

# **ACCESSING INTERNAL (APEX) APPLICATIONS USING APPLICATION PROXY IN AZURE ACTIVE DIRECTORY**

Author: Niels de Bruijn & Tim Theisen  
Version: 1.0  
Date: 04-FEB-2021

## 1 INTRODUCTION

A lot of organizations have their internal APEX environment single sign-on enabled based on either kerberos, OAuth2 or SAMLv2. If you haven't done this yet, I consider having a look at the following step-by-step tutorials:

[https://knowledgebase.mt-ag.com/q/apex\\_sso\\_kerberos](https://knowledgebase.mt-ag.com/q/apex_sso_kerberos)

[https://knowledgebase.mt-ag.com/q/apex\\_sso\\_oauth2](https://knowledgebase.mt-ag.com/q/apex_sso_oauth2)

[https://knowledgebase.mt-ag.com/q/apex\\_sso\\_samlv2](https://knowledgebase.mt-ag.com/q/apex_sso_samlv2)

Let us assume you have SSO set up and Azure AD is already used by your company (ie. due to Office365), thereby synchronizing your AD user accounts with Azure AD.

Now, to gain access to your APEX environment through the internet, you would normally setup a VPN connection to pretend that you are in the corporate network. However, if VPN is not an option, but you still want to provide a secure access to your APEX environment without exposing it to the public network, this document is for you. With the advent of Application Proxy in MS Azure, you can work securely with your internal APEX environment over the internet even without first establishing a VPN connection!

In cases where kerberos is used for authentication internally, this also enables you to use non-domain devices (like MacBooks or smartphones) when accessing your APEX environment through the internet.

