Contents

Set-OfficeWebAppsExcelUserDefinedFunction

Set-OfficeWebAppsFarm

Set-OfficeWebAppsMachine

Office Online Server

3/24/2021 • 2 minutes to read

Summary: Contains articles that will help you work with Office Online Server (the next version of Office Web Apps Server).

Audience: IT Professionals

Use the articles in the following table to learn about Office Online Server and Office Online with SharePoint Server. Office Online Server is the next version of Office Web Apps Server. It is an on-premises server.

ARTICLE	DESCRIPTION
Plan Office Online Server	Describes planning considerations for setting up your Office Online Server farm.
Deploy Office Online Server	Explains how to deploy Office Online Server in a test or production environment.
Configure Office Online Server for SharePoint Server 2016	Explains how to configure SharePoint Server to use Office Online Server for document editing.
Apply software updates to Office Online Server	Explains how to apply software updates to your Office Online Server farm.
Office Online Server release schedule	Explains the release schedule for new Office Online Server builds, support dates, and upgrade requirements.

Office Online Server can be downloaded from the Volume Licensing Service Center (VLSC). Office Online Server is a component of Office; therefore, it will be shown under each of the Office product pages including Office Standard 2016, Office Professional Plus 2016, and Office 2016 for Mac Standard.

For customers whose licenses qualify for OOS, but cannot obtain it through the VLSC, the following actions are possible:

- VL Open customers can contact their Support Center.
- Customers who purchased O365 online from Microsoft can submit a request from their Office 365 admin center or contact support.

Office Online Server version compatibility list

The following table shows the compatibility between Office Web Apps Server and Office Online Server with SharePoint Server, Exchange Server, and Skype for Business Server.

PRODUCT	OFFICE WEB APPS SERVER	OFFICE ONLINE SERVER
SharePoint Server 2013	Yes	Yes*
SharePoint Server 2016	No	Yes

PRODUCT	OFFICE WEB APPS SERVER	OFFICE ONLINE SERVER
SharePoint Server 2019	No	Yes
Lync Server 2013	Yes	Yes
Skype for Business Server 2015	Yes	Yes
Skype for Business Server 2019	Yes	Yes
Exchange Server 2013	No	No
Exchange Server 2016	No	Yes
Exchange Server 2019	No	Yes

NOTE

SharePoint Server 2013 cannot use the Excel Online external data connectivity and data refresh functionality in Office Online Server. This functionality is available starting with SharePoint Server 2016.

Office Online Server overview

3/24/2021 • 4 minutes to read

Summary: Learn about Office Online Server and how it provides browser-based Office functionality to supported hosts.

Audience: IT Professionals

Office Online Server delivers browser-based versions of Word, PowerPoint, Excel, and OneNote. A single Office Online Server farm can support users who access Office files through SharePoint Server, Exchange Server, shared folders, and web sites.

> **IMPORTANT**
>
> **Are you looking for help with Office Online on your desktop or mobile device?** You can find this information by searching for "Office Online" on Office Support.

About Office Online Server

Office Online Server is an Office server product that provides browser-based file viewing and editing services for Office files. Office Online Server works with products and services that support WOPI, the Web app Open Platform Interface protocol. These products, known as hosts, include SharePoint Server, and Exchange Server. An Office Online Server farm can provide Office services to multiple on-premises hosts, and you can scale out the farm from one server to multiple servers as your organization's needs grow. Office Online Server requires dedicated servers that run no other server applications, however, you can install Office Online Server on virtual machines if needed.

With Office Online Server, users can also view Office files that are stored outside SharePoint Server, such as those in shared folders or other web sites. This functionality is provided by a feature known as Online Viewers.

