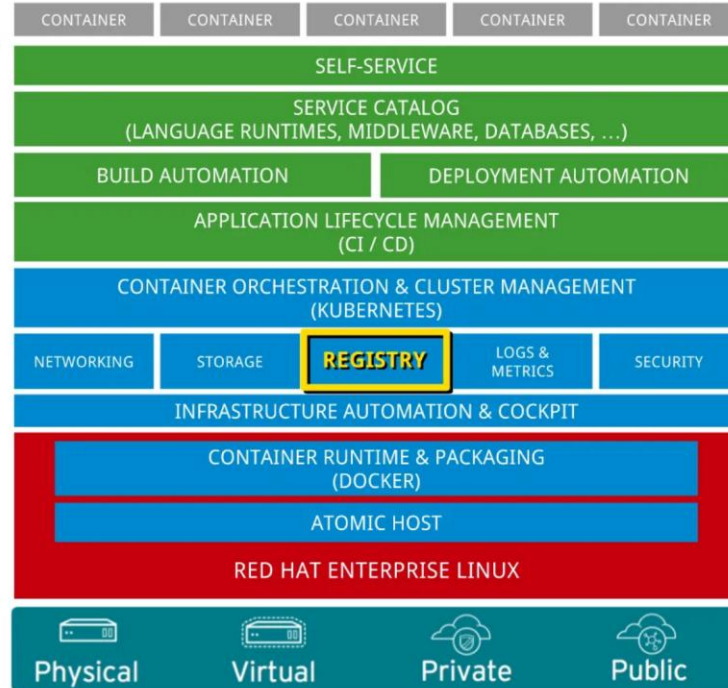


# Module 3

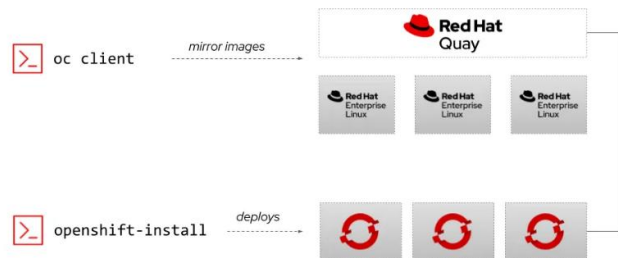
## Publication d'images de conteneurs d'entreprise



# Objectifs du module

- ✓ Comprendre le concept de **registre d'entreprise** et son rôle dans l'écosystème des conteneurs
- ✓ Identifier les différents types de registres et leurs caractéristiques spécifiques
- ✓ Maîtriser les mécanismes d'**autorisations d'accès** au registre OpenShift
- ✓ Configurer et gérer les politiques de sécurité pour les images de conteneurs
- ✓ Mettre en place un registre d'entreprise et configurer les autorisations d'accès

À la fin de ce module, vous serez capable de publier et gérer des images de conteneurs dans un environnement d'entreprise sécurisé.



# Notion de registre d'entreprise

## Qu'est-ce qu'un registre de conteneurs ?

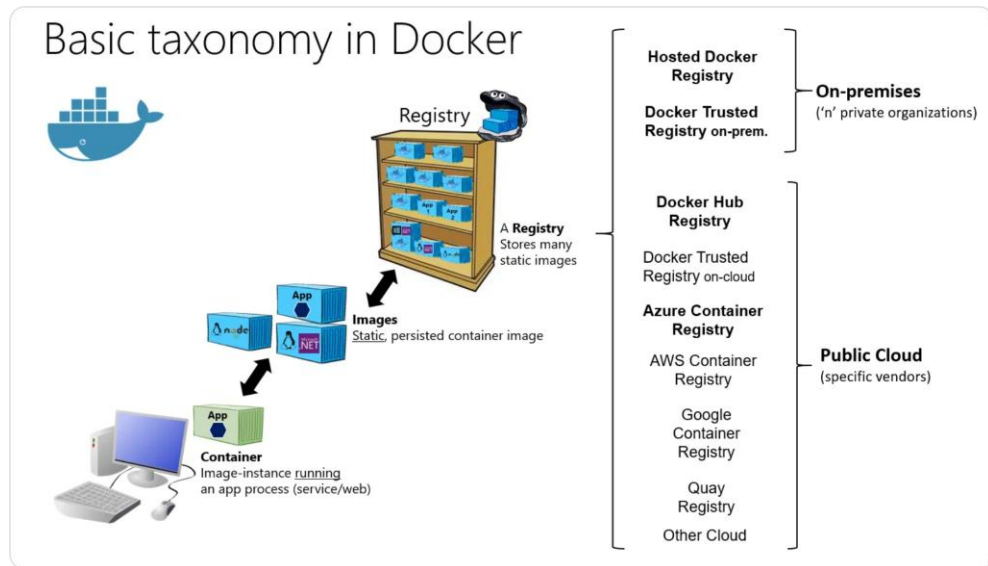
Un **registre de conteneurs** est un service de stockage et de distribution d'images de conteneurs. Il permet de :

