

TozID Powered OIDC Implicit Flow Third Party SSO

Purpose

[Create / Sign into Tozny Dashboard](#)

[Create TozID Realm](#)

[Create Realm Identity](#)

[Create Realm Client Application](#)

[Create SSO Application](#)

[Login to SSO Application using TozID](#)

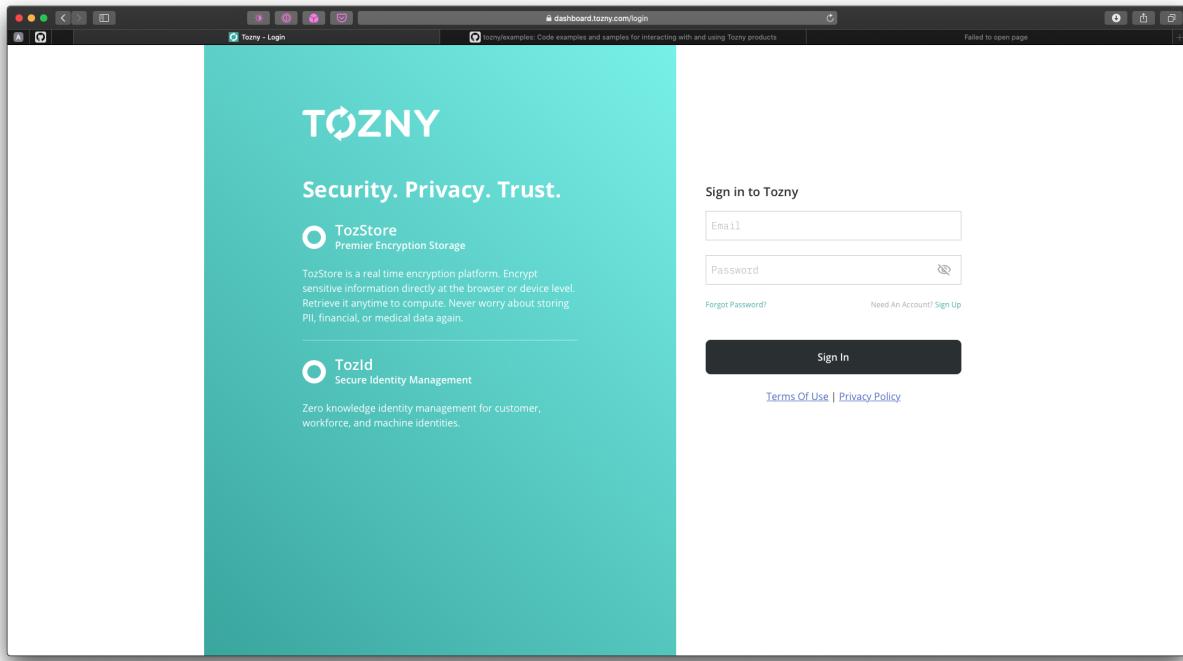
Purpose

The purpose of this document is to walk through how to use TozID to provide Single Sign On to a website using the OIDC implicit flow.

Create / Sign into Tozny Dashboard

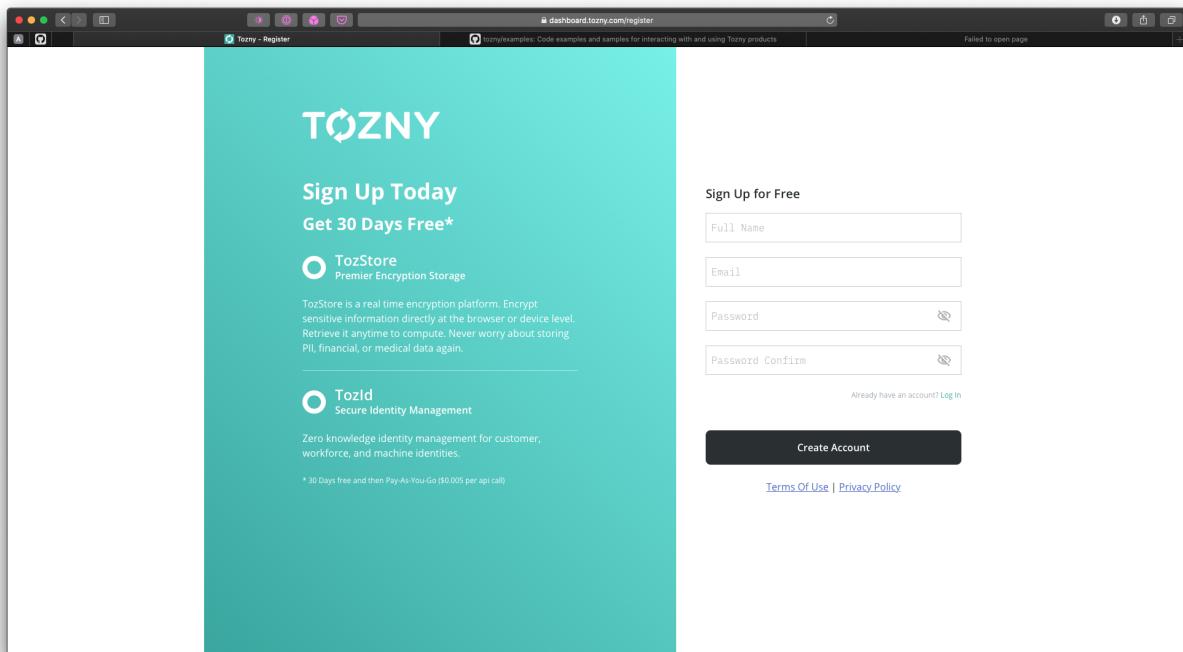
To get started, you will either need to create a Tozny account, or sign into your existing account.

Navigate to <https://dashboard.tozny.com>



If you have an existing account, use your username and password to login.

Otherwise, click 'Sign Up', enter your account name, email (which will serve as your username for login purposes) and choose a strong password to protect your account.



After filling out valid information and clicking ‘Create Account’ you will be presented with a paper key, which can be used to recover access to your account and associated data in the event you have lost your password. Make sure to store this paper key somewhere private and durable such as a password manager.

