# Cloudabrodo

## un cloud colabrodo

HackInBo®

Lab Edition

2021

16ª EDIZIONE

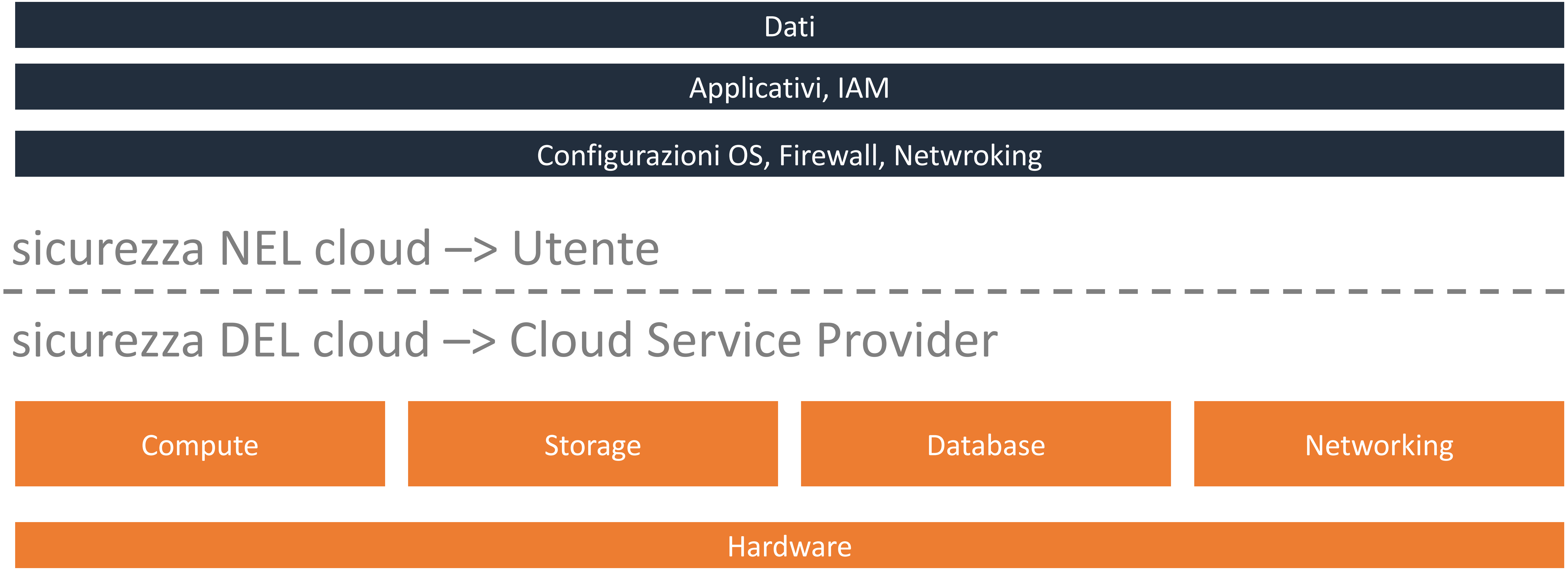
Passionate information security practitioner, researcher, speaker, lecturer.

He holds a Master's in electronic engineering from University La Sapienza of Rome, with years of experience in penetration testing, vulnerability assessments, embedded device and RF hacking.

He is currently employed as Red Team manager in one of the largest online fashion retail group, shaping new strategies to fight and simulate cyber adversaries.

