

## **PREMIERE PARTIE : PHASE D'INSERTION**

### **Résumé :**

La phase d'insertion est la section introductory du rapport de stage. Elle présente l'entreprise ou l'organisation où le stage s'est déroulé. Cette partie décrit le fonctionnement, l'activité et l'environnement de la structure d'accueil. Elle expose également les conditions dans lesquelles le stagiaire a été intégré et accueilli au sein de cette entité.

### **Aperçu :**

#### **INTRODUCTION**

- I. ACCUEIL ET INTEGRATION**
- II. PRESENTATION DE WORK ET YAMO**
- III. RESSOURCES DE L'ENTREPRISE**

#### **CONCLUSION**

## **INTRODUCTION**

Le rapport d'insertion est un compte rendu de l'accueil dans la structure hôte. C'est le livrable attendu par l'instance académique pour notre projet. Durant cette période, nous avons l'opportunité de découvrir la structure d'accueil, de nous imprégner de son fonctionnement, et de nous familiariser avec son environnement logiciel et matériel. Dans ce contexte, comment Work et Yamo est-elle présentée de manière générale ? Quelles sont ses ressources et ses réalisations ? Répondre à ces questions constituera l'objet de notre insertion. Pour ce faire, nous structurerons notre rapport de la manière suivante : présentation organisationnelle incluant la création, la mission et les attributions ; présentation des ressources et des réalisations ; présentation du sujet de stage et du planning prévisionnel.

## **I. ACCUEIL ET INTEGRATION**

Le lundi 8 juillet 2024 à 8h00 a marqué le début de notre stage académique de trois mois au sein de l'entreprise Work et Yamo. Nous avons été accueillis chaleureusement par M. Haruna Rashid, notre maître de stage, qui nous a présenté les règles et procédures internes de l'entreprise. Par la suite, Idriss-François Eliguene directeur fondateur de Work et yamo, nous a exposé la structure de l'organisation ainsi que les diverses formations et certifications offertes.

Au cours des semaines d'insertion, nous avons suivi des cours en cyber sécurité sur la plateforme Udemy, ce qui nous a permis d'acquérir des compétences essentielles dans ce domaine. Lors de la deuxième semaine, nous avons participé à une conférence enrichissante qui nous a permis d'approfondir nos connaissances et d'élargir nos perspectives professionnelles. À l'issue de cette conférence, M. TCHINDE YANICK, notre encadrant professionnel, nous a attribué des thèmes spécifiques à traiter.

Nous avons conclu ces deux semaines d'insertion en nous familiarisant avec notre environnement de travail, posant ainsi les bases solides pour la suite de notre stage.

## **II. PRESENTATION DE WORK ET YAMO**

### **1. STRUCTURE ORGANISATIONNELLE DE WORK ET YAMO**

Work et yamo est une Start up qui a été créée en 2020 avec la vision audacieuse de combler le fossé numérique en Afrique. Les fondateurs ont commencé par élaborer des programmes de formation IT visant à offrir des opportunités aux jeunes africains. Avec un engagement indéfectible envers l'excellence ils visent à former les acteurs au premier plan de la démocratisation de l'accès à la technologie de pointe, en rendant les services IT accessibles, abordables et efficaces pour les entreprises africaines de toutes tailles.

Work et Yamo est structurée en plusieurs divisions spécialisées pour répondre de manière holistique aux besoins diversifiés de nos clients :

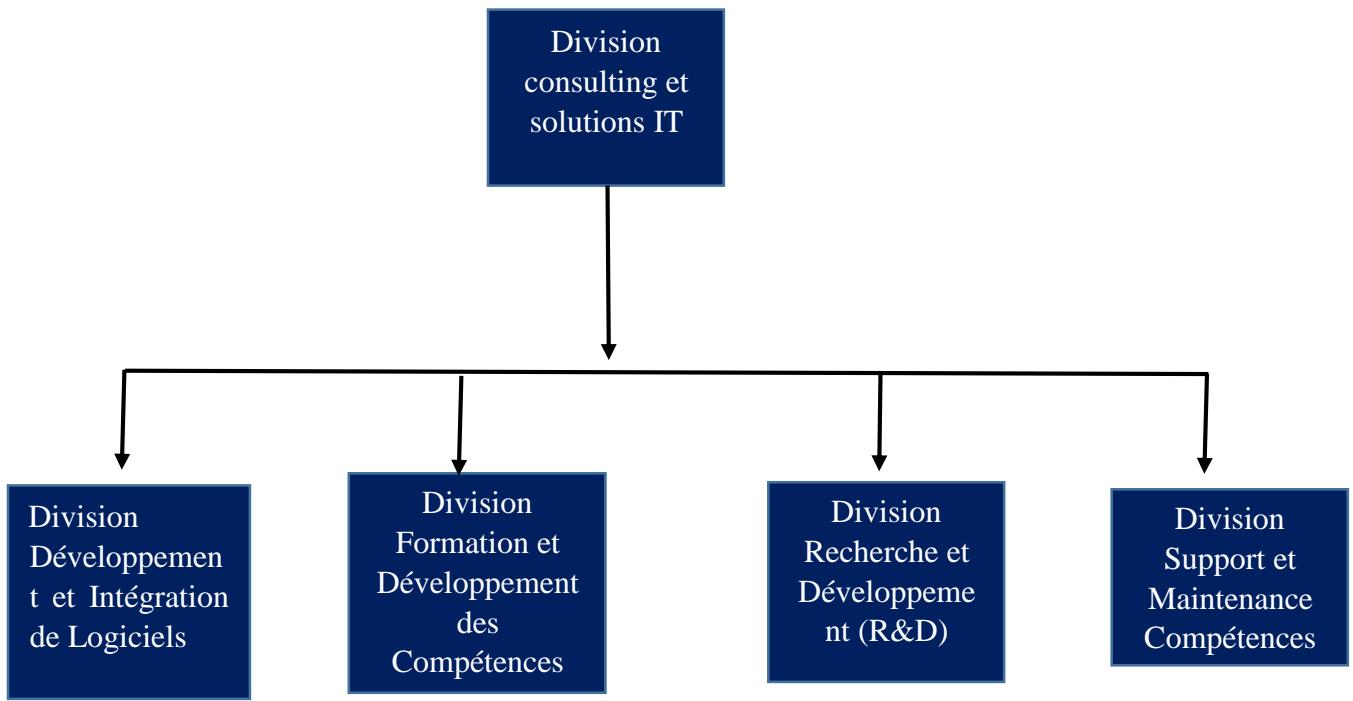


Figure 1: organigramme de Work et yamo

## 2. Présentation des différentes divisions

### 1. La division Consulting et Solutions IT

Le conseil détient les pouvoirs les plus étendus dans l'administration et la gestion de Work et Yamo. Il est notamment chargé de définir la politique générale de l'entreprise et de surveiller ses activités. Ses responsabilités incluent le contrôle de la gestion du personnel, la représentation de l'entreprise, la supervision des opérations, et la gestion des ressources. Le conseil prend également les décisions stratégiques et élabore les stratégies nécessaires à la réalisation des objectifs de l'entreprise. En résumé, la direction générale est responsable de la gestion globale et de l'atteinte des objectifs de Work et Yamo.

### 2. La division Consulting et Solutions IT

Cette division est chargée de fournir des conseils stratégiques et des solutions IT personnalisées. Elle collabore étroitement avec les entreprises pour comprendre leurs défis spécifiques et concevoir des solutions adaptées, allant de la gestion de l'infrastructure IT à la mise en œuvre de systèmes complexes.

### **3. La Division Développement et intégration**

Focusée sur la création de logiciels innovants, cette division se concentre sur le développement d'applications web, mobiles, et d'entreprise. Elle est également responsable de l'intégration de ces solutions dans les environnements existants des clients, assurant une transition en douceur et une interopérabilité optimale.

### **4. La division Formation et développement des compétences**

Convaincus que la clé du succès réside dans la formation continue, cette division propose des programmes de formation technique pour les entreprises et les individus. Elle développe des modules sur les technologies de pointe, assurant que nos clients et partenaires restent à la pointe de l'innovation.

### **5. La division Recherche et Développement (R&D)**

Pour rester en avance sur les tendances technologiques, IT Engineering Factory investit dans la R&D pour explorer de nouvelles technologies et solutions. Cette division est le cœur de notre innovation, travaillant sur des projets qui peuvent révolutionner les industries.

### **6. Division Support et Maintenance**

Un service de support client de classe mondiale est essentiel à notre mission. Cette division fournit un support technique continu, des services de maintenance proactive, et assure que les systèmes des clients fonctionnent de manière optimale 24/7

### III. RESSOURCES DE L'ENTREPRISE

#### 1. Ressources matérielles de l'entreprise

Tableau 1 : ressource matérielle de Work et yamo

Equipements	Marque	Quantité
Ordinateurs	➤ Dell ➤ Lenovo ➤ Hp	25
Imprimantes	Canon	02
Vidéo projecteur	Same screen	02
Commutateur	Huawei - S5700S-28P-LI-AC - Avec 24 ports 1000BASE-T	02
Téléviseurs	Hisense	05
Caméra IP de surveillance	Caméra IP PTZ et à dôme PTZ	15
climatiseur	NAGU	01
Onduleur	Light Wave DAVR 3000	01
Routeur	TP-Link Archer AX10	02
Batterie solaire	Felicitysolar	01

## 2. Ressources logicielles

*Tableau 2: ressource logicielle de Work et yamo*

Intituler	Version	Description
<b>Windows</b>	10, 11, server	Système développé par Microsoft
<b>Ubuntu</b>	12.1	Système d'exploitation Linux
<b>terraform</b>	1.10.0-alpha20240807	Est un outil qui permet de créer et de gérer des infrastructures informatiques (comme des serveurs, des réseaux, des bases de données) de manière automatisée et sécurisé.
<b>docker</b>	20.10.24	Plateforme de conteneurisation qui permet de développer, de déployer et de gérer des applications dans un conteneur logiciel.
<b>Gitlab et Github</b>	17.2 du 18 juillet 2024	Plateforme en ligne de développement de logiciel
<b>Xamp server et Nginx</b>	3.3.0	Server Local
<b>Visual code</b>	3.2.2 bluid 3211	Editeur de texte
<b>Suite office</b>	2021	Suite bureautique de référence utilisées pour la rédaction de notre document
<b>Suite adobe</b>	13.0.1x32	Logiciel pour la photographie et le graphisme

## **CONCLUSION**

À l'issue de cette période d'insertion, nous avons acquis une compréhension approfondie du fonctionnement hiérarchique et des diverses activités au sein de Work et yamo. Cette connaissance nous a permis de débuter notre stage sur des bases solides. Nous nous sommes également imprégnés des activités quotidiennes de l'académie, nous permettant ainsi de nous intégrer efficacement et de contribuer de manière pertinente aux projets en cours.

## **DEUXIEME PARTIE : PHASE TECHNIQUE**

### **Résumé :**

La phase technique est la deuxième phase de notre stage pendant laquelle on présente notre projet tout en faisant une étude profonde de l'existant, le critique de ce dernier, le décèlement de la problématique puis une proposition de solution afin de résoudre la problématique, tout cela en respectant les conditions du cahier des charges.

### **Aperçu :**

**CHAPITRE I : ANALYSE**

**CHAPITRE II : CAHIER DES CHARGES**

**CHAPITRE III : ETAT DE L'ART**

**CHAPITRE IV : IMPLEMENTATION DE LA  
SOLUTION**

**CHAPITRE V : RESULTATS ET COMMENTAIRES**

## **CHAPITRE I : ANALYSE**

*Aperçu :*

### **INTRODUCTION**

- I. PRESENTATION DU THEME**
- II. ETUDE DE L'EXISTANT**
- III. CRITIQUE DE L'EXISTANT**
- IV. PROBLEMATIQUE**
- V. PROPOSITION DE SOLUTION**

### **CONCLUSION**

## **INTRODUCTION**

Dans ce chapitre, intitulé ANALYSE, nous allons explorer notre projet en profondeur. Il s'agit d'examiner la situation actuelle, d'en faire une critique honnête, puis de proposer une solution concrète basée sur les problématiques que nous aurons identifiées. Cette partie est essentielle et primordiale, car c'est ici que nous allons véritablement comprendre l'entreprise, détecter ses failles, et proposer les ajustements nécessaires pour la faire évoluer.

## I. PRESENTATION

Pour valider notre diplôme d'ingénieur Informatique, nous avons entrepris un stage de 3 mois, durant lequel l'entreprise nous a confié la mission de travailler sur le thème « **Etude et implémentation de la sécurité dans le cloud : cas de AWS** ». Ce thème vise à étudier et à implémenter des solutions de sécurité dans le cloud, spécifiquement sur AWS. L'objectif est de superviser les différentes tâches liées à la sécurité des services hébergés sur AWS.

La sécurité dans le cloud consiste à surveiller en permanence l'état des ressources (comme les instances EC2, les bases de données, etc.), les performances des applications, ainsi que la connectivité et la sécurité des services déployés. Cette supervision sera réalisée à l'aide d'outils de sécurité intégrés à AWS, que nous présenterons plus en détail dans la suite de notre rapport.

## II. ETUDE DE L'EXISTANT

AWS (Amazon Web Services) est une plateforme de cloud computing qui propose divers outils pour sécuriser les données et les applications. La sécurité sur AWS repose sur une division des responsabilités : AWS protège ses infrastructures (serveurs, réseaux, etc.), tandis que les clients sont responsables de sécuriser leurs propres données et applications.

Cependant on peut voir que son architecture est constituée des différents services qui sont :

- IAM (identity and access manager) : qui gère les utilisateurs, le contrôle d'accès et les politiques d'accès.
- La sécurité du réseau : qui est géré par de multiples services qui sont VPC (création de réseau privés virtuels avec des sous-réseaux, des groupes de sécurités, et des listes de contrôles d'accès.), AWS SHIELD (conçu pour parer les attaques sophistiquées.), WAF (pour web application firewall est conçu pour protéger les applications web contre les menaces communes).
- La surveillance et la gestion des logs : géré par Cloudtrail (journalisation des appels d'API pour suivre les actions des utilisateurs et des services.), cloudWatch (surveillance en temps réel des ressources AWS avec des métriques et des alertes.), aws config (suivi des configurations des ressources AWS et gestion des modifications).
- Le Cryptage et la protection des données : géré par AWS Key Management Service (KMS) (Gestion des clés de cryptage pour protéger les données), AWS Secrets Manager (Stockage sécurisé et gestion des secrets tels que les mots de passe et les clés API.), S3 Server-Side Encryption (Cryptage des données au repos dans Amazon S3.)

- La gestion des vulnérabilités : géré par Amazon Inspector (Évaluation automatique des vulnérabilités des instances EC2.), AWS Security Hub (Centralisation des alertes de sécurité et des résultats d'évaluation.).

### III. CRITIQUE DE L'EXISTANT

AWS possède de nombreux atouts en terme de sécurité et nous pouvons citer entre autre :

- Une large gamme d'outils : il propose une grande variété d'outils pour sécuriser les données, les applications et les réseaux. Cela inclut des fonctionnalités pour gérer les accès, protéger les réseaux, surveiller les activités, et sécuriser les données.
- Une responsabilité partagée : il clarifie bien la répartition des responsabilités en matière de sécurité. AWS se charge de sécuriser l'infrastructure (serveurs, réseaux), tandis que les clients sont responsables de sécuriser leurs propres données et applications.
- La surveillance et le suivi : Les outils de surveillance comme CloudWatch et CloudTrail permettent de suivre en temps réel l'état des ressources et de voir l'historique des actions. Cela aide à détecter rapidement des problèmes ou des anomalies.
- La protection des données : il offre des options robustes pour le cryptage des données, garantissant que les informations sensibles sont protégées contre les accès non autorisés.

A côté de cette multitude d'avantage résident des inconvénients à améliorer, on peut citer entre autre :

- La gestion des vulnérabilités : Bien qu'il propose des outils pour évaluer les vulnérabilités, la détection et la gestion des risques restent une responsabilité du client. Il est essentiel de mettre en place des pratiques proactives pour identifier et corriger les failles de sécurité.
- La responsabilité client : Bien que la répartition des responsabilités soit claire, les clients doivent être bien informés et vigilants pour sécuriser leurs propres données et applications. Une mauvaise configuration de leur part peut entraîner des risques importants.
- La complexité de configuration : Certains outils de sécurité peuvent être complexes à configurer, surtout pour les utilisateurs qui ne sont pas familiarisés avec les concepts techniques. Cela peut rendre difficile la mise en place d'une sécurité efficace sans une certaine expertise.
- Le Coûts : La mise en œuvre de toutes les fonctionnalités de sécurité peut entraîner des coûts supplémentaires. Les entreprises doivent évaluer attentivement leurs besoins en sécurité pour éviter des dépenses excessives.

## **IV. PROBLEMATIQUE**

Les éléments cités plus haut dans l'étude de l'existant nous ont permis d'avoir une vue globale du fonctionnement de AWS car nous avons pu déceler dans la critique de l'existant plusieurs manques et faiblesses.

Ainsi nous sommes parvenus à problématique partant de ce problème :

**« Comment les organisations peuvent-elles optimiser l'implémentation des solutions de sécurité proposées par AWS pour garantir une protection efficace de leurs données et infrastructures tout en simplifiant la gestion et la configuration des outils ? »**

## **V. PROPOSITION DE SOLUTION**

Après une analyse approfondie de l'existant et l'identification des problèmes, nous proposons la mise en place d'un système de sécurité basé sur aws pour renforcer la protection des données et des ressources dans le cloud cette solution permettre aux administrateurs de surveiller et de gérer la sécurité des ressources AWS en temps réel, sans attendre qu'un incident se produise.

La solution proposée se concentre sur « **l'étude et l'implémentation de la sécurité dans le cloud : cas d'AWS** ». Elle offrira plusieurs avantages clés tels que la surveillance proactive, la gestion des risques, automatisation de la sécurité, optimisation des ressources.

## **CONCLUSION**

Pour conclure ce chapitre, l'objectif était de réaliser une analyse approfondie et détaillée de notre entreprise d'accueil, d'identifier ses problèmes, de les critiquer, de dégager une problématique, puis de proposer une solution pour résoudre cette problématique. Ce chapitre constitue la pierre angulaire du projet, et l'étude qui y est présentée a fourni une base solide pour les étapes suivantes du projet.

## CHAPITRE II : CAHIER DES CHARGES

*Aperçu :*

### Introduction

- I. CONTEXTE ET JUSTIFICATION DE L'ETUDE  
DU PROJET
- II. LES OBJECTIFS DU PRODUIT
- III. EXPRESSION DES BESOINS
- IV. ESTIMATION DU PROJET
- V. PLANIFICATION DU PROJET
- VI. CONTRAINTE DU PROJET
- VII. LES LIVRABLES

### Conclusion

## **INTRODUCTION**

À la fin de la phase d'intégration, notre mission consiste à faire une étude et à implémenter une solution de sécurité cloud dont nous prenons le cas spécifique de AWS, axée sur la protection des données et des infrastructures critiques. Avant de débuter, il est essentiel de réaliser une analyse approfondie des pratiques actuelles en matière de sécurité, afin d'identifier les vulnérabilités et les lacunes potentielles du système. Ce document, qui constitue le cahier des charges, présente une description de l'état actuel de la sécurité, ainsi que les exigences fonctionnelles et non fonctionnelles, les coûts, et les délais pour la mise en place de la solution de sécurité cloud. Le plan se décomposera comme suit : contexte, problématique, objectifs, méthodologie, recueil des exigences, modélisation du contexte de sécurité, livrables attendus, et planification prévisionnelle.

## **I. CONTEXTE ET JUSTIFICATION DU PROJET**

Dans un monde où les menaces cybernétiques sont de plus en plus sophistiquées, garantir la sécurité des données et des infrastructures cloud est devenu une priorité absolue pour les entreprises. La sécurisation de l'infrastructure cloud AWS est essentielle non seulement pour protéger les informations sensibles mais aussi pour assurer la continuité des opérations et maintenir la confiance des clients.

La mise en place d'une solution robuste de sécurité dans le cloud est indispensable pour répondre aux défis actuels de la cybersécurité. Le projet vise à étudier et à implémenter des mesures de sécurité adaptées à l'environnement AWS, afin de protéger les données critiques et garantir le bon fonctionnement des applications hébergées sur cette plateforme.

Il est crucial pour les entreprises de disposer d'une architecture de sécurité cloud efficace pour éviter les risques de violations de données, d'accès non autorisé, ou d'autres incidents de sécurité. Le besoin de renforcer la sécurité des systèmes cloud découle de la nécessité de protéger les informations sensibles contre les menaces potentielles, tout en garantissant la conformité aux normes et réglementations en vigueur.

Ce projet se justifie par l'importance de sécuriser les environnements cloud pour assurer la confidentialité, l'intégrité, et la disponibilité des données. Il permettra de définir des politiques de sécurité adaptées, de mettre en place des mécanismes de protection appropriés, et de répondre aux exigences spécifiques pour maintenir une posture de sécurité solide dans le cloud AWS.

## **II. OBJECTIF**

### **1. OBJECTIF GLOBAL**

Nous visons à mettre en place une solution de sécurité cloud qui assurera une protection efficace des données et des ressources, tout en offrant une visibilité en temps réel sur l'état de l'infrastructure cloud. Cette solution réduira considérablement les risques de vulnérabilités et de menaces potentielles, permettant ainsi aux entreprises de mieux gérer leur environnement AWS et d'assurer une sécurité robuste sans tracas.

### **2. OBJECTIF SPECIFIQUE**

Il s'agira, à travers cette problématique, d'effectuer une analyse approfondie afin de proposer et de mettre en place une solution de sécurité cloud sur AWS qui optimisera la gestion de l'infrastructure. Les objectifs spécifiques de ce projet sont les suivants :

- Assurer une accessibilité continue et sécurisée aux données et ressources cloud ;
- Mettre en place des mécanismes de suivi en temps réel de l'évolution des mesures de sécurité et des incidents ;
- Renforcer la posture de sécurité globale de l'entreprise ;
- Classifier les ressources et les accès selon des critères de sécurité définis ;
- Évaluer régulièrement les menaces potentielles et les vulnérabilités pour améliorer la résilience du système.

### III. LES BESOINS FONCTIONNELS DU FUTUR SYSTEME

#### 1. BESOINS FONCTIONNELS

Les besoins fonctionnels représentent les fonctionnalités essentielles pour garantir la sécurité et l'efficacité du système dans l'environnement cloud AWS. Sans ces fonctionnalités, le système ne serait pas complet ni conforme aux exigences de sécurité. Après avoir consulté les parties prenantes du projet et analysé leurs exigences en matière de sécurité, nous avons identifié plusieurs fonctionnalités clés, organisées en modules logiques. Ces modules regroupent les fonctionnalités nécessaires pour assurer une gestion sécurisée et efficace des ressources et des données dans le cloud AWS :

- **Gestion des comptes** : Implémentation de mécanismes de sécurité pour la gestion des identités et des accès, y compris l'authentification multi-facteurs et les contrôles d'accès basés sur les rôles (RBAC).
- **Gestion des incidents** : Surveillance et réponse aux incidents de sécurité, intégrant des alertes et des procédures de réponse automatisées pour détecter et traiter les menaces en temps réel.
- **Gestion des configurations** : Suivi et gestion des configurations de sécurité des ressources AWS, avec des outils pour l'audit et la conformité des paramètres de sécurité.
- **Gestion des accès** : Contrôle et gestion des permissions d'accès aux ressources, en s'assurant que les politiques de sécurité sont appliquées correctement et que les accès sont adaptés aux besoins des utilisateurs.
- **Gestion des logs** : Collecte, stockage et analyse des journaux d'activité pour détecter les anomalies et les incidents de sécurité, garantissant une visibilité complète sur les actions et les changements dans l'environnement cloud.
- **Protection des données** : Chiffrement des données au repos et en transit, Contrôle des accès aux données, Gestion des clés de chiffrement, Sauvegarde et restauration des données.
- **Gestion de la sécurité des applications et des infrastructures** : Sécurisation des interfaces et APIs, configuration des Pare-feux, systèmes de détection et de prévention d'intrusions, Sécurité des conteneurs et des machines virtuelles, Gestion des correctifs et des vulnérabilités

## a. Gestion des comptes

La gestion des comptes dans le cadre de la sécurité cloud AWS est effectuée par un administrateur, qui est responsable de la gestion des identités et des accès des utilisateurs. Ses tâches incluent :

- **Ajouter** : Créer de nouveaux comptes utilisateurs, en attribuant les rôles et les permissions appropriés en fonction des besoins de sécurité.
- **Modifier** : Mettre à jour les informations des utilisateurs, ajuster les permissions et les rôles selon les changements dans les besoins ou les responsabilités.
- **Consulter** : Examiner les détails des comptes utilisateurs, y compris les droits d'accès et les activités, pour assurer la conformité et détecter les anomalies.
- **Supprimer** : Retirer les comptes utilisateurs lorsqu'ils ne sont plus nécessaires ou lorsqu'un utilisateur quitte l'organisation, en garantissant que les accès sont révoqués de manière sécurisée.