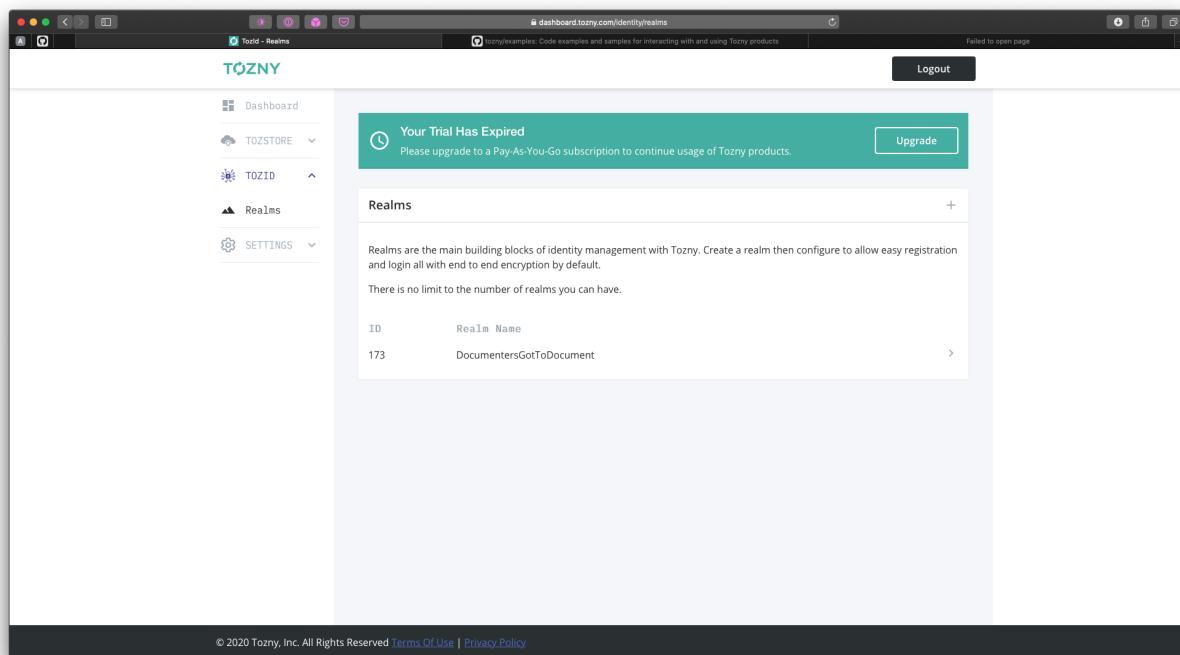
Once you’ve created or logged into your existing account, you will receive an email to verify your account.

Open that email and click the verification link.

Create TozID Realm

Now that your account has been verified, you can proceed to creating a realm. A realm is a virtual container for identities (users or devices) and applications (such as a web site or API service) that allows you to manage settings, authentication and authorization for identities and applications that are a part of the realm.

Select TozID from the side left menu bar, and click on realms.



The screenshot shows the Tozny dashboard interface. On the left, there is a sidebar with the following navigation options: Dashboard, TOZSTORE, TOZID (which is currently selected), and Realms. Below the sidebar, there is a 'SETTINGS' dropdown. The main content area has a teal header bar with the text 'Your Trial Has Expired' and a 'Upgrade' button. Below the header, there is a section titled 'Realms' with a brief description: 'Realms are the main building blocks of identity management with Tozny. Create a realm then configure to allow easy registration and login all with end to end encryption by default.' There is also a note stating 'There is no limit to the number of realms you can have.' A table lists one realm entry:

ID	Realm Name
173	DocumentersGotToDocument

At the bottom of the page, there is a footer bar with the text '© 2020 Tozny, Inc. All Rights Reserved [Terms Of Use](#) | [Privacy Policy](#)'.

Click the '+' icon in the middle left of the screen, and enter a unique realm name (using lowercase characters a-z or digits 0-9 only), then click ‘Create Realm’. After a few seconds, your realm will be created.

Select the realm you created from the list of realms

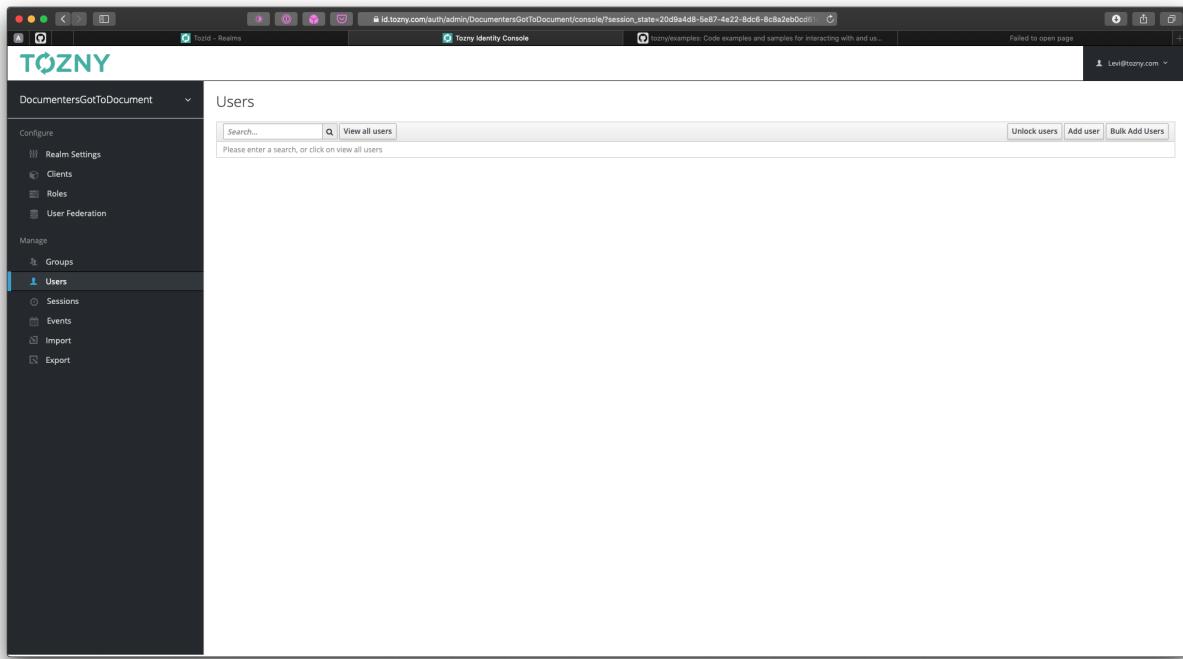
The screenshot shows the Tozny dashboard interface. At the top, there's a navigation bar with links for 'Dashboard', 'TOZSTORE', 'TOZID', 'Realms', and 'SETTINGS'. A prominent green banner at the top center says 'Your Trial Has Expired' with a note to upgrade to a Pay-As-You-Go subscription. Below the banner, the main content area is titled 'DocumentersGotToDocument' and shows details like 'Realm ID: 173' and 'Realm Admin: levi@tozny.com'. It also has a toggle switch for 'Email Recovery Enabled' which is set to 'ON'. A large red button labeled 'Manage Realm' is visible. Below this, there's a section titled 'Danger Zone' with a red button labeled 'Delete Realm'. At the bottom of the page, a copyright notice reads '© 2020 Tozny, Inc. All Rights Reserved [Terms Of Use](#) | [Privacy Policy](#)'.

Click 'Manage Realm' to be taken to the Realm Admin Portal for the given realm.

