

SELF-HOSTING > INSTALLATIONS- & BEREITSTELLUNGSANLEITUNGEN >

Vereinheitlichte Bereitstellung Beta

Ansicht im Hilfezentrum:

<https://bitwarden.com/help/install-and-deploy-unified-beta/>

Vereinheitlichte Bereitstellung Beta

Note

Diese Lösung befindet sich in der Beta-Phase und ist für den persönlichen Gebrauch bestimmt. Geschäftspläne sollten die offiziell unterstützte, standardmäßige Bereitstellungsoption verwenden.

Während sich die selbstgehostete Bereitstellung von Bitwarden Unified in der Betaphase befindet, **sollten diejenigen, die Unified installieren, keine** automatischen Upgrade-Verfahren einrichten, die die neuesten verfügbaren Images abrufen. Bitwarden empfiehlt, etwas Zeit für die Stabilisierung einer Veröffentlichung zu lassen, bevor Sie upgraden.

[Erfahren Sie, wie Sie Probleme melden](#).

Dieser Artikel führt Sie durch die Installation und den Start der einheitlichen, selbst gehosteten Bitwarden-Bereitstellung. Verwenden Sie diese Bereitstellungsmethode, um:

- Vereinfachen Sie die Konfiguration und optimieren Sie die Ressourcennutzung (CPU, Speicher) durch die Bereitstellung von Bitwarden mit einem einzigen Docker-Image.
- Nutzen Sie verschiedene Datenbanklösungen wie MSSQL, PostgreSQL, SQLite und MySQL/MariaDB.
- Laufen Sie auf ARM-Architektur für alternative Systeme wie Raspberry Pi und NAS-Server.

Systemanforderungen

Die einheitliche Bereitstellung von Bitwarden erfordert:

- Mindestens 200 MB RAM
- Speicher 1GB
- Docker Engine 19+

Docker installieren

Die einheitliche Bereitstellung wird auf Ihrem Rechner in einem [Docker-Container](#) ausgeführt. Die einheitliche Bereitstellung kann mit jeder Docker-Ausgabe oder jedem Plan ausgeführt werden. Bewerten Sie, welche Ausgabe am besten für Ihre Installation geeignet ist.

Installieren Sie Docker auf Ihrem Gerät, bevor Sie mit der Installation fortfahren. Beziehen Sie sich auf die folgende Docker-Dokumentation für Hilfe:

- [Installieren Sie Docker Engine](#)

Führen Sie Bitwarden vereinheitlicht aus

Die einheitliche Bereitstellung kann mit dem Befehl `docker run` ausgeführt werden (siehe [hier](#)) oder mit Docker Compose (siehe [hier](#)). In beiden Fällen müssen Sie Umgebungsvariablen für den Container angeben.

Umgebungsvariablen festlegen

Das Ausführen der einheitlichen Bereitstellung erfordert, dass Umgebungsvariablen für den Container festgelegt werden.

Umgebungsvariablen können durch Erstellen einer `settings.env` Datei festgelegt werden, von der Sie ein Beispiel in unserem [GitHub Repository](#) finden können, oder durch Verwendung der `--env` Flagge, wenn Sie die `docker run` Methode verwenden. Mehrere optionale Variablen stehen zur Verfügung, um ein personalisierteres, einheitliches Bereitstellungserlebnis zu ermöglichen. Zusätzliche Details zu diesen Variablen können [hier](#) gefunden werden.

Setzen Sie mindestens Werte für die Variablen, die unter dem Abschnitt **# Erforderliche Einstellungen #** der Beispiel-`.env`-Datei fallen:

Variable	Beschreibung
BW-DOMAIN	Ersetzen Sie <code>bitwarden.yourdomain.com</code> durch die Domain, auf die Bitwarden zugreifen wird.
BW_DB_ANBIETER	Der Datenbankanbieter, den Sie für Ihren Bitwarden-Server verwenden werden. Verfügbare Optionen sind <code>sqlserver</code> , <code>postgresql</code> , <code>sqlite</code> oder <code>mysql/mariadb</code> .
BW_DB_SERVER	Der Name des Servers, auf dem Ihre Datenbank läuft.
BW_DB_DATENBANK	Der Name Ihrer Bitwarden-Datenbank.
BW_DB_BENUTZERNAME	Der Benutzername für den Zugriff auf die Bitwarden-Datenbank.
BW_DB_PASSWORT	Das Passwort für den Zugriff auf die Bitwarden-Datenbank.
BW_DB_DATEI	Nur erforderlich für <code>sqlite</code> , wenn Sie den Pfad zu Ihrer Datenbankdatei angeben möchten. Wenn nicht angegeben, wird <code>sqlite</code> automatisch eine <code>vault.db</code> Datei unter dem <code>/etc/bitwarden</code> Volume erstellen.
BW_INSTALLATIONS_ID	Eine gültige Installations-ID, generiert von https://bitwarden.com/host/ .
BW_INSTALLATIONSSCHLÜSSEL	Ein gültiger Installations-Schlüssel, generiert von https://bitwarden.com/host/ .

Note

Im Gegensatz zur Standardbereitstellung von Bitwarden kommt die einheitliche Bereitstellung nicht direkt mit einer Datenbank. Sie können eine vorhandene Datenbank verwenden oder eine neue erstellen, wie in [diesem Beispiel](#) dokumentiert, und in beiden Fällen müssen Sie gültige Informationen in die hier dokumentierten `BW_DB_...` Variablen eingeben.

Die Verwendung von Nicht-MSSQL-Datenbankanbietern kann zu Leistungsproblemen führen, da die Unterstützung für diese Plattformen während der gesamten Beta-Phase weiterhin bearbeitet wird. Bitte verwenden Sie [diese Vorlage für Probleme](#), um alles in Bezug auf Ihre einheitliche Bitwarden-Bereitstellung zu berichten und schauen Sie sich [diese Seite](#) an, um bekannte Probleme zu verfolgen oder an der Diskussion teilzunehmen.

Mit Docker ausführen

Die einheitliche Bereitstellung kann mit dem Befehl `docker run` ausgeführt werden, wie im folgenden Beispiel:

Bash

```
docker run -d --name bitwarden -v /$(pwd)/bwdata:/etc/bitwarden -p 80:8080 --env-file settings.env bitwarden/self-host:beta
```

Der oben angezeigte Befehl hat mehrere erforderliche Optionen für den `docker run` Befehl, einschließlich:

Name, Kurzform	Beschreibung
<code>--abtrennen, -d</code>	Führen Sie den Container im Hintergrund aus und drucken Sie die Container-ID.
<code>--Name</code>	Geben Sie einen Namen für den Container an. <code>Bitwarden</code> wird im Beispiel verwendet.
<code>--Lautstärke, -v</code>	Binden Sie ein Volume. Mindestens, mounten Sie <code>/etc/bitwarden</code> .
<code>--veröffentlichen, -v</code>	Karten Sie Container-Ports auf den Host zu. Das Beispiel zeigt den gemappten Port <code>80:8080</code> . Port 8443 ist erforderlich, wenn SSL konfiguriert wird.
<code>--env-Datei</code>	Pfad der Datei, aus der Umgebungsvariablen gelesen werden sollen. Alternativ verwenden Sie die <code>--env</code> Flagge, um Umgebungsvariablen inline zu deklarieren (mehr erfahren).

Sobald Sie den Befehl ausgeführt haben, überprüfen Sie, ob der Container läuft und gesund ist mit:

Bash

```
docker ps
```

Gratulation! Ihre einheitliche Bereitstellung ist jetzt unter <https://your.domain.com> in Betrieb. Besuchen Sie den Web-Tresor in Ihrem Browser, um zu bestätigen, dass er funktioniert. Sie können sich jetzt ein neues Konto registrieren und anmelden.

