

## 2. OAuth Provider Setup

Roost supports various authentication mechanisms as mentioned below

1. Github
2. Google
3. Microsoft
4. Linkedin
5. Okta

### OKTA Auth Client Setup

- Sign in to your OKTA account with admin privileges (*If you do not have an existing Okta account, then sign-up at [Home | Okta Developer](#)* )
- From the left navigation menu, go to Applications -> Applications.
- Select Create App Integration → OIDC - OpenID Connect → Web Application, then click Next
- Fill in the suitable **App integration name**, upload the logo.
- Add **Sign-in redirect URIs**
  - [https://<DNS\\_NAME>/login](https://<DNS_NAME>/login)
- Allow Access to users thru Assignments → Controlled Access
  - Select the groups of users or Allow access to everyone
- Save and Make a note of the Okta Client ID and the Client Secret (It is needed later in the config below)
- From the left navigation menu, go to Security -> API
- Make a note of **Issuer URI** for default Authorisation Server
  - something like [https://{your\\_domain}.okta.com/oauth2/default](https://{your_domain}.okta.com/oauth2/default)

### Google Auth Client Setup

- [Integrating Google Sign-In into your web app](#) | [Google Sign-In for Websites](#) | [Google Developers](#)
- Login to <https://console.cloud.google.com/apis/credentials>
- Create Credentials, Select OAuth Client and Application Type as Web Application
- Add Authorised JavaScript Origin as
  - <https://roostapi.roost.io:60001>
  - [https://<DNS\\_NAME>](https://<DNS_NAME>)

- <http://localhost:3000>
  - <http://localhost:4200>
  - Add Authorised redirect URIs
    - [https://<DNS\\_NAME>/login](https://<DNS_NAME>/login)
    - [https://<DNS\\_NAME>/api/auth/redirect/google](https://<DNS_NAME>/api/auth/redirect/google)
    - <https://roostapi.roost.io:60001/auth/redirect/google>
  - Download the JSON
  - Make a note of the Google Client ID and the Client Secret (It is needed later in the config below)
- 

Revision #2

Created 15 March 2023 19:21:15 by Rakesh

Updated 15 March 2023 19:21:24 by Rakesh