

“Alta disponibilitat a AWS amb IaC”

Enunciat PAC

**Assignatura: M17 - Seguretat en sistemes, xarxes i serveis UF4: Seguretat en Xarxes**

**Curs: 2024-25**

**Índex**

1. [Objectius 3](#_bookmark0)
2. [Plantejament 3](#_bookmark1)
3. [Informació essencial 4](#_bookmark2)
4. [Esquema de qualificació 4](#_bookmark3)

[6. Distribució de les sessions 5](#_bookmark4)

[5. Enunciat 6](#_bookmark5)

[FASE 1: Preparació de l'entorn i configuració d’AWS CLI 7](#_bookmark6)

[FASE 2: Configuració de Terraform i preparació del backend 8](#_bookmark7)

[FASE 3: Creació del bucket S3 per allotjar la web estàtica 14](#_bookmark8)

[FASE 4: Configuració de CloudFront per servir la web estàtica 15](#_bookmark9)

[FASE 5: Implementació de HTTPS amb ACM 18](#_bookmark10)

[FASE 6: Configuració de Seguretat en S3 i Control d’Accés a CloudFront 19](#_bookmark11)

[FASE 7: Validació del Desplegament, Proves Finals i Auditoria de Seguretat 21](#_bookmark12)

# Objectius

L’objectiu principal d’aquesta pràctica és proporcionar als alumnes una experiència pràctica en la implementació d’una arquitectura segura i d’alta disponibilitat a AWS utilitzant Infraestructura com a Codi (IaC) amb Terraform. A través d’aquest projecte, els estudiants desenvoluparan les seves competències en administració de serveis al núvol, configuració de xarxes segures i gestió d’accessos mitjançant AWS.

Els objectius específics són:

* Comprendre i aplicar conceptes de seguretat i disponibilitat en serveis AWS.
* Aprendre a configurar i gestionar recursos com S3, CloudFront i ACM per a una web estàtica.
* Configurar AWS CLI i Terraform per automatitzar el desplegament de la infraestructura.
* Implementar polítiques de seguretat IAM i restriccions d’accés per protegir els recursos.
* Validar el funcionament del desplegament i realitzar proves de seguretat.
* Aprendre a gestionar certificats SSL/TLS amb AWS Certificate Manager (ACM).
* Aplicar pràctiques d’optimització i troubleshooting per millorar la infraestructura desplegada.

# Plantejament

Aquesta pràctica se centra en la implementació d’una web estàtica segura a AWS, aplicant Infraestructura com a Codi (IaC) amb Terraform per assegurar un desplegament automatitzat i escalable. Els estudiants treballaran en un entorn real de serveis AWS, configurant diferents capes de seguretat i optimització per garantir alta disponibilitat i protecció de dades.

Els alumnes seguiran les següents fases:

* Configuració inicial: Instal·lació i configuració d’AWS CLI i Terraform.
* Gestió de l’estat de la infraestructura amb un backend remot a AWS S3.
* Creació del bucket S3 per allotjar la web i configuració de permisos de seguretat.
* Implementació de CloudFront per distribuir el contingut amb optimització de la caché.
* Configuració d’un certificat SSL/TLS amb ACM per protegir el trànsit HTTP.
* Aplicació de polítiques de seguretat per restringir l’accés públic i millorar la privacitat.
* Validació i proves de seguretat per assegurar el correcte funcionament del desplegament.

Aquest enfocament permet als estudiants experimentar amb un escenari real, adquirint una comprensió pràctica de la gestió d’infraestructures a AWS i de les polítiques de seguretat necessàries per garantir un ús segur d’aquests serveis.

# Informació essencial

|  |  |
| --- | --- |
| **Data d'inici:** | **Data de finalització:** |
| **24/03/2025** | **07/04/2025** |
| **Qualificació:** | |
| Pràctica: 70% de la nota final. | |
| Examen teòric: 30% de la nota final. | |
| **Format d'entrega:** | |
| Document PDF amb les respostes a les fases (pots utilitzar aquest mateix). | |
| Captures de pantalla demostrant cada pas realitzat. | |
| Explicacions detallades de les configuracions aplicades. | |
| Codi de Terraform utilitzat per crear la infraestructura en format comprimit (ZIP). | |
| Documentació i explicació sobre possibles errors trobats i com s’han resolt. | |

# Esquema de qualificació

|  |  |
| --- | --- |
| **Fase** | **Puntuació** |
| **FASE 1: Preparació de l'entorn i configuració d’AWS CLI** | |
| 1.1. Preguntes | **0.5 punts** |
| 1.2. Tasques a realitzar | **0.5 punts** |
| 1.3. Bonus: Exploració de les regions d'AWS | **0.25 punts** |
| **FASE 2: Configuració de Terraform i preparació del backend** | |
| 2.1. Preguntes | **0.5 punts** |
| 2.2. Tasques a realitzar | **0.5 punts** |
| 2.3. Bonus: Comparació entre Terraform i AWS CloudFormation | **0.25 punts** |
| **FASE 3: Creació del bucket S3 per allotjar la web estàtica** | |
| 3.1. Preguntes | **0.5 punts** |
| 3.2. Tasques a realitzar | **1 punt** |
| 3.3. Bonus: Utilitzar AWS CLI per pujar fitxers a S3 | **0.25 punts** |
| **FASE 4: Configuració de CloudFront per servir la web estàtica** | |
| 4.1. Preguntes | **0.5 punts** |
| 4.2. Tasques a realitzar | **1 punt** |
| 4.3. Bonus: Invalidació de la memòria cau en CloudFront | **0.25 punts** |
| **FASE 5: Configuració del certificat SSL amb AWS Certificate Manager (ACM)** | |
| 5.1. Preguntes | **0.5 punts** |
| 5.2. Tasques teòriques | **0.5 punts** |
| 5.3. Bonus: Gestió de certificats SSL/TLS amb ACM | **0.25 punts** |
| **FASE 6: Configuració de permisos i seguretat en S3 i CloudFront** | |
| 6.1. Preguntes | **0.5 punts** |
| 6.2. Tasques a realitzar | **1 punt** |
| 6.3. Bonus: Seguretat en S3 i CloudFront | **0.25 punts** |
| **FASE 7: Validació del desplegament i proves finals** | |
| 7.1. Preguntes | **0.5 punts** |
| 7.2. Tasques a realitzar | **0.5 punts** |
| 7.3. Bonus: Automatització del desplegament i gestió de versions en S3 | **0.25 punts** |
| **TOTAL** | **10 punts** |

# Distribució de les sessions

A continuació es detallen les sessions previstes per a la realització de la pràctica. Aquestes sessions combinen explicacions teòriques, demostracions pràctiques i espais de treball supervisat per garantir que els alumnes entenguin els conceptes i puguin implementar-los correctament.

