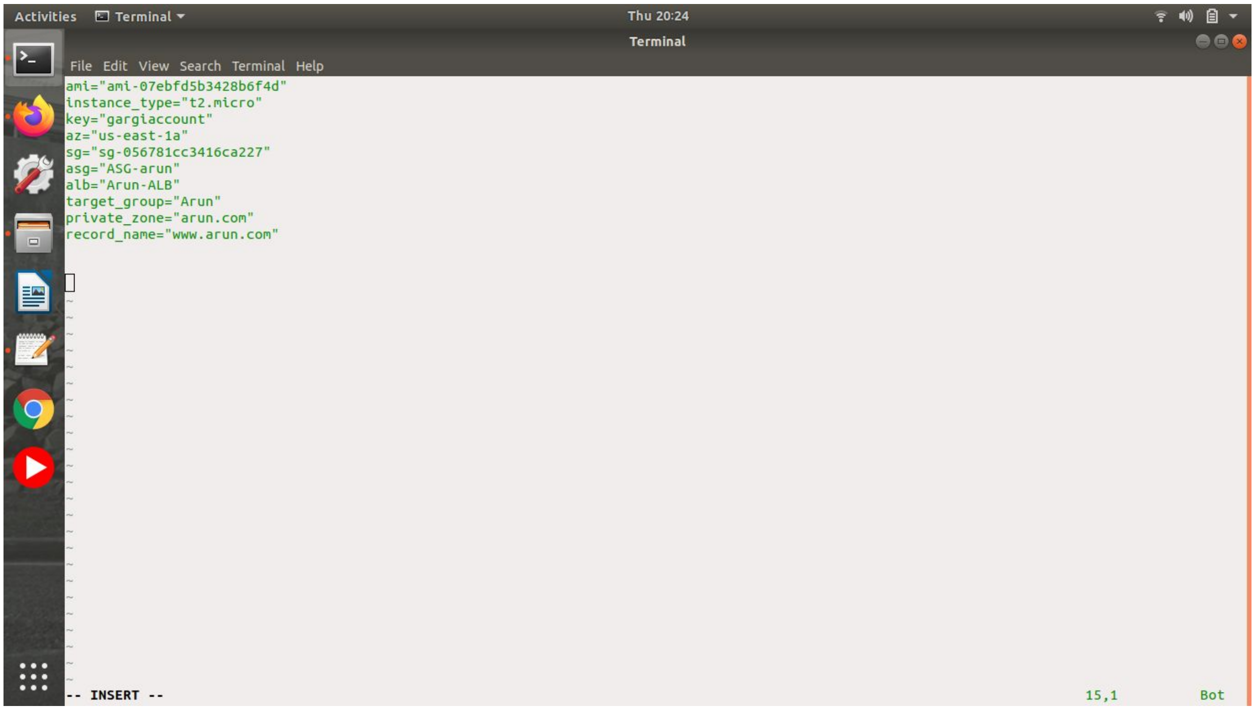
```
variable "aws_region" {
        description = "Name of the all the resources"
        type = string
}
variable "template_name" {
        description = "Name of the AMI"
        type = string
}
variable "ami" {
        description = "Count of the all the resources"
        type = string

}
variable "instance_type" {
        description = "Name of the all the instance type"
        type = string
}
variable "key" {
        description = "Name of the all the resources"
        type = string
}
variable "az" {
        description = "Name of the AMI"
        type = string
}
variable "sg" {
        description = "Count of the all the resources"
        type = string

}
variable  "asg" {
        description = "Name of the all the instance type"
        type = string
}

variable "alb" {
        description = "Name of the all the resources"
        type = string
}
variable "target_group" {
        description = "Name of the AMI"
        type = string
}
variable "private_zone" {
        description = "Count of the all the resources"
        type = string

}
variable "record_name" {
        description = "Name of the all the instance type"
        type = string
}
~
~
~
~
                                                                                     3,1           All
```

- Edit userdata.sh



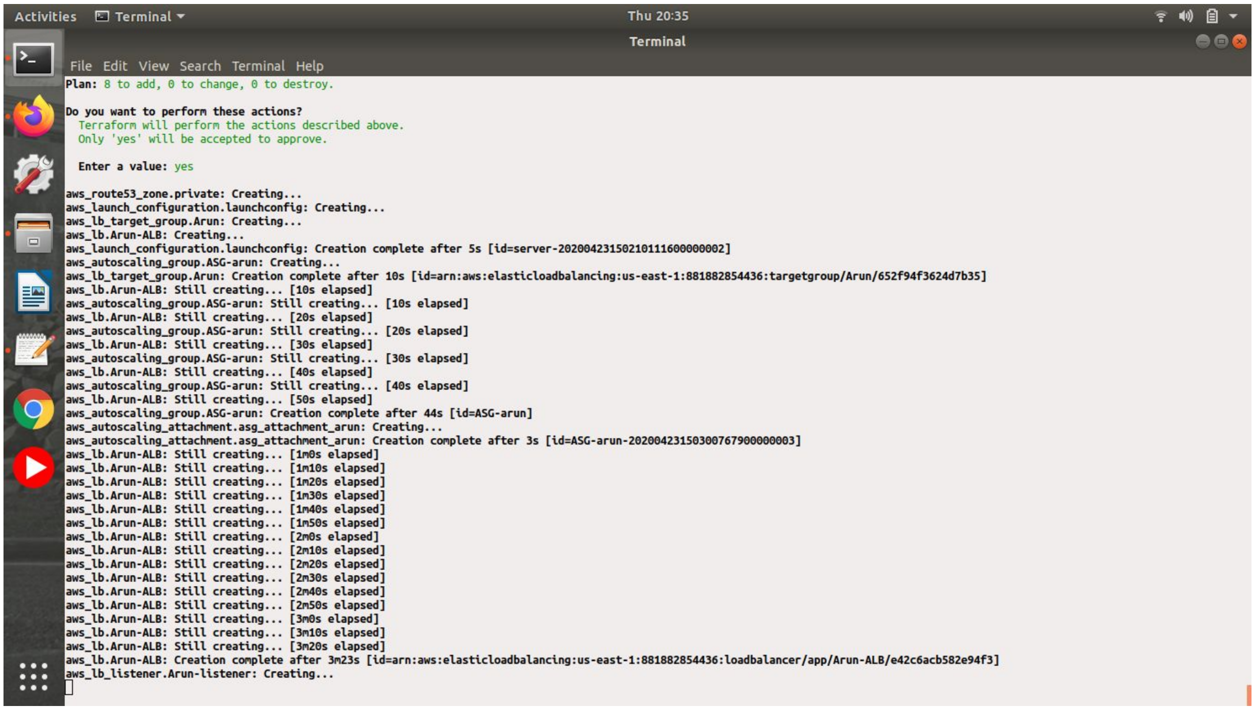
```
#!/bin/bash
sudo apt-get update -y
sudo apt-get install nginx -y

~
~
~
~
~
~
```

