2.2 Attributstyrning OIDC

Revisionshistorik

Version	Datum	Författare	Kommentar
0.1	27 Jun 2018	Unknown User (peterssond)	Upprättad
0.2	07 Sep 2018	Grim Skarsgård	Utkast
0.3	23 Oct 2018	Grim Skarsgård	Information om filtrering av authorizationScope
1.0	14 Jan 2019	Skarsgård, Grim	Fastställd
2.2	23 Nov 2021	Ehlert, Stefan	Uppdaterad med information kring allCommissions och allEmployeeHsaIds
2.2.1	10 Dec 2021	Ehlert, Stefan	Information kring authenticationMethod

1. Sammanfattning

Vid klientregistrering anges vilka attribut (claims) som skall finnas tillgängliga för klienten vid en autentiseringsbegäran.

Vid autentiseringsbegäran anges vilka attribut (claims) som efterfrågas, och huruvida de är tvingande (essential) eller inte.

2. Tillgängliga claims och scopes

- Claim = attribut
- Scope = en samling av claims

Attributlistan visar vilka claims och scopes som kan levereras av IdP. Notera att varje claim ingår i ett scope.

3. Klientregistrering

Vid klientregistrering anges vilka claims som är godkända för IdP att släppa ifrån sig till klienten.

Scopet "openid" och de claims som ingår däri är obligatoriska och behöver inte specificeras. Övriga tillåtna attribut kan anges ett och ett som claims eller gruppvis som scopes. Vid registreringen sparas allting som enskilda claims oavsett.

4. Autentiseringsbegäran

Autentiseringsbegäran måste specificera vilka attribut som skall returneras efter en lyckad autentisering.

Attributbegäran görs via två parametrar i autentiseringsbegäran: **scope** och/eller **claims**.

- Begärda claims eller scopes som inte finns definierade i attributlistan ignoreras av IdP.
- Det gör ingen skillnad huruvida ett attribut ingår i ett begärt scope eller om det begärs individuellt i claims-parametern, eller i både och. Dock går det i claims-parametern att specificera ytterligare villkor för det begärda attributet (se claims-avsnittet nedan).
- De begärda attributen filtreras och IdP returnerar endast de begärda attribut som är godkända i klientregistreringen enligt ovan.

4.1. scope

https://openid.net/specs/openid-connect-core-1_0.html#ScopeClaims

Det obligatoriska "openid"-scopet i autentiseringsbegäran kan kompletteras med ytterligare scopes.

Ex: Begära alla attribut via scope

```
scope = openid commission authorization_scope personal_identity_number
```

De claims som ingår i begärda scopes levereras i id-token i autentiseringssvaret.

4.2. claims

https://openid.net/specs/openid-connect-core-1_0.html#ClaimsParameter

- Begärs ett och ett.
- Anges separat huruvida attributet skall returneras i svarets **id-token** och/eller levereras från **UserInfo**-endpointen.
- Kan markeras som **essential**, d.v.s. tvingande.
    - Motsvarar fältet "isRequired" i SAML
    - Om attributet är flaggat som essential på minst en av *userinfo* och *id_token,* och IdP inte kan leverera attributet som begärt så kommer inloggningen att misslyckas.

Ex: Begära attribut via claims

```
claims = {
        "userinfo" : {
                "given_name" : null,
                "mobileTelephoneNumber" : {
                        "essential" : true
                },
                "healthCareUnitName" : {
                        "essential" : true
                },
                "commissionRight" : null
        },
        "id_token" : {
                "healthCareUnitHsaId" : {
                        "essential" : true
                },
                "healthCareUnitName" : null
        }
}
```

4.3. Filtrering av authorization_scope

- Styrning av vilka authorization_scope som returneras görs genom att skicka med en **value**-parameter (om det är ett ensamt godkänt värde) eller **values**-parameter (med en array av godkända värden) under authorizationScope i claims-parametern. Värdet jämförs med underattributet **authorizationScopeCode** och icke-matchande resultat filtreras bort. Detta kan användas tillsammans med *essential*-flaggan för att direkt kräva att en användare har en roll inom rätt behörighetsområde för att få utföra en inloggning.

Exempel på ett returnerat authorizationScope

```
"authorizationScope" : [{
            "authorizationScopePropertyName" : "Tjänstesupport",
            "authorizationScopeName" : "Säkerhetstjänster",
            "authorizationScopeCode" : "BIF",
            "authorizationScopePropertyCode" : "BIF;002",
            "authorizationScopeDescription" : "HSA Domain description",
            "authorizationScopePropertyDescription" : "Tjänstesupport Beskrivning",
            "adminCommissions" : [{
                    "adminCommissionHsaId" : "SE1804231406",
                    "sector" : [{
                            "sectorFlag" : true,
                            "name" : "SE111-JLL",
                            "unitHsaId" : "SE111-JLL"
                        },{
                            "sectorFlag" : false,
                            "name" : "SE111-IVA-NAME",
                            "unitHsaId" : "SE111-IVA"
                        }
                    ],
                    "adminCommissionResponsibleOrganisation" : "232100-0214"
                }
            ]
```

