Disseny d'Arquitectura Segura al Núvol amb Infraestructura com a Codi (IaC)

Miguel Martínez Arbues

Màster Universitari en Ciberseguretat i Privadesa

Seguretat empresarial

Professors

Miguel Ángel Flores Terrón
Josep Jorba Estebe

08/01/2024

Universitat Oberta de Catalunya

FITXA DEL TREBALL FINAL

Títol del treball:	Disseny d'Arquitectura Segura al Núvol amb Infraestructura com a Codi (IaC) i Validació de Seguretat amb Terrascan
Nom de l'autor:	Miguel Martínez Arbues
Nom del consultor/a:	Miguel Ángel Flores Terrón
Nom del PRA:	Josep Jorba Esteve
Data de lliurament (mm/aaaa):	01/2024
Titulació o programa:	Màster Universitari en Ciberseguretat i Privadesa
Àrea del Treball Final:	Seguretat empresarial
Idioma del treball:	*Català*
Paraules clau	Infraestructura com a codi (IaC), seguretat al núvol i validació de seguretat.

Resum del Treball

Aquest treball aborda la importància de la seguretat i la infraestructura al núvol en un context en què les empreses confien cada vegada més en aquesta tecnologia per augmentar l'eficiència i reduir costos. Amb aquest objectiu, aquest treball utilitza la pràctica de la Infraestructura com a Codi (IaC) amb l'eina Terraform i la validació de seguretat amb Terrascan i Prowler

Aquest enfocament permet automatitzar la gestió d'entorns al núvol i garantir el compliment de les normatives de seguretat com l'Esquema Nacional de Seguridad (ENS). El treball conclou amb la documentació dels procediments i resultats, oferint una guia pràctica per als professionals en ciberseguretat i els responsables d'infraestructura al núvol.

Abstract

This work addresses the importance of security and cloud infrastructure in a context where businesses increasingly rely on this technology to increase efficiency and reduce costs. To this end, this work uses the practice of Infrastructure as Code (IaC) with the Terraform tool and security validation with Terrascan and Prowler.

This approach allows for the automation of cloud environment management and ensuring compliance with security regulations such as the Esquema Nacional de Seguridad (ENS). The work concludes with the documentation of the procedures and results, offering a practical guide for professionals in cybersecurity and cloud infrastructure managers.

Índex

Llista de figures

1. Introducció

1.1. Context i justificació del Treball

Cada vegada més, empreses i organitzacions confien en emmagatzemar, gestionar i proporcionar serveis i aplicacions crítiques al núvol. Aquesta tecnologia ofereix grans avantatges en termes com escalabilitat, flexibilitat i eficiència operativa. No obstant això, amb l'augment de la utilització al núvol, també s'ha incrementat la complexitat i els reptes en termes de seguretat i compliment normatiu.

La integració de la infraestructura com a Codi (IaC)[1] i el control de l'arquitectura segura, pot ser una estratègia molt beneficiosa per aquells organismes que vulguin migrar o crear la seva infraestructura al núvol.

- Automatització de la Seguretat: L'ús d'eines d'IaC permet a les organitzacions definir i configurar la infraestructura de manera programada i automatitzada. Les polítiques de seguretat i els controls es poden incorporar directament en el codi i poden ser part integral de la creació i implementació de recursos, assegurant que es compleixin les mesures de seguretat des de l'inici.

- Validació de Polítiques de Seguretat: Amb eines com terrascan[2] es pot validar la seguretat d'IaC mitjançant l'escaneig de codi i la identificació de possibles vulnerabilitats i problemes de seguretat. Aquesta validació es realitza abans que la infraestructura es posi en funcionament, la qual cosa permet corregir possibles problemes de seguretat abans que puguin ser explotats.

- Gestió de Canvis i Conformitat: Les organitzacions que busquen certificar-se, per exemple amb la ISO 27001[3], o complir amb els requeriments de l'Esquema Nacional de Seguridad (ENS)[4], han de demostrar que mantenen un control rigorós sobre els canvis a la infraestructura i que aquesta està en conformitat amb els requisits de seguretat establerts. L'IaC facilita la gestió de canvis mitjançant registres de versions i control de codi, mentre que terrascan ofereix una visió continua de la conformitat amb les polítiques de seguretat aplicades.

- Auditoria i Informes: Amb l'ús d'IaC i terrascan, les organitzacions poden generar informes detallats de seguretat que mostren el compliment de les polítiques i els controls de seguretat. Aquests informes son útils per a l'auditoria i la demostració del compliment de l'ENS davant de les autoritats reguladores.

- Resposta Ràpida a Amenaces: Tenir polítiques de seguretat i controls incorporats a l'IaC i validar-les amb terrascan permet una resposta ràpida a les amenaces. Quan es detecta una possible

vulnerabilitat, es pot prendre accions immediates per mitigar-la i mantenir la seguretat de la infraestructura.

Hi ha tot un seguit d'empreses i organismes, que amb data 5 de maig del 2024[(5)] estaran obligats a complir amb els requisits de l'Esquema Nacional de Seguridad (ENS). A dia d'avui, hi ha forces organismes, com poden ser ajuntaments o universitats públiques, que no estan certificats[(6)], aquesta guia pot ser molt útil per assolir aquest propòsit.

La combinació de la utilització de la Infraestructura com a Codi i el control de l'arquitectura segura amb eines com Terrascan proporciona un enfocament efectiu i eficient per assegurar-se que les arquitectures al núvol compleixin amb els requisits de seguretat establerts per normatives com l'ENS. Aquest enfocament permet la gestió de seguretat automatitzada, la validació de polítiques i la documentació del compliment, tots ells essencials per a la certificació en aquest context.

Amb aquest treball es vol obtenir una guia pràctica que permeti als professionals en ciberseguretat i als responsables d'infraestructura al núvol implementar, validar i mantenir arquitectures segures de manera eficaç i automatitzada, a l'hora que es garanteixen els requisits de seguretat establerts per normatives de seguretat com l'Esquema Nacional de Seguridad (ENS). Aquesta guia estarà basada en una metodologia que incorpora les millors pràctiques, eines i processos per aconseguir aquesta fita.

1.2. Objectius del Treball

Objectiu general

Dissenyar una arquitectura segura al núvol, utilitzant infraestructura com a codi (IaC). A l'hora, aquesta arquitectura ha d'estar validada amb controls específics, utilitzant l'eina terrascan. Aquests controls[(7)] estaran alineats amb els requisits de seguretat establerts per l'Esquema Nacional de Seguridad (ENS).

Objectius específics

- Explorar i conèixer diferents proveïdors de serveis al núvol.
- Triar un proveïdor al núvol per tal d'executar aquest treball.
- Estudiar les diferents eines per a utilitzar la infraestructura com a codi (IaC)
- Conèixer l'eina Terrascan així com les diferents possibilitats que ofereix per tal d'afegir diferents polítiques i controls.
- Analitzar els diferents controls i requisits que demana l'Esquema Nacional de Seguridad (ENS).

- Estudiar, analitzar i aplicar les diferents guies que proporciona CCN-Cert[8] sobre el proveïdor triat.

- Triar una arquitectura, simple però coherent per tal de poder dissenyar una arquitectura segura al núvol.

- Proporcionar una metodologia per implementar arquitectures al núvol de manera segura, incorporant les millors pràctiques i eines.

- Proporcionar una metodologia per validar la seguretat de les arquitectures al núvol, mitjançant l'ús d'eines com Terrascan.

- Proporcionar una metodologia per mantenir la seguretat de les arquitectures al núvol, mitjançant la gestió de canvis i la monitorització de la seguretat. Convertint la combinació de Terraform i Terrascan en una eina de pipeline devsecops[9]

1.3. Limitacions

- No es podran utilitzar les eines de pagament que ofereixen els diferents proveïdors de servei al núvol, només es comprovarà que al codi de Terraform estan integrades

- No es te previst la creació d'un script que convertiria la combinació de Terraform i Terrascan en una eina de pipeline devsecops

1.4. Impacte en sostenibilitat, ètic-social i de diversitat

Aquest Treball de final de Màster (TFM), de la mateixa manera que la Universitat Oberta de Catalunya (UOC), està altament compromès amb l'impacte en sostenibilitat, ètic-social i de diversitat.

L'any 2015, les Nacions Unides, va aprovar l'agenda 2030[10]. Aquesta agenda compta amb 17 Objectius de Desenvolupament Sostenible, que inclouen des de la eliminació de la pobresa fins al combat al canvi climàtic, l'educació, la igualtat de la dona, la defensa del medi ambient o el disseny de les nostres ciutats.

El disseny d'Arquitectura Segura al Núvol amb Infraestructura com a Codi (IaC) i Validació de Seguretat amb terrascan compleix perfectament amb varis d'aquests objectius:

Dins de la dimensió diversitat, gènere i drets humans, existeix una reducció a la desigualtat en l'accés a les eines, ODS 10, ja que les dues eines principals a utilitzar, Terraform i terrascan son de codi obert.

Per un altre costat també influirà a la dimensió comportament ètic i de responsabilitat social, ja que el projecte aborda aspectes relacionats amb la ciberseguretat i la construcció de sistemes tecnològics que promouen la justícia i la seguretat, ODS 16.

Dins de la dimensió sostenibilitat podríem parlar dels objectius relacionats amb l'energia assequible i neta, ODS 7, i de l'acció pel clima referent a la reducció de l'empremta de carboni i l'acció pel clima, ODS 13. Tal com podem veure en diferents articles[11], els tres principals proveïdors de servei al núvol: aws, azure i Google cloud, han estan aplicant polítiques per assolir aquests objectius els darrers anys.

Dins d'aquesta mateixa dimensió, aquest projecte implica la implementació d'infraestructura al núvol de manera innovadora i podria contribuir al desenvolupament d'infraestructures tecnològiques més sostenibles, ODS 9. Com que aquest projecte es centra en la gestió eficient de recursos al núvol també estaria dins del ODS 12, producció i consum responsable.

1.5. Estat de l'art

En els darrers anys, la computació en núvol s'ha convertit en una opció popular per a les empreses que busquen augmentar la seva eficiència i reduir els costos en tecnologia de la informació (TI)[12]. Aquesta tendència ha portat a l'aparició de nous conceptes i eines relacionades amb la seguretat i la infraestructura al núvol:

- Infraestructura com a Codi (IaC): La pràctica de la Infraestructura com a Codi (IaC) ha emergit com una solució per a automatitzar la gestió i el desplegament de recursos de la infraestructura de manera eficient i fiable. Els sistemes d'IaC, com Terraform, s'han establert com eines inestimables per simplificar la creació i el manteniment d'entorns complexes al núvol.

- Seguretat al Núvol: La seguretat s'ha convertit en un element importantíssim de la infraestructura i els serveis que les empreses despleguen al núvol. La preocupació principal es centra en la protecció de dades i aplicacions crítiques de les amenaces en ciberseguretat i el garantir el compliment de les estrictes regulacions i estàndards de seguretat com l'ENS.

  L'Esquema Nacional de Seguridad (ENS) és una peça fonamental per a les organitzacions que operen dins de l'àmbit de l'administració pública. Aquesta normativa imposa requisits i directrius estrictes per garantir la salvaguarda de dades. Altres regulacions com el Reglament General de Protecció de Dades (RGPD) també son de vital importància.

- Validació de Seguretat amb terrascan: terrascan, una eina dedicada a la seguretat d'IaC. Mitjançant l'anàlisi del codi d'IaC, terrascan identifica potencials vulnerabilitats i assegura que les polítiques de seguretat s'estiguin complint. La integració d'aquesta eina en les pràctiques de

seguretat és essencial per a garantir que l'entorn al núvol és segur i conforme amb els requisits.

- Prowler[13] és una eina de programari lliure que permet analitzar múltiples serveis i recursos desplegats a AWS (Amazon Web Services) de forma manual o automàtica. Amb aquesta eina es poden generar informes de seguretat.

  S'utilitza per identificar vulnerabilitats i fer recomanacions de seguretat en els entorns d'AWS. Aquesta eina és especialment útil per a empreses i organitzacions que utilitzen AWS i volen assegurar-se que compleixen amb els requeriments de l'ENS.

  Existeix documentació específica per comprovar el compliment de la infraestructura segons l'ENS utilitzant Prowler[14]

- Monitoratge, gestió i automatització: Les empreses han de monitorar i gestionar la seva infraestructura de núvol per garantir que estigui funcionant correctament i per identificar qualsevol problema o vulnerabilitat. L'automatització és clau per a la creació i el manteniment de la infraestructura al núvol.

1.6. Enfocament i mètode seguit

Aquest treball està pensat per tal d'ajudar a les organitzacions a l'hora d'enfrontar-se amb les necessitats i els reptes associats amb la seguretat i la infraestructura al núvol. Per aconseguir-ho s'utilitzarà la infraestructura com a codi (IaC) i per fer una correcta validació de la seguretat, es comprovarà el compliment normatiu a l'ENS.

Per aconseguir aquest propòsit, s'han dividit les diferents tasques en diferents processos:

- Recerca, disseny i configuració de l'entorn
- Implementació, validació, auditories i gestió de canvis
- Memòria final i defensa

Als primers sprints, s'investigarà sobre les diferents eines a utilitzar, IaC i terrascan així com en els principals proveïdors de serveis al núvol, Aws, Azure i Google Cloud. També caldrà fer una investigació exhaustiva sobre el compliment normatiu de l'ENS al núvol.

Seguidament caldrà planificar i dissenyar l'entorn: Triar el proveïdor al núvol, triar les eines a utilitzar i instal·lar les diferents aplicacions, i triar l'entorn, per exemple: nombre d'instàncies, característiques o tipus d'accés.

Els següents sprints es basaran amb la implementació, validació, auditories i gestió de canvis. Concretament caldrà Implementar l'entorn i les polítiques de seguretat, validant aquestes polítiques de seguretat utilitzant auditories i informes. Per finalitzar aquests sprints també es tindrà un control de la gestió de canvis així com una avaluació dels resultats.

Finalment es documentarà els procediments i resultats obtinguts a la memòria del treball així com tots els requeriments per la defensa del treball final.

1.7. Planificació del Treball

1.7.1. Administració del projecte

Per tal de dur a terme aquest projecte, s'ha utilitzat l'eina Trello[15]. Es tracta d'una eina de col·laboració que organitza els projectes en taulers[16]. Dins dels taulers s'organitza en targetes i columnes que faciliten seguir el progrés del projecte. Aquest programari disposa d'un calendari i permet fer un diagrama de Gantt.

Per poder treballar amb Trello he creat tres processos diferents:

- Backlog[17]: On tindrem una llista de treballs ordenats per prioritat. En el cas d'aquest projecte, i al tractar-se d'un projecte individual, l'ordre de prioritats és temporal.