- Edit env.tfvars in which you will assign the variable values



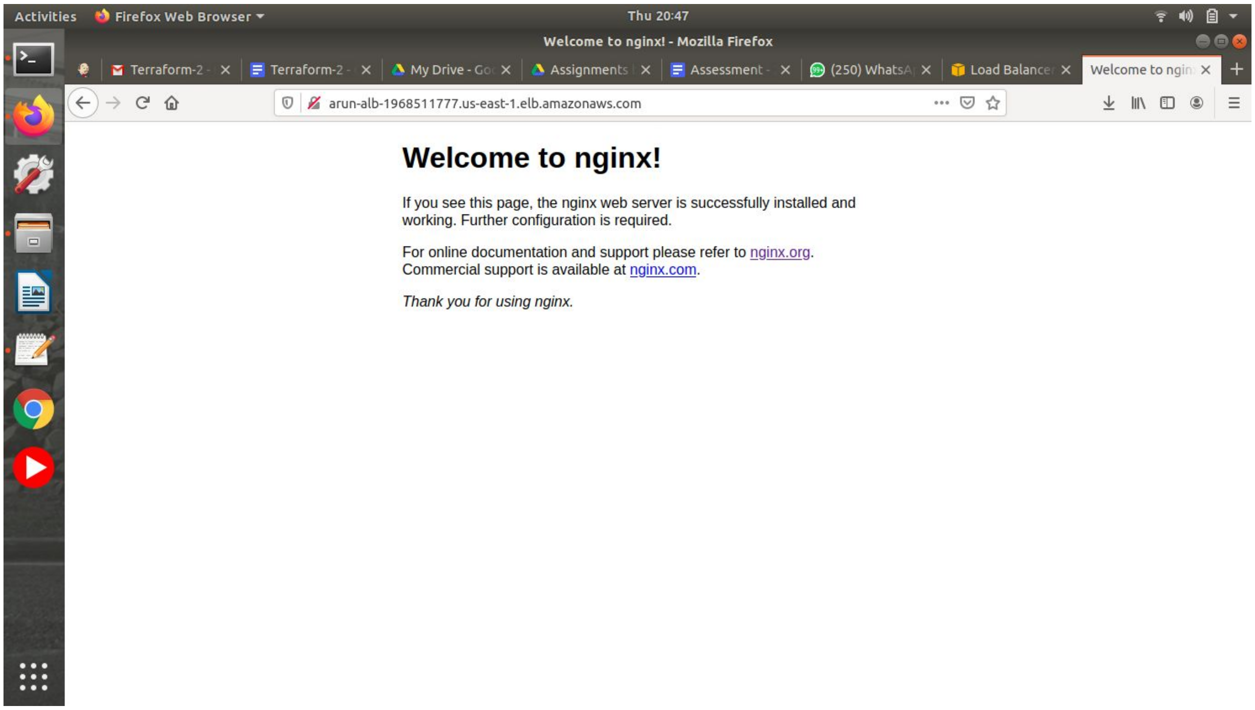
```
ami="ami-07ebfd5b3428b6f4d"
instance_type="t2.micro"
key="gargiaccount"
az="us-east-1a"
sg="sg-056781cc3416ca227"
asg="ASG-arun"
alb="Arun-ALB"
target_group="Arun"
private_zone="arun.com"
record_name="www.arun.com"


-- INSERT --                                                   15,1          Bot
```

- Terraform plan --var-file=”env.tfvars”-> terraform apply --var-file=”env.tfvars”



```
Plan: 8 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_route53_zone.private: Creating...
aws_launch_configuration.launchconfig: Creating...
aws_lb_target_group.Arun: Creating...
aws_lb.Arun-ALB: Creating...
aws_launch_configuration.launchconfig: Creation complete after 5s [id=server-20200423150210111600000002]
aws_autoscaling_group.ASG-arun: Creating...
aws_lb_target_group.Arun: Creation complete after 10s [id=arn:aws:elasticloadbalancing:us-east-1:881882854436:targetgroup/Arun/652f94f3624d7b35]
aws_lb.Arun-ALB: Still creating... [10s elapsed]
aws_autoscaling_group.ASG-arun: Still creating... [10s elapsed]
aws_lb.Arun-ALB: Still creating... [20s elapsed]
aws_autoscaling_group.ASG-arun: Still creating... [20s elapsed]
aws_lb.Arun-ALB: Still creating... [30s elapsed]
aws_autoscaling_group.ASG-arun: Still creating... [30s elapsed]
aws_lb.Arun-ALB: Still creating... [40s elapsed]
aws_autoscaling_group.ASG-arun: Still creating... [40s elapsed]
aws_lb.Arun-ALB: Still creating... [50s elapsed]
aws_autoscaling_group.ASG-arun: Creation complete after 44s [id=ASG-arun]
aws_autoscaling_attachment.asg_attachment_arun: Creating...
aws_autoscaling_attachment.asg_attachment_arun: Creation complete after 3s [id=ASG-arun-20200423150300767900000003]
aws_lb.Arun-ALB: Still creating... [1m0s elapsed]
aws_lb.Arun-ALB: Still creating... [1m10s elapsed]
aws_lb.Arun-ALB: Still creating... [1m20s elapsed]
aws_lb.Arun-ALB: Still creating... [1m30s elapsed]
aws_lb.Arun-ALB: Still creating... [1m40s elapsed]
aws_lb.Arun-ALB: Still creating... [1m50s elapsed]
aws_lb.Arun-ALB: Still creating... [2m0s elapsed]
aws_lb.Arun-ALB: Still creating... [2m10s elapsed]
aws_lb.Arun-ALB: Still creating... [2m20s elapsed]
aws_lb.Arun-ALB: Still creating... [2m30s elapsed]
aws_lb.Arun-ALB: Still creating... [2m40s elapsed]
aws_lb.Arun-ALB: Still creating... [2m50s elapsed]
aws_lb.Arun-ALB: Still creating... [3m0s elapsed]
aws_lb.Arun-ALB: Still creating... [3m10s elapsed]
aws_lb.Arun-ALB: Still creating... [3m20s elapsed]
aws_lb.Arun-ALB: Creation complete after 3m23s [id=arn:aws:elasticloadbalancing:us-east-1:881882854436:loadbalancer/app/Arun-ALB/e42c6acb582e94f3]
aws_lb_listener.Arun-listener: Creating...
```

