The Cloud Resume Challenge – AWS

1. Création des fichiers de base :

Création du fichier README.md qui contient le descriptif et la stack technique du projet (sera mis à jour au fil de l’avancement).  
Création du dossier CV dans lequel on créé le fichier HTML et le CSS du CV en format site web statique. Une fois que l’on a une bonne base, on passe à la suite.

1. Hébergement :

On va héberger notre site sur AWS en utilisant un compartiment de stockage S3 qui est un service de stockage d'objets qui offre une capacité de mise à l'échelle, une disponibilité des données, une sécurité et des performances de pointe. Les clients de toutes les tailles et de tous les secteurs stockent, gèrent, analysent et protègent n’importe quelle quantité de données pour la quasi-totalité des cas d’utilisation.

Avant de créer notre bucket S3, on s’assure d’être bien de la bonne région géographique ; dans notre cas on choisit eu-west-3 qui correspond à Paris (localisation la plus proche de notre position). On le créé avec les paramètres suivants :

* **Nom du compartiment** : cloud-resume-challenge-aws-website
* **Propriété d'objets**:Listes ACL désactivées
* **Paramètres de blocage de l'accès public**: désactivé
* **Gestion des versions de compartiment**: désactivé
* **Balises**: (Region : eu-west-3), (But : website)
* **Chiffrement par défaut**: SSE-S3
* **Clé de compartiment**: désactivé

Maintenant il faut activer l’hébergement de site web statique dans le bucket S3. On va dans l’onglet **Propriétés**, **Hébergement de site Web statique**: activé, **Type d'hébergement** : Héberger un site Web statique, **Document d'index**: index.html.

Ensuite, on va se rendre dans la section **Autorisations** du bucket S3, **Stratégie du compartiment** 🡪 **Modifier**et on ajoute le texte JSON suivant:{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "PublicReadGetObject",

"Effect": "Allow",

"Principal": "\*",

"Action": [

"s3:GetObject"

],

"Resource": [

"arn:aws:s3:::cloud-resume-challenge-aws-website/\*"

]

}

]

}

L’action “s3:GetObject” avec Effect “Allow” permettra aux utilisateurs d’accéder à notre site web.

Il ne reste plus qu’à ajouter les fichiers nécessaires au fonctionnement de notre site web dans le bucket S3, une fois ceci fait retourner dans **Propriétés**, **Hébergement de site Web statique**et tester la connexion avec le lien endpoint du compartiment fourni**.**

Passons maintenant à la sécurisation du trafic de notre site en privilégiant le protocole HTTPS plutôt que le simple HTTP. Pour cette partie, nous allons utiliser **Amazon Cloudfront** qui est le service de réseaux de diffusion de contenu (CDN) rapide qui fournit en toute sécurité des données, des vidéos, des applications et des API aux clients du monde entier avec une faible latence et des vitesses de transfert élevées.

On créé la distribution Cloudfront avec les infos suivantes :

* **Distribution options**: single website or app
* **Origin domain :** cloud-resume-challenge-aws-website.s3-website.eu-west-3.amazonaws.com (adresse de l’endpoint S3 de notre site web statique)
* **Politique de protocole d'utilisateur**: Redirect HTTP to HTTPS

On laisse tout le reste par défaut et on créé la distribution. Une fois tout ceci fait, on peut utiliser le **Nom de domaine de distribution** pour accéder à notre site web CV de façon plus sécurisée.

Maintenant, nous allons réellement sécuriser l’accès à notre site web via la distribution CDN Cloudfront. Dans notre distribution, allons dans la section **Origines** 🡪 **Modifier**Au lieu d’utiliser l’adresse exacte de l’endpoint du site web S3, on va plutôt utiliser l’adresse du compartiment S3, ainsi pour **Accès d'origine** on sélectionne **Origin access control settings (OAC)**.

On créé un nouvel **OAC**, on laisse le nom par défaut et on laisse bien **Demandes de signature** comme recommandé. Ensuite, on copie la stratégie de compartiment donnée par AWS et on vient remplacer notre précédente politique d’accès dans la section **Autorisations** 🡪 **Stratégie du compartiment** 🡪 **Modifier**.

Maintenant, l’accès se fera via la distribution qui fera la requête au compartiment S3 pour l’utilisateur ; ce ne sera plus un accès direct au contenu du bucket S3 par les utilisateurs finaux.

Enfin, on va modifier le chiffrement par défaut de notre compartiment en choisissant **Chiffrement côté serveur avec des clés AWS Key Management Service (SSE-KMS) 🡪 Choisir parmi vos clés AWS KMS 🡪 aws/s3 (clé par défaut)**. Ce choix nous fournit un suivi d'audit indiquant quand votre clé CMK a été utilisée et par qui ; et la possibilité de créer nous même notre clé de chiffrement avec **AWS KMS**.

Nous allons maintenant nous attaquer à l’attribution d’un nom de domaine DNS personnalisé pour notre CV, nous utiliserons **Amazon Route 53**. Je choisis d’utiliser ilian-bonsens-cloud-resume.com qui est disponible à 14.00 USD pour un an.

Une fois le nom de domaine créé, enregistré et propagé ; on le retrouve bien dans notre **Route 53** et on peut l’utiliser dans notre distribution **Cloudfront**. Pour l’ajouter, on tape simplement notre nom de domaine dans **Alternative domain name (CNAMEs)**. On aura aussi besoin d’un certificat TLS/SSL public via **AWS Cert Manager**. Une fois celui-ci demandé et validé par **Cert Manager** avec notre nom de domaine précédemment créé, peut le sélectionner dans **Cloudfront**.