Zitadel

Ein Identitätsanbieter, welcher Benutzer- und Berechtigungsmanagement bereitstellt sowie Authentifizierung und Autorisierung. Es lässt sich dank OpenID, SAML 2.0, LDAP und OAuth perfekt in bestehende Anwendungen und Umgebungen integrieren. Unterstützt wird auch SSO, sodass eine Anmeldung in Zitadel reicht, um in allen verbunden Applikationen angemeldet zu sein.

- Installation
- Installation hinter Reverse Proxy

Installation

Im folgenden wird die Installation von Zitadel mit Docker Compose beschrieben.

Die Anleitung beruht auf der offiziellen Dokumentation von Zitadel: Zitadel Docker compose

```
version: '3.8'

services:
  zitadel:
    restart: 'unless-stopped'
    image: 'ghcr.io/zitadel/zitadel:latest'
    container_name: zitadel
    command: 'start-from-init --masterkey "MasterkeyNeedsToHave32Characters" --tlsMode
disabled'
    environment:
      - 'ZITADEL_DATABASE_POSTGRES_HOST=zitadel-db'
      - 'ZITADEL_DATABASE_POSTGRES_PORT=5432'
      - 'ZITADEL_DATABASE_POSTGRES_DATABASE=zitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_USERNAME=zitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_PASSWORD=PasswortFuerZitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_SSL_MODE=disable'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_USERNAME=postgres'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_PASSWORD=PasswortFuerDenPostgresChef'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_SSL_MODE=disable'
      - 'ZITADEL_EXTERNALSECURE=false'
    depends_on:
      zitadel-db:
        condition: 'service_healthy'
    ports:
      - '8080:8080'

  zitadel-db:
    restart: 'unless-stopped'
    image: postgres:16-alpine
    container_name: zitadel-db
    environment:
```

```
      - POSTGRES_USER=postgres
      - POSTGRES_PASSWORD=PasswortFuerDenPostgresChef
    volumes:
      - /pfad/zu/zitadel/data:/var/lib/postgresql/data
    healthcheck:
      test: ["CMD-SHELL", "pg_isready", "-d", "db_prod"]
      interval: '10s'
      timeout: '30s'
      retries: 5
      start_period: '20s'
```

Bitte die Passwörter ersetzen sowie den Pfad beim Volume von Postgres an die eigene Umgebung anpassen.