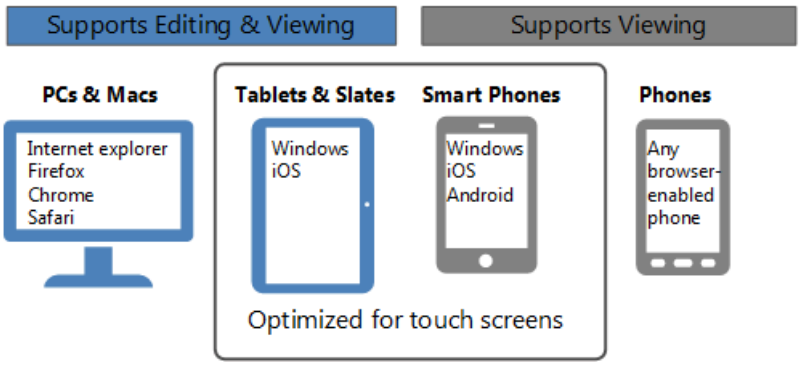
How SharePoint Server uses Office Online Server for viewing and editing Office documents

When used with SharePoint Server 2016, Office Online Server provides Word Online, Excel Online, PowerPoint Online, and OneNote Online. Users can view and, in some cases, edit Office documents in SharePoint libraries by using a supported web browser on computers and on many mobile devices, such as Windows Phones, iPhones, iPads, and Windows tablets.

Note that Office Online Server only works with SharePoint web applications that use claims-based authentication.

The following illustration summarizes the viewing and editing capabilities of Office Online on different kinds of devices.

Viewing and editing capabilities of Office Web Apps



Excel Online includes external data connectivity and data refresh features similar to those found in Excel Services in SharePoint Server 2013. (Excel Services has been removed from SharePoint in SharePoint Server 2016 - you use Excel Online instead.)

How Exchange Server and Outlook Web App use Office Online Server for previewing Office file attachments

When Exchange Server is configured to use Office Online Server, users of Outlook Web App can preview Office file attachments by using Word Online, Excel Online, and PowerPoint Online. These previews provide rich, full-fidelity viewing of Office files and any comments within them, without downloading the files before viewing them.

By default, the following file types are displayed using Office Online Server:

- Word documents (doc, docx, dotx, dot, dotm extensions)
- Excel documents (xls, xlsx, xlsm, xlm, xlsb extensions)
- PowerPoint documents (ppt, pptx, pps, ppsx, potx, pot, pptm, potm, ppsm extensions)

> **NOTE**
> Office Online Server won't be used to render any attachments in IRM protected messages.

Office Online integration for attachment previews is available to all Exchange Online customers. Exchange on-premises customers have to deploy Office Online Server to enable the functionality.

For more information about how to configure Exchange Server to use Office Online Server, see Office Online Server Integration.

How Office Online Server enables users to view Office files in shared folders and websites by using Online Viewers

Online Viewers enable users to use a web browser to view Excel, PowerPoint and Word files that are stored on web servers or shared folders in an organization. Users can conveniently view Office files in a web browser without having to open a separate application. In addition, Online Viewers do not require Office to be installed on users' computers. Online Viewers also generate the code that is required to link or embed the URL inside a webpage. You can use Online Viewers within your Intranet, or on the Internet.

Office Online Server provides a page at the address http:// *OfficeWebAppsServername*/op/generate.aspx that you can use to generate links to publicly available documents that have UNC or URL addresses. When a user selects a generated URL, Online Viewers enable Office Online Server to get the file from its location and then render it by using Office Online. The user can view the Word, Excel, or PowerPoint file in a browser with Office

features intact. Formatting and layout in Word documents are preserved, data in Excel workbooks can be filtered and sorted, and animations play in PowerPoint presentations. However, be aware that Online Viewers allow users to view but not edit files, and Online Viewers can't open any files that require authentication.

You can find more about Online Viewers in Planning for Online Viewers with Office Online Server.

> **NOTE**
> Microsoft hosts an Internet-only facing version of Create URL on Office.com.

See also

Plan Office Online Server

Deploy Office Online Server

Plan Office Online Server

7/14/2021 • 16 minutes to read

Summary: Describes Office Online Server requirements and prerequisites, including HTTPS, certificates, virtualization, load balancing, topologies, and security.

Audience: IT Professionals

Office Online Server delivers browser-based versions of Office apps in an on-premises environment, giving users more flexibility and collaboration opportunities. This article describes the requirements and steps you need to take to install Office Online Server in your organization.