- Hit on DNS of load balancer



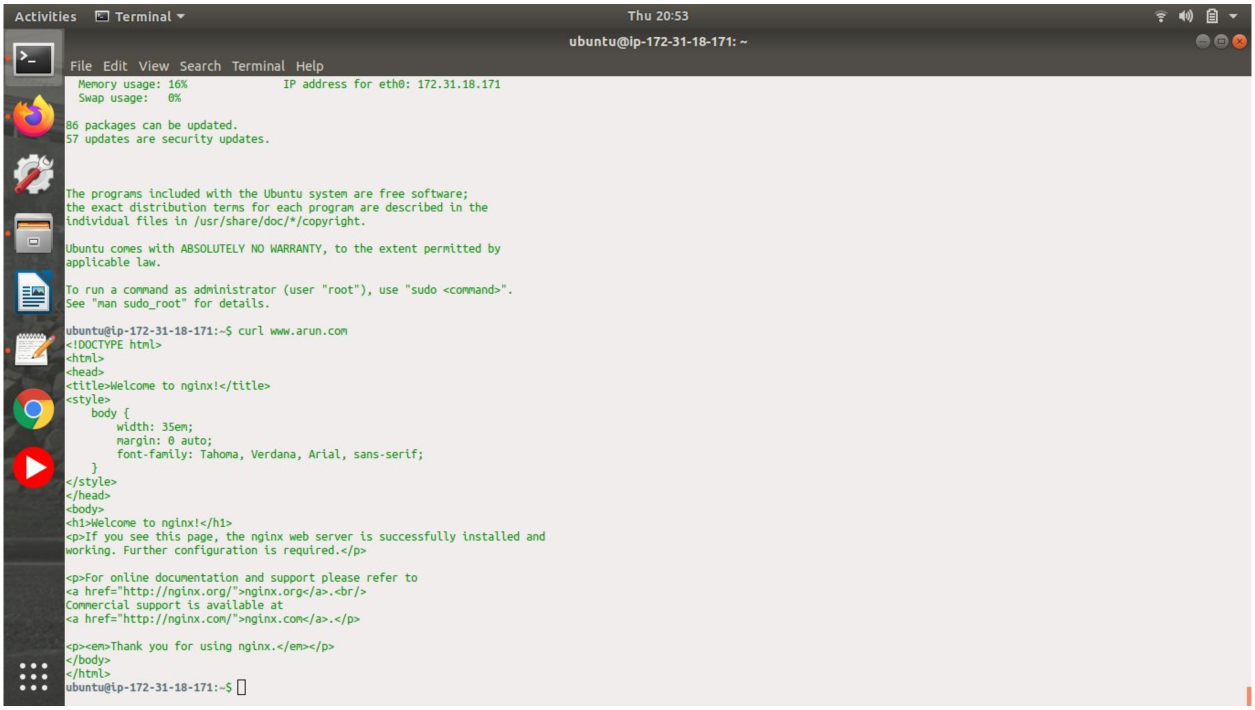
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Curl www.arun.com from instance