- En procés: El treball en el que s'està treballant

- Completats: Llistes de feines completades

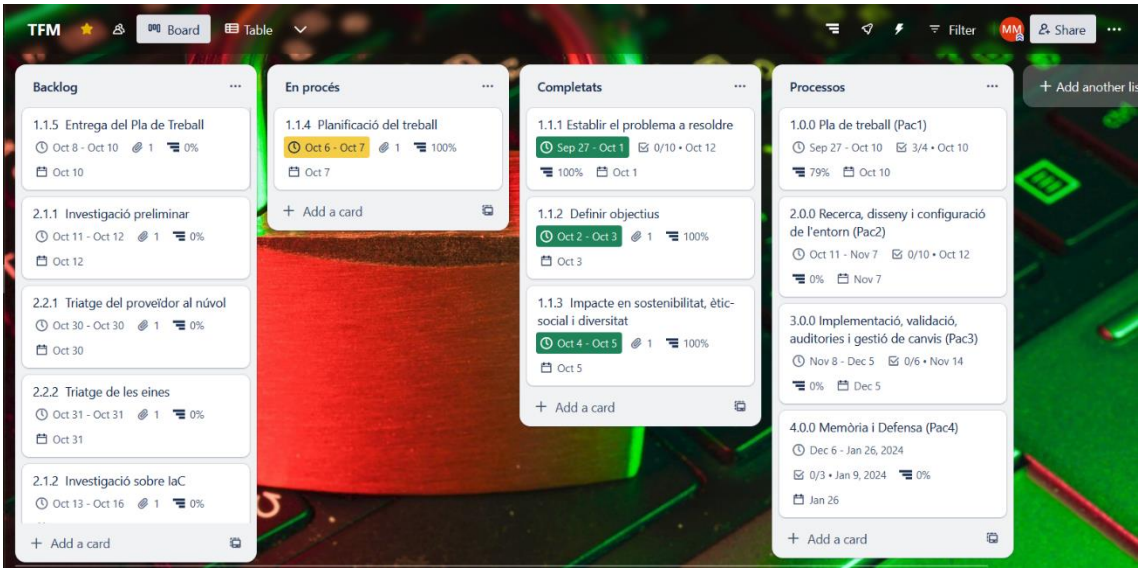


Figura 1: Desplegament del tauler Trello

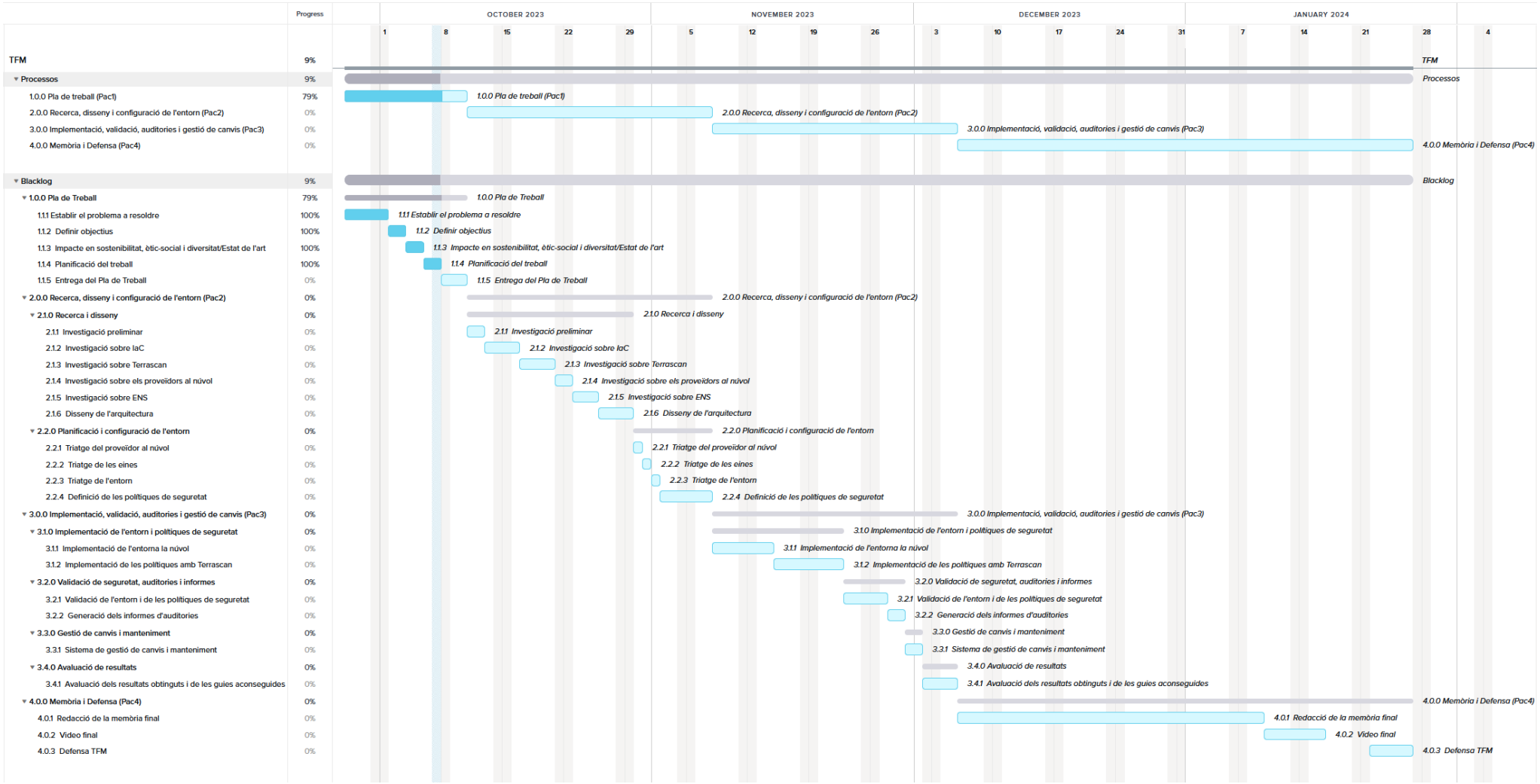
1.7.2. Llistat de feines

ID	FEINES	DATA INICI	DATA FI	DIES	COMPLETAT
1.0.0	**Pla de Treball (Pac1)**	27/9/2023	**10/10/2023**	14	80%
1.1.1	Establir el problema a resoldre	27/9/2023	1/10/2023	5	100%
1.1.2	Definir objectius	2/10/2023	3/10/2023	2	100%
1.1.3	Impacte en sostenibilitat, ètic-social i diversitat/ Estat de l'art	4/10/2023	5/10/2023	2	100%
1.1.4	Planificació del treball	6/10/2023	7/10/2023	2	100%
1.1.5	Entrega del Pla de Treball	8/10/2023	10/10/2023	3	0%
2.0.0	**Recerca, disseny i configuració de l'entorn (Pac2)**	11/10/2023	**7/11/2023**	28	0%
2.1.0	*Recerca i disseny*				
2.1.1	Investigació preliminar	11/10/2023	12/10/2023	2	0%
2.1.2	Investigació sobre IaC	13/10/2023	16/10/2023	4	0%
2.1.3	Investigació sobre Terrascan	17/10/2023	20/10/2023	4	0%
2.1.4	Investigació sobre els proveïdors al núvol	21/10/2023	22/10/2023	2	0%
2.1.5	Investigació sobre ENS	23/10/2023	25/10/2023	3	0%
2.1.6	Disseny de l'arquitectura	26/10/2023	29/10/2023	4	0%
2.2.0	*Planificació i configuració de l'entorn*				
2.2.1	Triatge del proveïdor al núvol	30/10/2023	30/10/2023	1	0%
2.2.2	Triatge de les eines	31/10/2023	31/10/2023	1	0%
2.2.3	Triatge de l'entorn	1/11/2023	1/11/2023	1	0%
2.2.4	Definició de les polítiques de seguretat	2/11/2023	7/11/2023	6	0%
3.0.0	**Implementació, validació, auditories i gestió de canvis (Pac3)**	8/11/2023	**5/12/2023**	28	0%
3.1.0	*Implementació de l'entorn i polítiques de seguretat*				
3.1.1	Implementació de l'entornal núvol	8/11/2023	14/11/2023	7	0%
3.1.2	Implementació de les polítiques amb Terrascan	15/11/2023	22/11/2023	8	0%
3.2.0	*Validació de seguretat, auditories i informes*				
3.2.1	Validació de l'entorn i de les polítiques de seguretat	23/11/2023	27/11/2023	5	0%
3.2.2	Generació dels informes d'auditories	28/11/2023	29/11/2023	2	0%
3.3.0	*Gestió de canvis i manteniment*				
3.3.1	Sistema de gestió de canvis i manteniment	30/11/2023	1/12/2023	2	0%
3.4.0	*Avaluació de resultats*				

3.4.1	Avaluació dels resultats obtinguts i de les guies aconseguides	2/12/2023	5/12/2023	4	0%
4.0.0	**Memòria i Defensa (Pac4)**	6/12/2023	**26/1/2023**	51	0%
4.0.1	Redacció de memòria final	6/12/2023	9/01/2024	34	0%
4.0.2	Vídeo final	10/1/2024	16/1/2024	6	0%
4.0.3	Defensa TFM	22/01/2024	26/01/2024	5	0%

Figura 2: Llistat de feines

1.7.3. Calendari



1.8. Breu sumari de productes obtinguts

Amb aquest projecte s'ha obtingut una implementació d'una infraestructura al núvol segura utilitzant Terraform per a l'automatització. S'ha configurat una xarxa VPC amb subxarxes específiques, instàncies EC2, i polítiques IAM robustes.

A més, s'ha realitzat una auditoria de seguretat amb Terrascan i Prowler, assegurant la conformitat amb els estàndards ENS 2022. Els productes finals inclouen scripts de Terraform, documents de configuració, i informes d'auditoria.

1.9. Breu descripció dels altres capítols de la memòria

Els diferents capítols de la memòria inclouen:

- Introducció: Inclou el context, justificació, objectius, limitacions, impacte ètic-social i en sostenibilitat, estat de l'art, enfocament i mètode, així com la planificació del projecte.

- Conceptes i Tecnologia: Aquesta secció cobreix l'arquitectura al núvol, proveïdors de Infraestructura com a Servei (com AWS, Azure, Google Cloud, entre d'altres), Infraestructura com a codi (IaC) amb eines com Terraform i AWS CloudFormation, i temes de seguretat al núvol.

- Arquitectura de la Infraestructura: S'aborda el flux i els controls sobre la infraestructura, incloent la creació, manteniment, proves, i implementació.

- Resultats: Presenta els resultats obtinguts, incloent codi Terraform, funcionalitats com la VPN, scripts de còpies de seguretat i l'ús d'eines com Terrascan i Prowler.

- Conclusions i Treballs Futurs: Ofereix conclusions sobre les eines i la infraestructura utilitzada, i contempla possibles treballs futurs.

- Glossari: Proporciona definicions de termes tècnics utilitzats en el treball.

- Bibliografia: Llista les referències bibliogràfiques utilitzades.

- Annexos: Inclou informació addicional, com la configuració de l'entorn de treball i la instal·lació d'eines específiques. També inclou una secció dedicada a la configuració de VPNs entre diferents instàncies.

2. Conceptes i Tecnologia

2.1. Arquitectura al núvol

L'arquitectura al núvol[18], és una forma d'organitzar, emmagatzemar i gestionar dades i recursos informàtics a través d'Internet. Aquesta tecnologia ha experimentat un gran creixement i transformació en les últimes dues dècades, i ha canviat la forma com les empreses i els individus emmagatzemen, processen i accedeixen a la informació.

Les principals característiques de l'arquitectura al núvol inclouen la virtualització, l'escalabilitat, la flexibilitat, la disponibilitat i la seguretat. Els serveis al núvol es poden classificar en tres models principals: IaaS (Infraestructura com a Servei), PaaS (Plataforma com a Servei) i SaaS (Programari com a Servei).

La Infraestructura com a Servei (IaaS) és un dels models principals dins de l'arquitectura en el núvol. Aquest model permet als usuaris accedir i gestionar recursos informàtics essencials, com ara servidors virtuals, emmagatzematge, xarxes o màquines virtuals, mitjançant proveïdors de serveis en línia. Aquests recursos informàtics es poden obtenir de forma escalable i flexible.

2.2. Proveïdors de Infraestructura com a Servei

Hi ha diversos proveïdors d'Infraestructura com a Servei (IaaS) els quals ofereixen una àmplia gamma de serveis i solucions.

2.2.1. Amazon Web Services (AWS)

AWS[19] és un dels proveïdors d'IaaS més grans i àmpliament utilitzat. Ofereix una àmplia gamma de serveis, com poden ser màquines virtuals, emmagatzematge, bases de dades, xarxes o eines de desenvolupament. Aquest proveïdor ofereix diversos serveis a diferents regions, així com una gran facilitat d'escalabilitat i flexibilitat. Disposa d'una gran comunitat i suport.

2.2.2. Microsoft Azure

Azure[20] és la plataforma al núvol de Microsoft i ofereix una àmplia integració amb els seus productes i eines. Proporciona una gran varietat de serveis, incloent màquines virtuals, bases de dades, serveis de xarxa i eines de desenvolupament. Ofereix suport a múltiples llenguatges i frameworks.

2.2.3. Google Cloud Platform (GCP)

GCP[21] és la plataforma al núvol de Google i es destaca per les seves capacitats d'intel·ligència artificial i aprenentatge automàtic. Ofereix recursos com màquines virtuals, emmagatzematge i eines d'anàlisi de dades. Disposa de

menys recursos en comparació amb AWS o Azure i te menys presència en empreses tradicionals.

2.2.4. IBM Cloud

IBM Cloud[22] proporciona serveis d'IaaS i PaaS, i destaca en l'oferta de solucions per a empreses, incloent seguretat i blockchain. Ofereix una àmplia gamma de recursos informàtics.

2.2.5. Alibaba Cloud

Alibaba Cloud[23] és el proveïdor líder a l'Àsia i ofereix una gamma completa de serveis al núvol, incloent màquines virtuals, serveis de xarxa i bases de dades. És menys conegut a nivell global.

2.2.6. DigitalOcean

És conegut per la seva simplicitat, està centrada en desenvolupadors. DigitalOcean[24] ofereix màquines virtuals i recursos senzills per a un desplegament ràpid d'aplicacions. Disposa de menys recursos i serveis avançats i és menys indicat per a empreses grans o projectes complexos.

2.2.7. Vultr

Vultr[25] és un altre proveïdor d'IaaS que es concentra en màquines virtuals amb preus competitius i una àmplia selecció de ubicacions de servidors. És fàcil d'utilitzar per a petits projectes i menys orientat a empreses. Disposa de menys recursos que els altres proveïdors.

Per elaborar aquest treball, s'ha triat Amazon Web Services (AWS) com a proveïdor principal, ja que proporciona una plataforma robusta per a l'aprenentatge i la implementació de la Infraestructura com a Codi (IaC). AWS te una gran quota de mercat a nivell mundial.

Companyia	2022 Ingressos	Mercat 2022 Compartir (%)	2021 Ingressos	Mercat 2021 Compartir (%)	2021-2022 Creixement (%)
Amazon	48,126	40.0	35,380	38.1	36.0
Microsoft	25,858	21.5	19,153	20.6	35.0
Grup Alibaba	9,281	7.7	9,060	9.8	2.4
Google	9,072	7.5	6,433	6.9	41.0
Huawei	5,249	4.4	4,190	4.5	25.3
Altres	22,746	18.9	18,565	20.0	22.5
Total	**120,333**	**100**	**92,782**	**100**	**29.7**

Figura 3: Quota de mercat mundial de serveis al Núvol Públic Iaas (milions de dòlars nord-americans) Font: Gartner(juliol 2023)[27]

Altres proveïdors com Azure o GCP també ofereixen recursos i informació per ajudar els clients a complir amb les normatives de seguretat, incloent-hi l'ENS, però AWS destaca per la seva àmplia documentació i eines específiques[28].

Tot i que el treball es basarà en la plataforma AWS, també es proporciona una comparativa dels serveis dels tres principals proveïdors en aquesta adreça web: https://cloud.google.com/docs/get-started/aws-azure-gcp-service-comparison?hl=es-419

2.2.8. Responsabilitat compartida

Quan es parla de seguretat al núvol, cal destacar que aquesta responsabilitat és compartida, existeix una divisió de responsabilitats entre el proveïdor de serveis al núvol i els seus clients. Aquesta divisió s'estableix per garantir que les dades i els sistemes al núvol estiguin protegits de manera adequada. En el cas d'AWS, la responsabilitat compartida es divideix en dues àrees principals: la seguretat del núvol i la seguretat al núvol.

Seguretat del núvol (Security of the Cloud): AWS és responsable de la seguretat de la infraestructura física i la virtualització subjacent que dona suport als seus serveis al núvol. Això inclou aspectes com la seguretat dels centres de dades, la xarxa, l'emmagatzematge o la disponibilitat dels serveis. AWS es fa càrrec de garantir que els seus centres de dades estiguin protegits contra amenaces físiques i de xarxa, i que els serveis estiguin disponibles de manera fiable.

Seguretat al núvol (Security in the Cloud): Els clients d'AWS son responsables de garantir la seguretat de les seves aplicacions, dades i sistemes que utilitzen els serveis d'AWS. Això inclou configurar i gestionar els tallafocs, sistemes operatius, aplicacions, xarxes i accés a les dades. Els clients també han d'aplicar les millors pràctiques de seguretat, com la gestió de la identitat i l'accés, el xifrat de dades, la monitorització de la seguretat i la gestió de les actualitzacions.

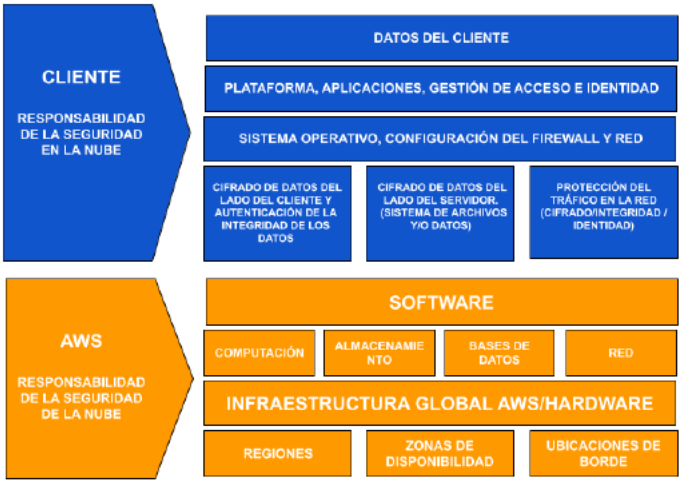


Figura 4: Representació del model de responsabilitat compartida en AWS. Extret de la CCN-STIC 887A Guía de configuración segura AWS[29]

2.3. Infraestructura com a codi (IaC)

La Infraestructura com a Codi (IaC) és una tècnica que consisteix en gestionar i configurar la infraestructura de tecnologia d'una organització utilitzant codi i eines d'automatització en lloc de fer aquestes tasques de manera manual.

La IaC tracta la infraestructura, que inclou servidors, xarxes, emmagatzematge i altres recursos, com si fos programari, el que permet una gestió més eficient i escalable.

Actualment existeixen diferent eines IaC:

2.3.1. Terraform[30]:

És una eina de codi obert desenvolupada per HashiCorp. Pot gestionar infraestructures multi-núvol (AWS, Azure, Google Cloud,...) mitjançant codi declaratiu.

En un codi declaratiu, s'indiquen quins recursos i configuracions es volen crear, però no s'especifica com s'han de crear o configurar exactament aquests recursos. Terraform s'encarrega de determinar com dur a terme aquesta creació i configuració.

Terraform disposa de diferents comandes:

- Terraform init: S'utilitza per inicialitzar un directori de treball de Terraform. Al treballar en un nou directori de projecte o en el cas d'haver canviat el fitxer de configuració, cal executar aquesta instrucció per descarregar les dependències del proveïdor i preparar l'entorn de treball.

- Terraform plan: Amb aquesta instrucció es pot veure una representació de com canviarà la infraestructura, permet detectar errors o canvis no desitjats abans d'aplicar-los realment.

- Terraform apply: Aplica la configuració de Terraform a la infraestructura real. Crea, actualitza o elimina els recursos segons la definició del fitxer de configuració.

- Terraform destroy: Elimina tots els recursos gestionats per Terraform que estan definits al fitxer de configuració.

- Terraform validate: Comprova la sintaxi i la validesa dels arxius de configuració de Terraform. Permet detectar errors sense executar una instrucció plan o apply.

- Terraform refresh: Actualitza l'estat local de Terraform amb l'estat actual de la infraestructura. És útil per sincronitzar l'estat amb els canvis que es fan fora de Terraform.

- Terraform state: Amb aquesta instrucció s'inspecciona l'estat actual dels recursos gestionats per Terraform i realitzar operacions relacionades amb l'estat.

Permet crear mòduls reutilitzables[31], el que facilita la creació de plantilles per a la infraestructura que es poden utilitzar en diferents projectes i es pot gestionar mitjançant sistemes de control de versions com Git, la qual cosa permet realitzar seguiment de canvis i col·laborar en equips.