It's important to carefully plan so that all hosts, such as SharePoint Server and Exchange Server can communicate with Office Online Server.

Check the Office Online Server version compatibility list to ensure that your hosts are compatible.

Software, hardware, and configuration requirements for Office Online Server

You can install Office Online Server as a single-server farm, or as a multi-server, load-balanced farm. You can use physical servers or virtual machines.

In environments that contain actual user data, we always recommend that you use HTTPS, for which you have to obtain a certificate. If you're using multiple servers in your farm, you have to configure a hardware or software load-balancing solution. You can learn more about these scenarios in the following sections.

Hardware requirements for Office Online Server

Office Online Server uses the same minimum hardware requirements as SharePoint Server 2016.

Supported operating systems for Office Online Server

You can run Office Online Server on the following operating systems:

- The 64-bit edition of Windows Server 2012 R2
- The 64-bit edition of Windows Server 2016 (Office Online Server November 2018 or later required).

> **NOTE**
> Office Online Server only supports the "Server with Desktop Experience" installation option of Windows Server 2016. For additional information about Windows Server offerings, see Windows Server Semi-annual Channel Overview.

Domain requirements for Office Online Server

All servers in the Office Online Server farm must be part of a domain. They can be in the same domain (recommended) or in domains that are in the same forest.

If you plan to use any Excel Online features that utilize external data access (such as Data Models, Power Pivot, or Power View), Office Online Server must reside in the same Active Directory forest as its users as well as any external data sources that you plan to access using Windows-based authentication.

Support schedule and upgrade requirements

Microsoft releases a new build of Office Online Server every six months or so. Once a new build has been

released, critical updates are no longer produced for the previous build. We highly recommend that you update your Office Online Server farm as new builds are released. For more information, see Office Online Server release schedule.

Compatibility with other workloads and services

Here are a few things to be aware of when you install Office Online Server.

- **Don't install any other server applications on the server that's running Office Online Server**. This includes Exchange Server, SharePoint Server, Skype for Business Server, and SQL Server. If you have a shortage of servers, consider running Office Online Server in a virtual machine on one of the servers you have.
- **Don't install any services or roles that depend on the Web Server (IIS) role on port 80, 443, or 809** because Office Online Server periodically removes web applications on these ports.
- **Don't install any version of Office**. If it's already installed, you'll need to uninstall it before you install Office Online Server.
- **Don't install Office Online Server on a domain controller**. It won't run on a server with Active Directory Domain Services (AD DS).

Support for virtualizing Office Online Server

Office Online Server is supported when you deploy it using Windows Server Hyper-V or other virtualization technology in your on-premises datacenter. If you plan to virtualize Office Online Server, follow these guidelines:

- Install Office Online Server in its own virtual machine. Don't install any other server applications, such as SharePoint Server, in this virtual machine.
- When using Hyper-V for multi-server Office Online Server farms, each virtual machine should be on a separate virtual machine host. This way, the Office Online Server farm will still be available if one of the hosts fails.

Firewall requirements for Office Online Server

Firewalls can cause problems by blocking communication between the web browser, the servers that run Office Online Server, and the servers that run SharePoint Server. These problems can be more complicated when the servers are in different parts of a network.

Make sure the following ports aren't blocked by firewalls on either the server that runs Office Online Server or the load balancer:

- Port 443 for HTTPS traffic
- Port 80 for HTTP traffic
- Port 809 for private traffic between the servers that run Office Online Server (if you're setting up a multi-server farm)

Load balancer requirements for Office Online Server

We recommend a load balancing solution when you run Office Online Server on two or more servers. Just about any load balancing solution will work, including a server that runs the Web Server (IIS) role running Application Request Routing (ARR). In fact, you can run ARR on one of the servers that runs Office Online Server.

Ideally, try to find a load balancing solution that supports the following features:

- Layer 7 routing
- Enabling client affinity or front-end affinity

If you use a load balancer, you'll need to install the certificate on the load balancer as described under Securing Office Online Server communications by using HTTPS .

