CULR webservice version 4

Indhold

Beskrivelse af CULR-systemet 1
Formål 1
Systemarkitektur 1
Definitioner 2
Provider 2
Account 2
Patron 2
Authentication 2
Operationerne 3
Uddybende forklaring af createAccounts 4
Uddybende forklaring af mergeAccounts 5
Eksempel på mergeAccounts 6

Beskrivelse af CULR-systemet

Formål

CULR-systemet modtager fra tilsluttede dataleverandører (fx biblioteker) oplysninger om de lokale id-numre som hver enkelt registreret biblioteksbruger råder over. Adgang til digitale indholdsprodukter for biblioteksbrugere er afhængig af den enkeltes bibliotekstilhørsforhold. Disse tilhørsforhold kan en tjenesteudbyder (fx en lokal eller national bibliotekstjeneste), som gerne vil give deres brugere adgang til digitale indholdsprodukter, få oplyst fra CULR. Dermed kan tjenesteudbyderen efterfølgende tilbyde deres brugere adgang til de relevante indholdsprodukter som deres biblioteks- eller matrikel-tilhørsforhold berettiger til.

Systemarkitektur

CULR-systemet består af 3 komponenter:

1. En webservice der varetager forbindelsen mellem de tilsluttede institutioner og CULR-systemet.
2. En database som opbevarer de afleverede data fra de tilsluttede institutioner der er dataleverandører
3. Et administrationssystem som varetager administration af brugerkonti (herunder oprettelse af passwords til institutioner) og som styrer adgangen til funktioner i webservicen for de tilsluttede institutioner.

CULR-adgang for dataleverandører og tjenesteudbydere administreres gennem DBCs VIP-base i form af en CULR-profil. Profilen styrer serviceaftagerens mulighed for at anvende de forskellige operationer i servicen samt tildeling af profilnavn der anvendes ved indlogning til webservicen.

Definitioner

Provider

En provider er en leverandør af data til CULR-servicen, dvs. en institution (fx et bibliotek) der har registrerede accounts tilknyttet. Det kan være et folkebibliotek, et uddannelsesbibliotek eller et forskningsbibliotek. Providere registreres i CULR-administrationssystemet (DBCs VIP-base) ved hjælp af deres ISIL nummer (dog uden DK-)

Account

Account indeholder informationer omkring en brugers registreringer hos en given provider. Accountoplysningerne består af:

- userIdType (fx CPR, CVR)
- userIdValue

Sammen med provider er userId+userIdType unik, således at en låner entydigt kan identificeres indenfor en given provideres brugerregistreringer.

Patron

Patron dækker over alle en persons, en virksomheds eller anden entitets, samlede biblioteks-brugerregistreringer. Patron bruges til at kæde accounts sammen på tværs af providers.
En patron har altid minimum en account tilknyttet og en account vil altid have en patron.

Når CULR-servicen opretter nye accounts i sin database, vil den knytte de nye accounts sammen med eksisterende ved hjælp af globalt unikke userIdTypes som eks. CPR eller CVR.

Authentication

For at bruge servicen kræves at serviceaftageren har en DanBib-licens hos DBC (netpunkt konto). Disse kontooplysninger skal - sammen med navnet på CULR-profilen - angives i enhver forespørgsel til servicen.

Kontooplysningerne angives i clientAuth-delen i en request; clientAuth består af 4 dele.

- userIdAut
  - netpunkt
- groupIdAut
  - biblioteksnummer (det same som ISIL, dog uden DK-)
- passwordAut
  - password til netpunkt
- profileName
  - profilnavnet til CURL servicen. Udleveres ved oprettelse af abonnement til servicen

Hvis CULR servicen kaldes uden korrekte brugeroplysninger, returneres en fejl.

Operationerne

Der findes to kategorier af operationer I CULR webservicen.

Den første kategori er for services og består kun af operationen: getAccounts - som henter accounts på tværs af providers vha. deres Patron.

Den anden er tiltænkt providers og er primært operationer som kan benyttes til at manipulere data i systemet, såsom oprettelse og sletning af konti.

Operationer som services kan kalde:

- GetAccounts:
    - Finder accounten der er søgt på
    - Ud fra account findes den tilknyttede Patron
    - Via Patron findes alle den tilknyttede accounts og returneres.

