**Nexus Repository Manager Manual**

**목차**

[**1. 개요**](#_4h6jlzjg9gn9) **3**

[1.1 목적](#_o2gwmpz2mz7m) 3

[1.2. 문서 소개](#_5o0q7bgxpmdp) 3

[**2. 서버 환경 세팅**](#_b1ffmxf836di) **4**

[2.1 Nexus 설치](#_l27rpr1d3q8f) 4

[2.1.1 Nexus 다운로드](#_irnrbjrf4ylu) 4

[2.1.2 압축 해제](#_qqleayw6rmr9) 4

[2.2 설정값 변경](#_tictehxcthur) 4

[2.2.1 포트 변경](#_a4qbzbvnpbxu) 4

[2.2.2 관리자 비밀번호 수정](#_46upkbf6gfii) 4

[2.3 Nexus 실행](#_sbdqpnrv3vu6) 5

[**3. Nexus UI 접속 및 설정**](#_oynjhvb03brm) **6**

[3.1 브라우저에서 Nexus 접속](#_eqmumyeaykmk) 6

[3.2 관리자 계정 로그인](#_99n1rvo0qaks) 6

[3.3 역할 관리](#_qgt83sirngz) 7

[3.4 계정 관리](#_uw43hpds1zjn) 8

[3.5 저장소 관리](#_66rocbn5gyth) 9

[**4. 배포 및 사용**](#_c5tov36rojn6) **10**

[4.1 로컬 환경 셋팅](#_t7h5uqj3n9tj) 10

[4.1.1 settings.xml 위치](#_8trx7go0nzos) 10

[4.1.2 setting.xml 편집](#_f9mcy0poa6nx) 10

[4.2 deploy](#_7q7opjrt2kcx) 11

[4.3 dependency 설정](#_abundi42vfpk) 12

[**5. 유의사항**](#_ainknlcq32fr) **13**

[5.1 dependency를 업데이트 할 경우](#_fvy05e4822xq) 13

[5.2 원격 저장소 내용이 반영이 되지 않을 경우](#_ky2dvmtkccig) 13

[5.3 맞춰줘야 하는 설정값 요약](#_82o4lruba3c5) 13

[5.4 참고자료](#_q032vsn2cyts) 14

# 

# 1. 개요

본 문서는 저장소 관리자 Nexus의 기본적인 사용법에 관한 문서이며, Nexus Repository Manager3를 기반으로 작성되었습니다.

### 1.1 목적

본 문서는 maven에서 사용 가능하고 가장 널리 사용되는 무료 repository 중의 하나인 Nexus를 이용하여 local repository를 설정하는 시나리오를 소개한다.

본 문서의 목적은 Nexus를 통해 설정한 local repository를 사용함으로써 팀 내에서 사용하는 공통 라이브러리들을 local nexus에 배포해서 팀간에 쉬운 repository 연동 편의성을 갖게 한다.

### 1.2. 문서 소개

본 문서는 크게 [<Nexus 서버 구축>](#2pzqesqsou58)과 [<배포 및 사용>](#85u3a4236sz6) 두 파트로 구분되며, 이미 구축되어있는 서버를 사용하려면 [<배포 및 사용>](#85u3a4236sz6) 파트부터 보면 된다.

# 2. 서버 환경 세팅

### 2.1 Nexus 설치

#### 2.1.1 Nexus 다운로드

* <https://www.sonatype.com/download-oss-sonatype>에서 Nexus3 다운로드
* ex) wget으로 다운로드하는 경우

$ wget http://download.sonatype.com/nexus/3/latest-unix.tar.gz

* 최신 이외의 버전이 필요하면 Download Archives에서 원하는 버전 다운로드

<https://help.sonatype.com/repomanager3/download/download-archives---repository-manager-3>

