Configure Office 365 with ADFS and SURFconext Step-by-Step

You can federate your local identity store with the Office 365 environment. The main reason to use federated authentication is so users only fill out their password in trusted environments. **This blog** highlights what makes a secure login. When using federated authentication, which Microsoft calls **'modern authentication'**, you can either do this with SURFconext or directly with the Microsoft cloud. Federated authentication is limited by the possibilities of the underlying protocols. Both Microsoft federated login and SURFconext use those protocols. From a functional point it makes no difference whether you federate through SURFconext or using Microsoft federated authentication.

Note that even though users will authenticate using the SAML 2.0 protocol, all your Office 365 users need to be created (provisioned) in Azure AD, in order for 'Office 365' to recognise users that want to login. The tool that is being used to sync your domain users with the Azure AD is called AAD Connect.

When deciding whether to use SURFconext when you choose for federated login, keep in mind: when you contact Microsoft support to troubleshoot issues, it's likely they don't know SURFconext and SAML, and they might assume your problem is related to SURFconext. You could be told to check with the SURFconext team. Troubleshooting in a complex environment with large companies sometimes is complex enough, so it might help when the complete connection is Microsoft-only .

For institutions opting for federated authentication via SURFconext, the below information helps you configure such a connection.

This Step-by-Step guide contains several Powershell scripts and explanations for the following steps:

You can use parts of the scripts or run every step on the servers you want to configure. Be aware that every step has its own variables where you will have to set your own configuration options.

Step 1: Install the ADDS Role and DNS on your server (s)

*****In case you already have a domain set up, you may skip this step and continue with step 2*****

To use the AAD Connect tool and sync your users between your (on-premise) domain and the Azure AD, you would need a domain and a domain controller. This Powershell script, will install the ADDS role and DNS.

Install ADDS Role and DNS

```
########################################### INSTALL ADDS ROLE AND DNS
###########################################
$ComputerName = "YOUR COMPUTER NAME"
$DomainName = "YOUR DOMAIN NAME"
$DatabasePath = "C:\Windows\NTDS"
$DomainMode = "Win2012R2"
$DomainNetbiosName = "YOUR DOMAIN NETBIOSNAME"
$ForestMode = "Win2012R2"
$Logpath = "C:\Windows\NTDS"
$SysvolPath = "C:\Windows\SYSVOL"

#### Get Windows features to check if the ADDS role is available ####
Get-windowsfeature

#### Installing the Active Directory Domain Service ####
Install-windowsfeature AD-Domain-Services

#### Import the required modules for the ADDS Deployment ####
Import-Module ADDSDeployment

#### Install new Domain Controller in a new Forest ####
Install-ADDSForest -DomainName $DomainName -NoDnsOnNetwork -DatabasePath $DatabasePath -DomainMode
$DomainMode -DomainNetbiosName $DomainNetbiosName -ForestMode $ForestMode -LogPath $Logpath -SysvolPath
$SysvolPath -CreateDnsDelegation:$false -InstallDns:$true -NoRebootOnCompletion:$false -Force:$true

#### Install ADDS Tools ####
Import-Module ServerManager
Add-WindowsFeature RSAT-ADDS-Tools
```

