TeamCity: logowanie kontem z Active Directory

Typ: procedura **Dotyczy: TeamCity 2026.2, Active Directory na dc01/dc02**

Czas: ~35 min

Efekt: logowanie do TeamCity kontem domenowym z Active Directory, z automatycznym przypisaniem do grupy TeamCity dla osób z odpowiedniej grupy w AD, bez zakładania kont lokalnych dla każdej nowej osoby.

Skróty użyte na tej stronie (2)

Skrót	Co znaczy
AD	Active Directory: katalog Microsoftu z kontami użytkowników, komputerów i grup w firmie
LDAP	Lightweight Directory Access Protocol: protokół do odpytywania katalogu użytkowników (AD, FreeIPA)

Zanim zaczniemy

Z ŻYCIA WZIĘTE

Tomek loguje się do TeamCity tym samym kontem domenowym, którym loguje się do komputera, do poczty i do Gitea. Nikt nie zakłada mu osobnego konta lokalnego i nie wysyła osobnego hasła mailem.

Pojęcie: moduł uwierzytelniania LDAP w TeamCity. Wtyczka wbudowana w serwer, którą włączamy z poziomu interfejsu, ale konfigurujemy edycją pliku tekstowego, nie formularzem, w przeciwieństwie do Gitea, opisanej na stronie Logowanie kontem z Active Directory w książce o Gitea. Plik jest wczytywany na nowo automatycznie przy każdej zmianie, bez restartu serwera.

Komendy PowerShell wykonujemy z konsoli na `dc01` z uprawnieniami administratora domeny. Edycję pliku właściwości wykonujemy na `ci01`, na koncie z prawem zapisu do katalogu danych TeamCity.

Wymaganie	Uwaga
Konto techniczne w AD do bind LDAP	zakładamy w kroku 1, prawo tylko do odczytu katalogu
Dwie grupy w AD: użytkownicy i administratorzy TeamCity	zakładamy w kroku 1
LDAPS na porcie 636 osiągalny z `ci01` do `dc01` / `dc02`	sprawdzamy zaporą, zanim zaczniemy konfigurację
Dostęp do katalogu danych TeamCity na `ci01`	`/docker/apps/teamcity/data/config/` w instalacji Dockerowej

Uważaj: konto techniczne wskazane przy bind LDAP powinno mieć wyłącznie prawo odczytu katalogu, na tej samej zasadzie najmniejszych uprawnień, na jakiej opieramy konto techniczne Gitea. TeamCity nie potrzebuje żadnego prawa zapisu w Active Directory do samego uwierzytelniania użytkowników.

Przebieg

1. Konto techniczne i grupy w Active Directory

Zakładamy konto techniczne do bind LDAP:

```
New-ADUser -Name "svc-teamcity-ldap" `
  -SamAccountName "svc-teamcity-ldap" `
  -UserPrincipalName "svc-teamcity-ldap@ad.zielona.example" `
  -Path "OU=KontaTechniczne,DC=ad,DC=zielona,DC=example" `
  -AccountPassword (Read-Host -AsSecureString "Haslo konta") `
  -Enabled $true `
  -PasswordNeverExpires $true `
  -CannotChangePassword $true
```

Zakładamy dwie grupy: jedną dla wszystkich osób z dostępem do TeamCity, drugą dla administratorów:

```
New-ADGroup -Name "TeamCity-Uzytkownicy" -GroupScope Global -Path
"OU=Grupy,DC=ad,DC=zielona,DC=example"
```

```
New-ADGroup -Name "TeamCity-Administratorzy" -GroupScope Global -Path
"OU=Grupy,DC=ad,DC=zielona,DC=example"
```

Dodajemy Olę do grupy administratorów:

```
Add-ADGroupMember -Identity "TeamCity-Administratorzy" -Members "ola.wisniewska"
```

2. Utworzenie pliku ldap-config.properties

Kopiujemy szablon dostarczony przez TeamCity, zamiast pisać plik od zera:

```
cd /docker/apps/teamcity/data/config
sudo cp ldap-config.properties.dist ldap-config.properties
```

Uzupełniamy plik `ldap-config.properties`:

```
java.naming.provider.url=ldaps://dc01.ad.zielona.example:636/DC=ad,DC=zielona,DC=example
java.naming.security.principal=CN=svc-teamcity-
ldap,OU=KontaTechniczne,DC=ad,DC=zielona,DC=example
java.naming.security.credentials=HasloKontaTechnicznego
java.naming.security.authentication=simple
java.naming.referral=follow

teamcity.users.login.filter=(&(objectClass=user)(sAMAccountName=$capturedLogin$)(memberOf:1.2.
840.113556.1.4.1941:=CN=TeamCity-Uzytkownicy,OU=Grupy,DC=ad,DC=zielona,DC=example))
teamcity.users.username=sAMAccountName
teamcity.users.property.displayName=displayName
teamcity.users.property.email=mail

teamcity.options.createUsers=true
teamcity.options.users.synchronize=true
teamcity.options.groups.synchronize=true
teamcity.groups.retrieveUsersFromNestedGroups=true
```

Uważaj: wartości w tym pliku są zwykłymi właściwościami Javy, więc odwrotny ukośnik w nazwie konta domenowego wymaga podwojenia (`ZIELONA\\\\jan.nowak`). Filtr logowania wyżej wpuszcza tylko konta należące, także pośrednio przez grupy zagnieżdżone, do `TeamCity-Uzytkownicy`, dzięki tej samej regule `LDAP_MATCHING_RULE_IN_CHAIN` (`1.2.840.113556.1.4.1941`), która pojawiła się już przy LDAP w Gitea.