#### 2.1.2 압축 해제

* 압축 파일을 해제하면 두 개의 디렉토리가 생성된다.
* nexus-<version>
* sonatype-work
* ex) tar명령어로 압축 해제
* $ tar -zxvf [파일명.tar.gz]

### 2.2 설정값 변경

#### 2.2.1 포트 변경

* <NEXUS\_PATH>/nexus-<version>/etc 디렉토리의 nexus-default.properties에서 확인 및 수정 가능

|  |
| --- |
| “nexus-default.properties”  # Jetty section  application-port=**<port>**  application-host=0.0.0.0  nexus-args=${jetty.etc}/jetty.xml,${jetty.etc}/jetty-http.xml,${jetty.etc}/jetty-requestlog.xml  nexus-context-path=/  # Nexus section  nexus-edition=nexus-oss-edition  nexus-features=\  nexus-oss-feature |

#### 2.2.2 관리자 비밀번호 수정

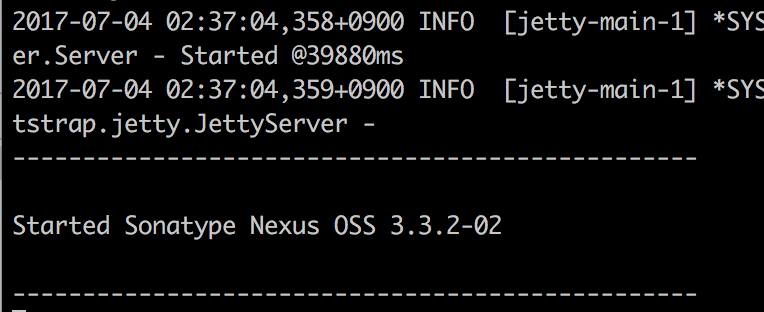
* 관리자의 ID는 admin 이며 PW는 <NEXUS\_PATH>/sonatype-work/nexus3/admin.password 에서 확인 및 수정이 가능하다.