- Centraliser le stockage des images
- Versionner les images de conteneurs
- Partager les images entre équipes
- Sécuriser l'accès aux images

## Rôle dans l'écosystème des conteneurs

Le registre est un composant **essentiel** dans le cycle de vie des applications conteneurisées, servant de pont entre les environnements de développement et de production.

Dans un contexte d'entreprise, un registre privé offre des fonctionnalités avancées de sécurité, de gouvernance et d'intégration avec les processus existants.



# Types de registres

## Registres publics

Services accessibles publiquement pour stocker et partager des images de conteneurs avec la communauté mondiale.

Accès ouvert (avec authentification)

Hébergement externe

Idéal pour projets open source



Docker Hub



Quay.io

## Registres privés d'entreprise

Services déployés et gérés au sein de l'infrastructure de l'entreprise avec des contrôles d'accès stricts.

Accès restreint et sécurisé

Hébergement interne ou cloud privé

Intégration avec les outils d'entreprise



OpenShift Registry



Harbor

Critère	Registre public	Registre privé d'entreprise
Sécurité	Basique	<b>Avancée</b> (analyse de vulnérabilités, signatures)
Contrôle d'accès	Limité	<b>Granulaire</b> (RBAC, intégration LDAP/AD)
Performance	Variable (dépend de la connexion internet)	<b>Optimisée</b> pour le réseau interne

# Avantages des registres d'entreprise



## Sécurité renforcée

Contrôle d'accès granulaire, analyse de vulnérabilités et signatures d'images pour garantir l'intégrité.



## Performance optimisée

Réduction de la latence réseau, mise en cache locale et distribution géographique des images.



## Gouvernance et conformité

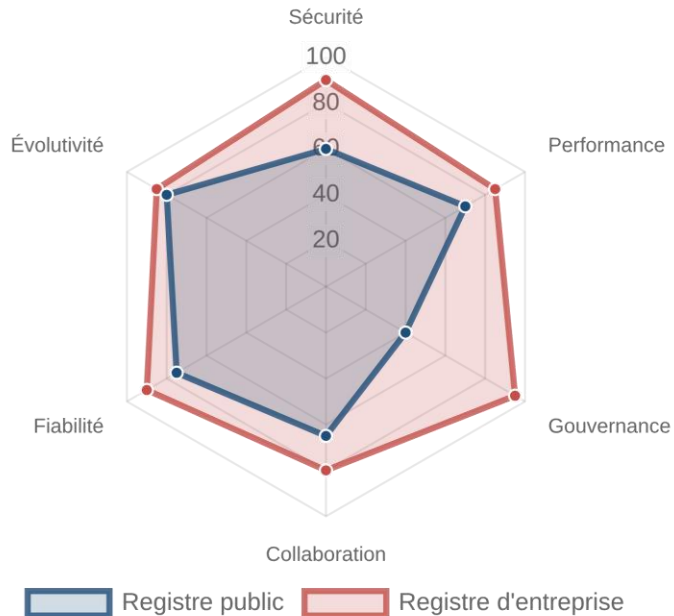
Traçabilité complète, politiques d'approbation et respect des normes de l'entreprise.



## Collaboration améliorée






Partage sécurisé entre équipes, standardisation des images et réutilisation des composants.

