Pierwsze uruchomienie: kreator, administrator, reverse proxy

Poziom: podstawy **Dotyczy: Gitea 1.27** **Czas: ~25 min**

Po pierwszym starcie kontenera albo usługi Gitea czeka kreator konfiguracji w przeglądarce. Ta strona prowadzi przez kreator, przez założenie pierwszego konta administratora i przez postawienie Nginksa przed Giteą, żeby nie wystawiać portu 3000 na świat wprost.

Skróty użyte na tej stronie (4)

Skrót	Co znaczy
HTTP	Hypertext Transfer Protocol: protokół stron WWW
SSH	Secure Shell: szyfrowane zdalne logowanie do serwera
TLS	Transport Layer Security: szyfrowanie połączeń, np. w HTTPS
URL	Uniform Resource Locator: adres strony lub zasobu

Fiszka

TYLKO ODCZYT

Otwarcie kreatora instalacji: adres w przeglądarce

```
http://docker01:3000
```

Przy pierwszym uruchomieniu, dopóki `INSTALL_LOCK` nie jest ustawiony na `true`, Gitea pokazuje zamiast strony logowania kreator instalacji.

□□ ZMIENIA SYSTEM

Nadanie pierwszych uprawnień administratora z CLI: gitea admin user create

```
docker exec -u git gitea gitea admin user create --username ola.wisniewska --password HasloAdministratora --email ola.wisniewska@zielona.example --admin
```

Alternatywa dla konta administratora zakładanego w samym kreatorze, przydatna przy automatyzacji albo gdy kreator zdążył już zablokować się bez utworzenia konta.

□□ **CLI** = Command Line Interface: praca przez wpisywanie komend w terminalu · **docker** = narzędzie do kontenerów

□□ TYLKO ODCZYT

Weryfikacja odpowiedzi Nginksa po podłączeniu Gitea: curl -I

```
curl -I https://git.zielona.example
```

Sprawdzamy, że reverse proxy w ogóle przekazuje ruch do kontenera, zanim zaczniemy szukać błędu po stronie samej Gitea.

Przykład:

```
HTTP/2 200
server: nginx
content-type: text/html; charset=UTF-8
set-cookie: i_like_gitea=...
```

□□ **curl** = client URL: pobieranie i wysyłanie przez HTTP z terminala

Z życia wzięte

□□ Z ŻYCIA WZIĘTE

Tomek, drugi tydzień pracy, dostaje zadanie postawienia Gitea w laboratorium. Przechodzi kreator, klika przez wszystkie pola bez czytania i zostawia `ROOT_URL` z domyślnym adresem `http://localhost:3000`. Dzień później powiadomienia e-mail o nowych pull requestach

> przychodzą z linkami wskazującymi na `localhost`, bezużytecznymi dla nikogo poza samym Tomkiem.

Kreator instalacji pole po polu

> **Pojęcie: kreator instalacji.** Formularz w przeglądarce, przez który Gitea przy pierwszym uruchomieniu zbiera minimalny zestaw ustawień i zapisuje go od razu do pliku `app.ini`. Po zatwierdzeniu kreator sam ustawia `INSTALL_LOCK = true`, więc powtórne wejście na ten adres pokazuje już zwykły ekran logowania.