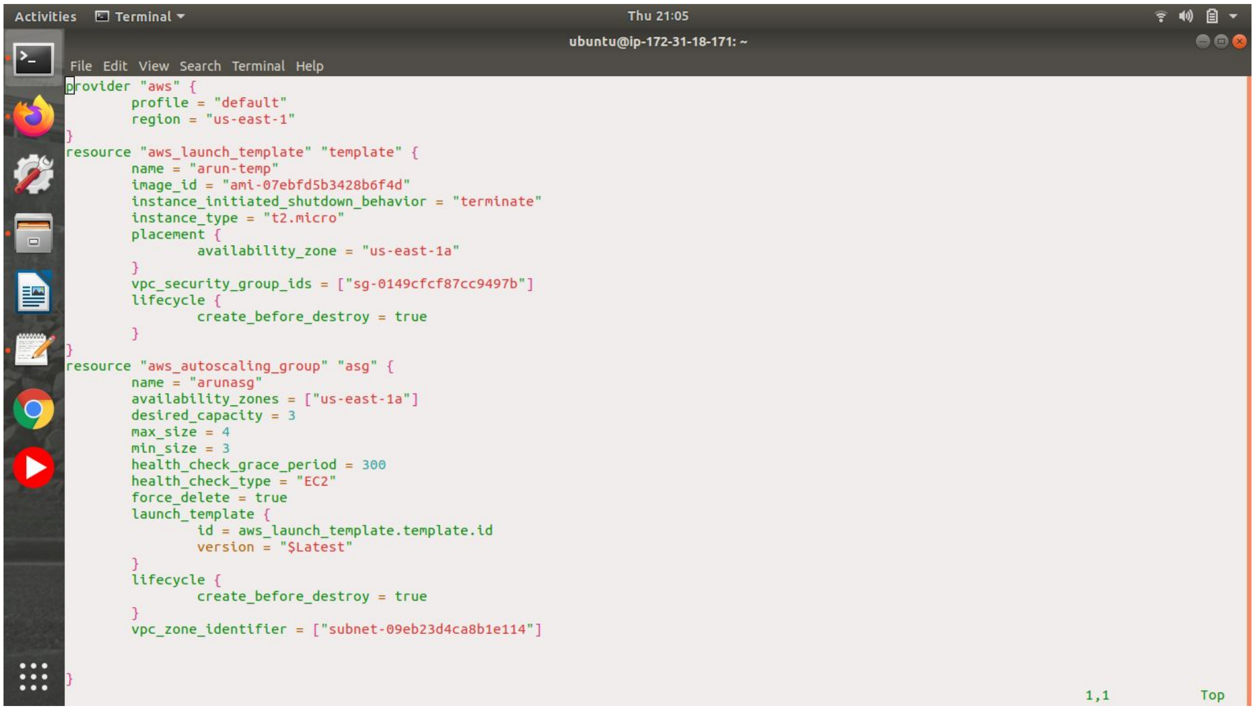


Activities
Terminal
Thu 20:53
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

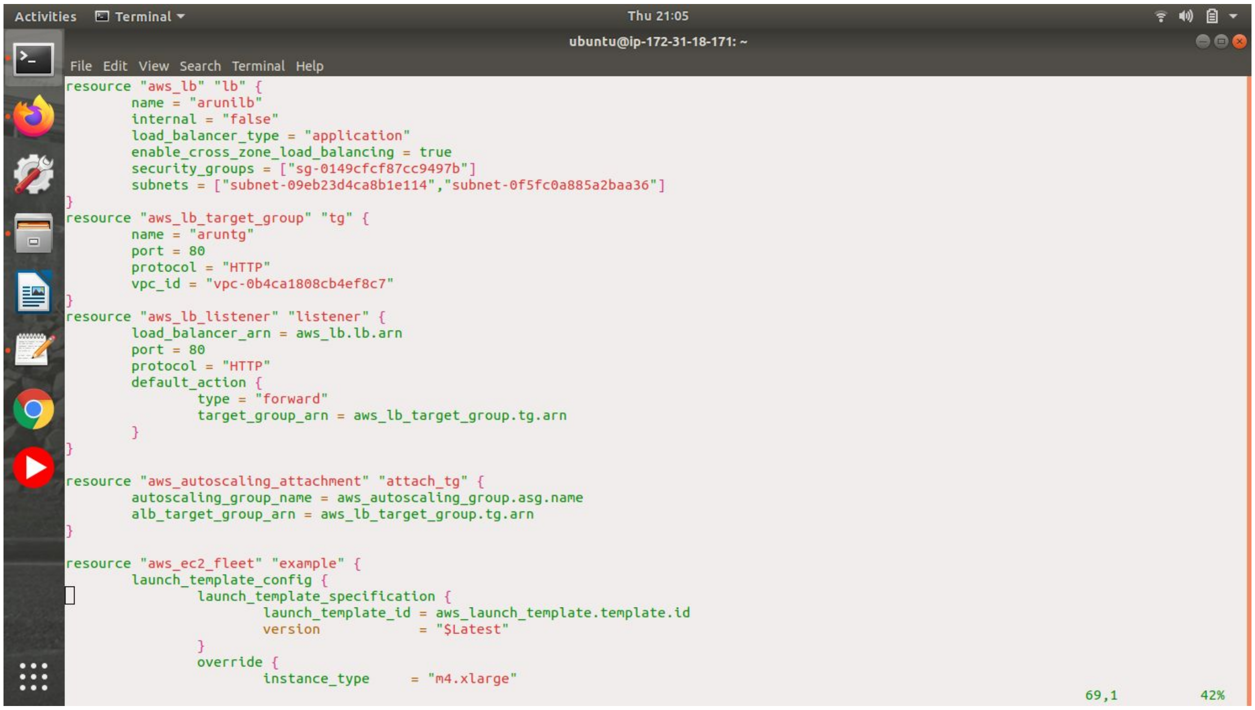
```
  Memory usage: 16%               IP address for eth0: 172.31.18.171
  Swap usage:   0%

86 packages can be updated.
57 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-18-171:~$ curl www.arun.com
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
ubuntu@ip-172-31-18-171:~$ 
```

5. Create ASG from Launch Template and use a mix of on demand and on spot instance type in the ASG. Instance Type for On Demand and Spot should be different.Enable Spot Feature to use multiple instance type if requested instance type is not available.