### Dilluns 24/03 (2h) - Introducció i Fases 1 i 2

* Introducció a la computació al núvol i AWS com a plataforma (IAM, S3, CloudFront, ACM, Terraform).
* Explicació teòrica sobre Infraestructura com a Codi (IaC) i la seva aplicació en Terraform.
* Demostració pràctica: Configuració inicial d'AWS CLI i Terraform (fases 1 i 2).
* Presentació de la pràctica i treball lliure supervisat (30 min).

### Dimarts 25/03 (2h) - Fases 3 i 4

* Explicació teòrica sobre la configuració d’infraestructures a AWS amb Terraform.
* Demostració pràctica: Creació d’un bucket S3 i configuració de polítiques d’accés (fase 3).
* Demostració pràctica: Configuració de CloudFront per servir la web de manera segura (fase 4).
* Explicació detallada de la rúbrica d’avaluació i estructura de la pràctica.
* Treball lliure supervisat amb resolució de dubtes (30 min).

### Dilluns 31/03 (2h) - Treball supervisat i optimització

* Demostració pràctica: Configuració de permisos i seguretat en S3 i CloudFront.
* Resolució de dubtes sobre les fases treballades.
* Treball lliure supervisat amb suport individualitzat.

### Dimarts 01/04 (2h) - Optimització i proves finals

* Demostració pràctica: Validació del desplegament i proves finals de seguretat.
* Anàlisi i correcció d’errors comuns en la configuració de l’entorn AWS.
* Ajustos finals i optimització del codi Terraform.
* Treball lliure supervisat per finalitzar la pràctica.

### Dilluns 07/04 (2h) - Última sessió per finalitzar la pràctica (Opcional)

* Revisions finals de la infraestructura desplegada.
* Suport en la documentació final i entrega de la pràctica.

### Dimarts 08/04 (2h) - Examen teòric

* Realització de l’examen final sobre computació al núvol, AWS i Infraestructura com a Codi (IaC).
* Avaluació dels conceptes apresos i aplicacions pràctiques treballades durant la PAC.

# Enunciat

### Desplegament segur d'una web estàtica amb AWS S3, CloudFront i Terraform

L’objectiu d’aquesta pràctica és desplegar una web estàtica segura i d’alta disponibilitat utilitzant AWS S3 com a sistema d’allotjament, CloudFront com a CDN per millorar el rendiment i la seguretat, i AWS Certificate Manager (ACM) per habilitar connexions HTTPS.

El desplegament es gestionarà mitjançant Terraform, seguint el model Infrastructure as Code (IaC). Aquest procés inclourà:

* La creació i configuració d’un bucket S3, amb restriccions d’accés i allotjament web.
* La implementació de CloudFront per servir els continguts amb una capa de caché i protecció HTTPS.
* L’ús d’AWS Certificate Manager (ACM) per generar i gestionar un certificat SSL/TLS.
* L’aplicació de polítiques IAM i restriccions d’accés per garantir que només CloudFront pugui accedir als fitxers de S3.
* La validació i proves de seguretat per verificar que l’accés als recursos està protegit i optimitzat.

El resultat final serà una web estàtica servida a través de CloudFront amb una configuració segura i escalable, seguint bones pràctiques d’alta disponibilitat i seguretat a AWS.

**SI US PLAU UTILITZA EL COLOR BLAU EN LES TEVES RESPOSTES.**

## FASE 1: Preparació de l'entorn i configuració d’AWS CLI

**OBJECTIU:** Instal·lar i configurar AWS CLI per gestionar recursos AWS des de la línia de comandes.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat)

### Què és AWS CLI i per a què serveix?

### Quina comanda es fa servir per configurar AWS CLI?

### Com pots verificar que AWS CLI està correctament configurat?

* 1. **TASQUES A REALITZAR (0.5 punts):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

### Instal·lar AWS CLI al teu sistema.

### Configurar les credencials AWS amb aws configure.

* + - * Accedeix a la consola d'AWS IAM: [AWS IAM Console](https://console.aws.amazon.com/iam)
      * A la barra lateral, fes clic a Users.
      * Selecciona l'usuari per al qual vols generar una clau d’accés.
      * Ves a la pestanya Security credentials.
      * A la secció Access keys, fes clic a Create access key.
      * Guarda l'Access Key ID i Secret Access Key en un lloc segur.

### Comprovar que tens accés a AWS executant aws s3 ls.

#### BONUS: Exploració de les regions d'AWS (0.25 punts)

Executa la comanda següent per veure totes les regions disponibles a AWS:

aws ec2 describe-regions --query "Regions[].RegionName"

Identifica la regió per defecte del teu compte i investiga com afecta l’elecció de la regió en termes de latència, alta disponibilitat i costos. Per què algunes empreses trien desplegar serveis en múltiples regions?

## FASE 2: Configuració de Terraform i preparació del backend

**OBJECTIU:** Instal·lar Terraform i configurar un backend per emmagatzemar l'estat de la infraestructura a S3.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat)

### Què és Terraform i per què és útil?

### Per què es recomana utilitzar un backend remot en Terraform?

### Quina comanda s’utilitza per inicialitzar un projecte Terraform?

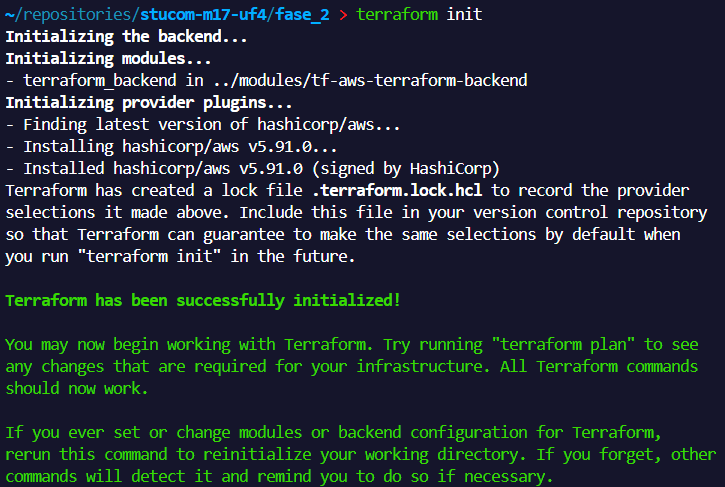
* 1. **TASQUES A REALITZAR (0.5 punts):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