## b. Gestion des incidents

La gestion des incidents est cruciale pour maintenir la sécurité et l'intégrité du système cloud. Cette fonction est effectuée par une équipe de sécurité dédiée qui peut :

- **Déetecter** : Surveiller en continu les systèmes pour identifier les incidents de sécurité en temps réel, en utilisant des outils d'analyse et des systèmes de détection d'intrusion.
- **Répondre** : Mettre en œuvre des procédures de réponse aux incidents pour contenir et résoudre les menaces rapidement, minimisant ainsi les impacts sur les opérations.
- **Analyser** : Examiner les incidents pour comprendre leur origine, leur impact et leur portée, afin de renforcer les mesures de sécurité et de prévenir les incidents futurs.
- **Documenter** : Enregistrer tous les détails des incidents, y compris les actions prises et les leçons apprises, pour améliorer les pratiques de sécurité et se conformer aux exigences réglementaires.

## c. Gestion des configurations

La gestion des configurations est essentielle pour assurer la conformité et la sécurité des ressources cloud. Cette fonction inclut :

- **Surveiller** : Contrôler en continu les configurations des ressources AWS pour détecter les écarts par rapport aux normes de sécurité établies.
- **Configurer** : Appliquer les paramètres de sécurité recommandés, tels que les groupes de sécurité, les listes de contrôle d'accès (ACL) et les configurations des instances.
- **Auditer** : Réaliser des audits réguliers des configurations pour vérifier leur conformité avec les politiques de sécurité et identifier les vulnérabilités potentielles.
- **Documenter** : Maintenir une documentation détaillée des configurations des ressources, des changements apportés et des exceptions autorisées, pour assurer la traçabilité et la conformité.

### d. Gestion des accès

La gestion des accès assure que les permissions sont correctement attribuées et maintenues selon les besoins de sécurité. Les responsabilités incluent :

- **Attribuer** : Définir et attribuer les rôles et permissions appropriés aux utilisateurs et groupes, en fonction de leurs besoins spécifiques.
- **Révoquer** : Retirer les accès lorsqu'un utilisateur n'en a plus besoin ou lorsque des rôles changent, garantissant ainsi que les permissions ne sont pas excessives.
- **Réviser** : Effectuer des revues régulières des permissions et des rôles pour vérifier leur adéquation et ajuster les accès en fonction des changements dans l'organisation ou les politiques de sécurité.
- **Surveiller** : Suivre les accès aux ressources pour détecter les activités non autorisées ou suspectes, et ajuster les contrôles d'accès en conséquence.

### e. Gestion des logs

La gestion des logs est essentielle pour la surveillance et l'analyse des activités du système. Cette fonction comprend :

- **Collecter** : Rassembler les journaux d'activité provenant de différentes sources, telles que les services AWS, les applications et les systèmes de sécurité.
- **Analyser** : Examiner les logs pour identifier les anomalies, les tentatives d'intrusion et les autres incidents de sécurité potentiels.
- **Stocker** : Conserver les logs de manière sécurisée et conformément aux exigences de rétention pour garantir leur disponibilité en cas d'audit ou d'analyse d'incidents.
- **Visualiser** : Utiliser des outils de visualisation et de reporting pour faciliter l'analyse des données des logs et fournir des informations exploitables sur l'état de la sécurité du système.

### f. Gestion de la protection des données

La gestion de la Protection des données passe par :

- Chiffrement des données au repos et en transit,
- Gestion des clés de chiffrement,
- Sauvegarde et restauration des données.

### g. Gestion de la sécurité des applications et des infrastructures

La Gestion de la sécurité des applications et des infrastructures passe par la

- Sécurisation des interfaces et APIs,
- Configuration des Pare-feux,

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

- Systèmes de détection et de prévention d'intrusions,
- Sécurité des conteneurs et des machines virtuelles,
- Gestion des correctifs et des vulnérabilités

## 2. BESOINS NON FONCTIONNELS

Les besoins non fonctionnels sont essentiels pour garantir la qualité globale de la solution de sécurité dans le cloud, même s'ils ne résolvent pas directement le problème de sécurité. Voici les besoins non fonctionnels adaptés à l'implémentation de la sécurité sur AWS :

- **Confidentialité** : Assurer la confidentialité des données stockées et traitées dans le cloud en implémentant des mécanismes de chiffrement robustes, une gestion des accès rigoureuse et des politiques de sécurité strictes.
- **Intégrité** : Garantir l'intégrité des données en transit et au repos, en détectant toute modification non autorisée et en mettant en place des mécanismes de vérification d'intégrité.
- **Disponibilité** : Assurer la disponibilité des ressources et services cloud, en implémentant des systèmes redondants, des sauvegardes régulières et des plans de reprise d'activité.
- **Authentification et Autorisation** : Mettre en place des mécanismes d'authentification solides (ex : authentification multi-facteurs) et un contrôle d'accès granulaire pour limiter les accès aux seules personnes autorisées.
- **Journalisation et Surveillance** : Mettre en place des systèmes de journalisation et de surveillance pour détecter les activités suspectes et pouvoir retracer les événements en cas d'incident.
- **Conformité** : Respecter les réglementations et normes de sécurité applicables pour le stockage et le traitement des données dans le cloud.
- **Résilience** : Concevoir des architectures cloud résilientes capables de résister à des pannes, des attaques ou des catastrophes naturelles sans interruption de service.
- **Évolutivité** : Permettre une évolutivité dynamique des mécanismes de sécurité pour s'adapter aux changements de charge et aux nouvelles menaces.
- **Portabilité** : Assurer la portabilité des données et des applications cloud pour éviter le verrouillage avec un fournisseur et faciliter les migrations.
- **Transparence** : Fournir une visibilité et une transparence sur les mesures de sécurité mises en place par le fournisseur cloud pour gagner la confiance des utilisateurs.

## IV. ESTIMATION DU COUT DU PROJET

Vu la complexité et l'envergure du projet, ainsi que les différents modules à mettre en place et les équipements nécessaires, il est crucial de réaliser une évaluation détaillée des coûts. Cette évaluation permettra de planifier efficacement les ressources et de gérer le budget du projet de manière transparente. Les coûts seront détaillés dans les tableaux ci-dessous.

## 1. Ressources humaines

*Tableau 3: Ressources humaines*

Postes	Nombres participants de	Prix semaine par	Nombre de jours	Prix par semaine (en Franc CFA)	Prix Total
Architecte de sécurité cloud	01	Cahier des charges	5	2 500 000	2 500 000
Ingénieur Sécurité cloud	01	Analyse et conception	15	4 500 000	4 500 000
Analyste de sécurité	01	Conception et Insertion	15	3 000 000	6 000 000
Testeur de pénétration	01	Test et validation	11	500 000	500 000
Total					13 000 000

## 2. Ressources matérielles

En ce qui concerne l'équipement à utiliser pour le développement de notre application, nous aurons besoins du matériel suivant :

*Tableau 4: Ressources matérielles*

Libellé	Quantité	Prix unitaire (En FCFA)	Prix total (En FCFA)
<b>Rame de format A4</b>	1	4500	4500
<b>Stylo à bille bleu</b>	2	400	800
<b>Modem</b>	1	20 000	20 000
<b>Laptop Lenovo</b>	1	368 000	368 000
<b>Imprimante</b>	1	70 000	70 000
		TOTAL	463 300

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

## 3. Ressources logiciels

Pour la réalisation de ce projet, nous aurons besoins des ressources logicielles suivantes :

Tableau 5: Ressources logiciel

OUTILS	VERSION	EDITEUR	DESCRIPTION
GANTT	8	Editeur de code Source	Editeur de diagramme
Edge	61	Microsoft	Navigateur
Microsoft Office	2016	Microsoft	Suite bureautique
Edraw Max	10.5.5	Wondershare	Création de plan de localisation

The screenshot shows the AWS Pricing Calculator interface. At the top, it displays a summary of the estimate: Initial Cost (0.00 USD), Monthly Cost (434.58 USD), and Total Cost (5214.96 USD, including the initial cost). On the right, there are buttons for 'Démarrer avec AWS' (Launch with AWS), 'Démarrer gratuitement' (Launch for free), and 'Contacter l'équipe commerciale' (Contact sales team). Below the summary, the 'My Estimate' section lists various AWS services with their costs and descriptions. Services listed include Amazon Simple Storage Service (S3), Amazon Route 53, AWS CloudTrail, AWS Key Management Service, Amazon EC2, AWS IAM Access Analyzer, AWS Web Application Firewall (WAF), Amazon CloudFront, and Amazon CloudWatch. Each service entry provides a detailed breakdown of its usage and associated costs. At the bottom of the page, there is a note about the accuracy of the estimation and a link to learn more.

Figure 2:estimation des différents service de aws

Le coût total de pour ces services sur 3 mois s'élève à 1303.74 \$ c'est-à-dire 774894,95 FCFA.

### Récapitulatif du coup du projet

Tableau 6: récapitulatif du projet

RESSOURCES	MONTANT EN Fcfa
Ressources humaines	13 000 000
Ressources matérielles	463 300
Ressource logicielle	<b>774894,95</b>
Total	<b>14 238 194,95</b>

## V. PLANIFICATION DU PROJET

La planification du projet bien que nécessaire pour une bonne structuration du temps qui nous est donnée pour la réalisation du dit projet est une étape délicate dans notre travail. Elle a été réalisée en fonction du thème qui nous a été soumis et du temps dédié suivant le principe de Gantt et via l'outil Gant projet.

### Acteurs du projet

Tableau 7: Acteurs du projet

Noms et Prénoms	Rôles	Fonction
<b>Work et Yamo</b>	Maitre d'ouvrage	Entreprise d'accueil
<b>Mme. KHALIDA</b>	Encadrant Académique	Enseignant à l'IAI
<b>TCHINDE Yanick</b>	Encadrant Professionnelle	Architecte Cloud

### Planning de travail

Le stage se déroulera sur une période de 3 mois allant du 10 juillet au 30 septembre 2023. Pour la réalisation du projet, les tâches seront réparties comme suit :

- Période d'insertion : 9 jours, du 01 juillet au 11 juillet 2024 ;
- Analyse du projet : 5 jours, du 12 juillet au 18 juillet 2024.
- Cahier des charges : 5 jours, du 19 juillet au 25 juillet 2024.
- Etat de l'art : 1 jours, du 26 août au 27 août 2024.
- Implémentation de la solution : 10 jours, du 29 juillet au 09 août 2024.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

- Résultats et commentaires : 6 jours, du 12 au 19 août 2024.
- Mise en forme, correction et finalisation du rapport : 04 jours du 20 au 23 août.

Toutes ces différentes étapes sont décrites graphiquement par le Diagramme de Gantt suivant :

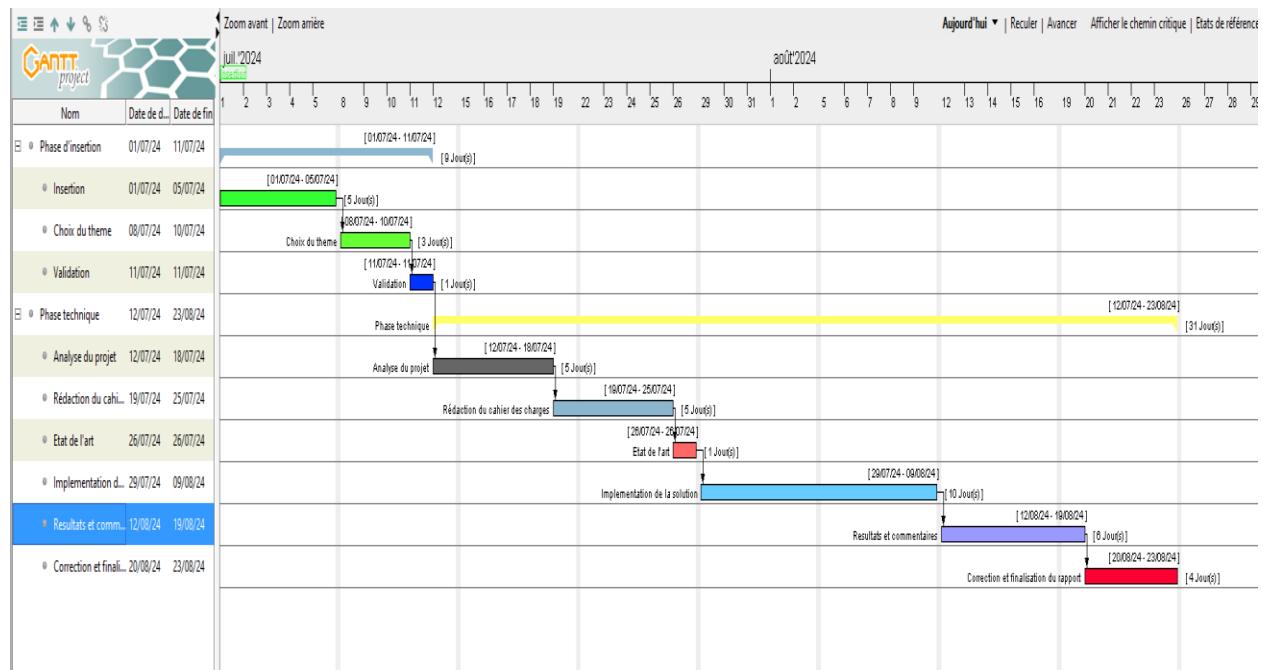


Figure 3: diagramme de gantt

## VI. LES CONTRAINTES DU PROJET

### 1. Les contraintes de cout

La réalisation de notre projet entraînera des dépenses pour les ressources humaines, matérielles et logicielles, s'élevant à un montant total de 15 447 488,3442 FCFA.

### 2. Les contraintes de temps

Notre projet devra être complété sur une période de trois mois, du 3 juillet au 30 Août.

### 3. Les contraintes de qualités

La mise en œuvre de notre projet devra respecter les contraintes de qualité suivantes :

- **Robustesse** : Le système devra fonctionner avec un minimum d'erreurs, garantissant ainsi une haute fiabilité.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

- **Évolutivité** : Il devra être capable de s'adapter aux évolutions et aux nouvelles exigences qui pourraient surgir au cours du temps.
- **Ergonomie** : Le système devra offrir un équilibre efficace entre les besoins fonctionnels et non fonctionnels, assurant une utilisation fluide et intuitive

## VII. LES LIVRABLES DU PROJET

À l'issue de cette étude, nous devrons fournir une solution complète de sécurité dans le cloud, comprenant :

- **Un rapport détaillé**, incluant :
  - Un cahier des charges
  - Un dossier d'analyse
  - Un dossier de conception
  - Un dossier de réalisation
  - Un guide d'utilisation
- **Un CD/DVD** contenant :
  - La présentation PowerPoint
  - La documentation complète
  - Les configurations et les scripts associés au projet

## **CONCLUSION**

Après avoir étudié les exigences du maître d'ouvrage, nous avons élaboré le cahier des charges présenté ci-dessus. Ce document servira de référence pour la validation du futur système, puisqu'il détaille toutes les spécifications et caractéristiques nécessaires. Les exigences du système étant désormais clairement définies, nous pouvons procéder à l'état de l'art.

## CHAPITRE 3 : ETAT DE L'ART

*Aperçu :*

### INTRODUCTION

- I. GENERALITES
- II. COMPARAISON DES DIFFERENTES SOLUTIONS
- III. CHOIX DE LA SOLUTION

### CONCLUSION

## **INTRODUCTION**

Le troisième chapitre de notre projet, intitulé "État de l'art", se concentre sur la présentation des généralités liées au thème abordé. Il inclut l'étude des différentes solutions disponibles, une analyse comparative de ces solutions, et la justification du choix de la solution retenue. De plus, il comprend une discussion sur les différents services associés au thème.

## **I. GENERALITES**

### **A. GENERALITES SUR LE CLOUD**

#### **1. Définition du cloud**

Le cloud computing est l'accès à la demande à des ressources IT sur internet avec une tarification proportionnelle à l'utilisation. Que vous exécutez des applications qui partagent des photos avec des millions d'utilisateurs mobiles ou que vous preniez en charge les opérations critiques de votre entreprise, une plate-forme de services cloud garantit un accès rapide à des ressources informatiques flexibles et à faible coût.

Grâce au cloud computing, vous n'avez plus à engager d'importants investissements initiaux en matériel et à consacrer un temps considérable dans les tâches chronophages liées à la gestion de ce matériel. Au lieu de cela, vous pouvez louer exactement le type et la quantité de ressources de calcul pour donner vie à vos toutes dernières idées ou les mettre au service de votre département informatique. Vous pouvez accéder à autant de ressources que vous en avez besoin, et ce, quasi instantanément, et payer uniquement pour celles que vous utilisez. Le cloud computing offre un moyen simple d'accéder à des serveurs, des espaces de stockage, des bases de données et une vaste gamme de services d'application sur Internet. Une plate-forme de services cloud telle qu'Amazon Web Services possède et assure la maintenance du matériel connecté au réseau nécessaire pour ces services d'application, tandis que vous louez et utilisez les ressources dont vous avez besoin via une application web.

#### **2. Caractéristiques essentielles**

Le cloud computing a transformé l'accès aux ressources informatiques en offrant de nouvelles possibilités. Pour mieux comprendre cette évolution, examinons les caractéristiques clés qui définissent le cloud et façonnent son utilisation moderne.

##### **a. Accès aux services par utilisateur à la demande**

Les utilisateurs peuvent accéder aux services cloud de manière instantanée, sans intervention préalable du fournisseur. Cette fonctionnalité permet aux utilisateurs de provisionner des ressources et d'utiliser des services selon leurs besoins, quand ils le souhaitent, sans délais ni contraintes.

##### **b. Accès réseau large bande**

Les services cloud sont accessibles via des connexions réseau de large bande, ce qui garantit une communication rapide et fluide entre les utilisateurs et les ressources cloud. Cette caractéristique permet un transfert efficace des données et un accès rapide aux applications et aux services hébergés dans le cloud.

### c. Réservoir de ressources (non localisées)

Les ressources cloud, telles que le stockage et le traitement, sont regroupées dans des centres de données distribués géographiquement, plutôt que d'être localisées sur un site physique unique. Cela permet une meilleure gestion des ressources, une réduction des coûts et une résilience accrue en cas de panne ou de défaillance d'un site.

### d. Redimensionnement rapide (élasticité)

Le cloud computing offre la capacité de redimensionner rapidement les ressources en fonction de la demande. Cette élasticité permet d'ajuster instantanément la capacité de traitement et de stockage en réponse à des variations de charge, assurant ainsi une performance optimale et une gestion efficace des coûts.

### e. Facturation à l'usage

Les services cloud sont facturés en fonction de l'utilisation réelle des ressources. Les utilisateurs paient uniquement pour les ressources qu'ils consomment, ce qui permet une gestion des coûts plus précise et une adaptation aux besoins spécifiques de chaque utilisateur ou entreprise.

### f. Les types de cloud

- **Cloud public** : Les ressources sont partagées entre plusieurs clients. Cette infrastructure est gérée par des entreprises spécialisée comme aws, Google Cloud, Microsoft Azure.
- **Cloud privé** : Les ressources sont dédiées à une seule organisation qui peut être gérée en interne ou par un tiers et peut être hébergée sur site ou hors site.
- **Cloud hybride** : Combinaison des deux, permettant de tirer parti des avantages de chacun.
- **Cloud Communautaire** : Infrastructure partagée par plusieurs organisations ayant des préoccupations communes (sécurité, conformité, etc.).

### g. Les Modèles de cloud computing

- **Infrastructure en tant que Service (IaaS)** : Offre des ressources de base comme des serveurs, du stockage, et des réseaux. Les utilisateurs peuvent installer et gérer des systèmes d'exploitation et des applications. Exemples : Amazon EC2, Google Compute Engine.
- **Plateforme en tant que Service (PaaS)** : Fournit un environnement pour le développement, la gestion et l'exécution d'applications. Les utilisateurs gèrent les applications et les données, tandis que le fournisseur gère l'infrastructure sous-jacente. Exemples : Google App Engine, Microsoft Azure App Services.
- **Logiciel en tant que Service (SaaS)** : Fournit des applications complètes via Internet, accessibles par des navigateurs web. Les utilisateurs ne gèrent ni l'infrastructure ni la plateforme. Exemples : Gmail, Salesforce, Office 365

### **h. Technologies et services connexes**

Le cloud computing est doté d'une panoplie de technologie qui sont favorable à son extension et celle de ses clients, on peut citer entre autres :

- **Virtualisation** : Technologie que vous pouvez utiliser pour créer des représentations virtuelles de serveurs, de stockage, de réseaux et d'autres machines physiques.
- **Conteneurisation** : Méthode pour emballer des applications et leurs dépendances, offrant une plus grande portabilité et une gestion efficace des ressources. Exemples : Docker, Kubernetes.
- **Big Data et Analyse** : Les services cloud offrent des outils puissants pour le traitement et l'analyse des grandes quantités de données.
- **Intelligence Artificielle et Machine Learning** : Accès à des services d'IA/ML pour l'analyse des données et l'automatisation.

## **B. GENERALITES SUR LA SECURITE CLOUD**

### **1. Définition**

La sécurité, dans le contexte général, se réfère à la protection contre les risques, les menaces et les vulnérabilités afin de préserver l'intégrité, la confidentialité et la disponibilité des actifs, des systèmes et des informations. Elle englobe diverses pratiques, technologies et stratégies mises en place pour protéger les personnes, les biens, les informations et les systèmes contre les dangers potentiels. Le principe de la sécurité dans le cloud reste le même. Les objectifs restant inchangés il est question de sécuriser les infrastructures et les données qui font du cloud ce qu'il est. Ce qui permet de faire fonctionner le cloud aws sont ses différents services et actuellement, il en possède plus de 175 qui sont chacun regroupé par thème. Nous travaillerons principalement dans ce devoir avec les principaux services destinés aux fonctionnement et à la sécurité.

La sécurité du cloud pour AWS est une priorité absolue. En tant que client d'AWS, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des organisations les plus pointilleuses en termes de sécurité. La sécurité dans le cloud est comparable à celle dans vos centres de données sur site, mais sans les coûts de maintenance des sites et du matériel. Dans le cloud, vous n'avez pas à gérer de serveurs physiques ou de périphériques de stockage. A la place, vous utilisez des outils de sécurité basés sur des logiciels pour assurer la surveillance et la protection du flux d'informations entrant et sortant de vos ressources cloud. Un des avantages du cloud AWS est que celui-ci vous permet de vous mettre à l'échelle et d'innover, tout en assurant la sécurité de l'environnement et en payant uniquement pour les services que vous utilisez.

La sécurité dans le cloud AWS est un sujet central pour les entreprises qui adoptent les services cloud. AWS propose une infrastructure sécurisée, avec une variété de services et d'outils pour

## **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---

protéger les données, gérer les accès et garantir la conformité aux normes. Voici une présentation générale de la sécurité dans le cloud AWS :

### **2. Introduction à la Sécurité AWS**

AWS adopte un modèle de sécurité partagée, où AWS est responsable de la sécurité du cloud (infrastructure physique, matériel, réseau global), tandis que les clients sont responsables de la sécurité dans le cloud (configurations des services, gestion des identités et des accès, données).

### **3. Les principaux services de sécurité de aws**

AWS propose une suite complète de services de sécurité conçus pour protéger vos données et applications dans le cloud. Ces services sont organisés autour de plusieurs piliers clés:

- La gestion des identités (IAM) : il gère le contrôle d'accès fin, authentification multi-facteur (MFA).
- Sécurité du réseau : provisionner par différent services qui sont entre autre
  - VPC ( virtual private cloud) : est un réseau privée virtuelle du cloud.
  - NACL et security groups : contrôle du trafic entrant et sortant.
  - Aws WAF (web application firewall): protection des applications contre les attaques communes
  - Aws shield : utiliser pour mitiguer les attaques DDOS.
- Protection des données : le chiffrement de vos données se fait au repôt et en transit et se fait via le service KMS (key management service)
- Détection et réponse aux menaces
  - Cloudwatch : collecte et analyse les journaux pour identifier les anomalies
  - Inspector : effectue et analyse les vulnérabilités des différentes instances ec2.
- Le stockage des données
  - S3 (simple storage service) : permet de stocker et de récupérer n'importe quelle quantité de données.

### **4. La chaîne de valeur de aws**

Une chaîne de valeur en sécurité est une représentation séquentielle et interconnectée des différentes étapes, processus et activités nécessaires pour assurer la sécurité d'un système, d'une application ou d'une organisation. C'est un peu comme une ligne de production, mais au lieu de fabriquer des produits, on fabrique de la sécurité.

