

# LU12a - Jenkins Secrets

Lokal werden Secrets oft in privaten `.env`-Dateien gespeichert. Bei Build-Pipelines müssen die Secrets auf dem Build-Server (oder einem 3rd-Party-Service) hinterlegt werden. Secrets direkt in die Jenkinsfile-Datei zu schreiben ist keine gute Idee, da sie für alle mit Lesezugriff auf das Code-Repository lesbar wären.

Die verschiedenen Typen von „secrets“, die Jenkins unterstützt, sind in folgender Tabelle aufgelistet.

Typ	Zweck	Beispiele	Verwendung in Pipeline
Secret text	Einzelner String-Wert	API Keys, Tokens	<code>withCredentials([string(...)])</code>
Username/Password	Login-Paare	DB, Docker Registry	<code>withCredentials([usernamePassword(...)])</code>
SSH Key	SSH Zugriff	Server Deploy, Git SSH	<code>sshUserPrivateKey</code>
Secret file	Datei als Secret	kubeconfig, JSON Keys, Zertifikate	<code>file(credentialsId, ...)</code>
Certificate	X.509 Zertifikate	mTLS, interne APIs	Zertifikatsbasierte Auth (zum Beispiel für Webhooks)

Nachfolgend ist die Verwendung in einem Jenkinsfile zu sehen. Die separate Helper-Funktion sorgt dafür, dass bei mehreren Stages nicht die Secret-ID mehrfach angegeben werden muss.

Jenkinsfile mit Passwort-Leak ☐	Jenkinsfile mit Secret-Helfer-Funktion ✓
<pre> environment {     DB_USER      = "appuser"     DB_PASSWORD  = "apppassword" } ... steps {     sh """         docker run -d \             --name \$DB_CONTAINER \             --restart unless- stopped \             --network infra-net \             -e POSTGRES_USER=\$DB_USER \             -e POSTGRES_PASSWORD=\$DB_PASSWORD \             -e POSTGRES_DB=\$DB_NAME \             -v \$DB_VOLUME:/var/lib/postgresql/data \             -v \$WORKSPACE/database:/docker-entrypoint-initdb.d \             postgres:17             """     } </pre>	<pre> def withDbCredentials(body) {     withCredentials([         usernamePassword(             credentialsId: 'postgres-creds',             usernameVariable: 'DB_USER',             passwordVariable: 'DB_PASSWORD'         )     ]) {         body()     } } ... steps {     withDbCredentials {         sh """             docker run -d \                 --name \$DB_CONTAINER \                 --restart unless- stopped \                 --network infra-net \                 -e POSTGRES_USER=\$DB_USER \                 -e POSTGRES_PASSWORD=\$DB_PASSWORD \                 -e POSTGRES_DB=\$DB_NAME \                 -v \$DB_VOLUME:/var/lib/postgresql/data \                 -v \$WORKSPACE/database:/docker-entrypoint-initdb.d \                 postgres:17                 """     } } </pre>

Ein wichtiger Grundsatz bezüglich der Sicherheit von Jenkins-Secrets lautet: „Wenn ein Benutzer eine Pipeline editieren kann, kann er in der Regel auch Secrets exfiltrieren.“

From:  
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:  
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/cicd/learningunits/lu12/a>

Last update: **2026/05/18 23:27**

