

HONORIS UNITED UNIVERSITIES







Chapitre 7 : Nexus

ESPRIT – UP ASI (Architecture des Systèmes d'Information) Bureau E204









Plan du cours



- Introduction
- Nexus : Définition et caractéristiques
- Release vs Snapshot
- Installation de Nexus
- Configuration de Nexus en standalone
- Configuration de Nexus avec Jenkins
- Travail à faire (Automatiser le déploiement des artéfacts en utilisant Jenkins)



Introduction



 Une fois les différents modules de l'application implémentés et testés convenablement en respectant les bonnes pratiques de développement, il est temps de déployer l'application dans les environnement adéquats.

 L'objectif dans notre chaîne DevOps est d'automatiser la gestion des livrables et l'alimentation automatique des différents environnements par les artéfacts correspondants.



Introduction



- Le développement de logiciels est un processus complexe qui implique de nombreux composants, dépendances et artefacts. Gérer efficacement ces éléments est essentiel pour garantir le succès d'un projet logiciel.
- C'est là qu'intervient le Gestionnaire de Dépôts Nexus de sonatype.





Nexus - Définition



- Nexus est une plateforme de gestion de dépôts, principalement utilisée pour héberger des artefacts.
- Ces artefacts sont des composants générés lors de la construction d'un projet, tels que des fichiers JAR, WAR, et autres, qui sont ensuite stockés dans Nexus à l'aide de l'outil Maven.



Nexus est fortement intégré à Maven pour la gestion des dépendances et le stockage de ces artefacts.



Nexus - La plateforme Sonatype



Nexus Repository

Your single source of truth for software artifacts.

Lifecycle

Automate open source governance across the entire SDLC.

Container

Protect your containers with vulnerability scanning, compliance, and remediation at runtime.



Auditor

Know the quality of open source inside your production apps.







Repository Firewall

Stop open source risk at the front door.





Vulnerability Scanner

Is your app hackable? Get a free scan and find out: sonatype.com/appscan



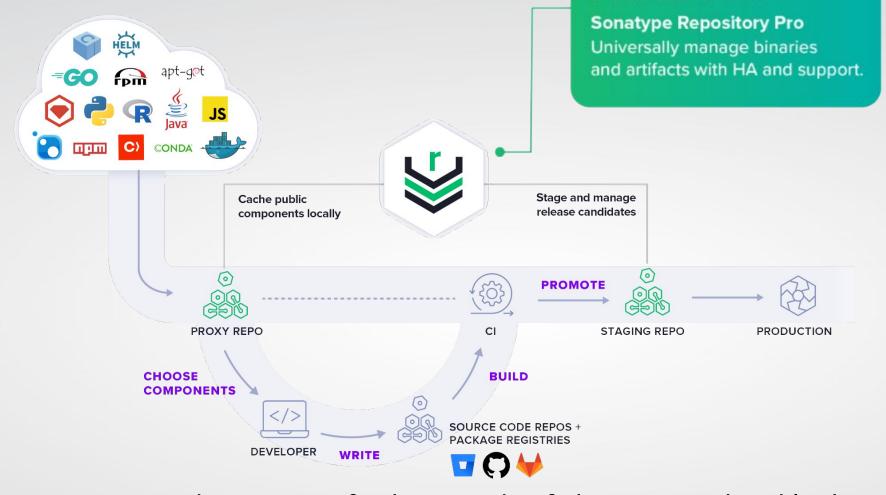


Nexus - Définition

Sonatype Repository

Universally manage binaries and artifacts for FREE.





Nexus permet de partager facilement des fichiers entre les développeurs d'un projet ou entre différents environnements comme Docker.



Nexus - Caractéristiques



- > Stockage centralisé: Nexus permet de centraliser le stockage de tous les composants et artefacts nécessaires pour un projet logiciel. Cela évite d'avoir des dépendances dispersées sur différents systèmes.
- Gestion des dépendances: Il facilite la gestion des dépendances en s'assurant que les versions correctes des bibliothèques et des composants sont disponibles pour les développeurs.







Nexus - Caractéristiques



- Distribution sécurisée : Nexus assure la distribution sécurisée des dépendances aux équipes de développement. Cela garantit que les composants sont fiables et exempts de vulnérabilités.
- Intégration avec les outils DevOps : Il s'intègre souvent dans les pipelines DevOps, ce qui permet d'automatiser le téléchargement, le déploiement et la gestion des dépendances.
- Amélioration des performances : Il peut contribuer à accélérer les temps de construction et de déploiement des applications en offrant un accès rapide aux composants nécessaires.







