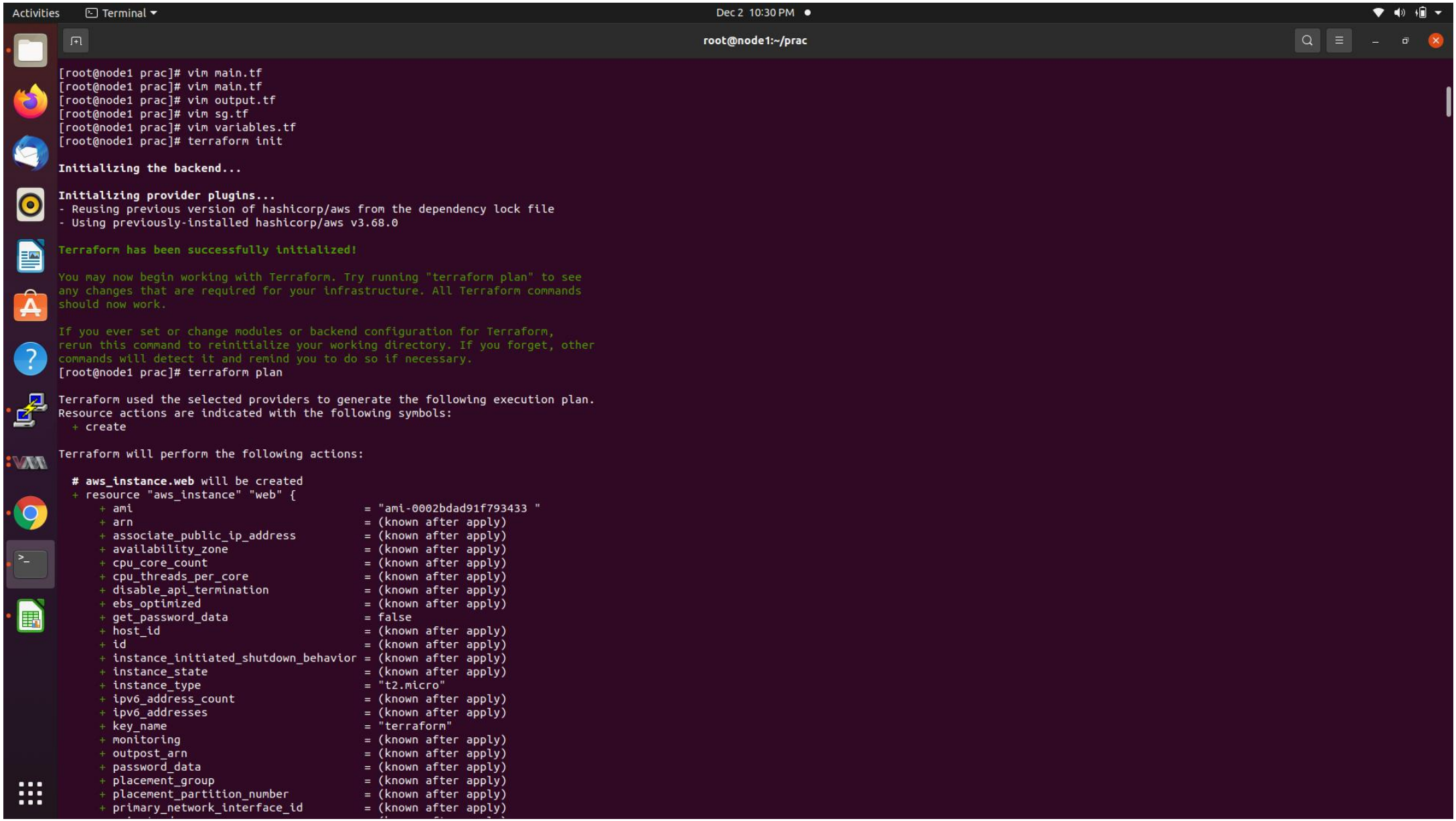
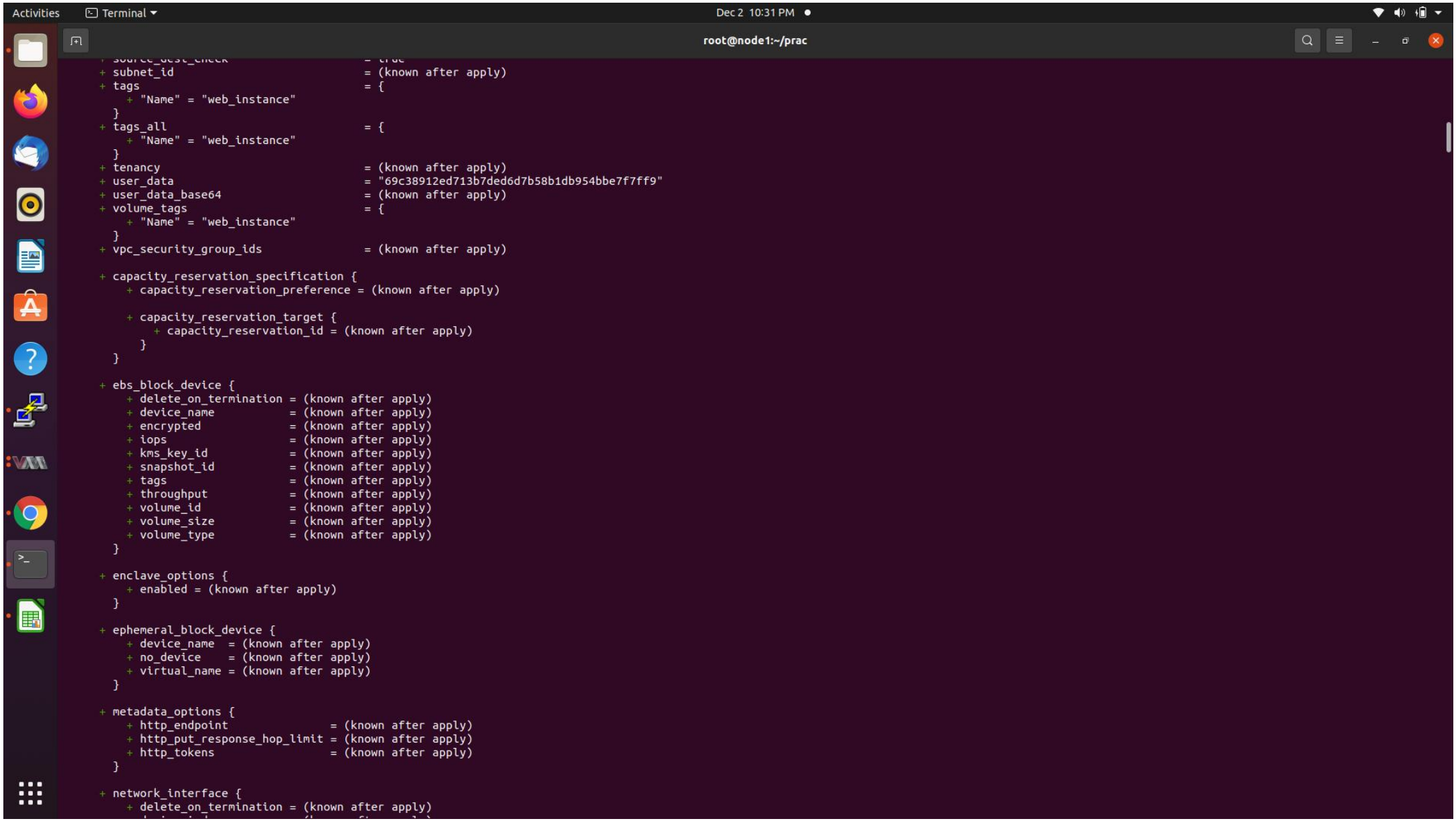
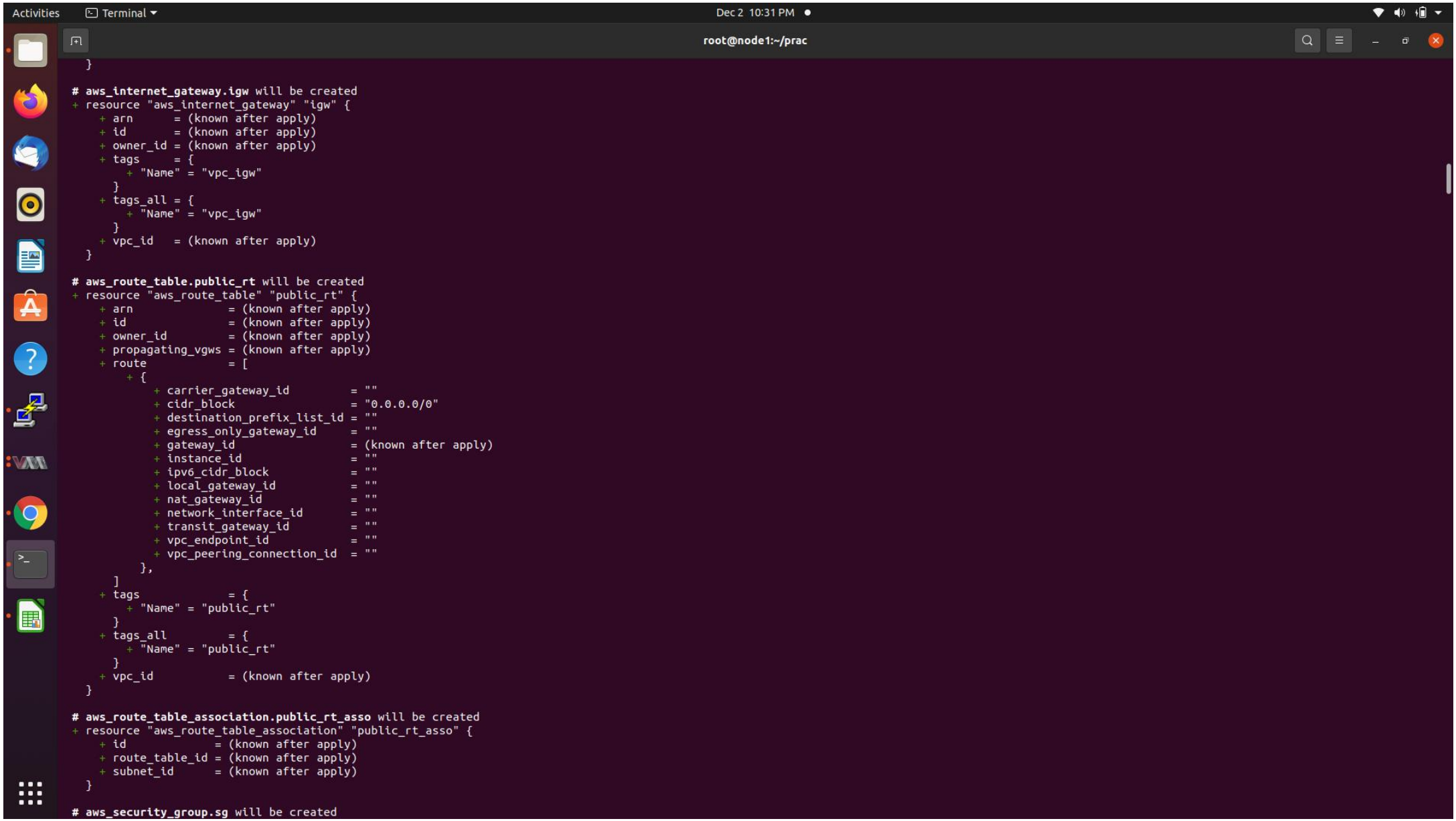
