

EnterpriseDB PostgreSQL with TDE Support @OL8 (Oracle Linux 8)

Pre-requisites

In my setup, I planned to virtualize VM in Parallels on Macintosh with M1 processor, because of this, I have chosen `arm64` version. But there is no RPM repository from EnterpriseDB PostgreSQL for ARM architecture. I decided to switch to another virtualization platform on `x64` architecture and proceed with `amd64` version..

Download linux distribution from Oracle

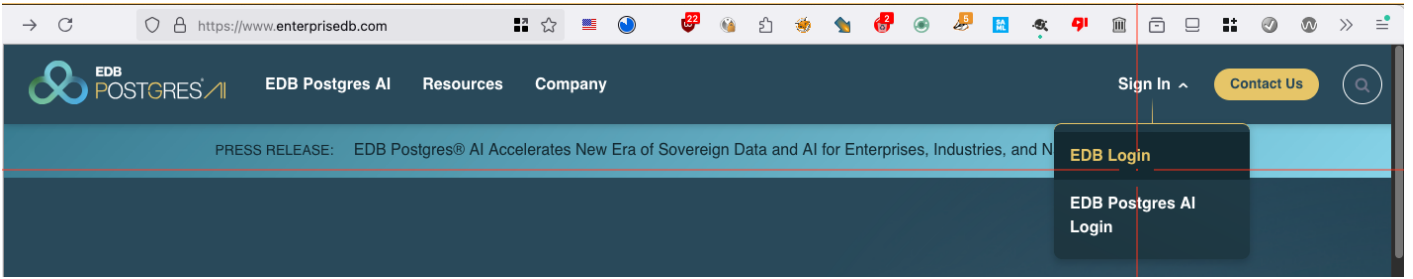
<https://yum.oracle.com/oracle-linux-isos.html>

https://yum.oracle.com/ISOS/OracleLinux/OL8/u10/x86_64/OracleLinux-R8-U10-x86_64-dvd.iso

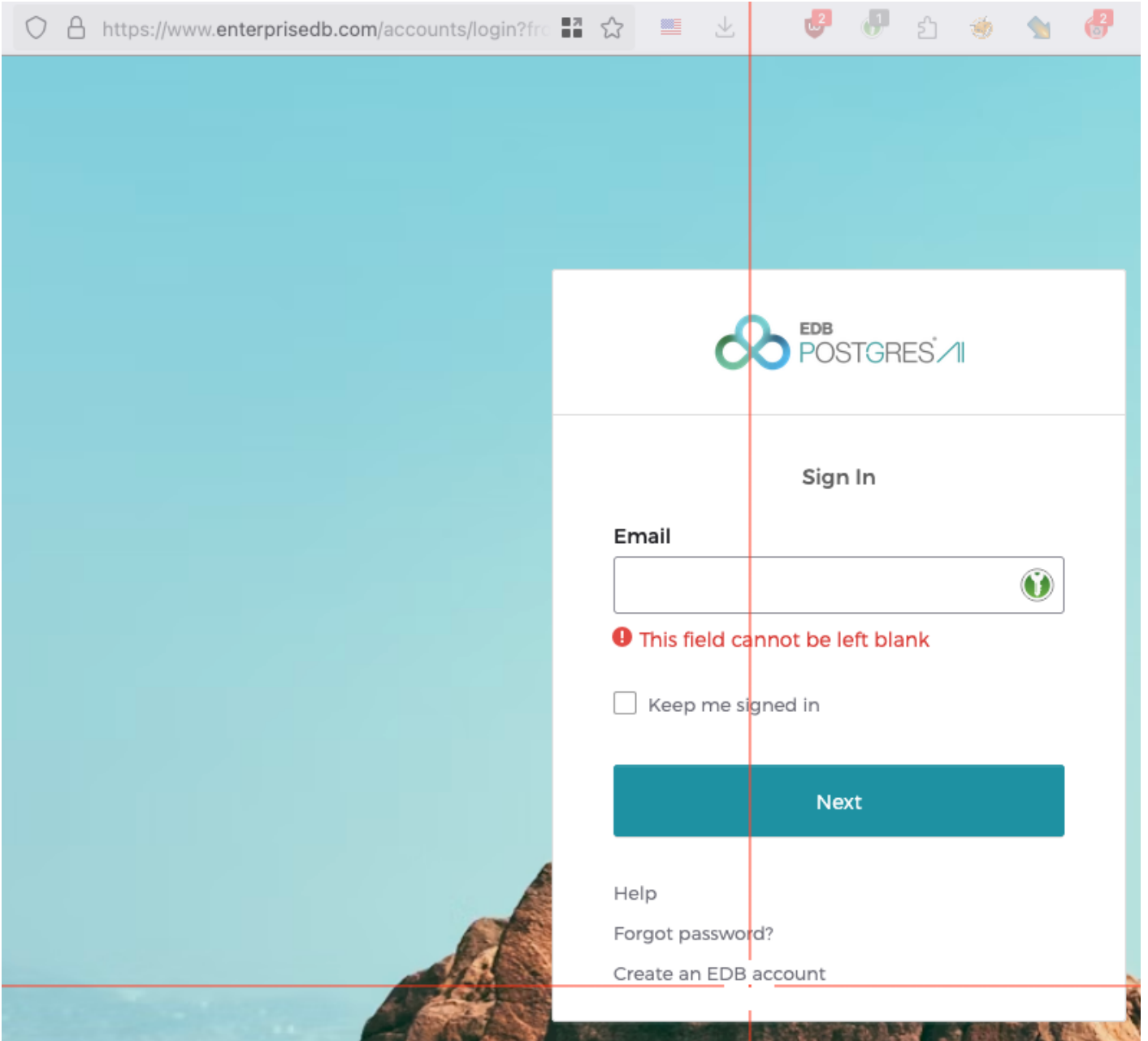
Create VM, do updates.