This screenshot shows the 'DocumentersGotToDocument' realm configuration page. On the left, a sidebar lists 'Configure' (selected) and 'Manage' sections. Under 'Configure', 'Realm Settings' is selected, showing sub-options for 'Clients', 'Roles', and 'User Federation'. Under 'Manage', options for 'Groups', 'Users', 'Sessions', 'Events', 'Import', and 'Export' are listed. The main content area is titled 'DocumentersGotToDocument' and contains tabs for 'General', 'Email', and 'Tokens'. The 'General' tab is active, displaying fields for 'Name' (set to 'DocumentersGotToDocument'), 'Display name' (also 'DocumentersGotToDocument'), and 'HTML Display name' (containing the code '<div class="kc-logo-text">DocumentersGotToDocument</div>'). It also shows 'Enabled' status as 'ON' and 'User-Managed Access' status as 'OFF'. There's a 'Endpoints' section with a dropdown set to 'OpenID Endpoint Configuration' and two buttons at the bottom: 'Save' and 'Cancel'.

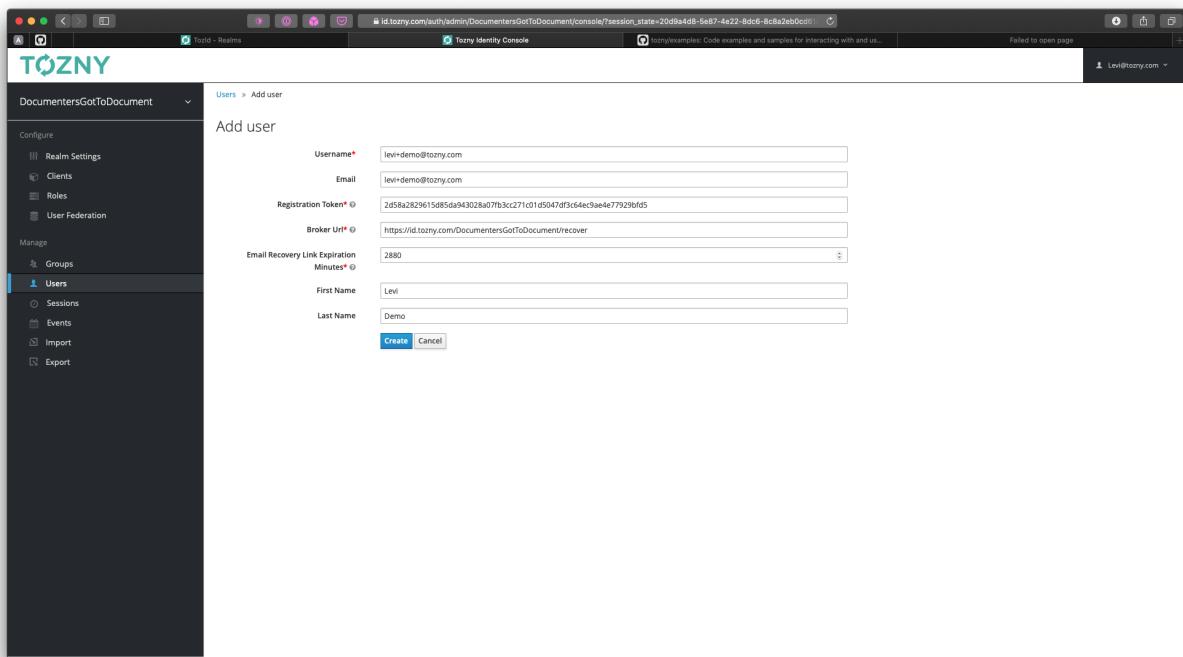
Create Realm Identity

From the lower left menu bar, click on 'Users'



The screenshot shows the 'Users' page in the Tozny Identity Console. The left sidebar has a dark theme with white text. It includes sections for 'Configure' (Realm Settings, Clients, Roles, User Federation) and 'Manage' (Groups, Users, Sessions, Events, Import, Export). The 'Users' section is currently selected and highlighted in blue. The main content area is titled 'Users' and contains a search bar with placeholder text 'Please enter a search, or click on view all users'. Below the search bar are three buttons: 'Unlock users', 'Add user', and 'Bulk Add Users'.

On the upper right side of the screen click on 'Add user', and enter information for the new identity.

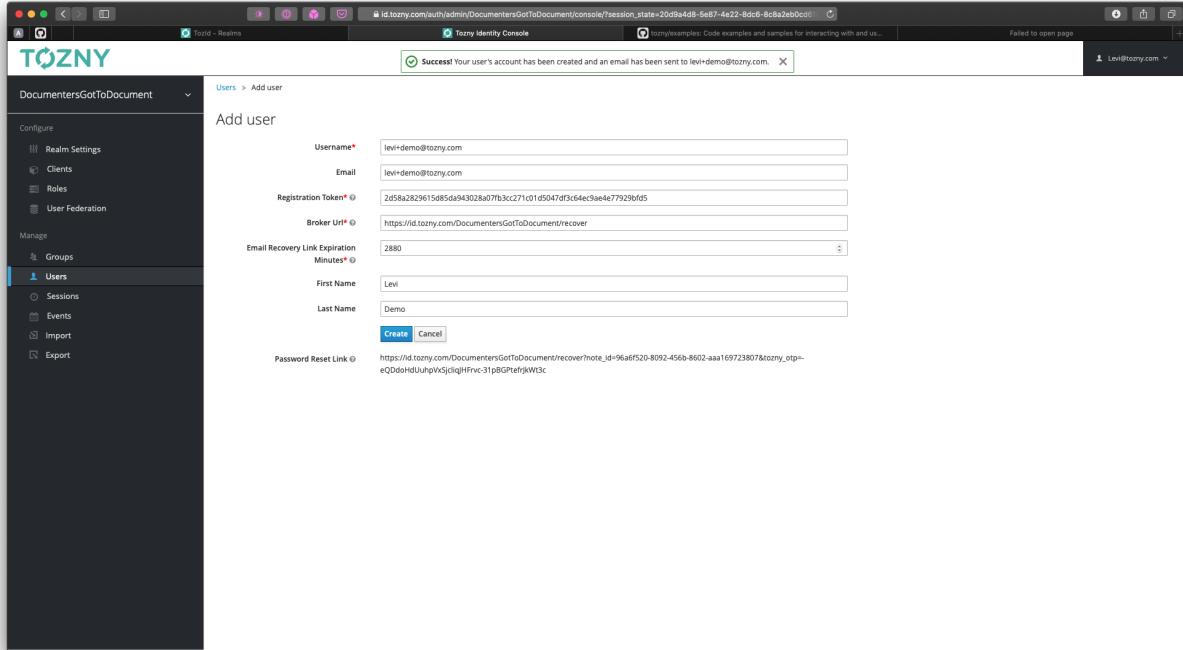


The screenshot shows the 'Add user' form in the Tozny Identity Console. The left sidebar is identical to the previous screenshot. The main form is titled 'Add user' and contains the following fields:

Username*	levi+demo@tozny.com
Email	levi+demo@tozny.com
Registration Token*	2d58a2829c515d85da943028a07fb3c271c01d5047df3c64ec9ae4e77929bf5
Broker Uri*	https://id.tozny.com/DocumentersGoToDocument/recover
Email Recovery Link Expiration Minutes*	2880
First Name	Levi
Last Name	Demo

At the bottom of the form are two buttons: 'Create' and 'Cancel'.