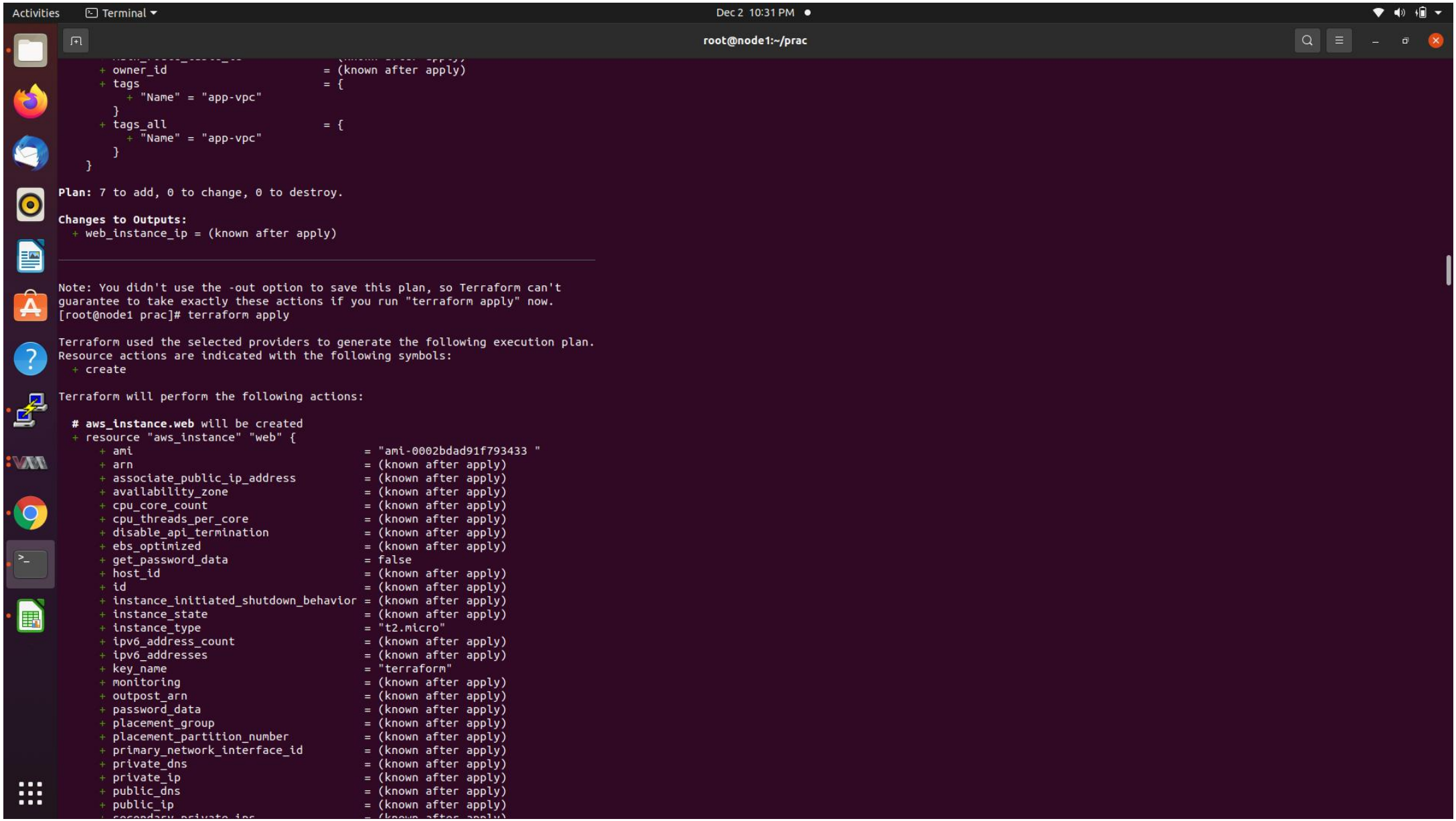
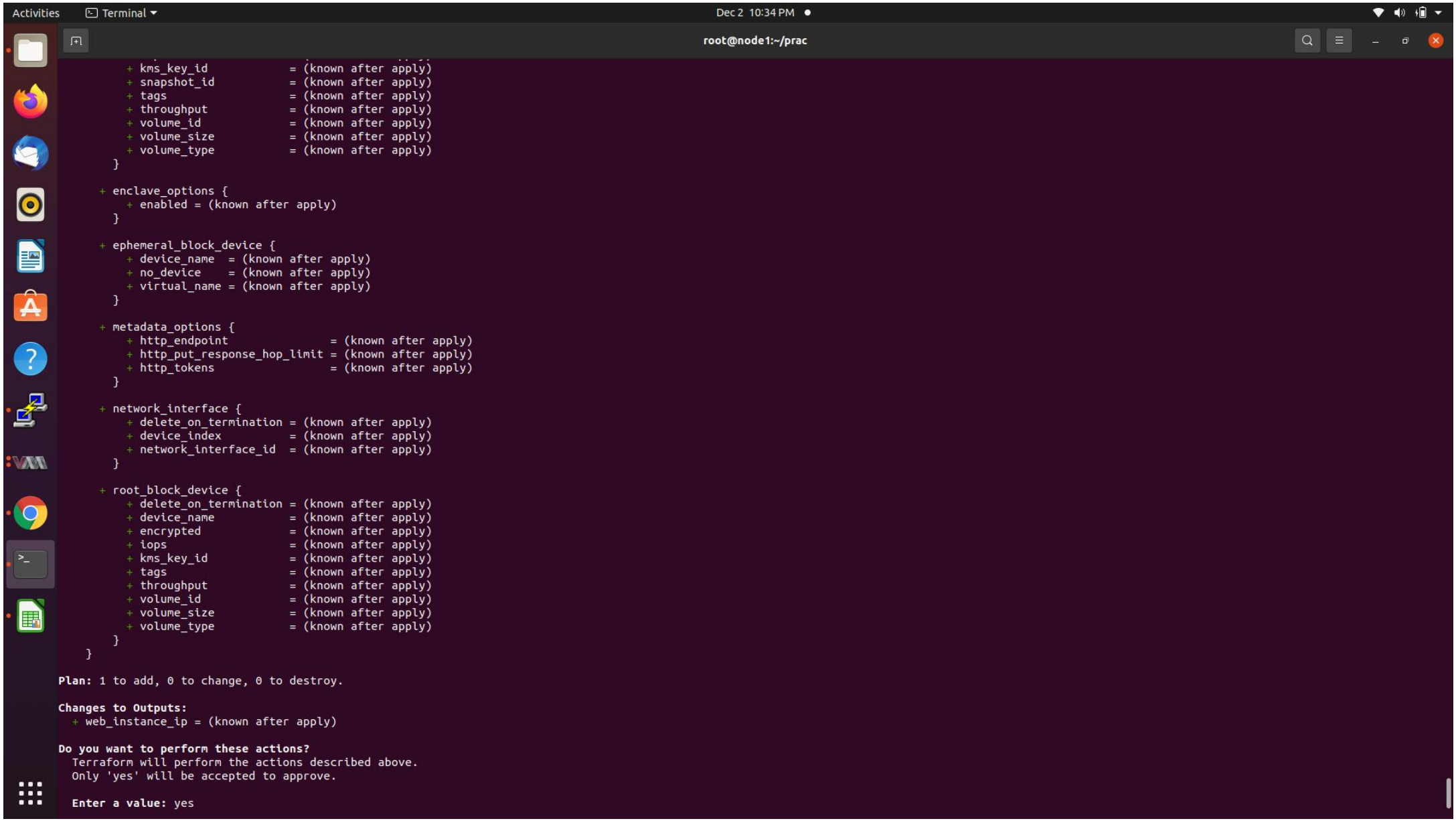
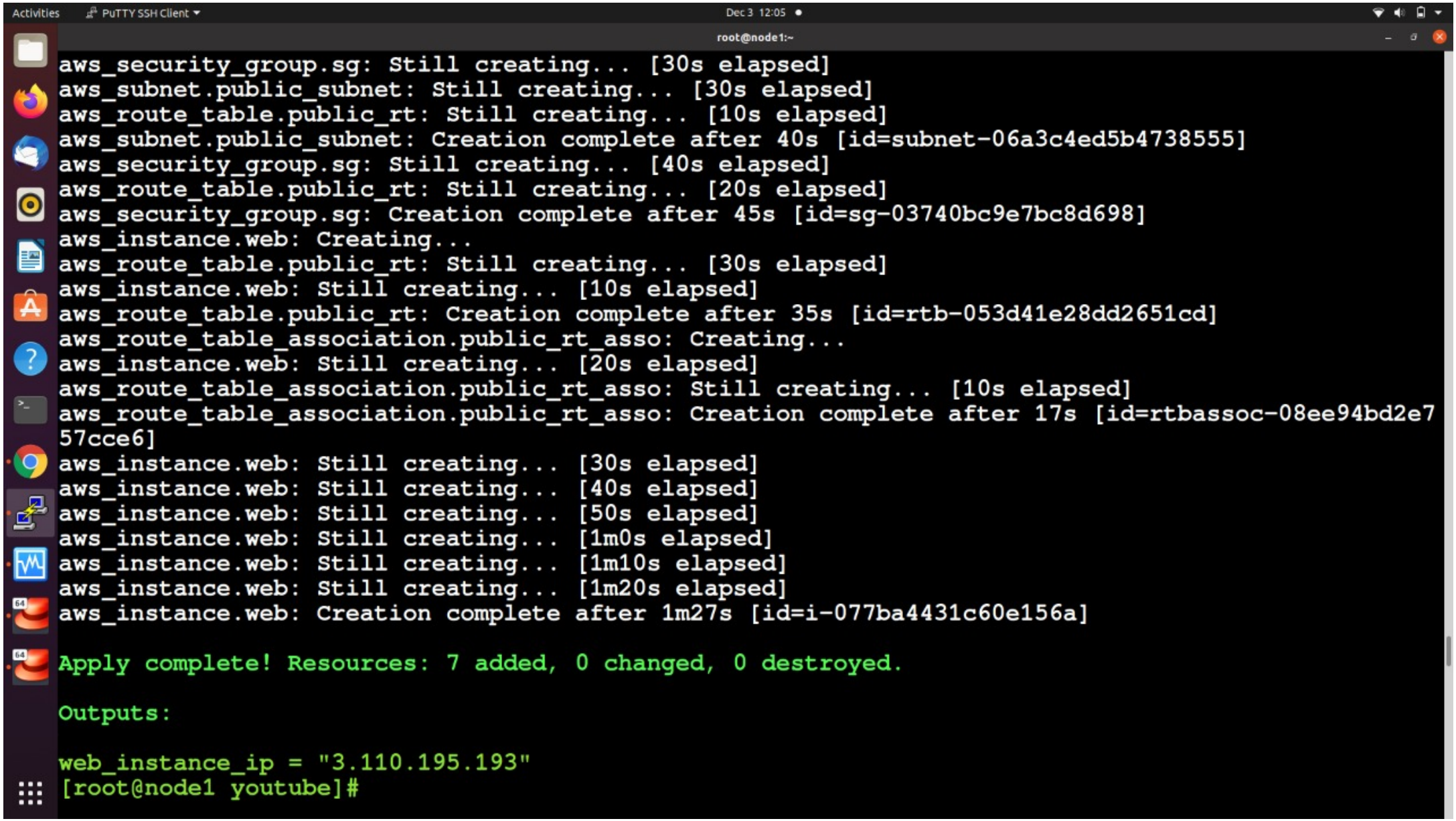

Activities
Terminal
Dec 2 10:30 PM
root@node1:~/prac
[root@node1 prac]# vim main.tf
[root@node1 prac]# vim main.tf
[root@node1 prac]# vim output.tf
[root@node1 prac]# vim sg.tf
[root@node1 prac]# vim variables.tf
[root@node1 prac]# terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v3.68.0
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.