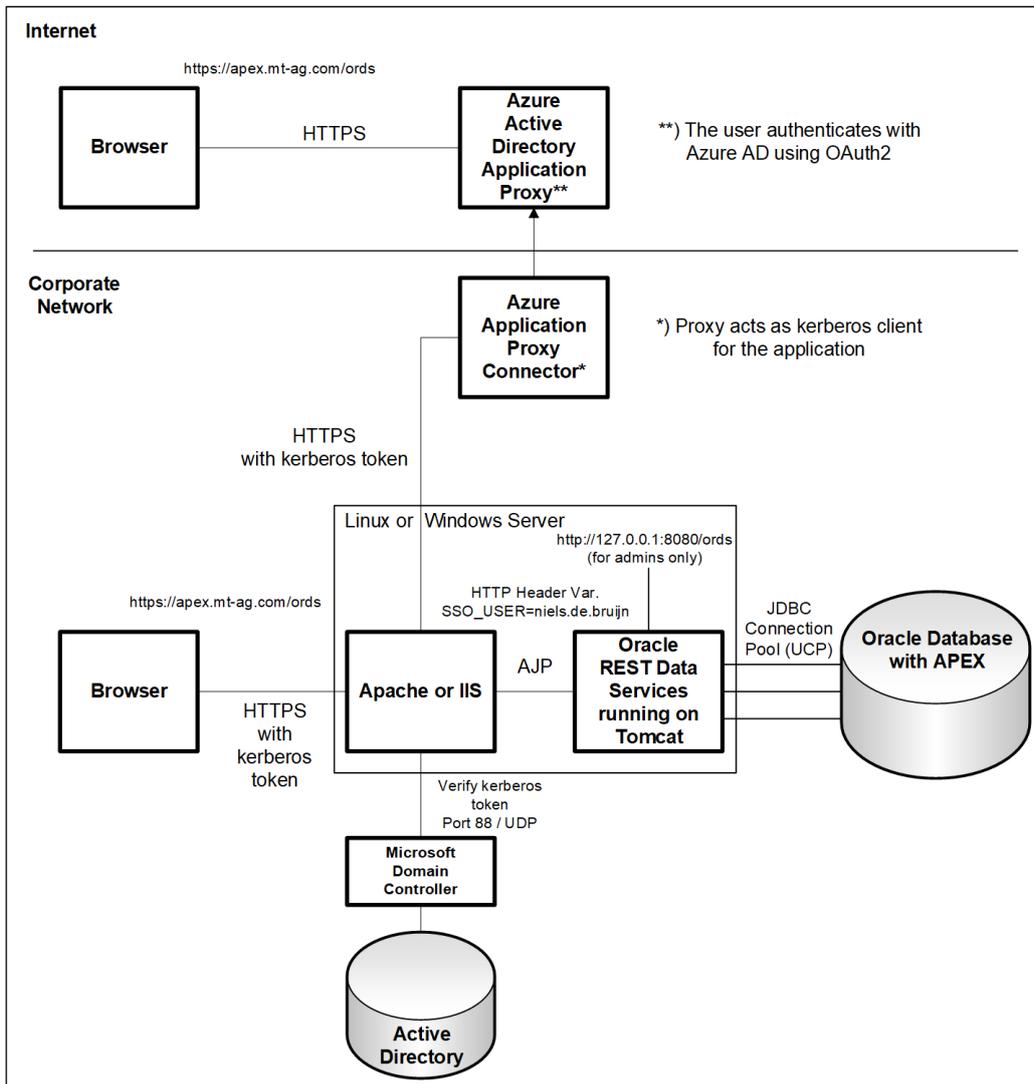


Image 1: Accessing an internal APEX environment through the internet without VPN.

To better understand how it all works together, here is what happens when you access the APEX environment from the internet after starting your browser:

1. The user enters the APEX URL to access the on-premises application through Application Proxy.
2. Application Proxy redirects the request to Azure AD authentication services to preauthenticate. At this point, Azure AD applies any applicable authentication and authorization policies, such as multifactor authentication. If the user is validated, Azure AD creates a token and sends it to the user.
3. The user passes the token to Application Proxy.
4. Application Proxy validates the token and retrieves the User Principal Name (UPN) from it, and then the Connector pulls the UPN, and the Service Principal Name (SPN) through a dually authenticated secure channel.
5. The Connector performs Kerberos Constrained Delegation (KCD) negotiation with the on-premises AD, impersonating the user to get a Kerberos token to the application.
6. Active Directory sends the Kerberos token for the application to the Connector.

7. The Connector sends the original request to the application server, using the Kerberos token it received from AD.

8. The application sends the response to the Connector, which is then returned to the Application Proxy service and finally to the user.

Remarks:

- To get this all to work, you will need a Microsoft Azure AD premium subscription
- In this document, we assume that the APEX environment uses kerberos for authentication, but SAMLv2 and OAuth2 are also supported.
- Here you can find more general documentation about this concept:

<https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/application-proxy>

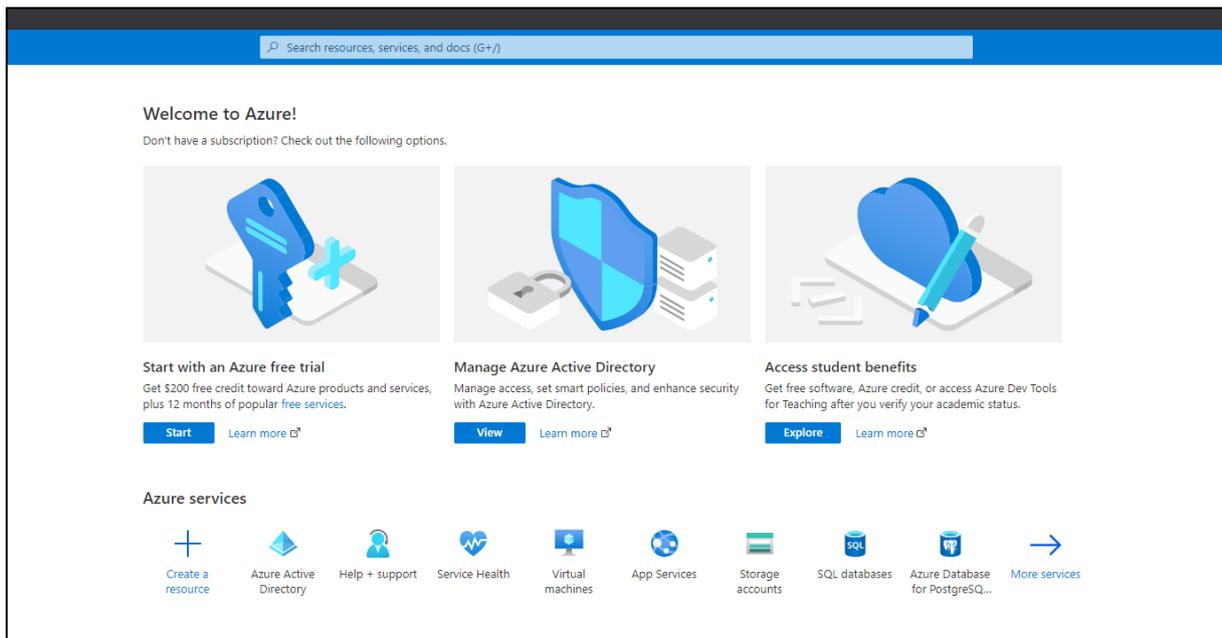
## 2 SETUP MS APPLICATION PROXY

### 2.1 INSTALL APPLICATION PROXY CONNECTOR

To install the Application Proxy Connector, you need a Windows server running Windows Server 2012 R2 or later. The connector connects to the Application Proxy service in Azure AD as well as the on-premises applications that you plan to publish. See <https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/application-proxy-add-on-premises-application> to learn how to set up the connector.