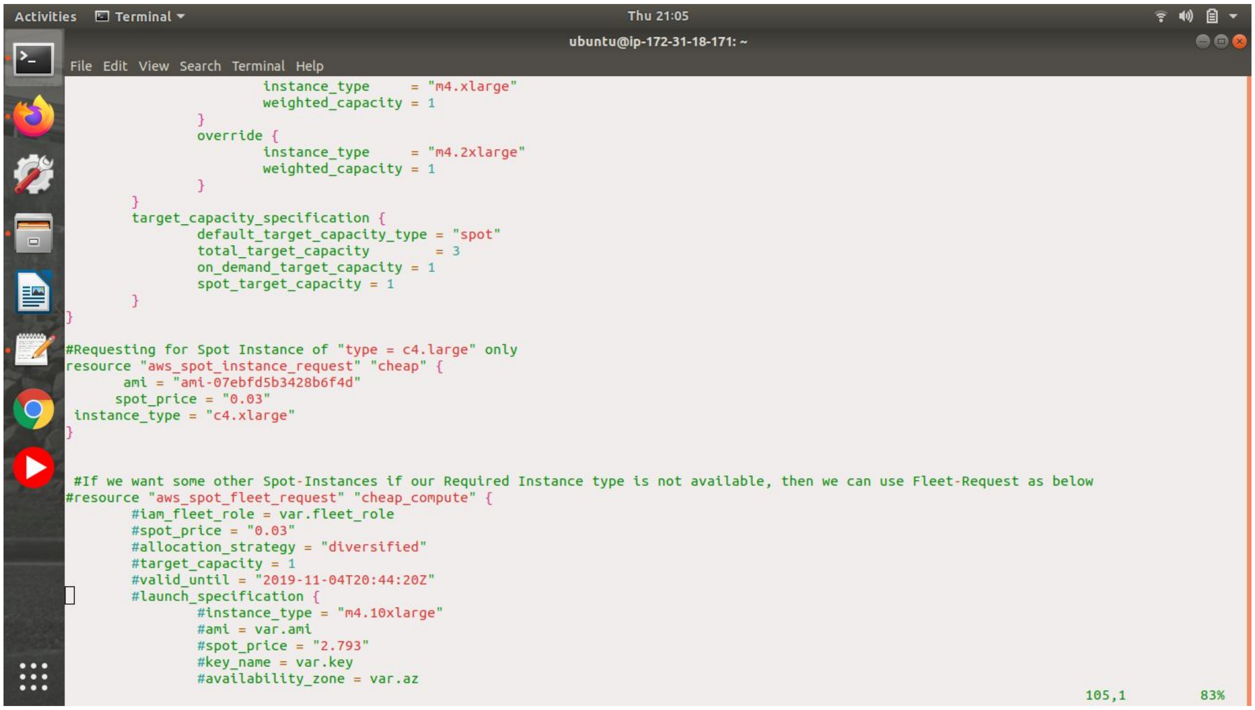
- main.tf file


Activities
Terminal
Thu 21:05
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

```
provider "aws" {
        profile = "default"
        region = "us-east-1"
}
resource "aws_launch_template" "template" {
        name = "arun-temp"
        image_id = "ami-07ebfd5b3428b6f4d"
        instance_initiated_shutdown_behavior = "terminate"
        instance_type = "t2.micro"
        placement {
                availability_zone = "us-east-1a"
        }
        vpc_security_group_ids = ["sg-0149cfcf87cc9497b"]
        lifecycle {
                create_before_destroy = true
        }
}
resource "aws_autoscaling_group" "asg" {
        name = "arunasg"
        availability_zones = ["us-east-1a"]
        desired_capacity = 3
        max_size = 4
        min_size = 3
        health_check_grace_period = 300
        health_check_type = "EC2"
        force_delete = true
        launch_template {
                id = aws_launch_template.template.id
                version = "$Latest"
        }
        lifecycle {
                create_before_destroy = true
        }
        vpc_zone_identifier = ["subnet-09eb23d4ca8b1e114"]


}
                                                                                                                                                  1,1           Top
```

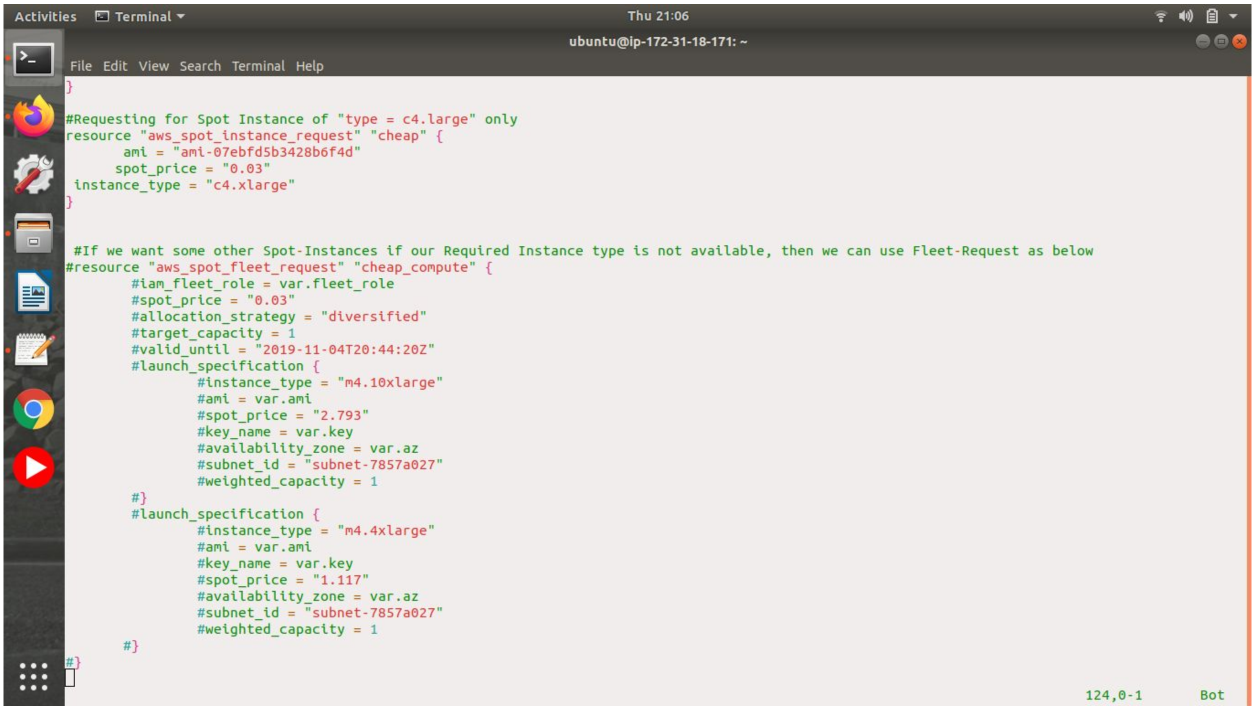

Activities
Terminal
Thu 21:05
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

```
resource "aws_lb" "lb" {
        name = "arunilb"
        internal = "false"
        load_balancer_type = "application"
        enable_cross_zone_load_balancing = true
        security_groups = ["sg-0149cfcf87cc9497b"]
        subnets = ["subnet-09eb23d4ca8b1e114","subnet-0f5fc0a885a2baa36"]
}
resource "aws_lb_target_group" "tg" {
        name = "aruntg"
        port = 80
        protocol = "HTTP"
        vpc_id = "vpc-0b4ca1808cb4ef8c7"
}
resource "aws_lb_listener" "listener" {
        load_balancer_arn = aws_lb.lb.arn
        port = 80
        protocol = "HTTP"
        default_action {
                type = "forward"
                target_group_arn = aws_lb_target_group.tg.arn
        }
}

resource "aws_autoscaling_attachment" "attach_tg" {
        autoscaling_group_name = aws_autoscaling_group.asg.name
        alb_target_group_arn = aws_lb_target_group.tg.arn
}

resource "aws_ec2_fleet" "example" {
        launch_template_config {

                launch_template_specification {
                        launch_template_id = aws_launch_template.template.id
                        version            = "$Latest"
                }
                override {
                        instance_type     = "m4.xlarge"
                                                                                                                                                  69,1          42%
```