Chaque maillon de cette chaîne est crucial et contribue à la protection globale. Si un maillon est faible, toute la chaîne est compromise. Les éléments typiques d'une chaîne de valeur en sécurité sont :

- L'identification et l'authentification : déterminer qui a accès à quoi.
- Autorisation : définir les actions que les utilisateurs peuvent effectuer.
- Protection des données : chiffrement, contrôle d'accès ; sauvegarde.
- Détection des menaces : surveillance des activités, analyse des logs.
- Réponses aux incidents : planification et exécution des actions en cas d'incident.
- Récupération : récupération des données après sinistre

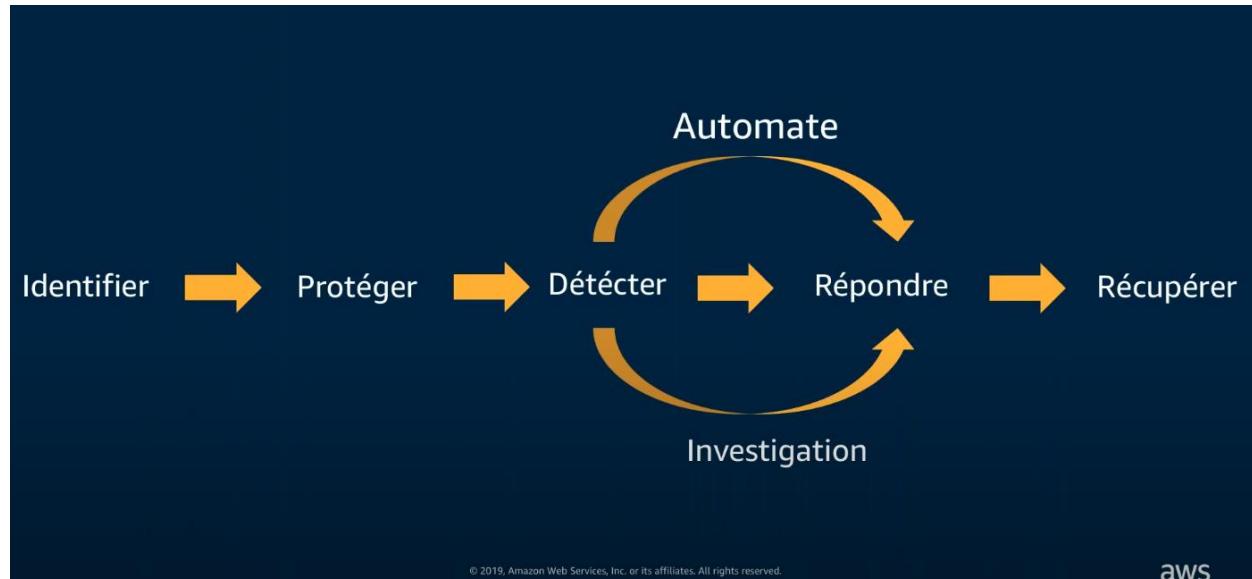


Figure 4: chaîne de valeur de la sécurité de aws

## I. COMPARAISON DES DIFFERENTES SOLUTIONS

### A. SECURITE TRADITIONNELLE VS SECURITE AWS : UNE PRESENTATION COMPARATIVE

#### 1. Sécurité Traditionnelle : Concepts et Architecture

Les solutions de sécurité traditionnelles sont déployées principalement sur site, au sein des infrastructures physiques des entreprises. Elles consistent en une combinaison d'outils matériels et logiciels qui protègent les réseaux, les serveurs, et les données contre diverses menaces. Voici un aperçu de leurs principaux composants:

- **Pare-feu Matériel et Logiciel** : Les pare-feu sont utilisés pour filtrer le trafic réseau en fonction de règles prédéfinies, protégeant ainsi l'infrastructure contre les accès non autorisés.
- **Systèmes de Détection et de Prévention d'Intrusion (IDS/IPS)** : Ces systèmes surveillent le trafic réseau et les activités système pour détecter et prévenir les tentatives d'intrusion.
- **Antivirus et Antimalware** : Ces logiciels sont déployés sur les postes de travail et les serveurs pour détecter, mettre en quarantaine, et éliminer les logiciels malveillants.
- **Gestion des Correctifs** : La sécurité traditionnelle inclut des processus de gestion des correctifs pour garantir que les systèmes sont à jour et protégés contre les vulnérabilités connues.
- **Contrôle d'Accès Physique** : Les centres de données et autres installations critiques sont protégés par des mesures de contrôle d'accès physique, comme les badges, les biométries, et la vidéosurveillance.
- **Surveillance et Journaux** : Les entreprises mettent en place des systèmes de surveillance et d'enregistrement des journaux pour suivre les activités et détecter les anomalies.
- **Chiffrement** : Le chiffrement des données est généralement appliqué à certains niveaux, comme le stockage et les transmissions critiques.

## 2. Sécurité AWS : Concepts et Architecture

AWS propose une approche de la sécurité basée sur le modèle de sécurité partagée, où AWS gère la sécurité de l'infrastructure cloud, tandis que les clients sont responsables de la sécurité de leurs propres données et configurations. Voici un aperçu de la sécurité AWS:

- **Gestion des Identités et des Accès (IAM)** : AWS IAM permet de contrôler l'accès aux ressources AWS en utilisant des rôles, des groupes et des politiques détaillées, garantissant que seuls les utilisateurs autorisés peuvent accéder aux ressources spécifiques.
- **Chiffrement des Données (KMS)** : AWS propose des services de chiffrement, notamment AWS KMS, qui permet de gérer et protéger les clés de chiffrement utilisées pour les données stockées et en transit.
- **Surveillance et Audit (CloudTrail, CloudWatch)** : AWS fournit des services comme CloudTrail pour l'audit des activités, et CloudWatch pour la surveillance des ressources et des applications en temps réel.
- **Détection des Menaces (GuardDuty, Macie)** : AWS propose des services pour la détection des menaces et la protection des données sensibles, notamment GuardDuty pour la détection des menaces réseau, et Macie pour la protection des données sensibles.
- **Protection contre les Attaques (AWS Shield)** : AWS Shield protège contre les attaques DDoS, offrant une protection standard et avancée pour les applications web critiques.
- **Conformité et Gouvernance (AWS Config, Security Hub)** : AWS propose des outils pour aider à gérer la conformité aux réglementations et à maintenir une gouvernance stricte, avec AWS Config pour le suivi des configurations et Security Hub pour une vue centralisée de la sécurité.

- **Automatisation** : AWS permet l'automatisation des tâches de sécurité telles que les audits, la gestion des configurations et les réponses aux incidents, réduisant ainsi les erreurs humaines.

Les solutions de sécurité traditionnelles offrent un contrôle direct sur l'infrastructure, mais exigent des investissements considérables en matériel et en maintenance. En revanche, AWS propose une sécurité moderne et intégrée dans le cloud, avec des outils automatisés, une grande flexibilité, et une gestion simplifiée, tout en respectant des standards élevés de sécurité et de conformité. Choisir entre une approche traditionnelle et AWS dépend des besoins spécifiques de l'organisation, de ses ressources et de son niveau de maturité en matière de sécurité.

## B. ARCHITECTURE SERVERLESS OU ARCHITECTURE TRADITIONNELLE: UNE PRESENTATION COMPARATIVE

### 1. Architecture Serverless

Une architecture serverless est un modèle de déploiement dans lequel les développeurs créent et déplacent des applications sans avoir à gérer les serveurs physiques ou virtuels sous-jacents. Dans ce modèle, la gestion de l'infrastructure, y compris la mise à l'échelle, le provisioning et la gestion des serveurs, est entièrement prise en charge par le fournisseur de services cloud.

Les principales caractéristiques de cette architecture sont :

- **Gestion Automatisée des Serveurs** : Les serveurs sont entièrement gérés par le fournisseur de cloud. Les développeurs n'ont pas besoin de provisionner, configurer ou gérer les serveurs.
- **Évolutivité Dynamique** : Les services serverless s'adaptent automatiquement à la charge. Par exemple, AWS Lambda peut gérer une augmentation soudaine du trafic sans nécessiter d'intervention manuelle.
- **Facturation à l'Usage** : Vous payez uniquement pour le temps d'exécution de votre code et les ressources utilisées. Cela peut entraîner des économies par rapport aux modèles de tarification basés sur des serveurs fixes.
- **Développement Simplifié** : Les développeurs peuvent se concentrer sur le code et la logique de l'application, sans se soucier de la gestion de l'infrastructure.
- **Déploiement Rapide** : Les mises à jour et les déploiements de code peuvent être réalisés rapidement, car les développeurs n'ont pas à gérer les serveurs ou les configurations complexes de l'infrastructure.
- **Gestion Automatique des Versions** : Les plateformes Serverless offrent souvent une gestion automatique des versions, permettant de déployer des versions multiples de la même fonction et de revenir à une version antérieure si nécessaire.

Les avantages d'une telle architecture résident dans :

L'architecture serverless présente plusieurs avantages significatifs qui la rendent attrayante pour de nombreuses entreprises et développeurs. Voici les principaux avantages:

### Avantages de l'Architecture

- **Réduction des Coûts :**

**Facturation à l'Usage** : Vous payez uniquement pour le temps d'exécution du code et les ressources réellement utilisées. Cela peut réduire les coûts par rapport aux modèles traditionnels où vous payez pour des serveurs ou des instances dédiées, qu'ils soient utilisés ou non.

- **Scalabilité Automatique** : L'architecture serverless s'adapte automatiquement à la demande. Les fonctions sont créées et mises à l'échelle en fonction du nombre de requêtes ou d'événements, sans intervention manuelle.
- **Déploiement Rapide et Simplifié** : Les développeurs peuvent déployer des fonctions rapidement, sans se soucier de la gestion des serveurs. Les mises à jour et les déploiements sont plus rapides et moins complexes.
- **Moins de Gestion d'Infrastructure** : Les développeurs n'ont pas besoin de gérer l'infrastructure sous-jacente, ce qui permet de se concentrer sur le code et les fonctionnalités de l'application.
- **Flexibilité et Agilité** : L'architecture serverless facilite le développement d'applications basées sur des microservices, permettant une évolution indépendante des différentes parties de l'application.
- **Haute Disponibilité et Fiabilité** : Les services serverless sont souvent déployés sur une infrastructure distribuée qui offre une haute disponibilité et une résilience face aux pannes.
- **Gestion Automatique des Versions** : La gestion des versions est souvent intégrée, facilitant le déploiement de nouvelles versions de code et le retour à des versions antérieures si nécessaire.
- **Évolutivité Fine et Granulaire** : Chaque fonction ou service peut évoluer indépendamment des autres, ce qui permet une gestion plus précise des ressources en fonction des besoins spécifiques.
- **Sécurité Renforcée** : Chaque fonction s'exécute dans un environnement isolé, ce qui améliore la sécurité en réduisant les interactions non autorisées entre les différentes parties de l'application.
- **Support Intégré pour les Événements** : Les fonctions serverless peuvent être déclenchées par une variété d'événements, comme des changements dans des bases de données, des messages dans des files d'attente ou des requêtes HTTP.
- **Facilité d'Intégration** : Les plateformes serverless offrent des intégrations faciles avec d'autres services cloud, facilitant la création de flux de travail complexes et d'architectures composées.
- **Débogage et Surveillance** :

**Outils Intégrés** : Les fournisseurs de cloud proposent souvent des outils intégrés pour la surveillance des performances, le débogage et l'analyse des journaux, ce qui simplifie la gestion et l'optimisation des fonctions.

## 2. Architecture traditionnelle (Basée sur des serveurs)

Elle repose sur des serveurs physiques ou virtuels que vous devez provisionner, configurer et gérer pour exécuter vos applications. Vous avez un contrôle complet sur l'infrastructure.

Les caractéristiques de cette architecture sont :

- **Contrôle Complet** : Vous avez un contrôle total sur les serveurs, y compris leur configuration, leur mise à jour et leur maintenance.
- **Évolutivité Manuelle** : La mise à l'échelle nécessite une gestion manuelle. Vous devez ajouter ou supprimer des serveurs selon les besoins de l'application.
- **Coût Fixe** : Vous payez pour la capacité des serveurs, qu'ils soient utilisés ou non. Cela peut entraîner des coûts fixes plus élevés, même si les ressources ne sont pas pleinement exploitées.
- **Développement et Déploiement** : Le déploiement et la gestion des applications nécessitent une gestion de l'infrastructure, ce qui peut augmenter la complexité.
- **Exemples de Solutions** : Machines virtuelles (VM) sur AWS EC2, serveurs physiques ou autres environnements de cloud IaaS.

## Avantages de l'architecture traditionnelle

- **Gestion Fine** : Vous avez un contrôle total sur les serveurs, y compris leur configuration, leur sécurité et leur gestion. Cela permet des ajustements précis en fonction des besoins spécifiques de l'application.
- **Personnalisation** : Possibilité de personnaliser les configurations du serveur, le système d'exploitation, les logiciels et les paramètres en fonction des exigences spécifiques.
- **Exécution Continue** : Contrairement à certaines solutions serverless qui ont des limites de temps d'exécution, les serveurs traditionnels permettent une exécution continue sans interruptions imposées.
- **Prévisibilité des Coûts** : Les coûts peuvent être plus prévisibles et stables, surtout si vous utilisez des serveurs physiques ou des instances réservées, car vous payez généralement pour la capacité provisionnée, indépendamment de l'utilisation.
- **Pas de Dépendance aux Limitations des Plateformes** : Vous évitez les limitations imposées par les fournisseurs de services cloud en termes de fonctionnalités, de quotas ou de contraintes d'exécution.
- **Performance Consistante** : Les ressources sont dédiées et ne sont pas partagées avec d'autres utilisateurs, ce qui peut offrir des performances plus prévisibles et constantes.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

- Gestion des Applications Longue Durée :** L'architecture traditionnelle est souvent plus adaptée aux applications monolithiques ou aux systèmes hérités qui nécessitent une gestion et une configuration précises de l'infrastructure.
- Flexibilité pour des Environnements Complexes :** Permet des configurations complexes et des intégrations avec des systèmes spécifiques ou des matériels dédiés.
- Meilleur Contrôle des Données Sensibles :** Pour les serveurs sur site, vous avez un contrôle direct sur la sécurité physique des serveurs et des données, ce qui peut être crucial pour les applications traitant des informations sensibles.
- Développement et tests :** Vous pouvez configurer des environnements de développement et de test qui reflètent précisément les environnements de production, ce qui facilite le débogage et la validation des applications

### Comparaison : Architecture Serverless vs Architecture Traditionnelle

critère	Architecture Serverless	Architecture Traditionnelle
<b>Gestion des serveurs</b>	✓ (Gérée par le cloud)	X (Gérée par l'utilisateur)
<b>Scalabilité</b>	✓ (Automatique et dynamique)	X (Manuelle, nécessite provisionnement)
<b>cout</b>	✓ (Facturation à l'usage)	X (Coût fixe pour la capacité provisionnée)
<b>Déploiement rapide</b>	✓ (Rapide et simplifié)	X (Peut être complexe et long)
<b>performances</b>	✓ (Ressources scalables)	✓ (Ressources dédiées)
<b>Intégration avec d'autres services</b>	✓ (Facile)	X (Peut nécessiter plus de configuration)
<b>Contrôle de l'infrastructure</b>	X (Moins de contrôle)	✓ (Contrôle complet)
<b>Evitements des restrictions</b>	X (Dépendance au fournisseur)	✓ (Moins de dépendance)
<b>Gestion des versions</b>	✓ (Versionnement intégré)	X (Gestion manuelle)

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

De ces deux modèles d'architecture le choix de la solution sera évidemment porté sur l'architecture serverless qui obtiens une meilleure note (7/9) sur toutes les caractéristiques comparées.

Caractéristique de sécurité	Architecture Serverless	Architecture Traditionnelle
<b>Surveillance et journaux</b>	✓ (Souvent centralisée et automatisée par le fournisseur de cloud. Des outils de logging avancés sont disponibles.)	✗ (Nécessite une mise en place complexe de systèmes de surveillance et de gestion des journaux.)
<b>Gestion des correctifs</b>	✓ (Automatisée par le fournisseur de cloud. Les mises à jour sont déployées rapidement et en toute sécurité)	✗ (Requiert une planification et une exécution manuelle des mises à jour, ce qui peut être chronophage et source d'erreurs.)
<b>Antivirus et antimalware</b>	✓ (Intégrés aux services cloud, mis à jour régulièrement.)	✗ (Nécessite l'installation et la maintenance de logiciels antivirus sur chaque serveur.)
<b>IDS et IPS</b>	✓ (Souvent inclus dans les offres de sécurité des fournisseurs de cloud)	✗ (Nécessite l'achat et la configuration d'équipements spécialisés)
<b>Chiffrement</b>	✓ ( Options de chiffrement au repos et en transit proposées par les fournisseurs de cloud)	✓ (Nécessite l'achat et la configuration d'équipements spécialisés)
<b>Audit</b>	✓ (Outils d'audit intégrés pour suivre les activités et les configurations)	✗ (Nécessite la mise en place de solutions de chiffrement spécifiques)
<b>Protection contre les menaces</b>	✗ (Protection contre un large éventail de menaces, y compris les DDoS, les injections SQL, etc)	✓ (Nécessite des outils d'audit spécifiques et des procédures manuelles)
<b>Automatisation</b>	✗ (Dépendance au fournisseur)	✓ (Protection moins complète, nécessitant des

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

		mises à jour régulières des logiciels de sécurité)
Gestion des versions	✓ (Fortement automatisée, ce qui réduit les erreurs humaines et accélère les processus.)	✗ (Moins automatisée, requiert une intervention humaine pour de nombreuses tâches.)

Les solutions de sécurité traditionnelles offrent un contrôle direct sur l'infrastructure, mais exigent des investissements considérables en matériel et en maintenance. En revanche, AWS propose une sécurité moderne et intégrée dans le cloud, avec des outils automatisés, une grande flexibilité, et une gestion simplifiée, tout en respectant des standards élevés de sécurité et de conformité. Choisir entre une approche traditionnelle et AWS dépend des besoins spécifiques de l'organisation, de ses ressources et de son niveau de maturité en matière de sécurité.

### III. CHOIX DE LA SOLUTION

Amazon Web Services (AWS) est une plateforme de services cloud fournie par Amazon, qui offre une large gamme de services informatiques basés sur le cloud. Son lancement effectif en 2006 et qui plus est parmi les premiers à proposer des services infrastructure cloud à grande échelle. Depuis son lancement, AWS a connu une croissance rapide et détient aujourd'hui près de 32% du marché mondial en terme de services cloud et 6% sur les hébergements mondiaux, ce qui fait de lui le leader du secteur.

Cependant, l'avènement du cloud computing a transformé le paysage de la sécurité. Des plateformes comme Amazon Web Services (AWS) offrent aujourd'hui des solutions de sécurité intégrées, évolutives, et adaptées aux besoins des entreprises de toutes tailles. Ces solutions permettent de simplifier la gestion, d'automatiser les tâches de surveillance, et d'optimiser les coûts tout en offrant un niveau de protection équivalent, supérieur, à celui des infrastructures traditionnelles. Maintenant il s'agit d'une présentation détaillée de la solution choisie.

#### A. Présentation de AWS Security

Amazon Web Services (AWS) offre une plateforme cloud complète qui intègre des services de sécurité sophistiqués pour protéger les données et les infrastructures des entreprises. En adoptant une approche basée sur le modèle de sécurité partagée, AWS fournit les outils nécessaires pour sécuriser les environnements cloud tout en permettant aux clients de personnaliser et de gérer leur propre sécurité.

## 1. Avantages de la Sécurité AWS

- **Scalabilité et Flexibilité** : Les solutions de sécurité AWS s'adaptent facilement à la croissance des entreprises et aux variations de la charge de travail, offrant une protection évolutive sans nécessiter d'investissements matériels lourds.
- **Automatisation et Intégration** : Les outils de sécurité AWS sont intégrés de manière transparente avec les autres services AWS, permettant l'automatisation des tâches de sécurité, la gestion des alertes et des réponses aux incidents.
- **Réduction des Coûts** : Avec AWS, les entreprises peuvent bénéficier d'un modèle de tarification à l'utilisation, ce qui réduit les coûts par rapport aux solutions de sécurité traditionnelles nécessitant des investissements initiaux importants.
- **Mises à Jour et Maintenance** : AWS gère les mises à jour et la maintenance des services de sécurité, assurant que les dernières protections sont en place sans nécessiter d'intervention manuelle.
- **Conformité** : AWS propose des certifications de conformité pour une variété de normes et de réglementations (PCI DSS, HIPAA, ISO 27001, etc.), facilitant le respect des exigences légales et réglementaires.

## 2. Inconvénients de la Sécurité AWS

- **Complexité de la Configuration** : La configuration des services de sécurité AWS peut être complexe, surtout pour les nouvelles organisations, nécessitant une expertise technique pour une mise en œuvre et une gestion appropriées.
- **Responsabilité Partagée** : Le modèle de sécurité partagée signifie que les clients doivent gérer certains aspects de la sécurité (comme les configurations des services et les accès aux données), ce qui peut nécessiter des efforts supplémentaires pour assurer une sécurité complète.
- **Dépendance à un Fournisseur** : Utiliser les solutions de sécurité d'AWS implique une dépendance au fournisseur, ce qui peut poser des défis en cas de besoin de portabilité ou de changement de fournisseur.
- **Risques de Configurations Incorrectes** : Une mauvaise configuration des services AWS peut exposer des vulnérabilités, ce qui nécessite une vigilance continue et des bonnes pratiques en matière de gestion de la sécurité.
- **Coût des Services Avancés** : Bien que les coûts initiaux puissent être réduits, certains services de sécurité avancés (comme AWS Shield Advanced) peuvent entraîner des frais supplémentaires, surtout pour des protections renforcées.

## C. ETUDE DES DIFFERENTS SERVICES

### 1. AWS VPC (virtual private cloud)

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

Un VPC aws un réseau privée dédié à votre compte aws, permettant aux différents utilisateurs de déployer leurs services dans un réseau isolé. Ce VPC est divisé en plusieurs sous réseau appelé subnet, ces subnets sont créer dans le but d'organiser les ressources. Il en existe deux types les public (ils ont accès à internet), private (qui n'ont pas accès à internet). Chaque subnet est associer à une zone de disponibilité appelé availability zone (est une zone géographique distincte au sein d'une région (AWS fonctionne par région. Il s'agit d'un emplacement physique dans le monde où nous regroupons des centres de données. Nous appelons chaque groupe de centres de données logiques une « Zone de disponibilité ») du cloud.) et ne peut s'étendre sur plusieurs zones. Il existe au total 69 zones de disponibilités dans 22 régions géographiques dont parmi elle on peut citer celle située en Afrique plus précisément au cap.

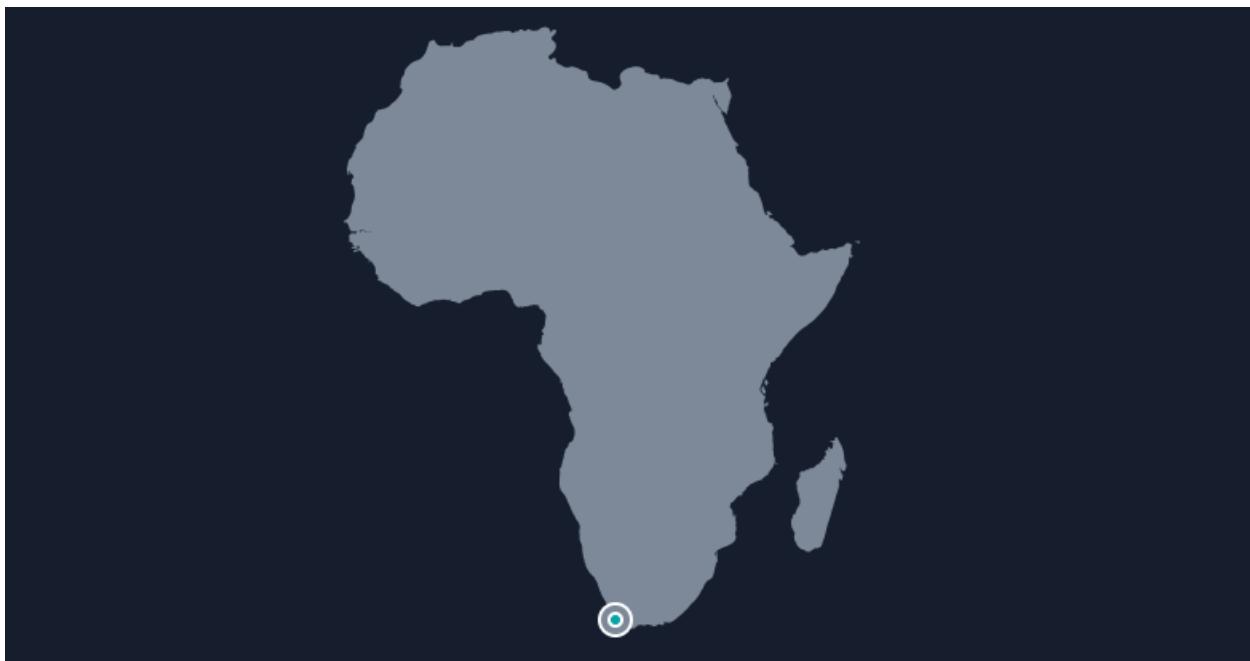


Figure 5: région Afrique (Cap) Zones de disponibilités: 3

Pour garantir les mesures de sécurité stricte de ce réseau plusieurs mécanismes sont mis en place. Tout part de l'entrée du trafic dans le VPC, ensuite de son transport enfin de sa redirection vers les ressources appropriées. Nous provisionnerons la sécurité de ce VPC avec les NACL (network access control lists), les WAF (web application firewall) et un serveur front.