Step 2: Create a Group Managed Service Account and install ADFS Role

*****In case you already have AD FS set up, you may skip this step and continue with step 3*****

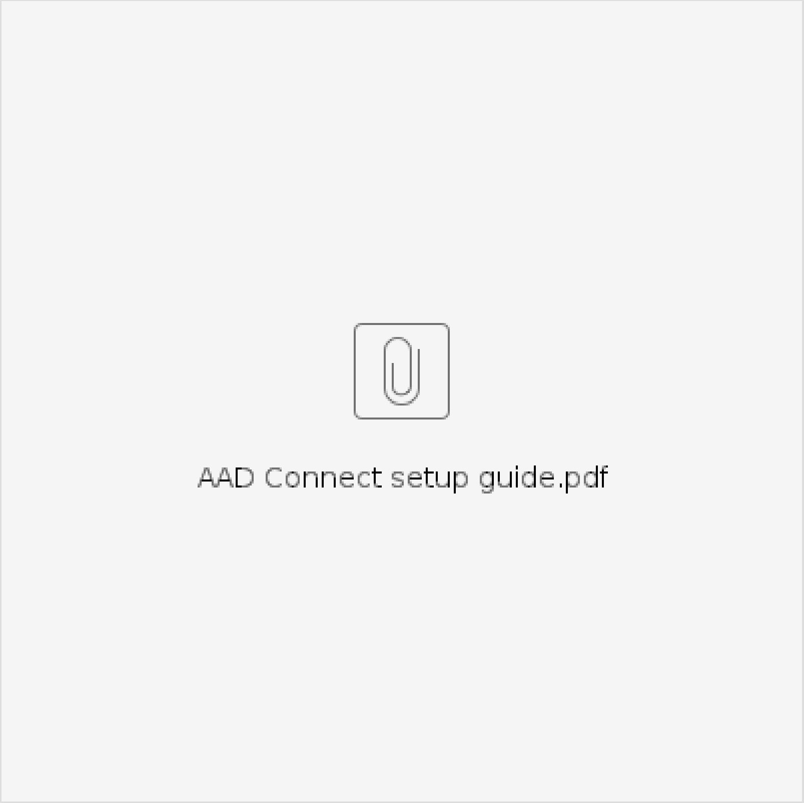
To be able to federate through ADFS, you would need to install the ADFS role. Also you will need a service account for ADFS. We used a Group Managed Account. Check this **blog** for more information about Group Managed Service Accounts.
We used a scenario without an ADFS Proxy (WAP), but you could add an ADFS proxy to this setup.

Create a gMSA and install ADFS Role

```
########################################### INSTALL ADFS ROLE ###########################################
$gMSAName = "gMSA-ADFS"
$DNSHostName = "YOUR ADFS DNS HOSTNAME (EG: adfs.yourdomain.com)"
$ServPrincName = "host/YOURADFSDNSHOSTNAME (EG: host/adfs.yourdomain.com)"
$Path = "SERVICE ACCOUNT PATH (EG: CN=Managed Service Accounts,DC=yourdomain,DC=com"

#### To create a group managed service account, you have to create a KDS Root Key ####
#### Create KDS Root Key (The -10 is only usefull in a testing environment and will ensure immediately
effectiveness) ####
Add-KdsRootKey -EffectiveTime (Get-Date).AddHours(-10)

#### Create new Group Managed Service Account
New-ADServiceAccount -Name $gMSAName -DNSHostName $DNSHostName -ServicePrincipalNames $ServPrincName -Path
$Path

#### Install IIS Role ####
Install-WindowsFeature -name Web-Server -IncludeManagementTools

#### Install ADFS Role ####
Install-windowsfeature adfs-federation -IncludeManagementTools
```

Step 3: Run and finish the AAD Connect tool setup before you continue

You will need to have a working AAD Connect configuration before continuing with the next steps. In case you don't have a working AAD Connect setup, please follow the instructions in the setup guide below.
This guide contains the configuration steps that we used and it is a working configuration for our reference topology. Of course there are many other configurations possible, so please choose the configuration, needed for your topology.



You can download the AAD Connect tool **here**.
You can find more information on supported topologies on **this** page.
There is also more information to be found about the **express** or **custom** installation of AAD Connect.