### Instal·lar Terraform.

### Configurar un backend remot en S3 per emmagatzemar l'estat i executar terraform init per inicialitzar el projecte.

### Per configurar el backend de Terrform, aquesta es la guia:

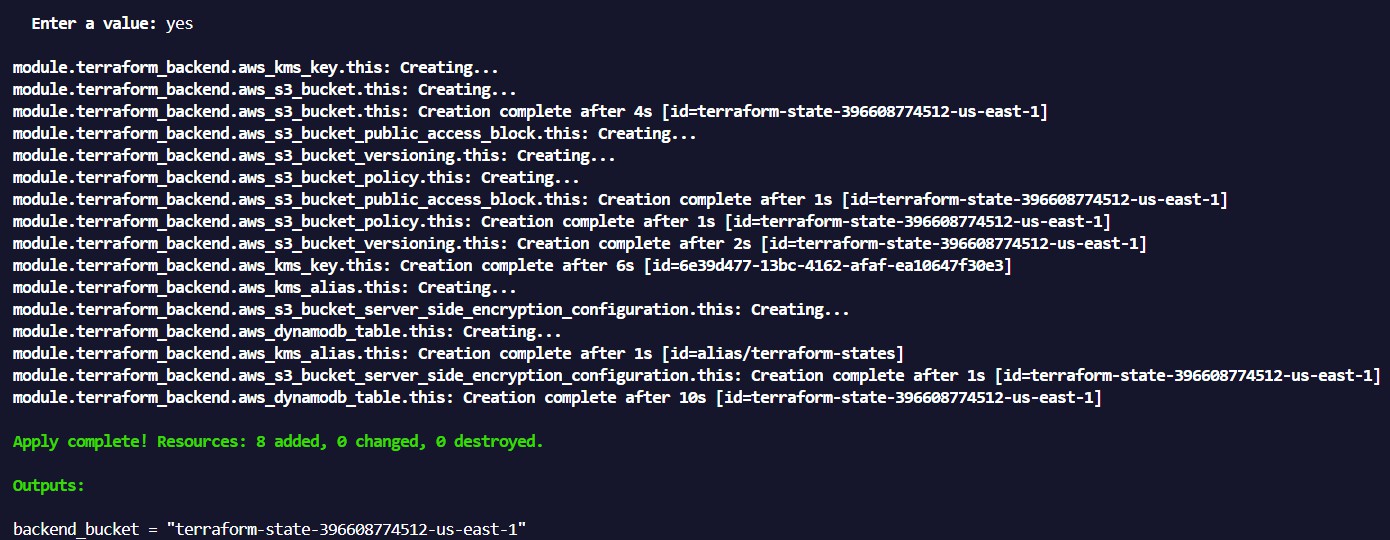
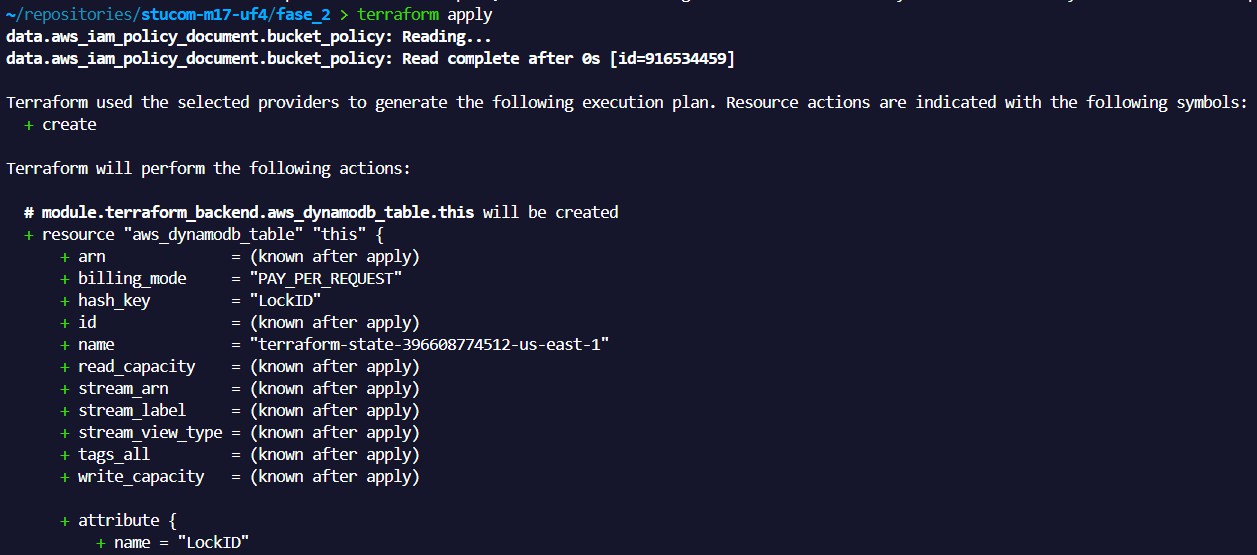
* + - * Des del directori fase\_2 executa la següent seqüencia de comandes:



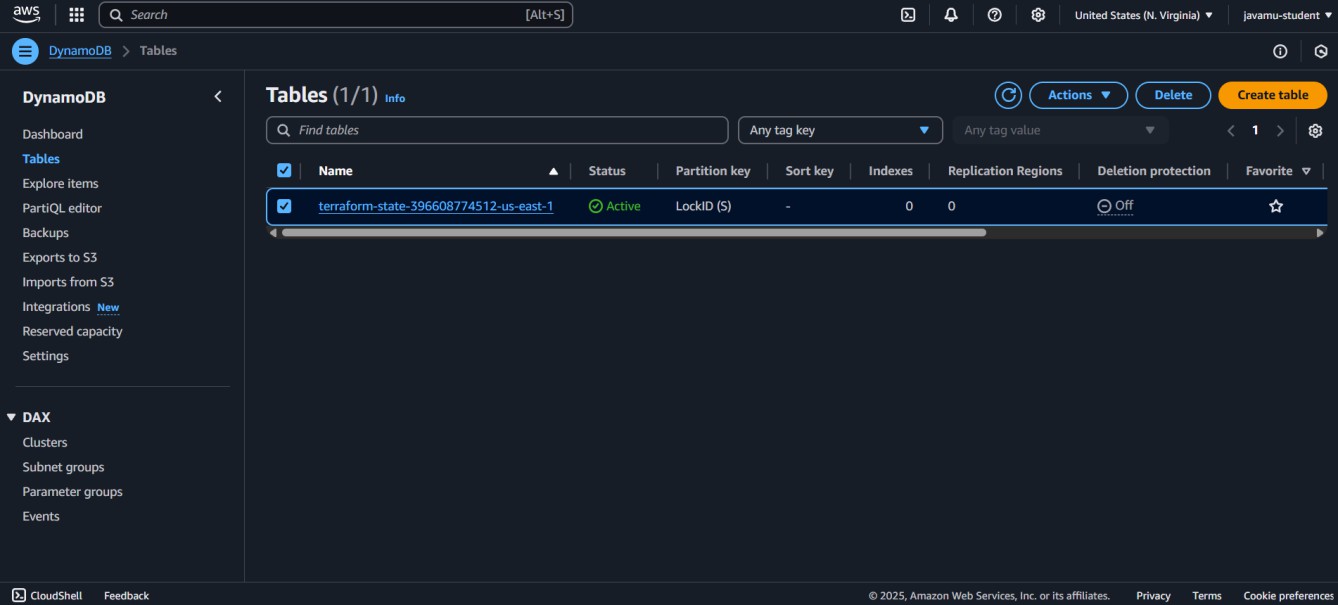
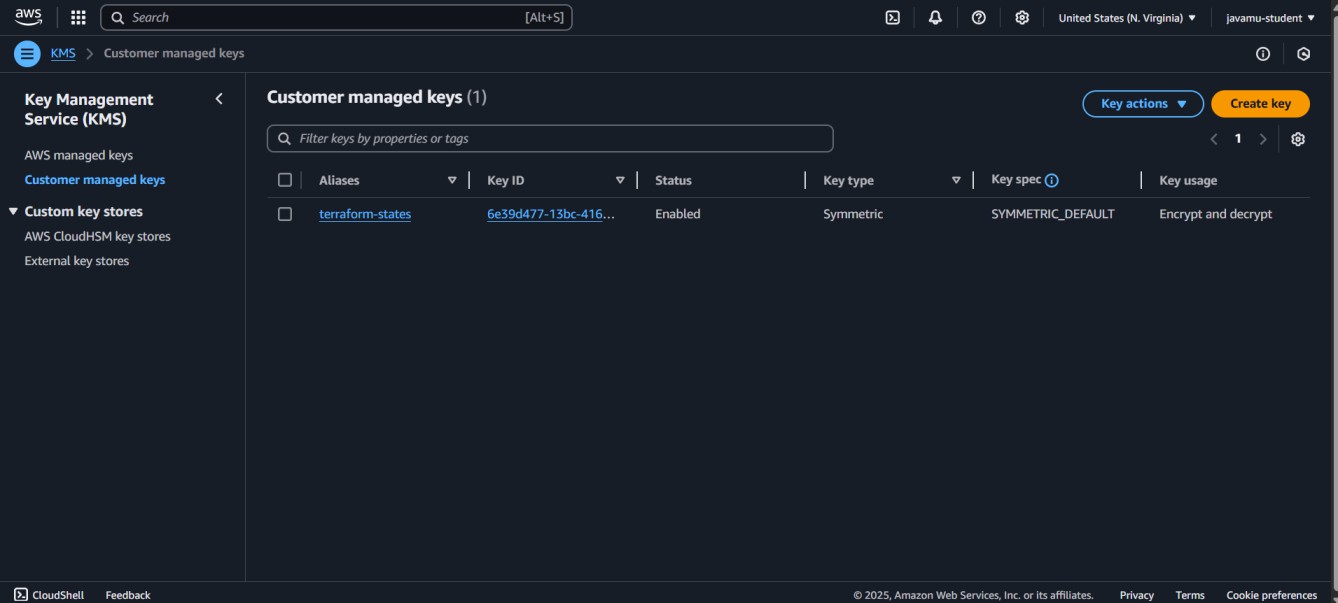
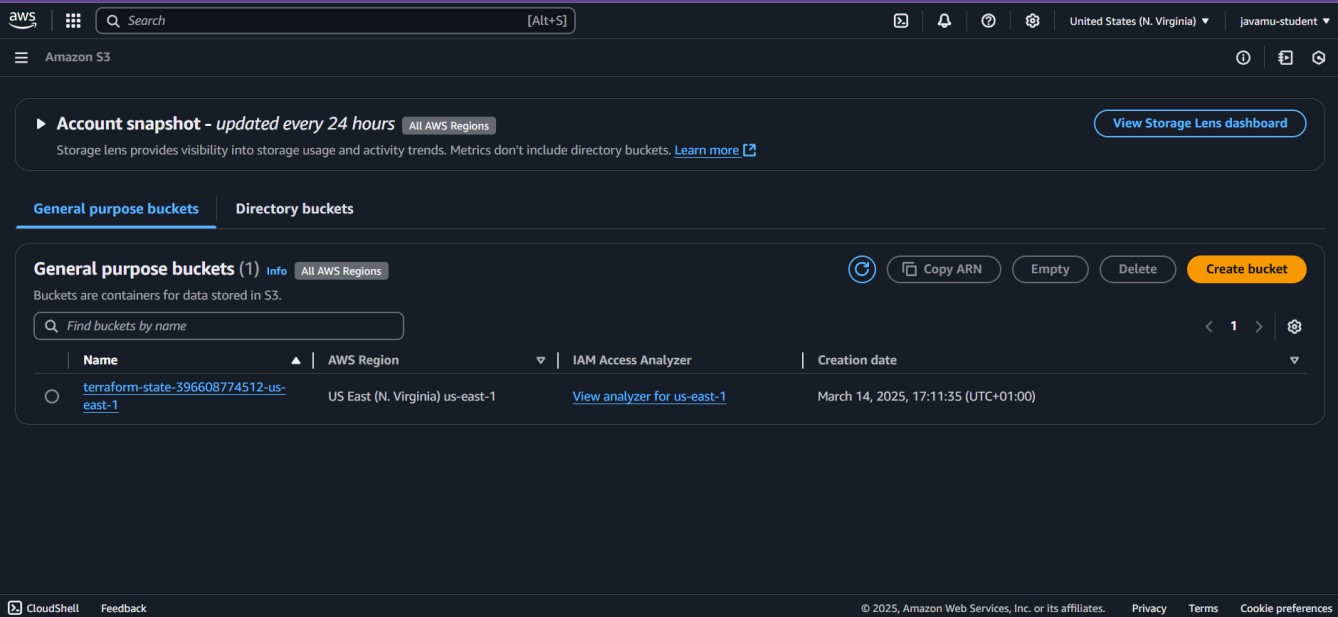
* + - * Ara que ja has inicialitzat el projecte de terraform, pots executar les següents comandes:

terraform plan terraform apply terraform init

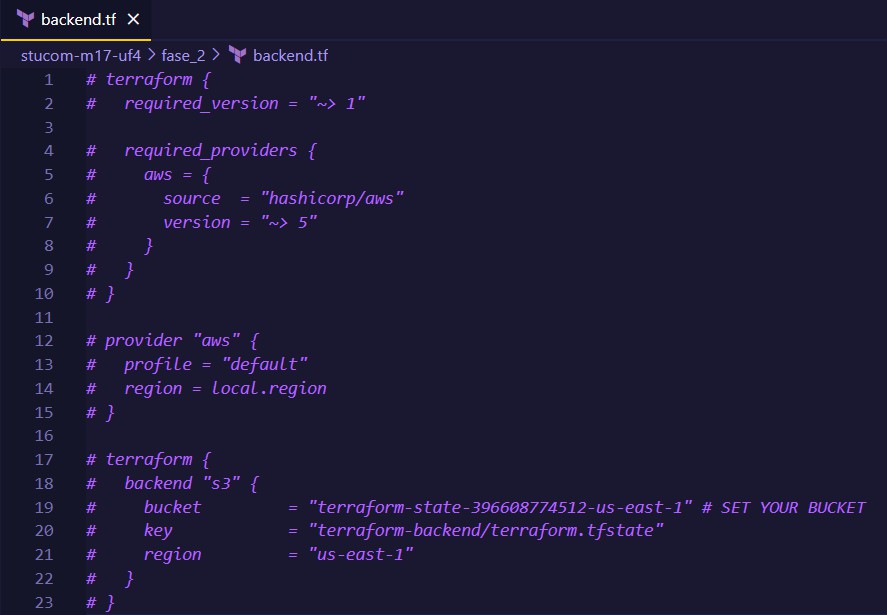
...



* + - * A partir d’aquest moment tots els components necessaris per al terraform backend ja estan creats, ho podem comprovar mitjançant la consola de AWS.



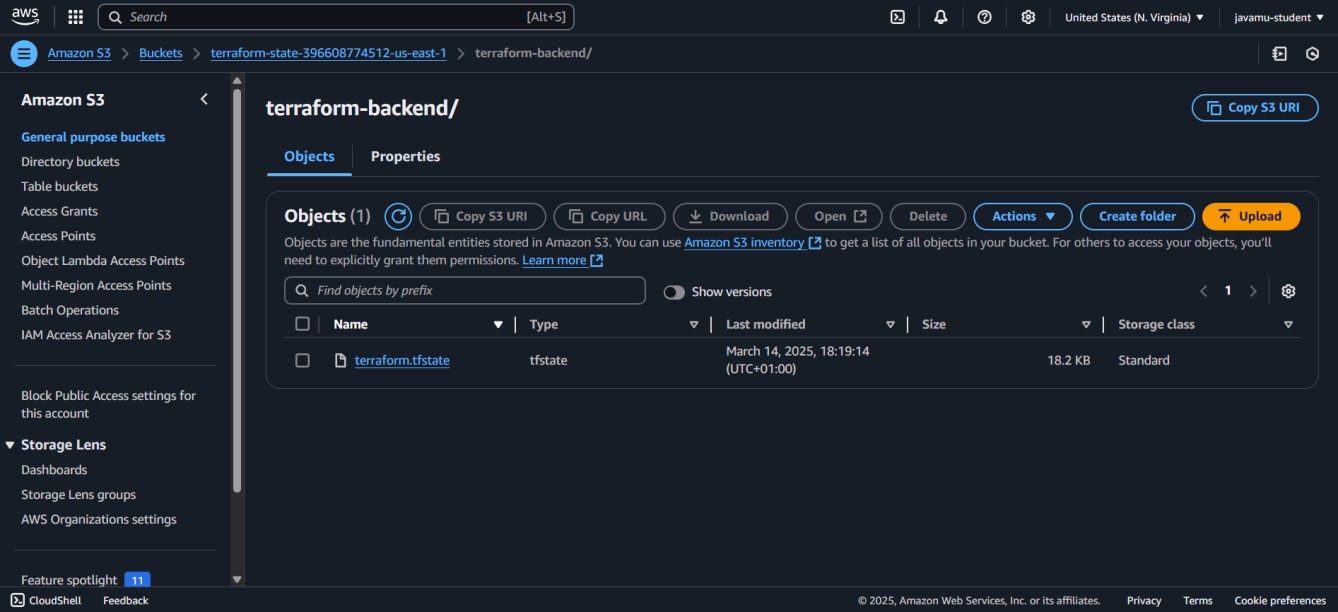
* + - * Ara es el moment de fer el reconfigure per al estat de terraform. Primer descomenta el fitxer backend.tf



* + - * I a continuació executa la comanda: terraform init –reconfigure



* + - * Si tot ha sortit correctament podràs veure com el teu estat ha pujat al bucket s3:



#### BONUS: Comparació entre Terraform i AWS CloudFormation (0.25 punts)

Terraform i AWS CloudFormation són eines per gestionar infraestructura com a codi (IaC).