# WHOAMI

# Cloud Security - Modello di responsabilità condivisa



Dati

Applicativi, IAM

Configurazioni OS, Firewall, Netwroking

sicurezza NEL cloud –> Utente

sicurezza DEL cloud –> Cloud Service Provider

Compute

Storage

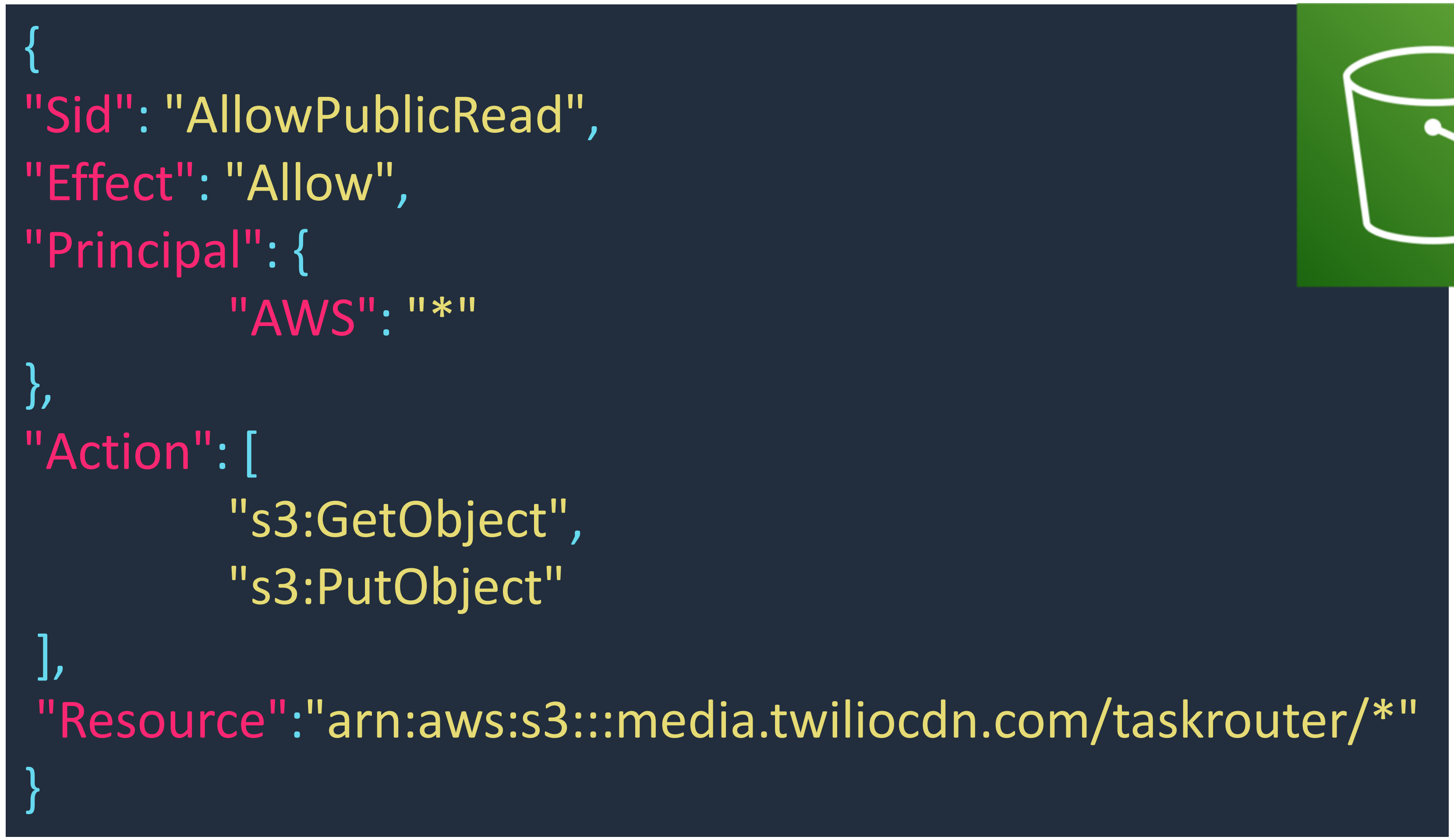
Database

Networking

Hardware

*https://aws.amazon.com/it/compliance/shared-responsibility-model/*

# TaskRouter JS SDK Security Incident - Luglio 2020



```
{
"Sid": "AllowPublicRead",
"Effect": "Allow",
"Principal": {
            "AWS": "*"
},
"Action": [
            "s3:GetObject",
            "s3:PutObject"
 ],
 "Resource":"arn:aws:s3:::media.twiliocdn.com/taskrouter/*"
}
```