### a. NACL

Un NACL (network access control list) est un mécanisme de sécurité qui filtre le trafic entrant et sortant au niveau des sous-réseaux dans un VPC. Elles se trouvent sur des sous-réseaux et évaluent le trafic en fonction de règles définies, puis déterminent si ce trafic doit être autorisé ou non à continuer. Cela applique un contrôle d'accès basé sur le trafic à votre réseau et constitue un moyen efficace de sécuriser votre environnement.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Les listes NACL sont « sans état » ce qui signifie que les informations sur le trafic précédemment envoyé ou reçu ne sont pas enregistrées et nécessitent la création de règles distinctes pour le trafic entrant et sortant. Ce n'est pas parce qu'un flux de données particulier est autorisé à entrer qu'il sera automatiquement autorisé à sortir. Ils sont traités dans l'ordre numérique. Par conséquent, si vous avez besoin que le trafic entre et sorte d'un sous-réseau protégé, vous devrez écrire des règles pour les deux sens.

Les règles sont appliquées par ordre de numéro, le plus bas étant évalué en premier. Il supporte à la fois les adresses ipv4 et ipv6. Chaque NACL ne peut contenir que 20 règles par direction par défaut mais cela peut être augmenter à 40.

La configuration des NACL dépendent de :

- Le numéro de la règle : chaque règle doit avoir un numéro unique (1-32766). Les règles sont évaluées par ordre croissant.
- Type de trafic : entrant et sortant
- Protocole : choix du protocole approprié, TCP, UDP, ICMP, ...en fonction des besoins de votre application ou service
- Les ports
- Les adresses IP sources/Destinations
- Action : allow et deny

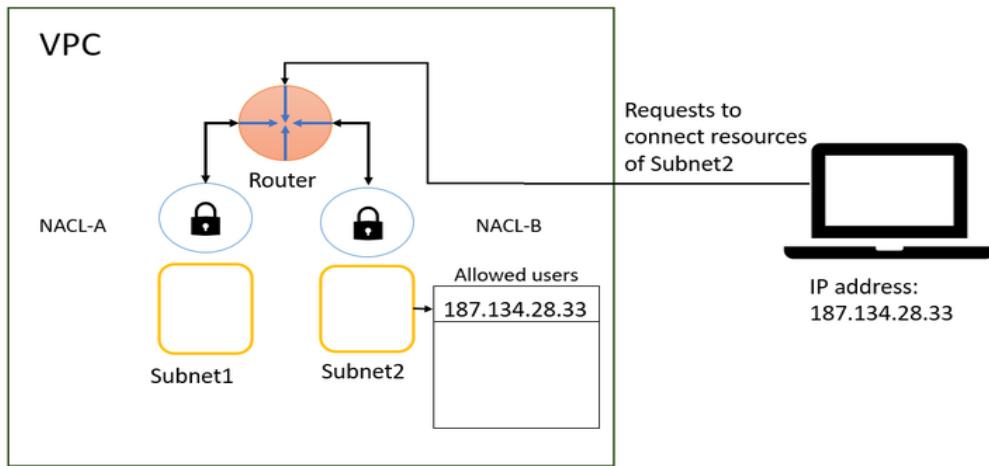


Figure 6: fonctionnement d'une NACL

### b. WAF

Un WAF ou pare-feu applicatif web aide à protéger les applications web en filtrant et en surveillant le trafic http et https entre une application Web et Internet. Il protège généralement les applications Web des attaques notamment de type falsification de site croisé, cross-site scripting (XSS) (consistent à insérer un code malveillant dans des sites Web par ailleurs fiables.) , d'inclusion de

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

fichier et d'injection SQL. En déployant un WAF devant une application web, un bouclier est placé entre l'application web et Internet.

Un serveur proxy protège l'identité d'une machine client en utilisant un intermédiaire ; un WAF est un type de proxy inversé qui protège le serveur en faisant passer les clients par le WAF avant d'atteindre le serveur.

Un WAF (Web Application Firewall) opère grâce à un ensemble de règles, communément appelées politiques. Ces politiques sont conçues pour protéger l'application en bloquant le trafic malveillant et en réduisant les vulnérabilités. L'efficacité d'un WAF réside en partie dans sa capacité à permettre des modifications rapides et faciles de ces politiques, offrant ainsi une réponse rapide aux divers vecteurs d'attaque. Par exemple, lors d'une attaque DDoS, le mécanisme de limitation du taux peut être activé rapidement en ajustant les politiques du WAF.

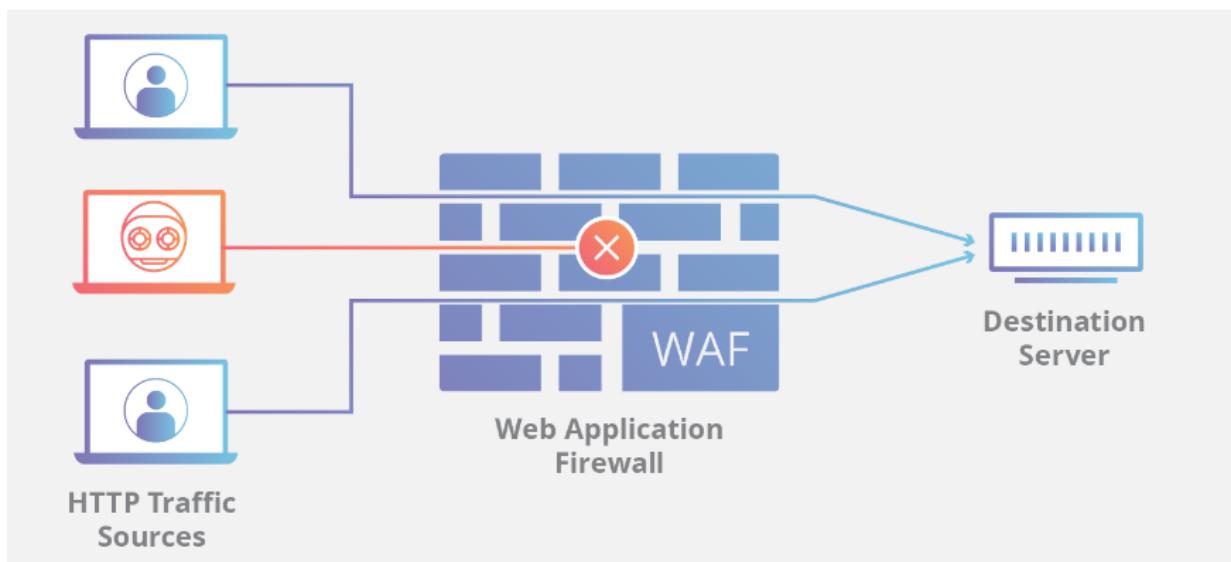


Figure 7: fonctionnement d'un WAF

### c. serveur front

Un serveur front est un serveur qui se trouve en première ligne dans une architecture réseau. Il est responsable de gérer les requêtes initiales des utilisateurs avant de les rediriger vers d'autres serveurs pour traitement. Les différents types de serveurs frontaux sont :

- ❖ Les load balancer
- ❖ Les reverses proxy
- ❖ Les CDN (réseau de diffusion de contenu)

Le serveur frontal endosse plusieurs rôles parmi lesquels on peut citer :

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

- Répartition de la charge : Il répartit les requêtes entre plusieurs serveurs backend, afin d'optimiser les performances et d'éviter la surcharge d'un seul serveur.
- Sécurité : Il peut filtrer les requêtes malveillantes, protéger les serveurs backend contre les attaques et assurer l'authentification des utilisateurs.
- Cache : Il peut stocker en mémoire cache les données fréquemment demandées, réduisant ainsi le temps de réponse et la charge sur les serveurs backend.
- Compression : Il peut compresser les données avant de les envoyer aux clients, réduisant ainsi le temps de transfert.
- Adaptation : Il peut adapter le contenu en fonction du type de client (ordinateur, smartphone, tablette), de sa langue ou de ses préférences.

Comment fonctionne un serveur frontal (une analogie)

Imaginez un serveur frontal comme un réceptionniste dans un grand bâtiment. Lorsque vous arrivez, le réceptionniste (le serveur frontal) vous demande ce que vous cherchez et vous dirige vers la personne ou le service approprié (le serveur backend). Il peut aussi vous donner des informations supplémentaires ou vérifier votre identité.

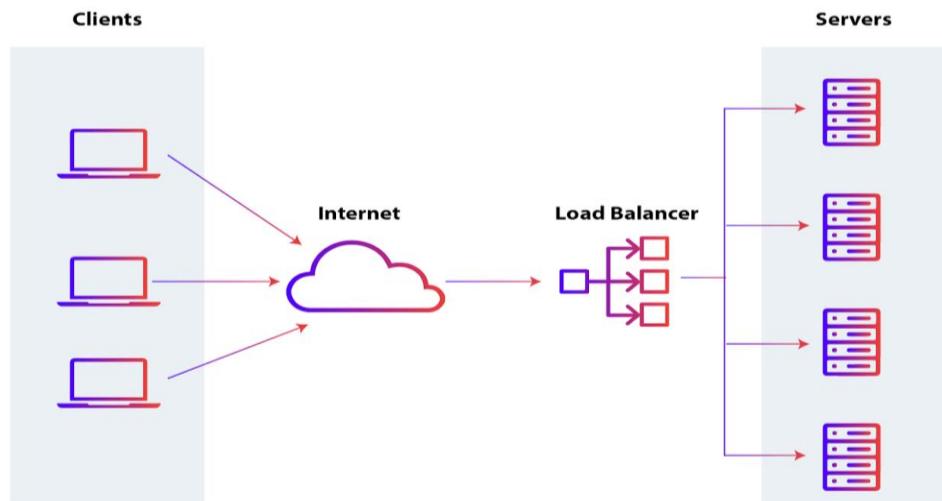


Figure 8: fonctionnement d'un load balancer

### d. ACM

AWS Certificate Manager (ACM) est un service entièrement géré qui simplifie la mise en service, la gestion et le déploiement des certificats SSL/TLS pour vos applications sur AWS. ACM simplifie la gestion des certificats en automatisant le processus de renouvellement, de validation et de déploiement des certificats, ce qui vous permet de vous concentrer sur la création et la mise à l'échelle de vos applications.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Les certificats SSL/TLS permettent aux navigateurs Web d'identifier et d'établir des connexions réseau chiffrées vers des sites Web à l'aide du protocole Secure Sockets Layer/Transport Layer Security (SSL/TLS). Les certificats sont utilisés dans un système cryptographique appelé « infrastructure à clé publique » (PKI, Public Key Infrastructure). L'infrastructure PKI permet à une partie d'établir l'identité d'une autre partie à l'aide de certificats si elles font toutes deux confiance à un tiers appelé « autorité de certification ».

Les certificats tant publics que privés aident les clients à identifier les ressources sur les réseaux et à sécuriser les communications entre ces ressources. Les certificats publics permettent d'identifier les ressources sur l'Internet public, alors que les certificats privés font cela pour les réseaux privés.

Principales fonctionnalités d'AWS ACM :

- **Renouvellement automatique des certificats** : ACM automatise le processus de renouvellement, garantissant que vos certificats sont toujours à jour et éliminant le risque de perturbations liées à l'expiration. Cette fonctionnalité est particulièrement avantageuse pour les organisations gérant un grand nombre de certificats.
- **Intégration aux services AWS** : ACM s'intègre parfaitement à d'autres services AWS, tels qu'Elastic Load Balancer (ELB), CloudFront et API Gateway. Cette intégration simplifie le processus d'association des certificats à ces services, réduisant ainsi le temps et les efforts nécessaires au déploiement.

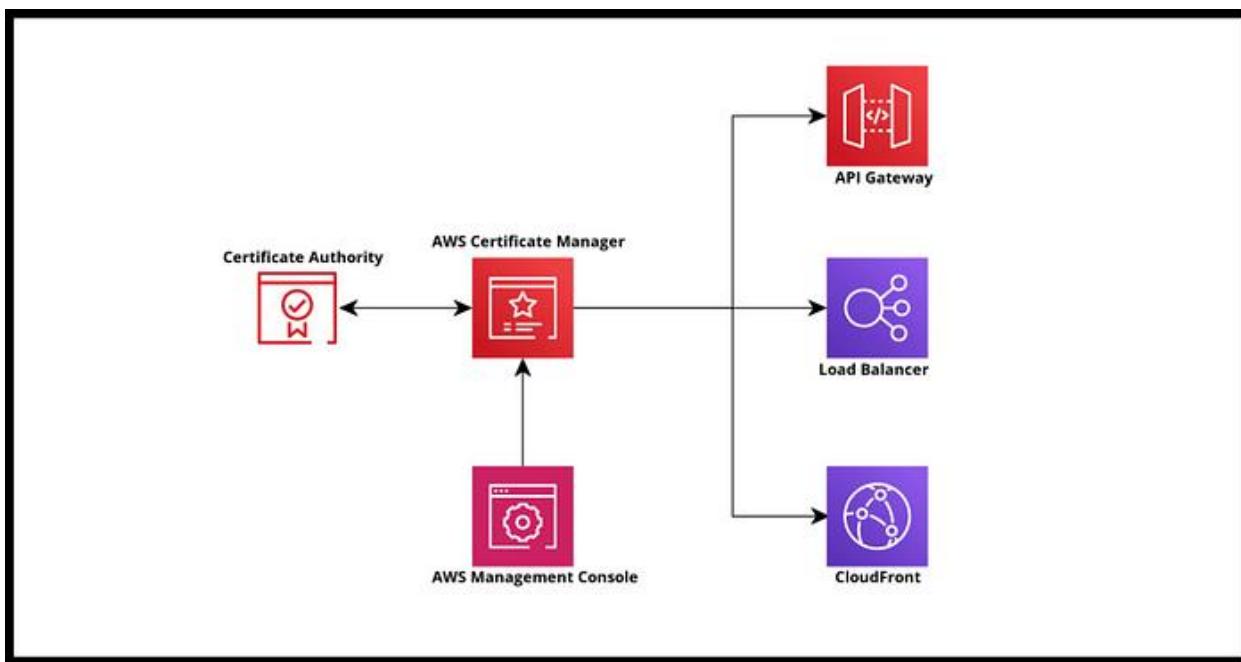


Figure 9: intégration de acm avec d'autres services aws

- **Couverture mondiale** : ACM prend en charge les déploiements mondiaux avec des certificats pouvant être utilisés dans plusieurs régions AWS. Cela est particulièrement utile pour les

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

entreprises ayant une présence mondiale, car cela garantit une sécurité cohérente sur différents sites géographiques.

- **Validation des certificats :** ACM gère automatiquement la validation de la propriété du domaine. Cela simplifie le processus d'obtention des certificats, évitant ainsi aux utilisateurs les tracas liés aux étapes de validation manuelle.

## 2. AWS EC2

Amazon Elastic Compute Cloud (EC2) est un service fondamental d'Amazon Web Services (AWS) qui permet aux utilisateurs de lancer et de gérer des serveurs virtuels dans le cloud. Que vous soyez développeur, professionnel de l'informatique ou que vous débutiez dans le cloud computing, AWS EC2 fournit une infrastructure évolutive et flexible pour répondre à vos besoins informatiques.

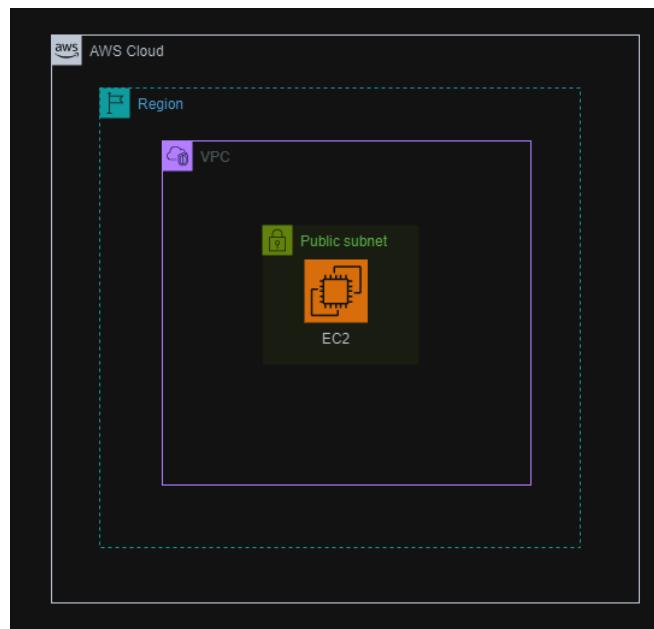


Figure 10: Disposition de EC2 dans aws

AWS EC2 est un service Web qui fournit une capacité de calcul redimensionnable dans le cloud. Il permet aux utilisateurs d'exécuter des serveurs virtuels, appelés instances, à la demande. Avec EC2, vous pouvez rapidement augmenter ou réduire vos ressources de calcul en fonction des besoins de votre application. Cette flexibilité en fait une solution idéale pour l'hébergement d'applications, l'exécution d'environnements de développement et la gestion de charges de travail de toutes tailles.

Avant d'aborder en profondeur le concept EC2, il serait préférable de définir les concepts clés :

## **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---

- Instances : Une instance est un serveur virtuel dans le cloud AWS. Elle représente la puissance de calcul que vous utilisez pour exécuter vos applications.
- Images de machine Amazon (AMI) : Les AMI sont des modèles préconfigurés qui contiennent les informations nécessaires pour lancer une instance. Vous pouvez choisir parmi une variété d'AMI fournies par AWS ou créer les vôtres.
- Types d'instances : EC2 propose une gamme de types d'instances optimisés pour différents cas d'utilisation, tels que les instances optimisées pour le calcul, la mémoire et le stockage.
- Régions et zones de disponibilités : L'infrastructure AWS est répartie sur différentes régions géographiques et zones de disponibilité. Les régions sont des zones géographiques distinctes, tandis que les zones de disponibilité sont des centres de données au sein de ces régions.

### **a. Security group**

Un groupe de sécurité AWS est un pare-feu virtuel associé à vos instances pour contrôler le trafic entrant et sortant. Ils sont appliqués au niveau des ressources, comme une instance EC2, et non au niveau du sous-réseau.

Les groupes de sécurité sont par nature avec état c'est-à-dire que les règles sont appliqués dans les deux sens. Par conséquent, toute modification applicable à une règle entrante sera également automatiquement appliquée à la règle sortante de la même manière. Par exemple, l'autorisation d'un port entrant 80 ouvrira automatiquement le port sortant 80, sans que vous ayez à diriger explicitement le trafic dans la direction opposée.

Le diagramme suivant montre un VPC avec un sous-réseau, une passerelle Internet et un groupe de sécurité. Le sous-réseau contient une instance EC2. Le groupe de sécurité est attribué à l'instance. Le groupe de sécurité agit comme un pare-feu virtuel. Le seul trafic qui atteint l'instance est le trafic autorisé par les règles du groupe de sécurité. Par exemple, si le groupe de sécurité contient une règle qui autorise le trafic ICMP vers l'instance à partir de votre réseau, vous pouvez envoyer une requête ping à l'instance à partir de votre ordinateur. Si le groupe de sécurité ne contient pas de règle qui autorise le trafic SSH, vous ne pouvez pas vous connecter à votre instance à l'aide de SSH.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

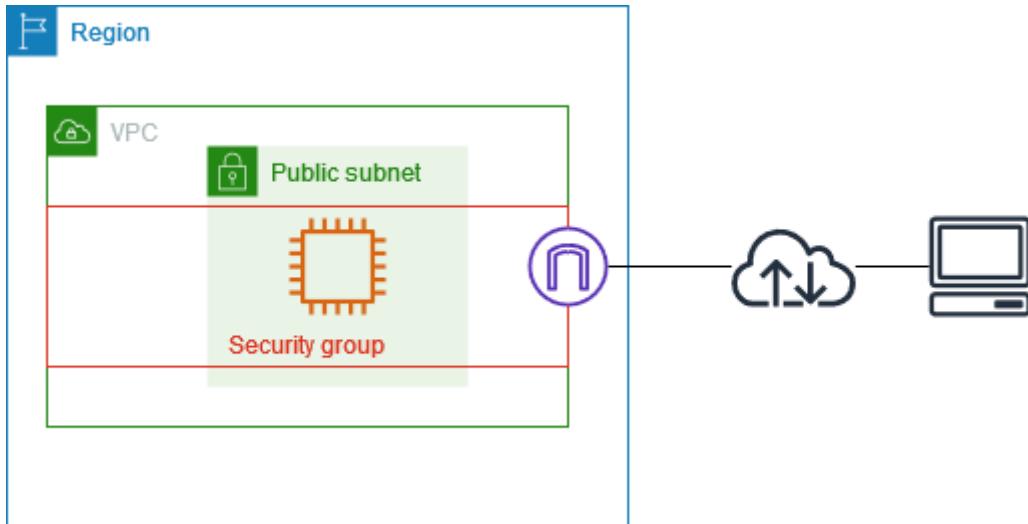


Figure 11: Security group

Les NACL et les groupes de sécurité (SG) ont des objectifs très similaires. Ils filtrent le trafic en fonction de règles, pour garantir que seul le trafic autorisé est acheminé vers sa destination. Nous allons ici souligner les différences entre les deux. Vous trouverez ci-dessous un graphique de haut niveau qui montre leur utilisation et compare les deux technologies.

Tableau 8: tableau comparatif des NACL et Security group

	NACL	Security group
<b>But</b>	Protéger les sous-réseaux	Protéger les instances
<b>Stateful ou stateless</b>	Sans état : doit spécifier à la fois l'entrée et la sortie	Avec état : le trafic de retour est toujours autorisé, quelle que soit la règle
<b>Niveau de fonctionnement</b>	Règles de processus par ordre numérique	Évalue toutes les règles
<b>Application</b>	S'applique automatiquement	L'application doit être spécifiée

### b. Protection des connexions via ec2

Il existe quatre méthodes pour se connecter aux instances Linux Amazon ec2 :

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

- Secure Shell (SSH)
  - SSH ne dispose pas d'un système intégré de journalisation et d'audit des connexions.
  - AWS déploie une paire de clés unique pour l'authentification via SSH sur chaque instance EC2.
  - Si vous perdez la paire de clés, vous ne pourrez pas la régénérer.
- EC2 Serial Console pour les instances Linux : L'EC2 Serial Console établit une connexion série avec les instances EC2 qui vous permet de résoudre les problèmes de démarrage et de connexion réseau. Limites : Une seule connexion à une console de série active est prise en charge par instance. Il doit y avoir un intervalle d'au moins 30 secondes entre les sessions.
- Gestionnaire de session, une fonctionnalité d'AWS Systems Manager
- Amazon Ec2 intance Connect : simplifie et sécurise les connexions SSH aux instances EC2, en supprimant le besoin de gérer des clés SSH directement et en utilisant les politiques IAM pour gérer l'accès. C'est particulièrement utile pour les utilisateurs occasionnels ou pour les scénarios où la gestion des clés SSH est compliquée ou risquée.

### 3. AWS IAM (identity and access management)

IAM est un service Web qui vous permet de contrôler en toute sécurité l'accès aux ressources AWS. Il vous permet de gérer les utilisateurs, les groupes, les rôles et les autorisations, garantissant ainsi que seules les entités autorisées et authentifiées peuvent accéder à vos ressources. Les caractéristiques principales sont :

- Gestion des utilisateurs : IAM permet la création et la gestion des utilisateurs et des groupes AWS, ainsi que leurs autorisations d'accès.
- Rôles et politiques : vous pouvez définir des rôles pour déléguer des autorisations aux services ou utilisateurs AWS, ainsi que des politiques pour spécifier les actions autorisées ou refusées.
- Authentification multifacteur (MFA) : améliorez la sécurité en exigeant un deuxième facteur d'authentification pour les connexions des utilisateurs.
- Intégration : s'intègre parfaitement à d'autres services AWS pour contrôler l'accès aux ressources.

#### A. Principe du Moindre Privilège (Least Privilege)

Le principe du moindre privilège signifie que les utilisateurs et les services ne doivent se voir accorder que les permissions nécessaires pour accomplir leurs tâches. Cela minimise les risques

## **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---

de sécurité en limitant l'accès aux ressources critiques et en réduisant l'impact potentiel en cas de compromission. La mise en place de ces priviléges passent par des étapes qui sont :

- Elaboration de rôles et groupes : créer des rôles et des groupes en fonction des responsabilités particulières et des missions assignées aux utilisateurs et aux services.
- Elaborer des directives précises : ici il s'agit de rédiger des directives d'IAM qui octroient uniquement des actions précises et en limitant l'accès aux ressources particulières.
- Vérification périodique des autorisations : Effectuez des audits réguliers des permissions pour s'assurer qu'elles restent alignées avec les besoins actuels des utilisateurs et des services.

Il a un avantage particulier de réduire les risques c'est-à-dire les limites et les impacts de sécurité en réduisant l'exposition des ressources et en plus de ça assure une conformité aux exigences réglementaire en minimisant l'exposition des ressources.