DNS requirements for Office Online Server

In environments that use HTTPS and load balancing, you have to update DNS so that the fully qualified domain name (FQDN) of the certificate resolves to either the IP address of the server that runs Office Online Server or to the IP address assigned to the load balancer for the Office Online Server farm.

Planning language packs for Office Online Server

Office Online Server Language Packs enable users to view web-based Office files in multiple languages from SharePoint Server document libraries, Outlook Web App (as attachment previews), and Skype for Business Server (as PowerPoint broadcasts). But, this depends on the languages that are configured on the host. To view web-based Office files from hosts in multiple languages, you must have the following in place:

- The host (such as SharePoint Server or Exchange Server) is configured to run applications in additional languages. The process of installing and configuring language packs on the host is independent of installing a language pack on the Office Online Server farm.
- The languages are installed and are available on all servers in the Office Online Server farm.

Here's where to download the language packs for Office Web Apps Server.

Topology planning for Office Online Server

At a minimum, an Office Online Server topology will include one physical or virtual machine running Office Online Server, and at least one host (for example, a server running Exchange Server or SharePoint Server). And of course, you'll need a client PC or device to connect to one of the hosts and use the functionality. From that minimal topology, you can add more hosts and more servers to your Office Online Server farm as required to suit the needs of your organization.

The following is a list of recommendations that you should keep in mind as your Office Online Server topology gets more complex.

- **Plan for redundancy.** If you use virtual machines, make sure you put them on separate virtual machine hosts for redundancy.
- **Stick to one data center.** Servers in an Office Online Server farm must be in the same data center. Don't distribute them geographically. Generally you need only one farm, unless you have security needs that require an isolated network that has its own Office Online Server farm.
- **The closer the hosts, the better.** The Office Online Server farm doesn't have to be in the same data center as the hosts it serves, but for heavy editing usage, we recommend you put the Office Online Server farm as close to the hosts as possible. This is less important for organizations that use Office Online primarily for viewing Office files.
- **Plan your connections.** Connect all servers in the Office Online Server farm only to one another. To connect them to a broader network, do so through a reverse proxy load balancer firewall.
- **Configure the firewall for HTTP or HTTPS requests.** Make sure the firewall allows servers running Office Online Server to initiate HTTP or HTTPS requests to hosts.

- **Plan for incoming and outgoing communications.** In an Internet-facing deployment, route all outgoing communications through a NAT device. In a multi-server farm, handle all incoming communications with a load balancer.
- **Make sure all servers in the Office Online Server farm are joined to a domain and are part of the same organizational unit (OU).** Use the **FarmOU** parameter in the New-OfficeWebAppsFarm cmdlet to prevent other servers that are not in this OU from joining the farm.
- **Use Hypertext Transfer Protocol Secure (HTTPS) for all incoming requests.**
- **If you have IPsec deployed in the network, use it to encrypt traffic among the servers.**
- **Plan for Office features that use the Internet.** If features such as clip art and translation services are needed, and the servers in the farm can't initiate requests to the Internet, you'll need to configure a proxy server for the Office Online Server farm. This will allow HTTP requests to external sites.

Plan Excel Online external data connectivity

Excel Online includes external data connectivity and data refresh features similar to those found in Excel Services in SharePoint Server 2013. Excel Services has been removed from SharePoint in SharePoint Server 2016 - you use Excel Online instead.

Data refresh for embedded data connections works with a standard Office Online Server installation. However, more advanced features, including Office Data Connection (ODC) file support and the IT Management Dashboard (part of SQL Server Power Pivot for SharePoint) require that you configure server-to-server authentication between Office Online Server and SharePoint Server 2016.

Security planning for Office Online Server

The following information introduces security guidance for Office Online Server.

Securing Office Online Server communications by using HTTPS

Office Online Server can communicate with SharePoint Server, Skype for Business Server, and Exchange Server by using the HTTPS protocol. In production environments, we strongly recommend that you use HTTPS. You'll have to install an Internet Server certificate that can be assigned to the server that runs Office Online Server (if you are using a single server) or to the load balancer (if you are using multiple servers that run Office Online Server).