### 2.3 Nexus 실행

* <NEXUS\_PATH>/nexus-<version>/bin 경로에서 nexus를 실행시킨다.
* 넥서스 구동은 다음 명령어로 실행 가능하다

$ ./nexus run

* 실행에 성공할 경우 아래와 같은 메시지가 출력된다

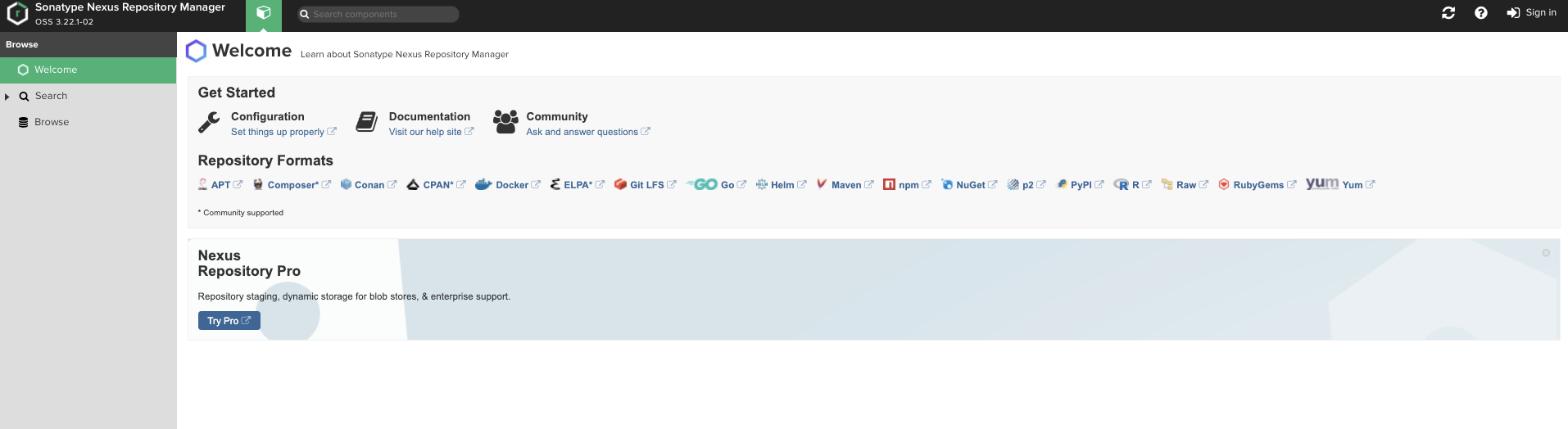


# 

# 3. Nexus UI 접속 및 설정

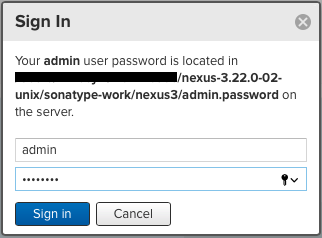
### 3.1 브라우저에서 Nexus 접속

* http://<**server\_host**>:<**port**>/
* 접속에 성공할 경우 다음과 같은 화면이 보인다.

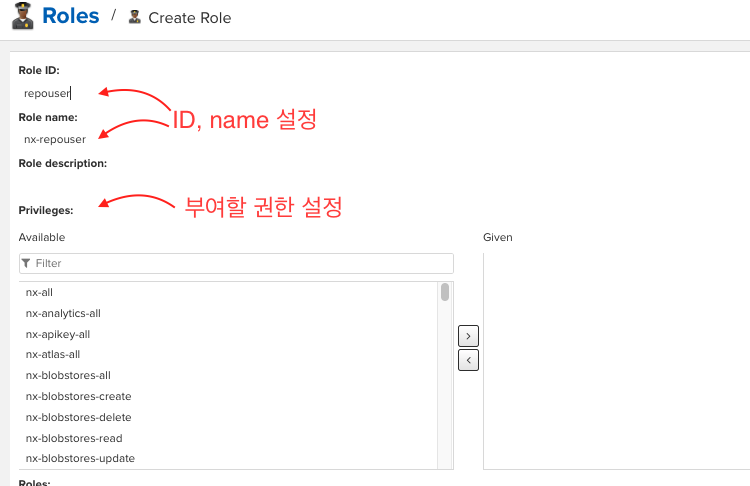
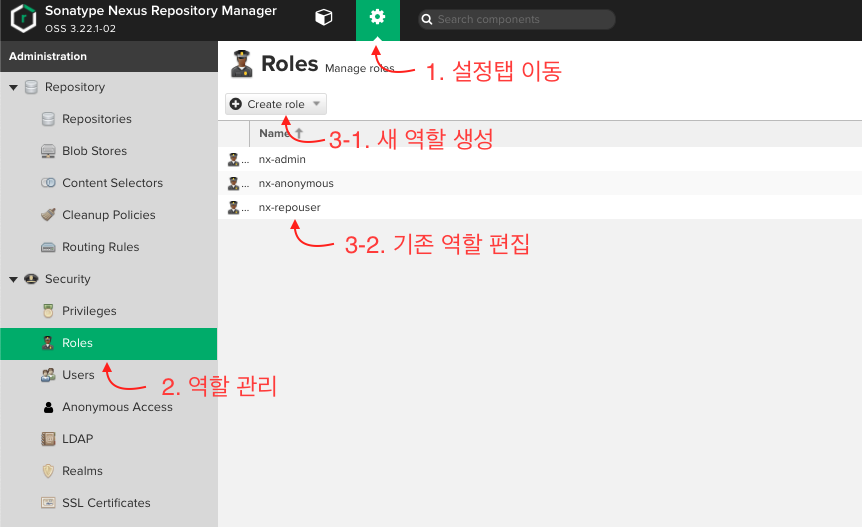


