

Microsoft Sharepoint als Group Provider (deprecated)



De tekst op deze pagina is verouderd. De VOOT-omgeving van SURFconext ondersteunt het opvragen van alle leden van een team niet meer.

Deze pagina beschrijft hoe je SharePoint 2010 kunt inzetten als externe Group Provider voor SURFconext. Hiervoor wordt gebruikgemaakt van het VOOT-protocol.

- Waar implementeer je de VOOT REST-interface?
- Groepen gebruikers tonen vanuit SharePoint
- De webservice
- Implementatie van de VOOT-interface
 - Toon mijn eigen groepen
 - Toon alle leden van een van mijn eigen groepen
 - Toon een bepaald persoon
- Authenticatie



Let op! De documentatie op deze pagina is nog niet aangepast naar de nieuwe omgeving. Op de nieuwe omgeving is het niet meer mogelijk om alle leden van een bepaalde groep te tonen. Dit endpoint hoeft je dus niet meer te implementeren op je *group provider* systemen.

Waar implementeer je de VOOT REST-interface?

Je kunt de VOOT REST-interface implementeren binnen of buiten SharePoint. Als je kiest voor de optie om het binnen SharePoint te doen, dan moet je ondersteuning inbouwen voor 'URL rewriting', want zonder aanpassingen op dit gebied werken de vereiste methoden zoals '/groups /@me' niet.

Op internet zijn de meningen erover verdeeld of dit werkt binnen SharePoint 2010. Het lijkt ons in ieder geval niet de juiste manier.

Als je de VOOT REST-interface implementeert buiten SharePoint, dan heb je een aparte webservice nodig om het 'URLrewrite'-probleem op te lossen. Deze service handelt de VOOT REST-interface af en vraagt de benodigde informatie op uit SharePoint. Hoe dat in zijn werk gaat, leggen we hierna uit.

Groepen gebruikers tonen vanuit SharePoint

SharePoint 2010 biedt 2 manieren om groepen gebruikers op te vragen. Dit zijn:

- Client Object Model
- SharePoint 2010 webservices

Microsoft adviseert om het 'Client Object Model' zo veel mogelijk te gebruiken (zie bijvoorbeeld: [Deciding which SharePoint 2010 API to use](#)). Het 'Client Object Model' dient als een wrapper om een webdienst heen, die verzoeken stuurt naar het 'Server Side Object Model'. Het resultaat hiervan zijn objecten uit de 'Object Model Hierarchy'.

De webservice

De webservice beschikt over dezelfde functionaliteit als is beschreven op de pagina: [Microsoft Active Directory als Group Provider](#).

Implementatie van de VOOT-interface

SURFconext ondersteunt de volgende methoden:

- Toon mijn eigen groepen - HTTP GET naar '/groups/@me'
- Toon alle deelnemers van een van mijn eigen groepen - HTTP GET naar '/people/@me/[groep-id]'
- Toon een bepaald persoon - HTTP GET naar '/people/[gebruiker-id]'

De methoden worden hier verder uitgelegd.

Toon mijn eigen groepen

Van SharePoint krijgen we een lijst met groepen, samen met de leden van elke groep:

```
// Get the collection of groups including their list of users.
GroupCollection groupsColl = sharePointContext.Web.SiteGroups;
sharePointContext.Load(groupsColl,
    groupsinc => groupsinc.Include(groupinc => groupinc.Id,
    groupinc => groupinc.Title,
    groupinc => groupinc.Users.Include(userinc => userinc.Title));
sharePointContext.ExecuteQuery();
```

In de code doorlopen we de lijst met groepen en zoeken we naar de desbetreffende gebruiker in de lijst met groepsleden.

Toon alle leden van een van mijn eigen groepen

Het tonen van de lijst met groepsleden van een bepaalde groep gebeurt in 2 stappen. Eerst wordt een lijst getoond met alle groepen in SharePoint door de volgende code:

```
// Get the groups from sharepoint. Just the owner title property.
GroupCollection groupColl = sharePointContext.Web.SiteGroups;
sharePointContext.Load(groupColl, groups => groups.Include(incgroup => incgroup.Title));
sharePointContext.ExecuteQuery();
```

Vervolgens vinden we de bijpassende groep en tonen we de gebruikers van die groep met de volgende code:

```
// Get the user collection.
UserCollection users = group.Users;
sharePointContext.Load(users);
sharePointContext.ExecuteQuery();
```

Het resultaat wordt geretourneerd in JSON-formaat.

Toon een bepaald persoon

Het SharePoint 'Client Object Model' bevat een object met de naam 'SiteUserInfoList', maar dit object is een SPList-object. Dit betekent dat we nog steeds een 'caml query' moeten gebruiken om de informatie over de gebruiker op te halen. De implementatie die we hier hebben gebruikt is dezelfde als de code voor 'toon mijn groepen', behalve dat we niet eerst alle groepen langs hoeven te gaan, maar dat de code kan stoppen na de eerste match met de desbetreffende gebruiker.

Authenticatie

Voor het authenticeren van het eindpunt van de gebruiker gebruiken we '3-legged OAuth versie 1.0a'. Om Single Sign On mogelijk te maken tussen de Service Provider, SURFconext en het externegroepeneindpunt, moet je de OAuth Provider (die verantwoordelijk is voor het aanmaken en uitdelen van toegangs-'tokens' na een succesvolle authenticatie) koppelen met SURFconext.

Hiervoor zijn verschillende .NET-componenten beschikbaar. SURFconext maakt gebruik van de component 'DotNetOpenAuth'. Een voorbeeld hiervan is een 'OpenAuthServiceProvider'. Dit voorbeeld hebben we als basis gebruikt voor de OAuth/implementatie. Het bevat een ASP.NET handler door de 3-legged OAuth handshake. Vervolgens wordt er een VOOT-webservice aan toegevoegd. Met een aparte webapplicatie kun je de functionaliteit van de implementatie testen.