[root@node1 prac]# terraform plan
Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:
+ create
Terraform will perform the following actions:
# aws_instance.web will be created
+ resource "aws_instance" "web" {
+ ami = "ami-0002bdad91f793433 "
+ arn = (known after apply)
+ associate_public_ip_address = (known after apply)
+ availability_zone = (known after apply)
+ cpu_core_count = (known after apply)
+ cpu_threads_per_core = (known after apply)
+ disable_api_termination = (known after apply)
+ ebs_optimized = (known after apply)
+ get_password_data = false
+ host_id = (known after apply)
+ id = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_state = (known after apply)
+ instance_type = "t2.micro"
+ ipv6_address_count = (known after apply)
+ ipv6_addresses = (known after apply)
+ key_name = "terraform"
+ monitoring = (known after apply)
+ outpost_arn = (known after apply)
+ password_data = (known after apply)
+ placement_group = (known after apply)
+ placement_partition_number = (known after apply)
+ primary_network_interface_id = (known after apply)

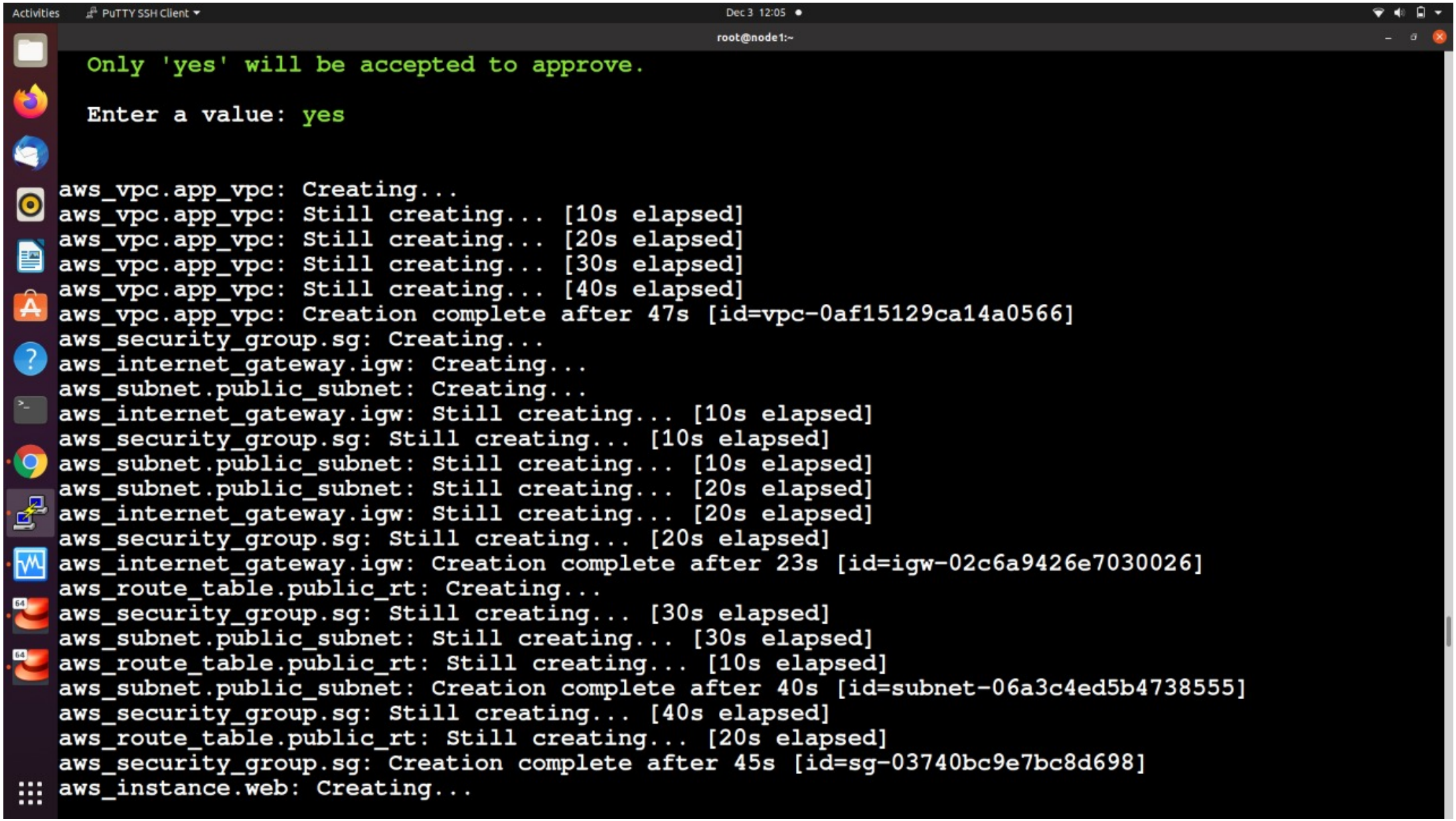

Activities
Terminal
Dec 2 10:31 PM
root@node1:~/prac
+ subnet_id = (known after apply)
+ tags = {
"Name" = "web_instance"
}
+ tags_all = {
"Name" = "web_instance"
}
+ tenancy = (known after apply)
+ user_data = "69c38912ed713b7ded6d7b58b1db954bbe7f7ff9"
+ user_data_base64 = (known after apply)
+ volume_tags = {
"Name" = "web_instance"
}
+ vpc_security_group_ids = (known after apply)
+ capacity_reservation_specification {
+ capacity_reservation_preference = (known after apply)
+ capacity_reservation_target {
+ capacity_reservation_id = (known after apply)
}
}
+ ebs_block_device {
+ delete_on_termination = (known after apply)
+ device_name = (known after apply)
+ encrypted = (known after apply)
+ iops = (known after apply)
+ kms_key_id = (known after apply)
+ snapshot_id = (known after apply)
+ tags = (known after apply)
+ throughput = (known after apply)
+ volume_id = (known after apply)
+ volume_size = (known after apply)
+ volume_type = (known after apply)
}
+ enclave_options {
+ enabled = (known after apply)
}
+ ephemeral_block_device {
+ device_name = (known after apply)
+ no_device = (known after apply)
+ virtual_name = (known after apply)
}
+ metadata_options {
+ http_endpoint = (known after apply)
+ http_put_response_hop_limit = (known after apply)
+ http_tokens = (known after apply)
}
+ network_interface {
+ delete_on_termination = (known after apply)


Activities
Terminal
Dec 2 10:31 PM
root@node1:~/prac
}
# aws_internet_gateway.igw will be created
+ resource "aws_internet_gateway" "igw" {
+ arn = (known after apply)
+ id = (known after apply)
+ owner_id = (known after apply)
+ tags = {
+ "Name" = "vpc_igw"
}
+ tags_all = {
+ "Name" = "vpc_igw"
}
+ vpc_id = (known after apply)
}
# aws_route_table.public_rt will be created
+ resource "aws_route_table" "public_rt" {
+ arn = (known after apply)
+ id = (known after apply)
+ owner_id = (known after apply)
+ propagating_vgws = (known after apply)
+ route = [
+ {
+ carrier_gateway_id = ""
+ cidr_block = "0.0.0.0/0"
+ destination_prefix_list_id = ""
+ egress_only_gateway_id = ""
+ gateway_id = (known after apply)
+ instance_id = ""
+ ipv6_cidr_block = ""