Create an account with
EnterpriseDB

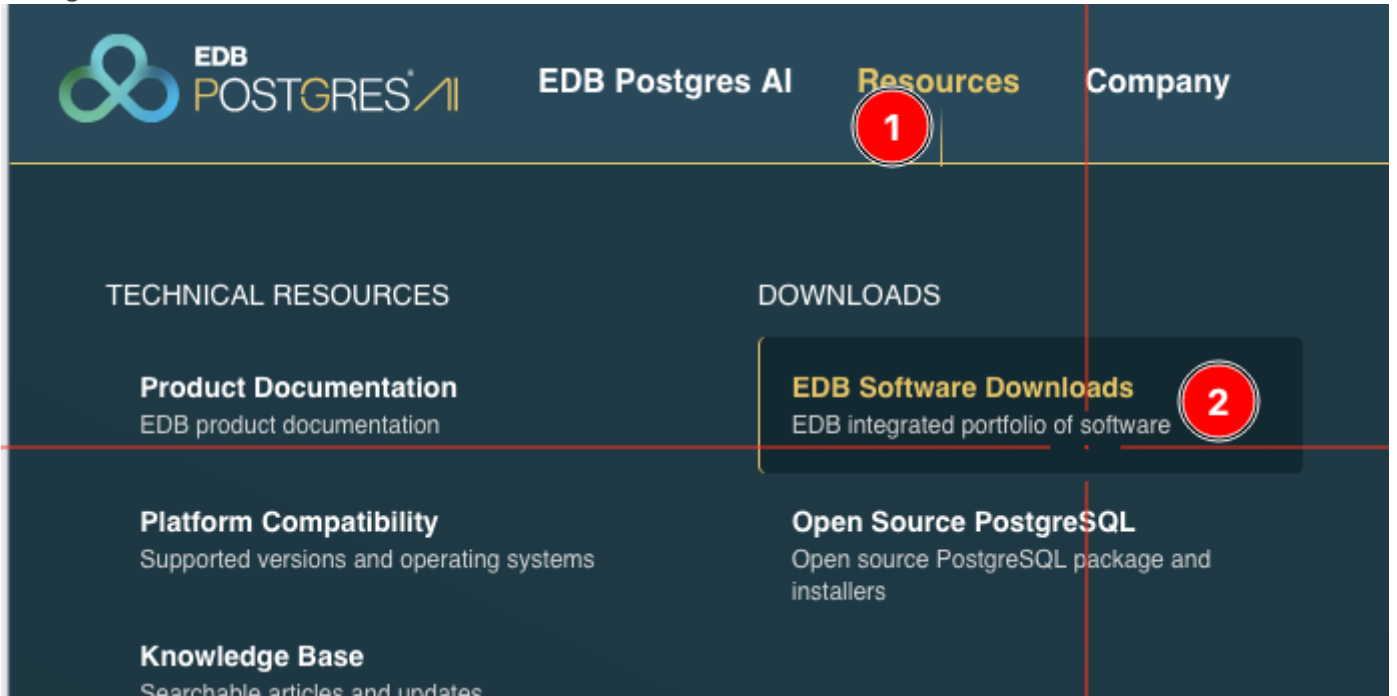
<https://www.enterprisedb.com/>



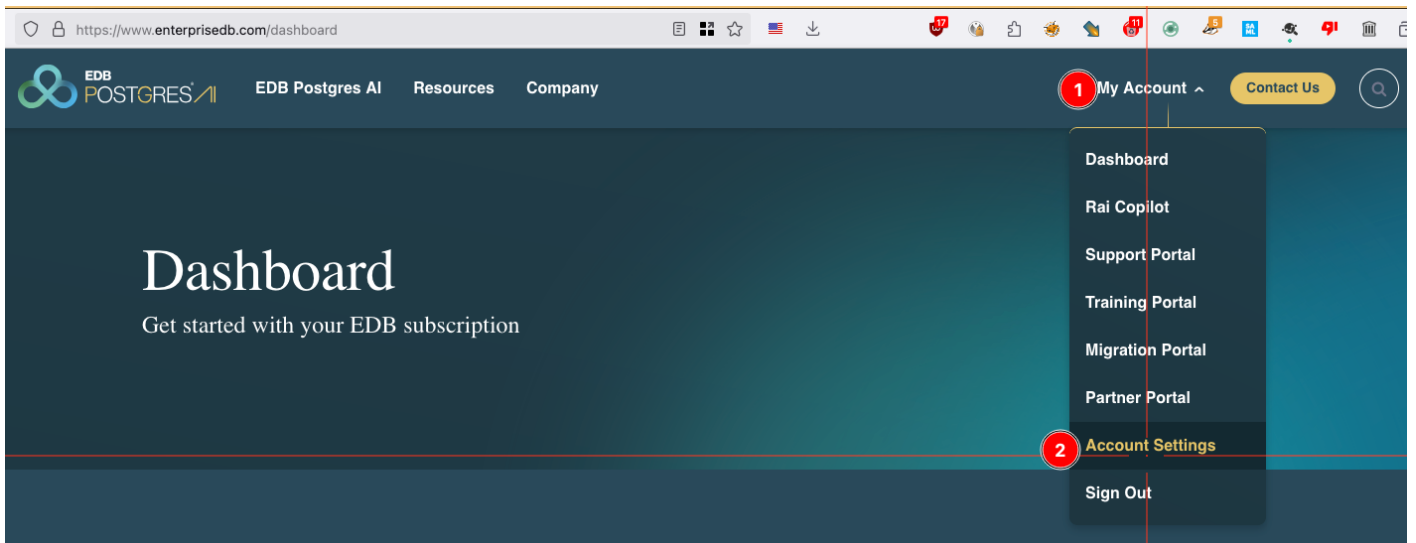
Navigate to



Navigate to



Find repo token





EDB Account

[EDB Postgres AI](#) · [Dashboard](#) · [Repo Access](#)

[Sign out](#)

Profile

Security

Repos 2.0 token



First Name

Anton

Last Name

[Redacted]

Work E-mail

[Redacted]

Business Phone

[Redacted]

Company

[Redacted]

Country

[Redacted]

[Edit](#)

Copy past a token.