El mòduls son un conjunt de recursos i configuracions que es poden encapsular i cridar en altres configuracions de Terraform. Per treballar amb mòduls, hi ha una carpeta principal on hi haurà un codi principal, main.tf, i un codi amb les variables principals, variables.tf.

A partir d'aquesta carpeta, hi haurà una carpeta "modules" on hi haurà una altra carpeta amb els mòduls que es vulguin crear. Dins de cada carpeta "modules", tindrem una altra carpeta per a cada mòdul que es vulgui crear, i dins de cada carpeta, els codis main.tf i variables.tf de cada mòdul.

A mode d'exemple, es mostra una infraestructura senzilla amb mòduls reutilitzables per a la VPC, les subxarxes, els grups de seguretat i les instàncies:

```
main.tf
variables.tf
modules/
  vpc/
    main.tf
    variables.tf
  subnets/
    main.tf
    variables.tf
  security_group/
    main.tf
    variables.tf
  instances/
    main.tf
    variables.tf
```

Figura 5: Exemple estructura en mòduls

Així doncs, cada mòdul disposarà del seu codi main.tf i variables.tf, com exemple simplificat la carpeta instances tindria els següents codis:

Figura 6: Exemple main.tf Terraform

Figura 7: Exemple variables.tf Terraform

A l'executar Terraform apply en el directori principal, Terraform crearà una instància EC2 mitjançant el mòdul example_module i mostrarà la sortida definida pel mòdul com a sortida global.

Terraform disposa d'una comunitat gran i activa on es poden trobar molts recursos i suport disponibles en línia, com mòduls predefinits o documentació. És compatible amb Terrascan.

Per un altre costat és difícil de depurar i no disposa d'interfície gràfica d'usuari[32].

2.3.2. AWS CloudFormation

CloudFormation[33] és una eina de gestió de la infraestructura com a codi (IaC) que es proporciona a través d'Amazon Web Services (AWS). Està estretament integrat a AWS. Gestiona recursos d'AWS de manera eficient, incloent-hi instàncies EC2, grups d'Auto Scaling o bases de dades RDS. Permet gestionar tots els recursos de la infraestructura com a codi, incloent-hi xarxes, IAM (Identity and Access Management), grups de seguretat i recursos d'aplicació.

Es poden crear plantilles de CloudFormation que es poden reutilitzar per a desplegaments similars en diferents entorns i així estalviar temps i evitar possibles errors. Es poden utilitzar controls de versions amb les plantilles de CloudFormation, les quals permeten rastrejar canvis i revertir a versions anteriors si és necessari.

Està dissenyat específicament per a AWS, de manera que està altament lligat a aquest proveïdor al núvol. No és de codi obert.

2.3.3. Google Cloud Deployment Manager

És una eina proporcionada per Google Cloud per gestionar la infraestructura com a codi (IaC) i automatitzar el desplegament d'entorns d'infraestructura a Google Cloud Platform (GCP)[34]. Aquest programari permet la creació de plantilles de desplegament en format YAML o Python.

Utilitza un enfocament declaratiu similar a Terraform. Aquest enfocament simplifica la creació de plantilles d'infraestructura reutilitzables.

Permet definir polítiques de seguretat i així garantir que la infraestructura es desplegui de manera segura i compleixi els requisits normatius. No és de codi obert.

2.3.4. Azure Resource Manager (ARM) Templates[35]

Es tracta d'una eina que es fa servir per definir i gestionar recursos a Microsoft Azure com a codi.

Les plantilles ARM utilitzen una descripció declarativa de la infraestructura les quals poden ser reutilitzables per a diferents implementacions i entorns d'Azure.

Disposa de controls de versions per rastrejar canvis i revertir-los a versions anteriors si és necessari. Les plantilles ARM ofereixen eines integrades per a la depuració de plantilles, el que facilita la identificació i la correcció d'errors. És una eina de codi obert.

2.3.5. Ansible

Ansible[36] és una eina de codi obert que permet la gestió de configuració i l'automatització de tasques. Utilitza un llenguatge de marcatge simple i llegible per a humans, anomenat YAML, per descriure les tasques a realitzar. Aquesta eina és àmpliament utilitzada per a la gestió de configuració, el desplegament d'aplicacions i l'automatització d'infraestructures.

Per realitzar aquest projecte s'ha triat l'eina IaC Terraform ja que ofereix una gran flexibilitat i és compatible amb diversos proveïdors de núvol importants, com AWS, Azure o Google Cloud. Es pot utilitzar Terraform per gestionar recursos en múltiples plataformes, permetent una major agilitat i adaptabilitat.

Gaudeix d'una àmplia comunitat d'usuaris i desenvolupadors. Per tant una gran quantitat de recursos, documentació i mòduls predefinits disponibles per a les diferents necessitats del projecte.

Proporciona un control granular sobre els recursos de la infraestructura. Amb les comandes *terraform plan* i *terraform apply*, es pot veure com es desplegarà la infraestructura abans d'aplicar els canvis, la qual cosa millora la seguretat. És de codi obert.

2.4. Terrascan

Terrascan[37] és una eina d'escaneig de seguretat d'infraestructura com a codi (IaC). És de codi obert i permet als desenvolupadors detectar problemes de seguretat en les seves configuracions d'IaC abans que es despleguin.

Terrascan s'integra amb diferents proveïdors de núvol, com ara AWS, Azure o Google Cloud, i escaneja els recursos i les plantilles d'infraestructura per identificar problemes de seguretat, vulnerabilitats i desviacions de les polítiques de seguretat.

Proporciona més de 500 polítiques prèvies per analitzar IaC amb estàndards de polítiques comuns com el CIS Benchmark[38]. Terrascan es pot integrar en pipelines CI/CD per fer complir les millors pràctiques de seguretat[37][40].

Es pot instal·lar localment o executar-se mitjançant Docker[41]. Terrascan aprofita el motor de l'Open Policy Agent (OPA), el qual permet crear polítiques personalitzades mitjançant el llenguatge de consulta Rego[42].

Aquestes polítiques ajuden a fer complir els requisits específics de compliment i seguretat.

A mode d'exemple, una política personalitzada creada per Terrascan la qual prohibiria la creació de qualsevol instància EC2, volum EBS o instantània

EBS que no estigui xifrada estaria formada per dos arxius, un arxiu Rego (deny_unencrypted.rego) i un arxiu JSON (deny_unencrypted.json) que definiria la política:

```
package main
deny_unencrypted[retVal] {
    input.resource.aws_instance[_].instance.tenancy == "default"
    input.resource.aws_instance[_].instance.ebs_optimized == false
    retVal = {
        "resource_type": "aws_instance",
        "resource_name": input.resource.aws_instance[_].name,
        "category": "Encryption and Key Management",
        "severity": "HIGH",
        "violation_description": "La instància EC2 no està xifrada",
        "rule_name": "deny_unencrypted"
    }
}

deny_unencrypted[retVal] {
    input.resource.aws_ebs_volume[_].encrypted == false
    retVal = {
        "resource_type": "aws_ebs_volume",
        "resource_name": input.resource.aws_ebs_volume[_].name,
        "category": "Encryption and Key Management",
        "severity": "HIGH",
        "violation_description": "El volum EBS no està xifrat",
        "rule_name": "deny_unencrypted"
    }
}

deny_unencrypted[retVal] {
    input.resource.aws_ebs_snapshot[_].encrypted == false
    retVal = {
        "resource_type": "aws_ebs_snapshot",
        "resource_name": input.resource.aws_ebs_snapshot[_].name,
        "category": "Encryption and Key Management",
        "severity": "HIGH",
        "violation_description": "La instantània EBS no està xifrada",
        "rule_name": "deny_unencrypted"
    }
}
```

Figura 8: deny_unencrypted.rego

```
{
  "name": "deny_unencrypted",
  "file": "deny_unencrypted.rego",
  "description": "Aquesta política prohibeix la creació de qualsevol instància EC2, volum EBS
o instantània EBS que no estigui xifrada.",
  "id": "AC_AWS_9999",
  "severity": "HIGH",
  "policy_type": "aws",
  "resource_type": "aws_instance",
  "category": "Infrastructure Security",
  "version": 1
}
```

Figura 9: deny_unencrypted.json

Aquests dos arxius es poden col·locar dins de la carpeta de polítiques de Terrascan o en una carpeta diferent i especificar-ho amb l'opció -p o --policy-path al executar Terrascan.

2.5. Seguretat al núvol

La seguretat al núvol és una preocupació crítica per a les empreses i organitzacions que emmagatzemen i gestionen dades a través de serveis al núvol. La seguretat al núvol és una combinació de mesures tecnològiques i

pràctiques de gestió que han de ser implementades i mantingudes per a protegir les dades i els recursos al núvol contra amenaces i vulnerabilitats.

Les consideracions principals per a garantir la seguretat en un entorn de núvol:

- Accés i autenticació segura: S'han d'establir polítiques d'autenticació forta i control d'accessos per assegurar que només els usuaris autoritzats puguin accedir als recursos al núvol. Aquest projecte no es centrarà en la seguretat per l'accés i l'autenticació segura. AWS està certificada a l'ENS fins al 5/5/2024 amb categoria alta i com podem al seu certificat(43), el servei Identity and Access Management (IAM) també està inclòs. Així doncs, aquest projecte es centrarà en la creació dels diferents accessos i autentificacions utilitzant aquest servei.

- Gestió d'identitats i accessos (IAM): Amb polítiques d'IAM es controlen els permisos d'accés dels usuaris i les seves capacitats d'administració.

- Xifrat de dades: Totes les dades han d'estar xifrades en trànsit i en repòs. Això protegeix la informació de ser interceptada o accedida sense autorització.

- Monitoratge i auditoria: Cal establir sistemes de monitoratge i auditoria que permetin identificar i respondre a possibles amenaces i activitats sospitoses.

- Protecció contra amenaces: Cal implementa solucions de seguretat com a serveis antivirus, antimalware i firewalls per a protegir-se contra amenaces.

- Gestió de vulnerabilitats: Realitza proves de seguretat i actualitzacions periòdiques del sistema per a identificar i solucionar vulnerabilitats.

- Pla de contingència i recuperació: Desenvolupa un pla de contingència i recuperació per assegurar la disponibilitat i integritat de les dades en cas d'incidents o fallades.

- Conformitat normativa: Assegura que les pràctiques de seguretat al núvol compleixin amb les regulacions i normatives. En el cas d'aquest projecte es provarà de complir amb l'ENS.

- Educació i formació: Proporcionar formació als empleats i usuaris sobre les millors pràctiques de seguretat en la núvol i els riscos associats.

  És important entendre que la seguretat al núvol és una responsabilitat compartida entre el proveïdor i l'usuari, i requereix una atenció constant i actualitzada.

2.5.1. Esquema Nacional de Seguridad (ENS) amb AWS

L'Esquema Nacional de Seguridad (ENS)[(4)] és un conjunt de directrius i mesures que estableixen les bases per a la protecció de la informació i la ciberseguretat en les administracions públiques espanyoles.

El seu objectiu és assegurar la confidencialitat, integritat i disponibilitat de la informació de les institucions governamentals. L'ENS proporciona un marc de referència per a establir polítiques de seguretat, identificar riscos i adoptar les mesures necessàries per protegir les dades i els sistemes d'informació.

Aquest projecte, busca que la infraestructura creada compleixi amb els requisits específics de compliment i seguretat especificats a l'ENS. AWS està certificada amb els requeriments a l'ENS, cosa que facilitarà part del procés de compliment d'aquest projecte amb l'ENS.

Així doncs un cop configurat el compte i els mètodes de pagament, caldrà tenir en compte:

Arquitectura segura: Es recomana utilitzar serveis com Amazon VPC, AWS WAF i AWS Shield Advanced per protegir la infraestructura i les aplicacions contra atacs.

Control d'accés: És important configurar adequadament els permisos i rols d'IAM per garantir que només les persones autoritzades tinguin accés als recursos d'AWS.

Segregació de funcions i tasques: S'han d'establir polítiques i controls per separar les responsabilitats i limitar l'accés als recursos segons les necessitats de cada usuari.

Gestió de la capacitat: És necessari dimensionar i administrar adequadament els recursos d'AWS per garantir un rendiment òptim i evitar problemes de capacitat.

Seguretat de les dades: Cal utilitzar xifrat per protegir les dades en repòs i en trànsit, així com implementar pràctiques de còpia de seguretat i recuperació de dades.

Monitoratge i detecció d'anomalies: S'han d'utilitzar eines com Amazon GuardDuty i Amazon Inspector per detectar i respondre a esdeveniments de seguretat i vulnerabilitats.

2.5.2. Prowler

Prowler[13] és una eina de software lliure que s'utilitza per analitzar serveis i recursos desplegats en AWS (Amazon Web Services). Es possible identificar vulnerabilitats i recomanacions de seguretat en entorns d'AWS. Permet generar informes de seguretat en diferents formats.

En aquest projecte s'utilitzarà Prowler des de la instància de seguretat. Aquest programa serà executat per un usuari (usSeguretat) el qual:

a. L'usuari ha de pertànyer al grup de seguretat, aquest grup disposa d'un seguit de polítiques i permisos, com son SecurityAudit y ViewOnlyAccess entre altres, amb les quals podran executar Prowler[14]. En aquest enllaç[44] és possible veure tots els permisos i per enviar recomanacions a AWS Security Hub també caldrà afegir aquests permisos[45].

b. La instància EC2 de seguretat ha de tenir instal·lat Prowler: Per instal·lar Prowler és necessari tindre Python 3.9 o superior amb el gestor de paquets PIP. Amb la instrucció *pip install prowler* s'instal·larà Prowler i totes les seves dependències.

c. Cal configurar l'usuari amb aws configure on demanarà la clau i key de l'usuari. En aquest cas la clau i la key de l'usuari usSeguretat s'ha creat directament a la web de AWS

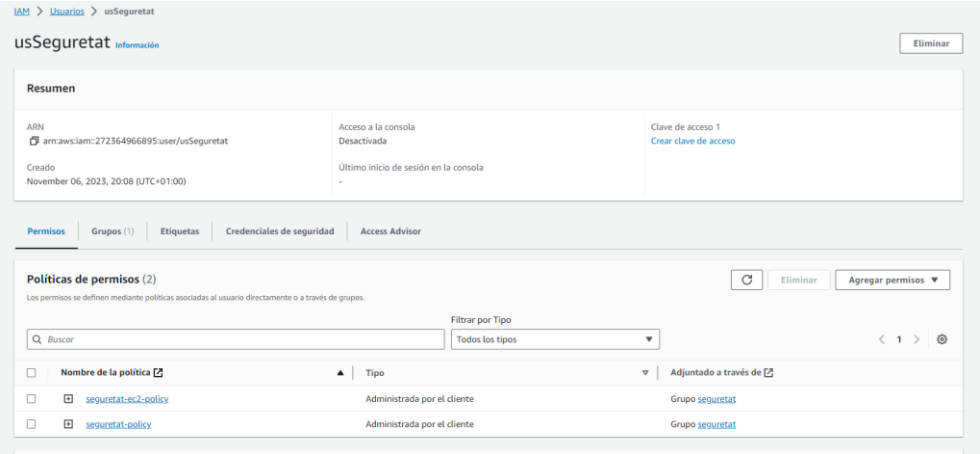


Figura 10: Credencials usSeguretat a AWS

Aquestes credencials, igual que tota la infraestructura creada amb IaC, s'han creat des d'un usuari "master", el qual disposa de més privilegis, és l'administrador de la infraestructura.

d. Un cop tot instal·lat i configurat, amb la instrucció *prowler --compliance ens_rd2022_aws* es revisarà el grau de compliment de la infraestructura a l'ENS

Els informes generats proporcionen una visió detallada dels resultats dels controls de seguretat realitzats per Prowler en els entorns d'AWS. Aquests informes permeten analitzar l'estat de la seguretat de la infraestructura i prendre decisions per millorar la postura de seguretat al núvol.

Prowler pot ser desplegat de forma automàtica o manual i també pot enviar recomanacions a AWS Security Hub per a una millor gestió de la seguretat. En aquest projecte es desplegarà de forma manual però s'automatitzarà amb un script.

Per generar informes de seguretat amb Prowler de forma automàtica, es pot crear un script de shell o un script de Python. Aquest script s'executarà des de la instància EC2 de seguretat, utilitzarà les credencials d'AWS del grup de seguretat i executar Prowler. Utilitzant l'eina de programació de tasques del sistema operatiu de les instàncies, cron[46], es programarà el script per que s'executi cada dia a una hora determinada.

Figura 11: Script bash de Prowler

Amb aquesta eina és possible l'automatització del compliment de les mesures establertes per l'ENS.

3. Arquitectura de la infraestructura

Tenint en compte tots els requisits esmentats, s'ha optat per una arquitectura que consta de:

- Tres instàncies: una on es guardarà tota la informació subministrada pels ciutadans, una altra, la qual serà utilitzada pels funcionaris, per poder treballar tant amb les dades subministrades pels ciutadans com amb les dades generades per ells. Finalment es crearà una instancia pel grup de seguretat on s'executarà l'eina Prowler.

- Tres grups d'usuaris:
    a. El grup de Ciutadans, el qual agruparà tots el usuaris ciutadans.
    b. El grup de Funcionaris, el qual agruparà tots els usuaris funcionaris
    c. El grup de Seguretat, els quals tindran la responsabilitat de supervisar i controlar tota la infraestructura. Agruparà tots els usuaris seguretat.

A partir d'aquests grups d'usuaris, es crearan usuaris individuals. Cadascun amb els seus permisos pertinents assignats a cada grup. En aquest projecte, s'anomenaran usCiutada, usFuncionari i usSeguretat.