Mit Docker Compose verwenden

Das Ausführen der einheitlichen Bereitstellung mit Docker Compose erfordert Docker Compose Version 1.24 oder höher. Um die einheitliche Bereitstellung mit Docker Compose auszuführen, erstellen Sie eine `docker-compose.yml` Datei, zum Beispiel:

Bash

```
---
version: "3.8"

services:
  bitwarden:
    depends_on:
      - db
    env_file:
      - settings.env
    image: bitwarden/self-host:beta
    restart: always
    ports:
      - "80:8080"
    volumes:
      - bitwarden:/etc/bitwarden

  db:
    environment:
      MARIADB_USER: "bitwarden"
      MARIADB_PASSWORD: "super_strong_password"
      MARIADB_DATABASE: "bitwarden_vault"
      MARIADB_RANDOM_ROOT_PASSWORD: "true"
    image: mariadb:10
    restart: always
    volumes:
      - data:/var/lib/mysql

volumes:
  bitwarden:
  data:
```

In der `docker-compose.yml` Datei, nehmen Sie alle gewünschten Konfigurationen vor, einschließlich:

- Zuordnung von Volumen für Protokolle und Bitwarden-Daten.
- Ports mappen.

- Konfigurieren eines Datenbankbildes.^a

^a Richten Sie eine Datenbank nur in `docker-compose.yml` ein, wie im obigen Beispiel, wenn Sie **einen neuen Datenbankserver zur Verwendung mit Bitwarden erstellen** möchten. Beispielkonfigurationen für MySQL, MSSQL und PostgreSQL sind in unserer [Beispieldatei](#) enthalten.

Sobald Ihre `docker-compose.yml` und `settings.env` Datei erstellt sind, starten Sie Ihren vereinheitlichten Server durch Ausführen von:

Bash

```
docker compose up -d
```

Überprüfen Sie, ob alle Container korrekt laufen:

Bash

```
docker ps
```

Gratulation! Ihre einheitliche Bereitstellung ist jetzt unter <https://your.domain.com> in Betrieb. Besuchen Sie den Web-Tresor in Ihrem Browser, um zu bestätigen, dass er funktioniert. Sie können sich jetzt ein neues Konto registrieren und anmelden.

Aktualisieren Sie Ihren Server

Um Ihre einheitliche Bereitstellung zu aktualisieren:

⇒ Docker laufen Aktualisierung

1. Stoppen Sie den laufenden Docker-Container:

Bash

```
docker stop bitwarden
```

2. Entfernen Sie den Docker-Container:

Bash

```
docker rm bitwarden
```

3. Führen Sie den folgenden Befehl aus, um das aktuellste Bitwarden Unified-Image zu ziehen:

Bash

```
docker pull bitwarden/self-host:beta
```

4. Starten Sie den Docker-Container erneut:

Bash

```
docker run -d --name bitwarden -v /$(pwd)/bwdata:/etc/bitwarden -p 80:8080 --env-file settings.  
env bitwarden/self-host:beta
```

⇒ Docker Compose Aktualisierung

1. Stoppen Sie den laufenden Docker-Container:

Bash

```
docker compose down
```

2. Führen Sie den folgenden Befehl aus, um das aktuellste Bitwarden Unified-Image zu ziehen:

Bash

```
docker compose pull
```

3. Erstellen Sie alle Behälter neu, die eine Aktualisierung benötigen:

Bash

```
docker compose up -d
```

4. Überprüfen Sie, ob die Container laufen:

Bash

```
docker compose ps
```

Umgebungsvariablen

Die einheitliche Bereitstellung wird standardmäßig ohne mehrere der standardmäßigen Bitwarden-Dienste betrieben. Dies ermöglicht eine erhöhte Anpassung und Optimierung Ihrer einheitlichen Bereitstellung. Konfigurieren Sie diese Dienste und weitere optionale Einstellungen, indem Sie verschiedene Umgebungsvariablen bearbeiten.

Note

Immer wenn Sie eine Umgebungsvariable ändern, muss der Docker-Container neu erstellt werden. Mehr dazu erfahren Sie [hier](#).

Webserver-Ports

Variable	Beschreibung
BW_PORT_HTTP	Ändern Sie den Port, der für HTTP-Verkehr verwendet wird. Standardmäßig, 8080 .
BW_PORT_HTTPS	Ändern Sie den Port, der für HTTPS-Verkehr verwendet wird. Standardmäßig, 8443 .

SSL

Verwenden Sie diese Werte, um die Zertifikatseinstellungen zu ändern.

