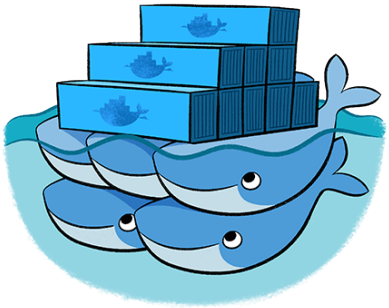
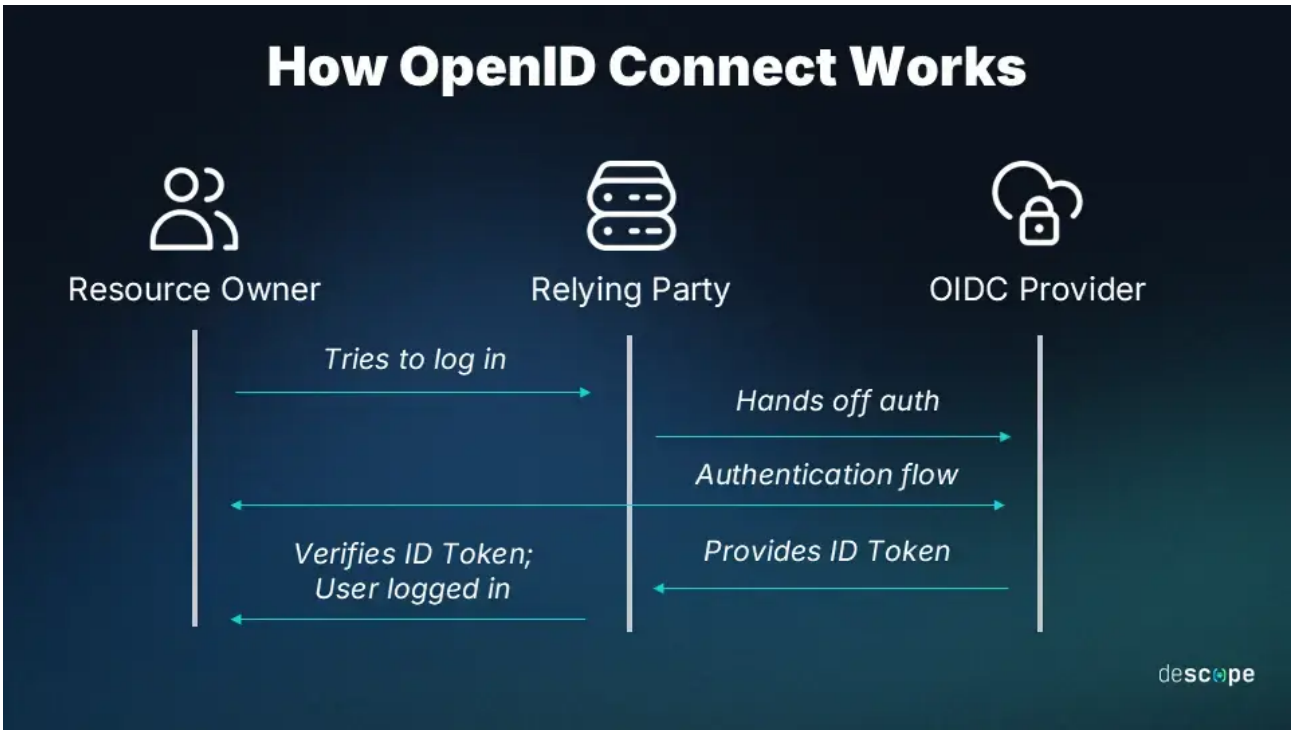
What is OIDC?



Overview

> This page shows **one** example use case of OpenID Connect (OIDC), using Traefik as a reverse proxy, Authelia as an identity provider (OIDC Provider), and Linkwarden (the "Relying Party"). Please note that OIDC supports **many** different use cases and flow types—this walkthrough is just to demonstrate **one** approach.



Example OIDC Flow

1. **User goes to the Relying Party (Linkwarden).**
   The user attempts to access Linkwarden, which is behind Traefik. Because Linkwarden requires authentication, the user must log in.
2. **User chooses to log in with the OIDC Provider (Authelia).**
   When the user selects a “Login with Authelia” option, Linkwarden (through Traefik) redirects the user to Authelia.
3. **User gets redirected to the OIDC Provider (Authelia).**
   The browser is sent to Authelia’s login page.
4. **User logs in with the OIDC Provider (Authelia).**
   Authelia verifies the user’s credentials (for example, via LDAP, a local user database, or some other method).
5. **OIDC Provider (Authelia) generates an ID Token (JWT).**
   - This ID Token contains “claims” (such as username, groups, and email) based on the **scopes** defined in Authelia’s configuration.
   - Authelia signs the ID Token (it is a JWT) before sending it back to Linkwarden (the Relying Party).

   Here’s a simple table of possible scopes and example claim data:

Scope	Claim
Profile	Name
Groups	Groupa, Groupb, Groupc
Email	test @gmail.com

6. **Relying Party (Linkwarden) reads the ID Token to grant access.**
   Linkwarden looks at the returned claims within the token (only what was allowed by the configured OIDC scopes) and decides whether to allow the user in. It then notifies the user's browser that login was successful.
7. **User is logged in.**
   The user is now recognized as authenticated in Linkwarden.

What Are OIDC Scopes?

OIDC **scopes** determine what information the Relying Party can request (and potentially receive) about the user. Typical scopes include:

- **openid**
  Required for OIDC; indicates that the client (Relying Party) intends to use OIDC to verify user identity.
- **groups**
  Allows access to group membership claims (e.g., Groupa, Groupb).
- **email**
  Gives the relying service access to the user's email address (if available).
- **profile**
  Allows for basic profile details, such as name or preferred username.

For example, if Linkwarden requests the scopes:

```
- openid
- groups
- email
- profile
```

it may receive your group memberships, email address, and display name in the returned ID Token.

What Is a JWT?

A **JWT** (JSON Web Token) is the format often used to transmit information securely between parties as a JSON object:

- **hmac_Secret** = A random secret known only to Authelia (or the OIDC Provider).

- **JSON Web Token** = The data payload + a signature + a header.

  You can inspect or verify a JWT at jwt.io to ensure nobody has modified the data.
- **Important**: JWTs are **not** encrypted by default. They are **signed** to ensure the content hasn't been tampered with, but anyone who has the token can read the data inside. If encryption is needed, an additional layer (e.g., HTTPS in transit or encrypted tokens at rest) must be used.

Conclusion

Using OIDC with Traefik, Authelia, and Linkwarden is just one practical illustration of OpenID Connect flows. Authelia serves as the OIDC Provider, creating JWT-based ID Tokens. Linkwarden (the Relying Party) receives these signed tokens, reads the claims (like email or group memberships), and grants access. You can tailor the scopes and claims for your setup, making this flow flexible and secure for various applications.

Revision #12
Created 1 February 2025 20:07:38 by aeoneros
Updated 4 February 2025 14:22:53 by aeoneros