https://www.twilio.com/blog/incident-report-taskrouter-js-sdk-july-2020

| Platform | Tactic | Technique ID | Technique | Description |
|---|---|---|---|---|
| AWS | Initial Access | T1078 | Valid Accounts: Cloud Accounts | S3 Resource Policy – Public R/W |
| AWS | Initial Access | T1195 | Supply Chain Compromise: Compromise Software Dependencies and Development Tools | Malicious Javascript embedded in twilio library |
| AWS | Exfiltration | T1048 | Exfiltration Over Alternative Protocol | Twilio served malicious js used in malvertising campaigns |

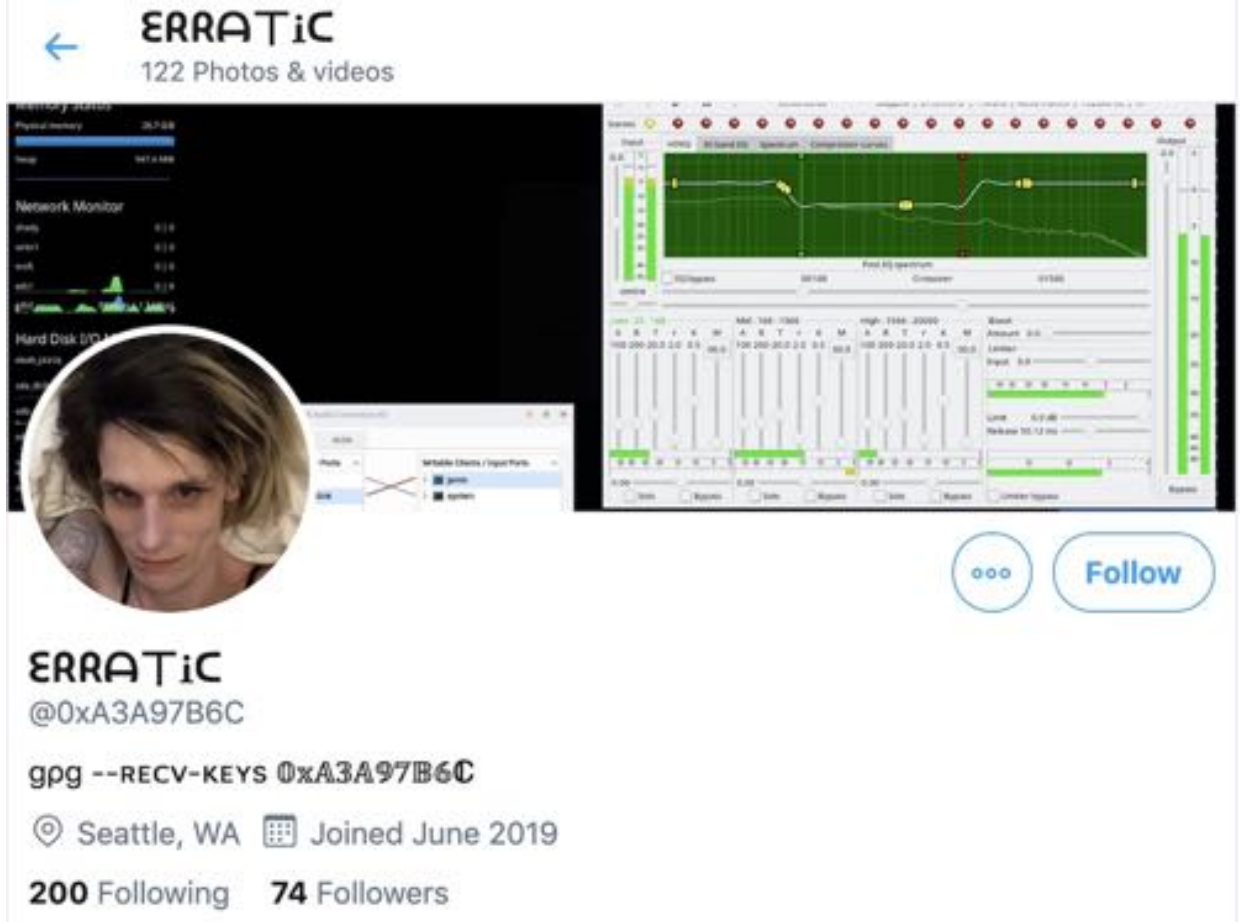
https://www.twilio.com/blog/incident-report-taskrouter-js-sdk-july-2020