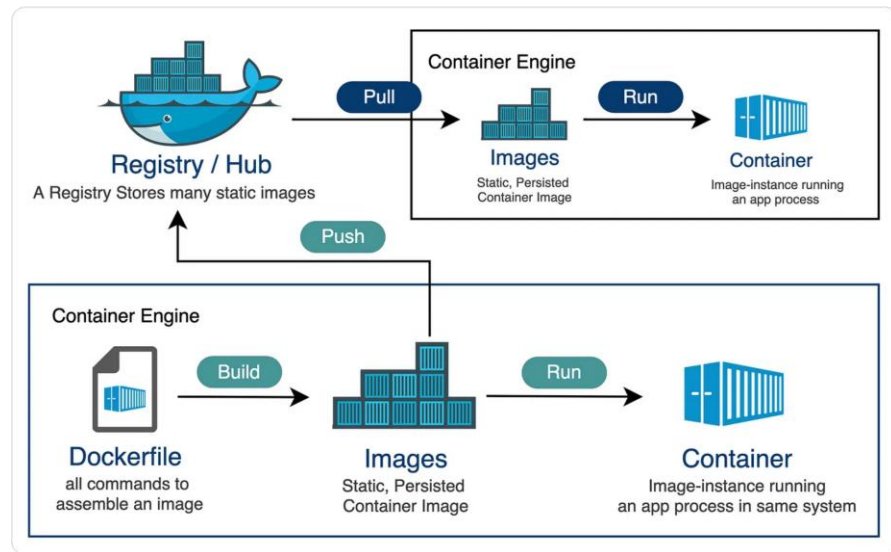
## Comparaison des capacités



# Architecture d'un registre d'entreprise

## Composants principaux

-  **Stockage** : Système de stockage persistant pour les images et métadonnées
-  **Authentification** : Système de gestion des identités et accès
-  **API** : Interface pour les opérations push/pull
-  **Indexation** : Catalogue et recherche d'images
-  **Gestion** : Interface d'administration et monitoring



## Intégration avec OpenShift

OpenShift intègre nativement un registre interne et peut se connecter à des registres externes via des **secrets d'authentification** et des **imagestreams** pour faciliter le déploiement continu.

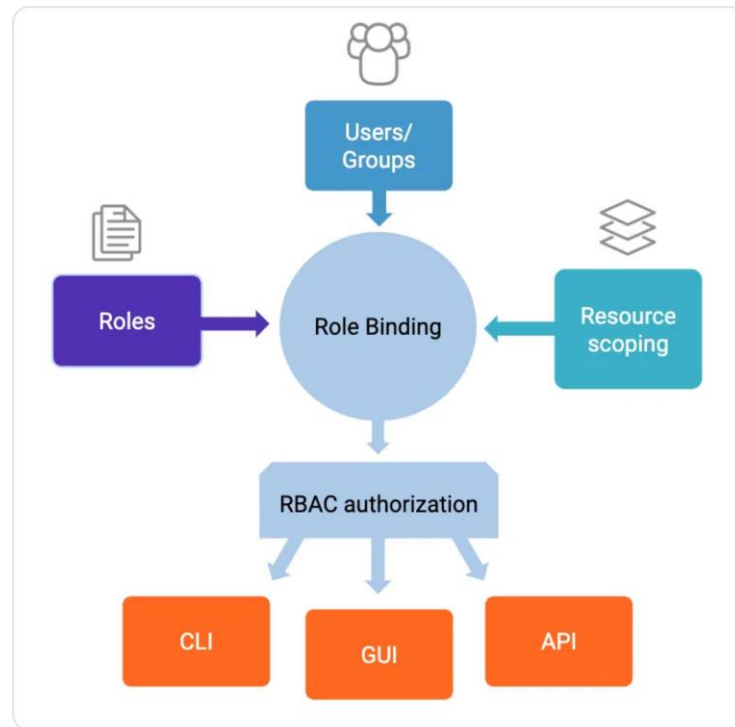
# Autorisations d'accès au registre OpenShift

## Importance de la sécurité des registres

Les registres d'images contiennent des **actifs critiques** pour l'entreprise et nécessitent des mécanismes d'autorisation robustes.

## Mécanismes d'autorisation dans OpenShift

- 🛡️ Contrôle d'accès basé sur les rôles (RBAC)
- 👤 Gestion des utilisateurs et groupes
- 🔑 Secrets et tokens d'authentification
- 🔒 Politiques de sécurité des images



OpenShift intègre des mécanismes de sécurité avancés permettant de contrôler précisément **qui** peut accéder à **quelles images** et avec **quelles permissions** (pull, push, admin).