Click create, and after a few seconds the new identity will be created.



An email will be sent to the specified email address containing a link for the new identity to set their password, or you can also use the 'Password Reset Link' value (which will only be shown on initial identity creation, but a password reset can be triggered at a later time as well) to give to the user over another secure channel.

Tozny

A Password Reset H...

To: levi+demo@tozny.com

Inbox - ...tozny.com 2:12 PM Details T

Hello User!

A password reset request has occurred for Tozny Identity, please click the button below to reset your password.

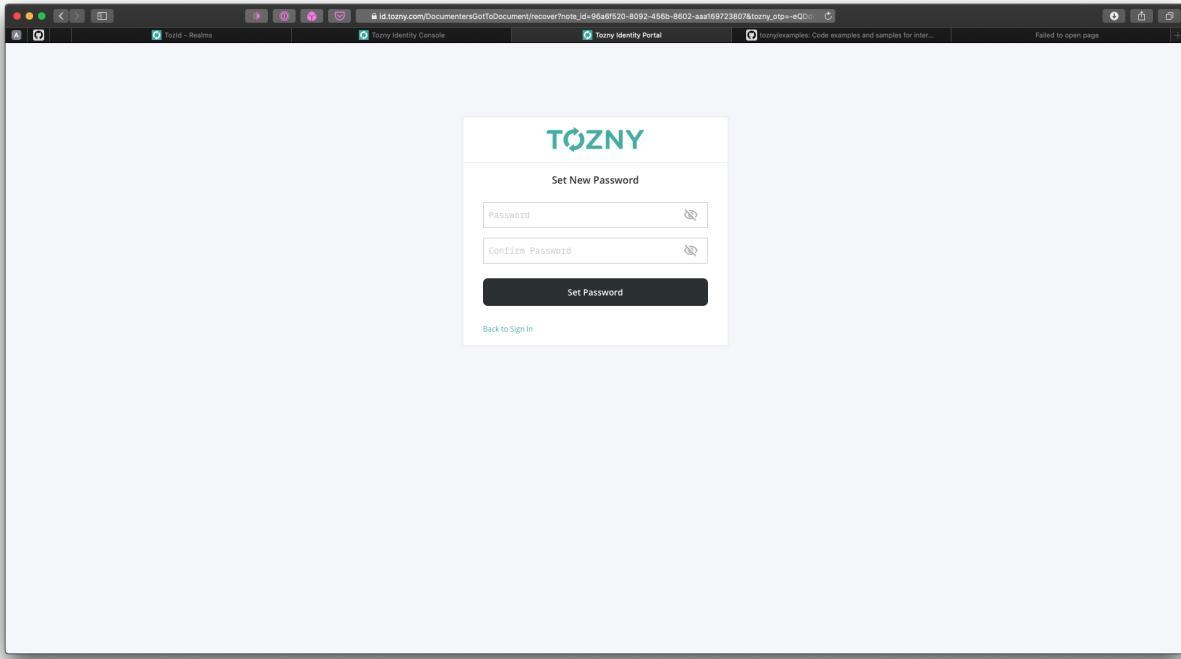
[Reset Password](#)

If you did not request password assistance, please disregard this email.

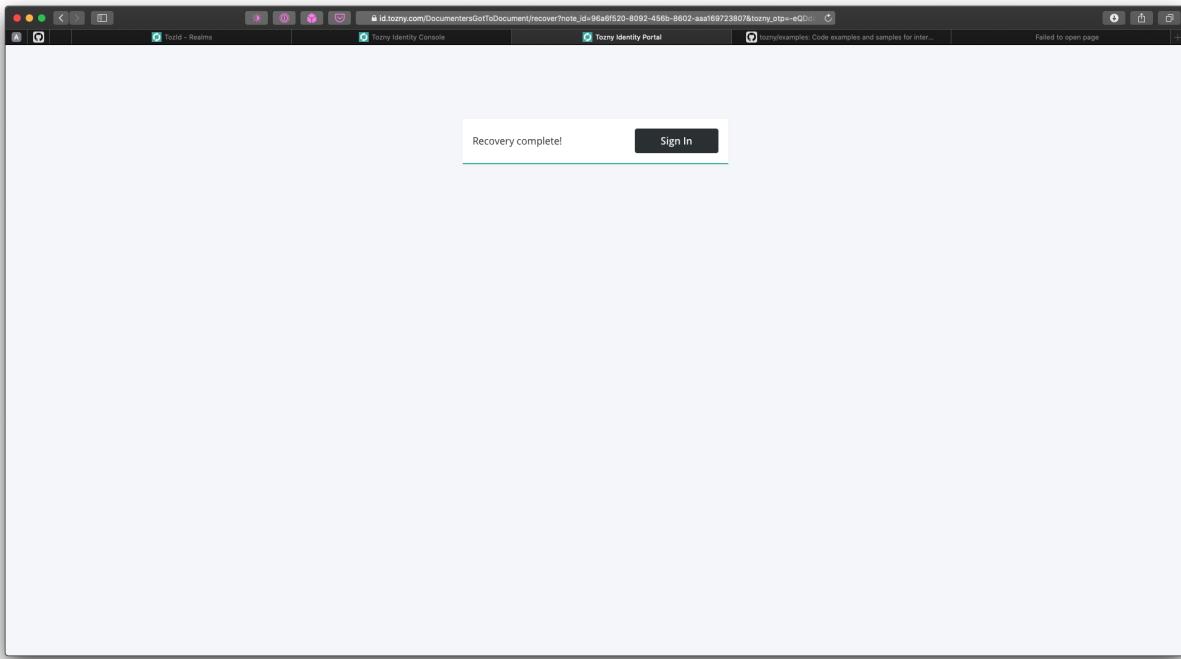
This is an automated email. Please do not reply to this message.