Activities
Terminal
Thu 21:05
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

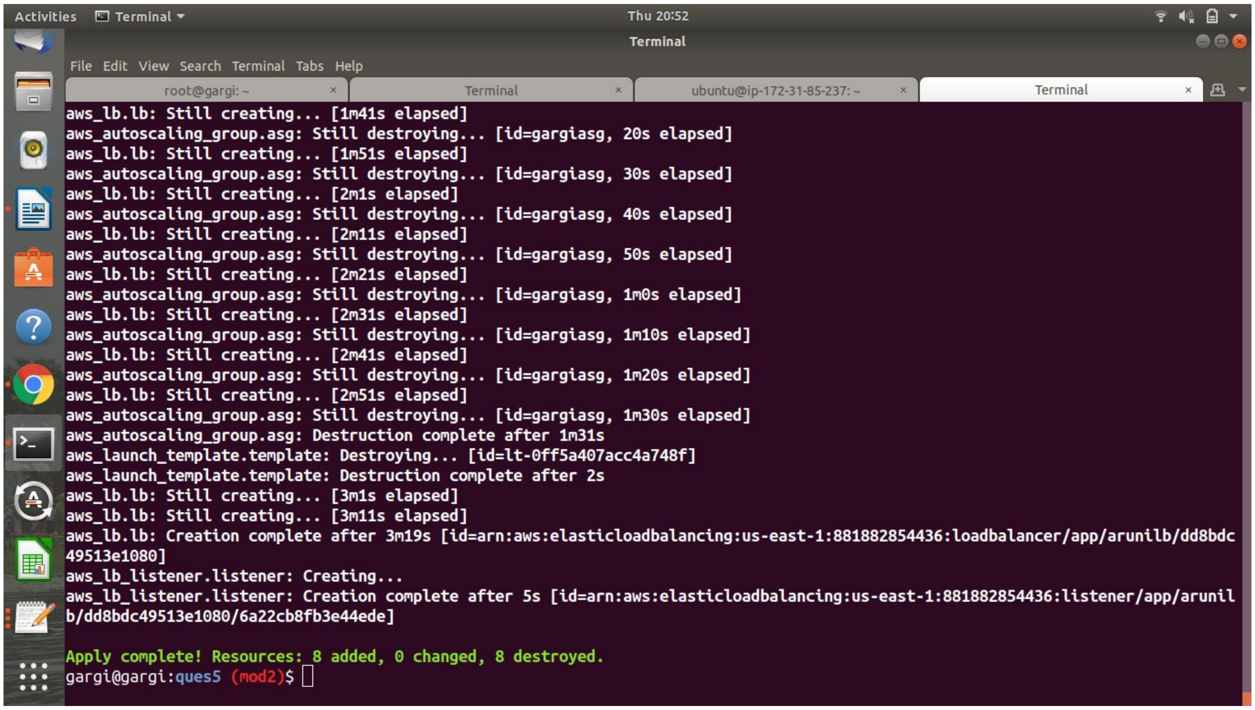
```
                        instance_type     = "m4.xlarge"
                        weighted_capacity = 1
                }
                override {
                        instance_type     = "m4.2xlarge"
                        weighted_capacity = 1
                }
        }
        target_capacity_specification {
                default_target_capacity_type = "spot"
                total_target_capacity        = 3
                on_demand_target_capacity = 1
                spot_target_capacity = 1
        }
}

#Requesting for Spot Instance of "type = c4.large" only
resource "aws_spot_instance_request" "cheap" {
       ami = "ami-07ebfd5b3428b6f4d"
      spot_price = "0.03"
 instance_type = "c4.xlarge"
}

 #If we want some other Spot-Instances if our Required Instance type is not available, then we can use Fleet-Request as below
#resource "aws_spot_fleet_request" "cheap_compute" {
        #iam_fleet_role = var.fleet_role
        #spot_price = "0.03"
        #allocation_strategy = "diversified"
        #target_capacity = 1
        #valid_until = "2019-11-04T20:44:20Z"
        #launch_specification {
                #instance_type = "m4.10xlarge"
                #ami = var.ami
                #spot_price = "2.793"
                #key_name = var.key
                #availability_zone = var.az
                                                                                                                                                105,1         83%
```



Activities
Terminal
Thu 21:06
ubuntu@ip-172-31-18-171: ~
File Edit View Search Terminal Help