Configure repositories

ref

```
https://www.enterprisedb.com/docs/pgd/latest/essential-how-to/install/02-configure-repositories/
```

```
export EDB_SUBSCRIPTION_TOKEN=<your-token>
export EDB_REPO_TYPE=rpm

curl -1sSLf
"https://downloads.enterprisedb.com/$EDB_SUBSCRIPTION_TOKEN/enterprise/setup.$EDB_REPO_TYPE.sh" | sudo
-E bash
# curl -1sSLf
"https://downloads.enterprisedb.com/$EDB_SUBSCRIPTION_TOKEN/postgres_distributed/setup.$EDB_REPO_TYPE.
sh" | sudo -E bash
```

Installation of EnterpriseDB PostgreSQL Advanced Server

ref

```
# Extended (PGE)
https://www.enterprisedb.com/docs/pge/latest/installing/linux\_x86\_64/pge\_rhel\_9/

# Advanced (EPAS)
https://www.enterprisedb.com/docs/epas/latest/installing/linux\_x86\_64/epas\_rhel\_9/
```

Add repository to the operating system

```
sudo dnf install -y https://yum.enterprisedb.com/edbrepos/edb-repo-latest.noarch.rpm
```

Verify repositories are added

```
dnf repolist | grep enterprisedb
```

Enable required modules

```
sudo dnf module enable -y edb-postgresql15
```

Install EDB PostgreSQL with TDE Support

```
sudo dnf install -y edb-as15-server edb-as15-tde
```

Initialization of the Database with Transparent Data Encryption (TDE)

Create a directory for the key:

```
sudo mkdir -p /var/lib/edb/as15/data
sudo mkdir -p /var/lib/edb/as15/keys
sudo chown -R enterprisedb:enterprisedb /var/lib/edb/as15/
```

Create the master key:

```
sudo su - enterprisedb
openssl rand -hex 32 > /var/lib/edb/as15/keys/efm.key
chmod 600 /var/lib/edb/as15/keys/efm.key
```

Export the key location:

```
export EDB_TDE_KEY=/var/lib/edb/as15/keys/efm.key
```

Initialize with UTF-8 encoding enabled and TDE:

```
PGSETUP_INITDB_OPTIONS="-E UTF-8" /usr/edb/as15/bin/initdb -D /var/lib/edb/as15/data --data-encryption
```

Configure the Server to Use the Key

Add the following to ~/.bash_profile or systemd environment:

```
export EDB_TDE_KEY=/var/lib/edb/as15/keys/efm.key
```

For systemd service: create or edit drop-in file:

```
sudo systemctl edit edb-as-15
```

Add:

```
[Service]
Environment="EDB_TDE_KEY=/var/lib/edb/as15/keys/efm.key"
```

Refresh daemon and restart the service

```
sudo systemctl daemon-reexec
sudo systemctl daemon-reload
sudo systemctl start edb-as-15
sudo systemctl enable edb-as-15
sudo systemctl status edb-as-15
```

Check that server is listening for connections

```
ss -ntap | grep 5432
```

Confirm that database server is running and TDE is active

Execute as enterprisedb user:

```
sudo su - enterprisedb
psql edb
```

```
SHOW data_encryption;
```

Change admin password

```
ALTER ROLE enterprisedb IDENTIFIED BY password;
```

Conclusion

- Database deployed.
- TDE enabled
- Backup master key. Once key is lost, database could not be opened.
- Ensure it has 400 permissions only, as any other keys.
- Proceed with normal database creation.

Testing

Create db, table, insert and select.

```
sudo su - enterprisedb
psql edb

CREATE DATABASE hr;
\c hr

CREATE TABLE public.dept (deptno numeric(2)
    NOT NULL CONSTRAINT dept_pk PRIMARY KEY,
    dname varchar(14) CONSTRAINT dept_dname_uq UNIQUE,
    locvarchar(13));
INSERT INTO dept VALUES (10,'ACCOUNTING','Wonderland');
INSERT into dept VALUES (20,'RESEARCH','Intercity');

SELECT * FROM dept;
```

Revision #1

Created 24 July 2025 07:42:28 by Anton

Updated 24 July 2025 09:52:27 by Anton