Nexus - Dépôts



Les dépôts dans Nexus se réfèrent à des emplacements de stockage centralisés où vous pouvez conserver des artefacts, des composants logiciels, des bibliothèques et d'autres éléments nécessaires à votre projet. Nexus Repository Manager est conçu pour gérer ces dépôts.





Nexus - Dépôts



- Hosted: Les dépôts hébergés dans Nexus sont des espaces de stockage qui contiennent les dépôts créés par les utilisateurs. Par défaut, Nexus est configuré avec les types de dépôts hébergés suivants:
 - 3rd Party: Pour stocker des bibliothèques externes qui ne se trouvent pas dans les dépôts Maven publics.
 - Releases: Pour les bibliothèques stables de votre organisation.
 - Snapshots: Pour les bibliothèques en cours de développement.



Nexus - Dépôts



- Proxy: Les dépôts proxy agissent comme un relais pour accéder à des artefacts externes, tels que des bibliothèques open source provenant de dépôts publics sur Internet. Nexus peut les mettre en cache localement pour accélérer le téléchargement.
- Group: Les dépôts de groupe permettent de regrouper plusieurs dépôts, ce qui facilite la recherche et l'accès à des artefacts à partir de sources multiples.
- → Nous allons opter pour l'utilisation des dépôts hébergés "Hosted" pour nos projets.



Versions- Release vs Snapshot



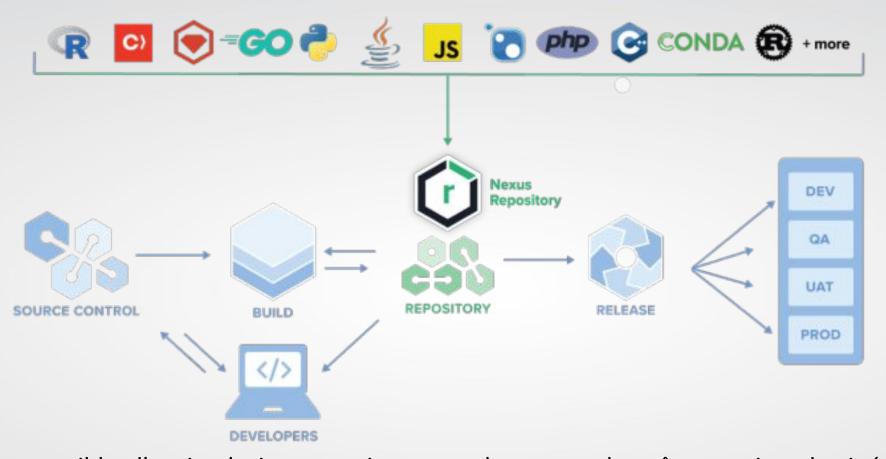
> Une <u>release</u> est une version stable et figée d'un projet, englobant toutes les fonctionnalités requises pour une itération donnée (ou sprint, dans le contexte de Scrum).

> Une <u>version snapshot</u> est une version en cours de développement, comprenant seulement certaines des fonctionnalités à implémenter.



Versions- Release vs Snapshot





Il est possible d'avoir plusieurs versions snapshots pour le même projet, destinées à divers usages, tels que la livraison à l'équipe de test, la mise en place d'une solution en cours de validation, et plus encore.



Nexus - Installation



1. <u>Téléchargez l'image Docker de nexus</u>:

2023-2024 - Module DevOps - Introduction

- > Connectez-vous à votre machine virtuelle Ubuntu en utilisant un client SSH. Pour ce faire, démarrez VirtualBox, ouvrez une fenêtre PowerShell et exécutez les commandes vagrant up et vagrant ssh.
- > Assurez-vous d'avoir effectué un chmod au préalable pour éviter les problèmes de droits d'accès.
- > Utilisez la commande docker pull pour télécharger l'image nexus3 depuis le Docker Hub.