Operationer som providere kan kalde:

- CreateAccount:
    - Hvis accounten ikke findes, oprettes en account . Hvis der i forvejen findes en patron med useridValue og userIdType, oprettes accounten under den patron
    - Hvis der ikke findes en patron, oprettes der en Patron for accounten.
- DeleteAccount:
    - Sletter accounten og tjekker om den tilknyttede patron har flere accounts under sig.
    - Hvis der ikke er flere accounts under den tilknyttede Patron, slettes account'en
- UpdateAccount:
    - Finder accounten som der søges på
    - Via account'en findes Patron
    - Opdaterer eller opretter et municipalityNumber på Patron
- MergePatrons:
    - Tager to accounts som input,en der skal merges til og en der skal merges fra
    - Accounts'ne skal have forskellige userIdType
    - Tager account'en der skal merges til og finder dens Patron
    - Tager Accounten der skal merges fra og finder dens Patron
    - Updater Accountens patronId der skal merges fra til at være lig med Patronens id der skal merges til.
- GetAccountFromProvider
    - Finder accounten der søges på

Uddybende forklaring af createAccounts

Oprettelse af brugere i CULR ved hjælp af createAccount operationen:
Providere kan benytte operationen createAccount til at oprette en ny bruger i CULR.

Tvungne parametre til createAccount:

- userIdType
- userIdValue
- userIdAut
- groupIdAut
- passwordAut
- profileName

Valgfri parametre til createAccount:

- municipalityNo
- uidType
- uidValue

Når man opretter en ny account i CULR vil servicen undersøge hvorvidt der allerede eksisterer accounts som den nye account kan relateres til.

Dvs. at hvis der i forvejen findes en account med følgende credentials:

Folkebib låner 1:
userIdType : cpr
userIdValue : 1231231230
provider : 775100

- og man opretter den nye account med følgende:

Folkebib låner 2:
userIdType : cpr
userIdValue : 1231231230
provider : 761500

- så vil de to accounts blive relaterede under samme patron og således begge optræde i resultatet af en getAccounts.

Det er kun accounts med userIdType (eller med den valgfri parameter uidType) 'cpr' eller 'cvr' som automatisk bliver forsøgt relaterede (uId = "universal Id").

Hvilket vil betyde at denne account:

Unibib låner 1:
userIdType : cphBib1
userIdValue : localVal1
provider : 870001

- ikke bliver automatisk relateret til denne:

Unibib låner 2:
userIdType : cphBib1
userIdValue : localVal1
provider : 870002
uidType : cpr
uidValue : 1231231320

- da der ikke er nogen garanti for at userId-numrene er globale inden for "localValue1"-userIdType'en.

Unibib låner 2 vil blive relateret med folkebib låner 1 og 2 da den har uidType + en uidValue som matcher allerede eksisterende account. Den vil således også optræde i et søgeresultat fra getAccounts.

Uddybende forklaring af mergeAccounts

Når to accounts skal merges i CULR, gøres det ved at man ændrer patronId i den account der skal merges, således at man ved at kalde get Accounts får den nye account med i søgeresultatet.

Tvungne parametre til mergeAccounts:

accountToMergeWith:

- userIdType
- userIdValue
- provider

accountToBeMerged

- userIdType
- userIdValue
- provider

For mergeAccounts gælder ydermere:

- Man kan ikke merge til en account som ikke indeholder CPR/CVR
- Man kan ikke merge accounts som er af samme userIdType

Servicen vil godt senere kunne udbygges med flere globale userIdTypes (udover CPR/CVR), efter aftale med de relevante dataleverandører/providers og Kulturstyrelsen.

Eksempel på mergeAccounts

accountToMergeWith
Folkebib låner 1:
userIdType : cpr
userIdValue : 1231231230
provider : 000001

accountToBeMerged:
Unibib låner 1:
userIdType : cphBib1
userIdValue : localVal1
provider : 870001

Effekten af mergeAccounts betyder at en søgning med getAccounts på userIdType = 'cpr' og userIdValue = '1231231230' samt provider = 000001 nu vil returnere begge de ovenstående accounts samt alle andre relaterede accounts der måtte eksistere.

Tilsvarende vil en søgning på userIdType = cphBib1 + userIdValue = localVal1 + provider = 870001, returnere begge accounts og alle de relaterede accounts der måtte eksistere.