# Alcuni epic fails

# Alcuni epic fails

# Il breach è avvenuto tra il 22 e il 23 marzo, è stata scoperto quasi quattro mesi dopo (19 luglio)

- numeri di carte di credito
- date di nascita
- indirizzi
- nomi
- numeri di telefono
- cronologia delle transazioni
- 140.000 numeri di previdenza sociale
- 80.000 numeri di conti bancari



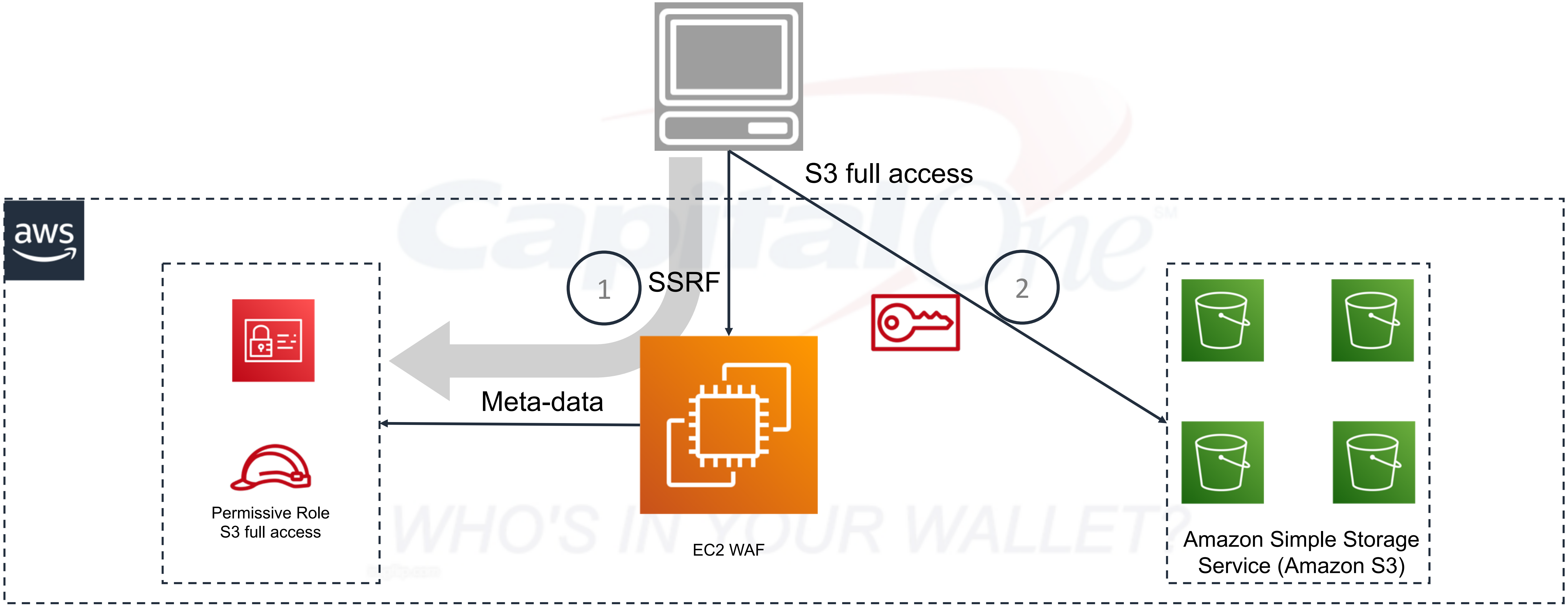
ERRATiC
122 Photos & videos

ERRATiC
@0xA3A97B6C

gpg --RECV-KEYS 0xA3A97B6C

Seattle, WA  Joined June 2019

200 Following  74 Followers



# Alcuni epic fails

| Platform | Tactic | Technique ID | Technique | Description |
|---|---|---|---|---|
| AWS | Initial Access | T1190 | Exploit Public-Facing Application | SSRF in webapp |
| AWS | Credential Access | T1522 | Unsecured Credentials: Cloud Instance Metadata API | AWS keys from EC2 metadata |
| AWS | Collection | T1530 | Data from Cloud Storage Object | Access improperly secured cloud storage |
| AWS | Initial Access | T1078 | Valid Accounts: Cloud Accounts | IAM Role with S3FullAccess |
| AWS | Exfiltration | T1020 | Automated Exfiltration | S3 Sync |