* Cerca dues diferències clau entre Terraform i AWS CloudFormation.
* Explica en quin cas seria millor utilitzar CloudFormation en comptes de Terraform.

## FASE 3: Creació del bucket S3 per allotjar la web estàtica

**OBJECTIU:** Crear i configurar un bucket S3 per allotjar la web estàtica de manera segura.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat)

### Quina diferència hi ha entre un bucket S3 públic i privat?

### Com podem assegurar-nos que només CloudFront podrà accedir als fitxers del bucket?

* 1. **TASQUES A REALITZAR (1 punt):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

### Crear un bucket S3 amb Terraform.

### Configurar el bucket per permetre l’allotjament d’una web estàtica.

### Pujar fitxers HTML i d’index i error. (Si aquests no t’agraden troba la manera de posar els que et semblin mes adients)

### Configurar una política IAM per restringir l’accés.

#### BONUS: Utilitzar AWS CLI per pujar fitxers a S3 (0.25 punts)

* Puja un fitxer amb aws s3 cp i demostra que es guarda correctament.
* Investiga, a part de fitxers estàtics per a una web, què més es pot emmagatzemar en S3?
* Què passa si la quantitat de fitxers en un bucket S3 es descontrola?

## FASE 4: Configuració de CloudFront per servir la web estàtica

### OBJECTIU: Configurar CloudFront per millorar el rendiment i la seguretat de la web allotjada a S3.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat).

### Per què és recomanable utilitzar CloudFront en comptes d’exposar S3 directament?

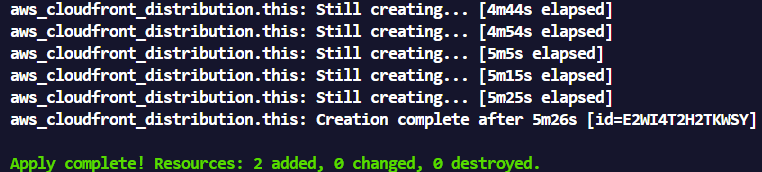
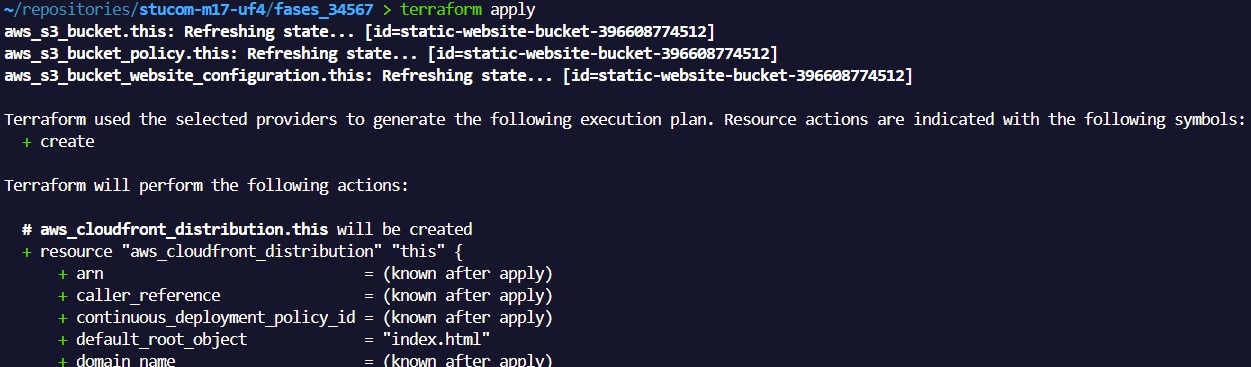
### Com es pot restringir CloudFront perquè només pugui accedir al bucket S3?

* 1. **TASQUES A REALITZAR (1 punt):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

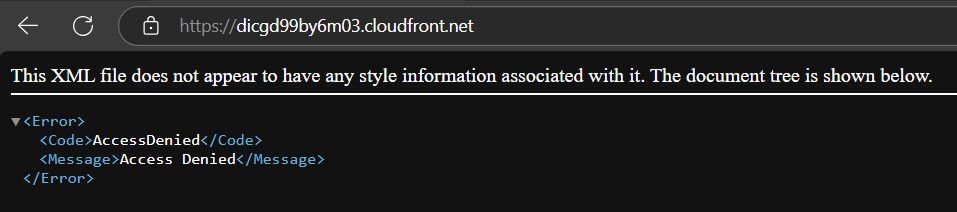
### Crear una distribució CloudFront amb Terraform.

Per crear la distribució CloudFront, primer cal descomentar el codi del fitxer cloudfront.tf. Aquest fitxer defineix la configuració de la distribució i enllaça CloudFront amb el bucket S3.

Executa la següent comanda per aplicar la configuració:



Un cop finalitzat el desplegament, obtindràs un URL de CloudFront. Accedeix-hi des del navegador i comprova que la web es carrega correctament.



Però ens trobem que l’accés està denegat.

### Configurar la caché per optimitzar el lliurament de continguts.

CloudFront utilitza una memòria cau per accelerar la càrrega de la web i reduir el nombre de peticions al bucket S3. Per optimitzar-la, cal modificar la configuració al fitxer cloudfront.tf, ajustant els temps de caché i els paràmetres de les peticions.

* + 1. **Restringir l'accés a S3 perquè només CloudFront hi pugui accedir.** Has intentat accedir a la teva web a través de CloudFront i t’ha aparegut un AccessDenied, oi? Això passa perquè el bucket S3 està configurat com a privat, i CloudFront no té permisos per accedir-hi. Ara corregirem aquest problema!

