ADFS-integration

MPASSid läggs till i ADFS med PowerShell-kommandon. Nedan följer ett exempel på PowerShell-kommandon som används för att lägga till MPASSid i ADFS. Det finns två versioner av kommandon beroende på om MPASSid:s metadata läses via internet eller från en lokal kopia.

Tillfogande av MPASSid i ADFS

MPASSid:s metadata med en url-adress

```
$name = "mpass-proxy"
$metadataUrl = "https://mpass-proxy.csc.fi/Shibboleth.sso/Metadata"

Add-ADFSRelyingPartyTrust -MetadataUrl $metadataUrl -Name $name -AutoUpdateEnabled $true -EncryptClaims $true -
SignedSamlRequestsRequired $true -EncryptionCertificateRevocationCheck none  -SigningCertificateRevocationCheck none
```

MPASSid:s metadata lokalt

Kopiera först MPASSid:s metadata till ett önskat index på ADFS-servern med namnet mpass-proxy-metadata.xml.

Om du använder något annat filnamn ska du uppdatera metadatafilens placering och namn i kommandona nedan.

```
$name = "mpass-proxy"
$metadataFile = "c:\<hakemisto>\mpass-proxy-metadata.xml"

Add-ADFSRelyingPartyTrust -MetadataFile $metadataFile -Name $name -EncryptClaims $true -SignedSamlRequestsRequired $t
rue -EncryptionCertificateRevocationCheck none  -SigningCertificateRevocationCheck none


# MPASSid:s data kan uppdateras från metadata med kommandot
Update-AdfsRelyingPartyTrust -TargetName $name -MetadataFile $metadataFile
```

Konfiguration av attribut som skickas till MPASSid

På ADFS definieras med hjälp av claim rules vilka attribut om användaren som ska förmedlas i samband med inloggning i tjänsten. De nödvändiga konfigurationerna kan göras antingen med PowerShell-kommandon, genom att kopiera och vid behov redigera claim rules-exempel från denna anvisning eller genom att definiera dem i ADFS administrationskonsol. Du hittar mer information om claim rules i denna anvisning under rubriken Claim Rules.

Uppdatera nedanstående kommandon med motsvarande AD-attribut för de attribut som ska skickas till MPASSid. Under Types anger du i vilket format attributen ska skickas till MPASSid och under Query anger du i vilket AD-attribut det motsvarande attributet finns. Attributen är i samma ordning under Types och Query.

Konfiguration av MPASSid:s claim rules

```
$issuanceTransformRules = '@RuleName = "Send MPASSid Attributes"c:[Type == "http://schemas.microsoft.com/ws/2008
/06/identity/claims/windowsaccountname", Issuer == "AD AUTHORITY"] => issue(store = "Active Directory", types =
("mpassUserIdentity", "mpassGivenName", "mpassSurname", "mpassAccountName", "mpassCryptID", "mpassLearnerId",
"mpassMunicipalityCode", "mpassSchoolCode", "mpassClassLevel", "mpassClassCode", "mpassUserRole"), query = ";
objectGuiD,givenName,sn,userPrincipalName,<cryptId>,<learnerId>,<municipalityCode>,<schoolCode>,<classLevel>,
<classCode>,<userRole>;{0}", param = c.Value);';

$issuanceAuthorizationRules = '@RuleTemplate = "AllowAllAuthzRule" => issue(Type = "http://schemas.microsoft.com
/authorization/claims/permit", Value = "true");'

$name = "mpass-proxy"

Set-ADFSRelyingPartyTrust -TargetName $name -IssuanceAuthorizationRules $issuanceAuthorizationRules -
IssuanceTransformRules $issuanceTransformRules
```

Att skicka flera skolkoder

Om eleverna eller lärarna har flera skolor kan du förmedla skolkoderna i attributet mpassSchoolCode separerade med semikolon.

ADFS data till MPASSid

Innan inloggningen fungerar måste ADFS data läggas till i MPASSid. Data tillfogas med hjälp av ADFS metadata. Metadata finns på default-ADFS-servern på adressen https://<serverns_namn>/FederationMetadata/2007-06/FederationMetadata.xml. Skicka ADFS metadata till adressen mpass@oph.fi.

Claim Rules

Nedan finns exempel på claim rule-konfigurationer. Dessa kan kopieras och tillfogas till konfigurationerna av ADFS Relying Party i ADFS administrationskonsol. I konfigurationen måste man ange i vilket attribut i AD det attribut som ska skickas till MPASSid finns.