In test environments that contain no user data, you can use HTTP for SharePoint Server and Exchange Server and skip the certificate requirement. Skype for Business Server supports only HTTPS.

Certificates used by Office Online Server need to meet the following requirements:

- The certificate must come from a trusted Certificate Authority and include the fully qualified domain name (FQDN) of your Office Online Server farm in the SAN (Subject Alternative Name) field. (If the FQDN is not in the SAN when you try to use the certificate, the browser will either show security warnings or won't process the response.)
- The certificate must have an exportable private key. On single-server farms, this option is selected by default when you use the Internet Information Services (IIS) Manager snap-in to import the certificate.
- The Friendly name field must be unique within the Trusted Root Certificate Authorities store. If you have multiple certificates that share a Friendly Name field, farm creation will fail because the New-OfficeWebAppsFarm cmdlet won't know which of those certificates to use.

- Office Online Server doesn't require any special certificate properties or extensions. For example, Client Enhanced Key Usage (EKU) extensions or Server EKU extensions are not required.
- You must install the "Allow HTTP Activation" Windows Communication Foundation (WCF) feature on Windows Server.

The certificate must be imported as follows:

- **For single-server farms** You must import the certificate directly on the server that runs Office Online Server. Don't bind the certificate manually. The New-OfficeWebAppsFarm cmdlet you run later will do this for you. If you bind the certificate manually, it'll be deleted every time the server restarts.
- **For load-balanced farms** If you're offloading SSL, the certificate must be imported on the hardware load balancer. If you're not offloading SSL, you'll need to install the certificate on each server in the Office Online Server farm.

> **NOTE**
> Don't use self-signed certificates except in non-critical test environments.

Using SSL offloading for hardware load balancers

When you set up a new Office Online Server farm, SSL offloading is set to Off by default. If you're using a hardware load balancer, we recommend you set SSL offloading to On so that each Office Online Server in the farm can communicate with the load balancer by using HTTP. Setting SSL offloading to On also provides the following advantages:

- Simplified certificates management
- Improved soft affinity
- Improved performance

> **NOTE**
> When you use HTTP, traffic from the load balancer to the servers that run Office Online Server isn't encrypted, so you need to make sure the network itself is secure. Use of a private subnet can help protect traffic.

Restrict which servers can join an Office Online Server farm based on OU membership

You can prevent unauthorized servers from joining an Office Online Server farm by creating an organizational unit for those servers and then specifying the FarmOU parameter when you create the farm. For more information about the FarmOU parameter, see New-OfficeWebAppsFarm.

Limit host access for Office Online Server by using the Allow List

The Allow List is a security feature that prevents unwanted hosts from connecting to an Office Online Server farm and using it for file operations without your consent. By adding the domains that contain approved hosts to the Allow List, you can limit the hosts to which Office Online Server allows file operations requests, such as file retrieval, metadata retrieval, and file changes.

You can add domains to the Allow List after you've created the Office Online Server farm. To learn how to add domains to the Allow List, see New-OfficeWebAppsHost.

IMPORTANT

If you do not add domains to the Allow List, Office Online Server allows file requests to hosts in any domain. Don't leave this list blank if your Office Online Server farm can be accessed from the Internet. Otherwise, anyone can use your Office Online Server farm to view and edit content.

Planning for Online Viewers with Office Online Server

By default, Online Viewers functionality is enabled after you install Office Online Server. Review the following guidelines if you're planning to use Online Viewers in your organization. In some cases, you might want to disable some features within Online Viewers. These guidelines refer to parameters that are set by using the Microsoft PowerShell cmdlets New-OfficeWebAppsFarm and Set-OfficeWebAppsFarm.

Security considerations for Online Viewers

Files that are intended to be viewed through a web browser by using Online Viewers must not require authentication. In other words, the files must be available publicly because Online Viewers can't perform authentication when it is retrieving files. We strongly recommend that the Office Online Server farm that you use for Online Viewers is only able to access either the intranet or the Internet, but not both. This is because Office Online Server doesn't differentiate between requests for intranet and Internet URLs. Somebody on the Internet could request an intranet URL, for example, causing a security leak if an internal document is viewed.