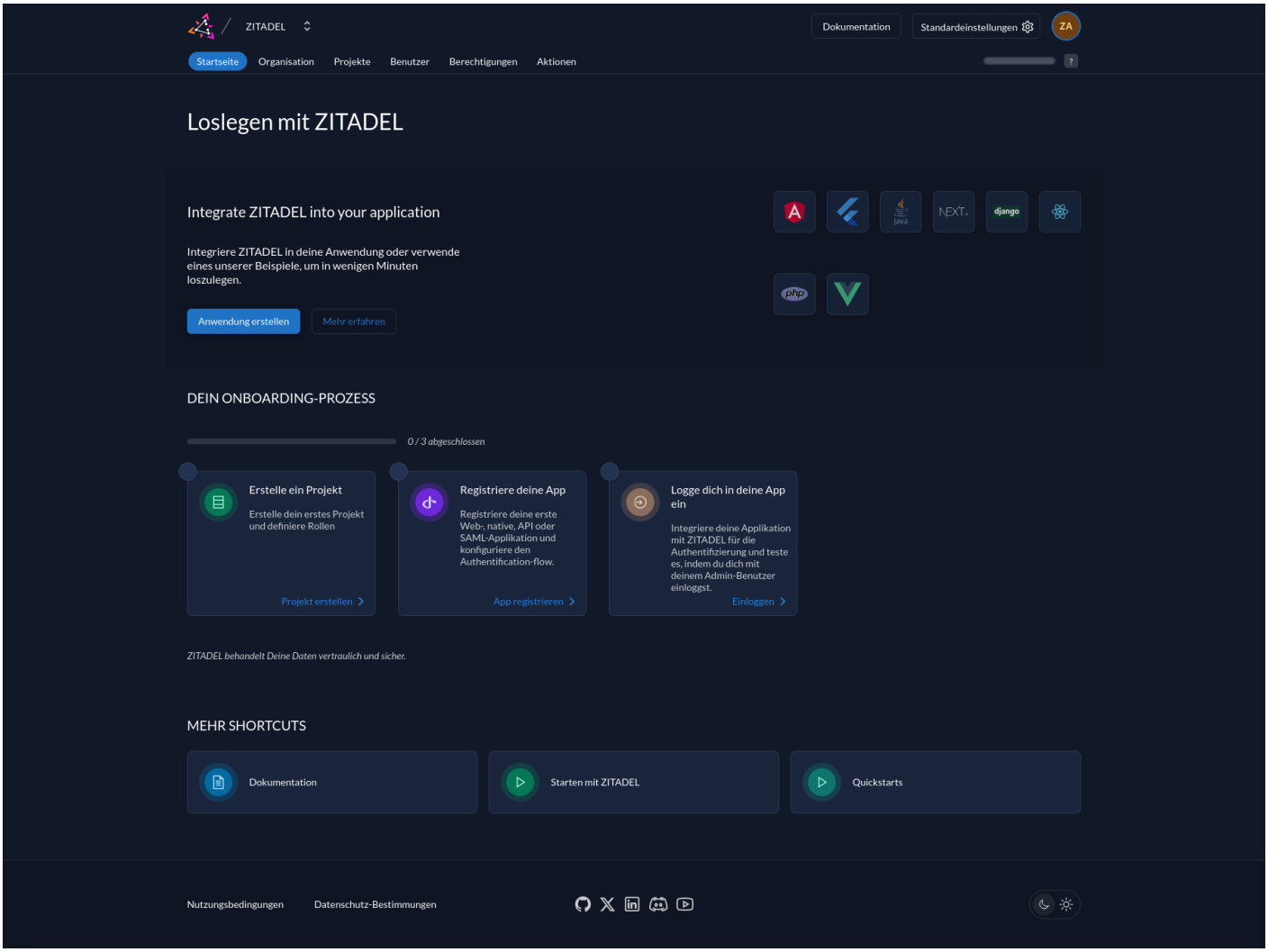
> Diese Konfiguration ist nur für Tests geeignet, da SSL deaktiviert ist und somit die Verbindung nicht verschlüssel ist.

Um die Docker Compose Konfiguration auszuführen, kann am besten in das Verzeichnis der YAML Datei gewechselt werden. Danach wird je nach nach gewählter Installation `sudo docker-compose up -d` oder `sudo docker compose up -d` (keine Bindestrich zwischen docker und compose) eingegeben, um die Standard Konfiguration `docker-compose.yml` zu starten. Compose erstellt dann die gewünschten Container mit den angegeben Optionen. Sollten die Container bereits mit dieser Compose Konfiguration erstellt worden sein, so werden die Container in dieser neu erstellt, dessen Konfiguration geändert wurde.

Nachdem die Instanz gestartet wurde, ist diese unter http://localhost:8080/ui/console/ erreichbar.

Der Standard-Login ist wie folgt:
Benutzername: zitadel-admin@zitadel.localhost
Passwort:*Password1!*

Nach ein paar kleineren Einstellungen in einem geführten Dialog (Passwort ändern, 2FA hinzufügen) öffnet sich die Startseite, auf der unter anderem auch ein Onboarding Prozess angezeigt wird.