Identity Powered by [Tozny](#)

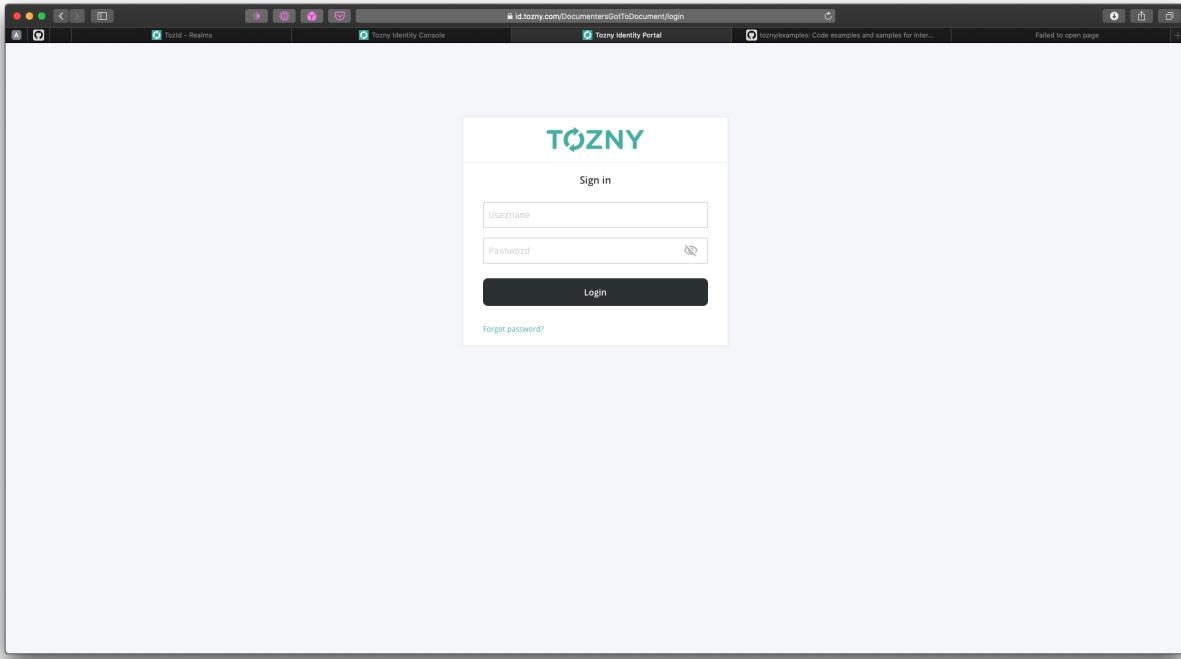
To finish setting up the identity, either click the 'Reset Password' button in the email or navigate to the url specified by the 'Password Reset Link' .



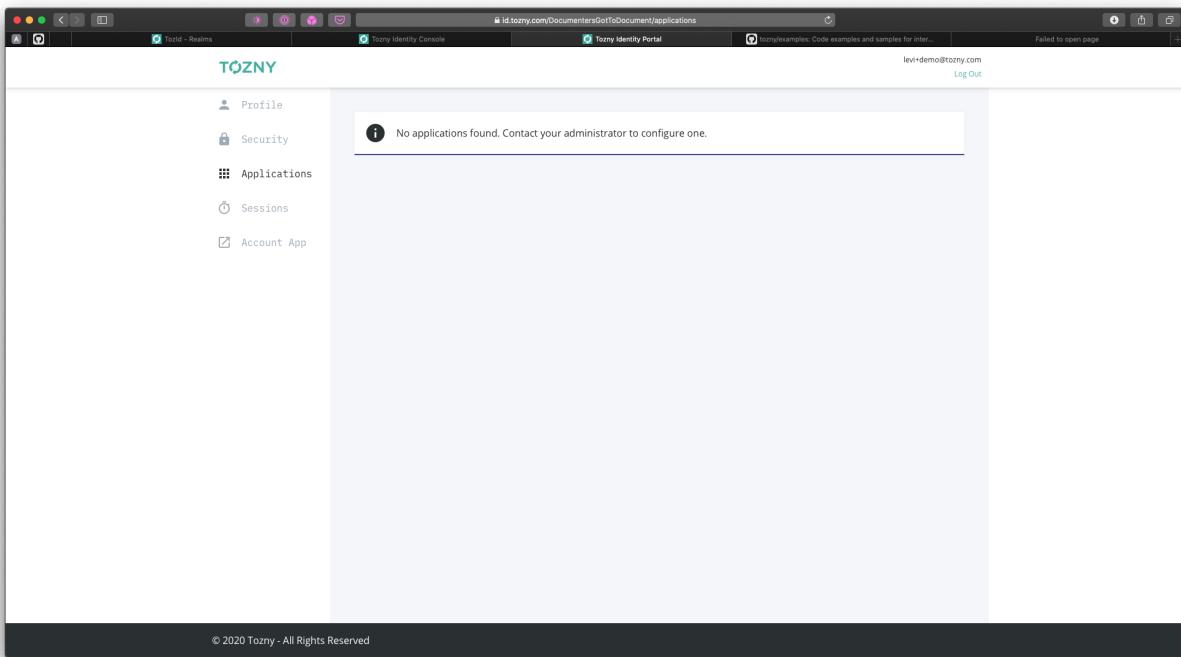
Enter a strong password and click 'Set Password'. Upon success the following screen will be shown:



Click 'Sign In' and enter the username and password associated with the identity

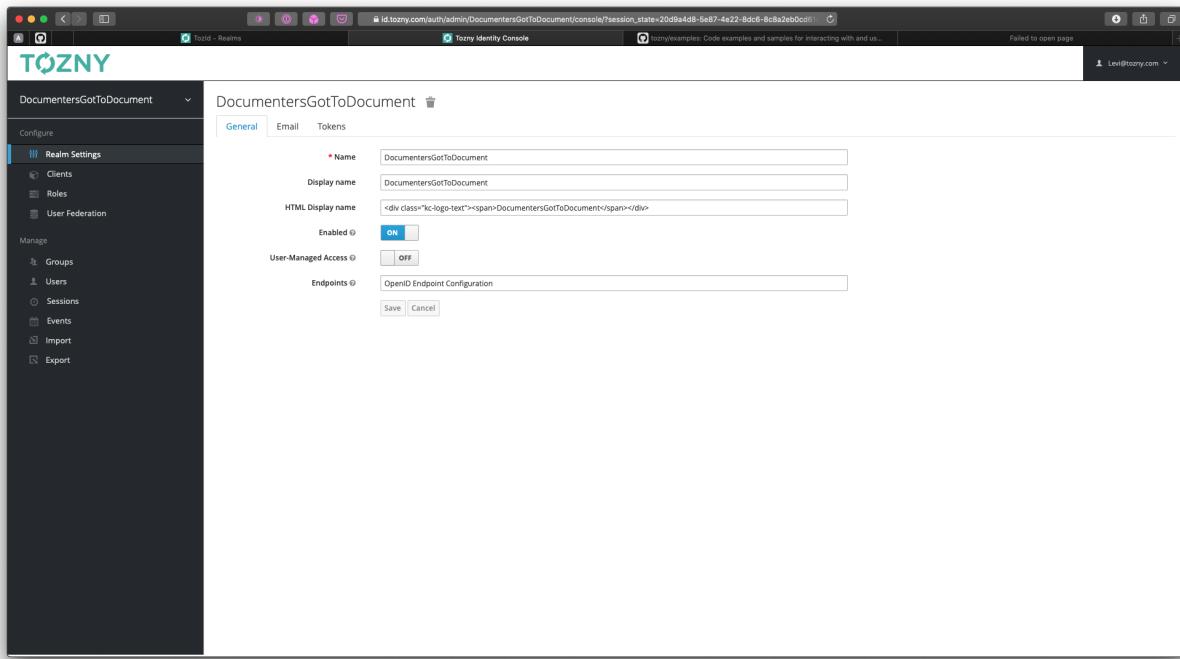


Click 'Login' and you will be taken to the Identity Portal page, at this time the identity will have no applications associated with them.