+ local_gateway_id = ""
+ nat_gateway_id = ""
+ network_interface_id = ""
+ transit_gateway_id = ""
+ vpc_endpoint_id = ""
+ vpc_peering_connection_id = ""
},
]
+ tags = {
+ "Name" = "public_rt"
}
+ tags_all = {
+ "Name" = "public_rt"
}
+ vpc_id = (known after apply)
}
# aws_route_table_association.public_rt_asso will be created
+ resource "aws_route_table_association" "public_rt_asso" {
+ id = (known after apply)
+ route_table_id = (known after apply)
+ subnet_id = (known after apply)
}
# aws_security_group.sg will be created


Activities
Terminal
Dec 2 10:31 PM
root@node1:~/prac
+ owner_id = (known after apply)
+ tags = {
+ "Name" = "app-vpc"
}
+ tags_all = {
+ "Name" = "app-vpc"
}
}
Plan: 7 to add, 0 to change, 0 to destroy.
Changes to Outputs:
+ web_instance_ip = (known after apply)
Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
[root@node1 prac]# terraform apply
Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:
+ create
Terraform will perform the following actions:
# aws_instance.web will be created
+ resource "aws_instance" "web" {
+ ami = "ami-0002bdad91f793433 "
+ arn = (known after apply)
+ associate_public_ip_address = (known after apply)
+ availability_zone = (known after apply)
+ cpu_core_count = (known after apply)
+ cpu_threads_per_core = (known after apply)
+ disable_api_termination = (known after apply)
+ ebs_optimized = (known after apply)
+ get_password_data = false
+ host_id = (known after apply)
+ id = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_state = (known after apply)
+ instance_type = "t2.micro"
+ ipv6_address_count = (known after apply)
+ ipv6_addresses = (known after apply)
+ key_name = "terraform"
+ monitoring = (known after apply)
+ outpost_arn = (known after apply)
+ password_data = (known after apply)
+ placement_group = (known after apply)
+ placement_partition_number = (known after apply)
+ primary_network_interface_id = (known after apply)
+ private_dns = (known after apply)
+ private_ip = (known after apply)
+ public_dns = (known after apply)
+ public_ip = (known after apply)


Activities
Terminal
Dec 2 10:34 PM
root@node1:~/prac
+ kms_key_id = (known after apply)
+ snapshot_id = (known after apply)
+ tags = (known after apply)
+ throughput = (known after apply)
+ volume_id = (known after apply)
+ volume_size = (known after apply)
+ volume_type = (known after apply)
}
+ enclave_options {
+ enabled = (known after apply)
}
+ ephemeral_block_device {
+ device_name = (known after apply)
+ no_device = (known after apply)
+ virtual_name = (known after apply)
}
+ metadata_options {
+ http_endpoint = (known after apply)
+ http_put_response_hop_limit = (known after apply)
+ http_tokens = (known after apply)
}
+ network_interface {
+ delete_on_termination = (known after apply)
+ device_index = (known after apply)
+ network_interface_id = (known after apply)
}
+ root_block_device {
+ delete_on_termination = (known after apply)
+ device_name = (known after apply)
+ encrypted = (known after apply)
+ iops = (known after apply)
+ kms_key_id = (known after apply)
+ tags = (known after apply)
+ throughput = (known after apply)
+ volume_id = (known after apply)
+ volume_size = (known after apply)
+ volume_type = (known after apply)
}
}
Plan: 1 to add, 0 to change, 0 to destroy.