Claim rule gör en förfrågan till AD och hämtar de attribut som definieras i förfrågan

Under Types anges i vilket format attributen ska skickas till MPASSid och under Query anges i vilket AD-attribut det motsvarande attributet finns.

Ordningen är densamma under Types och Query. Exempelvis hämtas alltså attributet mpassUserIdentity från AD:s attribut objectGuid.

Du kan vid behov lägga till, ändra och ta bort attribut. Om det till exempel inte finns någon kommunkod i användarens uppgifter i AD, ta bort det motsvarande attributet under "mpassMunicipalityCode" i Types och under Query.

```
Send MPASSid Attributes
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD
AUTHORITY"]
 => issue(store = "Active Directory", types = ("mpassUserIdentity", "mpassAccountName", "mpassGivenName", "mpass
Surname", "mpassCryptId", "mpassLearnerId", "mpassMunicipalityCode", "mpassSchoolCode", "mpassClassLevel", "mpas
sClassCode", "mpassUserRole"), query = ";objectGuiD,userPrincipalName,givenName,sn,<cryptID>,<learnerId>,
<municipalityCode>,<schoolCode>,<classLevel>,<classCode>,<userRole>;{0}", param = c.Value);
```

Exempel på olika claim rules

Om du konfigurerar attribut som skickas till MPASSid med separata claim rules, ska du se till att de inte skickas även i något annat attribut.

Skicka kommunkoden som fast värde

```
Send mpassMunicipalityCode
=> issue(Type = "mpassMunicipalityCode", Value = "123");
```

Ändra numeriska rolldata till data i textform

```
#  Hämta användarens roll i AD:s attribut employeType och lägg till (add) den i användarens attribut

Add Employee Type as Role
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD
AUTHORITY"]
 => add(store = "Active Directory", types = ("http://schemas.microsoft.com/ws/2008/06/identity/claims/role"),
query = ";employeeType;{0}", param = c.Value);

# Rollens värde undersöks och en roll i textformat som motsvarar värdet skickas vidare (issue)
Send mpassUserRole oppilas
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/role", Value =~ "^1" ]
 => issue(Type = "mpassUserRole", Value = "oppilas");

Send mpassUserRole opettaja
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/role", Value =~ "^2" ]
 => issue(Type = "mpassUserRole", Value = "opettaja");
```

Skicka rollinformation enligt OU-strukturen.

Detta kan utnyttjas om lärare och elever är i sina egna OU och uppgift om roll inte finns i något attribut i deras användarobjekt.

```
# Första claim rule – hämtas från användarens distinguishedName och placeras i claim:distinguishedName.
# Denna claim vidarebefordras inte från (=> add() vs => issue()).
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer == "AD
AUTHORITY"]
=> add(store = "Active Directory", types = ("claim:distinguishedName"), query = ";distinguishedName;{0}", param
= c.Value);

# Andra claim rule som undersöker om claim innehåller texten OU = Elever. Om så, returneras mpassUserRole som
Elev.
c:[Type == "claim:distinguishedName", Value =~ "(OU=Oppilaat)"]
=> issue(Type = "mpassUserRole", Value = "Oppilas");

# Tredje Claim rule som undersöker om claim innehåller texten OU = Lärare. Om så, returneras mpassUserRole som
Lärare.
c:[Type == "claim:distinguishedName", Value =~ "(OU=Opettajat)"]
=> issue(Type = "mpassUserRole", Value = "Opettaja");
```

Windows Server 2019 ADFS

Windows Server 2019 ADFS lägger som standard till contact person address som ett tomt element i metadata. För att valideringen av metadata ska fungera måste email address vara definierat.

```
# Kontrollera först om contact person redan har angetts:
(Get-AdfsProperties).ContactPerson

# Om EmailAddresses är tomt som resultat av det föregående kommandot, kan du lägga till det här.
# Obs! Detta överskriver eventuella tidigare ContactPerson-konfigurationer, så all information måste ställas in
på nytt.
 $CP = New-AdfsContactPerson [-Company <string>] [-EmailAddress <string[]>] [-GivenName <string>] [-
TelephoneNumber <string[]>] [-Surname <string>]
Set-AdfsProperties -ContactPerson $CP
```