### 3.2 관리자 계정 로그인

* 초기 로그인시에는 관리자 계정으로 로그인한다.
* ID는 admin 이며 PW는 <NEXUS\_PATH>/sonatype-work/nexus3/admin.password 에서 확인 가능하다



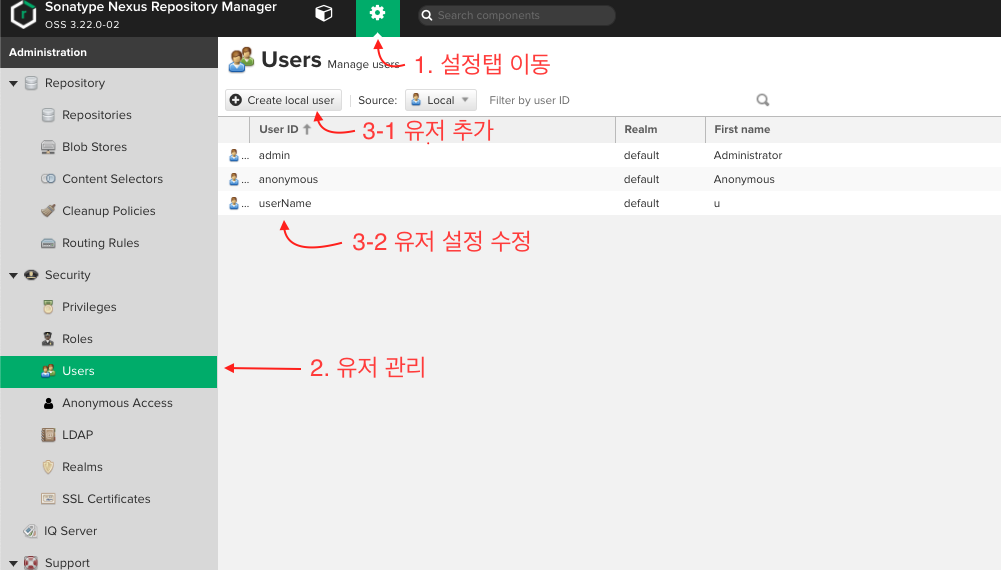
### 3.3 역할 관리

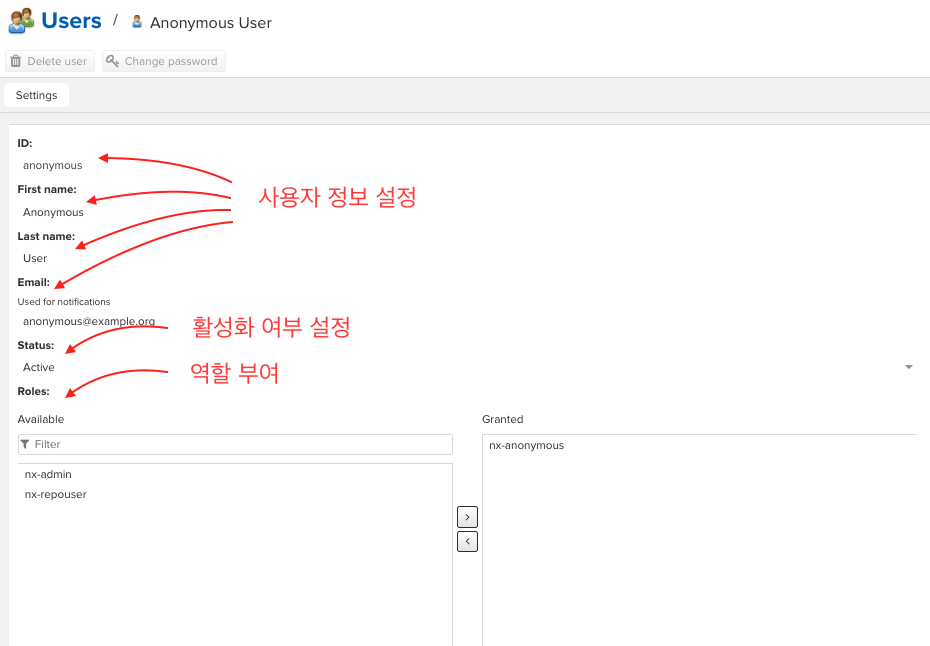


* Roles의 Privilege 설정을 통해 역할을 분배하거나 저장소별 권한을 따로 부여할 수 있다.
* 전체 저장소 사용을 위해 추가해주어야 하는 권한은 nx-repository-view-\*-\*-\*이다.