For the same reason, if you have set up the Office Online Server to connect only to the Internet, we strongly recommend that you disable UNC support in Online Viewers. To disable UNC support, set the OpenFromUncEnabled parameter to False by using the Microsoft PowerShell cmdlets New-OfficeWebAppsFarm (for new farms) or Set-OfficeWebAppsFarm (for existing farms).

As an additional security precaution, Online Viewers are limited to viewing Office files that are 10 MB or less.

Configuration options for Online Viewers

You can configure Online Viewers by using the following Microsoft PowerShell parameters in New-OfficeWebAppsFarm (for new farms) or Set-OfficeWebAppsFarm (for existing farms).

- **OpenFromUrlEnabled** Turns the Online Viewers on or off. This parameter controls Online Viewers for files that have URL and UNC paths. By default, this parameter is set to False (disabled) when you create a new Office Online Server farm.
- **OpenFromUncEnabled** When Online Viewers are turned on (set to True by using OpenFromUrlEnabled), this parameter turns on or off the ability for Online Viewers to display files in UNC paths. By default, this parameter is set to True, but make sure OpenFromUrlEnabled is also set to True before you enable opening files from UNC paths. As described earlier, we recommend you set this parameter to False if you have set up Office Online Server to connect to the Internet.
- **OpenFromUrlThrottlingEnabled** Throttles the number of "open from URL" requests from any given server in a time period. The default throttling values, which are not configurable, make sure that an Office Online Server farm does not overwhelm a single server by sending requests for content to be viewed in the Online Viewers.

Planning updates for Office Online Server

Before deploying Office Online Server, you need to decide how your organization will manage software updates to your Office Online Server farm. Although software updates help improve server security, performance, and reliability, installing updates incorrectly can cause issues with the Office Online Server.

Applying Office Online Server updates by using the Microsoft automatic updates process isn't supported with

Office Online Server. Updates to an Office Online Server must be applied in a specific way, as described in Apply software updates to Office Online Server. If Office Online Server updates are applied automatically, users might be unable to view or edit documents in Office Online. If this happens, you have to rebuild your Office Online Server farm.

We recommend that you manage updates by using Windows Server Update Services (WSUS) or by using Microsoft Endpoint Configuration Manager, which uses WSUS. WSUS allows you to fully manage the distribution of updates that are released through Microsoft Update for each server in the Office Online Server farm. By using WSUS, you can decide which updates can be automatically applied to the server farm and which updates, such as Office Online Server updates, have to be manually applied. For more information about WSUS, see Windows Server Update Services.

If you do not use WSUS or Microsoft Endpoint Configuration Manager, set Microsoft automatic updates on each server in the Office Online Server farm to **Automatically download but notify user for install**. When you're notified of an Office Online Server update, follow the steps in Apply software updates to Office Online Server. To have Windows updates applied and keep your servers secure, accept the Windows updates when you're notified that updates are available.

ULS Logs Changes from 2018 Update

The 2018 update of Office Online Server will see a few changes on the format of ULS logs, detailed below:

COLUMN	CHANGES
TimestampUtc	• **Date format changed** for MM/dd/yyyy to yyyy-MM-dd to make sorting more natural • Time is now always written in **UTC** • Column name changed from Timestamp to TimestampUtc • Timestamps no longer have a trailing ' ' or '*' indicating continuations (see Message column below)
Process	• Process name is no longer truncated to 32 characters • ProxyTraceTag no longer appends the process ID that sent the proxied trace • IIS W3WP processes get the AppPool ID appended. Ex: w3wp.exe#StatusViewer-status-MSOSP80 (0x631C)
ThreadId	• Column name changed from TID to ThreadId
Area	• Area is no longer truncated to 32 characters
Category	• Category is no longer truncated to 32 characters
EventId	• Column name changed from EventID to EventId
Level	(no changes)

COLUMN	CHANGES
Message	• Message length has been expanded from 800 to 31000 characters in size • **Messages over 31000 characters are truncated**, not continued in a second message • Since messages don't continue, there are no '...'s
Correlation	• **Correlations use a new stack** that pushes/pops/peeks appropriately • Correlation Stack no longer has a max depth of 32 • Correlations can follow Tasks across threads, and cross AppDomain boundaries

See also

Office Online Server overview

Deploy Office Online Server

Deploy Office Online Server

9/20/2021 • 11 minutes to read

Summary: Explains how to deploy Office Online Server on-premises for use by SharePoint Server, Skype for Business Server, and Exchange Server.