Nexus - Installation



2. <u>Exécutez le conteneur nexus</u> :

> Vous pouvez maintenant lancer un conteneur nexus en utilisant la commande docker run.

```
[vagrant@localhost ~]$ docker run -d -p 8081:8081 --name nexus sonatype/nexus3
18eefef10a943a58e02f314afc74042ae965e3f2f23eac3566bc726878d04b5b
[vagrant@localhost ~]$ docker ps
CONTAINER ID
                                        COMMAND
                                                                 CREATED
                                                                                     STATUS
                                                                                                          PORTS
                    IMAGE
                                                                                                                                   NAMES
18eefef10a94
                   sonatype/nexus3
                                        "/opt/sonatype/nex..." 10 seconds ago
                                                                                     Up 8 seconds
                                                                                                          0.0.0.0:8081->8081/tcp
                                                                                                                                   nexus
[vagrant@localhost ~]$ _
```

- -d: Lance le conteneur en en arrière-plan
- --name sonarqube: Attribue un nom au conteneur, ici "nexus"
- -p 8081:8081 : Le port 8081 du conteneur est mappé sur le port 8081 de votre hôte.



ATTENTION: Choisissez un autre port, si vous utilisez déjà le port 8081 pour Jenkins.



Nexus - Installation



3. Attendez que le conteneur Nexus 3 soit entièrement démarré :

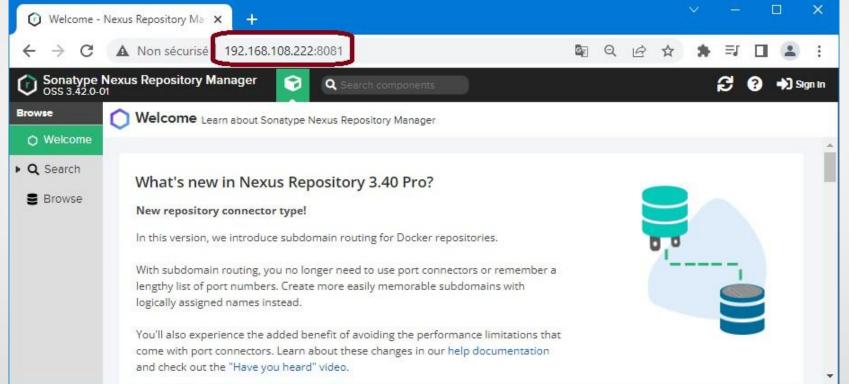
> Vous pouvez vérifier les journaux du conteneur pour voir quand il est prêt. Vous pouvez utiliser la commande : docker logs -f nexus

Nexus - Utilisation





- 4. Accédez à l'interface web de Nexus Repository Manager :
- **4.1.** Vous pouvez maintenant accéder à l'interface web de nexus repository manager en ouvrant votre navigateur et en visitant <a href="http://<adresse-ip-vm>:8081">http://<adresse-ip-vm>:8081.



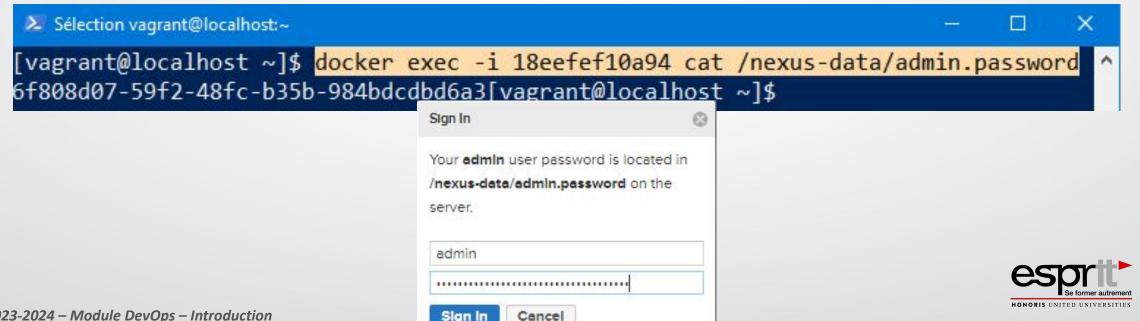


Nexus - Configuration nexus repository manager has been sometimed by sonature





- **4.2.** Vous devrez vous connecter avec les informations d'identification suivantes:
- > username : admin
- password : Accédez au conteneur en utilisant la commande docker exec <id conteneur> et affichez le fichier indiqué dans l'interface d'authentification en utilisant la commande cat.