Variable	Beschreibung
BW_ENABLE_SSL	Verwenden Sie SSL/TLS. wahr/falsch . Standard falsch . SSL ist erforderlich, damit Bitwarden ordnungsgemäß funktioniert. Wenn Sie SSL nicht im Bitwarden-Container konfiguriert haben, sollten Sie Bitwarden mit einem SSL-Proxy voranstellen.
BW_SSL_ZERTIFIKAT	Der Name Ihrer SSL-Zertifikatdatei. Die Datei muss sich im /etc/bitwarden Verzeichnis innerhalb des Containers befinden. Standard ssl.crt .
BW_SSL_SCHLÜSSEL	Der Name Ihrer SSL-Schlüsseldatei. Die Datei muss sich im /etc/bitwarden Verzeichnis innerhalb des Containers befinden. Standard ssl.key .
BW_ENABLE_SSL_CA	Verwenden Sie SSL mit einem von einer Zertifizierungsstelle (CA) unterstützten Dienst. wahr/falsch . Standard falsch .

Variable	Beschreibung
BW_SSL_CA_ZERTIFIKAT	Der Name Ihres SSL CA-Zertifikats. Die Datei muss sich im <code>/etc/bitwarden</code> Verzeichnis innerhalb des Containers befinden. Standard <code>ca.crt</code> .
BW_ENABLE_SSL_DH	Verwenden Sie SSL mit Diffie-Hellman-Schlüsselaustausch. <code>wahr/falsch</code> . Standardmäßig <code>falsch</code> .
BW_SSL_DH_ZERTIFIKAT	Der Name Ihrer Diffie-Hellman-Parameterdatei. Die Datei muss sich im <code>/etc/bitwarden</code> Verzeichnis innerhalb des Containers befinden. Standard <code>dh.pem</code> .
BW_SSL_PROTOCOLS	SSL-Version, die von NGINX verwendet wird. Für empfohlene Standardeinstellungen leer lassen. Erfahren Sie mehr .
BW_SSL_CIPHERS	SSL-Ciphersuites, die von NGINX verwendet werden. Für empfohlene Standardeinstellungen leer lassen. Erfahren Sie mehr .

Note

Wenn Sie ein bestehendes SSL-Zertifikat verwenden, müssen Sie die entsprechenden SSL-Optionen in `settings.env` aktivieren. SSL-Dateien müssen in `/etc/bitwarden` gespeichert werden, auf die in der `docker-compose.yml` Datei verwiesen werden kann. Diese Dateien müssen den Namen entsprechen, die in `settings.env` konfiguriert sind.

Das Standardverhalten besteht darin, ein selbstsigniertes Zertifikat zu generieren, wenn SSL aktiviert ist und keine vorhandenen Zertifikatdateien an der erwarteten Stelle vorhanden sind (`/etc/bitwarden`).

Dienste

Zusätzliche Dienste können für spezifische Anwendungsfälle, wie Enterprise- oder Teambedürfnisse, durch Änderung der folgenden Werte aktiviert oder deaktiviert werden:

Variable	Beschreibung
BW_ENABLE_ADMIN	<p>Deaktivieren Sie diesen Dienst nicht.</p> <p>Erfahren Sie mehr über die Fähigkeiten des Administrator-Panels hier. Standard <code>wahr</code>.</p>

Variable	Beschreibung
BW_ENABLE_API	<p>Deaktivieren Sie diesen Dienst nicht.</p> <p>Standard wahr.</p>
BW_ENABLE_EVENTS	<p>Aktivieren oder deaktivieren Sie Bitwarden Ereignisprotokolle für Teams und Enterprise-Ereignisüberwachung. Standard falsch.</p>
BW_ENABLE_ICONS	<p>Aktivieren oder deaktivieren Sie Bitwarden-Markensymbole, die mit den URI's der Zugangsdaten-Einträge festgelegt sind. Mehr dazu erfahren Sie hier. Standard wahr.</p>
BW_ENABLE_IDENTITY	<p>Deaktivieren Sie diesen Dienst nicht.</p> <p>Standard wahr.</p>
BW_AKTIVIERE_BENACHRICHTIGUNGEN	<p>Aktivieren oder deaktivieren Sie Benachrichtigungsdienste, um Push-Benachrichtigungen auf mobile Geräte zu erhalten, mit Geräte-Zugangsdaten, mobiler Tresor-Synchronisation und mehr. Standard wahr.</p>
BW_ENABLE_SCIM	<p>Aktivieren oder deaktivieren Sie SCIM für Enterprise-Organisationen. Standardmäßig falsch.</p>
BW_ENABLE_SSO	<p>Aktivieren oder deaktivieren Sie SSO-Dienste für Enterprise-Organisationen. Standard falsch.</p>
BW_ICONS_PROXY_ZU_CLOUD	<p>Die Aktivierung dieses Dienstes leitet Anfragen für den Icon-Dienst über Cloud-Dienste um, um die System-Speicherlast zu verringern. Wenn Sie sich für diese Einstellung entscheiden, sollte BW_ENABLE_ICONS auf falsch gesetzt werden, um die Containerlast zu reduzieren. Standard falsch.</p>