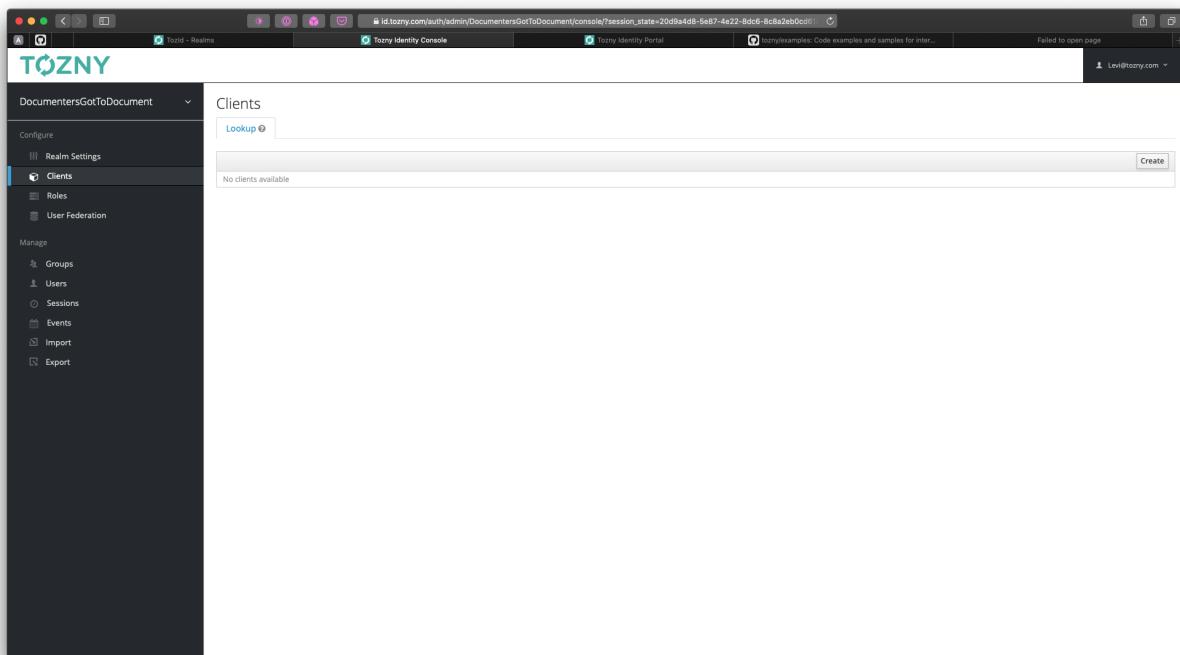


Create Realm Client Application

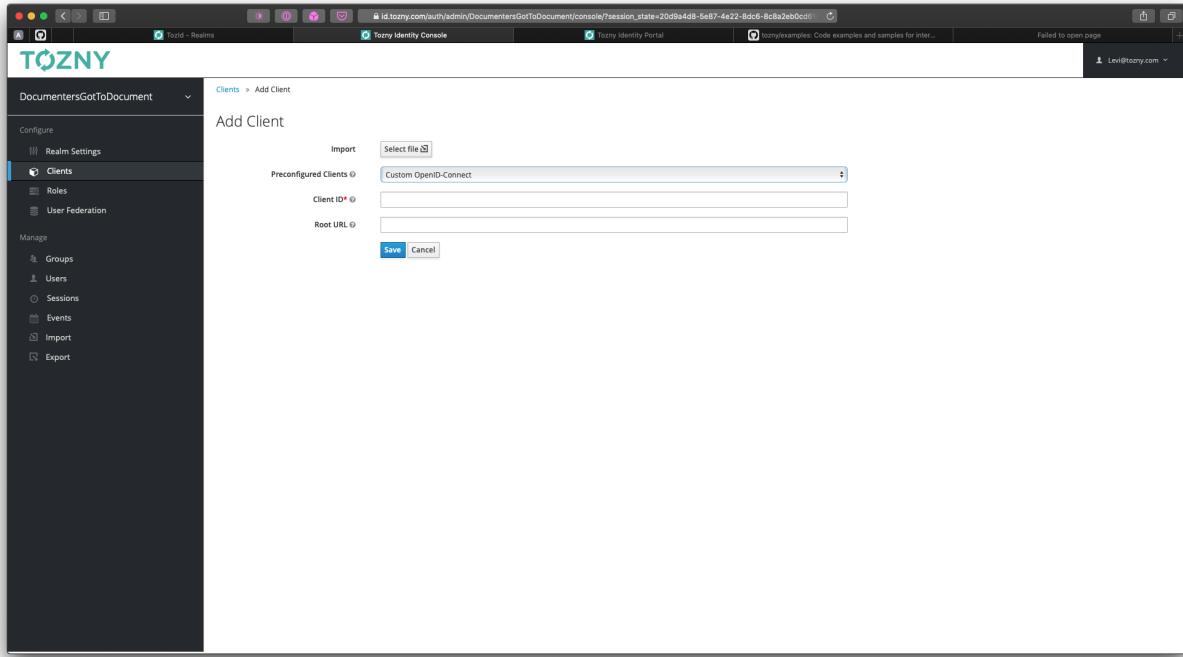
Next you will create an application for the identity to be able to log in to. Navigate back to the Realm Admin Portal for the previously created realm.



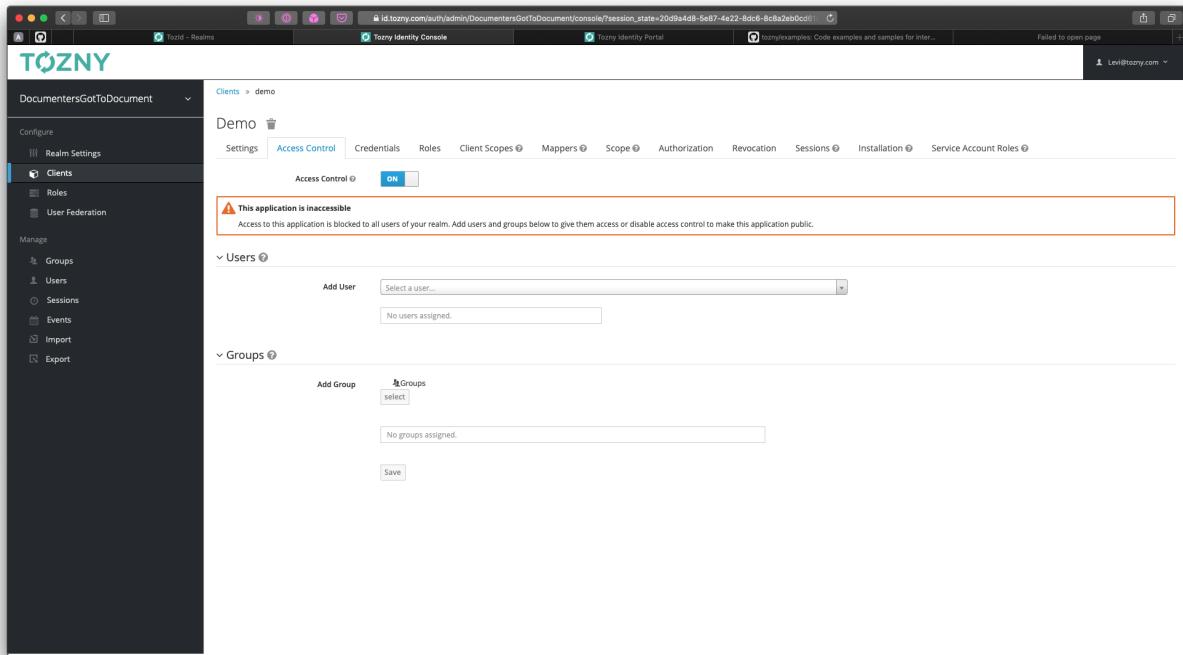
On the upper left menu, click on 'Clients'



