Output of my repo code

1. validate of the code

```
admin1@system:~/Downloads/cloudposse_tf/terraform-aws-vpc-peering-multi-account/examples/complete$ terraform validate
Success! The configuration is valid.
```

2. Output of the code

```
Apply complete! Resources: 34 added, 0 changed, 0 destroyed.

Outputs:

network_firewall_arn = "arn:aws:network-firewall:us-east-2:640262166032:firewall/eg-test-network-firewall"
network_firewall_name = "eg-test-network-firewall"
network_firewall_policy_arn = "arn:aws:network-firewall:us-east-2:640262166032:firewall-policy/eg-test-network-firewall"
network_firewall_policy_name = "eg-test-network-firewall"
network_firewall_status = tolist([
  {
    "sync_states" = toset([
      {
        "attachment" = tolist([
          {
            "endpoint_id" = "vpce-031fe22fab5cf0a8d"
            "subnet_id" = "subnet-0b2787a468d7b4423"
          },
        ])
        "availability_zone" = "us-east-2b"
      },
      {
        "attachment" = tolist([
          {
            "endpoint_id" = "vpce-0965a880b693bd3b5"
            "subnet_id" = "subnet-0dd629e8bf6b42bfb"
          },
        ])
        "availability_zone" = "us-east-2a"
      },
    ])
  },
])
network_firewall_update_token = "3dcb4d5b-2ac3-42b5-93a7-15a8caf6ca17"
private_subnet_cidrs = tolist([
  "172.19.0.0/19",
  "172.19.32.0/19",
])
public_subnet_cidrs = tolist([
  "172.19.96.0/19",
  "172.19.128.0/19",
])
vpc_cidr = "172.19.0.0/16"
```

3. list of state file created

```
admin1@system:~/Downloads/cloudposse_tf/terraform-aws-network-firewall/examples/complete$ terraform state list
module.network_firewall.aws_networkfirewall_firewall.default[0]
module.network_firewall.aws_networkfirewall_firewall_policy.default[0]
module.network_firewall.aws_networkfirewall_logging_configuration.default["0"]
module.network_firewall.aws_networkfirewall_rule_group.default["stateful-inspection-for-blocking-packets-from-going-to-destination"]
module.s3_log_storage.data.aws_caller_identity.current[0]
module.s3_log_storage.data.aws_partition.current[0]
module.subnets.data.aws_availability_zones.default[0]
module.subnets.aws_network_acl.private[0]
module.subnets.aws_network_acl.public[0]
module.subnets.aws_network_acl_rule.private4_egress[0]
module.subnets.aws_network_acl_rule.private4_ingress[0]
module.subnets.aws_network_acl_rule.public4_egress[0]
module.subnets.aws_network_acl_rule.public4_ingress[0]
module.subnets.aws_route.public[0]
module.subnets.aws_route_table.private[0]
module.subnets.aws_route_table.private[1]
module.subnets.aws_route_table.public[0]
module.subnets.aws_route_table_association.private[0]
module.subnets.aws_route_table_association.private[1]
module.subnets.aws_route_table_association.public[0]
module.subnets.aws_route_table_association.public[1]
module.subnets.aws_subnet.private[0]
module.subnets.aws_subnet.private[1]
module.subnets.aws_subnet.public[0]
module.subnets.aws_subnet.public[1]
module.vpc.aws_default_security_group.default[0]
module.vpc.aws_internet_gateway.default[0]
module.vpc.aws_vpc.default[0]
module.s3_log_storage.module.aws_s3_bucket.data.aws_canonical_user_id.default[0]
module.s3_log_storage.module.aws_s3_bucket.data.aws_iam_policy_document.aggregated_policy[0]
module.s3_log_storage.module.aws_s3_bucket.data.aws_iam_policy_document.bucket_policy[0]
module.s3_log_storage.module.aws_s3_bucket.data.aws_partition.current[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_acl.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_lifecycle_configuration.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_ownership_controls.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_policy.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_public_access_block.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_server_side_encryption_configuration.default[0]
module.s3_log_storage.module.aws_s3_bucket.aws_s3_bucket_versioning.default[0]
module.s3_log_storage.module.aws_s3_bucket.time_sleep.wait_for_aws_s3_bucket_settings[0]
module.subnets.module.utils.data.aws_regions.default
module.subnets.module.utils.data.aws_regions.not_opted_in
module.subnets.module.utils.data.aws_regions.opted_in
```