# RBAC (Role-Based Access Control)

## Principes du RBAC dans OpenShift

Le **RBAC** est un mécanisme de contrôle d'accès basé sur les rôles qui permet de définir précisément qui peut faire quoi dans le cluster OpenShift, y compris l'accès aux registres.

### Rôles

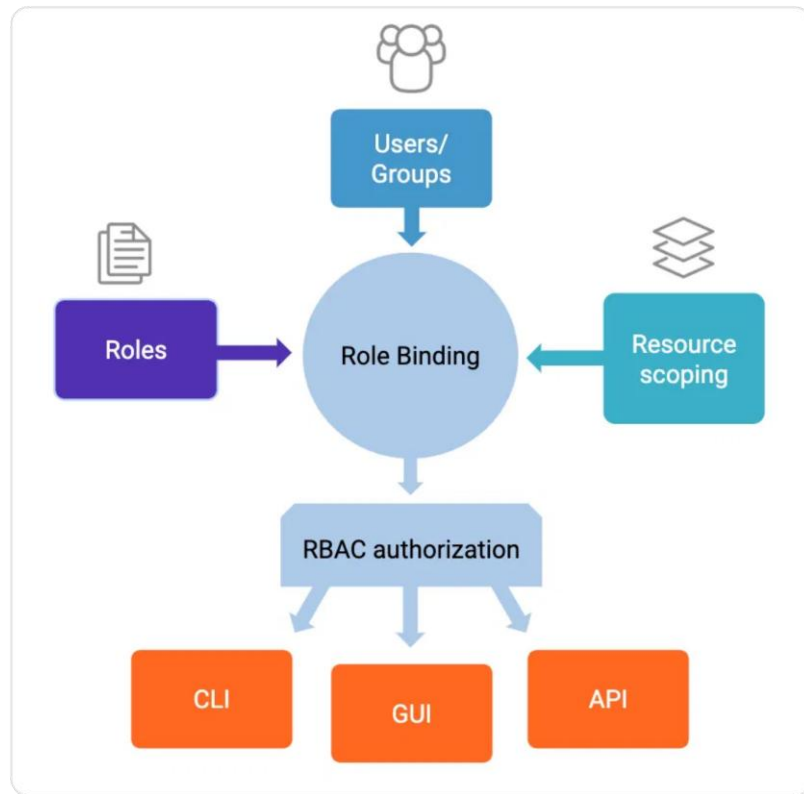
Ensemble de règles définissant les actions autorisées sur des ressources

### RoleBindings

Association entre des utilisateurs/groupe et des rôles

### ClusterRoles / ClusterRoleBindings

Équivalents des rôles et rolebindings mais à l'échelle du cluster



## Application au registre d'images




Dans le contexte du registre d'images, le RBAC permet de contrôler qui peut **pousser** (push) ou **tirer** (pull) des images, ainsi que qui peut gérer les configurations du registre.



# Gestion des utilisateurs et groupes

## Utilisateurs et groupes dans OpenShift

OpenShift permet de gérer finement les accès au registre d'images via la gestion des **utilisateurs** et des **groupes**.

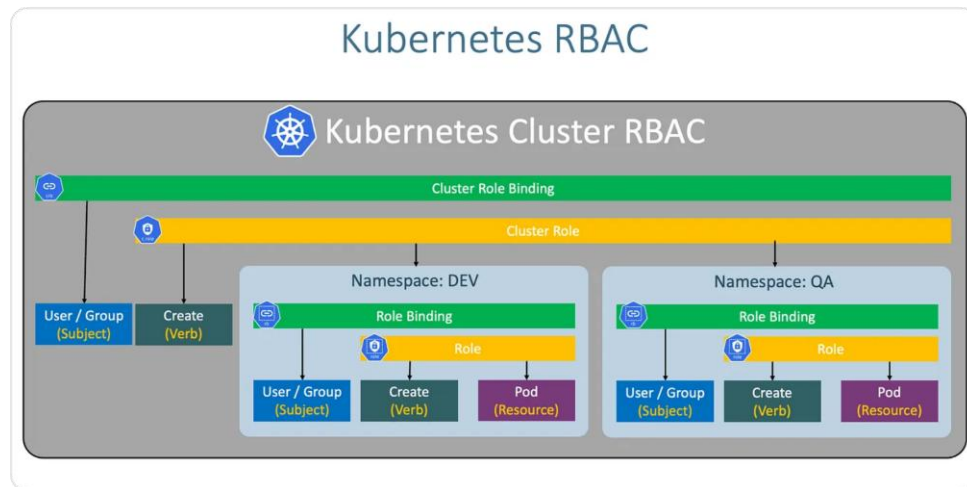
-  **Utilisateurs** : Entités individuelles avec des identifiants uniques
-  **Groupes** : Collections d'utilisateurs pour faciliter la gestion des droits
-  **ServiceAccounts** : Identités techniques pour les processus automatisés