Click 'Create' on the right side of the screen



Select 'Custom OpenID-Connect' for 'Preconfigured Clients', choose a meaningful and unique name, leave the 'Root URL' blank, and select 'Save'.



Under the 'Users' section, type the username of the previously created identity.

This application is inaccessible
Access to this application is blocked to all users of your realm. Add users and groups below to give them access or disable access control to make this application public.

Add User

Username	Actions
levi-demo@tozny.com	Remove

Add Group

select

No groups assigned.

Save

Hit save and now that identity will be authorized to login to this application using TozID.

Success! Access control settings saved.

Add User

Username	Actions
levi-demo@tozny.com	Remove

Add Group

select

No groups assigned.

Save

Finally, we must make the roles available in the client scope. This adds the roles to the token claims. Under the ‘Client Scopes’ section, add ‘role’ to the ‘Assigned Default Client Scopes’. Hit save.

The screenshot shows the Keycloak configuration interface for a realm named 'Demo'. On the left, there's a sidebar with 'Clients' selected. The main area has tabs for 'Client Scopes', 'Sessions', 'Installation', and 'Service Account Roles'. Under 'Client Scopes', there are two sections: 'Default Client Scopes' and 'Available Client Scopes'. The 'Available Client Scopes' section contains the 'roles' scope. The 'Assigned Default Client Scopes' section contains 'email', 'profile', and 'web-origins'. There's also a 'Remove selected' button at the bottom of the 'Assigned' list.

Create SSO Application

Next, we will set up and run a web server that will use TozID via the OIDC implicit flow to allow authorized identities to log into it. You can clone and run the example application provided via the Tozny `examples` GitHub repo:

<https://github.com/tozny/examples/tree/trunk/tozid/sso/oidc/implicit-flow>

In a terminal, clone the Tozny examples repo

```
Last login: Wed Sep 30 14:39:32 on ttys002
cd [octo@ValleyOfTheForge examples]$ cd ~/forges/
[octo@ValleyOfTheForge forges]$ git clone git@github.com:tozny/examples.git
```

Change into that repo, and navigate to the `tozid/sso/oidc/implicit-flow` directory.

Open the `main.js` file in the `public/javascript` directory.

Update the values to match that of your created realm and client application, e.g.

```
...
const TOZID_REALM_NAME = 'DocumentersGotToDocument';
const TOZID_CLIENT_ID = 'demo';
const TOZID_HOSTNAME = 'https://id.tozny.com';
...
```

Save the file. Navigate back to the `tozid/sso/oidc/implicit-flow` directory.

Install all dependencies for the project by running

...

npm install

...

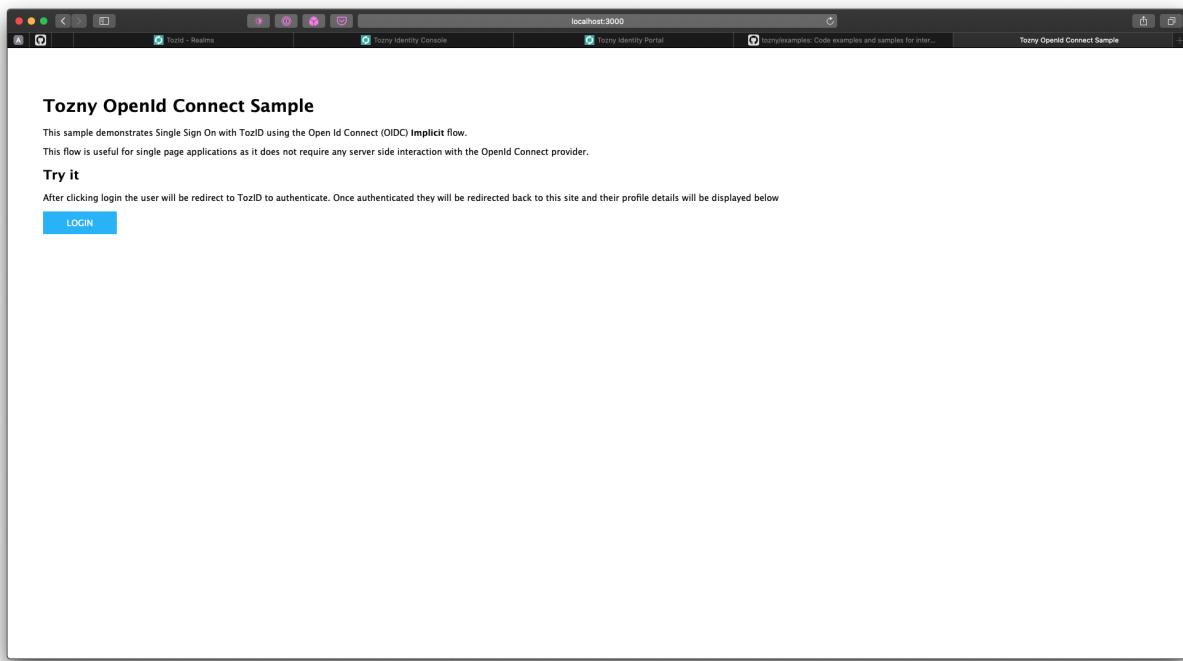
Start the web server by running

...

npm run start

...

Navigate to the web server in your browser at `<http://localhost:3000>`



Navigate back to the realm client application, toggle the 'Implicit Flow Enabled' to on and update the redirect URLs to include '<http://localhost:3000>', otherwise the realm will not allow the user to be redirected back to the application once they have signed in or for the application to use the Implicit Flow to log the user in.