Changes to Outputs:
+ web_instance_ip = (known after apply)
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


Activities
PuTTY SSH Client
Dec 3 12:05
root@node1:~
aws_security_group.sg: Still creating... [30s elapsed]
aws_subnet.public_subnet: Still creating... [30s elapsed]
aws_route_table.public_rt: Still creating... [10s elapsed]
aws_subnet.public_subnet: Creation complete after 40s [id=subnet-06a3c4ed5b4738555]
aws_security_group.sg: Still creating... [40s elapsed]
aws_route_table.public_rt: Still creating... [20s elapsed]
aws_security_group.sg: Creation complete after 45s [id=sg-03740bc9e7bc8d698]
aws_instance.web: Creating...
aws_route_table.public_rt: Still creating... [30s elapsed]
aws_instance.web: Still creating... [10s elapsed]
aws_route_table.public_rt: Creation complete after 35s [id=rtb-053d41e28dd2651cd]
aws_route_table_association.public_rt_asso: Creating...
aws_instance.web: Still creating... [20s elapsed]
aws_route_table_association.public_rt_asso: Still creating... [10s elapsed]
aws_route_table_association.public_rt_asso: Creation complete after 17s [id=rtbassoc-08ee94bd2e7
57cce6]
aws_instance.web: Still creating... [30s elapsed]
aws_instance.web: Still creating... [40s elapsed]
aws_instance.web: Still creating... [50s elapsed]
aws_instance.web: Still creating... [1m0s elapsed]
aws_instance.web: Still creating... [1m10s elapsed]
aws_instance.web: Still creating... [1m20s elapsed]
aws_instance.web: Creation complete after 1m27s [id=i-077ba4431c60e156a]
Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
Outputs:
web_instance_ip = "3.110.195.193"
[root@node1 youtube]#


Activities
PuTTY SSH Client
Dec 3 12:05
root@node1:~
Only 'yes' will be accepted to approve.
Enter a value: yes
aws_vpc.app_vpc: Creating...
aws_vpc.app_vpc: Still creating... [10s elapsed]
aws_vpc.app_vpc: Still creating... [20s elapsed]
aws_vpc.app_vpc: Still creating... [30s elapsed]
aws_vpc.app_vpc: Still creating... [40s elapsed]
aws_vpc.app_vpc: Creation complete after 47s [id=vpc-0af15129ca14a0566]
aws_security_group.sg: Creating...
aws_internet_gateway.igw: Creating...
aws_subnet.public_subnet: Creating...
aws_internet_gateway.igw: Still creating... [10s elapsed]
aws_security_group.sg: Still creating... [10s elapsed]
aws_subnet.public_subnet: Still creating... [10s elapsed]
aws_subnet.public_subnet: Still creating... [20s elapsed]
aws_internet_gateway.igw: Still creating... [20s elapsed]
aws_security_group.sg: Still creating... [20s elapsed]
aws_internet_gateway.igw: Creation complete after 23s [id=igw-02c6a9426e7030026]
aws_route_table.public_rt: Creating...
aws_security_group.sg: Still creating... [30s elapsed]
aws_subnet.public_subnet: Still creating... [30s elapsed]
aws_route_table.public_rt: Still creating... [10s elapsed]
aws_subnet.public_subnet: Creation complete after 40s [id=subnet-06a3c4ed5b4738555]
aws_security_group.sg: Still creating... [40s elapsed]
aws_route_table.public_rt: Still creating... [20s elapsed]
aws_security_group.sg: Creation complete after 45s [id=sg-03740bc9e7bc8d698]
aws_instance.web: Creating...