# Alcuni epic fails

# Amazon Simple Storage Service (Amazon S3)

- L’accesso è gestito tramite AWS Identity and Access Management **(IAM**), per creare utenti e gestirne gli accessi
- Liste di controllo accessi **(ACL)**, per rendere singoli oggetti accessibili a utenti autorizzati
  - Proprietario del bucket (il tuo account AWS)
  - Chiunque (accesso pubblico)
  - Gruppo di utenti autenticati (chiunque abbia un account AWS)
- **Policy** bucket, per configurare le autorizzazioni per tutti gli oggetti all'interno di un singolo bucket S3
- Autenticazione tramite AWS signature, per consentire ad altri accesso a tempo limitato tramite URL temporanei

https://aws.amazon.com/it/s3/security

https://docs.aws.amazon.com/AmazonS3/latest/userguide/managing-acls.html

# Amazon Simple Storage Service (Amazon S3)

Virtual-hosted–style access

- https://*bucket-name*.s3.*Region*.amazonaws.com/*key name*

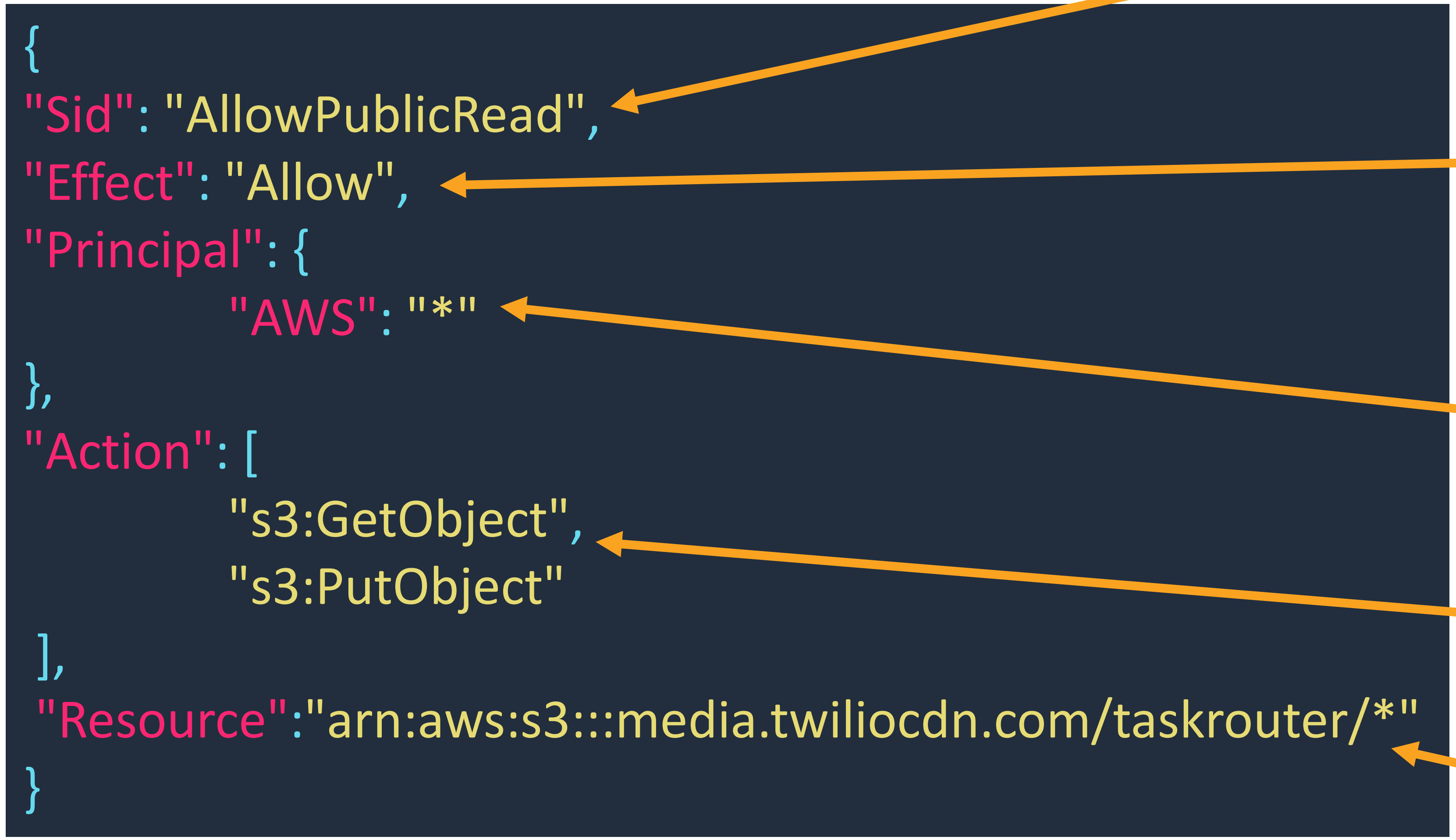
Path-style access

- https://s3.*Region*.amazonaws.com/*bucket-name*/*key name*

Website endpoints

- s3-website dash (-) Region - http://*bucket-name*.s3-website-*Region*.amazonaws.com
- s3-website dot (.) Region - http://*bucket-name*.s3-website.*Region*.amazonaws.com

# Resource-based policies



```
{
"Sid": "AllowPublicRead",
"Effect": "Allow",
"Principal": {
            "AWS": "*"
},
"Action": [
            "s3:GetObject",
            "s3:PutObject"
 ],
 "Resource":"arn:aws:s3:::media.twiliocdn.com/taskrouter/*"
}
```

https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html

Il **Sid** (statement ID) è un identificatore facoltativo fornito per la dichiarazione della policy

L'elemento **Effect** è obbligatorio e specifica se l'istruzione restituisce un consenso o un rifiuto esplicito, di default, l’accesso alle risorse è denied.