### 

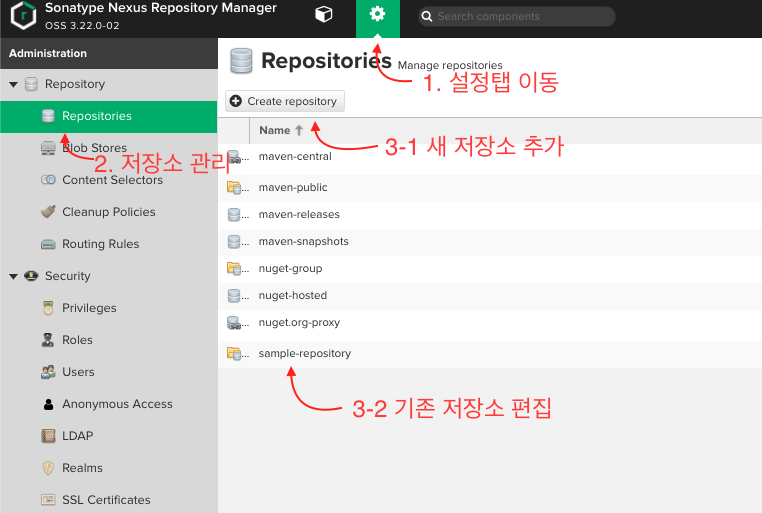
### 3.4 계정 관리

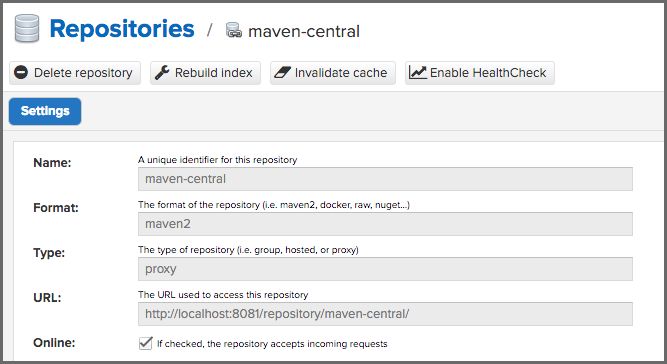




* 저장소 보안 관리를 위해 유저마다 다른 Roles를 설정 할 수 있다.

### 3.5 저장소 관리





* Name : 저장소 이름
* Format : maven 저장소로 사용할 것이므로 ‘maven2’ 선택
* Type : group, hosted, proxy 세 종류가 있다.

일반 저장소 - hosted, 여러 저장소를 하나의 url로 관리하려면 - group

# 

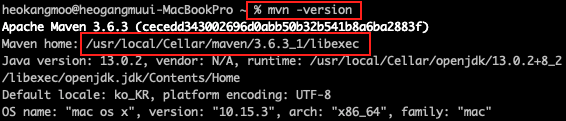
# 4. 배포 및 사용

### 4.1 로컬 환경 셋팅

* 메이븐과 원격 저장소를 연결시키기 위한 환경을 세팅한다.
  + 메이븐으로 빌드하는 프로젝트 파일(pom.xml)에 기술하면 되지만, 모든 프로젝트마다 적용되게 하려면 메이븐의 로컬 저장소에 settings.xml 파일에 설정하면 된다.
  + 본 문서는 settings.xml에 설정하는 방법을 적용한다.
  + maven이 설치되어 있지 않다면 배포 및 사용을 위해서 maven이 먼저 설치되어야 한다. Mac 유저라면 homebrew를 이용한 maven 설치를 권장한다.

