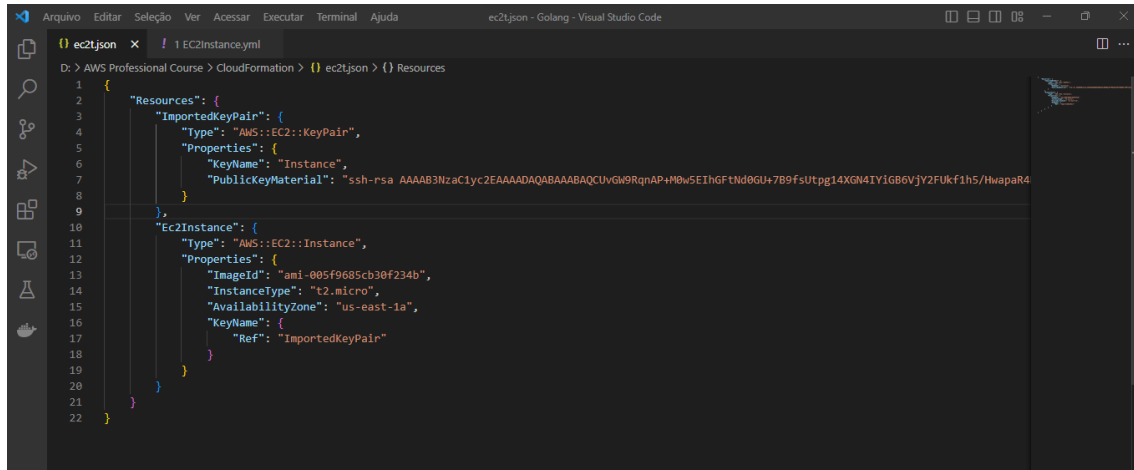


PoC – Segurança no uso de EC2

Está PoC teve como objetivo demonstrar como utilizar uma instância EC2 com segurança para fazer chamadas API para um Bucket S3 criado, com o intuito de armazenar arquivos, fotos ou vídeos que a instância poderia receber caso fosse uma aplicação Web.

Template para iniciar a instância, efetuado via IaC (infraestrutura como código)

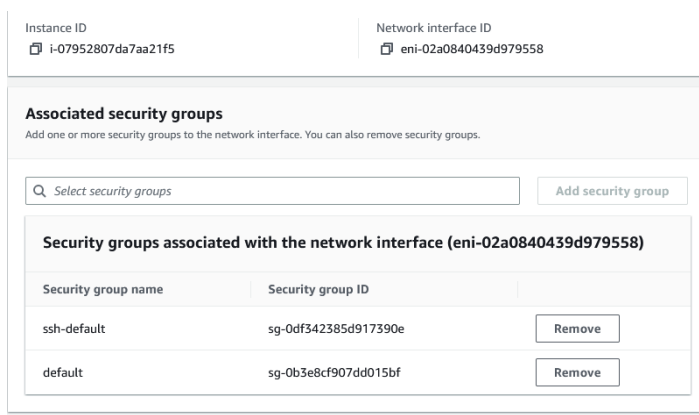


```
1 {
2   "Resources": {
3     "ImportedKeyPair": {
4       "Type": "AWS::EC2::KeyPair",
5       "Properties": {
6         "KeyName": "Instance",
7         "PublicKeyMaterial": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCUvGw9RqnAP+M0w5EIhGFTNd0GU+7B9fsUtpg14XGN4IY16B6VjY2Fukf1h5/HwapaR4
8       }
9     },
10    "Ec2Instance": {
11      "Type": "AWS::EC2::Instance",
12      "Properties": {
13        "ImageId": "ami-005f9685cb30f234b",
14        "InstanceType": "t2.micro",
15        "AvailabilityZone": "us-east-1a",
16        "KeyName": {
17          "Ref": "ImportedKeyPair"
18        }
19      }
20    }
21  }
22 }
```

O template foi produzido em Json e com ele é possível de maneira rápida lançar recursos via CloudFormation que será tema de outra PoC dos meus estudos, vamos direto ao ponto. Com a instância criada, como conseguiremos conectar em nosso Bucket S3, de maneira segura?