### **B. Les Rôles IAM**

Un *rôle* IAM est une IAM identité que vous pouvez créer dans votre compte et qui dispose d'autorisations spécifiques. Un rôle IAM est similaire à un IAM utilisateur, dans la mesure où il s'agit d'une AWS identité dotée de politiques d'autorisation qui déterminent ce que l'identité peut et ne peut pas faire AWS. En revanche, au lieu d'être associé de manière unique à une personne, un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. En outre, un rôle ne dispose pas d'informations d'identification standard à long terme comme un mot de passe ou des clés d'accès associées. Au lieu de cela, lorsque vous adoptez un rôle, il vous fournit des informations d'identification de sécurité temporaires pour votre session de rôle.

Vous pouvez utiliser des rôles pour déléguer l'accès à des utilisateurs, à des applications ou à des services qui n'ont normalement pas accès à vos AWS ressources. Par exemple, vous pouvez autoriser les utilisateurs de votre AWS compte à accéder à des ressources dont ils ne disposent pas habituellement, ou autoriser les utilisateurs d'un compte à Compte AWS accéder aux ressources d'un autre compte. Vous pouvez également autoriser une application mobile à utiliser AWS des ressources, mais ne pas intégrer de AWS clés dans l'application (où elles peuvent être difficiles à mettre à jour et où les utilisateurs peuvent éventuellement les extraire). Parfois, vous souhaitez donner AWS accès à des utilisateurs dont l'identité est déjà définie à l'extérieur AWS, par exemple dans le répertoire de votre entreprise. Ou, vous pouvez accorder l'accès à votre compte à des tiers afin de leur permettre de réaliser un audit de vos ressources.

Pour ces scénarios, vous pouvez déléguer l'accès aux AWS ressources à l'aide d'un *rôle IAM*. Cette section présente les rôles et les différentes façons de les utiliser.

### C. MFA (Multi-Factor Authentication)

AWS Identity and Access Management (IAM) est un outil essentiel pour sécuriser l'accès à vos ressources AWS. Une fonctionnalité clé d'IAM est l'authentification multifacteur (MFA), qui ajoute une sécurité supplémentaire en exigeant plusieurs formes d'identification pour les utilisateurs IAM avant qu'ils puissent accéder aux ressources. Cela réduit considérablement le risque d'accès non autorisé, même si un mot de passe est compromis, car un pirate aurait besoin d'un second facteur, tel qu'un jeton de sécurité ou une authentification biométrique.

Pourquoi la MFA est-elle importante ? Les mots de passe peuvent être vulnérables au piratage ou à l'ingénierie sociale. La MFA offre une couche de protection supplémentaire, rendant plus difficile l'accès non autorisé et protégeant ainsi vos ressources AWS contre les menaces potentielles.

Elle se base sur des principes de :

- Quelque chose que vous avez (possession-based) : Appareil MFA (il peut s'agir d'un smartphone avec une application d'authentification ex : authenticator, authy ; un code envoyé par SMS sur le téléphone mobile de l'utilisateur, un jeton de sécurité matériel qui génère des codes...)
- Quelque chose que vous savez (knowledge-based) : Mot de passe ou PIN
- Quelque chose que vous êtes (Inherence-based) : Empreinte digitale, reconnaissance faciale, reconnaissance vocale, scan de l'iris ou de la rétine.

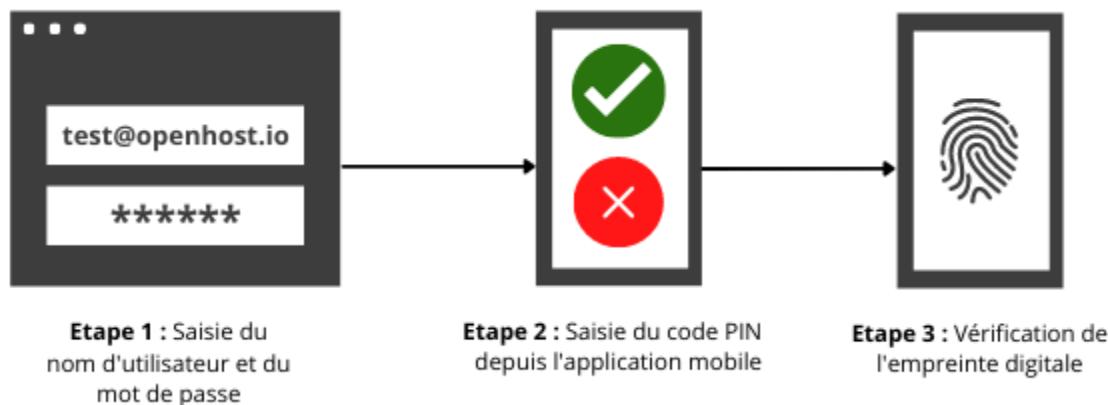


Figure 12: fonctionnement de la MFA

### A. Politiques IAM (IAM Policies)

## **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---

Une politique est un objet AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit leurs autorisations. AWS évalue ces politiques lorsqu'un IAM principal (utilisateur ou rôle) fait une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées AWS sous forme de JSON documents. AWS prend en charge 6 types de politiques :

- La politique basée sur l'identité : est un ensemble de règles qui définissent les permissions qu'une identité spécifique (un utilisateur, un groupe ou un rôle) peut exercer sur les ressources AWS. En d'autres termes, elle détermine ce que cette identité est autorisée à faire ou non dans votre compte AWS.
- La politique basée sur les ressources : Contrairement aux politiques basées sur l'identité qui sont attachées à un utilisateur, un groupe ou un rôle, les politiques basées sur les ressources sont directement associées à une ressource AWS spécifique (comme un bucket S3).
- Les limitations d'autorisations : Une limite d'autorisation (permissions boundary) dans AWS IAM est une fonctionnalité avancée qui vous permet de définir le plafond maximum de permissions qu'une entité IAM (utilisateur, groupe ou rôle) peut détenir, quelle que soit la politique attachée à cette entité.
- Politiques de contrôle des services : Une Politique de Contrôle des Services (SCP) dans AWS Organizations est un type de politique qui s'applique à l'ensemble d'une organisation ou à des unités organisationnelles (OUs).
- Politique basée sur les sessions : Les politiques de session sont un mécanisme puissant dans AWS IAM qui vous permet d'exercer un contrôle encore plus fin sur les permissions accordées à un utilisateur pendant une session spécifique. Ces politiques sont dynamiques et sont attachées à une session plutôt qu'à une identité permanente.
- Contrôle des listes d'accès : c'est un mécanisme plus ancien utilisé dans AWS pour gérer les permissions sur des ressources spécifiques, en particulier dans S3 et certains services réseau. Elle fonctionne en associant directement des permissions à des utilisateurs individuels ou à des groupes d'utilisateurs sur une ressource donnée.

## **4. S3**

Amazon Simple Storage Service (S3) est un service de stockage d'objets évolutif largement utilisé pour stocker et récupérer des données. L'une de ses principales fonctionnalités est la réPLICATION interrégionale (CRR), qui permet la copie automatique et asynchrone d'objets dans différentes régions AWS. Cette fonctionnalité est essentielle pour la reprise après sinistre, la redondance des données et l'amélioration de la latence d'accès aux données dans différentes zones géographiques.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

Amazon S3 propose de stocker les données dans des buckets (Un bucket S3 est essentiellement un conteneur utilisé pour stocker des objets, qui peuvent être des fichiers de n'importe quel type : images, vidéos, documents, etc.). A l'intérieur de chaque bucket vous pouvez déposer des fichiers (on parle d'objet) et y associer des métadonnées. Vous pouvez indiquer ce que vous voulez dans ces métadonnées (l'auteur du fichier par exemple)

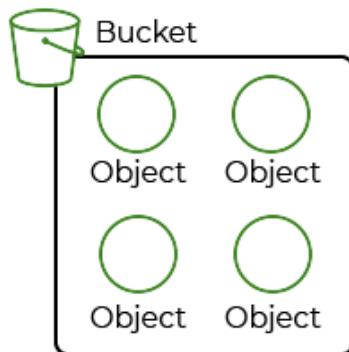


Figure 13:Bucket S3

Les utilisateurs peuvent définir des politiques d'accès pour contrôler qui peut accéder à un bucket ou à des objets spécifiques. Ces politiques permettent de restreindre l'accès en fonction d'identifiants tels que des utilisateurs, des rôles, ou des adresses IP, garantissant la sécurité des données stockées.

S3 offre différentes classes de stockage, adaptées à divers besoins de coût et de performance, telles que S3 Standard pour un accès fréquent, S3 Infrequent Access pour des données consultées moins souvent, et S3 Glacier pour l'archivage à long terme.

Enfin, S3 est conçu pour être très durable, avec une durabilité de 99,99999999%, ce qui signifie que la probabilité de perdre des données est extrêmement faible, même sur de longues périodes.

Voici une petite liste d'avantage que S3 vous apporte :

- Le Versionnement des fichiers : vous pouvez revenir à une version précédente à tout moment.
- Vos fichiers peuvent avoir une date d'expiration et être supprimés automatiquement.
- La réplication automatique des fichiers dans plusieurs Datacenters AWS. Ainsi, vous diminuez le risque de perdre des données importantes.
- Il n'y a pas de limite d'espace.
- Le chiffrement des données.

### A. KMS

AWS Key Management Service (AWS KMS) est un service géré qui vous permet de créer et de contrôler facilement les clés cryptographiques utilisées pour protéger vos données. C'est un élément essentiel de votre stratégie de sécurité dans le cloud, car il vous permet de protéger vos données sensibles, qu'elles soient stockées dans S3, RDS, ou utilisées par d'autres services AWS.

KMS permet de générer des clés de chiffrement symétriques (une seule clé pour chiffrer et déchiffrer) et asymétriques (une paire de clés publique et privée). Lorsqu'une clé est créée dans KMS, elle est stockée de manière sécurisée dans des modules matériels de sécurité (HSM) conformes aux normes de sécurité, ce qui empêche l'accès direct aux clés.

Lorsque vous souhaitez chiffrer des données, vous envoyez ces données à KMS avec la clé de chiffrement que vous souhaitez utiliser. KMS chiffre les données et retourne le texte chiffré, sans que vous ayez jamais besoin de voir ou manipuler directement la clé.

Pour déchiffrer les données, vous envoyez le texte chiffré à KMS avec la même clé, et KMS retourne les données en clair. Ce processus garantit que les clés restent protégées et que le chiffrement et le déchiffrement sont effectués de manière sécurisée.

KMS peut également être intégré avec d'autres services AWS pour chiffrer automatiquement les données, par exemple, les objets dans S3, les volumes EBS, ou les secrets dans AWS Secrets Manager.

De plus, KMS offre des fonctionnalités de gestion fine des accès, permettant de définir qui peut utiliser les clés et dans quelles circonstances. Cela se fait via des politiques de contrôle d'accès, assurant que seules les entités autorisées peuvent accéder aux clés de chiffrement.

### B. contrôle d'accès à l'origine

Le contrôle d'accès à l'origine (Origin Access Control) est un mécanisme de sécurité utilisé pour gérer et restreindre l'accès aux ressources qui se trouvent à l'origine d'un service de distribution, comme un serveur ou un bucket de stockage. Ce concept est particulièrement pertinent lorsqu'on utilise un réseau de distribution de contenu (CDN) comme Amazon CloudFront pour distribuer du contenu aux utilisateurs finaux.

Le contrôle d'accès à l'origine permet de restreindre l'accès direct aux ressources stockées à l'origine, par exemple dans un bucket S3. Au lieu de permettre à quiconque d'accéder directement

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

---

à ces ressources via l'URL de S3, on configure le bucket pour qu'il ne puisse être accessible que via le CDN (comme CloudFront).

Lorsque CloudFront reçoit une requête pour une ressource, il vérifie si la requête est autorisée en fonction des paramètres définis dans l'Origin Access Control. Si la requête est valide, CloudFront se connecte au bucket S3 pour récupérer la ressource demandée et la renvoie à l'utilisateur final.

Ce mécanisme ajoute une couche de sécurité en empêchant les utilisateurs d'accéder directement aux ressources d'origine, ce qui réduit les risques de contournement des contrôles d'accès ou de surcharge du serveur d'origine.

Le contrôle d'accès à l'origine peut aussi inclure des signatures URL, des politiques d'accès temporaires, et des restrictions basées sur l'IP pour s'assurer que seuls les utilisateurs autorisés accèdent aux ressources.

Ainsi, le contrôle d'accès à l'origine garantit que les ressources sensibles ou coûteuses ne sont accessibles que par les canaux sécurisés et prévus, comme un CDN, tout en protégeant l'infrastructure d'origine contre les abus ou les accès non autorisés.

### C. Cloudfront

Amazon CloudFront est un réseau de distribution de contenu (Content Delivery Network, ou CDN) fourni par Amazon Web Services (AWS). Son objectif principal est de distribuer des contenus web et des ressources à des utilisateurs à travers le monde de manière rapide, sécurisée et fiable.

Amazon CloudFront est un service Web qui accélère la distribution de votre contenu Web statique et dynamique, tel que les fichiers .html, .css, .js et les fichiers image, à vos utilisateurs. CloudFront diffuse votre contenu par le biais d'un réseau mondial de centres de données appelés emplacements périphériques. Lorsqu'un utilisateur demande le contenu que vous diffusez CloudFront, la demande est acheminée vers l'emplacement périphérique offrant la latence la plus faible (délai), afin que le contenu soit diffusé avec les meilleures performances possibles.

- Si le contenu se trouve déjà dans l'emplacement périphérique où la latence est la plus faible, CloudFront il est diffusé immédiatement.
- Si le contenu ne se trouve pas dans cet emplacement périphérique, CloudFront récupérez-le à partir d'une origine que vous avez définie, telle qu'un compartiment Amazon S3, un MediaPackage canal ou un serveur HTTP (par exemple, un serveur Web) que vous avez identifié comme source de la version définitive de votre contenu.

Par exemple, supposons que vous diffusez une image à partir d'un serveur Web traditionnel, et non à partir de CloudFront. Par exemple, vous pouvez diffuser une image, sunsetphoto.png, à l'aide de l'URL <https://example.com/sunsetphoto.png>.

## **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---

Vos utilisateurs peuvent facilement accéder à cette URL et voir l'image. Néanmoins, jusqu'à ce que l'image soit trouvée, ils ignorent probablement que leur demande a été transmise d'un réseau à un autre par le biais de l'enchevêtrement complexe de réseaux interconnectés qui forment Internet.

CloudFront accélère la diffusion de votre contenu en acheminant chaque demande utilisateur via le réseau AWS principal vers l'emplacement périphérique le plus à même de servir votre contenu. Il s'agit généralement d'un serveur CloudFront Edge qui fournit la diffusion la plus rapide au spectateur. L'utilisation du AWS réseau réduit considérablement le nombre de réseaux par lesquels les demandes de vos utilisateurs doivent passer, ce qui améliore les performances. Les utilisateurs bénéficient d'une latence plus faible (durée nécessaire au chargement du premier octet du fichier) et de débits de transfert des données plus élevés.

Il en résulte également une fiabilité et une disponibilité accrues, car des copies des fichiers (également appelés *objets*) sont désormais détenues (ou mises en cache) dans plusieurs emplacements périphériques situés aux quatre coins du monde.

## **CONCLUSION**

Cette partie liée à l'état de l'art a permis de faire une étude détaillée sur les différents services de sécurité AWS qui nous aidera dans la prochaine partie destinée à l'implémentation. La fin de ce chapitre nous met dans de bonne disposition afin d'entamer la phase sur l'implémentation a proprement dit.

## **CHAPITRE 4 : IMPLEMENTATION DE LA SOLUTION**

*Aperçu :*

### **INTRODUCTION**

- I. ARCHITECTURE DE LA SOLUTION**
- II. CONFIGURATION DE LA SOLUTION**

### **CONCLUSION**

## **INTRODUCTION**

L'objectif visé dans cette partie est de sécuriser une application qui tourne au sein de AWS permettant de superviser, de gérer, de connaître en temps réel ce que le système subit et de prévenir les attaques qui pourraient avoir lieu. Nous présenterons l'architecture, ensuite nous passerons aux différentes configurations des services.

## I. ARCHITECTURE DE LA SOLUTION

Pour intégrer notre solution à l'entreprise voici l'architecture que nous proposons.

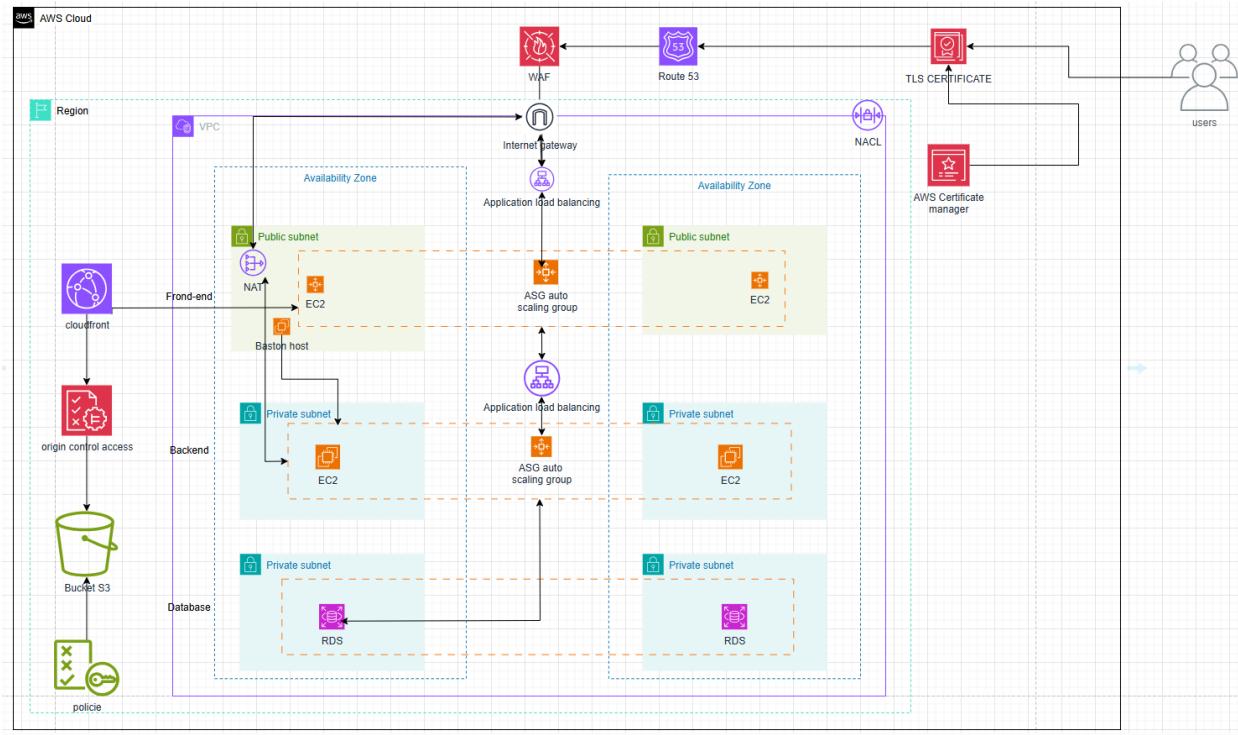


Figure 14: architecture de la solution avec draw.io

## II. IMPLEMENTATION DE LA SOLUTION

Pour ce qui est des prérequis pour cette implémentation, l'administrateur en charge devra maîtriser la console aws et la configuration de ses différents services.

### 1. Configuration et mise en place du réseau

Un VPC est un réseau privé dédié cloud, il est divisé en plusieurs parties appelées subnets (sous réseau).

#### Etape 1 : accéder à la console aws

Pour se faire il suffit d'aller dans le navigateur et d'y créer un compte ou d'avoir des informations de connexions d'un compte iam créé par un administrateur qui nous donne l'accès aux différents accès.

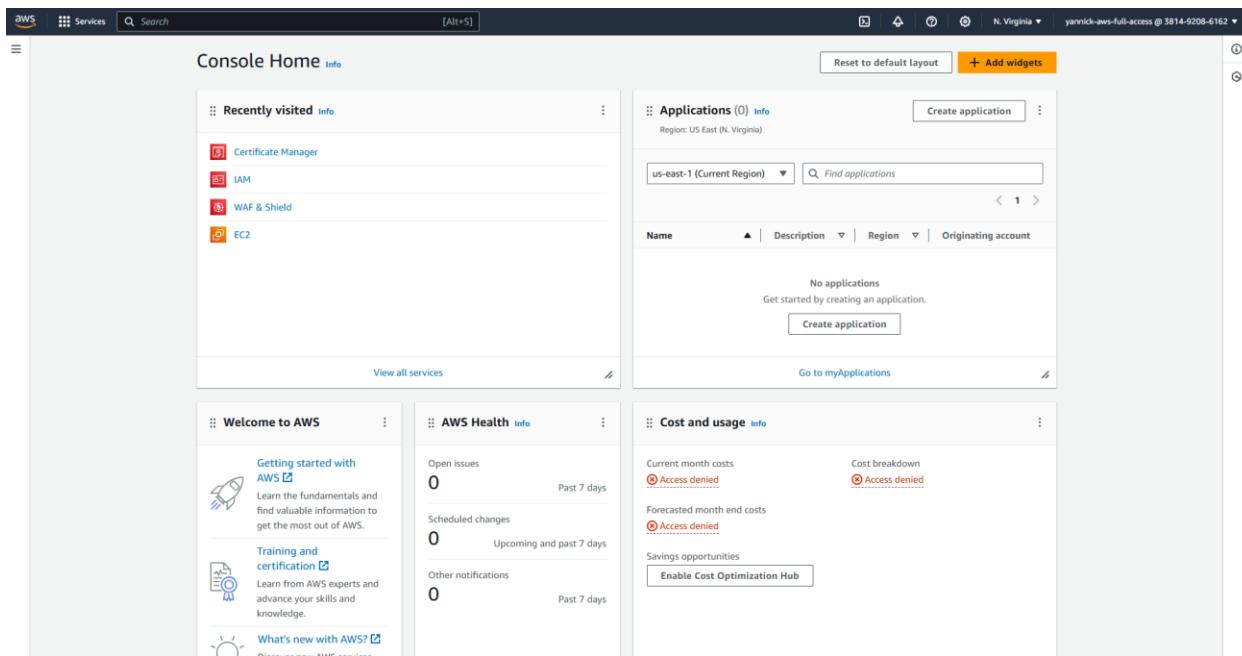


Figure 15: console aws

Dans cette figure on peut voir les services récemment utilisés en haut à gauche « **Recently visited** ». Ensuite nous avons plus à droite l'onglet « **Application** » qui va nous permettre de visualiser les applications que nous aurons lancées.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

L'onglet « **cost and usage** » détermine le coût et l'usage des différents services que nous aurons à utiliser.

L'onglet AWS Health permet de donner le niveau de santé de toute notre architecture. Et on le dernier onglet « **Welcome AWS** » qui est celui réservé à l'apprentissage dans aws.

L'espace de recherche situé juste au-dessus de l'onglet recently visited va nous permettre de trouver les différents services dont nous aurons besoin.

Après avoir recherché VPC et cliquer sur le service nous sommes rediriger vers cette page qui servira à la création et la sécurisation de celui-ci.