#### 4.1.1 settings.xml 위치

* $ mvn -version 명령어로 MAVEN\_PATH 확인
* settings.xml 위치 -<MAVEN\_PATH>/conf/settings.xml
* 예시



#### 4.1.2 setting.xml 편집

Nexus에서 생성한 유저 정보를 추가를 통해 Repository에 접근할 때마다 유저의 정보를 입력할 필요가 없게 된다.

|  |
| --- |
| “setting.xml”  <servers>  <server>  <id>release repository ID</id> <!--프로젝트 배포시 사용할 Release Repository ID-->  <username>계정 ID</username> <!--Nexus 계정 정보-->  <password>비밀번호</password>  </server>  <server>  <id>snapshot repository ID</id> <!--프로젝트 배포시 사용할 Snapshot Repository ID-->  <username>계정 ID</username> <!--Nexus 계정 정보-->  <password>비밀번호</password>  </server>  </servers> |

### 4.2 deploy

저장소에 프로젝트를 배포하기 위해서 pom.xml에 아래 내용을 추가한 후, $ mvn deploy 명령어를 실행한다.

|  |
| --- |
| “pom.xml”  <groupId>org.example</groupId> <!--Repo 내의 디렉토리 이름과 경로-->  <artifactId>Project Name</artifactId> <!--프로젝트 이름, groupId 내에서 유일해야 한다-->  <version>1.0-SNAPSHOT</version> <!--“SNAPSHOT”이 있다면 Snapshot Repo에, 없다면 Release Repo에 저장-->  <distributionManagement> <!--프로젝트가 배포될 저장소 정보와 설정-->  <repository>  <id>release repository ID</id> <!--settings.xml에서 설정한 Release Repository ID->  <name>maven-releases</name> <!--Release Repo 이름(Optional)-->  <url>http://<Server\_IP>:<Port>/repository/<Repo Name>/</url> <!--Release Repo url-->  </repository>  <snapshotRepository>  <id>snapshot repository ID</id> <!--settings.xml에서 설정한 Snapshot Repository ID-->  <name>maven-snapshots</name> <!--Snapshot Repo 이름(Optional)-->  <url>http://<Server\_IP>:<Port>/repository/<Repo Name>/</url> <!--Snapshot Repo url-->  </snapshotRepository>  </distributionManagement> |

* Repository url 은 앞의 [3.5 저장소 관리](#f3i29nn1req1) 참고

Repository 클릭 후 화면에 나오는 URL을 복사한다.

### 

### 4.3 dependency 설정

저장소에 있는 라이브러리를 사용하기 위해 pom.xml에 아래 내용을 추가한 후 필요한 라이브러리의 의존성을 추가한다.

|  |
| --- |
| “pom.xml”  <repositories>  <repository>  <id>user</id>  <url>http://<Server\_IP>:<Port>/repository/<(Group) Repo Name>/</url> <!--Repository url-->  <releases><enabled>true</enabled></releases>  <snapshots><enabled>true</enabled></snapshots>  </repository>  </repositories>  ....  <dependency>  <groupId>org.example</groupId>  <artifactId>maventest</artifactId>  <version>1.0-SNAPSHOT</version>  </dependency>  <!--Release & Snapshot Repo가 속해있는 Group Repo를 불러오면 한 번에 두 Repo를 호출할 수 있다-->  <!--단일 Repo만을 불러오는 경우에는 하나의 Repo 이름만 적어주면 된다.--> |

# 

# 

# 5. 유의사항

### 5.1 dependency를 업데이트 할 경우

* 아래의 명령어를 사용하여 업데이트 할 수 있다.