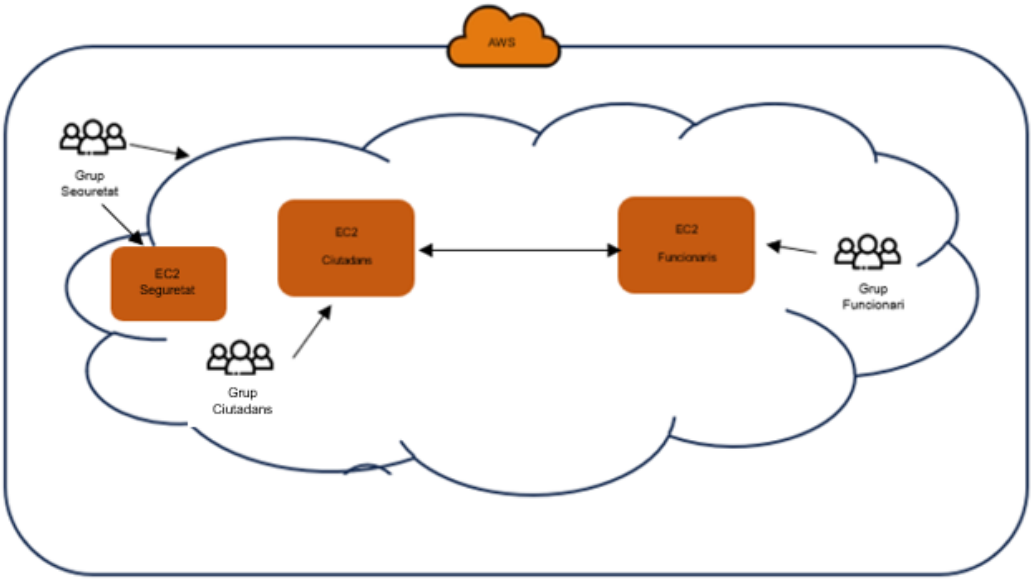


Figura 12: Infraestructura

Aquesta infraestructura podria ser, de forma simplificada, la d'un departament d'un ajuntament. Els ciutadans poden carregar documentació d'un impost pagat o números de compte a l'hora de domiciliar un impost, en una base de dades instal·lada en una instància EC2 Ciutadans. Cada ciutadà tindrà una política associada que li permetrà fer aquest acció.

Els funcionaris de l'ajuntament treballen en una base de dades EC2 Funcionaris, on poden afegir informació. A més, aquesta instància pot obtenir informació de la instància Ciutadans com a consultes a la seva base de dades. És important assegurar-se de controlar i gestionar adequadament l'accés a aquestes dades, ja que podria ser una possible preocupació de seguretat i de privadesa.

El grup de Seguretat comprovarà tota la infraestructura i realitzarà diferents auditories i controls sobre aquesta infraestructura.

Aquesta mateixa infraestructura es podria replicar a altres departaments fent petites modificacions al codi, per exemple caldria canviar els noms de les instàncies, grups, usuaris... canviar les configuracions de les xarxes o dels segurity group.

El fet de que Terraform treballi amb mòduls facilitaria molt aquesta feina.

Així doncs, si es crea un anàlisi de riscos de seguretat sobre aquesta infraestructura tindríem:

Actius Importants

a. Dades dels ciutadans i funcionaris emmagatzemades a les instàncies EC2.
b. Configuracions i polítiques d'IAM per als usuaris i grups.
c. Comunicacions entre les instàncies.
d. Dades de còpies de seguretat.
e. Els sistemes i eines de monitorització i gestió de la infraestructura

Amenaces Possibles

a. Accés no autoritzat a les dades dels ciutadans o funcionaris.
b. Fuita de dades a causa d'errors de configuració o atacs.
c. Interrupció de servei o indisponiblitat de les instàncies.
d. Accés no autoritzat a les polítiques d'IAM.
e. Violació de la privadesa de les comunicacions.
f. Atacs cibernètics com atacs DDoS o intents de phishing.
g. Pèrdua de dades a causa de fallades en les còpies de seguretat.

Per mitigar aquests riscos, considerar aquesta infraestructura segura (apartat 2.5 d'aquest treball) i complir amb els requeriments de l'ENS, cal tenir en compte una sèrie de consideracions:

- Gestió de la capacitat: És necessari dimensionar i administrar adequadament els recursos d'AWS per garantir un rendiment òptim i evitar problemes de capacitat. Aquest apartat queda fora d'aquest projecte, cada institució que vulgui aprofitar aquest treball haurà de comprovar quina és la capacitat que els hi caldrà. Per gestionar la capacitat és molt recomanable la utilització de AWS Auto Scaling.

- Els usuaris han d'estar classificats en grups i amb polítiques específiques per a cada grup. Els usuaris es crearan utilitzant el servei AWS IAM.

- Les instàncies han d'estar xifrades (AWS KMS) i han de tenir còpies de seguretat (AWS ELB). Les còpies també han de ser xifrades. L'ús de tres instàncies assegura una segregació a nivell d'instància.

  AWS i Terraform proporcionen varies maneres de crear les còpies de seguretat de les instàncies, en aquest cas concret s'ha creat un volum EBS i a partir d'aquest volum s'han creat instantànies (snapshot).

  Aquesta manera de crear còpies de seguretat te certs avantatges i desavantatges respecte altres manera de fer les còpies com podria ser per exemple AWS Backup.

Avantatges

a. És més granular[47]: Permet fer còpies de seguretat de volums individuals o instantànies de tota la instància, segons les necessitats específiques

b. Flexibilitat[47]: Permet copiar instantànies entre regions d'AWS utilitzant la funció de còpia d'instantànies d'Amazon EBS

c. Cost reduït[47]: Les instantànies d'Amazon EBS només emmagatzemen els canvis realitzats des de l'última instantània, el que redueix els costos d'emmagatzematge

Desavantatges

a. Complexitat: La gestió de còpies de seguretat de volums i instantànies pot ser complexa, especialment en entorns amb moltes instàncies

b. Manca de funcionalitats avançades[48]: No ofereix les mateixes funcionalitats avançades que AWS Backup, com la gestió de polítiques de retenció i la programació de còpies de seguretat

Aquest darrer punt és altament important, per solucionar-ho el grup de seguretat disposa de dues opcions, la primera és utilitzant el servei AWS Data Lifecycle Manager (DLM)[49] per automatitzar la creació i la gestió de snapshots de forma programada.

L'altre opció consisteix en crear un script que executi les còpies de seguretat de les instàncies. Aquests script es podria programar amb l'eina de programació de tasques del sistema operatiu, cron[46] a la instància de seguretat. S'ha triat aquesta segona opció ja que DLM te un cost associat al seu servei.

- S'utilitzaran diferents subxarxes i taules d'enrutament per a cada instància amb els seus grups de seguretat corresponents. L'ús de dues subxarxes assegura la segregació a nivell de xarxa.

- Les comunicacions entre les instàncies han de ser segures. La millor manera de que la comunicació sigui segura és utilitzant una VPN.

AWS i Terraform proporcionen varies maneres de crear la VPN. En aquest cas concret, es crearà la VPN mitjançant AWS Site-to-Site VPN, aquesta és una opció que ofereix AWS per simplificar la configuració de VPN.

A diferència de la configuració tradicional amb Terraform, AWS Site-to-Site VPN tot i que pot ser menys flexible en alguns aspectes, és una opció efectiva i ofereix una configuració senzilla sense necessitat de detalls específics.

- S'utilitzaran segurity groups per assegurar la connexió entre grups d'usuari i les seves instàncies. Les comunicacions entre els grups d'usuaris i les instàncies han de ser segures (per exemple amb connexió https o ssh).

- El Grup Seguretat utilitzarà diferents eines per gestionar i monitoritzar la infraestructura, com ara Prowler, AWS CloudTrail o AWS CloudWatch. Per tal de complir amb l'ENS, tots els registres generats per aquestes eines es guardaran en un bucket de S3, el qual també estarà xifrat.

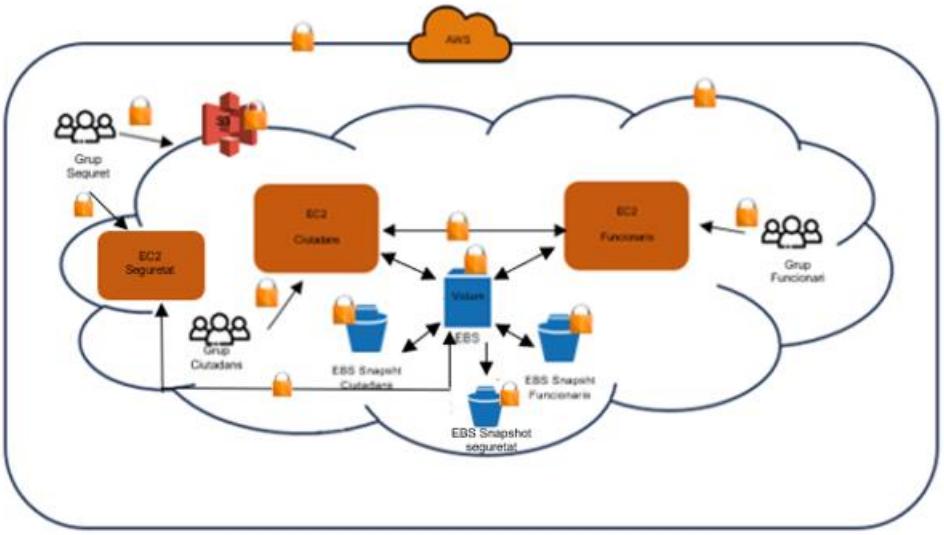


Figura 13: Infraestructura segura

3.1. Flux i controls sobre la Infraestructura

3.1.1. Flux de creació i manteniment de la Infraestructura

Per crear la infraestructura amb les eines seleccionades:

1. Crearem el codi amb Terraform

2. S'executarà *terraform init* per tal de inicialitzar el directori de treball, si hi ha errors en el codi caldrà depurar i refer-lo si no hi ha errors seguir.

3. S'executarà *terraform plan*, es podrà veure una representació de la infraestructura i detectar errors o canvis no desitjats abans d'aplicar-los realment. Si hi ha errors en el codi caldrà depurar i refer-lo si no hi ha errors, seguir.
4. S'executarà *terrascan scan* (o en el cas de voler utilitzar una política personalitzada *terrascan scan -p "política"*). En el cas de no superar els control/s cal refer el codi segons la informació donada per Terrascan. Si es superen tots el controls, seguir.

5. S'executarà *terraform apply*, aquesta instrucció crearà la infraestructura, si hi ha errors, per exemple podria ser no tenir permisos suficients per crear instàncies o problemes amb el compte de AWS, depurar i corregir els errors. Si no hi ha errors, seguir.

6. S'executarà *terrascan scan* on es tornarà a revisar tota la infraestructura creada.

7. La infraestructura ja estarà creada i serà segura.

Un cop creada la infraestructura, utilitzant l'eina Prowler, s'anirà revisant constantment, en el cas de detectar algun canvi o problema es possible començar amb el flux utilitzant la instrucció *terraform refresh*.

En aquest cas és important entendre que al refer novament la infraestructura amb *terraform apply*, destruirà les instàncies i les tornarà a crear de nou, per tant esborrarà tot el contingut d'aquestes. Aquesta informació la podrem veure al aplicar *tarraform plan* i *terraform apply*. El contingut de les instàncies és possible recuperar-lo amb les còpies de seguretat però cal tenir-ho en compte i guarda una còpia en lloc segur abans d'executar *terraform apply*.

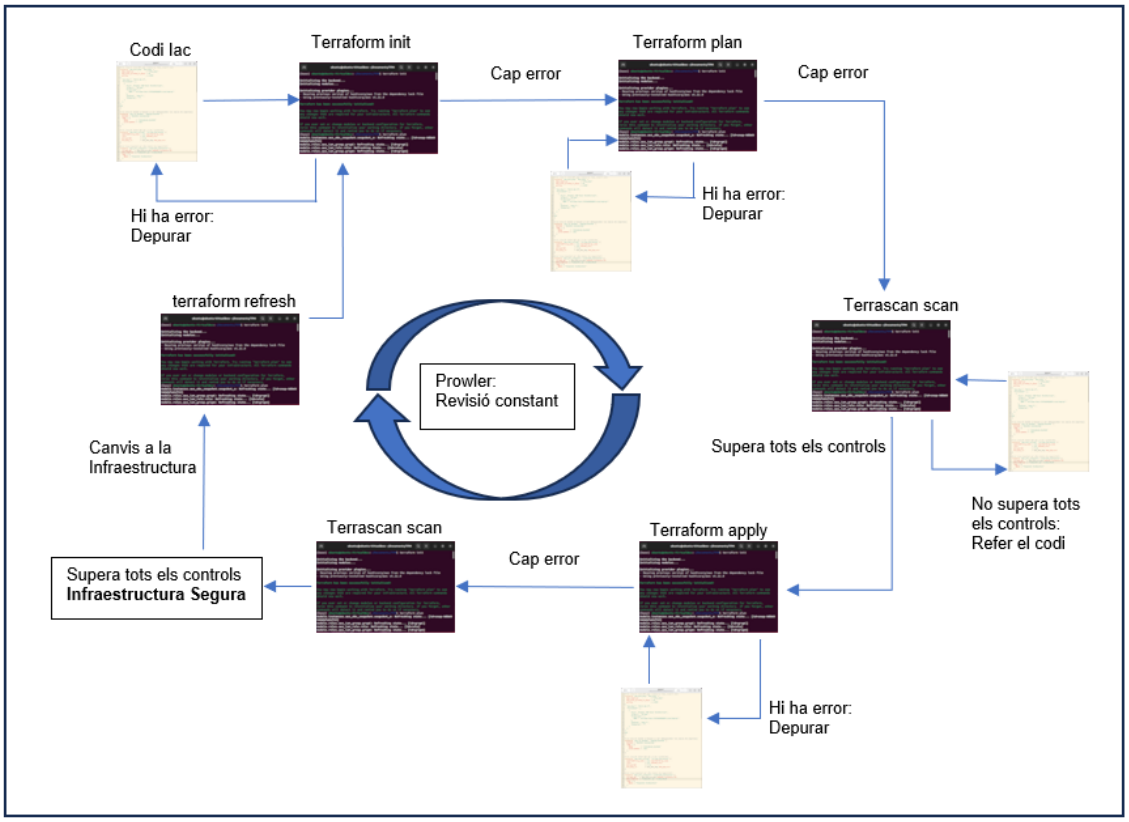


Figura 14: Flux Infraestructura Segura

3.1.2. Controls sobre la Infraestructura

Com es pot veure en la creació del flux de la infraestructura, caldrà aplicar tot un seguit de polítiques amb Terrascan, aquestes polítiques seran examinades abans de la creació de la infraestructura, i un cop creada, aprofitant Prowler, s'examinarà tota la infraestructura, es crearan els registres corresponents i s'anirà comprovant constantment la infraestructura.

Polítiques de Terrascan

Amb terrascan scan es revisaran tot un seguit de controls de l'ENS utilitzant diferents polítiques que vigilaran que:

a. Tots els grups de usuaris tinguin polítiques associades.

b. Totes les instàncies estiguin xifrades

c. Totes les instàncies tinguin còpies de seguretat xifrades

d. Els grups de seguretat estiguin utilitzant les eines AWS CloudTrail, AWS CloudWatch . Aquestes son les eines que necessitarà per treballar amb Prowler.

Controls amb Prowler

Un cop creada la infraestructura, s'executarà l'eina Prowler i es crearan els scripts necessaris . Amb aquesta revisió de controls es podrà veure clarament el nivell de compliment de l'ENS a la infraestructura.

3.1.3. Proves sobre la Infraestructura

Per tal d'assegurar que aquesta infraestructura compleix amb l'ENS, es faran tot un seguit de proves:

a. Es provaran les diferents polítiques creades amb Terrascan.
b. Es comprovarà el funcionament correcte de Prowler així com el funcionament correcte del script associat per automatitzar els seus controls.
c. Es comprovarà que el script de còpies de seguretat de les instàncies funciona correctament.

3.2. Implementació de la infraestructura

En el desenvolupament i implementació de la infraestructura, s'ha decidit treballar sobre Ubuntu 22.04 LTS[51] ja que, a més de la seva gran estabilitat, disposa d'una alta compatibilitat amb una àmplia gamma d'eines de software, incloent Terraform i Terrascan, que son essencials pel desenvolupament d'aquest projecte.

3.2.1. Configuració de l'entorn de treball

La instal·lació i configuració de Terraform i Terrascan a Ubuntu 22.04 LTS, està reflectida a l'annex "Configuració de l'Entorn de Treball". Aquest annex proporciona una guia pas a pas per configurar l'entorn necessari per a una implementació eficient i segura de la infraestructura.

Un cop instal·lades les aplicacions necessàries, cal crear una directori de treball:

```
main.tf
variables.tf
user_credentials.tfvars
modules/
  iam/
    main.tf
    variables.tf
  instancies/
    main.tf
    variables.tf
  xarxa/
    main.tf
    variables.tf
  serveis/
    main.tf
    variables.tf
```

Figura 15: Estructura del projecte

3.2.2. Creació del codi

Per començar a crear el codi, primer caldrà tenir un usuari IAM a AWS amb permisos suficients, això implicarà que ha de poder crear IAM, instàncies EC2, VPC, KMS, S3 així com permisos pel desplegament de les diferents eines i serveis de AWS.

Dins del main.tf principal caldrà identificar aquest usuari, pel tal de protegir aquestes credencials d'accés s'utilitza el fitxer "user_credentials.tfvars" on hi ha les credencials d'aquest usuari. Tot el codi es comparteix utilitzant el servei Github[52] i l'arxiu amb les credencials està afegit a ".gitignore". Per reutilitzar aquest codi, caldrà crear un nou document user_credentials.tfvars amb aquesta estructura:

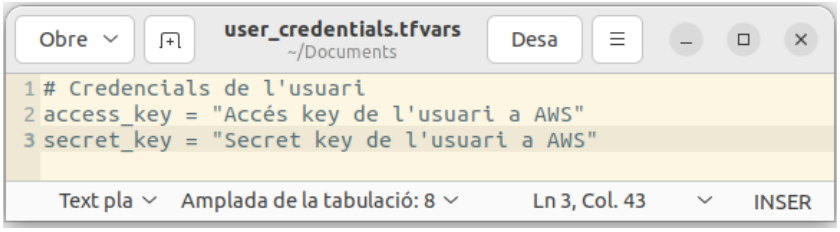


Figura:16 user_credentials.tfvars

Aquest codi es pot executar amb la comanda:
terraform init -var-file=user_credentials.tfvars on s'utilitzen aquestes credencials.