## Commandes de gestion

```
# Créer un utilisateur  
oc create user developer
```

```
# Créer un groupe et y ajouter des utilisateurs  
oc adm groups new registry-admins  
oc adm groups add-users registry-admins developer
```

OpenShift peut s'intégrer avec des fournisseurs d'identité externes comme **LDAP**, **Active Directory** ou **Oauth** pour synchroniser les utilisateurs et groupes.



# Politiques de sécurité des images

## Mécanismes de sécurité des images



### Analyse de vulnérabilités

Détection automatique des failles de sécurité connues dans les images



### Signature d'images

Vérification de l'intégrité et de l'authenticité des images



### Politiques d'admission

Contrôle des images autorisées à être déployées



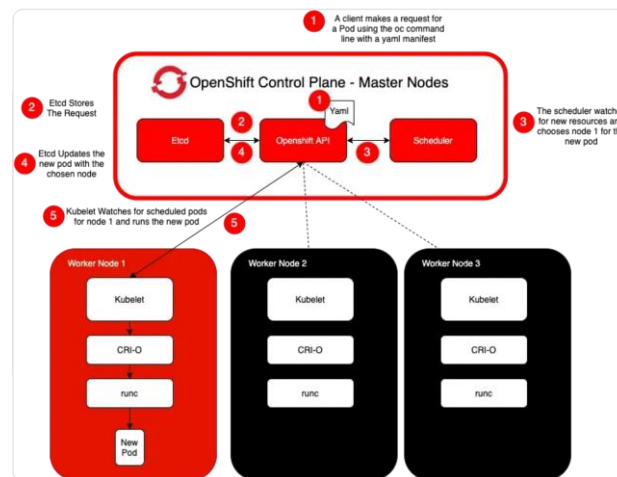
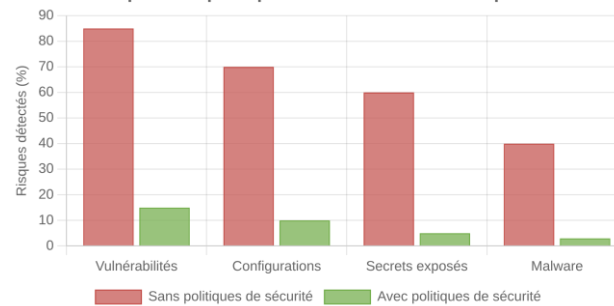
### Gestion des versions

Contrôle des versions d'images autorisées

## Security Context Constraints (SCC)

Les **SCC** dans OpenShift permettent de contrôler les privilèges avec lesquels les conteneurs peuvent s'exécuter, ajoutant une couche de sécurité supplémentaire au-delà des politiques d'images.

Impact des politiques de sécurité sur les risques



# Configuration des secrets et tokens

## Secrets dans OpenShift

Les **secrets** sont des objets Kubernetes qui stockent des informations sensibles comme des mots de passe, des tokens OAuth et des clés SSH.