$ mvn -U clean install

### 5.2 원격 저장소 내용이 반영이 되지 않을 경우

* 원격 저장소에서 수정이 되어도 로컬저장소부터 찾기 때문에 수정사항이 반영이 되지 않는 문제가 발생할 수 있다. 이 때는 로컬 캐시를 정리하면 된다.
* 만약 로컬 캐시를 완전히 정리해야 할 경우 아래의 명령어 사용

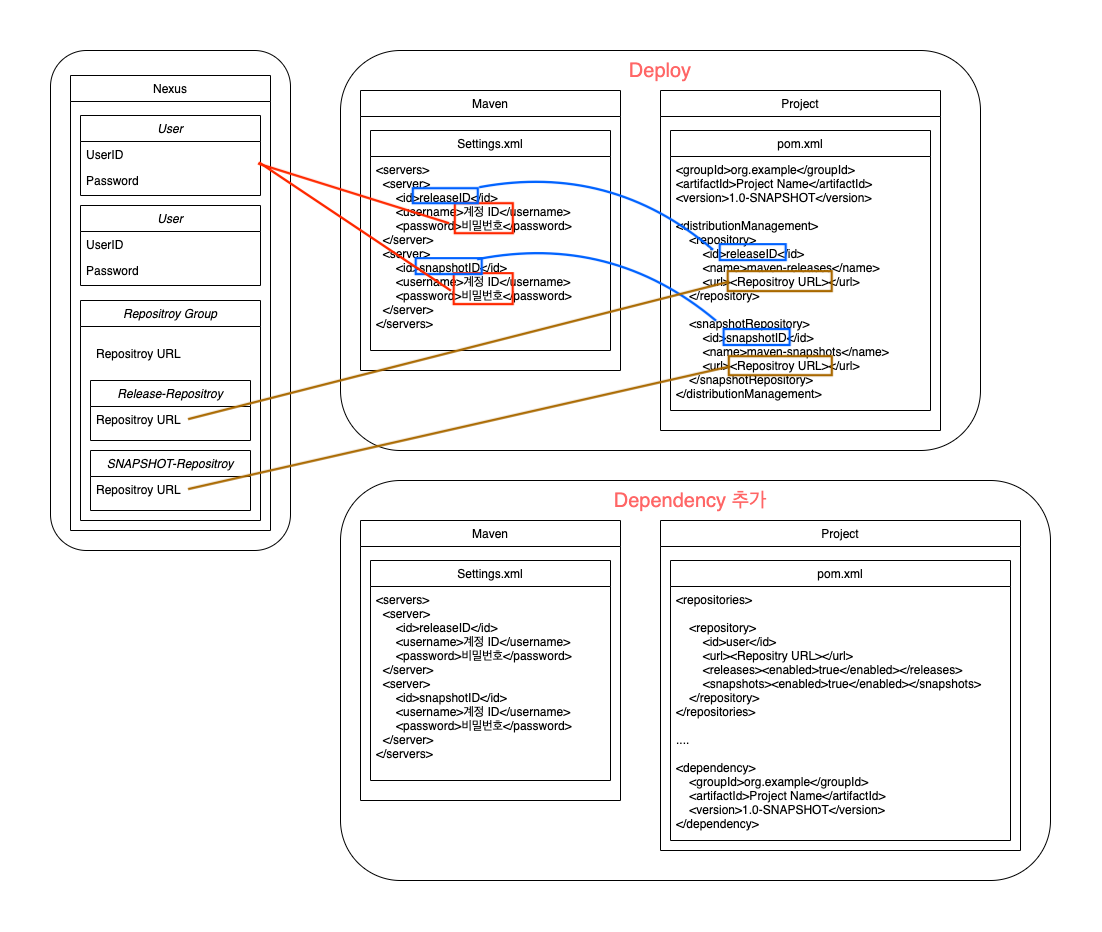
$ mvn dependency:purge-local-repository install