A l'arxiu main.tf principal es defineix el proveïdor que s'utilitzarà, en aquest cas AWS, així com els recursos i la invocació dels diferents mòduls, l'arxiu variables.tf s'utilitza per definir les variables que s'utilitzaran en el codi Terraform.

Per tal de complir amb els requeriments de l'ENS, cal ser molt curós a l'hora d'escriure el codi i tenir tota una sèrie de consideracions particulars.

- Mòdul IAM:

Dins del mòdul IAM cal crear els diferents grups d'usuaris: ciutadans, funcionaris i seguretat on cada grup d'usuaris ha de tenir les seves polítiques concretes utilitzant el concepte de mínims privilegis, donar els permisos mínims necessaris.

En el cas dels grups ciutadans i funcionaris aquests permisos han de ser per poder interactuar amb les seves instàncies ec2

```
# Política per ciutadans
resource "aws_iam_policy" "ciutada_policy" {
  name        = "ciutada-policy"
  description = "Política per a ciutadans"

  policy = jsonencode({
    Version   = "2012-10-17",
    Statement = [
      {
        Action   = [
          "ec2:DescribeInstances",
          "ec2:StartInstances",
          "ec2:StopInstances",
          "ec2:RebootInstances",
          "ec2:TerminateInstances",
          "ec2:AuthorizeSecurityGroupIngress",
        ],
        Effect   = "Allow",
        Resource = "*"
      },
    ],
  })
}
```

Figura 17: Política per a ciutadans

En el cas del grup de seguretat aquests permisos han de ser majors ja que a més dels permisos per poder interactuar amb les instàncies ec2 també han de poder interactuar amb S3 i Prowler. A la guia ràpida de Prowler de CCN-STIC 887B[14] hi ha el detall de tots aquests permisos.

Un cop creades les diferents polítiques, s'han d'associat a cada grup, s'han de creat els usuaris i s'ha d'associat l'usuari al seu grup corresponent, d'aquesta manera s'assegura el requeriment de l'ENS on diu que els permisos no han d'estar associats directament als usuaris.

```
# Es crea el grup dels ciutadans
resource "aws_iam_group" "ciutadans" {
  name = "ciutadans"
}

# Associar la política amb el grup "ciutadans"
resource "aws_iam_group_policy_attachment" "ciutada_attachment" {
  policy_arn = aws_iam_policy.ciutada_policy.arn
  group      = aws_iam_group.ciutadans.name
}

# Es crea l'usuari usCiutada
resource "aws_iam_user" "usCiutada" {
  name = "usCiutada"
}

# Associació de l'usuari "usCiutada" al grup "ciutadans"
resource "aws_iam_group_membership" "ciutadans_membership" {
  name = "ciutadans-membership"
  users = [aws_iam_user.usCiutada.name]
  group = aws_iam_group.ciutadans.name
}
```

Figura 18: Assignació i creació per a ciutadans

Per tal de poder executar els scripts de còpies de seguretat i també per poder enviar les troballes de Prowler a CloudWatch, també s'ha tingut de crear un rol IAM associat a la instància ec2 de seguretat

```
# Creació del iam per la instancia de seguretat per cloudwatch
resource "aws_iam_role" "cloudwatch_agent_role" {
  name = "cloudwatch_agent_role"
  assume_role_policy = jsonencode({
    Version = "2012-10-17",
    Statement = [
      {
        Action = "sts:AssumeRole",
        Effect = "Allow",
        Principal = {
          Service = "ec2.amazonaws.com"
        },
      },
    ],
  })
}

resource "aws_iam_role_policy_attachment" "cloudwatch_agent_policy_attachment" {
  role       = aws_iam_role.cloudwatch_agent_role.name
  policy_arn = "arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy"
}

resource "aws_iam_instance_profile" "cloudwatch_agent_instance_profile" {
  name = "cloudwatch_agent_instance_profile"
  role = aws_iam_role.cloudwatch_agent_role.name
}
```

```
# Permisos per script de copies de snapshots
resource "aws_iam_policy" "copia_snapshot" {
  name        = "copia_snapshot"
  description = "Política amb permisos addicionals per la creació de captures instantànies"
  policy = jsonencode({
    Version = "2012-10-17",
    Statement = [
      {
        Effect = "Allow",
        Action = [
          "ec2:DescribeInstances",
          "ec2:DescribeVolumes",
          "ec2:CreateTags",
          "ec2:CreateSnapshot",
          "ec2:DeleteSnapshot",
          "ec2:DescribeSnapshots"
        ],
        Resource = "*"
      }
    ]
  })
}

resource "aws_iam_role_policy_attachment" "copia_snapshot_attachment" {
  role       = aws_iam_role.cloudwatch_agent_role.name
  policy_arn = aws_iam_policy.copia_snapshot.arn
}

# output pel modul instàncies i així associar el rol a la instància ec2 de seguretat
output "cloudwatch_agent_instance_profile_name" {
  value = aws_iam_instance_profile.cloudwatch_agent_instance_profile.name
}
```

Figura 19: IAM per instància ec2 seguretat

- Mòdul Instancies

En aquest mòdul es creen totes les instàncies de la infraestructura, per complir amb els diferents requeriments de l'ENS, caldrà tenir tot un seguit de consideracions:

- Cal una clau KMS per tal de xifrar totes les instàncies, aquesta es generarà amb Terraform.

- Tots els volums ebs han d'estar xifrats

```
# Assegurem que tots els volums ebs estiguin xifrants per defecte
resource "aws_ebs_encryption_by_default" "esb_xifrat" {
  enabled = true
}
```

Figura 20: ebs_encryption_by_default

Així doncs caldrà crear les diferents instàncies: ec2_ciutadans, ec2_funcionaris i ec2_seguretat. Abans de crear aquestes instàncies es

necesiten un joc de claus (pública/privada) que s'associarà a aquestes instàncies, aquestes claus es generaran amb la instrucció ssh-keygen

Figura 21: codi per ec2_seguretat

A la Figura 21 es pot apreciar com s'han creat les instàncies, aquesta instància, a diferència de les altes dues, disposa de l'associació al IAM de CloudWatch creat al mòdul anterior.

Analitzat el codi, es pot apreciar que cada instància està associada a la seva subnet i a la seva vpc. Com s'ha pogut veure a l'apartat arquitectura de la infraestructura, cada instància disposarà de la seva vpc, de la seva subnet, del seu security group i de les seves taules d'enrutament. D'aquesta manera es garanteix una bona segregació a tots els nivells. Aquesta part del codi està dins del mòdul Xarxa.

També es pot comprovar com es xifren les instàncies utilitzant la clau KMS i com es creen les còpies de seguretat també xifrades amb la mateixa clau.

Com s'ha pogut veure a l'apartat flux i controls de la infraestructura, al crear el codi s'ha anat analitzant amb Terrascan, en primera instància, al executar Terrascan, ha trobat certes troballes d'incompliment de polítiques

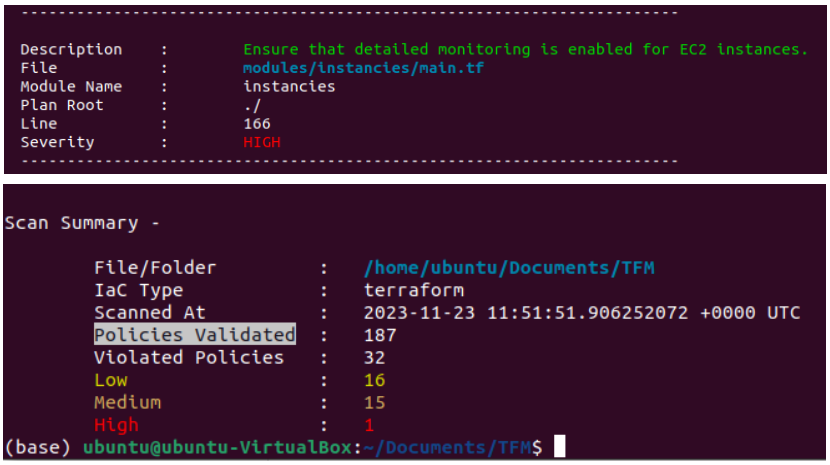


Figura 22: terrascan scan

Aquesta troballa es refereix al fet que d'entrada les instàncies EC2 no tenien activat el monitoratge, tal com es pot veure a la figura 21 això s'ha solucionat afegint monitorint=true

- Mòdul Xarxa

Dins del mòdul Xarxa, s'han creat una VPC, una subxarxa, un taula d'enrutament i un security group associat a cada instància. A més per garantir el funcionament de la VPN, en aquest cas s'ha utilitzat el software openvpn, s'ha utilitzat el servei peering connection de AWS.

```
# Configuració de la VPN, la resta es farà manualment a cada ec2
# Connectem les dues vpc per la vpn
resource "aws_vpc_peering_connection" "peering_ciutadans_funcionaris" {
  peer_vpc_id   = aws_vpc.vpc_funcionaris.id
  vpc_id        = aws_vpc.vpc_ciutadans.id
  auto_accept   = true

  tags = {
    Name = "Peering Ciutadans-Funcionaris"
  }
}
```

Figura 23: peering_connection

Cada vpc disposa de connectivitat a internet, això és necessari per tal de poder actualitzar les diferents instàncies. Aquesta connectivitat s'aconsegueix amb els security group i les taules d'enrutament associades.

De forma resumida, la instància ciutadans i la funcionaris han de ser accessibles via ssh per l'usuari de seguretat, en el codi es pot veure que només és accessible per una ip pública concreta.

A aquestes dues instàncies també se'l hi ha habilitat connectivitat https, port 443, per tal que els diferents usuaris, ja siguin clients o funcionaris puguin interactuar amb les seves instàncies

```
# Creació de Security Groups
resource "aws_security_group" "sg_ciutadans" {
  vpc_id = aws_vpc.vpc_ciutadans.id
  ingress {
    from_port   = 443
    to_port     = 443
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
   ingress {
    from_port   = 1194  # ports de la vpn
    to_port     = 1194
    protocol    = "udp"
    cidr_blocks = [var.vpc_cidr_block_funcionaris]
  }

  ingress {
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["92.178.237.113/32"] #ip pública de l'usuari de seguretat
  }

   egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
  tags = {
    Name = "SG Ciutadans"
  }
}
```

Figura 24: sg-ciutadans

- Mòdul Serveis

Bona part dels requeriments de l'ENS per considerar una arquitectura segura estan dirigits al control i seguiment d'aquesta, per aquest motiu en aquesta infraestructura s'han utilitzats tot un seguit de serveis, com son Cloudtrail i CloudWatch.

El servei cloudtrail s'està emmagatzemant en un bucket s3, tot xifrat amb una clau KMS especifica. Els logs de CloudWatch també s'estan xifrant amb aquesta clau KMS.

Un cop creada la infraestructura cal connectar-se via ssh a les instàncies i instal·lar i o programar, tot una sèrie de programari i scripts per tenir configurades les còpies de seguretat i la VPN entre les instàncies ciutadans i funcionaris. La connexió VPN està explicada a l'annex "VPN entre instàncies".

- Scripts còpies de seguretat snapshot

Per crear les còpies de seguretat de les instàncies, tal com s'ha comentat a l'apartat "Infraestructura de l'arquitectura", s'utilitzarà el següent script, executat des de la instància seguretat:

Figura 25: script copia_snapshots.sh

Consta de les següents parts:

- Definició de variables: Defineix les variables DATE i REGION. DATE s'utilitza per veure quan es va crear la snapshot. REGION especifica la regió d'AWS on es troben les instàncies EC2.

- Obtenir la llista de totes les instàncies: Utilitza la instrucció *aws ec2 describe-instances* per a obtenir una llista de totes les instàncies EC2 en la regió especificada.
- Bucle per a cada instància: Per a cada instància en la llista, el script fa el següent:
- Obtenir el nom de la instància: Utilitza la instrucció *aws ec2 describe-tags* per a obtenir el nom de la instància.
- Bucle per a cada volum: Per a cada volum associat a la instància, el script fa el següent:
    - Crear una snapshot: Utilitza la instrucció *aws ec2 create-snapshot* per a crear una snapshot del volum.
    - Afegir una etiqueta "Name" a la snapshot: Utilitza la instrucció *aws ec2 create-tags* per a afegir una etiqueta "Name" a la snapshot.
- Neteja de snapshots antigues: Després de crear snapshots per a tots els volums de cada instància, el script neteja les snapshots antigues. Manté les tres snapshots més recents i la primera snapshot creada per a cada instància, aquesta part és summament important ja que si no es fes, a l'executar *terraform apply* esborraria i tornaria a crear totes les instàncies ja que es crea una snapshot a la creació de cada instància. Utilitza la instrucció *aws ec2 delete-snapshot* per a eliminar les snapshots antigues.

Aquest script es crearà connectant-se a la instància seguretat amb la instrucció:

```
ssh -i keySeguretat ubuntu@ec2-34-248-170-172.eu-west-1.compute.amazonaws.com
```

On keySeguretat és la clau privada associada a la clau pública que s'ha assignat a la instància seguretat a la seva creació. Només l'usuari usSeguretat i des de la seva ip pública pot connectar-se via SSH a aquestes instàncies. Aquest usuari disposa de permisos del security group i de les claus privades associades a les claus públiques de cada instància.

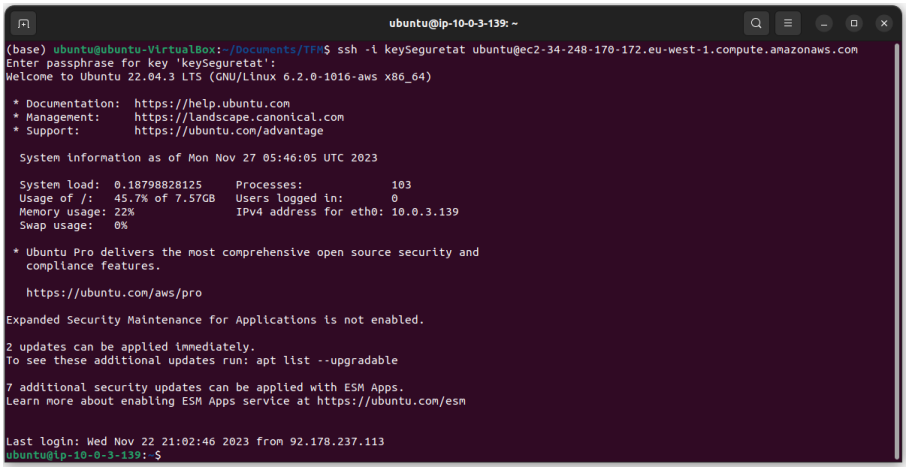


Figura 26: connexió instància seguretat

Un cop connectat a la instància, es crearà l'script amb la instrucció:

```
sudo nano copia_snapshots.sh
```

i se li donarà els permisos necessaris per poder executar l'script amb

```
sudo chmod +x copia_snapshots.sh
```

(fonts aws[53])

- Instal·lació Prowler a la instància seguretat

Per instal·lar Prowler a la instància seguretat, un cop connectat via ssh caldrà:

1. Comprovar si està instal·lat python3 i pip, amb les comandes

   `python3 –version` `pip3 –version`

   En el cas de no tenir-ho instal·lat caldrà instal·lar com a mínim la versió 3.9 de python.
2. Instal·lar el client d'AWS amb a instrucció `sudo apt install awscli` i configurar les credencials de l'usuari de seguretat, figura 10, amb la instrucció. `aws configure`
3. Instal·lar Prowler amb la instrucció `pip install prowler`
   Es pot executar amb la instrucció:

   ```
   prowler --compliance ens_rd2022_aws --filter-region eu-west-1
   ```

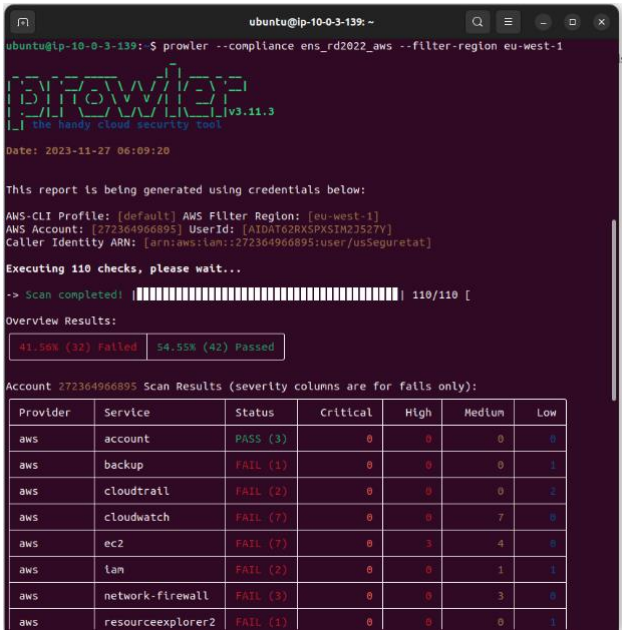


Figura 27: Prowler

Un cop instal·lar Prowler, per enviar les troballes a CloudWatch caldrà instal·lar l'agent CloudWatch[54] a la instància:

4. Descarregar l'agent amb la instrucció

```
wget https://s3.amazonaws.com/amazoncloudwatch-agent/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb
```

