

LU12a - Jenkins Secrets

Anstatt Secrets direkt in die Jenkinsfile-Datei zu schreiben, wo es für alle mit Lesezugriff auf das Code-Repository lesbar wäre, können auf Jenkins „secrets“ hinterlegt werden.

TODO: Tabelle Arten von Secrets

Jenkinsfile mit Passwort-Leak ☐	Jenkinsfile mit Secret-Helper-Funktion ✓
<pre> environment { DB_PASSWORD = "apppassword" DB_NAME = "appdb" } ... 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.“

- .env-Files
- Secrets in Github
- Secrets in Jenkins
- Sonar-Token
- Webhook?

TODO

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

Permanent link:
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/cicd/learningunits/lu12/a?rev=1778778837>

Last update: **2026/05/14 19:13**