* 캐시 정리 후 Reimport를 위해서 우측 상단 Maven 탭의 Reimport 버튼 클릭

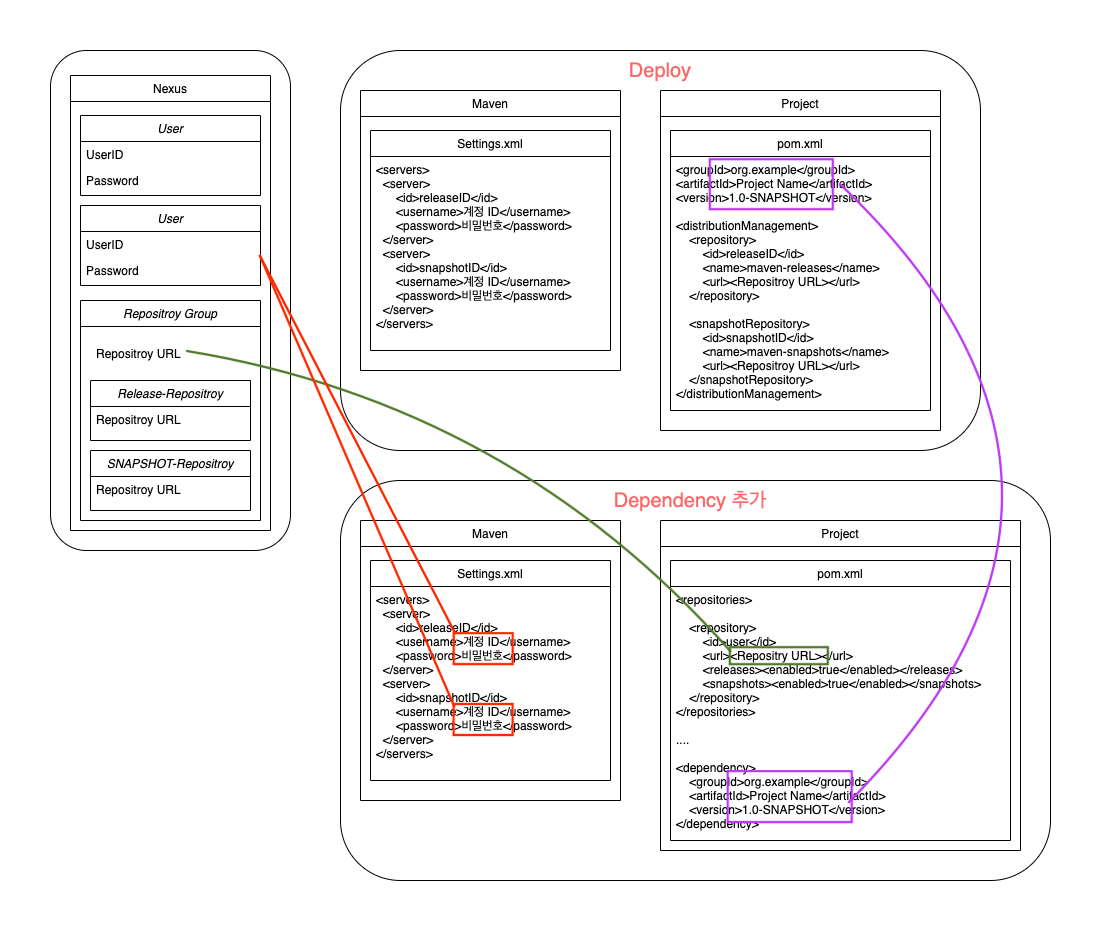


### 5.3 맞춰줘야 하는 설정값 요약

* Deploy 하는 경우 아래의 값들을 일치시켜 줘야한다.



* Dependency 추가하는 경우 아래의 값들을 일치시켜 줘야한다.



### 5.4 참고자료

* Nexus 공식 문서

<https://help.sonatype.com/repomanager3>