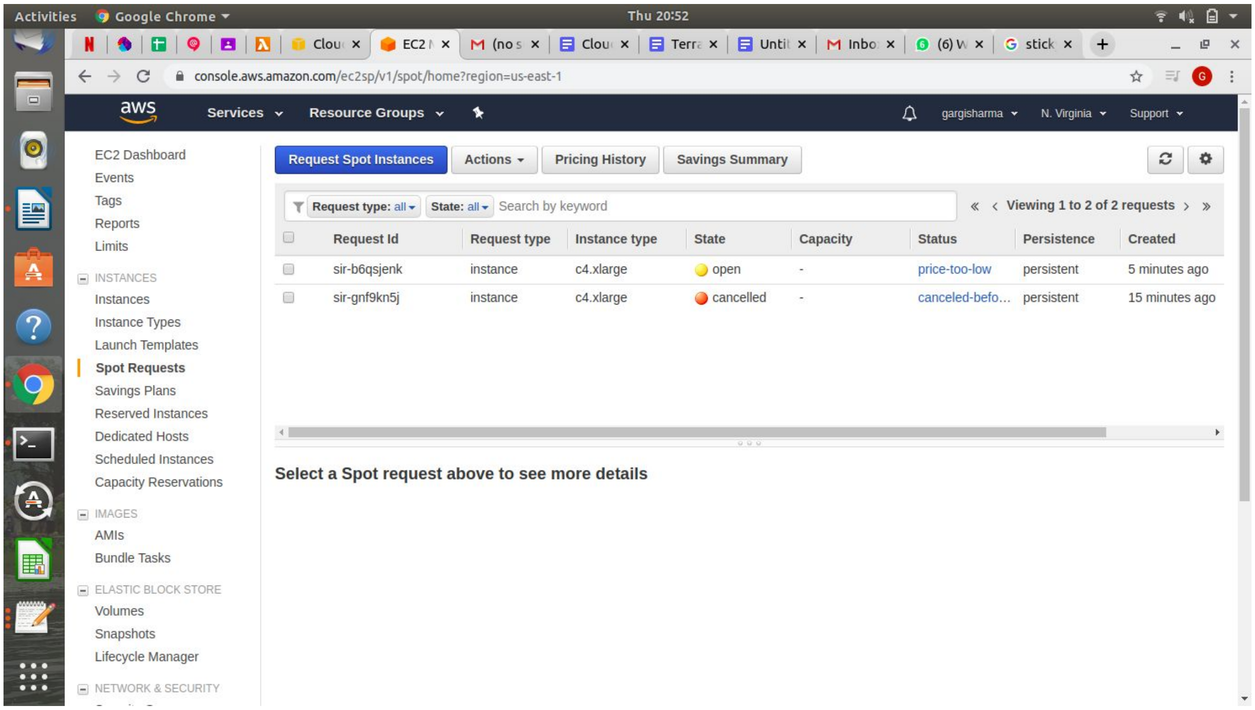
```
}

#Requesting for Spot Instance of "type = c4.large" only
resource "aws_spot_instance_request" "cheap" {
       ami = "ami-07ebfd5b3428b6f4d"
      spot_price = "0.03"
 instance_type = "c4.xlarge"
}

 #If we want some other Spot-Instances if our Required Instance type is not available, then we can use Fleet-Request as below
#resource "aws_spot_fleet_request" "cheap_compute" {
        #iam_fleet_role = var.fleet_role
        #spot_price = "0.03"
        #allocation_strategy = "diversified"
        #target_capacity = 1
        #valid_until = "2019-11-04T20:44:20Z"
        #launch_specification {
                #instance_type = "m4.10xlarge"
                #ami = var.ami
                #spot_price = "2.793"
                #key_name = var.key
                #availability_zone = var.az
                #subnet_id = "subnet-7857a027"
                #weighted_capacity = 1
        #}
        #launch_specification {
                #instance_type = "m4.4xlarge"
                #ami = var.ami
                #key_name = var.key
                #spot_price = "1.117"
                #availability_zone = var.az
                #subnet_id = "subnet-7857a027"
                #weighted_capacity = 1
       #}
#}

                                                                                                                                                124,0-1       Bot
```

- Terraform apply


Activities
Terminal
Thu 20:52
Terminal
File Edit View Search Terminal Tabs Help
root@gargi: ~
Terminal
ubuntu@ip-172-31-85-237: ~
Terminal

```
aws_lb.lb: Still creating... [1m41s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 20s elapsed]
aws_lb.lb: Still creating... [1m51s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 30s elapsed]
aws_lb.lb: Still creating... [2m1s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 40s elapsed]
aws_lb.lb: Still creating... [2m11s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 50s elapsed]
aws_lb.lb: Still creating... [2m21s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 1m0s elapsed]
aws_lb.lb: Still creating... [2m31s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 1m10s elapsed]
aws_lb.lb: Still creating... [2m41s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 1m20s elapsed]
aws_lb.lb: Still creating... [2m51s elapsed]
aws_autoscaling_group.asg: Still destroying... [id=gargiasg, 1m30s elapsed]
aws_autoscaling_group.asg: Destruction complete after 1m31s
aws_launch_template.template: Destroying... [id=lt-0ff5a407acc4a748f]
aws_launch_template.template: Destruction complete after 2s
aws_lb.lb: Still creating... [3m1s elapsed]
aws_lb.lb: Still creating... [3m11s elapsed]
aws_lb.lb: Creation complete after 3m19s [id=arn:aws:elasticloadbalancing:us-east-1:881882854436:loadbalancer/app/arunilb/dd8bdc
49513e1080]
aws_lb_listener.listener: Creating...
aws_lb_listener.listener: Creation complete after 5s [id=arn:aws:elasticloadbalancing:us-east-1:881882854436:listener/app/arunil
b/dd8bdc49513e1080/6a22cb8fb3e44ede]

Apply complete! Resources: 8 added, 0 changed, 8 destroyed.
gargi@gargi:ques5 (mod2)$ 
```