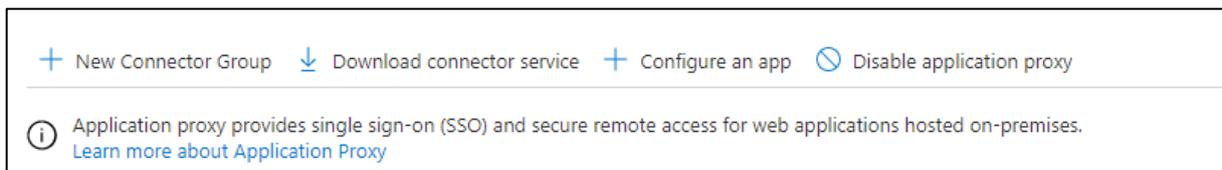
The connector server and the web applications servers should belong to the same Active Directory domain or span trusting domains. Having the servers in the same domain or trusting domains is a requirement for using single sign-on (SSO) with Integrated Windows Authentication (IWA) and Kerberos Constrained Delegation (KCD). Also, make sure that all internal users in AD are automatically replicated in Azure AD by using either “Azure PTA (passthrough authentication)” or “ADFS”.

To download the connector, logon to <https://portal.azure.com>.



Click on “View” at “Manage Azure Active Directory”.

Within “Application proxy”, click on “Download connector service”.



## 2.2 CONFIGURE ACTIVE DIRECTORY

In Active Directory:

- Go to “Tools” > “Users and Computers”.
- Select the server running the connector.
- Right-click and select “Properties” > “Delegation”.
- Select “Trust this computer for delegation to specified services only”.
- Select “Use any authentication protocol”.

Under “Services” to which this account can present delegated credentials add the value for the SPN identity of the application server. This enables the Application Proxy Connector to impersonate users in AD against the applications defined in the list.

Managed By	Object	Security	Dial-in	Attribute Editor	BitLocker Recovery
General	Operating System	Member Of	Delegation	Password Replication	Location

Delegation is a security-sensitive operation, which allows services to act on behalf of another user.

Do not trust this computer for delegation  
 Trust this computer for delegation to any service (Kerberos only)  
 Trust this computer for delegation to specified services only

Use Kerberos only  
 Use any authentication protocol

Services to which this account can present delegated credentials:

Service Type	User or Computer	Port	Service
HTTP	apex-physical.mt-ag.com		

< III >

Expanded

## 2.3 REGISTER AN APP FOR AZURE APPLICATION PROXY

Go back to Azure Active Directory, <https://portal.azure.com>.

Within “Application Proxy”, click on “Configure an app”.

+ New Connector Group
↓ Download connector service
+ Configure an app
⊘ Disable application proxy

---

ⓘ Application proxy provides single sign-on (SSO) and secure remote access for web applications hosted on-premises. [Learn more about Application Proxy](#)

Fill out the form as shown below and click on “Add”.

+ Add
✕ Discard

Application proxy provides single sign-on (SSO) and secure remote access for web applications hosted on-premises. [Learn more about Application Proxy](#)

**Basic Settings**

Name \* ⓘ  ✓

Internal Url \* ⓘ  ✓

External Url ⓘ  📄

Pre Authentication ⓘ  ▼

Connector Group ⓘ  ▼

**Additional Settings**

Backend Application Timeout ⓘ  ▼

Use HTTP-Only Cookie ⓘ  Yes  No

Use Secure Cookie ⓘ  Yes  No

Use Persistent Cookie ⓘ  Yes  No

Translate URLs In

Headers ⓘ  Yes  No

Application Body ⓘ  Yes  No

ⓘ To access your application using a custom domain you must configure a CNAME entry in your DNS provider which points 'apex.mt-ag.com' to 'apex-mtagratingen.msappproxy.net'

Important: Set an CNAME entry in your public DNS as described above.