L’elemento **Principal** specifica a chi è rivolta la direttiva di allow o deny per l’accesso alle risorse.

L'elemento **Action** descrive l'azione o le azioni specifiche che verranno consentite o negate

L'elemento **Resource** specifica l'oggetto o gli oggetti a cui è riferita la policy

## 1 – Create a canonical request for Signature Version 4

**CanonicalRequest** = HTTPRequestMethod + '\n' + CanonicalURI + '\n' + CanonicalQueryString + '\n' + CanonicalHeaders + '\n' + SignedHeaders + '\n' + HexEncode(Hash(RequestPayload))

## 2 – Create a string to sign for Signature Version 4

**StringToSign** = Algorithm + \n + RequestDateTime + \n + CredentialScope + \n + HashedCanonicalRequest

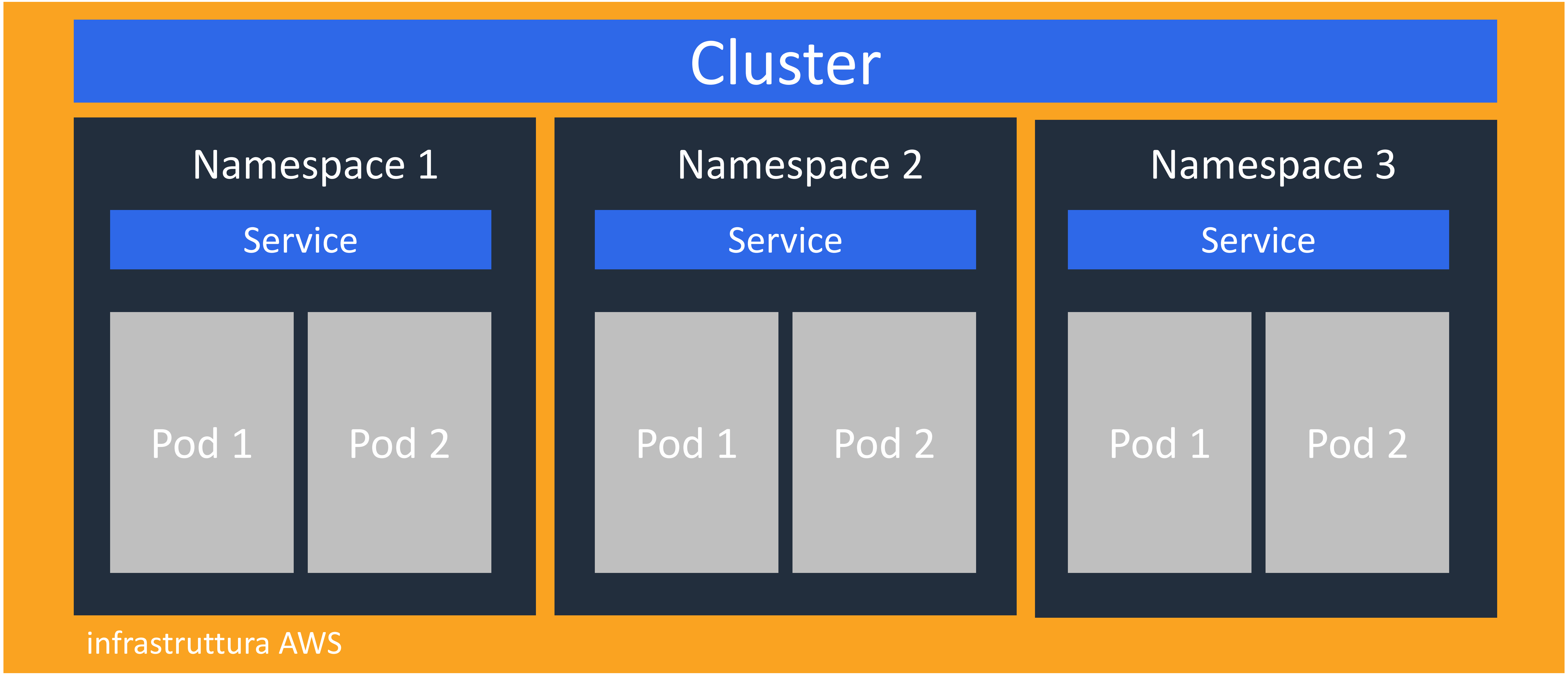
## 3 – Calculate the signature for AWS Signature Version 4

**Signature=** HMAC(HMAC(HMAC(HMAC("AWS4" + SecretAccessKey,"20150830"),’region’),"iam"),"aws4_request")

https://docs.aws.amazon.com/general/latest/gr/sigv4_signing.html

# AWS signing

IT'S
LAB TIME



Cluster

Namespace 1

Service

Pod 1

Pod 2

Namespace 2

Service

Pod 1

Pod 2

Namespace 3

Service

Pod 1

Pod 2

infrastruttura AWS

# Elastic Kubernetes Service (EKS)

# EKS configurazioni di default

eksctl-*<cluster-name>*-nodegroup-NodeInstanceRole

AmazonEKSWorkerNodePolicy

Questa policy permette ai worker node EKS di integrarsi in un EKS Clusters.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets",
        "ec2:DescribeVolumes",
        "ec2:DescribeVolumesModifications",
        "ec2:DescribeVpcs",
        "eks:DescribeCluster"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

eksctl-*<cluster-name>*-nodegroup-NodeInstanceRole

AmazonEKSWorkerNodePolicy

Questa policy permette ai worker node EKS di integrarsi in un EKS Clusters.