Post

Konfigurieren Sie die SMTP-Einstellungen für Ihre einheitliche Bereitstellung. Kopieren Sie Informationen von Ihrem gewählten Mail SMTP-Anbieter in die folgenden Felder:

Variable	Beschreibung
<code>globalSettings__mail__AntwortAnEmail</code>	Geben Sie die Antwort-E-Mail-Adresse für Ihren Server ein.
<code>globalSettings__mail__smtp__host</code>	Geben Sie die Domain des Hosts für Ihren SMTP-Server ein.
<code>globalSettings__mail__smtp__port</code>	Geben Sie die Port-Nummer vom SMTP-Host ein.
<code>globaleEinstellungen__Mail__smtp__ssl</code>	Wenn Ihr SMTP-Host SSL verwendet, geben Sie true ein. Setzen Sie den Wert auf falsch , wenn Ihr Host den TLS-Dienst verwendet.
<code>globalSettings__mail__smtp__benutzername</code>	Geben Sie den SMTP-Benutzername ein.
<code>globalSettings__mail__smtp__passwort</code>	Geben Sie das SMTP-Passwort ein.

Yubico API (YubiKey)

Variable	Beschreibung
<code>globaleEinstellungen__yubico__clientId</code>	Ersetzen Sie den Wert durch die ID, die Sie von Ihrem Yubico Key erhalten haben. Melden Sie sich für Yubico Key hier an.
<code>globaleEinstellungen__yubico__Schlüssel</code>	Geben Sie den Schlüsselwert ein, den Sie von Yubico erhalten haben.

Datenbankkonfigurationen

Die Nutzung der Vielzahl von Datenbankoptionen, die mit der einheitlichen Bereitstellung kompatibel sind, erfordert zusätzliche **.env** Konfigurationen.

⇒MySQL/MariaDB

In **settings.env**:

Bash

```
# Database
BW_DB_PROVIDER=mysql
BW_DB_SERVER=db
BW_DB_DATABASE=bitwarden_vault
BW_DB_USERNAME=bitwarden
BW_DB_PASSWORD=super_strong_password
```

⇒MSSQLIn `settings.env`:**Bash**

```
# Database
BW_DB_PROVIDER=sqlserver
BW_DB_SERVER=db
BW_DB_DATABASE=bitwarden_vault
BW_DB_USERNAME=bitwarden
BW_DB_PASSWORD=super_strong_password
```

⇒SQLiteIn `settings.env`:**Bash**

```
# Database
BW_DB_PROVIDER=sqlite
BW_DB_FILE=/path/to/.db
```

Die Zuweisung des `sqlite` Werts wird automatisch eine `vault.db` Datei im `/etc/bitwarden` Volume erstellen. `BW_DB_FILE` wird nur benötigt, wenn Sie den Pfad zu einer anderen Datenbankdatei angeben möchten.

⇒PostgreSQLIn `settings.env`:

Bash

Database

BW_DB_PROVIDER=postgresql

BW_DB_SERVER=db

BW_DB_DATABASE=bitwarden_vault

BW_DB_USERNAME=bitwarden

BW_DB_PASSWORD=super_strong_password

Sonstiges**Variable**

globaleEinstellungen__BenutzerregistrierungDeaktivieren

globaleEinstellungen__hibpApiKey

adminEinstellungen__Admins

BW_REAL_IPS

BW_CSP

BW_DB_PORT

Beschreibung

Aktivieren oder deaktivieren Sie die Registrierungsfähigkeiten für Benutzerkonten.