Important remarks from Microsoft about the persistent cookie setting:

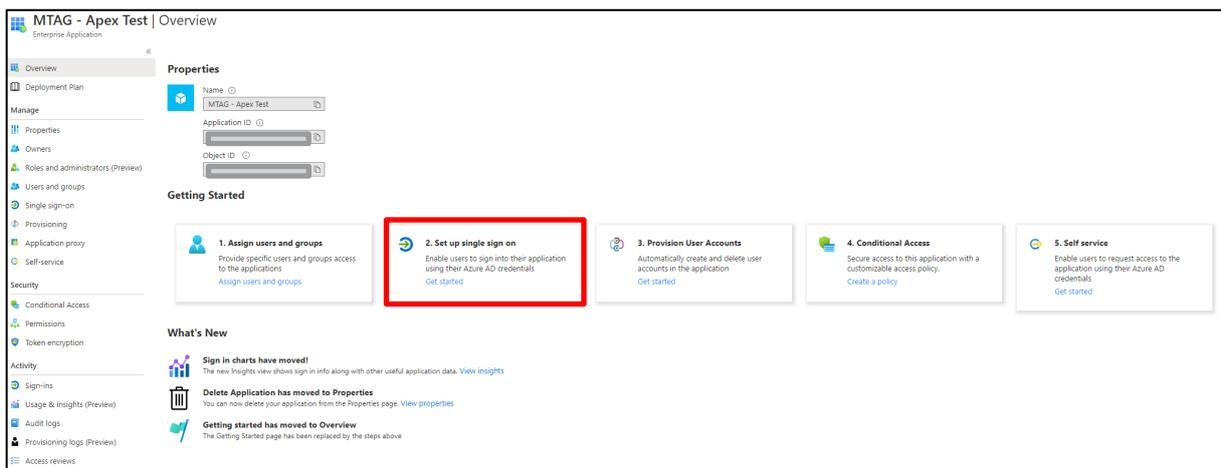
*When a user authenticates with the Microsoft identity platform, a single sign-on session (SSO) is established with the user's browser and the Microsoft identity platform. The SSO token, in the form of a cookie, represents this session. The SSO session token is not bound to a specific resource/client application. SSO session tokens can be revoked, and their validity is checked every time they are used.*

*The Microsoft identity platform uses two kinds of SSO session tokens: persistent and nonpersistent. Persistent session tokens are stored as persistent cookies by the browser. Nonpersistent session tokens are stored as session cookies. (Session cookies are destroyed when the browser is closed.) Usually, a nonpersistent session token is stored. But, when the user selects the Keep me signed in check box during authentication, a persistent session token is stored.*

*Nonpersistent session tokens have a lifetime of 24 hours. Persistent tokens have a lifetime of 90 days. Anytime an SSO session token is used within its validity period, the validity period is extended another 24 hours or 90 days, depending on the token type. If an SSO session token is not used within its validity period, it is considered expired and is no longer accepted.*

Go back to Azure Active Directory and click on "Enterprise applications".

Search for your application, open it and choose "Set up Single Sign-On".



The screenshot shows the Azure Active Directory portal for an enterprise application named "MTAG - Apex Test". The "Getting Started" section is visible, with the second step, "2. Set up single sign on", highlighted with a red box. The other steps are: 1. Assign users and groups, 3. Provision User Accounts, 4. Conditional Access, and 5. Self service. The "What's New" section below shows updates regarding sign-in charts, deleting applications, and the new Getting Started page.

Fill out the form as shown below:

**MTAG - Apex Test | Configure Integrated Windows Authentication (IWA)**  
Enterprise Application

Save Discard Disable Change single sign-on modes

Overview  
Deployment Plan  
**Manage**  
Properties  
Owners  
Roles and administrators (Preview)  
Users and groups  
Single sign-on  
Provisioning  
Application proxy  
Self-service

Configure Integrated Windows Authentication (IWA)

Internal Application SPN \* ⓘ

Delegated Login Identity \* ⓘ

**i** The Application Proxy connector must be installed on a computer that is domain joined for Integrated Windows Authentication to work. ⓘ

Go to “Application proxy”, open it and choose “Click here to upload a certificate”

**⚠ Certificate** >

[Click here to upload a certificate](#)

Upload your \*.pfx certificate file

**SSL certificate**

Subject	Thumbprint	Expiry Date
No SSL certificate is bound.		

File ⓘ  
 

Password

[Upload Certificate](#)

That’s it. When you now open <https://apex.mt-ag.com/ords> on the internet, you first need to authenticate against Azure AD and then you will see the APEX environment. Using :APP\_USER in APEX, you will get the logged on user and act upon it.

**Disclaimer:**

Just to make sure: MT AG is not responsible for any damage, outages or loss of profit resulting from the usage of this document. Use it at your own risk. Have fun!