Ara, edita el fitxer s3.tf i modifica la política perquè només l’OAI de CloudFront pugui accedir als fitxers:

*####################################################################### #########*

*# S3 Bucket Policy to allow CloudFront access ####################################################################### #########*

resource "aws\_s3\_bucket\_policy" "this" { bucket = aws\_s3\_bucket.this.id

policy = jsonencode({ Version = "2012-10-17" Statement = [

{

Sid = "AllowCloudFrontAccess" Effect = "Allow"

Principal = {

AWS = aws\_cloudfront\_origin\_access\_identity.this.iam\_arn

}

Action = "s3:GetObject"

Resource = "${aws\_s3\_bucket.this.arn}/\*"

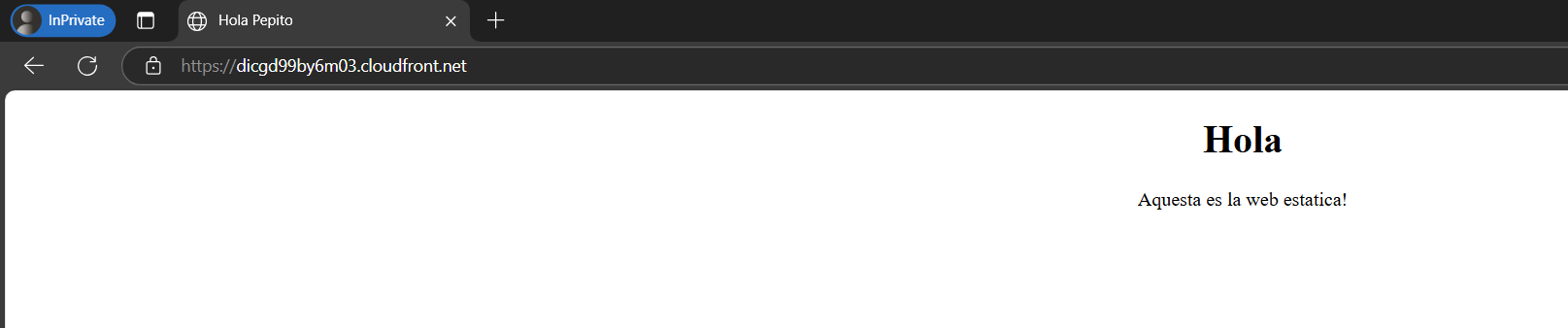
}

]

})

}

I aplica terraform, en principi ja has de poder accedir!



#### BONUS: Invalidació de la memòria cau en CloudFront (0.25 punts)

* Què significa invalidar la memòria cau en CloudFront i en quins casos és necessari fer- ho?
* Executa una invalidació de caché a CloudFront mitjançant AWS CLI.

Via AWS CLI s’executa la següent comanda per invalidar la caché de tots els fitxers servits per CloudFront:

aws cloudfront create-invalidation --distribution-id <ID\_DE\_CLOUDFRONT> --paths "/\*"

* Després d’invalidar la caché, comprova si els canvis en els fitxers de la web es reflecteixen immediatament.

Accedeix a l’URL de CloudFront i refresca la pàgina.

Si els canvis no es reflecteixen, prova a netejar la caché del navegador.

Si encara es veu la versió antiga, comprova a la consola d’Invalidations d’AWS si la petició està completada.

## FASE 5: Implementació de HTTPS amb ACM

**OBJECTIU:** Configurar un certificat SSL/TLS amb AWS Certificate Manager per garantir connexions segures via HTTPS.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat).

### Quin impacte té HTTPS en la seguretat i rendiment d'una aplicació web?

### Com es pot assignar un certificat SSL a una distribució CloudFront?

* 1. **TASQUES TEÒRIQUES (0.5 punts):** A causa de les limitacions de costos en la creació d’un domini Route 53, aquesta fase s’ha convertit en teoria. Contesta aquestes preguntes per demostrar la comprensió del procés.

Proporciona una petita explicació (màxim 10 línies per a cada apartat).

### Com es crea un certificat SSL/TLS amb ACM?

### Per què CloudFront requereix que els certificats ACM estiguin a us-east-1?

### Configurar la redirecció automàtica d’HTTP a HTTPS.

### *​BONUS:* Gestió de certificats SSL/TLS amb ACM (0.25 punts)

* Per què és recomanable utilitzar ACM en lloc de importar un certificat manualment a AWS?
* Els certificats d’ACM es renoven automàticament o cal fer-ho manualment? Per què això és útil?

## FASE 6: Configuració de Seguretat en S3 i Control d’Accés a CloudFront

**OBJECTIU:** Implementar mesures de seguretat per protegir els fitxers allotjats en S3 i CloudFront, bloquejant accessos no autoritzats i aplicant restriccions per garantir el compliment de bones pràctiques de seguretat.

* + 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat).

### Què passa si un bucket S3 és públic? Com podria afectar la seguretat d’una empresa?

### Què significa xifrar dades en repòs i en trànsit en AWS?

* + 1. **TASQUES A REALITZAR (1 punt):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

### Afegir una política de bucket que bloquegi l’accés públic.

Per garantir que el bucket S3 no sigui accessible públicament, cal aplicar una política que bloquegi qualsevol accés públic no autoritzat. Això es fa mitjançant el recurs aws\_s3\_bucket\_public\_access\_block a Terraform.

* + - * + Afegir el bloc següent al fitxer s3.tf i aplica terraform:

*####################################################################### #########*

*# Block public access to the S3 bucket ####################################################################### #########*

resource "aws\_s3\_bucket\_public\_access\_block" "this" { bucket = aws\_s3\_bucket.this.id

block\_public\_acls = true block\_public\_policy = true ignore\_public\_acls = true restrict\_public\_buckets = true

}

*####################################################################### #########*

*# Enable server-side encryption for the S3 bucket ####################################################################### #########*

resource "aws\_s3\_bucket\_server\_side\_encryption\_configuration" "this" { bucket = aws\_s3\_bucket.this.id

rule { apply\_server\_side\_encryption\_by\_default { sse\_algorithm = "AES256"

}

}

}

### Aplicar restriccions d’accés geogràfic en CloudFront.

CloudFront permet restringir l’accés als continguts segons la ubicació dels usuaris, evitant que certes regions puguin accedir a la web.