AmazonEC2ContainerRegistryReadOnly

Accesso in lettura ai repositories Amazon EC2 Container Registry

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecr:GetAuthorizationToken",
        "ecr:BatchCheckLayerAvailability",
        "ecr:GetDownloadUrlForLayer",
        "ecr:GetRepositoryPolicy",
        "ecr:DescribeRepositories",
        "ecr:ListImages",
        "ecr:DescribeImages",
        "ecr:BatchGetImage",
        "ecr:GetLifecyclePolicy",
        "ecr:GetLifecyclePolicyPreview",
        "ecr:ListTagsForResource",
        "ecr:DescribeImageScanFindings"
      ],
      "Resource": "*"
    }
  ]
}
```

# EKS configurazioni di default

eksctl-*<cluster-name>*-nodegroup-NodeInstanceRole

AmazonEKSWorkerNodePolicy

Questa policy permette ai worker node EKS di integrarsi in un EKS Clusters.

AmazonEC2ContainerRegistryReadOnly

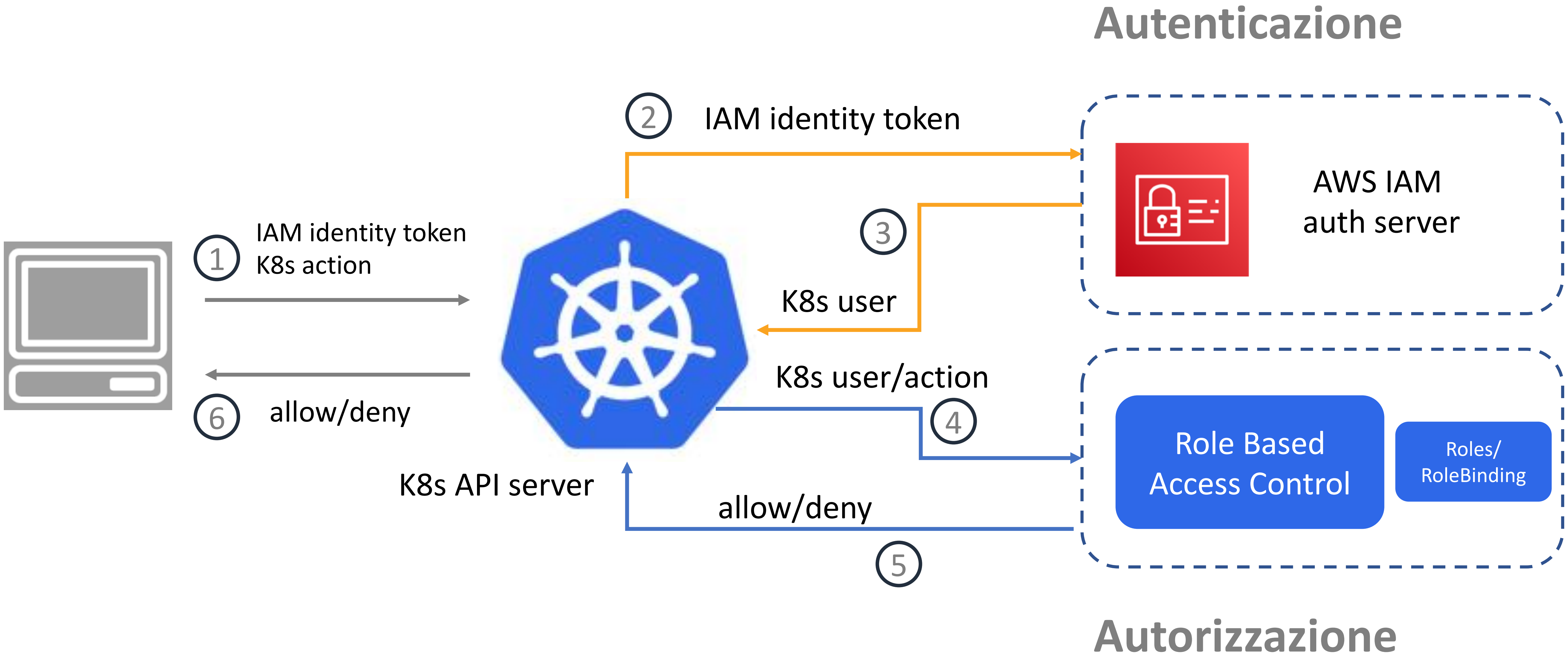
Accesso in lettura ai repositories Amazon EC2 Container Registry

AmazonEKS_CNI_Policy

VPC CNI Plugin (amazon-vpc-cni-k8s) la policy permette al EKS worker node di cambiare configurazioni di networking.

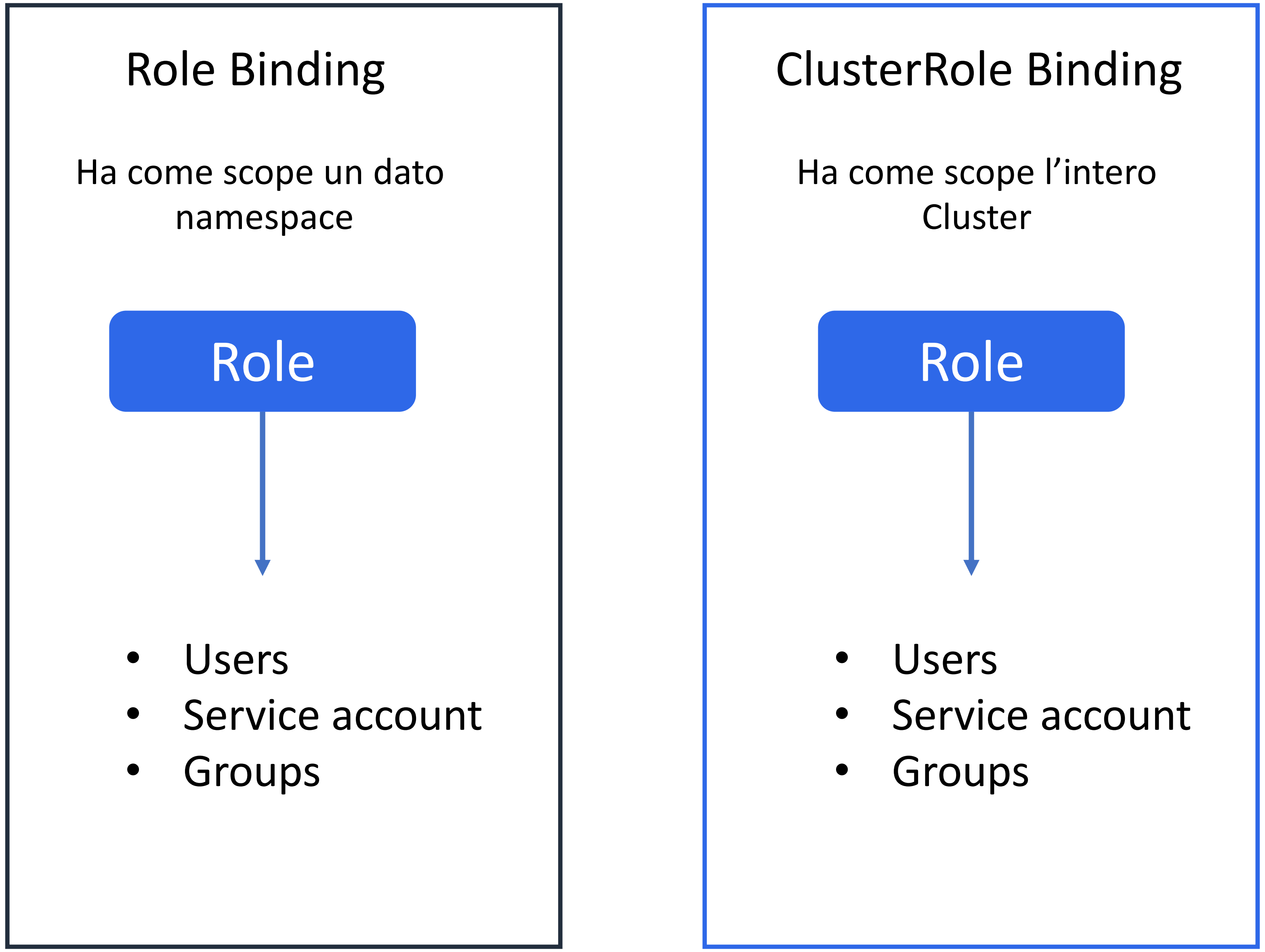
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:AssignPrivateIpAddresses",
        "ec2:AttachNetworkInterface",
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
        "ec2:DescribeInstances",
        "ec2:DescribeTags",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeInstanceTypes",
        "ec2:DetachNetworkInterface",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:UnassignPrivateIpAddresses"
      ],
      "Resource": "*"
    },{
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:network-interface/*"
      ]
    }
```