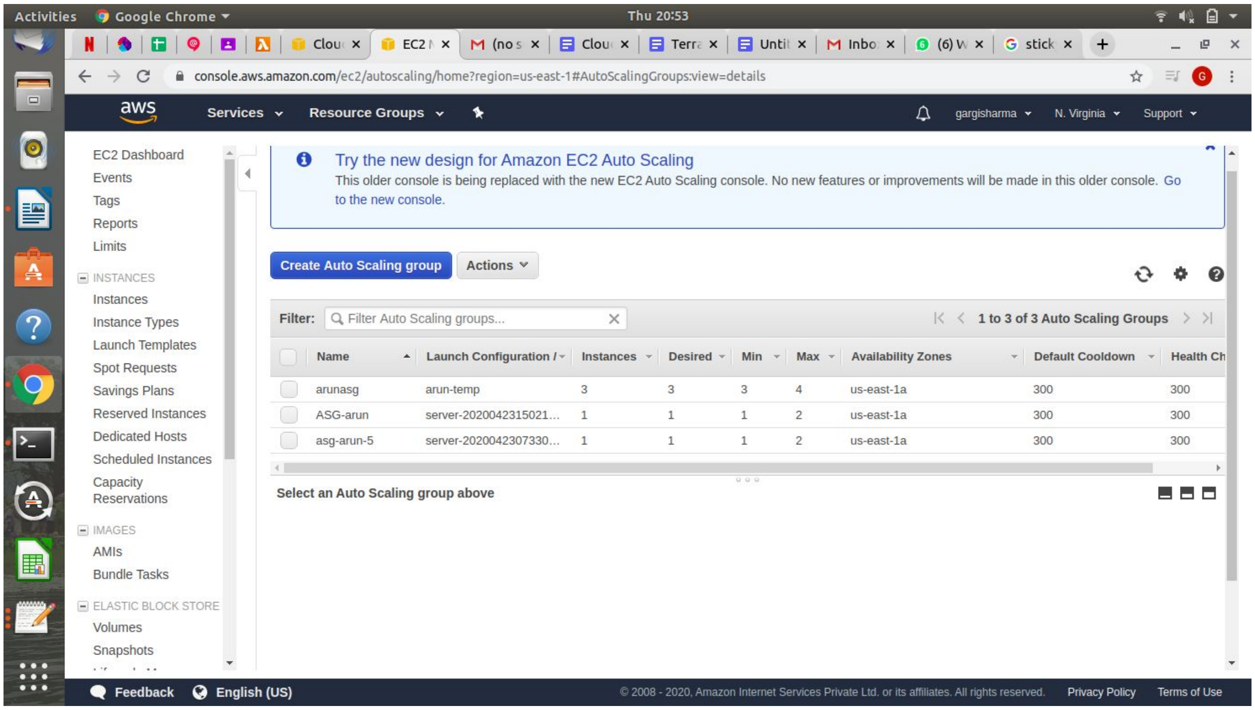
- Spot requests and spot instances



Viewing 1 to 2 of 2 requests

☐	Request Id	Request type	Instance type	State	Capacity	Status	Persistence	Created
☐	sir-b6qsjenk	instance	c4.xlarge	● open	-	price-too-low	persistent	5 minutes ago
☐	sir-gnf9kn5j	instance	c4.xlarge	● cancelled	-	canceled-befo…	persistent	15 minutes ago

Select a Spot request above to see more details



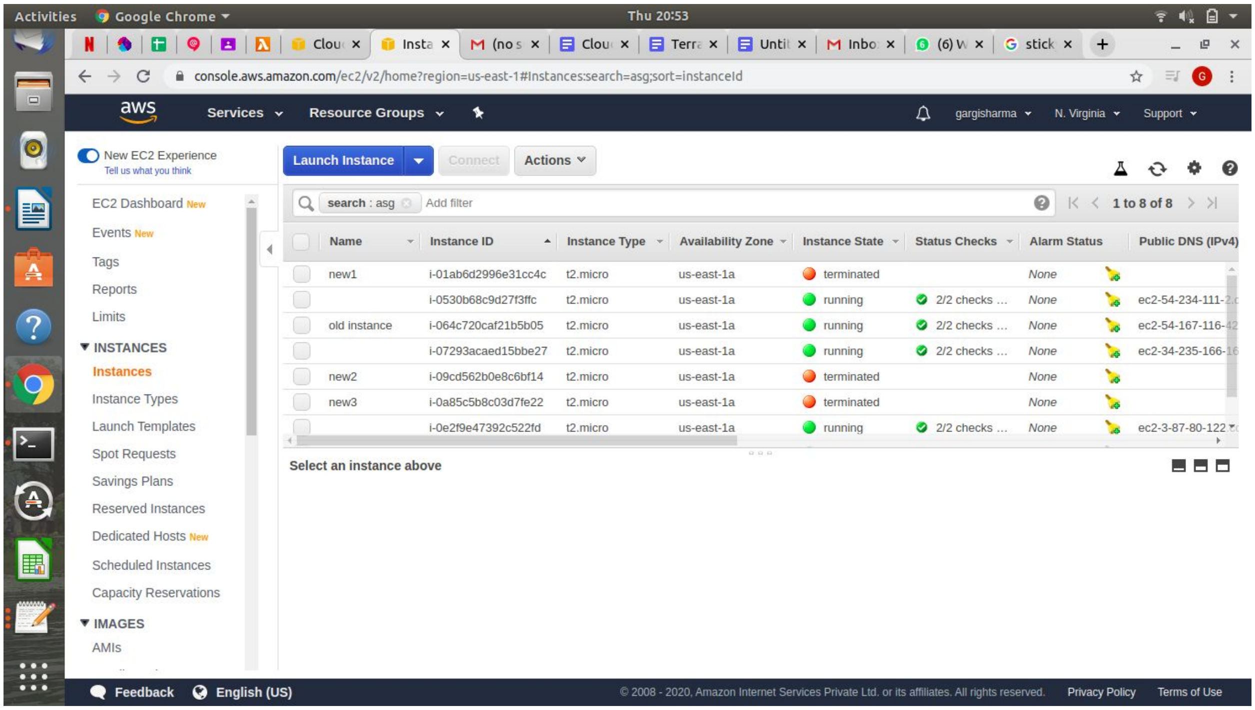
Try the new design for Amazon EC2 Auto Scaling

This older console is being replaced with the new EC2 Auto Scaling console. No new features or improvements will be made in this older console. Go to the new console.

Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Ch
arunasg	arun-temp	3	3	3	4	us-east-1a	300	300
ASG-arun	server-2020042315021...	1	1	1	2	us-east-1a	300	300
asg-arun-5	server-2020042307330...	1	1	1	2	us-east-1a	300	300

Select an Auto Scaling group above

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



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
new1	i-01ab6d2996e31cc4c	t2.micro	us-east-1a	● terminated		None	
	i-0530b68c9d27f3ffc	t2.micro	us-east-1a	● running	✓ 2/2 checks ...	None	ec2-54-234-111-2.c
old instance	i-064c720caf21b5b05	t2.micro	us-east-1a	● running	✓ 2/2 checks ...	None	ec2-54-167-116-42
	i-07293acaed15bbe27	t2.micro	us-east-1a	● running	✓ 2/2 checks ...	None	ec2-34-235-166-16
new2	i-09cd562b0e8c6bf14	t2.micro	us-east-1a	● terminated		None	
new3	i-0a85c5b8c03d7fe22	t2.micro	us-east-1a	● terminated		None	
	i-0e2f9e47392c522fd	t2.micro	us-east-1a	● running	✓ 2/2 checks ...	None	ec2-3-87-80-122

Select an instance above

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