Ex: Begäran om specifik loa nivå

```
claims = {
        "id_token" : {
                "acr" : {
                        "value" : "http://id.sambi.se/loa/loa3",
                        "essential" : true
                }
        }
}
```

Ex: claims-begäran som kräver att användaren har minst ett authorizationScope med authorizationScopeCode lika med "BIF"

```
claims = {
        "userinfo" : {
                "authorizationScope" : {
                        "value" : "BIF",
                        "essential" : true
                }
        },
        "id_token" : {
                "authorizationScope" : {
                        "value" : "BIF",
                        "essential" : true
                }
        }
}
```

Ex: claims-begäran som filtrerar bort authorizationScope som inte har authorizationScopeCode "SYS1" eller "SYS2"

```
claims = {
        "userinfo" : {
                "authorizationScope" : {
                        "values" : ["SYS1", "SYS2"]
                }
        },
        "id_token" : {
                "authorizationScope" : {
                        "values" : ["SYS1", "SYS2"]
                }
        }
}
```

5. Autentiseringsmetod

I fallet att ett anslutande system har flera autentiseringsmetoder påslagen går det att förvälja autentiseringsmetoden för en enskild inloggning. Mer specifikt så styrs detta med hjälp av attributet **authenticationMethod** där det går att ange tre olika värden:

- **SITHS_EID_SAME_DEVICE** (SITHS eID på denna enhet)
- **SITHS_EID_OTHER_DEVICE** (SITHS eID på annan enhet)
- **MTLS** (Net iD Enterprise)

Dock så går det inte att välja en autentiseringsmetod med hjälp av **authenticationMethod** om inte den autentiseringsmetoden är påslagen för det anslutande systemet. Om ett sådant försök görs misslyckas inloggningen.

Ex: claims-begäran som väljer SITHS eID på samma enhet som autentiseringsmetod

```
claims = {
        "id_token" : {
                "authenticationMethod" : {
                        "value":"SITHS_EID_SAME_DEVICE"
                }
        }
}
```

6. Uppdragsval

Användaren kommer endast ställas inför ett uppdragsval ifall det är nödvändigt, det vill säga om båda dessa villkor är uppfyllda:

1. Uppdragsspecifika attribut är begärda.
2. Användaren har fler än ett uppdrag.
    - Har användaren endast ett uppdrag så väljs det implicit.

Om användaren inte har några uppdrag så presenteras inte heller något uppdragsval, och om några av de uppdragsspecifika attributen då är markerade som tvingande (essential i claims-parametern) så kommer inloggningen att misslyckas.

6.1. Samtliga uppdragsval som claim

Om man vill undvika uppdragsval i de fall som en användare har flera uppdrag och man vill ha all uppdragsinformation så ska man begära attributet allCommissions. Detta kan vara användbart i situationer då SP:n vill ha användarens fullständiga behörighet.

Observera dock att SP:n fortfarande kan begära andra attribut som triggar ett uppdragsval som måste göras i IdP:n. Ett sådant exempel är ifall SP:n begär allCommissions samt commissionPurpose. I det fallet måste uppdragsvalet göras men SP:n kommer ändå få listan på samtliga uppdragsval tillbaka efter lyckad autentisering.

Ex: claims-begäran som begär samtliga uppdragsval

```
claims = {
        "id_token" : {
                "allCommissions" : {
                        "essential" : true
                }
        }
}
```

6.2. Samtliga HSA ID:n som claim

Om man vill undvika uppdragsval i de fall som en användare har flera HSA ID:n och/eller flera uppdrag och man bara vill ha ett HSA ID:n kan man begära attributet allEmployeeHsaIds. Då returneras en lista av HSA ID:n och SP:n kan då välja ett av dessa, t ex via en användardialog.

Ex: claims-begäran som begär samtliga HSA ID:n

```
claims = {
        "id_token" : {
                "allEmployeeHsaIds" : {
                        "essential" : true
                }
        }
}
```

7. Filtrering av organisationsnummer, HSA-id, personnummer och orgAffiliation

Likt PrincipalSelection för SAML finns motsvarande filtreringsmöjligheter för OIDC. Här ges möjligheten att på förhand göra ett specifikt val över vilket HSA-id, organisationsnummer, personnummer eller orgAffiliation som användaren ska bli inloggad med. Dess värden användas enskilt men också kombineras med varandra. När filtreringen görs försöker IdP:n göra ett val automatiskt baserad på den informationen som tagits emot via de claims som skickats in. O m det är möjligt kan alltså exempelvis ett specifikt tjänste-id pekas ut och användaren slipper göra ett aktivt val i IdP:n.

I exemplet nedan förväntas användaren bli inloggad med HSA-id:t **TSTNMT2321000156-10NG** för organisationen med organisationsnumret **232100-0214**.

Ex: claims-begäran som filtrerar ut ett HSA-id och en organisation

```
claims = {
        "id_token" : {
                "employeeHsaId" : {
                        "value": "TSTNMT2321000156-10NG",
                    "essential": true
                },
                "organizationIdentifier": {
                        "value": "232100-0214",
                        "essential": true
                }
        }
}
```