### **Etape 2 : création d'un VPC**

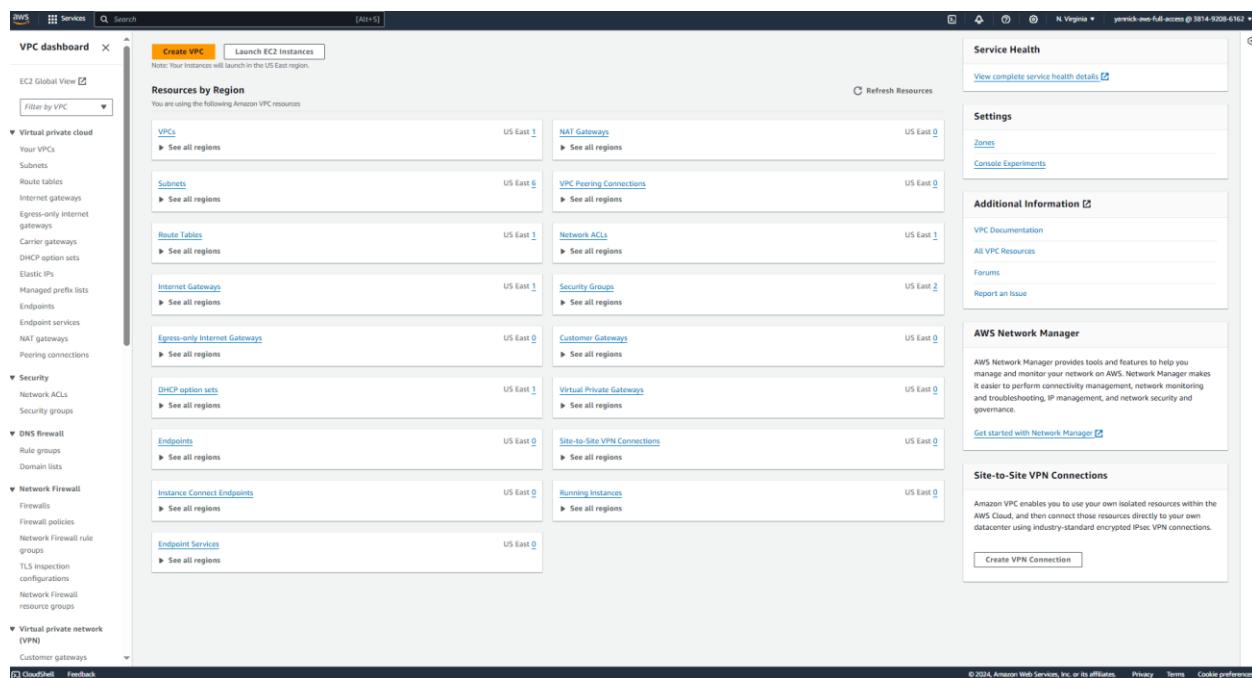
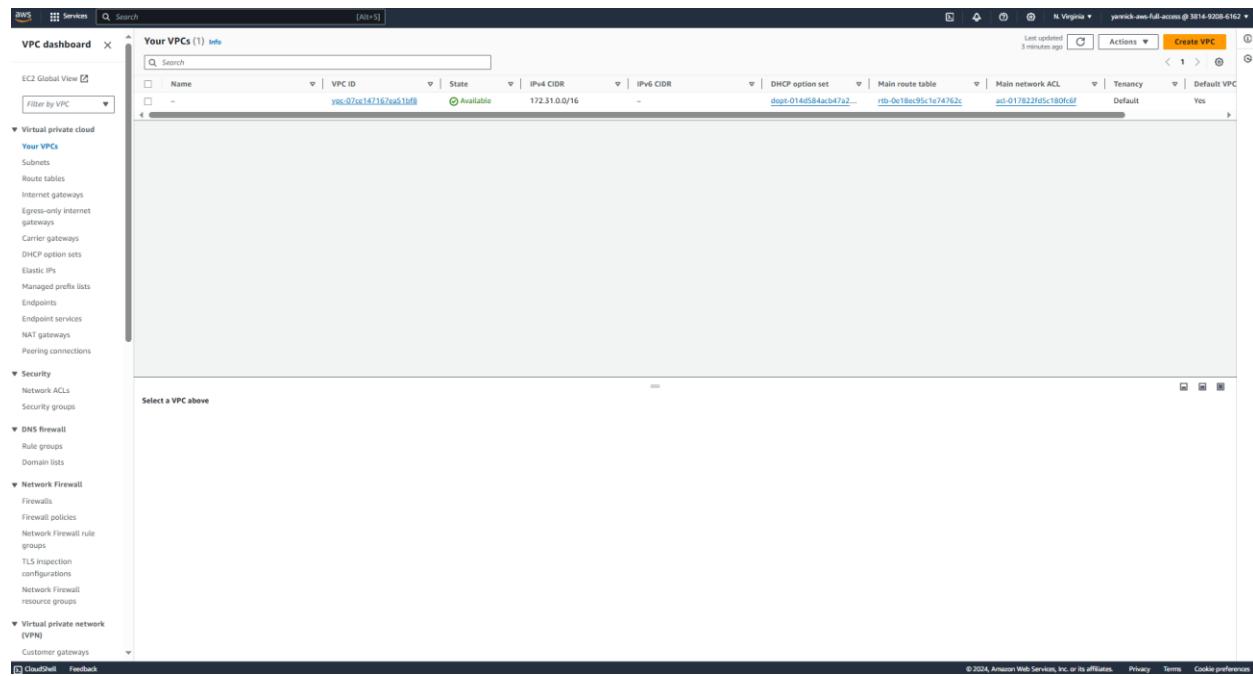


Figure 16: Interface de présentation de création d'un VPC

Nous allons commencer par cliquer sur VPCs pour commencer la configuration.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS



The screenshot shows the AWS VPC Dashboard. On the left, a sidebar lists various network-related services: EC2 Global View, Filter by VPC, Virtual private cloud (with sub-options like Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, NAT gateways, Peering connections), Security (Network ACLs, Security groups), DNS Firewall (Rule groups, Domain lists), Network Firewall (Firewalls, Firewall policies, Network Firewall rule groups, TLS inspection configurations, Network Firewall resource groups), and Virtual private network (VPN) (Customer gateways). The main panel displays a table titled "Your VPCs (1) info" with one row. The row contains the following information: Name (vpc-07ce147167ea51bf8), VPC ID (vpc-07ce147167ea51bf8), State (Available), IPv4 CIDR (172.31.0.0/16), IPv6 CIDR (-), DHCP option set (dhcp-014d3b4ac5e47a2...), Main route table (rtb-0e18c95e1e74762c), Main network ACL (acl-017822f05e180f0d), Tenancy (Default), and Default (Yes). A "Create VPC" button is located at the top right of the main panel.

Après avoir cliqué ici nous sommes redirigées ici qui nous montre tous les VPCs que nous avons créé et ceux à venir, ce qui nous intéresse c'est la création d'un VPC alors on va cliquer sur le bouton orange qui se trouve en haut à droite.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the 'Create VPC' wizard on the AWS Management Console. The first step, 'VPC settings', is selected. Under 'Resources to create', 'VPC only' is chosen. A 'Name tag - optional' field contains 'IAI-vpc'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, with '10.0.0.0/24' entered. Under 'IPv6 CIDR block', 'No IPv6 CIDR block' is selected. In the 'Tenancy' section, 'Default' is chosen. The second step, 'Tags', is shown with one tag 'Name: IAI-vpc' added. At the bottom right are 'Cancel' and 'Create VPC' buttons.

Figure 17: Création d'un VPC

Pour la configuration lors du VPC nous commençons d'abord par sélectionner si nous créer un VPC à part à entière ou l'associer à divers autres éléments, nous choisissons la première option qui correspond au VPC uniquement. Ensuite nous donnons un nom au VPC, ensuite nous l'attribuons une plage d'adresse qui a été au préalable coché plus haut, ensuite nous cochons l'option qui spécifie que nous n'aurons pas besoin d'une IPV6 pour notre VPC ensuite nous cliquons sur create VPC et ensuite il est créé.

Après avoir créé le réseau nous créons des sous-réseaux

### Etape 3 : Création des subnets

Pour créer un subnet nous allons rester dans le service VPC et cliquer sur l'option correspondante « subnets » à gauche de l'écran et ensuite nous nous rendrons sur le bouton create qui se trouve en haut à droite de l'écran.

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the AWS VPC dashboard with the 'Subnets' section selected. The main area displays a table of subnets with the following columns: Name, Subnet ID, State, VPC, IPv4 CIDR, IPv6 CIDR, IPv6 CIDR association ID, Available IPv4 addresses, and Availability Zone. There are seven subnets listed, all in the 'Available' state. The VPC column shows 'vpc-07ce147167ea510f8'. The IPv4 CIDR column lists ranges from 172.31.32.0/20 to 172.31.48.0/20. The Availability Zone column shows 'us-east-1c', 'us-east-1a', 'us-east-1b', 'us-east-1d', and 'us-east-1e'. A search bar at the top is set to '(All+5)'. On the left sidebar, under the 'Subnets' section, there is a link to 'Create subnet'.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association ID	Available IPv4 addresses	Availability Zone
-	subnet-0a48f720fd4328cf6	Available	vpc-07ce147167ea510f8	172.31.32.0/20	-	-	4091	us-east-1c
-	subnet-0e502f2802dd5a5d6	Available	vpc-07ce147167ea510f8	172.31.80.0/20	-	-	4091	us-east-1a
-	subnet-0f31b0420478021	Available	vpc-07ce147167ea510f8	172.31.16.0/20	-	-	4091	us-east-1b
-	subnet-daf709e67b507e5	Available	vpc-07ce147167ea510f8	172.31.0.0/20	-	-	4091	us-east-1d
-	subnet-0ef54f7480c16cc01	Available	vpc-07ce147167ea510f8	172.31.64.0/20	-	-	4091	us-east-1f
-	subnet-0dabcc00942a85601	Available	vpc-07ce147167ea510f8	172.31.48.0/20	-	-	4091	us-east-1e

Figure 18: interface de vue des subnets ayant déjà été créé

En première vu nous observons l'ID du VPC à laquelle est associé notre subnet et l'adresse du VPC, plus bas nous nommons le subnet « **IAI** », et nous choisissons comme zone de disponibilité

« **US East (N. Virginia) us-east-1a** » l'adresse IP qui lui est assigné est la même que celle du VPC mais nous changeons l'IP plus bas ici car il s'agit d'un subnet privée. Ensuite nous cliquons sur « **create subnet** »

Nous répétons la même étape pour les 5 autres subnets à venir.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the 'Create subnet' wizard in the AWS VPC service. The 'VPC ID' dropdown is set to 'vpc-07ce147167ea51bf8'. Under 'Associated VPC CIDRs', the IPv4 CIDR is listed as '172.31.0.0/16'. In the 'Subnet settings' section, 'Subnet 1 of 1' is being configured. The 'Subnet name' is 'IAI', and the 'Availability Zone' is 'US East (N. Virginia) / us-east-1a'. The 'IPv4 VPC CIDR block' is '172.31.0.0/16'. The 'IPv4 subnet CIDR block' is '192.168.0.0/16', which is highlighted with a blue border. The 'Tags - optional' section contains a single tag 'Name: IAI'. At the bottom right are 'Cancel' and 'Create subnet' buttons.

Figure 19: création des subnets

### Etape 4 : création des security groups

Les security groups sont présent dans le Dashboard du vpc dans l'onglet security a gauche plus bas. Pour créer un security group nous cliquons sur create security group

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the AWS VPC dashboard with the 'Security Groups' section selected. The table lists two security groups:

Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound
-	sg-03462737e99b01b30	launch-wizard-1	vpc-07ce147167ea51bf8	launch-wizard-1 created 2024-08-14T...	381492086162	3 Perm
-	sg-01092d209195e96fe	default	vpc-07ce147167ea51bf8	default VPC security group	381492086162	4 Perm

Figure 20: tableau de bord des security groups

The screenshot shows the 'Create security group' interface. The 'Basic details' section is filled with the following information:

- Security group name: 'My security group'
- Description: 'Allows SSH access to developers'
- VPC: 'vpc-07ce147167ea51bf8'

The 'Inbound rules' section contains one rule:

Type	Protocol	Port range	Source	Description
Custom TCP	TCP	0	Custom	Allow SSH access to developers

The 'Outbound rules' section contains one rule:

Type	Protocol	Port range	Destination	Description
All traffic	All	All	Custom	0.0.0.0/0

At the bottom right, there are 'Cancel' and 'Create security group' buttons.

Figure 21: Interface de création d'un security group

Dans l'onglet « **basic detail** » nous entrons le nom la description du groupe de sécurité et nous attachons le VPC auquel il est alloué.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Ensuite on se retrouve dans l'onglet « **Inbound rule** », il s'agit de la configuration des règles d'entrées c'est-à-dire les règles qui devront être appliquée au trafic entrant dans le réseau.

Enfin nous avons « **Inbound rules** » qui fait référence aux configurations des règles du trafic sortant du réseau. Ici nous permettons à tous les trafic de pouvoir sortir. Ensuite nous cliquons sur « **Create security group** »

### *Etape 4 : création des NACL*

Comme les security group les subnets se trouvent dans le dashboard du VPC au niveau de l'onglet « **security** ».

Name	Network ACL ID	Associated with	Default	VPC ID	Inbound rules count	Outbound rules count	Owner
acl-01782fd5c180fc6f	6 Subnets	Yes	vpc-07ce147167ea51bf8	2 Inbound rules	2 Outbound rules	38149208162	

Figure 22: tableau de bord des NACL

Pour créer un NACL nous allons tous simplement cliquer sur le bouton « **Create network ACL** » qui se trouve en haut et à droite.

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

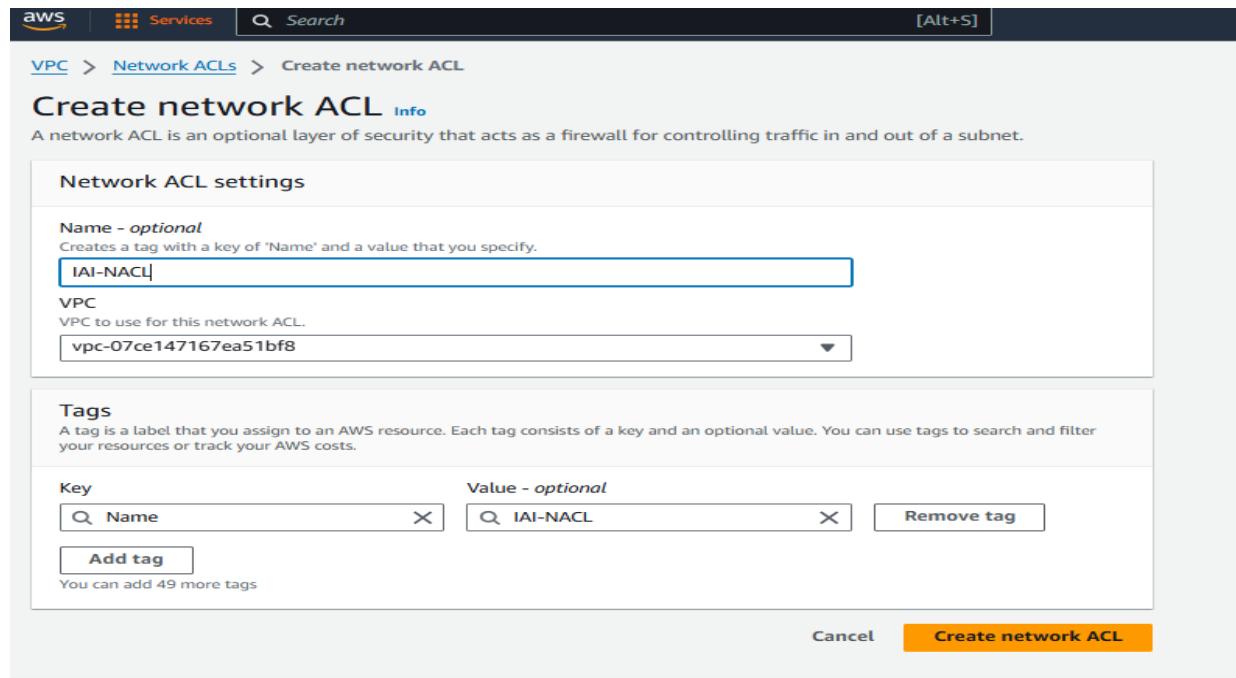


Figure 23: création des NACL

Après avoir rempli le nom du Nacl (IAI-NACL) et que nous l'ayons attaché au VPC correspondant nous lui assignons une clé de nom « **Name** » et de valeur « **IAI-NACL** » ensuite nous créons le NACL en cliquant sur le boutons « **Create NACL** »

## Etape 5 : Création du Gateway

Le gateway va permettre que le réseau puisse directement communiquer avec internet. Il est directement lié au VPC dans lequel il est créé. Il est répertorié dans le dashboard du VPC dans l'onglet « **Virtual private cloud** »

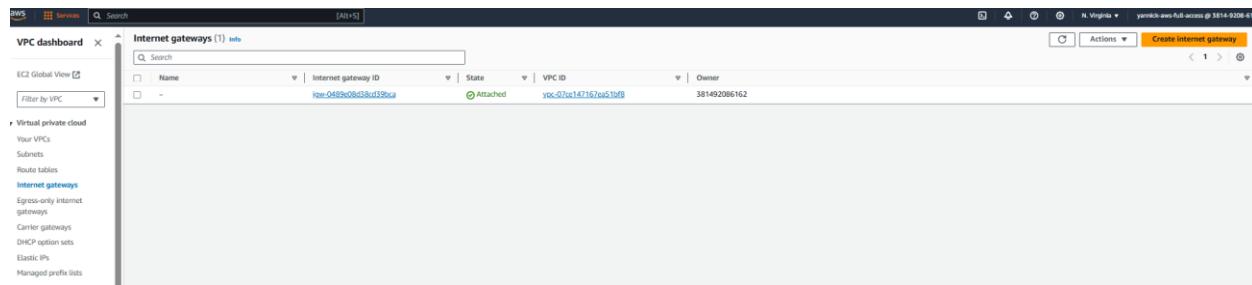


Figure 24: dashboard gateway

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the 'Create internet gateway' configuration page. At the top, there's a breadcrumb navigation: 'VPC > Internet gateways > Create internet gateway'. Below it, the title 'Create internet gateway' is followed by a small 'Info' link. A descriptive text explains that an internet gateway is a virtual router connecting a VPC to the internet, and prompts the user to specify a name for the gateway. The main section is titled 'Internet gateway settings'. It contains two main fields: 'Name tag' and 'Tags - optional'. In the 'Name tag' field, the value 'IAI-gateway' is entered. Under 'Tags - optional', a single tag 'Name: IAI-gateway' is listed. There are buttons for 'Add new tag' and 'Remove'. At the bottom right, there are 'Cancel' and 'Create internet gateway' buttons, with the latter being orange.

Figure 25: création de la passerelle

Nous assignons un nom à la passerelle « **IAI-gateway** » et nous cliquons sur « **create internet gateway** »

## 2. Création des instance EC2, ASG et load balancers (auto scaling group)

### *Etape 1 : création d'une ASG*

Après avoir recherché et trouver le service Amazon EC2 Auto Scaling. Nous sommes renvoyés sur cette page qui est relative à toute les configurations tournant autour de EC2. Nous cliquons juste sur « **Create Auto Scaling group** »

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

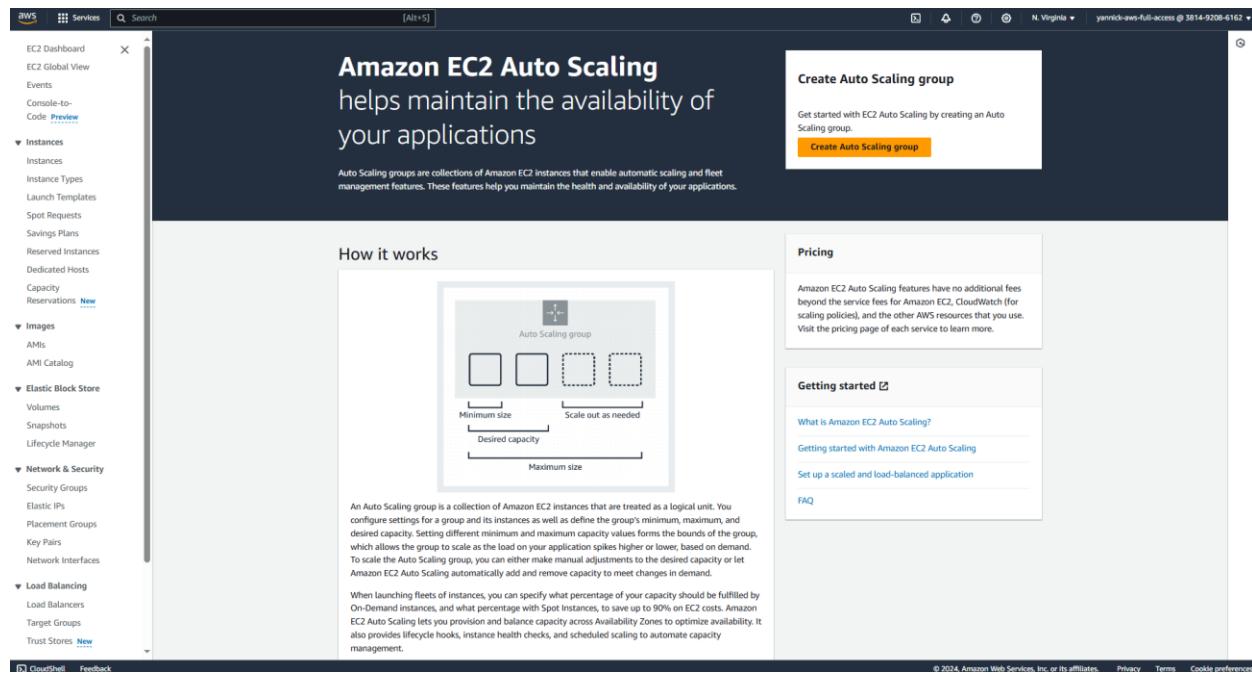


Figure 26: interface de création d'une ASG

A screenshot of the "Create Auto Scaling group" wizard. Step 1: Choose launch template. It asks for a "Name" (Auto Scaling group name) and a "Launch template". The "Name" field contains "IAI-AUTOSCALING\_group". A note says: "For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023." The "Launch template" dropdown is empty. Buttons at the bottom are "Cancel" and "Next Step".

Figure 27:création d'une ASG

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Nous sommes renvoyés sur cette page pour créer notre ASG, nous entrons le nom de notre ASG « **IAI-autoscaling\_group** » ensuite nous allons créer un template en cliquant sur le texte bleuté « **Create a launch template** » qui va nous permettre de créer et de lier directement notre instance EC2 à notre ASG.

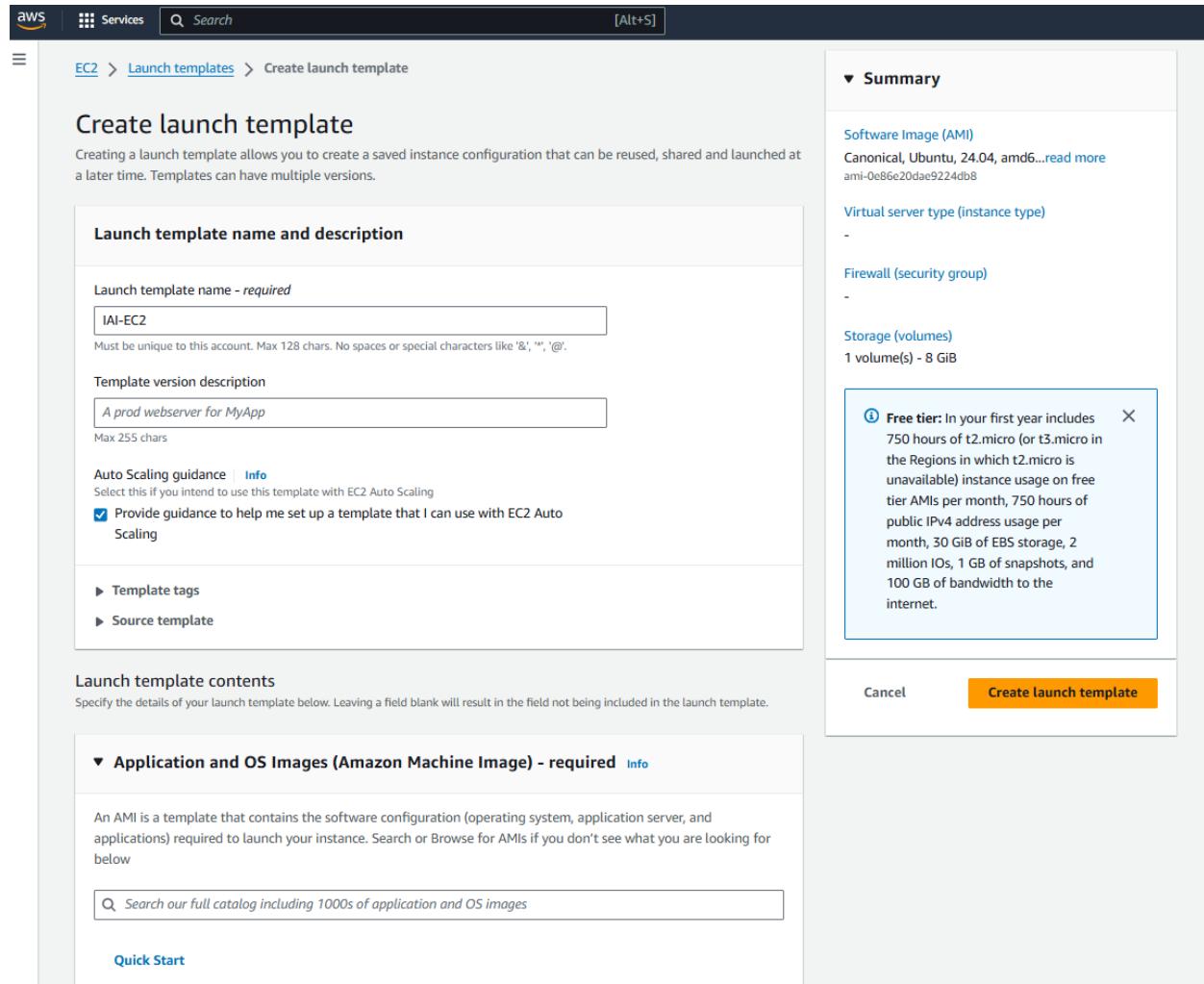


Figure 28: création de l'instance EC2

On va entrer le nom du template (**IAI-EC2**) ensuite nous allons choisir le système d'exploitation dans lequel nous allons choisir de lancer nos instances. Le noms que l'on donne a ces systèmes sont des AMI (Amazon machine image). Nous avons choisi la machine ubuntu ensuite nous cliquons sur « **create a launch template** »