# EKS configurazioni di default


Autenticazione
2
IAM identity token
AWS IAM
auth server
1
IAM identity token
K8s action
3
K8s user
K8s user/action
4
6
allow/deny
K8s API server
Role Based
Access Control
Roles/
RoleBinding
allow/deny
5
Autorizzazione

# k8s RBAC

# k8s RBAC



Role Binding

Ha come scope un dato namespace

- Users
- Service account
- Groups

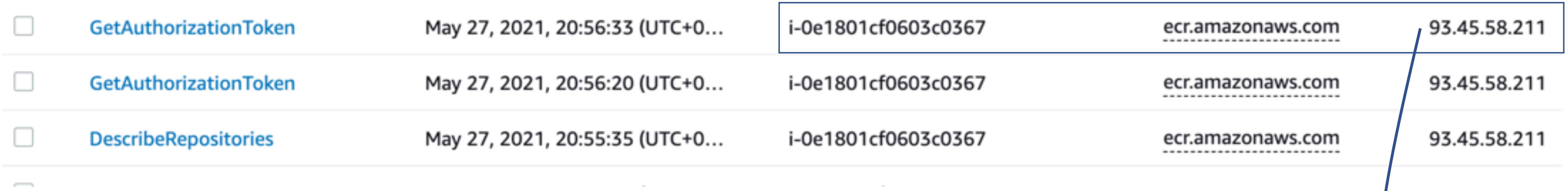
ClusterRole Binding

Ha come scope l’intero Cluster

- Users
- Service account
- Groups

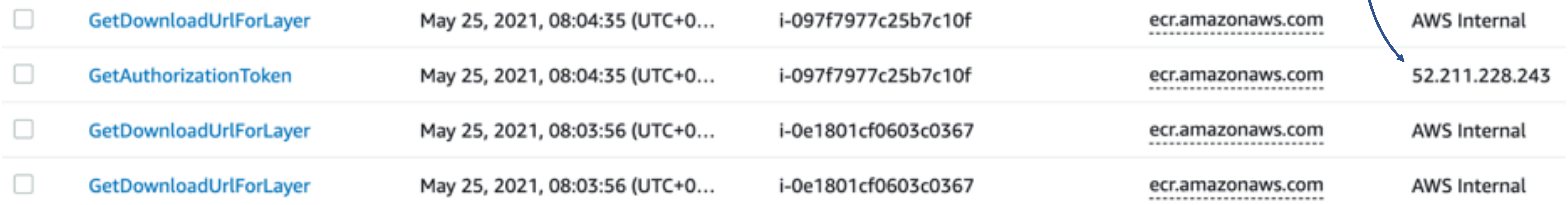
```yaml
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: secret-reader
rules:
- apiGroups: [""]
  resources: ["secrets"]
  verbs: ["get", "list", "watch"]
```

## Flusso anomalo



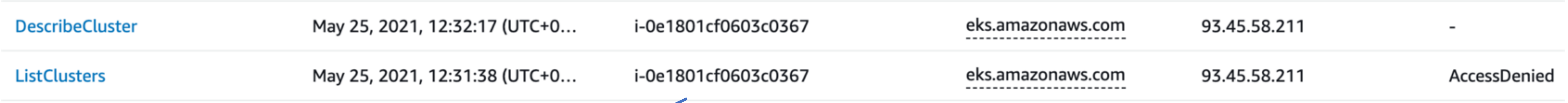
| | | | | | |
|---|---|---|---|---|---|
| ☐ | GetAuthorizationToken | May 27, 2021, 20:56:33 (UTC+0... | i-0e1801cf0603c0367 | ecr.amazonaws.com | 93.45.58.211 |
| ☐ | GetAuthorizationToken | May 27, 2021, 20:56:20 (UTC+0... | i-0e1801cf0603c0367 | ecr.amazonaws.com | 93.45.58.211 |
| ☐ | DescribeRepositories | May 27, 2021, 20:55:35 (UTC+0... | i-0e1801cf0603c0367 | ecr.amazonaws.com | 93.45.58.211 |

## Flusso normale



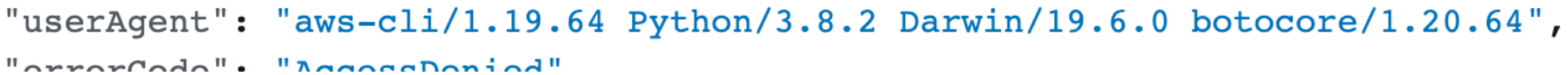
| | | | | | |
|---|---|---|---|---|---|
| ☐ | GetDownloadUrlForLayer | May 25, 2021, 08:04:35 (UTC+0... | i-097f7977c25b7c10f | ecr.amazonaws.com | AWS Internal |
| ☐ | GetAuthorizationToken | May 25, 2021, 08:04:35 (UTC+0... | i-097f7977c25b7c10f | ecr.amazonaws.com | 52.211.228.243 |
| ☐ | GetDownloadUrlForLayer | May 25, 2021, 08:03:56 (UTC+0... | i-0e1801cf0603c0367 | ecr.amazonaws.com | AWS Internal |
| ☐ | GetDownloadUrlForLayer | May 25, 2021, 08:03:56 (UTC+0... | i-0e1801cf0603c0367 | ecr.amazonaws.com | AWS Internal |

# Detection – Anomalie ECR



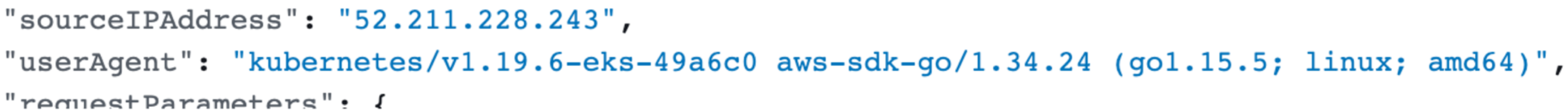
| | | | | | |
|---|---|---|---|---|---|
| DescribeCluster | May 25, 2021, 12:32:17 (UTC+0... | i-0e1801cf0603c0367 | eks.amazonaws.com | 93.45.58.211 | - |
| ListClusters | May 25, 2021, 12:31:38 (UTC+0... | i-0e1801cf0603c0367 | eks.amazonaws.com | 93.45.58.211 | AccessDenied |

## user-agent anomalo



```
"userAgent": "aws-cli/1.19.64 Python/3.8.2 Darwin/19.6.0 botocore/1.20.64",
"errorCode": "AccessDenied"
```

## user-agent di un nodo k8s in EKS



```
"sourceIPAddress": "52.211.228.243",
"userAgent": "kubernetes/v1.19.6-eks-49a6c0 aws-sdk-go/1.34.24 (go1.15.5; linux; amd64)",
"requestParameters": {
```

# Detection – Anomalie EKS