Pole kreatora	Co wpisujemy w Zielonej	Dlaczego
Database Type	PostgreSQL	zgodnie z bazą przygotowaną na `pg01` albo w kontenerze `db`
Host	`pg01:5432` (produkcja) albo `db:5432` (laboratorium)	adres bazy z kroku instalacji
Username / Password	`gitea` / hasło z `.env`	konto założone wcześniej w PostgreSQL
Database Name	`gitea`	nazwa bazy założonej wcześniej
Site Title	`Gitea Zielona`	nazwa widoczna w interfejsie i w tytule karty przeglądarki
Repository Root Path	pozostawiamy domyślne `/data/git/repositories`	ścieżka wewnątrz kontenera, mapowana przez wolumen na `/docker/apps/gitea/data`
SSH Server Domain	`git.zielona.example`	adres, pod jakim Gitea podaje polecenia klonowania przez SSH
SSH Server Port	`2222`	port zmapowany na hoście, patrz Instalacja Gitea w Dockerze
Gitea Base URL (ROOT_URL)	`https://git.zielona.example/`	rzeczywisty adres, pod jakim użytkownicy otwierają Giteę, nie adres wewnętrzny kontenera
Optional Settings → Admin Account	konto Oli jako pierwszego administratora	pierwsza osoba z pełnymi uprawnieniami w systemie

Uważaj: pole `Gitea Base URL` odpowiada kluczowi `ROOT_URL` w `app.ini`. Jeśli zostanie tam adres wewnętrzny albo `localhost`, wszystkie linki generowane przez Giteę, w mailach, w webhookach, w podpowiedziach klonowania, wskażą na adres bezużyteczny dla kogokolwiek poza tą jedną maszyną.

ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran kreatora instalacji Gitea, sekcja „General Settings", z widocznymi polami „Site Title" ustawionym na „Gitea Zielona" i „Gitea Base URL" ustawionym na `https://git.zielona.example/`.

Pierwsze konto administratora

Konto założone w sekcji „Admin Account Settings" kreatora dostaje od razu najwyższy poziom uprawnień w całej instalacji: zarządzanie użytkownikami, organizacjami, źródłami uwierzytelniania i globalną konfiguracją widoczną z poziomu interfejsu. W Zielonej to konto Oli, jako prowadzącej zespół programistów.

ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran „Site Administration” po zalogowaniu kontem Oli, zakładka „Dashboard”, z widoczną wersją Gitea i liczbą zarejestrowanych użytkowników.

app.ini: najważniejsze sekcje

> **Pojęcie: app.ini.** Główny plik konfiguracyjny Gitea. Kreator zapisuje go przy pierwszym uruchomieniu, a dalsze zmiany, których kreator nie obejmuje, wprowadzamy edytując go ręcznie i restartując usługę, albo nadpisując wybrane klucze zmiennymi środowiskowymi `GITEA__sekcja__KLUCZ`.

Sekcja [server]. `ROOT_URL` to adres, pod jakim instancja jest widoczna z zewnątrz. `DOMAIN` to sama nazwa hosta, bez protokołu i ścieżki, używana między innymi przy generowaniu ciasteczek. `SSH_DOMAIN` i `SSH_PORT` to adres i port, jakie Gitea podaje w podpowiedziach `git clone` przez SSH, niezależnie od tego, pod jakim portem SSH faktycznie nasłuchuje kontener wewnątrz.

```
[server]
DOMAIN     = git.zielona.example
ROOT_URL   = https://git.zielona.example/
SSH_DOMAIN = git.zielona.example
SSH_PORT   = 2222
```

Sekcja [service]. `DISABLE_REGISTRATION` ustawione na `true` wyłącza samodzielną rejestrację kont z poziomu strony logowania: w Zielonej konta zakłada się przez import z Active Directory albo ręcznie przez administratora, nie przez otwarty formularz.

```
[service]
DISABLE_REGISTRATION = true
```

Sekcja [security]. `INSTALL_LOCK` ustawiony na `true` blokuje ponowne uruchomienie kreatora. Bez tego ustawienia dowolna osoba ze świeżym dostępem do adresu instalacji mogłaby przejść przez kreator ponownie i nadpisać konfigurację produkcyjną.

```
[security]
INSTALL_LOCK = true
```

Sekcja [mailer]. Włącza wysyłkę powiadomień e-mail, na przykład o nowych pull requestach albo wzmiankach w komentarzu. Bez tej sekcji Gitea działa normalnie, tylko bez powiadomień.

```
[mailer]
ENABLED  = true
SMTP_ADDR = smtp.zielona.example
SMTP_PORT = 587
FROM      = gitea@zielona.example
USER      = gitea@zielona.example
PASSWD    = HasloSkrzynki
```