Step 4: Create the mandatory claims descriptions for SURFconext

****In case you already have a SURFconext connection, you may skip this step and continue with step 5****

In this step you will create the (mandatory) claims descriptions for SURFconext

SURFconext claims descriptions

```
########################################### Create ADFS Claim Descriptions
###########################################
#### ADD UID CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:uid -ClaimType urn:mace:dir:attribute-def:uid -
ShortName uid -IsAccepted $false -IsOffered $false

#### ADD MAIL CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:mail -ClaimType urn:mace:dir:attribute-def:mail -
ShortName mail -IsAccepted $false -IsOffered $false

#### ADD DISPLAYNAME CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:displayName -ClaimType urn:mace:dir:attribute-def:
displayName -ShortName displayName -IsAccepted $false -IsOffered $false

#### ADD schacHomeOrganization CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name schacHomeOrganization -ClaimType urn:mace:terena.org:attribute-def:
schacHomeOrganization -ShortName schacHomeOrganization -IsAccepted $true -IsOffered $true

#### ADD eduPersonAffiliation CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:eduPersonAffiliation -ClaimType urn:mace:dir:
attribute-def:eduPersonAffiliation -ShortName eduPersonAffiliation -IsAccepted $true -IsOffered $true

#### ADD eduPersonEntitlement CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:eduPersonEntitlement -ClaimType urn:mace:dir:
attribute-def:eduPersonEntitlement -ShortName eduPersonEntitlement -IsAccepted $false -IsOffered $false

#### ADD employeeNumber CLAIM DESCRIPTION ####
Add-ADFSClaimDescription -Name urn:mace:dir:attribute-def:employeeNumber -ClaimType urn:mace:dir:attribute-
def:employeeNumber -ShortName employeeNumber -IsAccepted $false -IsOffered $false
```

Step 5: Create the SURFconext Relying Party Trust with the mandatory claims rules

*****In case you already have a SURFconext connection, you may skip this step and continue with step 6*****

In this step you will create (mandatory) claims rules. There are claims rules that are mandatory for SURFconext, but also claims rules that are mandatory for the use of Azure AD.
One of the attributes that will have to be provided, is the ImmutableID. This claims rule is included in the text file below. You can find more information about AD FS claims and Azure AD on this **site**.

Also needed is an attribute containing the user's mail address. Azure AD expects this attribute to have the name `IDPEmail`. However, when your Identity Provider already discloses the standard SURFconext attribute `urn:mace:dir:attribute-def:mail`, its value is automatically mapped to the `IDPEMail` attribute by SURFconext. When you are using this page for your configuration, you won't have to add a separate claims rule for IDPEmail.

You can download the required claim issuance rules file for the $ClaimIssuanceFile parameter here: **ClaimIssuanceRules.txt**

Create SURFconext Relying Party Trust

```
#### CREATE SURFCONEXT RELYING PARTY TRUST ####
$RelyingPartyTrustName = "SURFconext"
$MetaDataURL = "https://metadata.surfconext.nl/sp-metadata.xml"
$ClaimIssuanceFile = "THE LOCATION OF YOUR CLAIM ISSUANCE RULE FILE"
$ACPName = "Permit everyone"

Add-ADFSRelyingPartyTrust -Name $RelyingPartyTrustName -MetadataUrl $MetaDataURL -IssuanceTransformRulesFile
$ClaimIssuanceFile -AutoUpdateEnabled:$true -MonitoringEnabled:$true -AccessControlPolicyName $ACPName
```

Step 6: Configure the SURFconext Federation

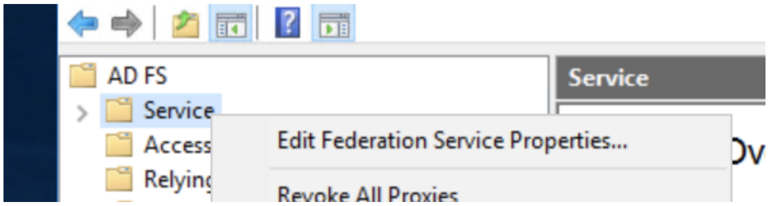
In this step, you will configure the SURFconext Federation for Office 365. You will have to send your metadata URL to SURFconext, so they can configure it. This metadata URL usually looks like this: https://adfs.yourdomain.nl/FederationMetadata/2007-06/FederationMetadata.xml .
You will receive a passive logon URI from SURFconext, that you will have to use for the $sso parameter in the script below. You can find the certificate key on **this** page, that you will need for the $crt parameter in the script below.