### docker-registry

Authentification auprès des registres d'images externes



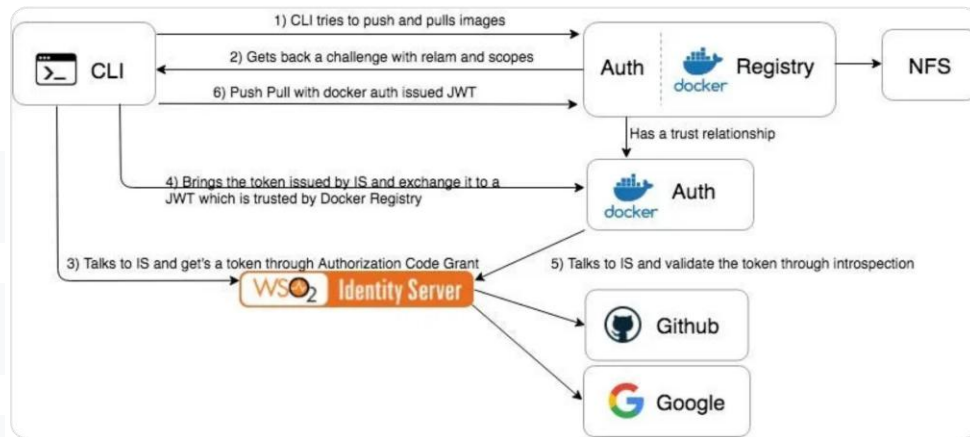
### basic-auth

Authentification par nom d'utilisateur et mot de passe



### tls

Certificats pour les connexions sécurisées



## Exemple de création d'un secret pour un registre

```
$ oc create secret docker-registry my-registry-secret \
--docker-server=registry.example.com \
--docker-username=username \
--docker-password=password
```

# Bonnes pratiques de sécurité

## Sécurisation des registres d'entreprise



**Principe du moindre privilège** : Attribuer uniquement les permissions nécessaires aux utilisateurs et services



**Authentification multi-facteurs** : Activer l'authentification à deux facteurs pour les comptes administrateurs



**Analyse des vulnérabilités** : Scanner automatiquement les images pour détecter les failles de sécurité



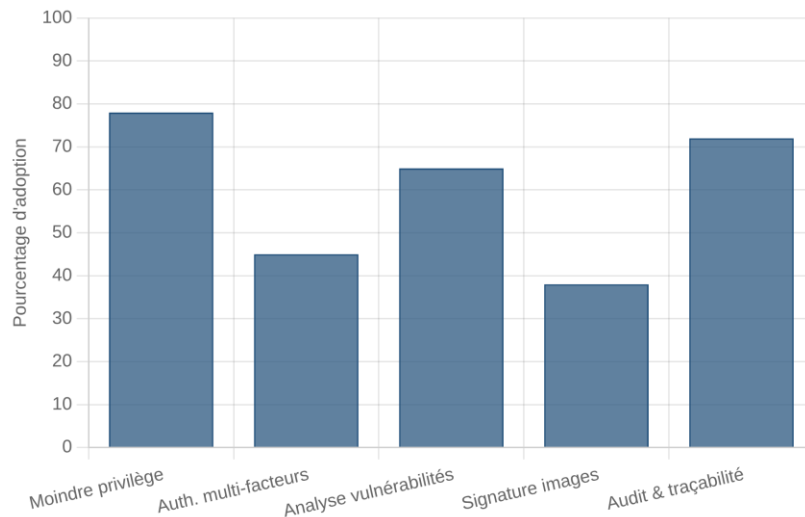
**Signature des images** : Mettre en place un système de signature cryptographique des images



**Audit et traçabilité** : Enregistrer toutes les actions effectuées sur le registre

## Recommandation clé

Mettre en place une **stratégie de sécurité en profondeur** combinant plusieurs couches de protection pour sécuriser efficacement vos registres d'images et vos déploiements OpenShift.



Niveau d'adoption des pratiques de sécurité dans les entreprises

# Introduction aux travaux pratiques

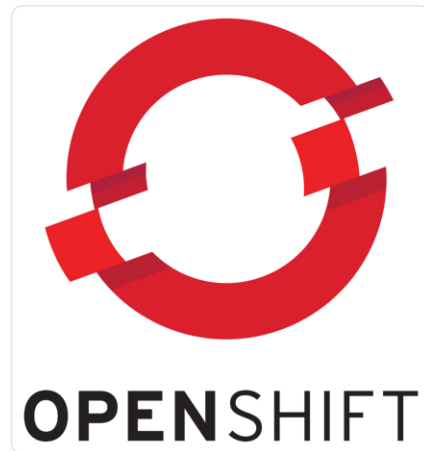
## Mise en pratique des concepts

Nous allons mettre en pratique les concepts vus dans ce module à travers un exercice complet sur la **création d'un registre d'entreprise** et la configuration des **autorisations d'accès**