1) Primeiramente será efetuado a PoC sem boas práticas de segurança e posteriormente, faremos de maneira segura, lembrando que todo o Stack deverá ser excluído para evitar futuras cobranças.

Passo 1: Adicione a regra de security group para que possamos conectar na instância, siga os seguintes passo: Clique em Action > Security > Change Security Group



Note que já estamos com a regra adiciona, mas para adicionar clique no botão de select e posteriormente em add security group.

Feito isso vamos a conexão da nossa instância.

Passo 2: Com o security group adicionado, abra o Putty para efetuarmos a conexão SSH na instância.

```
ec2-user@ip-172-31-0-62:~$  
Using username "ec2-user".  
Authenticating with public key "Instance"  
  
 _ _ _ _ _  
 _ | ( _ _ | _ )  
 _ | \ _ _ | _ _ |  
 _ _ _ _ _ Amazon Linux 2 AMI  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-0-62 ~]$
```

Observação: Nesta PoC, foi criado um usuário genérico na conta AWS, que possui as seguintes Manage Policy da AWS S3ReadOnly e EC2ReadOnly

Após criar o usuário e digitarmos o comando `aws s3 ls`, identificamos o seguinte cenário:

```
 _ _ _ _ _  
 _ | ( _ _ | _ )  
 _ | \ _ _ | _ _ |  
 _ _ _ _ _ Amazon Linux 2 AMI  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-0-62 ~]$ aws s3 ls  
Unable to locate credentials. You can configure credentials by running "aws configure".  
[ec2-user@ip-172-31-0-62 ~]$
```

Este ponto é o crucial para a segurança da sua instância, temos duas opções adicionarmos a Access Key e a Secret Key criada ou adicionamos a role? Seguindo o padrão deste capítulo adicionarei os dados localmente na instância, com o comando `aws configure`

```
 _ _ _ _ _  
 _ | ( _ _ | _ )  
 _ | \ _ _ | _ _ |  
 _ _ _ _ _ Amazon Linux 2 AMI  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-0-62 ~]$ aws configure  
AWS Access Key ID [None]: AKIAUH6BGWW4M2C4EKAC  
AWS Secret Access Key [None]: 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  
Default region name [None]:  
Default output format [None]:  
[ec2-user@ip-172-31-0-62 ~]$ aws configure
```

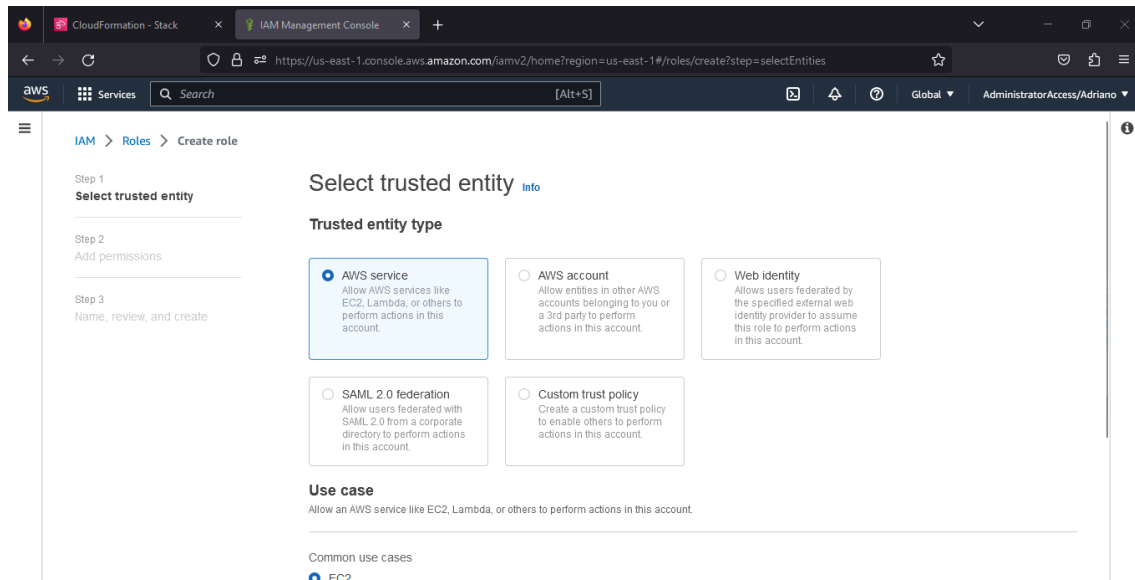
Digitaremos novamente o mesmo comando `aws s3 ls`, e veja o resultado

```
 _ _ _ _ _  
 _ | ( _ _ | _ )  
 _ | \ _ _ | _ _ |  
 _ _ _ _ _ Amazon Linux 2 AMI  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-0-62 ~]$ aws configure  
AWS Access Key ID [None]: AKIAUH6BGWW4M2C4EKAC  
AWS Secret Access Key [None]: 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  
Default region name [None]:  
Default output format [None]:  
[ec2-user@ip-172-31-0-62 ~]$ aws s3 ls  
2023-03-04 14:14:51 archawsadsa  
2023-03-07 15:44:05 aws-cloudtrail-logs-291926095288-bf6a3363  
2022-10-13 16:31:25 awsengadsa
```