Configure the SURFconext Federation

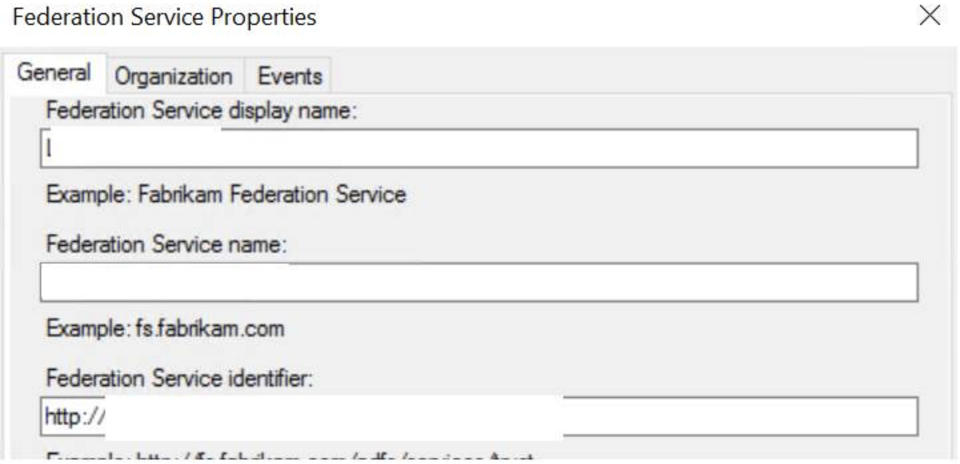
```
########################################### CONFIGURE THE SURFCONEXT FEDERATION
###########################################
$dom = "YOUR DOMAIN NAME"
$slo = "https://engine.surfconext.nl/logout"
$idp = "YOUR FEDERATION SERVICE IDENTIFIER URL"
$crt =
"MIID7DCCAtSgAwIBAgIJAIgMqnMYZ+t6MA0GCSqGSIb3DQEBCwUAMIGFMQswCQYDVQQGEwJOTDEQMA4GA1UECAwHVXRyZWNodDEQMA4GA1UE
BwwHVXRyZWNodDEVMBMGA1UECgwMU1VSRm5ldCBCLlYuMRMwEQYDVQQLDApTVVJGY29uZXh0MSYwJAYDVQQDDB1lbmdpbmUuc3VyZmNvbmV4d
C5ubCAyMDE4MTIxMzAeFw0xODEyMTMxNTI5MjBaFw0yMzEyMTMxNTI5MjBaMIGFMQswCQYDVQQGEwJOTDEQMA4GA1UECAwHVXRyZWNodDEQMA
4GA1UEBwwHVXRyZWNodDEVMBMGA1UECgwMU1VSRm5ldCBCLlYuMRMwEQYDVQQLDApTVVJGY29uZXh0MSYwJAYDVQQDDB1lbmdpbmUuc3VyZmN
vbmV4dC5ubCAyMDE4MTIxMzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALPOGS+fBERfmWiV8aV85z45QsuFw3gkq0HbWR1JGz7c
jqhjV6YZHFXyRt4ikG//9BIHS0xc/cW1sOMnSuCjDhY8Oh/dOk01zfgFXUcv+0iNlkEKGMlT/xJpIDIy
/N4WjpGvkJO2oJHfrQUY115Du56MSMqd0gPvo1OsDvXroYivqxYpTTHzaf5TYQYPf6n/3rEfsu3u6L3pzE3/q38jnEyxfQ1UoZ9VF2Fy6oe
/StlwhPUJhVwHlKDMqQ+T+tljDt26Ok9QL3zzW9JtBo+pnydMT
/rg5h7NW8A9HASLnRLK8rFD9nBEdAPkK+elTE6QddRiTh9H84KCs0fQiiT6YFsCAwEAAaNdMFswHQYDVR0OBBYEFAJuZa7u0f0o2kB9uRPoB
/ekx04sMB8GA1UdIwQYMBaAFAJuZa7u0f0o2kB9uRPoB/ekx04sMAsGA1UdDwQEAwIHgDAMBgNVHRMEBTADAQH
/MA0GCSqGSIb3DQEBCwUAA4IBAQBXh5l8u+ncPXkMyDqDuikNLe/X5j0KNjvqUtQ6QPRSt8MMvjRYWZdVC0gMOtKEAY1/cYnA2y+0yrGqmy9I
/zBdLV73BBLnVlV2WYATYOZLWNW36kjBtdSbH0oXBp7HOu/I4lP+Sv69eRN6p2
/9CmDyKc5JUpXU3PEftv5Lwsqco8MMqqENhwzYlxRb96LFq08Un2QQoV60HqX4Ks79qUrnjRL5pKtoP4ujLmPqQIieHpTgsvHSqSa+9tZMnyE
aJEvl7vpNn1M7v1bWOWwjQvMlYnSq5b0U5gHXgpdBYSfWnCwwpq4h8KHZ7/XVvOVsdYpjHap+907OGhqXGBsIqf9U"
$sso = "THE PASSIVE LOGON URI YOU RECEIVED FROM SURFCONEXT"


#### CONNECT TO OFFICE 365 ####
Connect-MsolService

#### SET THE AUTHENTICATION TO MANAGED FIRST ####
Set-MsolDomainAuthentication -DomainName $dom -Authentication Managed

#### SET THE AUTHENTICATION TO FEDERATED, INCLUDING THE FEDERATION SETTINGS ####
Set-MsolDomainAuthentication -DomainName $dom -FederationBrandName $dom -Authentication Federated -
PassiveLogOnUri $sso -SigningCertificate $crt -IssuerUri $idp -LogOffUri $slo -
PreferredAuthenticationProtocol Samlp
```