## Objectifs de l'exercice

- ✓ Déployer un registre privé dans OpenShift
- ✓ Configurer les autorisations d'accès avec RBAC
- ✓ Publier et gérer des images dans le registre

**Prérequis :** Accès à un cluster OpenShift avec droits d'administration et outils CLI (oc) installés.



# Exercice pratique

## Création d'un registre d'entreprise

### 1 Déployer un registre privé dans OpenShift

```
oc new-project registry
oc new-app --name=registry \
-e REGISTRY_HTTP_SECRET=mysecret \
registry.redhat.io/rhel8/registry-proxy:latest
```

### 2 Exposer le registre avec une route sécurisée

```
oc create route edge --service=registry \
--hostname=registry.apps.example.com
```

### 3 Configurer le stockage persistant

```
oc set volume deployment/registry --add \
--name=registry-storage -t pvc \
--claim-size=10Gi --overwrite
```

## Configuration des autorisations d'accès

### 4 Créer un groupe et des utilisateurs

```
oc adm groups new registry-admins
oc adm groups add-users registry-admins developer1
oc adm groups new registry-users
oc adm groups add-users registry-users developer2
```

### 5 Configurer les rôles RBAC

```
oc policy add-role-to-group admin registry-admins -n registry
oc policy add-role-to-group view registry-users -n registry
```

### 6 Tester l'accès au registre


```
podman login -u developer1 -p $(oc whoami -t) \
registry.apps.example.com
podman pull busybox
podman tag busybox registry.apps.example.com/registry/busybox
podman push registry.apps.example.com/registry/busybox
```


### Validation de l'exercice


Vérifiez que vous pouvez pousser et tirer des images depuis votre registre privé avec les différents utilisateurs selon leurs autorisations. Confirmez que les politiques RBAC sont correctement appliquées en testant l'accès avec un utilisateur non autorisé.

# Conclusion et récapitulatif

## Points clés du module

 Les **registres d'entreprise** sont essentiels pour stocker, gérer et distribuer les images de conteneurs de manière sécurisée.

 Le **RBAC** permet de contrôler finement qui peut accéder aux images et avec quelles permissions.

 Les **secrets et tokens** permettent une authentification sécurisée aux registres.

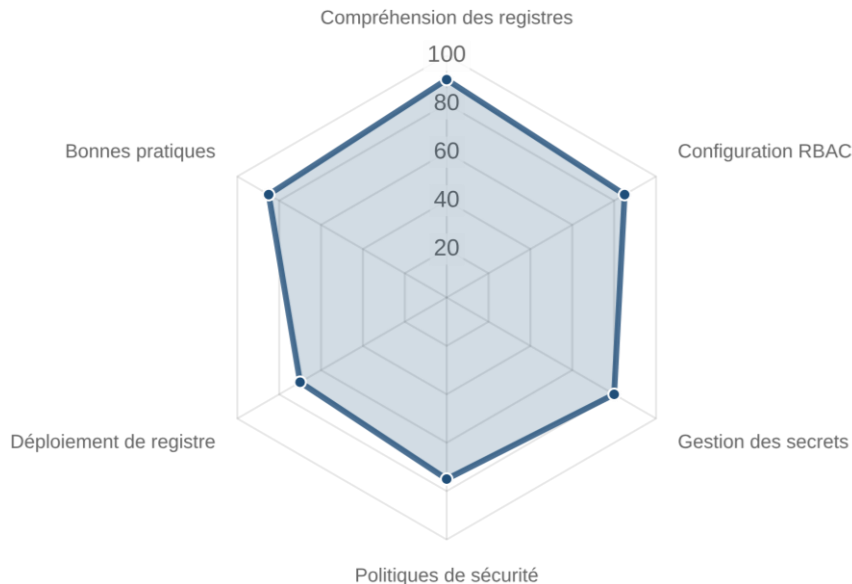
 Les **politiques de sécurité** des images renforcent la protection de votre chaîne de déploiement.

### Prochaines étapes

Dans le prochain module, nous explorerons les stratégies avancées de déploiement et la gestion des applications sur OpenShift.

## Questions ?

N'hésitez pas à poser vos questions sur les concepts abordés dans ce module ou sur l'exercice pratique à venir.



Compétences acquises dans ce module