# Feide pålogging i Sentralimport

Feide pålogging i sentralimport-appen fungerer ved at appen redirecter brukeren til dataporten.no hvor brukeren logger seg på.

Appen må være registrert på [https://dashboard.dataporten.no](https://dashboard.dataporten.no/) med en redirect uri. Her kan man kan også sette opp hvilke providers som brukere kan logge på med (f.eks IDporten), og hvilke tilganger (scopes) appen ber om å få se.

Ved pålogging sender appen brukeren til <https://auth.dataporten.no/oauth/authorization>?client\_id={client\_id}&redirect\_uri={redirect\_uri}&scope=openid userid-feide gk\_testing&response\_type=id\_token token&state={state}&nonce={nonce}

Parametrene:

client\_id: finner man i dataporten der applikasjonen ble registrert. Generert av dataporten

redirect\_uri: redirect\_uri’en man registrerte med appen hos dataporten. Man kan registrere flere redirect\_uri’er. Den som blir sendt med her er den man blir redirected tilbake til ved vellykket innlogging

scope: hvilke tilganger appen ber om å få se. Minimum av rettigheter appen trenger for innlogging er openid, userid-feide og api-gateway-scopet, men om det skulle være behov kan man sette opp flere og be om f.eks email

response\_type: appen bruker for øyeblikket OpenId med implicit flow og ber derfor dataporten direkte om en access-key. Dette får man ved å sette id\_token token for response\_type. Om man senere vil implementere en authorization flow må man be om code her.

State: en string appen setter som sier noe om state. Denne bør sjekkes at er den samme i responsen

Nonce: en randomly generated string som sendes med som ekstra sikkerhet. Må sjekkes at den er den samme i responsen.

Når brukeren har logget inn via dataporten blir man redirected tilbake til den adressen som er gitt i redirect\_uri slik:

http://{redirect\_uri} #access\_token={accesss\_token} &token\_type=Bearer&expires\_in=28799&scope=openid+userid-feide+{api\_gateway\_scope} &state={state}&id\_token={jwt\_token}

jwt\_token er en json web token som inneholder bl.a aud (client\_id), nonce og tidspunkt for expiration og innlogging.

Her bør det sjekkes at state og nonce er den samme som ble sendt. Appen setter da access\_token i localstorage og brukeren er logget inn i appen.

Når appen skal ta kontakt med api’er bruker vi dataportens API-Gateway. Api’et som skal kontaktes registreres i dataporten med ett api-endepunkt og setter appen som skal kontakte api’et som authorized client.

Appen sender så ett request til https://{gateway}.dataporten-api/{path} slik:

GET / {path} HTTP/1.1

Host: {gateway}.dataporten-api.no

Authorization: Bearer {access\_token}

Parametre:

Gateway: navnet på gatewayen. Settes når man registrerer api hos dataporten

Path: pathen til endepunktet man forsøker å nå

Access\_token: token man fikk da man logget inn

Access token blir da autentisert av dataporten og om det er vellykket blir requestet sendt videre til https://{api\_endepunkt}/{path} slik:

GET /foo/bar HTTP/1.1

Host: {api\_endepunkt}

Authorization: Basic XXXXXXXX

X-Dataporten-clientid: {client\_id fra klienten som ønsker adgang}

X-dataporten-gatekeeper: {api\_gatekeeper\_navnet}

X-dataporten-token: {token fra dataporten}

X-Dataporten-userid-sec: feide:xxxx@uninett.no

Api\_endepunkt er endepunktet som ble satt ved registrering av API-Gateway I dataporten.

Hvilke ting man får med her er litt avhengig av hvilke scopes som ble requested.

Autentisering foregår da med basic auth med brukernavn og passord fra dataporten. Videre må api’et sjekke X-Dataporten-userid-sec (feide-id’en) for å se hvilken bruker som faktisk er logget på for å sjekke tilgangsstyring innad i api’et.

Sentralimporten er foreløpig tilknyttet to Api’er:

Pia-rest-api’et og crisrest-api’et.

Logout

Logout utføres ved å sette variabelen «authorized» i localstorage til false og deretter kalle <https://auth.dataporten.no/logout> for å gjøre token inaktiv.

Dette vil for øyeblikket føre til at brukeren ender opp på dataporten sine utlogging-side, men de arbeider med å innføre en redirect parameter slik at man kan bli redirected tilbake til vår side.