* + - * + Afegir el bloc següent al fitxer cloudfront.tf i aplica terraform:

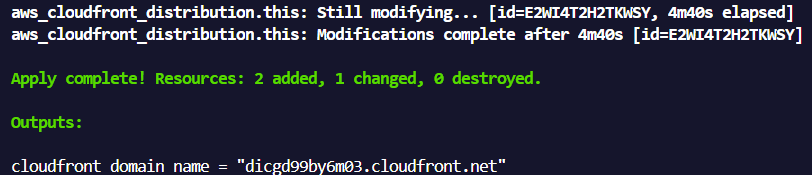
restrictions { geo\_restriction {

restriction\_type = "whitelist"

locations = ["DE", "FR", "ES", "IT", "GB"]

}

}



Després d’implementar la configuració de les seccions **6.2.1 i 6.2.2**, proporciona una **breu explicació** (5-10 línies) sobre:

* + - * + Per què podríem voler restringir l’accés a determinades regions?
        + En quins casos aquesta funcionalitat seria útil en una arquitectura AWS? (proporciona un exemple)
        + Com afecta això la seguretat i el compliment de regulacions de protecció de dades?

#### ​BONUS: Seguretat en S3 i CloudFront (0.25 punts)

* Quina diferència hi ha entre una política IAM i una política de bucket en S3?

Defineix cada tipus de política i explica en quin cas és millor utilitzar una o l’altra. Com poden treballar conjuntament per millorar la seguretat d’S3?

## FASE 7: Validació del Desplegament, Proves Finals i Auditoria de Seguretat

**OBJECTIU:** Comprovar que la infraestructura desplegada a AWS funciona correctament, validar la configuració de seguretat i identificar possibles millores en l’entorn.

* 1. **PREGUNTES (0.5 punts):** Respon aquestes preguntes a continuació i justifica les fonts que has utilitzat per a la resolució (màxim 5 línies per a cada apartat).

### Com es pot verificar que el bucket S3 no és públic?

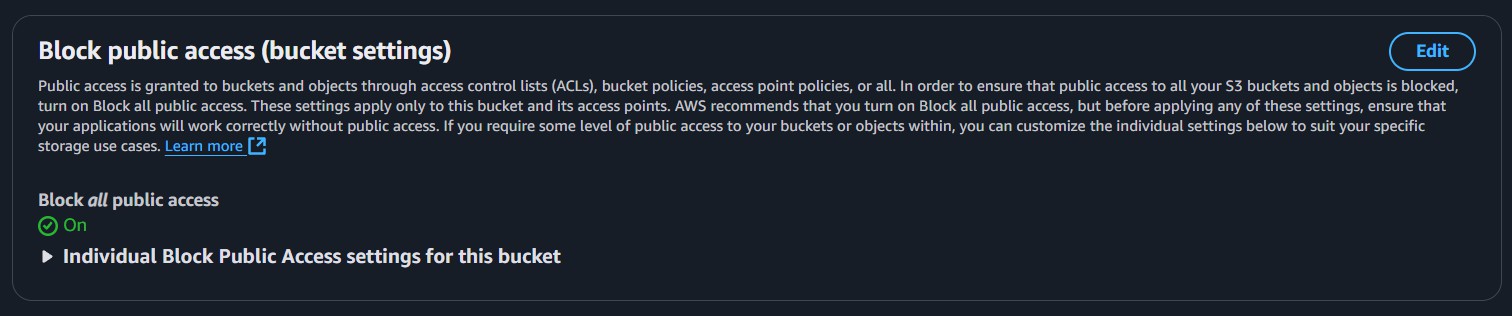
### Com es pot validar que HTTPS funciona correctament?

* 1. **TASQUES A REALITZAR (0.5 punt):** Proporciona captures de pantalla i una petita explicació (màxim 10 línies per a cada apartat).

### Validació de l’accés i configuració de seguretat del bucket S3. Comprova que el bucket S3 està correctament protegit i només CloudFront pot accedir-hi.

Revisió de permisos del bucket:

* + - * Accedir a AWS S3 > Seleccionar bucket > Pestanya "Permissions".
      * Confirmar que "Block Public Access" està activat i que la política de bucket només permet accés des de CloudFront.



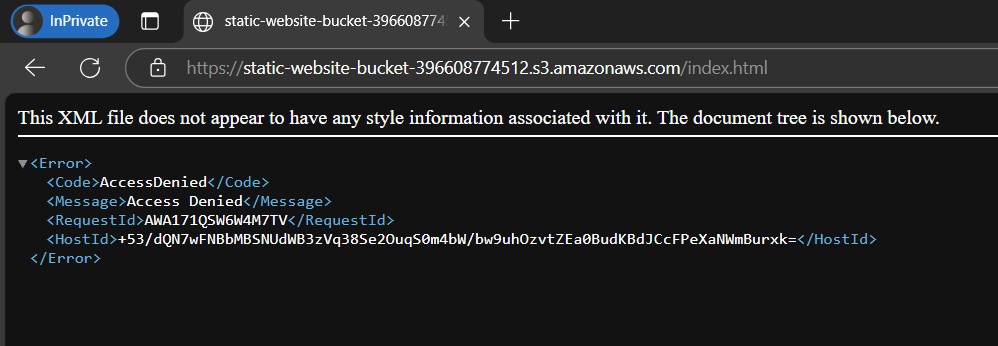
### Quina es la commanda per revisar-ho via AWS CLI?

aws s3api get-bucket-policy --bucket NOM\_DEL\_BUCKET

### Intenta accedir a l’URL del bucket S3

### (http://NOM\_DEL\_BUCKET.s3.amazonaws.com/index.html). Pots?

Resultat esperat, Si es mostra "Access Denied", la configuració és correcta.



### Validació de la càrrega de la web mitjançant CloudFront.

* + - * Accedir a l’URL de CloudFront proporcionat per Terraform (https://dXXXXXXXX.cloudfront.net).
      * Comprovar que la web es carrega correctament.
      * Si es veu l’error "Access Denied", revisar la política de bucket i les restriccions d’accés de CloudFront.
      * Si CloudFront encara serveix una versió antiga de la web, forçar la seva actualització invalidant la caché.

Proporciona evidencies.

### *​BONUS:* Automatització del desplegament i gestió de versions en S3 (0.25 punts)

* Quines opcions tens per automatitzar un desplegament de fitxers en S3? A més d’AWS CLI i Terraform, existeixen eines com GitHub Actions i AWS

CodePipeline que permeten automatitzar la pujada de fitxers en S3 després d’una actualització en el repositori. Investiga com podries configurar una acció de GitHub per desplegar automàticament els canvis quan es fa un git push.

* Com podries fer rollback a una versió anterior de la web si un desplegament té errors? Pots aportar com ho faries en terraform?