



Aujourd'hui nous allons évoquer **Azure Container Registry (ACR)**. C'est un registre, qui permet de stocker des images de conteneurs.

Comme **Docker Hub** ?

Exactement.

D'ailleurs ACR est basé sur le registre **Docker** open source 2.0

Avec les différentes solutions de conteneurisation offertes par Azure, c'est une excellente chose de proposer une offre de registre.

Oui, d'autant plus que tu peux t'en servir avec des services comme **Azure Kubernetes Service**, ou **Azure Container Apps** pour ne citer qu'eux.

Tu peux aussi t'en servir avec des orchestrateurs déployés localement comme **Kubernetes** ou **Docker Swarm**.

C'est génial !

ACR permet de stocker des **images** et leurs **artéfacts**, c'est à dire les fichiers, librairies, ..., utiles à leur fonctionnement.

En plus,



J'ai aussi entendu parlé de ACR Tasks ?!

ACR Tasks permet

de **générer des images** lorsque le code a été validé sur GitHub,

de **recréer des images** automatiquement lorsque les images de bases sont à jour,

mais aussi de **créer des tâches spécifiques**, comme pour automatiser la création, le test et la correction des images.

Cool, cela permet ainsi de **gérer le cycle de vie** de tes images.

Aujourd'hui, ACR est disponible en 3 niveaux de service que sont de **Base**, **Standard** et **Premium**.



Et comment cela marche exactement ?

C'est très simple, comme toujours !

Tu déploies un nouveau registre, tu définis le type d'accès réseau que tu souhaites et c'est tout !

C'est tout ?

Bien sûr, lorsque tu pousses ton image, il faut définir un **Repository**, qui est un dépôt dans lequel ton image sera stockée, ainsi qu'un **tag** afin de versionner ton image.

Oui !

Effectivement, c'est plutôt simple comme fonctionnement.

Hum, une autre question,

Si je possède un **ACR dans plusieurs régions**, il faudra que je pousse mon image dans chacun d'eux ?

Non, ne t'inquiètes pas, l'équipe produit a pensé à tout.

ACR propose une option de **géoreplication** qui se charge de synchroniser tes images, entre les différentes régions que tu as configuré.

Génial,

Mais comment cela se passe si je souhaite déployer dans une région où mon image n'est pas encore synchronisée ?

Excellente question !

Grâce au systèmes de **Webhooks**.

Des Webhooks ?

C'est ce qui permet de **déclencher une action** suite à un événement.

Un Webhook se charge de réaliser le **docker push**, c'est à dire de pousser l'image dans la ou les régions secondaires.

Ensuite, il notifie ton application, qu'une nouvelle image est disponible à être déployée.

Excellent.



Et en terme de sécurité ?

Alors, tu peux exposer ou non ton ACR sur internet, mais aussi le consommer via un **Private Endpoint**.

C'est cool. Et en terme de protection des images ?

Elles sont évidemment **chiffrées au repos**.

Soit en utilisant ta propre clé de chiffrement, ou en utilisant celle fournie par Microsoft.

Et je suppose qu'elles sont chiffrées en **transit** via le protocole **HTTPS**.

Forcément !

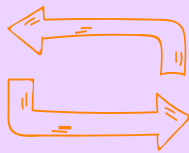
L'authentification au registre peut se réaliser par une **combinaison login/mot de passe**, ou via le **RBAC**, ce qui est très utile lorsque le déploiement est automatisé avec des **Workflows CI/CD**.



Hum, je n'y avais pas pensé.

Tu peux également protéger ACR avec **Microsoft Defender for Cloud**.

Cela te permettra, notamment, d'analyser tes images à la recherche de **vulnérabilités** de sécurité.



C'est Top, car on peut parfois oublier le **cycle de vie** et de **mises à jour** des images existantes.

Excellente remarque.

ACR est l'allié parfait, lorsque tu utilises une solution de conteneurs dans Azure.

Cela te permet de tout **centraliser** en un même endroit, ce qui **simplifie** grandement la **gestion** et la **sécurité**.



Si vous souhaitez continuer à **apprendre**, de façon ludique, sur **l'écosystème Azure**, et ne rater aucune de nos illustrations ...

... N'hésitez pas à vous abonner sur :



<https://aka.ms/grow-una>



<https://www.youtube.com/@grow-una>

Et si le contenu vous plaît, partagez-le ;o)

A très vite !