3. Mapowanie grupy administratorów: ldap-mapping.xml

Plik `/docker/apps/teamcity/data/config/ldap-mapping.xml` łączy grupę z Active Directory z grupą wewnętrzną TeamCity, którą zakładamy w kolejnym kroku:

```
<mapped-groups>
  <group name="Administratorzy TeamCity" mapped-key="CN=TeamCity-
Administratorzy,OU=Grupy,DC=ad,DC=zielona,DC=example"/>
</mapped-groups>
```

4. Grupa w TeamCity i nadanie uprawnień

W interfejsie zakładamy grupę o dokładnie takiej nazwie, jaką wskazaliśmy w `ldap-mapping.xml`:

```
Administracja → Users → Groups → Create group
```

Nazwa grupy: `Administratorzy TeamCity`. Grupie tej przypisujemy rolę administratora systemu:

```
Administracja → Users → Groups → Administratorzy TeamCity → Roles → System Admin
```

Od pierwszej synchronizacji po zalogowaniu Oli jej konto trafia do tej grupy automatycznie, bez ręcznego dodawania.

5. Włączenie modułu LDAP

```
Administracja → Authentication → Add module → LDAP
```

Plik `ldap-config.properties` jest wczytywany na nowo automatycznie po każdej zmianie: nie restartujemy kontenera serwera po edycji.

◫ ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran Administracja → Authentication z widocznym wierszem modułu LDAP w stanie włączonym, obok domyślnego modułu haseł lokalnych.

Weryfikacja

Testujemy logowanie kontem domenowym na ekranie logowania TeamCity, używając loginu `tomek.nowak` i hasła domenowego, zamiast konta lokalnego.

Sprawdzamy log modułu LDAP, gdyby logowanie się nie powiodło:

```
docker exec teamcity-server tail -n 50 /opt/teamcity/logs/teamcity-ldap.log
```

Przykład:

```
[TeamCity LDAP] Successfully authenticated user: tomek.nowak
[TeamCity LDAP] User groups: CN=TeamCity-Uzytkownicy,OU=Grupy,DC=ad,DC=zielona,DC=example
```

Sprawdzamy, że Ola po zalogowaniu widzi grupę `Administratorzy TeamCity` przypisaną do swojego konta:

```
Administracja → Users → ola.wisniewska
```

◫ ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: profil konta `ola.wisniewska` w TeamCity, sekcja „Groups", z widoczną grupą „Administratorzy TeamCity" i źródłem „LDAP".

Wycofanie zmian

Wyłączenie modułu LDAP bez usuwania zsynchronizowanych kont, gdy chcemy tymczasowo wrócić do logowania lokalnego:

```
Administracja → Authentication → moduł LDAP → Disable
```

Pełne wyłączenie synchronizacji, z zachowaniem samego pliku konfiguracyjnego na przyszłość:

```
teamcity.options.users.synchronize=false
teamcity.options.groups.synchronize=false
```

Gdy coś nie zadziała

Objaw	Przyczyna	Co robimy
W logu `teamcity-ldap.log` wpis `LDAP Result Code 49 Invalid Credentials`	zły `java.naming.security.principal` albo hasło konta technicznego	sprawdzamy dane bind komendą `ldapsearch` z hosta `ci01` przed poprawką w pliku
`x509: certificate signed by unknown authority` w logu	certyfikat firmowego CA niezaufany w kontenerze serwera	dokładamy certyfikat CA do magazynu zaufania kontenera, tak jak przy LDAPS w Gitea
Konto istnieje w AD, ale logowanie odrzucone	filtr logowania bez `$capturedLogin$` albo konto poza grupą wskazaną w filtrze	sprawdzamy `teamcity.users.login.filter` i członkostwo konta w `TeamCity-Uzytkownicy`, także pośrednie
Konto loguje się, ale nie widzi uprawnień administratora mimo członkostwa w grupie	brak wpisu w `ldap-mapping.xml` albo nazwa grupy w TeamCity niezgodna z `mapped-key`	sprawdzamy zgodność nazwy grupy w interfejsie z wpisem w `ldap-mapping.xml`

Powiązane: Instalacja TeamCity w Dockerze · Logowanie kontem z Active Directory (Gitea) · Active Directory: konta, grupy, jednostki organizacyjne

Źródła: jetbrains.com/help/teamcity, rozdział „LDAP Integration" · jetbrains.com/help/teamcity, rozdział „Typical LDAP Configurations" · Microsoft Learn, „Search Filter Syntax" (LDAP_MATCHING_RULE_IN_CHAIN)

Revision #31
Created 2026-09-27 11:19:36 UTC by Tomasz Konieczny
Updated 2026-10-26 14:17:48 UTC by Tomasz Konieczny