Nexus - Configuration pexus repository manager by sonatype





4.3. Modifiez le mot de passe pour nexus, par exemple, et autorisez les accès anonymes.

Setup	1 of 4
his wizard will help you complete required setup tasks.	
	Next

Configure Anonymous Access	3 of 6
Enable anonymous access means that by default, users can sear	ch, browse and download
components from repositories without credentials. Please consid e	er the security implications
for your organization.	
Disable anonymous access should be chosen with care, as it will	require credentials for all
users and/or build tools.	
More information	
Enable anonymous access	
Disable anonymous access	



Complete	4 of 4
The setup tasks have been completed, enjoy using Nexus Repository Manager!	
	Finish



Nexus - Configuration nexus repository manager by sonatype





- Pour déployer notre application sur Nexus, qui repose sur Maven pour la construction du projet, modifiez le fichier settings.xml situé dans le dossier conf de Maven (préalablement installé).
- > Vous pouvez le modifier en utilisant un éditeur tel que vi, par exemple :

vagrant@localhost:~ [vagrant@localhost ~]\$ sudo vi /usr/share/maven/conf/settings.xml_



Nexus - setting.xml





 Supprimez les commentaires de la section associée au serveur Nexus (et modifiez les champs username et password si nécessaire).

```
vagrant@localhost:~
   <server>
     <id>deploymentRepo</id>
     <username>admin</username>
     <password>nexus</password>
   </server>
   <!-- Another sample, using keys to authenticate.
   <server>
     <id>siteServer</id>
     <privateKey>/path/to/private/key</privateKey>
     <passphrase>optional; leave empty if not used.</passphrase>
  INSERT --
```

Nexus - pom.xml





Modifiez votre fichier pom.xml afin de le configurer de manière à ce qu'il fasse référence au dépôt Nexus "deploymentRep".

```
<!-- Deploy to Nexus -->
<distributionManagement>
 <repository>
   <id>deploymentRepo</id>
   <name>Nexus Repository</name>
   <url>http://localhost:8081/repository/maven-releases/</url>
 </repository>
</distributionManagement>
```

Configuration Nexus avec Jenkins



- L'ensemble du processus de construction du projet, ainsi que le déploiement du résultat dans Nexus, sera automatisé grâce à Jenkins.
- Cela peut inclure l'ajout d'une étape spécifique à Nexus, où la commande mvn deploy sera exécutée, avec l'option de "skip" les tests.





Configuration Nexus avec Jenkins



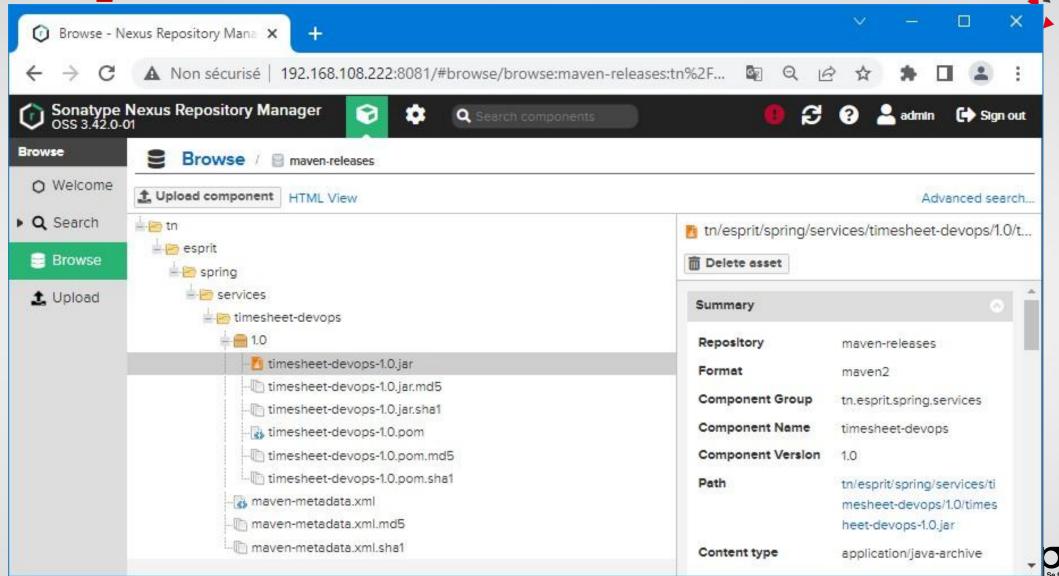
	ILING SONARQU	MOCKITO	
1s 4		252ms 252ms	13s 13s







Déploiement sur Nexus





"Apprendre par le projet, c'est découvrir

par l'action, créer par la compréhension, et réussir par la persévérance."



ESPRIT – UP ASI (Architecture des Systèmes d'Information) Bureau E204