5. Instal·lar amb la instrucció

```
sudo dpkg -i -E ./amazon-cloudwatch-agent.deb
```

6. Configura l'agent amb la instrucció

```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard
```

Un com instal·lat, l'agent quedarà:

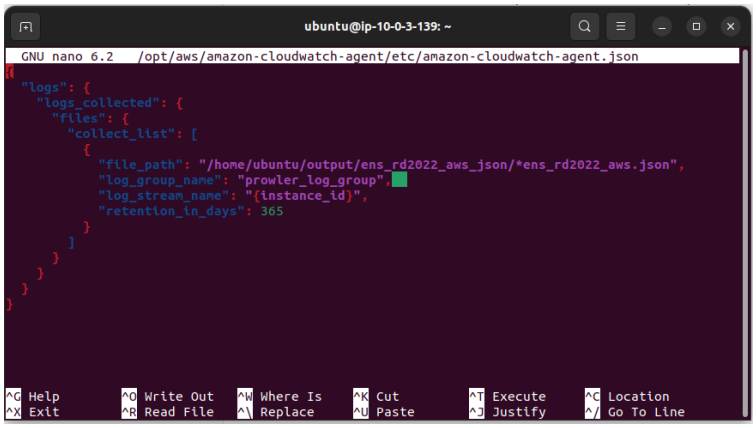


Figura 28: agent CloudWatch

L'arxiu de sortida amb les troballes específiques de l'ENS de Prowler, es guarden a la carpeta /home/ubuntu/outputs, en format csv, per poder treballar amb CloudWatch caldrà canviar el format a json

Per convertir aquestes dades s'utilitzarà un script de python utilitzant pandas:

- S'instal·la pandas amb la comanda `pip3 install pandas`
- Es crea l'script amb `nano csv_to_json.py`

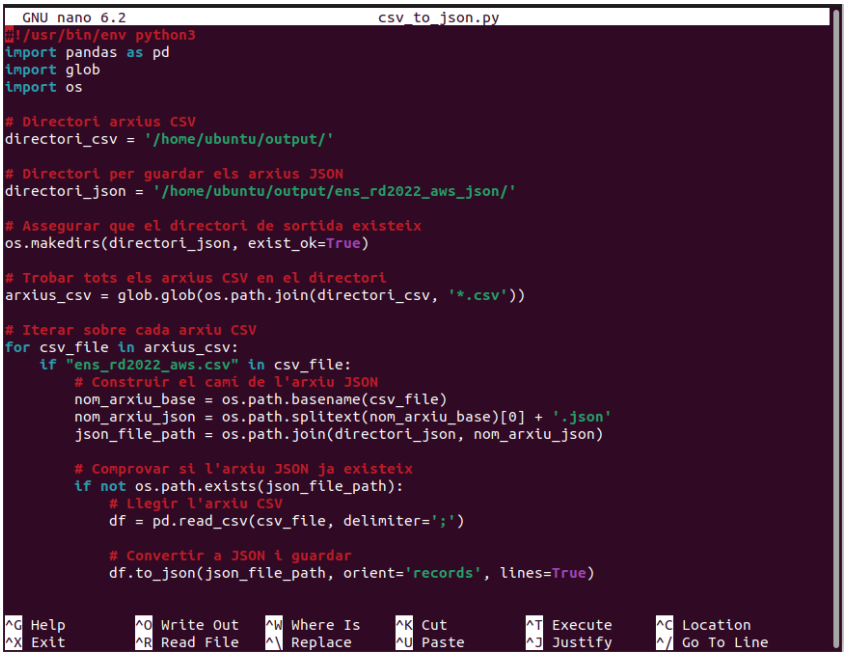


Figura 29: csv_to_json

Per executar Prowler de forma automatitzada es crea un script amb la comanda `nano run_prowler.sh`

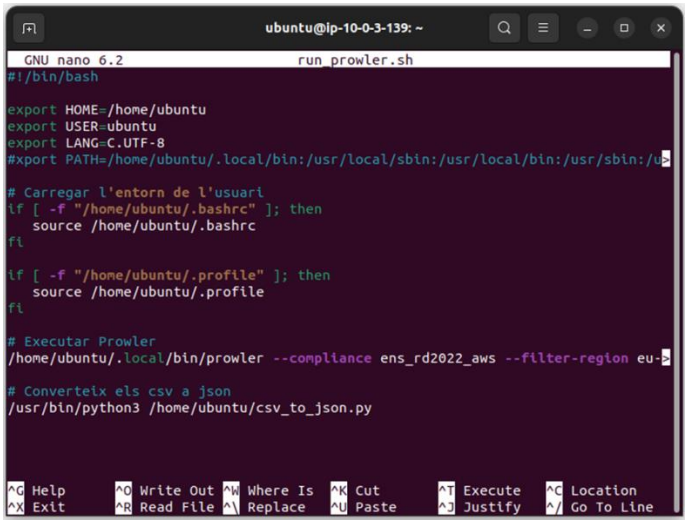


Figura 30: run_prowler.sh

On es carrega el perfil de l'usuari usSeguretat, la instrucció contab no carrega automàticament aquest perfil i és necessari per l'execució de Prowler, i s'executarà també l'script csv_to_json.py el qual convertirà els arxius al format adequat per CloudWatch. Al estar l'agent de CloudWatch funcionant a la instància, enviarà les troballes al log de CloudWatch.

Per tant de complir amb els requeriments de l'ENS, s'utilitzarà l'eina cron del sistema operatiu Ubuntu per automatitzar la creació de les còpies de seguretat i l'escaneig amb Prowler. Amb la instrucció `crontab -e` s'obra un editor de text on es programa les còpies de seguretat, aquestes s'executaran diàriament a mitjanit i l'escaneig amb Prowler mensualment a la una de la matinada.

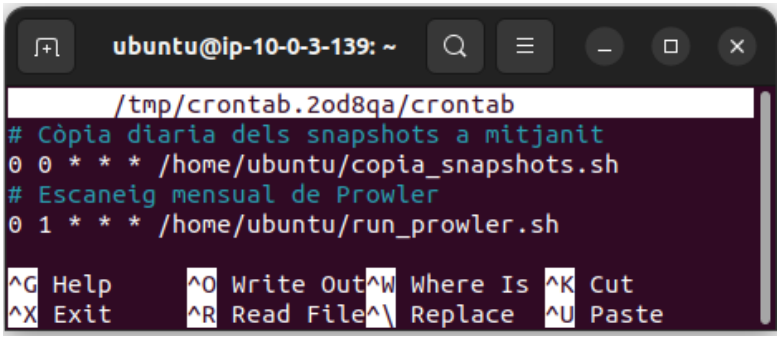


Figura 31: crontab

3.2.3. Implementació de les Proves sobre la Infraestructura

Encara que les polítiques de Terrascan ja cobreixen moltes de les necessitats en seguretat específiques de l'ENS, també permet crear polítiques personalitzades[55]. Aquestes polítiques han d'estar formades per dos arxius, un rego i l'altre amb format json.

Per comprovar aquestes polítiques es poden especificar amb una instrucció de Terrascan indican el camí on està la política, instrucció: `terrascan -p <directori>` o també es poden afegir en carpetes al directori de polítiques AWS de Terrascan (~/.terrascan/pkg/policies/opa/rego/aws). Per exemple, per escanejant codi de Terraform, si es vulgues comprovar una política concreta situada a la carpeta policy, s'utilitzaria la instrucció: `terrascan scan -i terraform -p /home/ubuntu/policy`

S'han creat un seguit de polítiques de Terrascan amb les quals podrem comprovar:

a. Tots els grups de usuaris tinguin polítiques associades.

Figura 32: Política grups

La política busca tots els grups IAM definits al codi (*input.aws_iam_group[_]*), per cada grup IAM, mira si existeix alguna política IAM associada a aquest grup (*input.aws_iam_group_policy_attachment[_]*) i utilitza l'expressió *group_has_policy := policy_attachment.group == group.name* per comprovar si el nom del grup en l'assignació de la política coincideix amb el nom del grup IAM.

Si un grup IAM no te una política assignada (*not group_has_policy*), la política registra aquesta no conformitat en el retorn (*retVal*), indicant l'identificador del grup (Id), i altres detalls com el tipus de bloc de codi Terraform on es troba la no conformitat. El resultat inclou informació com el tipus de reemplaçament (*ReplaceType*), tipus de codi (*CodeType*), l'atribut Terraform afectat (*Attribute i AttributeDataType*), i els valors esperats i reals (*Expected i Actual*).

b. Totes les instàncies estiguin xifrades i Totes les instàncies tinguin còpies de seguretat xifrades

Figura 33: instàncies xifrades

Aquesta política busca tots els volums EBS definits al codi (*input.aws_ebs_volume[_]*), per cada volum EBS, comprova si està xifrat utilitzant la funció *checkEncryption*. Si un volum EBS no està xifrat (*isEncrypted*), la política registra aquesta no conformitat en el retVal.

Seguidament Cerca totes les instàncies EC2 definides (*input.aws_instance[_]*) i per cada instància EC2, comprova si el dispositiu de bloc arrel està xifrat. Registra no conformitats si una instància EC2 te el dispositiu de bloc arrel no xifrat.

El resultat inclou detalls com l'identificador del volum (Id), tipus de canvi requerit (*ReplaceType*), tipus de codi (*CodeType*), atribut afectat (*Attribute*), el seu tipus de dades (*AttributeDataType*), i els valors esperats i reals (*Expected i Actual*).

c. Els grups de seguretat estiguin utilitzant les eines AWS CloudTrail

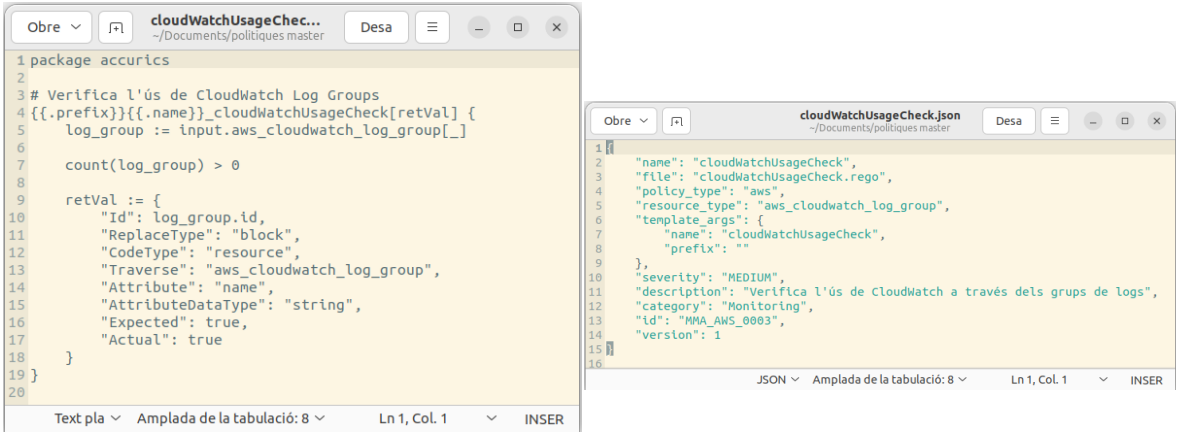


Figura 34: política clouwatch

La política busca i comprova si hi ha grups de logs CloudWatch definits (*input.aws_cloudwatch_log_group[_]*). Si no hi ha grups de logs CloudWatch, es registra com a no conformitat. Els resultats inclouen informació sobre l'identificador del grup de logs, el tipus de bloc de codi i l'atribut afectat.

d. Els grups de seguretat estiguin utilitzant les eines AWS CloudTrail

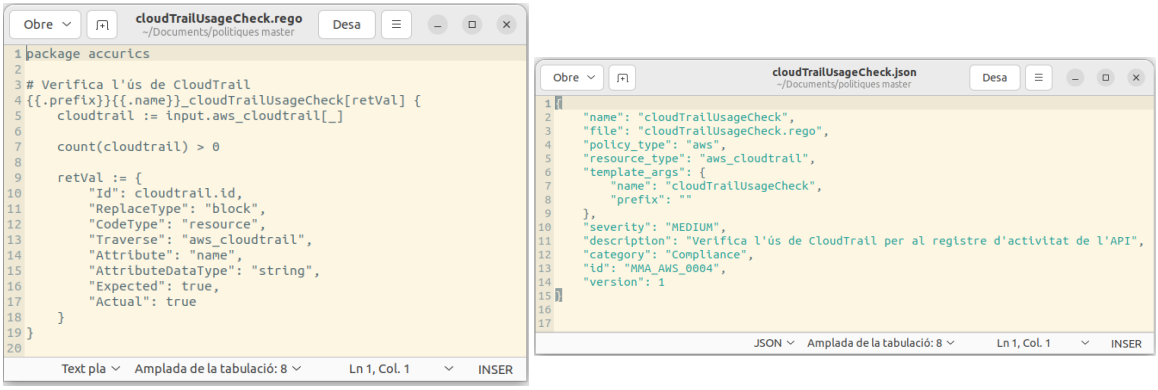


Figura 35: política cloudTrail

La política cerca i verifica la presència de recursos CloudTrail definits (*input.aws_cloudtrail[_]*). Si no es troben recursos CloudTrail, es registra com a no conformitat. Els resultats reflecteixen informació sobre els recursos CloudTrail.

d. Funcionament correcte de Prowler

A l'apartat instal·lació de Prowler, es pot verificar el correcte funcionament amb els diferents resultat de l'exploració d'aquesta aplicació.

4. Resultats

4.1. Codi Terraform

Cal indicar que el codi obtingut s'ha de considerar una guia per tal de poder desplegar una infraestructura de forma segura. El codi complet de la infraestructura està disponible a https://github.com/mmartinezarb/terraform-aws-ens2022.git on podem observar els diferents mòduls de Terraform.

4.1.1. Funcionament VPN entre les instàncies ciutadans i funcionaris

Per tal de comprovar el correcte funcionament de la VPN entre les instàncies "ciutadans" i "funcionaris", a l'annex VPN entre instàncies s'especifica la seva instal·lació, caldrà:

- Connectar-se a cada instància via ssh, tan sols ho pot fer l'usuari usSeguretat des de la seva IP.

- Amb la comana ifconfig tun0 mirar la xarxa de la connexió VPN

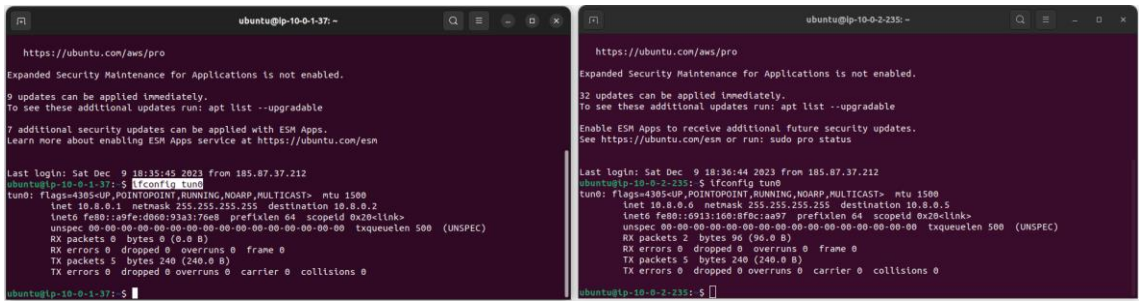


Figura 36: ifconfig tun0 (Ciutadans/Funcionaris)

- Fem un ping des de la instància ciutadans a la ip inet del client 10.8.0.6 i des de la instància funcionaris a la ip inet del servidor 10.8.0.1.

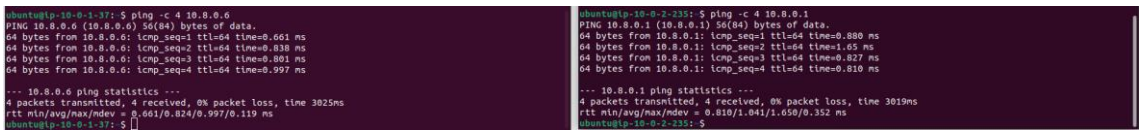


Figura 37: ping (Ciutadans/Funcionaris)

Com es pot observar en les figures, els pings han estat exitosos en ambdues direccions, confirmant així que la comunicació a través de la VPN està funcionant correctament entre les dues instàncies.

4.1.2. Funcionament script còpies de seguretat

Per comprovar el correcte funcionament de l'script de còpies de seguretat de les instàncies, així com el correcte funcionament de l'automatització d'aquestes amb cron, es pot veure directament connectant a la pàgina de AWS

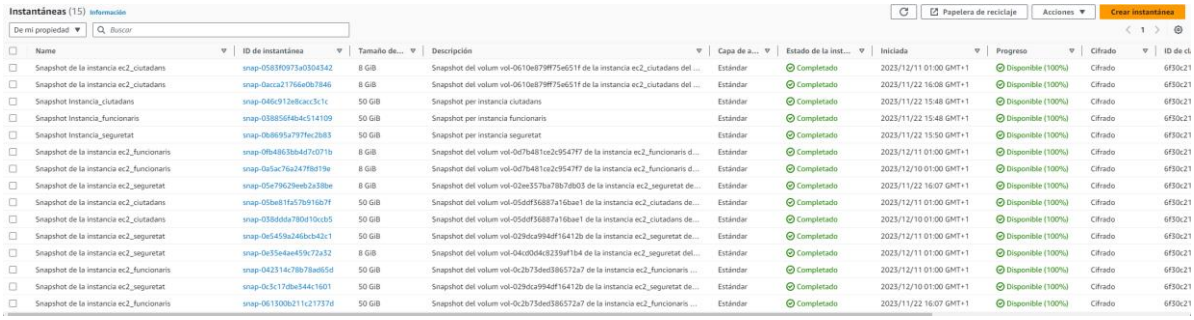


Figura 38: Snapshot AWS

On es pot observar les diferents còpies de seguretat creades, tant des de Terraform com des de l'script, amb les dates i les hores de creació es pot comprovar també que l'automatització funciona correctament.