Reverse proxy przed Giteą

Gitea sama w sobie serwuje ruch HTTP na porcie 3000, bez obsługi TLS. W Zielonej certyfikat kończy Nginx stojący na tym samym hoście `docker01`, opisany szerzej na stronie Reverse proxy i równoważenie obciążenia: tutaj tylko konkretny blok gotowy do skopiowania.

```
server {
    listen 443 ssl;
    server_name git.zielona.example;

    ssl_certificate     /etc/nginx/ssl/git.zielona.example.crt;
    ssl_certificate_key /etc/nginx/ssl/git.zielona.example.key;

    client_max_body_size 512m;
```

```
    location / {
        proxy_pass http://127.0.0.1:3000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }
}

server {
    listen 80;
    server_name git.zielona.example;
    return 301 https://$host$request_uri;
}
```

`client_max_body_size` podnosimy wyraźnie ponad domyślny 1MB: repozytoria z dużymi plikami binarnymi albo push z wieloma commitami naraz łatwo przekraczają domyślny limit, kończąc się błędem 413 bez żadnego komunikatu ze strony samej Gitea.

◫ ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: strona logowania Gitea otwarta pod adresem `https://git.zielona.example`, z widoczną kłódką certyfikatu w pasku adresu przeglądarki.

Częste błędy

Objaw	Przyczyna	Co robimy
Linki w mailach i webhookach wskazują na zły adres	`ROOT_URL` niezgodny z rzeczywistym adresem	poprawiamy `ROOT_URL` w `app.ini`, restartujemy kontener
Push przez SSH podaje w podpowiedzi zły port	`SSH_DOMAIN` albo `SSH_PORT` niezgodne z rzeczywistym mapowaniem portów	poprawiamy sekcję `[server]` na wartości pasujące do `docker compose.yaml`
Duży push kończy się błędem 413	brak albo zbyt niski `client_max_body_size` w Nginksie	podnosimy limit w bloku `server` przed Giteą
Kreator pokazuje się ponownie po restarcie	`INSTALL_LOCK` nie zapisał się (błąd zapisu do `app.ini`)	sprawdzamy uprawnienia do pliku `app.ini` dla konta, na którym działa Gitea

Ćwiczenia

1. Wyjaśniamy, dlaczego pole `Gitea Base URL` w kreatorze ma znaczenie nawet wtedy, gdy Gitea jest dostępna wyłącznie z sieci wewnętrznej firmy.

Odpowiedź

To pole trafia do `ROOT_URL` i jest używane przy generowaniu każdego linku bezwzględnego: w mailach z powiadomieniami, w linkach webhooków, w podpowiedziach klonowania. Nawet w sieci wewnętrznej adres musi być tym, pod którym rzeczywiście łączą się użytkownicy, inaczej linki prowadzą donikąd.

1. Push dużego repozytorium kończy się błędem 413 mimo poprawnej konfiguracji samej Gitea. Wskazujemy, gdzie szukamy przyczyny i jak ją naprawiamy.

Odpowiedź

Błąd 413 pochodzi zwykle od reverse proxy, nie od Gitea: sprawdzamy `client_max_body_size` w konfiguracji Nginksa przed Giteą i podnosimy limit odpowiednio do rozmiaru repozytoriów.

Powiązane: Instalacja Gitea w Dockerze · Reverse proxy i równoważenie obciążenia · Logowanie kontem z Active Directory

Źródła: docs.gitea.com, rozdział „Config Cheat Sheet" · docs.gitea.com, rozdział „Initialization"

Revision #33
Created 2026-09-27 10:32:05 UTC by Tomasz Konieczny
Updated 2026-10-26 14:17:29 UTC by Tomasz Konieczny