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

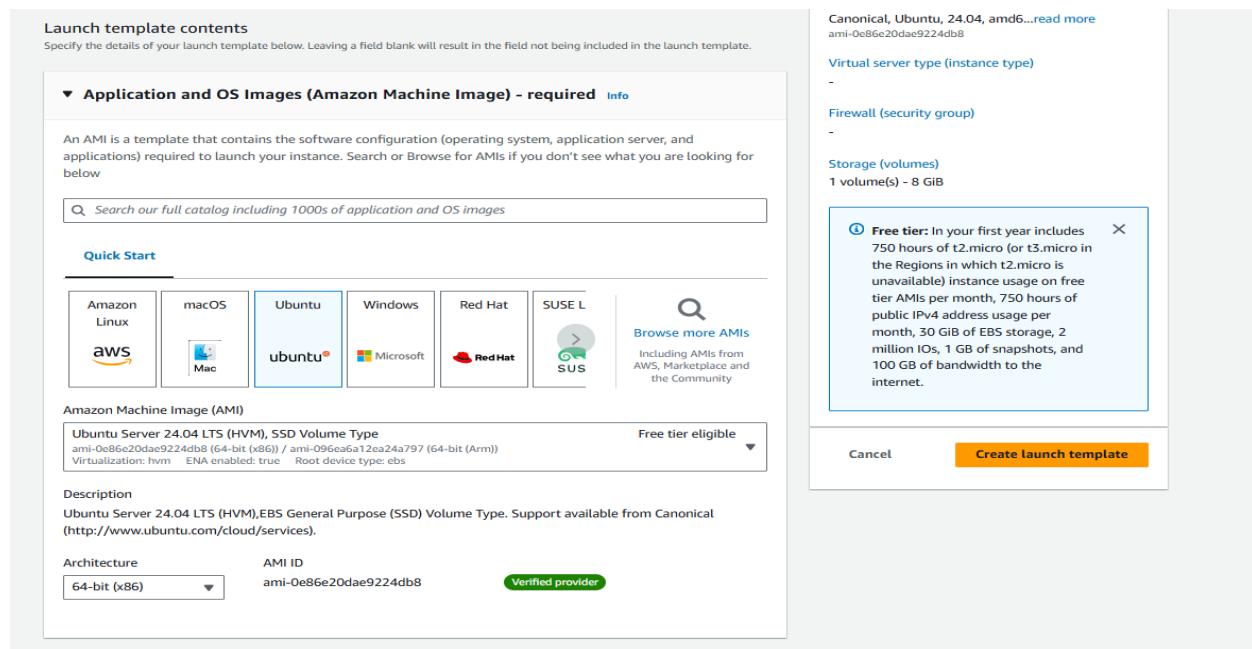


Figure 29: création d'une instance EC2

## Etape 2 : création et configuration des application load balancers

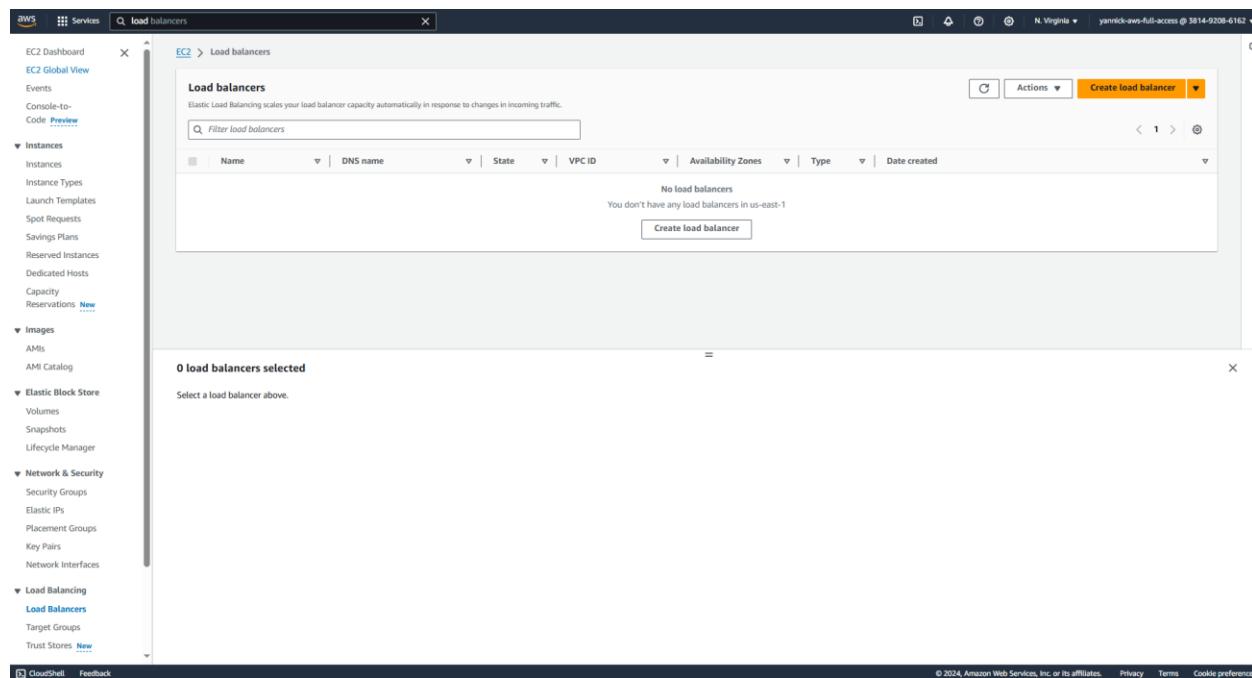


Figure 30: dashboard load balancers

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Cette option se trouve dans l'onglet « **Load balancing** » du service « **ASG** ». On clique sur « **create load balancer** » pour sa création.

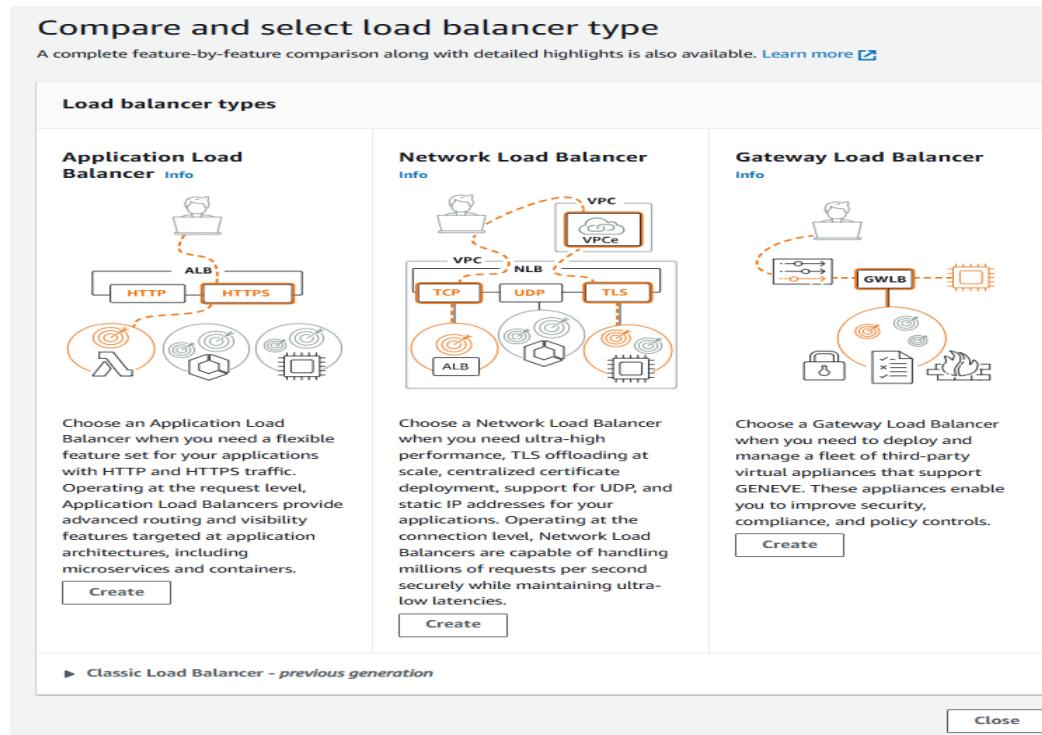


Figure 31: sélection de l'ALB

Ensuite nous sommes renvoyés à cette page où nous cliquons sur « **create** » de la partie « **Application Load Balancer** »

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

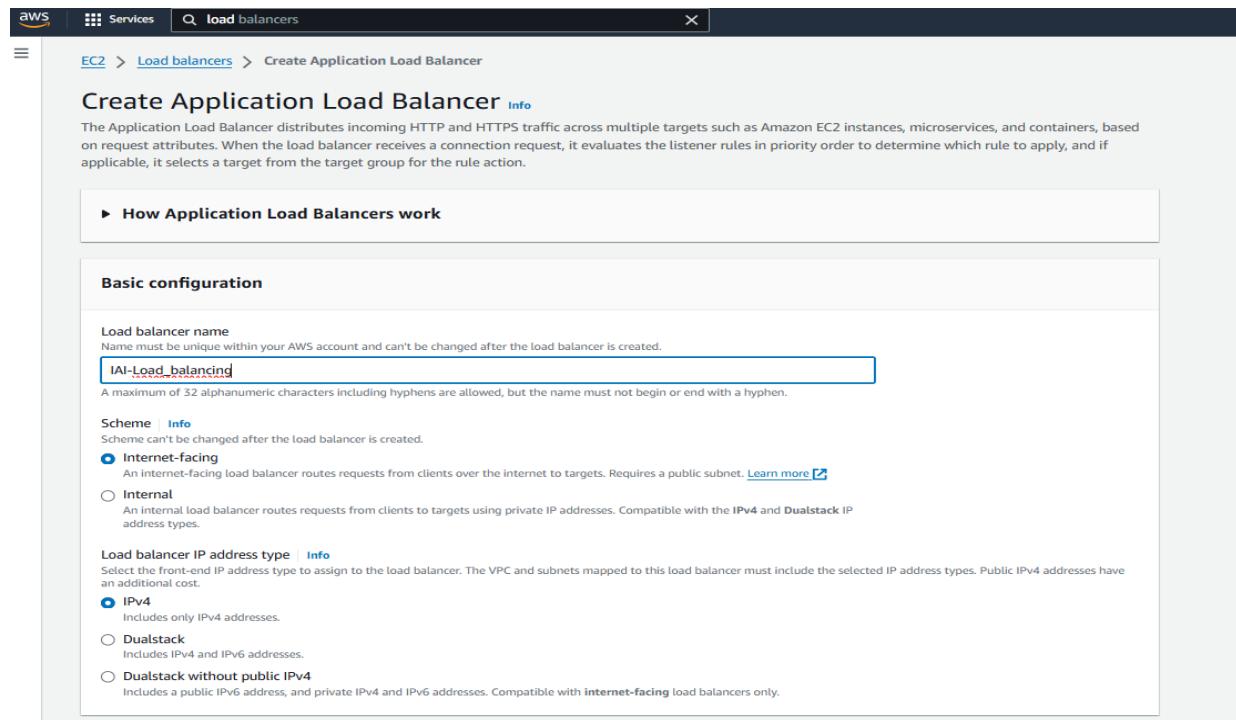


Figure 32: configuration de L'ALB

Après avoir entré le nom du load balancer (**IAI-Load\_balancing**) nous sélectionnons « **Internet-facing** » de l’option « **scheme** » pour permettre à l’ALB de répondre aux requêtes des clients venant d’internet le plus rapidement possible. Et nous cliquons sur « **IPv4** » de l’option « **Load balancer IP address type** » pour qu’il n’inclue que les adresses IPV4.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the 'Review' step of creating an Application Load Balancer (ALB). It displays various configuration sections:

- Summary**: Shows basic configurations like IAI-Load\_balancing (Internet-facing, IPv4), Security groups (default sg-01092d209195e96fe), Network mapping (VPC vpc-07ce147167ea51bf8, Subnet not defined), and Listeners and routing (HTTP:80 default to Target group not defined).
- Service integrations**: Lists AWS WAF and AWS Global Accelerator.
- Tags**: Shows 'None'.
- Attributes**: A note states: "Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer."
- Creation workflow and status**: A section titled "▶ Server-side tasks and status" notes: "After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring."
- Buttons**: 'Cancel' and 'Create load balancer' (highlighted in orange).

Figure 33: configuration d'un ALB

Dans ce dernier onglet nous voyons tous ce qui a été configurés pour enfin valider sur le bouton « **create load balancer** »

### 3. Création et configuration de l'ACM

Etant dans la console nous tapons ACM dans la barre de recherche et nous cliquons sur le service correspondant. Et nous sommes renvoyés à la page d'accueil du service.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

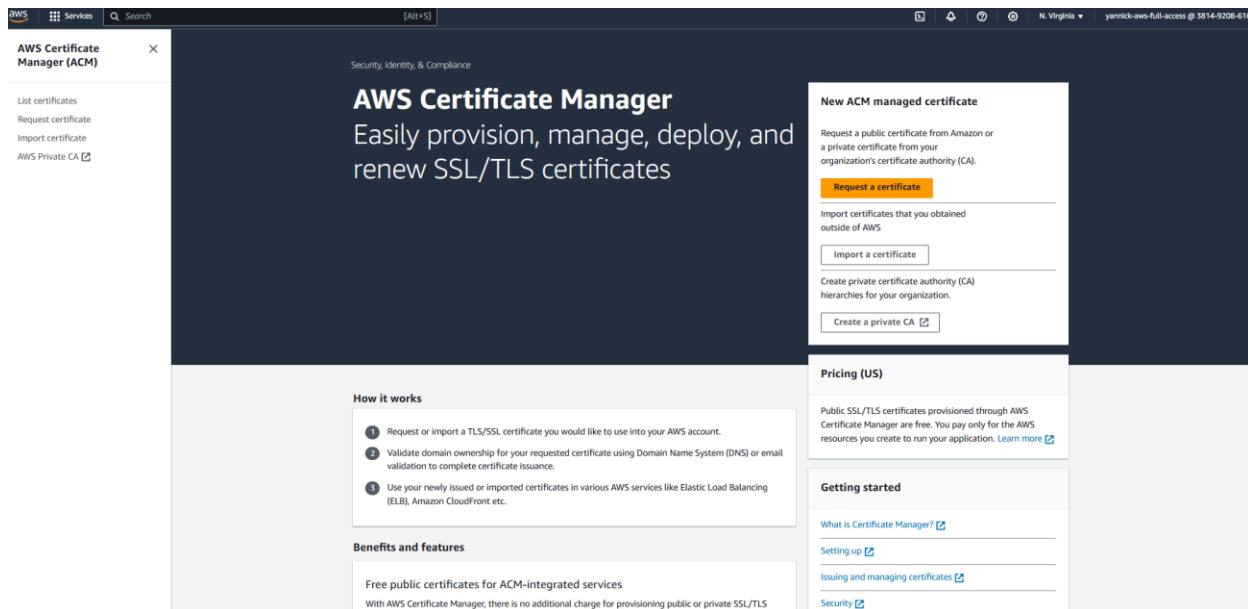


Figure 34: page d'accueil de ACM

Dans cette page on clique sur « **Request certificate** », mais nous pouvons très bien importer un certificat ou en créer un à l'aide des options respective « **Import a certificate** » et « **create a private CA** ».

This is a screenshot of the 'Request certificate' wizard. The breadcrumb navigation shows 'AWS Certificate Manager > Certificates > Request certificate'. The main heading is 'Request certificate'. A section titled 'Certificate type' includes a note: 'ACM certificates can be used to establish secure communications access across the internet or within an internal network. Choose the type of certificate for ACM to provide.' Two radio buttons are shown: 'Request a public certificate' (selected) and 'Request a private certificate'. A note below states: 'Requesting a private certificate requires the creation of a private certificate authority (CA). To create a private CA, visit AWS Private Certificate Authority.' At the bottom right are 'Cancel' and 'Next' buttons.

Figure 35: demande de certificat

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

On fait la requête d'un certificat public. Et ensuite on clique sur « **Next** »

The screenshot shows the 'Request public certificate' wizard in AWS Certificate Manager. The current step is 'Domain names'. A navigation bar at the top shows: AWS Certificate Manager > Certificates > Request certificate > Request public certificate. The main section is titled 'Domain names' with the sub-instruction 'Provide one or more domain names for your certificate.' A text input field contains 'IAI.worketyamo.com'. Below it is a button 'Add another name to this certificate' and a note about adding additional names. The next section is 'Validation method' with 'DNS validation - recommended' selected. The final section is 'Key algorithm' with 'RSA 2048' selected. At the bottom are 'Tags' and 'Cancel/Previous/Request' buttons.

Figure 36: validation de la requête d'un certificat public

Nous renseignons le nom de domaine (**IAI.worketyamo.com**). Nous choisissons la méthode de validation comme étant celle avec le DNS et l'algorithme de configuration choisit est le « **RSA 2048** »

Ensuite nous cliquons sur le bouton Request.

### 4. Création et configuration du WAF (web application firewall)

Dans l'option de recherche, recherchons le service AWS WAF et cliquons sur le service correspondant.

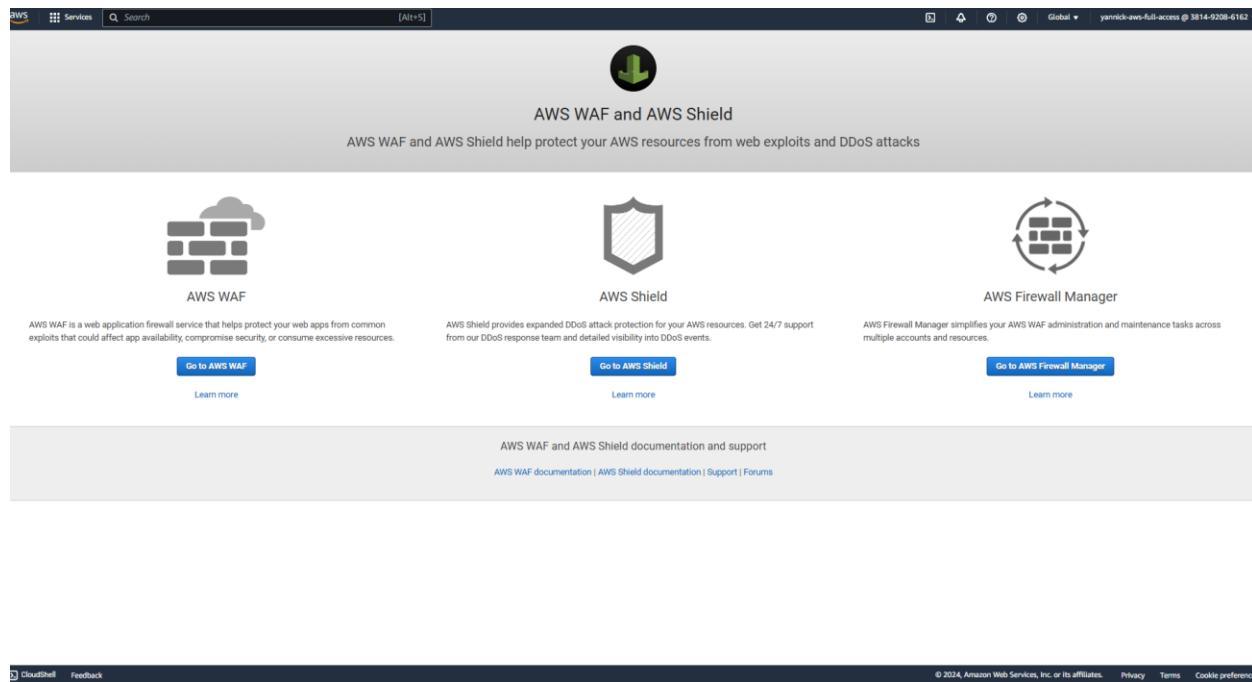


Figure 37: choix du parefeu approprié

Dans cette page nous cliquons sur AWS WAF. Le clique va nous rediriger sur une page

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

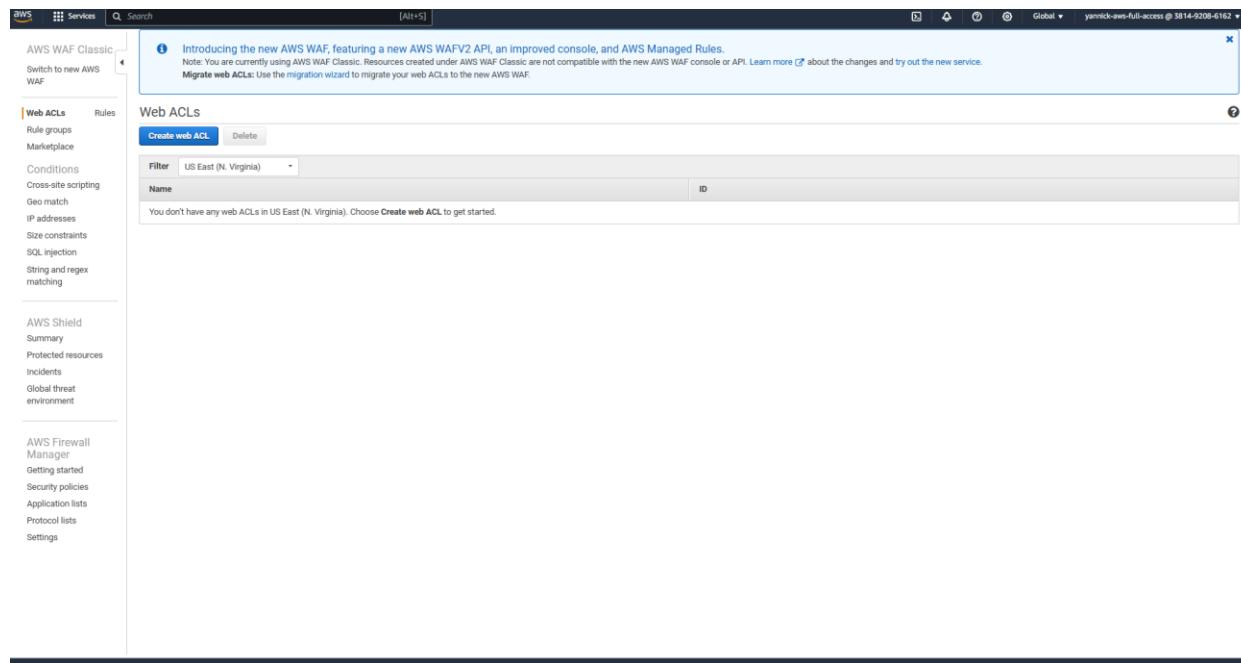


Figure 38: dashboard du WAF

Nous sommes dans le dashboard du WAF, ici on crée un WAF et ses règles en cliquant sur create a « **web ACL** ».

Step 1: Name web ACL

To create a web ACL that you want to use to filter web requests, type a name for your web ACL, and then choose Next. [Learn more](#)

Web ACL name\*

CloudWatch metric name\*

Region\*

Resource type to associate with web ACL

You can associate additional resource types after you finish creating this web ACL. Add additional resource types on the Rules tab for this web ACL.

\* Required

Cancel Previous Next

Figure 39: création d'un WAF 1

Nous entrons le nom du WAF « **IAI-web** » et le nom de la métrique cloudwatch est entré immédiatement avec le nom du WAF. Après la sélection de la région (**USEast (N. Virginia)**). On clique sur « **NExt** »

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

## Set up a web access control list (web ACL)

The screenshot shows the AWS WAF 'Create conditions' interface. It includes sections for:

- Cross-site scripting match conditions:** A table with 'Name' and 'Create condition' columns. A note says: "You don't have any cross-site scripting match conditions. Choose [Create XSS match condition](#) to get started." Below it, a description states: "A cross-site scripting match condition specifies the parts of a web request (such as a User-Agent header) that you want AWS WAF to inspect for cross-site scripting threats. [Learn more](#)".
- Geo match conditions:** A table with 'Name' and 'Create condition' columns. A note says: "You don't have any geo match conditions. Choose [Create condition](#) to get started." Below it, a description states: "A geo match condition lets you allow, block, or count web requests based on the geographic origin of the request. [Learn more](#)".
- IP match conditions:** A table with 'Name' and 'Create condition' columns. A note says: "You don't have any IP match conditions. Choose [Create IP match condition](#) to get started." Below it, a description states: "An IP match condition specifies the IP addresses and/or IP address ranges that you want to use to control access to your content. Put IP addresses that you want to allow and IP addresses that you want to block into separate IP match conditions. [Learn more](#)".
- Size constraint conditions:** A table with 'Name' and 'Create condition' columns. A note says: "You don't have any size constraint conditions. Choose [Create size constraint condition](#) to get started." Below it, a description states: "A size constraint condition specifies the parts of a web request (such as a User-Agent header) that you want AWS WAF to inspect for size constraints. [Learn more](#)".

To the right, a sidebar titled 'Concepts overview' contains:

- Web ACL example:** if requests match
  - Rule 1, Bad User-Agents, then block:** IP match condition: Suspicious IPs
  - and**
  - String match condition:** Bad bots
- or if requests match
  - Rule 2, Detect SQLI, then block:** SQL injection match condition: SQLi checks
- otherwise, perform the default action
  - Default action:** Allow requests that don't match any rules

Figure 40: création des conditions

C'est dans cette partie que nous créons des conditions pour protéger notre architecture contre les attaques de différents types. Comme mentionnée plus haut. Et dans l'onglet nous avons la configuration des web ACL.