ZITADEL
Dokumentation
Standardeinstellungen
ZA
Startseite
Organisation
Projekte
Benutzer
Berechtigungen
Aktionen
Loslegen mit ZITADEL
Integrate ZITADEL into your application
Integriere ZITADEL in deine Anwendung oder verwende eines unserer Beispiele, um in wenigen Minuten loszulegen.
Anwendung erstellen
Mehr erfahren
DEIN ONBOARDING-PROZESS
0 / 3 abgeschlossen
Erstelle ein Projekt
Erstelle dein erstes Projekt und definiere Rollen
Projekt erstellen
Registriere deine App
Registriere deine erste Web-, native, API oder SAML-Applikation und konfiguriere den Authentification-flow.
App registrieren
Logge dich in deine App ein
Integriere deine Applikation mit ZITADEL für die Authentifizierung und teste es, indem du dich mit deinem Admin-Benutzer einloggst.
Einloggen
ZITADEL behandelt Deine Daten vertraulich und sicher.
MEHR SHORTCUTS
Dokumentation
Starten mit ZITADEL
Quickstarts
Nutzungsbedingungen
Datenschutz-Bestimmungen

Installation hinter Reverse Proxy

Die auf der vorherigen Seite erklärte Installation ist aufgrund der fehlenden Verschlüsselung unsicher. Statt beim Container SSL zu aktivieren und ein Zertifikat einzuspielen, kann am besten ein sog. Reverse Proxy verwendet werden.

Im folgenden werden die notwendigen Anpassungen an Zitadel beschrieben, jedoch wird davon ausgegangen, dass bereits ein Reverse Proxy im Einsatz und mit einem gültigen SSL-Zertifikat ausgestattet ist. Außerdem wird angenommen, dass der Reverse Proxy die Domäne id.beispiel.de nutzt auf Port 443 (Standard HTTPS Port).

Die Compose Konfiguration ist wie folgt anzupassen.

```
services:
  zitadel:
    image: 'ghcr.io/zitadel/zitadel:latest'
    container_name: zitadel
    command: 'start-from-setup --init-projections --masterkey
"MasterkeyNeedsToHave32Characters"'
    environment:
      - 'ZITADEL_DATABASE_POSTGRES_HOST=zitadel-db'
      - 'ZITADEL_DATABASE_POSTGRES_PORT=5432'
      - 'ZITADEL_DATABASE_POSTGRES_DATABASE=zitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_USERNAME=zitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_PASSWORD=PasswortFuerZitadel'
      - 'ZITADEL_DATABASE_POSTGRES_USER_SSL_MODE=disable'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_USERNAME=postgres'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_PASSWORD=PasswortFuerDenPostgresChef'
      - 'ZITADEL_DATABASE_POSTGRES_ADMIN_SSL_MODE=disable'
      - 'ZITADEL_EXTERNALSECURE=true'
      - 'ZITADEL_EXTERNALPORT=443'
      - 'ZITADEL_TLS_ENABLED=false'
      - 'ZITADEL_EXTERNALDOMAIN=beispiel.de'
    depends_on:
      zitadel-db:
```

```
        condition: 'service_healthy'
    restart: unless-stopped

  zitadel-db:
    image: postgres:16-alpine
    container_name: zitadel-db
    environment:
      - 'POSTGRES_USER=postgres'
      - 'POSTGRES_PASSWORD=PasswortFuerDenPostgresChef'
    volumes:
      - /root/Docker/zitadel/data:/var/lib/postgresql/data
    healthcheck:
      test: ["CMD-SHELL", "pg_isready", "-d", "db_prod"]
      interval: '10s'
      timeout: '30s'
      retries: 5
      start_period: '20s'
    restart: unless-stopped
```

Anschließend den Container einfach über *sudo docker-compose up -d* neu erstellen lassen.

> Nicht vergessen: Im Reverse Proxy ist natürlich Zitadel mit der dazugehörigen Domäne zu hinterlegen und an diesen Container weiterzuleiten.

Danach ist die Weboberfläche unter folgender Adresse erreichbar: https://beispiel.de/ui/console

Durch die Domäne verändert sich der Standard-Login wie folgt:
Benutzername: zitadel-admin@zitadel.beispiel.de
Passwort: Password1!