Geben Sie den API-Schlüssel ein, der von Have I Been Pwnd bereitgestellt wurde. Registrieren Sie sich, um den API-Schlüssel [hier](#) zu erhalten.

Geben Sie die E-Mail-Adressen des Administrators ein.

Definieren Sie echte IPs in [nginx.conf](#) in einer durch Kommas getrennten Liste. Nützlich zur Definition von Proxy-Servern, die die IP-Adresse des Clients weiterleiten. [Erfahren Sie mehr.](#)

Content-Security-Policy-Parameter. Eine Neukonfiguration dieses Parameters kann Funktionen beschädigen. Durch die Änderung dieses Parameters übernehmen Sie die Verantwortung für die Aufrechterhaltung dieses Wertes.

Geben Sie einen benutzerdefinierten Port für den Datenbankverkehr an. Wenn nicht angegeben, hängt der Standard von Ihrem gewählten Datenbankanbieter ab.

Starten Sie den Container neu

Um Ihren Docker-Container nach dem Ändern von Umgebungsvariablen neu zu starten, führen Sie die folgenden Befehle aus dem Bitwarden-einheitlichen Bereitstellungsverzeichnis aus:

⇒ Docker ausführen

1. Stoppen Sie den laufenden Docker-Container:

Bash

```
docker stop bitwarden
```

2. Entfernen Sie den Docker-Container:

Bash

```
docker rm bitwarden
```

3. Starten Sie den Docker-Container erneut:

Bash

```
docker run -d --name bitwarden -v /$(pwd)/bwdata:/etc/bitwarden -p 80:8080 --env-file settings.env bitwarden/self-host:beta
```

⇒ Docker Compose

1. Stoppen Sie den laufenden Docker-Container:

Bash

```
docker compose down
```

2. Erstelle die Container neu:

Bash

```
docker compose up -d
```

3. Stellen Sie sicher, dass die Container mit folgendem Befehl ordnungsgemäß laufen:

```
Bash
```

```
docker compose ps
```

Speicherverbrauch

Standardmäßig wird der Bitwarden-Container den verfügbaren Speicher verbrauchen, oft mehr als das Minimum, das zum Laufen benötigt wird. Für speicherbewusste Umgebungen können Sie `docker -m` oder `--memory=` verwenden, um die Speichernutzung des Bitwarden-Containers zu begrenzen.

Name, Kurzform	Beschreibung
<code>--speicher=, -s</code>	Die maximale Menge an Speicher, die der Container verwenden kann. Bitwarden erfordert mindestens 200m. Siehe die Docker-Dokumentation , um mehr zu erfahren.

Um den Speicherverbrauch mit Docker Compose zu steuern, verwenden Sie den Schlüssel `mem_limit`:

```
Bash
```

```
services:  
  bitwarden:  
    env_file:  
      - settings.env  
    image: bitwarden/self-host:beta  
    restart: always  
    mem_limit: 200m
```

Berichterstattung über Probleme

Während die einheitliche Bereitstellung von Bitwarden weiterhin in der Beta-Version bleibt, ermutigen wir Sie, Probleme zu melden und Feedback über GitHub zu geben. Bitte verwenden Sie [diese Vorlage für Probleme](#), um alles in Bezug auf Ihre einheitliche Bitwarden-Bereitstellung zu berichten und schauen Sie sich [diese Seite](#) an, um bekannte Probleme zu verfolgen oder an der Diskussion teilzunehmen.

Zusätzliche Ressourcen

- Wenn Sie planen, eine Bitwarden Organisation selbst zu hosten, sehen Sie [eine Organisation selbst hosten](#) um zu beginnen.

Für weitere Informationen zur standardmäßigen selbst gehosteten Bereitstellung von Bitwarden sehen Sie:

- [Installieren und Bereitstellen - Linux](#)
- [Installieren und Bereitstellen - Windows](#)

- [Installieren und Bereitstellen - Handbuch](#)