4.2. Resultats amb Terrascan

Les diferents polítiques creades amb de Terrascan estan disponibles a https://github.com/mmartinezarb/terraform-aws-ens2022/tree/main/Politiques%20Terrascan, cal considerar que a més de les polítiques personalitzades utilitzades també s'ha aprofitat les polítiques per AWS[56] que ja disposa Terrascan.

4.3. Resultats i integració Prowler i CloudWatch

Un cop integrat Prowler amb CloudWatch, s'ha creat un panell de CloudWatch, utilitzant codi de Terraform (per un problema de mida, només s'ha afegit una petita part del codi, la resta està dins del mòdul serveis):

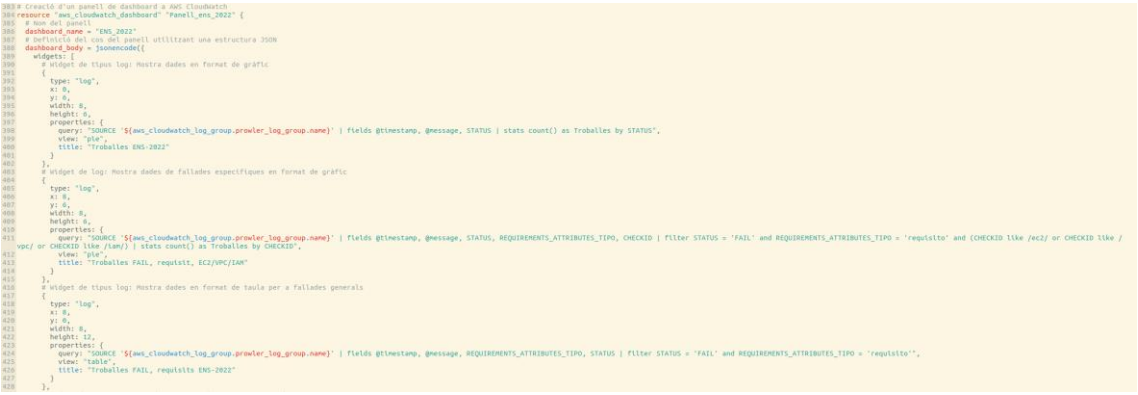


Figura 39: Part del codi panell CloudWatch

On s'obtindrà el següent panell de CloudWatch:

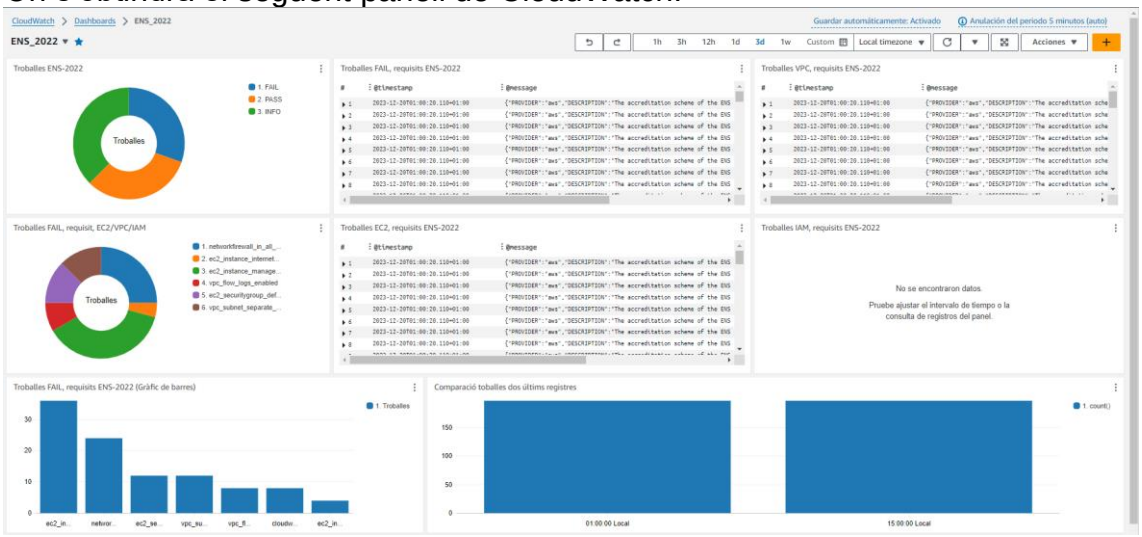


Figura 40: Panell CloudWatch

Aquest panell, anomenat "ENS_2022", està dissenyat per ajudar a entendre l'estat de les mesures de seguretat segons els estàndards de l'Esquema Nacional de Seguretat (ENS). Les diferents parts del panell son:

- Gràfics Troballes ENS-2022 i Troballes FAIL, requisit, EC2/VPC/IAM: Aquests widgets mostren la distribució de les "Troballes" segons el seu estat (com FAIL, PASS, INFO) i detalls específics com EC2/VPC/IAM. Aquest tipus de gràfics, faciliten veure ràpidament la proporció de diferents tipus de troballes.

- Gràfic de Barres Troballes FAIL, requisits ENS-2022 (Gràfic de barres): Aquest gràfic mostra el nombre de troballes que han fallat segons diferents categories o identificadors de comprovació (CHECKID). És útil per identificar ràpidament quines àrees tenen més incidències i requereixen major atenció.

- Taules de Logs: S'inclouen tres taules, cadascuna enfocada en un servei específic d'AWS (EC2, VPC, IAM). Aquestes taules mostren els registres detallats de les incidències o troballes que han fallat els requisits de seguretat, incloent informació com la data, el missatge associat i altres atributs rellevants.

- Gràfic de barres Comparació troballes dos últims registres: Aquest gràfic compara el nombre de troballes dels dos últims escanejats de Prowler, permetent una comparació visual. És important considerar que, mentre la resta de widgets son informatius per a un únic dia d'escaneig, aquest te sentit amb un mínim de dos dies.

Aquests widgets junts proporcionen una vista comprensiva de l'estat de la seguretat i permeten als administradors de sistemes i a l'equip de seguretat detectar ràpidament problemes, prioritzar incidències, i prendre mesures per a la seva resolució. El disseny i la configuració del panell faciliten la interpretació visual de les dades per a una gestió eficient de la seguretat.

5. Conclusions i treballs futurs

Desgraciadament, avui en dia, els ciberatacs son més freqüents a tots els nivells, tant a empreses privades com a entitats públiques, i a l'hora la utilització dels diferents serveis al núvol son una eina molt potent per gestionar les necessitats d'aquestes, com poden ser l'emmagatzematge o l'accés a les dades.

Degut a aquesta problemàtica, estan sorgint diferents estàndards, fins i tot a nivell legislatiu com pot ser l'ENS, per tal de protegir tots els recursos, la informació o l'accés.

La utilització de les diferents eines descrites, poden ajudar a aquesta finalitat i la combinació d'aquestes, conjuntament amb una bona metodologia de treball, poden ajudar a més a més a mantenir la seguretat sobre tota la infraestructura.

5.1. Conclusions sobre IaC

La utilització de Terraform com a eina IaC és una solució molt potent per aconseguir els resultat descrits anteriorment, cal destacar la facilitat a l'hora d'escriure i llegir per crear les diferents infraestructura així com la facilitat de trobar diferents solucions gràcies a la seva gran comunitat d'usuaris.

També cal destacar la gran integració amb els diferents recursos, com pot ser la integració amb CloudWatch. El fet de poder crear el panell CloudWatch amb totes les seves consultes directament amb Terraform és bàsic ja que aquest codi es pot reaprofitar per la creació i control d'altres infraestructures.

5.2. Conclusions sobre Terrascan

Terrascan és una eina que pot ajudar a la creació d'una infraestructura segura, cal destacar la seva capacitat per vigilar nombroses polítiques així com l'avantatge de poder controlar la infraestructura abans de crear-la.

Encara que Terrascan permet crear polítiques personalitzades, el fet d'utilitzar-la per controlar tots els requeriments específics de l'ENS pot ser una feina molt extensa. En el cas concret d'aquest projecte s'ha utilitzat les polítiques ja creades per Terrascan, les quals cobreixen bona part del requeriments de l'ENS.

5.3. Conclusions sobre Prowler

Prowler és una eina molt potent que ajuda al manteniment d'una infraestructura segura, a més d'estar més preparada que Terrascan pel control dels requeriments de l'ENS.

D'altra banda, aquesta eina permet una gran integració amb el servei CloudWatch d'AWS. La sortida de dades de Prowler està en diferents formats i

en el cas d'aquest projecte, el fet que tingui una sortida en format csv ajuda, un cop canviat el format a json, a implementar un panell de CloudWatch molt útil per la verificació i seguiment de compliment amb l'ENS.

Encara que aquesta eina és molt útil, cal estudiar les diferents troballes. Prowler penalitza el no ús de serveis concrets d'AWS, com per exemple no utilitzar AWS Backup però no identifica si s'ha utilitzat altres alternatives, com en aquest cas l'script de còpies de seguretat.

5.4. Conclusions sobre la combinació d'eines

Tal com s'ha comentat a l'apartat "Flux de creació i manteniment de la Infraestructura"(3.1.1.), aquest projecte utilitza la combinació de les diferents eines descrites a les conclusions: Terraform, Terrascan i Prowler.

La combinació d'aquestes tres eines ha estat totalment exitosa.

Amb Terraform es crea una infraestructura amb IaC, aprofitant tots els avantatges ja descrits.

Amb Terrascan es comprova si aquest codi compleix amb les millors polítiques de seguretat abans de crear la infraestructura.

Amb Prowler es revisa el compliment específic de l'ENS i es manté un control i seguiment sobre l'estat de seguretat de la infraestructura.

5.5. Conclusions sobre la infraestructura creada i treballs futurs

Encara que la infraestructura creada compleix amb bona part dels requeriments específics de l'ENS, no ha estat possible pujar més el grau de compliment amb totes les seves seccions.

Aquest projecte s'ha centrat en el compliment màxim de requeriments de l'ENS en la secció d'usuaris (IAM) deixant per treballs futurs la resta de seccions com poden ser VPC o EC2.

En termes de seguretat i considerant les especificacions de l'ENS en control i vigilància, el mòdul serveis pot ser una molt bona base per ser reutilitzat en altres infraestructures a AWS. Com a treballs futurs també s'inclouria la integració dels resultats d'aquest mòdul a altres serveis de AWS com pot ser cloudTrail.

També seria molt interesant com a treballs futurs, la creació d'alguna alarma de CloudWatch la qual avises si en la anàlisis de Prowler es trobés una nova troballa, indicant al panell de CloudWatch un log amb aquesta troballa.

6. Glossari

AWS Auto Scaling: Permet definir plans d'escalat per a serveis i recursos d'AWS, com Amazon Aurora, Amazon EC2, Amazon Elastic Container Service, Amazon DynamoDB i Spot Fleet.
AWS Backup: Un servei de còpies de seguretat que permet gestionar i automatitzar la protecció de les dades i la recuperació en AWS.
AWS CloudTrail: Un servei que registra les activitats i els esdeveniments de la compte i els recursos d'AWS per a la conformitat i la seguretat.
AWS CloudWatch: Un servei de monitoratge i generació d'informes que proporciona visibilitat i control de les operacions d'AWS.
AWS Config: Un servei que permet avaluar, auditar i gestionar la configuració dels recursos d'AWS per a complir amb les polítiques de seguretat i conformitat.
AWS Data Lifecycle Manager: Un servei que permet gestionar el cicle de vida de les dades, incloent la seva retenció i eliminació automàtica.
AWS ELB (Elastic Load Balancing): Un servei que distribueix el tràfic a instàncies d'Amazon EC2 per millorar l'escalabilitat i la disponibilitat.
AWS IAM (Identity and Access Management): Un servei que permet gestionar l'accés i els permisos dels usuaris i recursos a AWS.
AWS KMS (Key Management Service): Utilitzat per al xifrat de la informació en repòs i la protecció de claus criptogràfiques.
AWS Security Hub: Reuneix informació de diferents alertes i fonts per a proporcionar una vista completa de l'estat general de seguretat i compliment.
AWS Shield Advanced: Proporciona protecció contra atacs de denegació de servei (DoS) i denegació de servei distribuïda (DDoS).
AWS VPN (Virtual Private Network): Un servei que permet establir connexions xifrades entre xarxes i recursos d'AWS per a la seguretat de la comunicació.
Amazon GuardDuty: Eina de detecció d'intrusió que ajuda a identificar esdeveniments de seguretat i amenaces.
Amazon Inspector: Permet realitzar revisions regulars de les vulnerabilitats del sistema.
Amazon VPC (Virtual Private Cloud): Utilitzat per protegir la infraestructura i les aplicacions mitjançant la creació de xarxes privades virtuals.
AWS WAF (Web Application Firewall): Un servei de tallafoc de les aplicacions web utilitzat per protegir les aplicacions web distribuïdes per Amazon CloudFront.
Prowler: Eina d'auditoria i compliment de normatives com l'ENS.
Terraform: Eina d'Infraestructura com a Codi per a la gestió i desplegament d'infraestructures en el núvol.
Terrascan: Eina de revisió de codi per a la conformitat amb polítiques de seguretat abans de la seva implementació.

7. Bibliografia

(1) Infraestructura como código(Atlassian)[enlínia][consulta:29/9/23]disponible a: https://www.atlassian.com/es/microservices/cloud-computing/infrastructure-as-code
(2) Terrascan (Tenable) [en línia] [consulta:29/9/23] disponible a: https://runterrascan.io/
(3) Seguridad y Privacidad de la Información: ISO 27001 e ISO 27701(Aenor) [en línia] [consulta:30/9/23] disponible a: https://www.aenor.com/certificacion/tecnologias-de-la-informacion/seguridad-informacion
(4) Esquema Nacional de Seguridad – ENS (Gobierno de España) [en línia] [consulta:30/9/23] disponible a: https://administracionelectronica.gob.es/pae_Home/pae_Estrategias/pae_Seguridad_Inicio/pae_Esquema_Nacional_de_Seguridad.html
(5) Actualizadas las preguntas frecuentes del nuevo ENS (CCN) [en línia] [consulta:30/9/23] disponible a: https://www.ccn.cni.es/index.php/es/actualidad-ccn/931-actualizadas-las-preguntas-frecuentes-del-nuevo-ens
(6) CERTIFICADOS (Gobernanza de la Ciberseguridad Naciona) [en línia] [consulta:30/9/23] disponible a: https://gobernanza.ccn-cert.cni.es/certificados
(7) Controles ENS (ENS-CCN) [en línia] [consulta:1/10/23] disponible a: https://ens.ccn.cni.es/ens-html/index.html
(8) 800 Guías Esquema Nacional de Seguridad (CCN-CERT) [en línia] [consulta:1/10/23] disponible a: https://www.ccn-cert.cni.es/guias/guias-series-ccn-stic/800-guia-esquema-nacional-de-seguridad.html
(9) DevSecOps Pipeline (xenonstack) [en línia] [consulta:1/10/23] disponible a: https://www.xenonstack.com/insights/guide-devsecops-pipeline
(10) La Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe. Objetivos, metas e indicadores mundiales (Naciones Unidas) [en línia] [consulta:2/10/23] disponible a: https://www.cepal.org/es/publicaciones/40155-la-agenda-2030-objetivos-desarrollo-sostenible-oportunidad-america-latina-caribe#:~:text=La%20Agenda%202030%20para%20el%20Desarrollo%20Sostenible%2C%20aprobada,de%20esta%20visi%C3%B3n%20durante%20los%20pr%C3%B3ximos%2015%20a%C3%B1os.
(11) Amazon, Google, Microsoft: Here's Who Has the Greenest Cloud(wired) [en línia] [consulta:2/10/23] disponible a: https://www.wired.com/story/amazon-google-microsoft-green-clouds-and-hyperscale-data-centers/
How the Big Three cloud providers are helping customers manage their energy consumption and carbon emissions (Protocol) [en línia] [consulta:2/10/23] disponible a: https://www.protocol.com/enterprise/aws-microsoft-google-energy-carbon
(12) Information technology (IT) (TechTarget) [en línia] [consulta:2/10/23] disponible a: https://www.techtarget.com/searchdatacenter/definition/IT

(13) Prowler-cloud(github) [en línia] [consulta:18/10/23] disponible a: https://github.com/prowler-cloud/prowler
(14) CCN-STIC-887B Guía rápida de Prowler.pdf (CCN-CERT) [en línia] [consulta:18/10/23] disponible a: https://www.ccn-cert.cni.es/es/800-guia-esquema-nacional-de-seguridad/6990-ccn-stic-887b-guia-rapida-de-prowler-1/file.html
(15) Acerca de Trello (Trello) [en línia] [consulta:4/10/23] disponible a: https://trello.com/about
(16) What is Trello? (atlassian) [en línia] [consulta:4/10/23] disponible a: https://support.atlassian.com/trello/docs/what-is-trello/
(17) Backlog del producto: qué es y cómo crearlo (atlassian) [en línia] [consulta:4/10/23] disponible a: https://www.atlassian.com/es/agile/scrum/backlogs
(18) What Is Cloud Architecture?(akamai) [en línia] [consulta:4/10/23] disponible a:https://www.akamai.com/blog/cloud/what-is-cloud-architecture
(19) Comience a crear con AWS hoy mismo(aws) [en línia] [consulta:11/10/23] disponible a: https://aws.amazon.com/es/
(20) Posibilidades infinitas: desde el perímetro hasta la nube(azure) [en línia] [consulta:11/10/23] disponible a: https://azure.microsoft.com/es-es
(21) La nueva forma de la nube comienza aquí(Google Cloud) [en línia] [consulta:11/10/23] disponible a: https://cloud.google.com/?hl=es
(22) IBM Cloud. Hybrid. Open. Resilient.(IBM) [en línia] [consulta:11/10/23] disponible a: https://www.ibm.com/cloud
(23) Alibaba Cloud (Alibaba Cloud) [en línia] [consulta:01/11/23] disponible a: https://www.alibabacloud.com/es?spm=a3c0i.272861.6791778070.26.327124afCNi7nZ
(24) Dream it. Build it. Grow it. (Digital Ocean) [en línia] [consulta:11/10/23] disponible a: https://www.digitalocean.com/
(25) The Everywhere Cloud (Vultr) [en línia] [consulta:11/10/23] disponible a: https://www.vultr.com/
(26) https://www.ccn-cert.cni.es/es/guias.html
(27) Gartner Says Worldwide IaaS Public Cloud Services Revenue Grew 30% in 2022, Exceeding $100 Billion for the First Time (Gartner) [en línia] [consulta:11/10/23] disponible a: https://www.gartner.com/en/newsroom/press-releases/2023-07-18-gartner-says-worldwide-iaas-public-cloud-services-revenue-grew-30-percent-in-2022-exceeding-100-billion-for-the-first-time
(28) Guías CCN-STIC(CCN-CERT) [en línia] [consulta:11/10/23] disponible a: https://www.ccn-cert.cni.es/es/guias.html
(29) Guía de configuración segura AWS.pdf(CCN-CERT) [en línia] [consulta:11/10/23] disponible a: https://www.ccn-cert.cni.es/es/800-guia-esquema-nacional-de-seguridad/5449-ccn-stic-887a-guia-de-configuracion-segura-aws/file.html
(30) Automate infrastructure on any cloud with Terraform (HashiCorp) [en línia] [consulta:11/10/23] disponible a: https://www.terraform.io/
(31) Reuse Configuration with Modules (HashiCorp) [en línia] [consulta:11/10/23] disponible a: https://developer.hashicorp.com/terraform/tutorials/modules