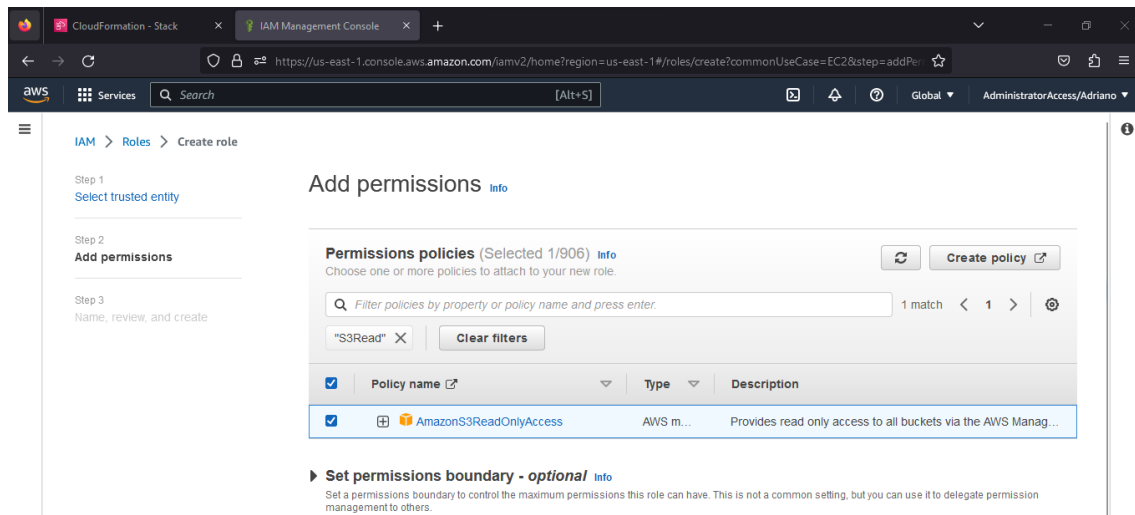
2) Após fazermos uma PoC insegura, vamos seguir para o que a AWS recomenda como boas práticas de segurança, limpe a secret e Access key adicionadas na instância com o comando `rm -rd credentials`

Este comando limpará as configurações de credenciais, agora seguiremos para o próximo passo.