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

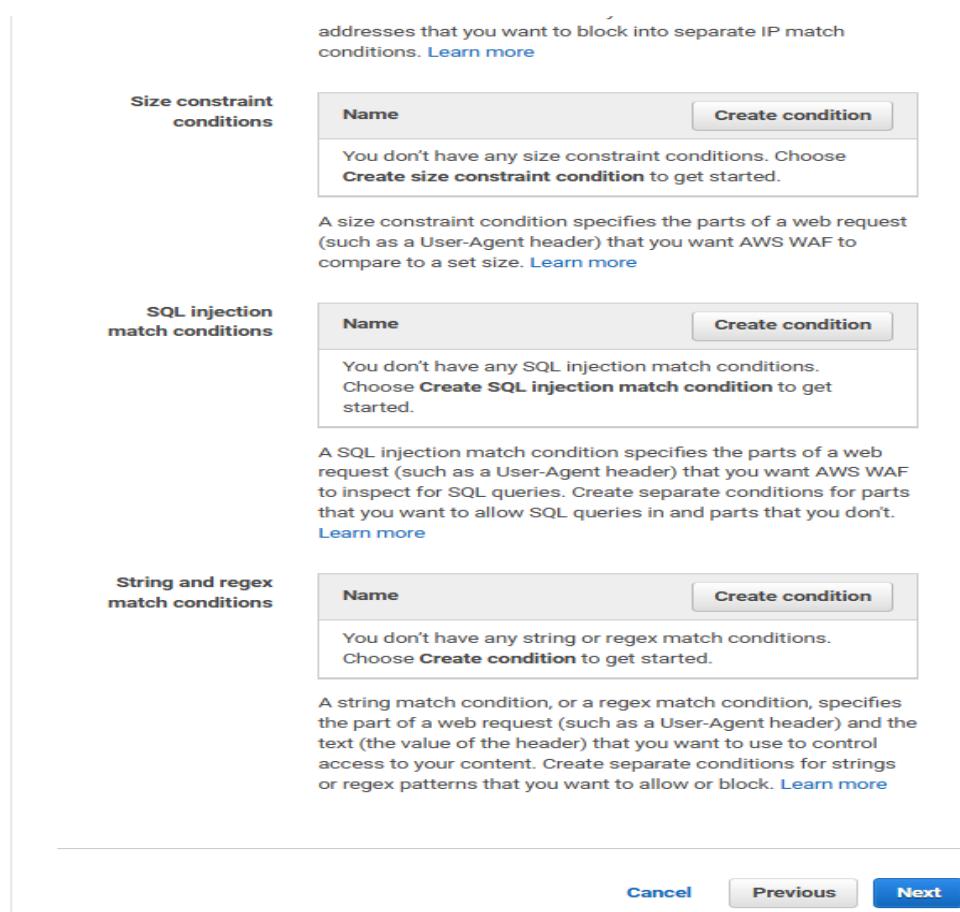


Figure 41: continuation des conditions

The screenshot shows the AWS WAF Rules creation interface. On the left, there's a sidebar with navigation steps: Concepts overview, Step 1: Name web ACL, Step 2: Create conditions, **Step 3: Create rules**, and Step 4: Review and create. The main area is titled "Create rules" and contains the following components:

- Create rules**: A title with a help icon.
- Rules**: A dropdown menu with "Select a rule" and a "Create rule" button.
- Conditions**: A section with two tables:
  - If a request matches all of the conditions in a rule, take the corresponding action**
  - If a request doesn't match any rules, take the default action**
- Default action\***: Radio buttons for "Allow all requests that don't match any rules" and "Block all requests that don't match any rules".
- Buttons**: "Cancel", "Previous", "Review and create".

**Concepts overview** sidebar (right side of the interface):
 

- Web ACL example**: if requests match
  - Rule 1, Bad User-Agents, then block**
    - IP match condition: Suspicious IPs
    - and
    - String match condition: Bad bots
- or if requests match
  - Rule 2, Detect SQLI, then block**
    - SQL injection match condition: SQLi checks
- otherwise, perform the default action
  - Default action**: Allow requests that don't match any rules

Figure 42: création des règles

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Pour que le pare-feu puisse filtrer il lui faut des règles et c'est à ce niveau que ce font ces configurations. On ajoute une règle à partir du bouton « **create rule** » après l'ajout des règles on peut terminer la configuration en cliquant sur le bouton « **confirm and create** ».

### Set up a web access control list (web ACL)

The screenshot shows the 'Review and create' step of the AWS WAF configuration wizard. It includes:

- Concepts overview**: Step 4: Review and create (highlighted).
- Step 1: Name web ACL**: IAI-waf
- Step 2: Create conditions**
- Step 3: Create rules**
- Step 4: Review and create**

**Review and create** section:  
Review your settings, and then choose Confirm and create to finish creating your web ACL.

Web ACL name	IAI-waf
CloudWatch metric name	IAIwaf

**Rules and actions** section:  
AWS WAF inspects each web request that an AWS resource receives and compares the request with the conditions in the following rules in the order listed. If a request doesn't match all of the conditions in at least one rule, AWS WAF takes the default action.

Order	Rule	Action
Create new rule using IP match or string match conditions created in previous step.		

**If a request matches a condition in a rule, take the corresponding action**

Order	Rule	Action
Create new rule using IP match or string match conditions created in previous step.		

**If a request doesn't match any rules, take the default action**

Default action
Allow

**AWS resources using this web ACL**

Resource	Type
No resource is using this web ACL.	

**Buttons**: Cancel, Previous, Confirm and create.

Figure 43:fin de la configuration du WAF

## 5. Configuration de S3

Nous recherchons le service ensuite nous cliquons dessus, ensuite nous sommes renvoyés sur cette page.

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

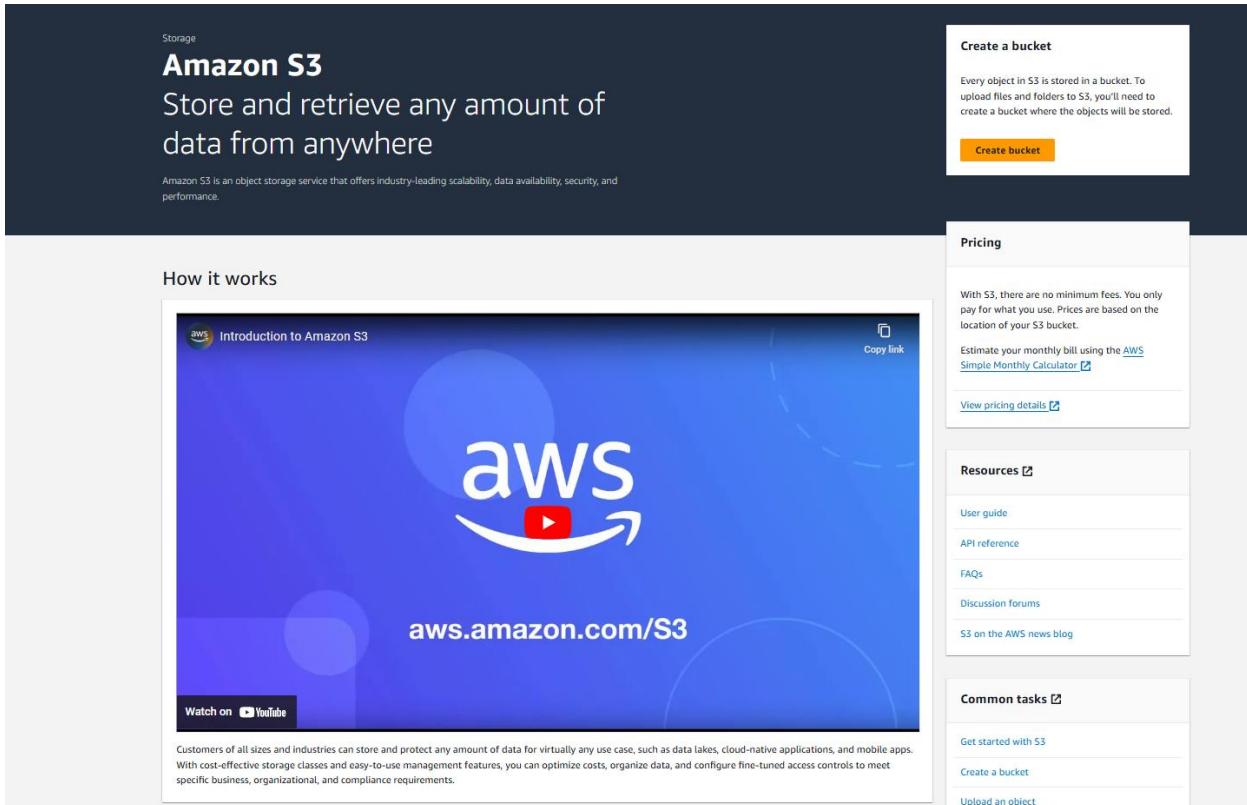


Figure 44: page d'accueil de S3

Ensuite nous cliquons sur « create bucket »

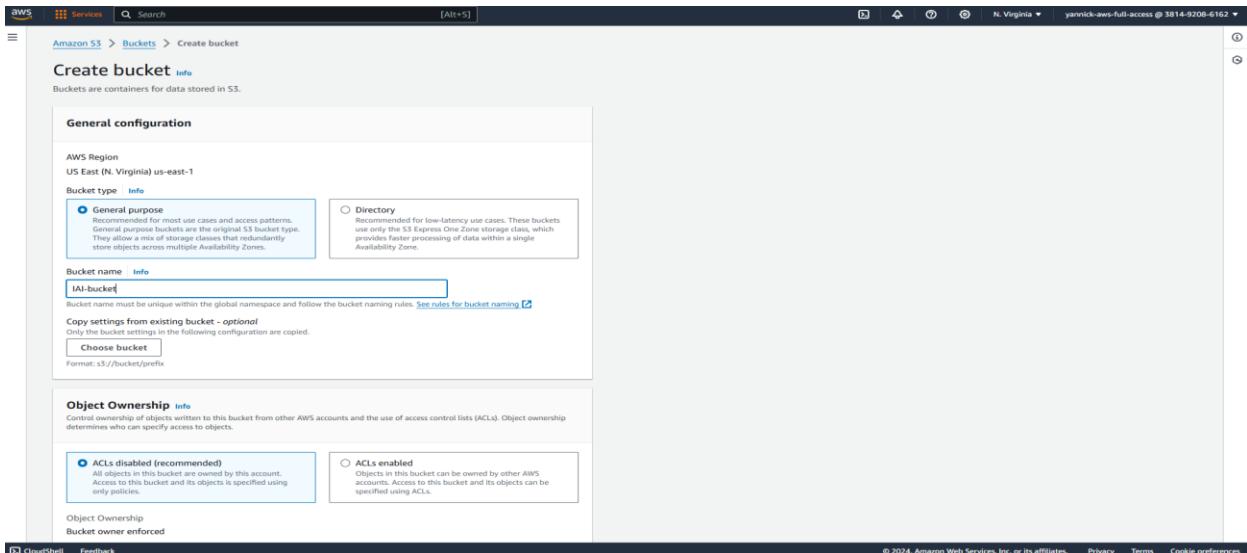


Figure 45: création d'un bucket

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Nous choisissons le type de bucket « **General purpose** » ensuite nous indiquons le nom de notre bucket S3 « **IAI-bucket** » et dans l'onglet « **object ownership** » nous choisissons l'option recommandé qui est de désactivé les ACL.

The screenshot shows the AWS S3 Bucket Configuration page for a bucket named "IAI-bucket".

- Block Public Access settings for this bucket:** The "Block all public access" checkbox is checked. A note states: "Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another." The four sub-options are all checked:
  - Block public access to buckets and objects granted through new access control lists (ACLs)**: S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
  - Block public access to buckets and objects granted through any access control lists (ACLs)**: S3 will ignore all ACLs that grant public access to buckets and objects.
  - Block public access to buckets and objects granted through new public bucket or access point policies**: S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
  - Block public and cross-account access to buckets and objects through any public bucket or access point policies**: S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.
- Bucket Versioning:** The "Disable" radio button is selected.
- Tags - optional (0)**: No tags are associated with this bucket. An "Add tag" button is available.
- Default encryption**:
  - Info**: Server-side encryption is automatically applied to new objects stored in this bucket.
  - Encryption type**:
    - Info**: Server-side encryption with Amazon S3 managed keys (SSE-S3) is selected.
    - Server-side encryption with AWS Key Management Service keys (SSE-KMS)**

Figure 46: configuration de s3 1

Dans cette configuration nous bloquons toutes les adresses IP public afin qu'elle ne puisse pas accéder au bucket S3 ensuite nous désaktivons le versionning. Et dans la case « **Default**

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

« encryption » nous activons le chiffrement des données au repôt en cliquant sur « **Server-side encryption with Amazon S3 managed keys (SSE-S3)** » ensuite pour finir nous cliquons sur le bouton « **create bucket** ».

## 6. Configuration de route 53

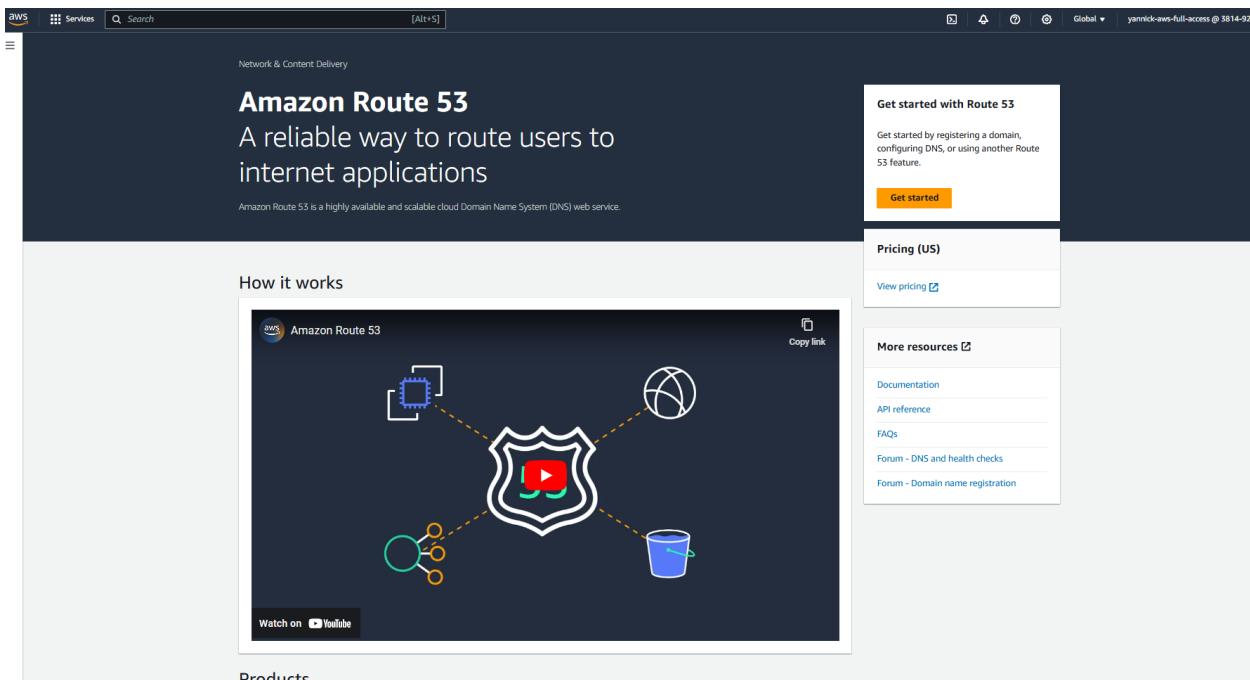


Figure 47: Page d'accueil de route 53

Etant sur la page d'accueil de route 53 nous allons cliquer sur « **Get started** ».

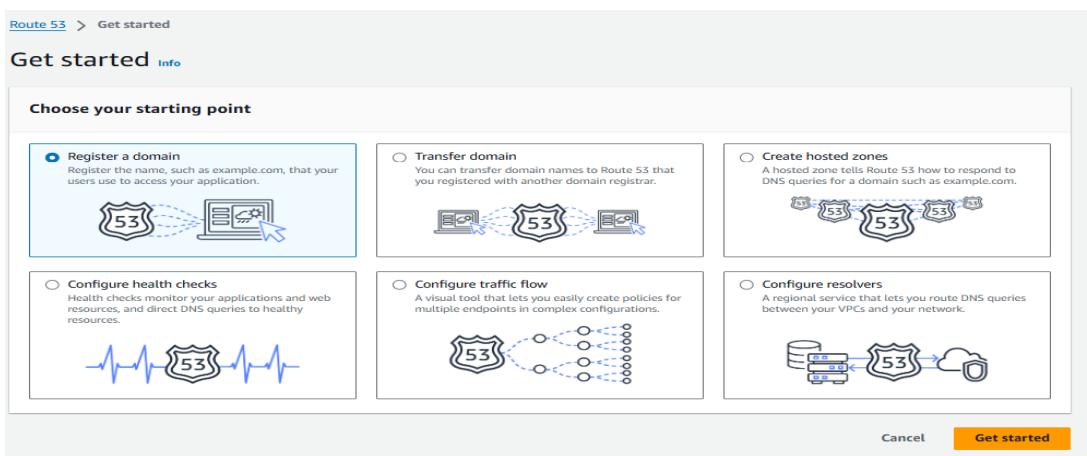


Figure 48: choix de l'option de route 53

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

On choisit l'option relative à l'enregistrement d'un nom de domaine ensuite on clique sur « Get started »

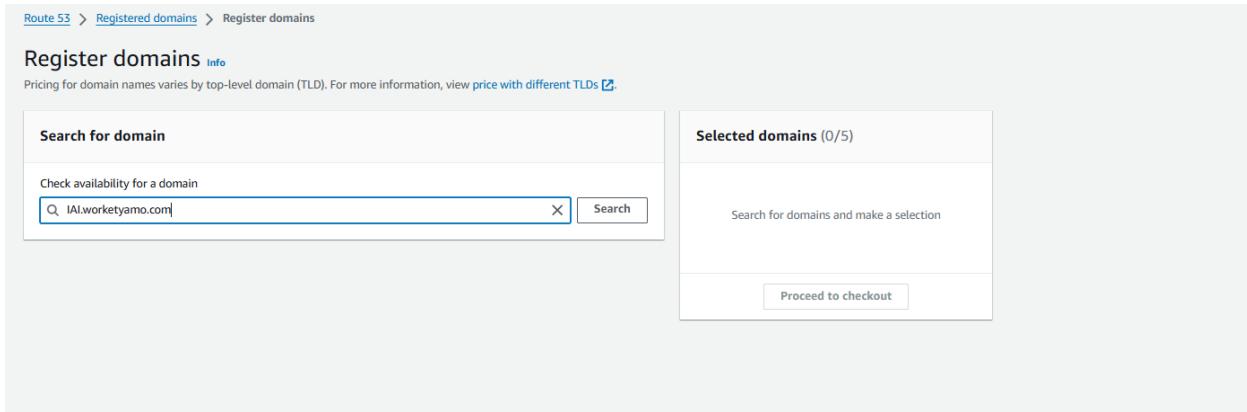


Figure 49: enregistrement du nom de domaine

Nous enregistrons le nom de domaine ensuite nous cliquons sur search pour voir s'il est valide ensuite nous procédon à la sélection du nom de domaine.

## 7. Configuration de cloudfront

The screenshot shows the Amazon CloudFront homepage under the 'Networking &amp; Content Delivery' section. It features a main heading 'Amazon CloudFront' with the subtext 'Securely deliver content with low latency and high transfer speeds'. Below this, a paragraph describes CloudFront as a fast content delivery network (CDN) service. To the right, a 'Get started with CloudFront' box contains a 'Create a CloudFront distribution' button. The page is divided into several sections: 'Benefits and features' (with 'Reduce latency', 'Cut costs', 'Improve security', and 'Customize delivery' sections), 'Use cases' (with 'Deliver fast, secure websites', 'Accelerate dynamic content delivery and APIs', 'Stream video live and on-demand', and 'Distribute software, game patches, and IoT OTA updates' sections), and 'AWS Free Tier' and 'Pricing (US)' tables. The 'AWS Free Tier' table shows free usage limits: 1 TB of data transfer out, 10,000,000 HTTP or HTTPS requests, 2,000,000 CloudFront Function invocations, and each month, always free. The 'Pricing (US)' table shows rates for First 1 TB data transfer free each month, 10 TB/month (\$0.085 per GB), HTTP requests (\$0.0075 per 10,000), and HTTPS requests (\$0.0100 per 10,000). A 'CloudFront Security Savings Bundle' section at the bottom right details a self-service pricing plan combining CloudFront with AWS WAF to provide significant savings in exchange for a monthly spend commitment for a 1-year term.

Figure 50: page d'accueil de cloudfront

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

Pour créer une distribution CloudFront nous cliquons sur « **create a CloudFront distribution** ».

**Create distribution**

**Origin**

**Origin domain**  
Choose an AWS origin, or enter your origin's domain name.

**Origin path - optional**  
Enter a URL path to append to the origin domain name for origin requests.

**Name**  
Enter a name for this origin.

**Add custom header - optional**  
CloudFront includes this header in all requests that it sends to your origin.

**Enable Origin Shield**  
Origin shield is an additional caching layer that can help reduce the load on your origin and help protect its availability.  
 No  
 Yes

**► Additional settings**

**Default cache behavior**

**Path pattern** | [Info](#)

**Compress objects automatically** | [Info](#)  
 No  
 Yes

**Viewer**

**Viewer protocol policy**  
 HTTP and HTTPS  
 Redirect HTTP to HTTPS  
 HTTPS only

**Allowed HTTP methods**

Figure 51: configuration de cloudfront

Dans la case de l'origine du domaine entrons le nom de domaine entrer lors de la configurations de route 53ensuite nous donnons un nom à la distribution qui est « IAI-cloudfront ».

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the 'Settings' tab of a CloudFront distribution configuration page. It includes sections for Price class (with 'Info' link), Alternate domain name (CNAME) (with 'Add item' button), Custom SSL certificate (with dropdown 'Choose certificate' and 'Request certificate' button), Supported HTTP versions (checkboxes for HTTP/2 and HTTP/3, with HTTP/2 checked), Default root object (text input field), Standard logging (checkboxes for Off and On, with Off checked), IPv6 (checkboxes for Off and On, with On checked), and a Description field. At the bottom are 'Cancel' and 'Create distribution' buttons.

Figure 52: configuration de cloudfront(fin)

Pour finir la création cliquons sur « **Create Distribution** »

## 8. création et configuration de RDS

# ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

The screenshot shows the AWS RDS dashboard. On the left, a sidebar lists various RDS management options like Databases, Query Editor, and Performance insights. The main area displays 'Resources' (DB Instances, DB Clusters, Snapshots, etc.) and 'Recommended services' (No recommendations yet). A central section allows creating a new database, with a prominent 'Create database' button. Below this is a 'Service health' status bar indicating everything is operating normally.

Figure 53: dashboard de RDS

Dans la page d'accueil de RDS cliquons sur « create database »

This screenshot shows the 'Create database' wizard. It starts with a choice between 'Standard create' (which allows setting all configuration options) and 'Easy create' (which uses recommended best-practices). The 'Easy create' option is selected. The next step, 'Configuration', allows choosing the engine type: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible) (which is selected), MySQL, MariaDB, PostgreSQL, Microsoft SQL Server, or Oracle. The 'DB instance size' dropdown is set to 'Dev/Test db.t4g.large'. Other visible options include 'Production db.r6g.2xlarge' and '8 vCPUs 64 GiB RAM'.

Figure 54: choix de la base de données

Ensuite nous faisons le choix de la base de données qui est « Aurora (postgreSQL Compatible) »

## ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS

**DB cluster identifier**  
Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.  
**database-1**

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

**Master username** [Info](#)  
Type a login ID for the master user of your DB instance.  
**postgres**

1 to 16 alphanumeric characters. The first character must be a letter.

**Credentials management**  
You can use AWS Secrets Manager or manage your master user credentials.

**Managed in AWS Secrets Manager - most secure**  
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

**Self managed**  
Create your own password or have RDS create a password that you manage.

**ⓘ If you manage the master user credentials in AWS Secrets Manager, additional charges apply. See [AWS Secrets Manager pricing](#). Additionally, some RDS features aren't supported. See limitations [here](#).**

**Select the encryption key** [Info](#)  
You can encrypt using the KMS key that Secrets Manager creates or a customer managed KMS key that you create.  
**aws/secretsmanager (default)** [▼](#)

[Add new key](#)

**► Set up EC2 connection - optional**  
You can also set up a connection to an EC2 instance after creating the database. Go to the database list page or the database details page, choose **Actions**, and then choose **Set up to EC2 connection**.

**► View default settings for Easy create**  
Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

**ⓘ You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.**

[Cancel](#) [Create database](#)

Figure 55: configuration de RDS

Le nom que nous attribuons à la base de donnée est « **database-1** » le login de connexion pour cette base de donnée est « **postgres** » ensuite nous cliquons sur « **create database** » pour terminer la configuration.

# **ETUDE ET IMPLEMENTATION DE LA SECURITE DANS LE CLOUD : CAS DE AWS**

---