(32) Terraform Consulting : Advantages and disadvantages of terraform (Ismilete) [en línia] [consulta:11/10/23] disponible a: https://www.ismiletechnologies.com/es/technology/terraform-consulting-advantages-and-disadvantages-of-terraform/
(33) ¿Qué es AWS CloudFormation? (AWS) [en línia] [consulta:14/10/23] disponible a: https://docs.aws.amazon.com/es_es/AWSCloudFormation/latest/UserGuide/Welcome.html
(34) Diseña el futuro: desarrolla mejor software y más rápido (Google Cloud) [en línia] [consulta:11/10/23] disponible a: https://cloud.google.com/deployment-manager/docs?hl=es-419
(35) ARM template best practices (Microsoft) [en línia] [consulta:11/10/23] disponible a: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/best-practices
(36) Automation for everyone (ansible) [en línia] [consulta:17/10/23] disponible a: https://www.ansible.com/
(37) Terrascan (tenable) [en línia] [consulta:17/10/23] disponible a: https://runterrascan.io/
(38) CIS Benchmarks List (CIS) [en línia] [consulta:17/10/23] disponible a: https://www.cisecurity.org/cis-benchmarks
(39) Tenable / terrascan (github) [en línia] [consulta:17/10/23] disponible a: https://github.com/tenable/terrascan
(40) Infrastructure as Code vulnerability scan with Terrascan (a cloud journey) [en línia] [consulta:17/10/23] disponible a: https://www.acloudjourney.io/blog/infrastructure-as-code-vulnerability-scan-with-terrascan
(41) Terrascan: Cloud Compliance and Security Scanner for IaC (Virtualizationhowto.com) [en línia] [consulta:17/10/23] disponible a: https://www.virtualizationhowto.com/2023/05/terrascan-cloud-compliance-and-security-scanner-for-iac/
(42) Policy Overview (tenable) [en línia] [consulta:17/10/23] disponible a: https://runterrascan.io/docs/policies/policies/
(43) Certificado categoria ALTA AWS_vf.pdf (BDO) [en línia] [consulta:23/10/23] disponible a: https://www.ccn-cert.cni.es/amparo/API/public/certificaciones/2300/download/1229
(44) Prowler-additions-policy.json (gitub) [en línia] [consulta:17/10/23] disponible a: https://github.com/prowler-cloud/prowler/blob/master/permissions/prowler-additions-policy.json
(45) Prowler-security-hub.json (gitub) [en línia] [consulta:17/10/23] disponible a: https://github.com/prowler-cloud/prowler/blob/master/permissions/prowler-security-hub.json
(46) How To Use Cron to Automate Tasks on Ubuntu 18.04 (DigitalOcean) [en línia] [consulta:2/11/23] disponible a: https://www.digitalocean.com/community/tutorials/how-to-use-cron-to-automate-tasks-ubuntu-1804
(47) Respaldo y recuperación de Amazon EC2 con instantáneas y AMI (AWS) [en línia] [consulta:2/11/23] disponible a:

https://docs.aws.amazon.com/es_es/prescriptive-guidance/latest/backup-recovery/ec2-backup.html
(48) Amazon EC2 Backup and Restore Using AWS Backup (AWS) [en línia] [consulta:2/11/23] disponible a: https://aws.amazon.com/es/getting-started/hands-on/amazon-ec2-backup-and-restore-using-aws-backup/
(49) Amazon Data Lifecycle Manager (AWS) [en línia] [consulta:2/11/23] disponible a: https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/snapshot-lifecycle.html
(50) How to add multiple AWS ClientVPN Routes using Terraform (stack overflow) [en línia] [consulta:2/11/23] disponible a: https://stackoverflow.com/questions/69174714/how-to-add-multiple-aws-clientvpn-routes-using-terraform
(51) Download Ubuntu Desktop (canonical) [en línia] [consulta:23/11/23] disponible a: https://ubuntu.com/download/desktop
(52) Let's build from here (github) [en línia] [consulta:23/11/23] disponible a: https://github.com/
(53) Ejemplo 7: Ejecución de comandos o scripts(aws) [en línia] [consulta:23/11/23] disponible a: https://docs.aws.amazon.com/opsworks/latest/userguide/cookbooks-101-basics-commands.html
create-script(aws) [en línia] [consulta:23/11/23] disponible a: https://docs.aws.amazon.com/cli/latest/reference/gamelift/create-script.html
(54) Descargue del paquete de del agente de CloudWatch(aws) [en línia] [consulta:24/11/23] disponible a: https://docs.aws.amazon.com/es_es/AmazonCloudWatch/latest/monitoring/download-cloudwatch-agent-commandline.html
(55) Policy Overview (terrascan by tenable) [en línia] [consulta:22/11/23] disponible a: https://runterrascan.io/docs/policies/policies/
(56) AWS Policies(terrascan by tenable) [en línia] [consulta:22/11/23] disponible a: https://runterrascan.io/docs/policies/aws/

8. Annexos

8.1. Configuració de l'Entorn de Treball

8.1.1. Instal·lació de Terraform

La instal·lació de Terraform en un sistema ubuntu es pot fer afegint el repositori i instal·lant amb les comandes:

```
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
```

```
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

```
sudo apt update && sudo apt install terraform
```

Podem comprovar la instal·lació de Terraform a amb la instrucció:

```
terraform –version
```

Font: Install Terraform (terraform) [en línia] [consulta:22/11/23] disponible a: https://developer.hashicorp.com/terraform/install

8.1.2. Instal·lació de Terrascan

La descarrega i instal·lació de Terrascan en un sistema Ubuntu es pot fer amb les comandes:

```
curl -L "$(curl -s https://api.github.com/repos/tenable/terrascan/releases/latest |
grep -o -E "https://.+?_Linux_x86_64.tar.gz")" > terrascan.tar.gz
```

```
tar -xf terrascan.tar.gz terrascan && rm terrascan.tar.gz
```

```
sudo install terrascan /usr/local/bin && rm terrascan
```

Font: Getting Started(terrascan by tenable) [en línia] [consulta:22/11/23] disponible a: https://runterrascan.io/docs/getting-started/#installing-terrascan

8.2. VPN entre instàncies

Per poder instal·lar i configurar la VPN a les instàncies ciutadans i funcionaris, cal connectar-se via ssh a cada instància, només l'usuari usSeguretat i des de la seva ip pública pot connectar-se via ssh a aquestes instàncies. Aquest usuari disposa de permisos del security group i disposa de les claus privades associades a les claus públiques de cada instància.

Un cop connectat, cal instal·lar openvpn a ambdues instància amb la instrucció `sudo apt install openvpn` , la instància ciutadans operarà de servidor i la instància funcionaris de client.

Seguidament cal instal·lar a la instància ciutadans easy-rsa amb la instrucció `sudo apt install easy-rsa` i inicialitzar l'entorn d'Easy-RSA, crear una nova CA, i generar certificats i claus per al servidor VPN:

- Inicia Easy-RSA amb les comandes:

```
cd /usr/share/easy-rsa i sudo ./easyrsa init-pki
```

- Construeix la CA:

```
sudo ./easyrsa build-ca nopass
```

- Genera Clau i Certificat del Servidor:

```
sudo ./easyrsa build-server-full server nopass
```

- Genera Paràmetres Diffie-Hellman:

```
sudo ./easyrsa gen-dh
```

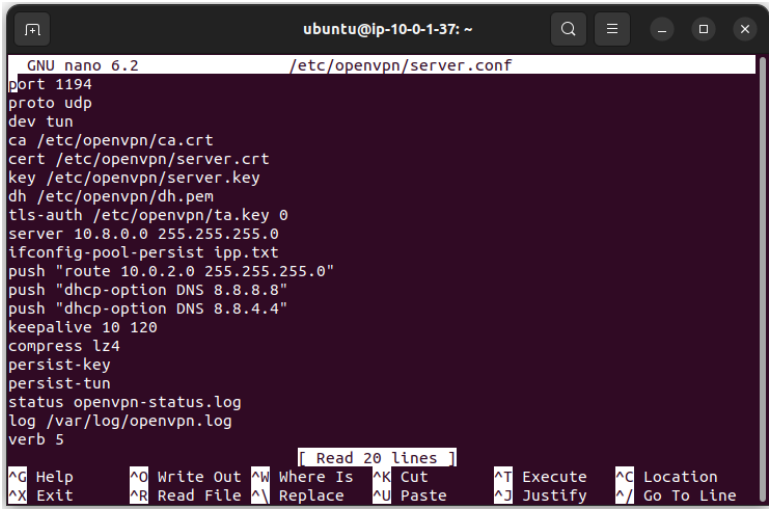
- Genera una Clau HMAC:

```
sudo openvpn --genkey secret ta.key
```

- Copiar els Certificats i Claus al Directori d'OpenVPN:

```
sudo cp pki/ca.crt /etc/openvpn
sudo cp pki/issued/server.crt /etc/openvpn
sudo cp pki/private/server.key /etc/openvpn
sudo cp pki/dh.pem /etc/openvpn
sudo cp ta.key /etc/openvpn
```

- Configurar el Servidor OpenVPN, editar el fitxer de configuració del servidor OpenVPN: `sudo nano /etc/openvpn/server.conf`



Configuració servidor VPN

- Habilitar el Ruting IP: Edita el fitxer /etc/sysctl.conf i des comentar net.ipv4.ip_forward=1. Aplica els canvis amb `sudo sysctl -p`
- Configurar Regles d'Iptables amb la instrucció

```
sudo iptables -t nat -A POSTROUTING -s 10.8.0.0/8 -o eth0 -j
```

i aplicar permanentment les Regles d'Iptables amb

```
sudo apt-get install iptables-persistent
```

- Amb la instrucció

```
sudo systemctl start openvpn@server
```

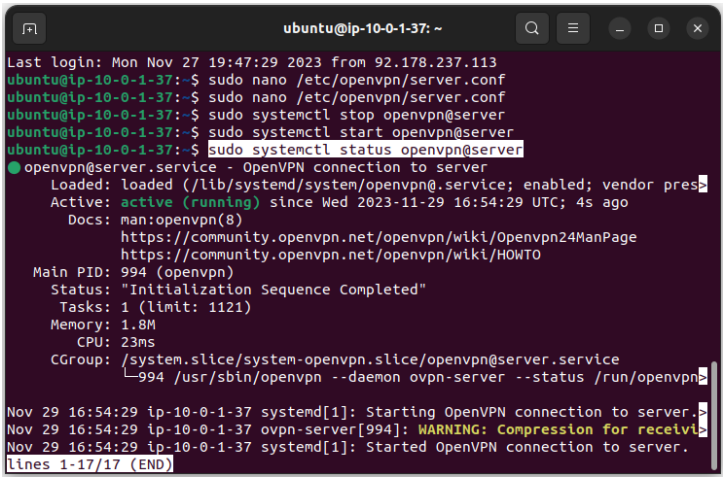
s'inicia el servidor i es pot veure l'estat del servidor amb la instrucció:

```
sudo systemctl status openvpn@server
```

amb la instrucció

```
sudo systemctl enable openvpn@server
```

s'habilitarà l'arrencada automàtica del servidor



Estat servidor VPN

Un cop configurat el servidor, instància ciutadans, es configurarà la instància funcionaris:

- Des de la instància ciutadans es genera el certificat i claus per al client

```
./easyrsa build-client-full client nopass
```

que generar els arxius client.crt i client.key

- Cal passar els arxius ca.crt, client.crt, client.key i ta.key a la instància funcionaris, per fer-ho primes es copien els fitxers a la carpeta arrel amb la instrucció

```
sudo cp ca.crt client.crt client.key ta.key /home/ubuntu/
```

- Es canvien els permisos amb

```
sudo chmod 644 /home/ubuntu/{ca.crt,client.crt}
sudo chmod 600 /home/ubuntu/client.key
```

- Es descarreguen al ordinador de l'usuari usSeguretat amb les comandes (cal tenir la clau privada de la instància ciutadans):

```
scp -i ./keyCiutadans ubuntu@18.202.128.24:/home/ubuntu/ca.crt ~/Documents/vpn/ca.crt
scp -i ./keyCiutadans ubuntu@18.202.128.24:/home/ubuntu/client.crt ~/Documents/vpn/client.crt
scp -i ./keyCiutadans ubuntu@18.202.128.24:/home/ubuntu/client.key ~/Documents/vpn/client.key
scp -i ./keyCiutadans ubuntu@18.202.128.24:/home/ubuntu/ta.key ~/Documents/vpn/ta.key
```

- Es carreguen els arxius a la instància funcionaris (cal tenir la clau privada de la instància funcionaris):

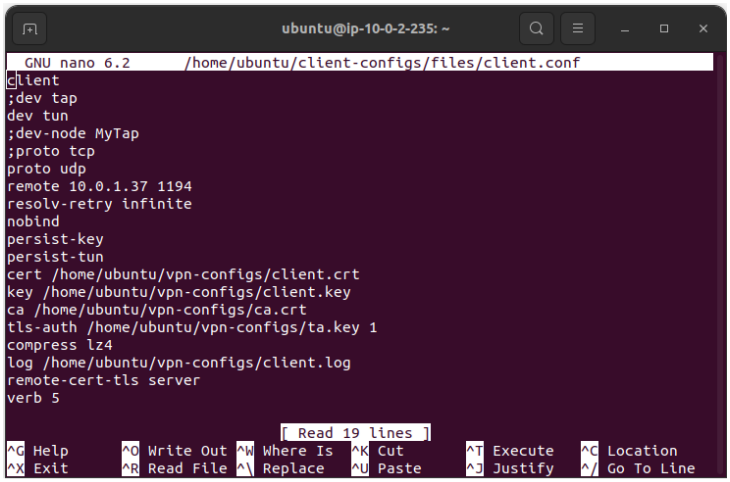
```
scp -i keyFuncionaris ~/Documents/vpn/client.crt ubuntu@52.213.183.195:/home/ubuntu/client.crt
scp -i keyFuncionaris ~/Documents/vpn/ca.crt ubuntu@52.213.183.195:/home/ubuntu/ca.crt
scp -i keyFuncionaris ~/Documents/vpn/client.key ubuntu@52.213.183.195:/home/ubuntu/client.key
scp -i keyFuncionaris ~/Documents/vpn/ta.key ubuntu@52.213.183.195:/home/ubuntu/ta.key
```

- Es mouen a la carpeta /etc/openvpn/ amb la instrucció

```
sudo mv ca.crt client.crt client.key ta.key /etc/openvpn/
```

- Es configura el client amb la instrucció

```
sudo nano ~/client-configs/files/client.conf
```



Configuració client VPN

- Amb la instrucció

```
sudo systemctl start openvpn@client.service
```

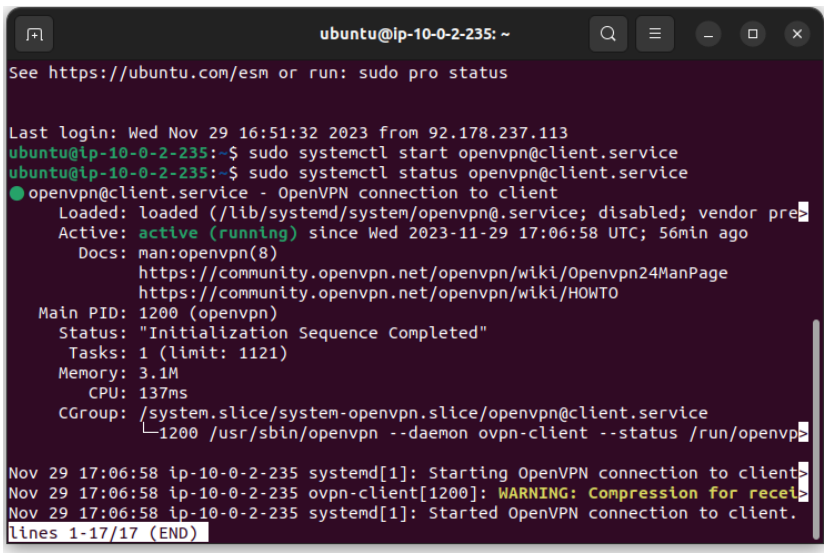
s'inicia el servidor i es pot veure l'estat del servidor amb la instrucció:

```
sudo systemctl status openvpn@client.service
```

- amb la instrucció

```
sudo systemctl enable openvpn@client.service
```

s'habilitarà l'arrencada automàtica del client.



Estat client VPN