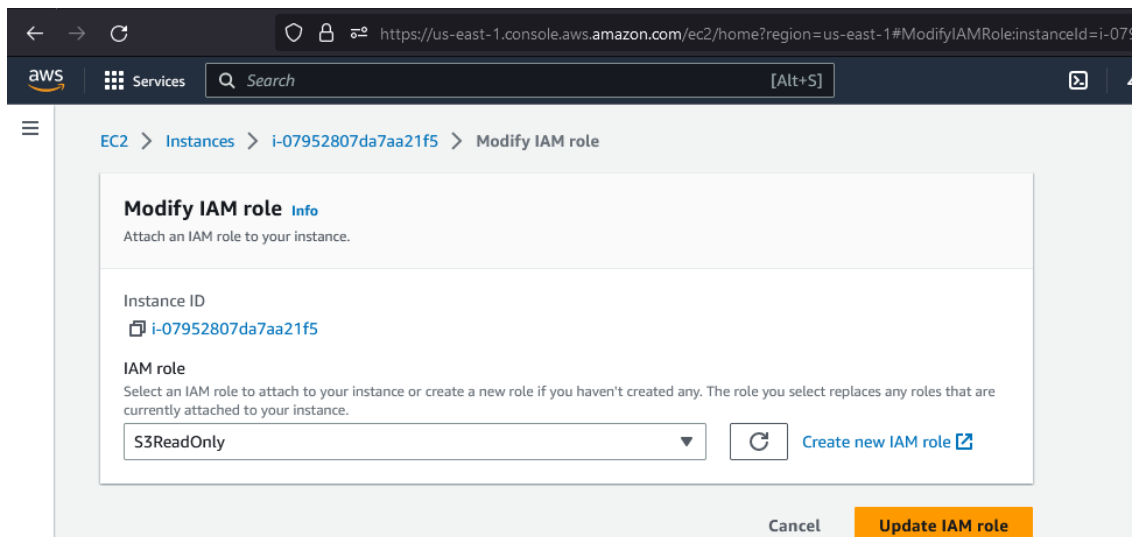
Passo 2: Crie uma role que utilizará como trusted Entity a API do EC2.



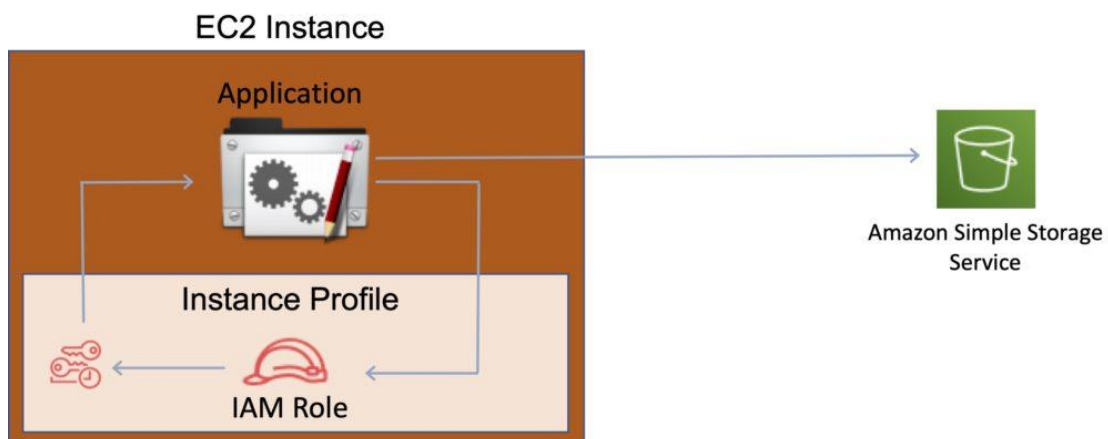
Após selecionar, clique em avançar e localize a policy S3ReadOnly, clique em avançar novamente e defina um nome para sua Role.



Com a Role criada, vá na instância EC2 que está rodando e clique em Action > Security > Modify IAM Role



Com este comando simples teremos a conexão segura efetuada entre uma EC2 e um Bucket S3, utilizar de recursos de Role Based Access controls, garante uma maior autonomia, segurança e elimina riscos de vazamento de credenciais críticas, devido ao formato com a chamada para autenticação funciona, deixarei abaixo um exemplo de comunicação e troca de dados que a AWS aplica ao utilizar uma Role neste cenário que efetuei a PoC.



Referencias utilizadas:

AWS CloudFormation - https://docs.aws.amazon.com/pt_br/cloudformation/index.html

IAM Best Pratics - https://docs.aws.amazon.com/pt_br/IAM/latest/UserGuide/best-practices.html

AWS EC2 Functions - https://docs.aws.amazon.com/pt_br/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html