Regarding the value for '*$idp = "YOUR FEDERATION SERVICE IDENTIFIER URL"':*

- is the URL that is configured in AD FS. You can find this at the following location: Start -> Administrative Tools -> AD FS Management .
- right click on Service in the AD FS management console and click Edit Federation Service Provider



- Use the value from *Federation Service Identifier URL*:



Federation Service Properties

General | Organization | Events

Federation Service display name:

|

Example: Fabrikam Federation Service

Federation Service name:

Example: fs.fabrikam.com

Federation Service identifier:

http://

Example: http://fs.fabrikam.com/adfs/services/trust

Regarding the value for '*$sso = "THE PASSIVE LOGON URI YOU RECEIVED FROM SURFCONEXT"*':

- That value is included in the SAML metadata. If you don't know how to retrieve it, you can request SURF via an email to support@surfconext.nl in which you explain what you're doing and you would like to receive the 'instellings-specifieke SSO-locatie'.
- A sample value looks like "https://engine.surfconext.nl/authentication/idp/single-sign-on/key:20181213/377be5f3a9544971303163432419360" (fictitious value)

Step 7: Set Modern authentication on Exchange Online to be able to use rich clients

When your users are using Microsoft Outlook or other rich clients, you will have to enable **modern authentication**. This can be done with the script below.

Set modern authentication on Exchange Online

```
############################ SET MODERN AUTHENTICATION TO BE ABLE TO USE RICH CLIENTS SUCH AS OUTLOOK
#################################

#### CREATE EXCHANGE ONLINE SESSION ####
$Session = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionUri https://outlook.office365.com
/powershell-liveid/ -Credential $UserCredential -Authentication Basic -AllowRedirection

#### IMPORT EXCHANGE ONLINE SESSION ####
Import-PSSession $Session

#### SET MODERN AUTHENTICATION TO TRUE ####
Set-OrganizationConfig -OAuth2ClientProfileEnabled $true

#### REMOVE EXCHANGE ONLINE SESSION ####
Remove-PSSession $Session
```

Debugging and helpful tools

To check if you send the required claims to SURFconext, you can use **this** debug tool.