The screenshot shows the Tozny Identity Console interface. On the left, there's a sidebar with 'Configure' and 'Manage' sections. Under 'Configure', 'Clients' is selected, showing a list with 'demo'. Under 'Manage', there are links for 'Groups', 'Users', 'Sessions', 'Events', 'Import', and 'Export'. The main area is titled 'Demo' and contains a 'Settings' tab. The 'Settings' tab has several configuration fields:

- Client ID:** demo
- Name:** (empty)
- Description:** (empty)
- Enabled:** ON
- Consent Required:** OFF
- Login Theme:** (empty)
- Client Protocol:** openid-connect
- Access Type:** public
- Standard Flow Enabled:** ON
- Implicit Flow Enabled:** ON
- Direct Access Grants Enabled:** ON
- Allow API Access:** OFF
- Root URL:** (empty)
- Valid Redirect URIs:** http://localhost:3000
- Base URL:** (empty)
- Admin URL:** (empty)
- Web Origins:** (empty)

At the bottom, there are links for 'Fine Grain OpenID Connect Configuration' and 'OpenID Connect Compatiblity Modes'.

Click 'Save'

Login to SSO Application using TozID

Navigate back to the local web server, and click 'Login'.

The screenshot shows a local web browser window at 'localhost:3000'. The title bar says 'Tozny OpenId Connect Sample'. The page content is as follows:

Tozny OpenId Connect Sample

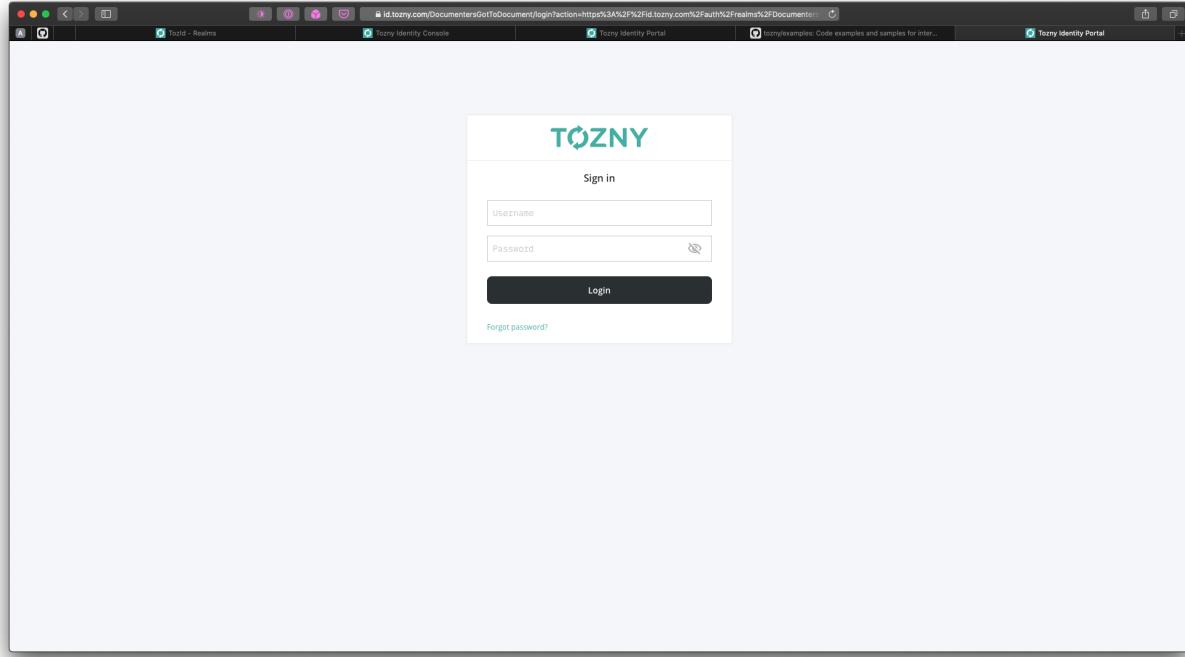
This sample demonstrates Single Sign On with TozID using the Open Id Connect (OIDC) Implicit flow.
This flow is useful for single page applications as it does not require any server side interaction with the OpenId Connect provider.

Try it

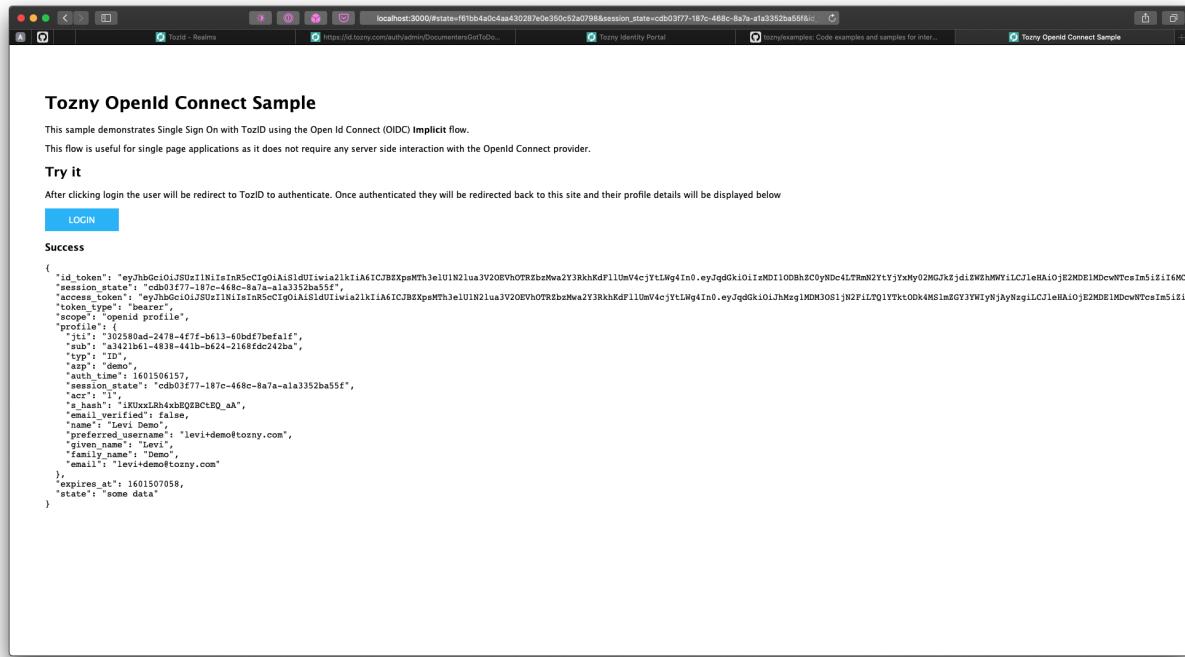
After clicking login the user will be redirect to TozID to authenticate. Once authenticated they will be redirected back to this site and their profile details will be displayed below

LOGIN

Login to TozID using the username and password associated with the previously created and authorized identity



Upon success you will be redirected back to the local web server application.



Congratulations, you've just successfully set up a simple but powerful and flexible single sign on flow for your application.