Audience: IT Professionals

Office Online Server is the next version of Office Web Apps Server. Deploying Office Online Server involves installing some prerequisite software and running a few Microsoft PowerShell commands, but overall the process is designed to be pretty straightforward. This article walks you through the procedures to get your servers ready, then gives you the Microsoft PowerShell commands to configure the on-premises Office Online Server farm.

Prepare servers to run Office Online Server

Perform these procedures on all servers that will run Office Online Server. This server must be Windows Server 2012 R2 or Windows Server 2016.

> **IMPORTANT**
> Windows Server 2016 requires Office Online Server April 2017 or later.

Office Online Server was designed and tested for server operating systems configured with default settings. If you need to deploy with non-default settings, it is recommended to begin installation and setup with the default settings. Once the system is verified as working, then incrementally add and test Group Policies, security settings and other modifications.

Step 1: Install prerequisite software for Office Online Server

To install Office Online Server

1. Open the Microsoft PowerShell prompt as an administrator and run this command to install the required roles and services.

   Windows Server 2012 R2:

```
Add-WindowsFeature Web-Server,Web-Mgmt-Tools,Web-Mgmt-Console,Web-WebServer,Web-Common-Http,Web-Default-Doc,Web-Static-Content,Web-Performance,Web-Stat-Compression,Web-Dyn-Compression,Web-Security,Web-Filtering,Web-Windows-Auth,Web-App-Dev,Web-Net-Ext45,Web-Asp-Net45,Web-ISAPI-Ext,Web-ISAPI-Filter,Web-Includes,InkandHandwritingServices,NET-Framework-Features,NET-Framework-Core,NET-HTTP-Activation,NET-Non-HTTP-Activ,NET-WCF-HTTP-Activation45,Windows-Identity-Foundation,Server-Media-Foundation
```

Windows Server 2016:

```
Add-WindowsFeature Web-Server,Web-Mgmt-Tools,Web-Mgmt-Console,Web-WebServer,Web-Common-Http,Web-Default-Doc,Web-Static-Content,Web-Performance,Web-Stat-Compression,Web-Dyn-Compression,Web-Security,Web-Filtering,Web-Windows-Auth,Web-App-Dev,Web-Net-Ext45,Web-Asp-Net45,Web-ISAPI-Ext,Web-ISAPI-Filter,Web-Includes,NET-Framework-Features,NET-Framework-45-Features,NET-Framework-Core,NET-Framework-45-Core,NET-HTTP-Activation,NET-Non-HTTP-Activ,NET-WCF-HTTP-Activation45,Windows-Identity-Foundation,Server-Media-Foundation
```

If prompted, restart the server.

2. Install the following software:

- .NET Framework 4.5.2
- Visual C++ Redistributable Packages for Visual Studio 2013
- Visual C++ Redistributable for Visual Studio 2015
- Microsoft.IdentityModel.Extention.dll

Step 2: Install Office Online Server

Next, we'll install Office Online Server.

If you plan to use any Excel Online features that utilize external data access (such as Data Models, Power Pivot, or Power View), note that Office Online Server must reside in the same Active Directory forest as its users as well as any external data sources that you plan to access using Windows-based authentication.

Complete these steps on any servers that will run Office Online Server.

To install Office Online Server

1. Download Office Online Server from the Volume Licensing Service Center (VLSC). Office Online Server is a component of Office, so it is available under each of the product pages including Office Standard 2016, Office Professional Plus 2016, and Office 2015 for Mac Standard.
2. Run Setup.exe.
3. On the **Read the Microsoft Software License Terms** page, select **I accept the terms of this agreement** and click **Continue**.
4. On the **Choose a file location** page, select the folder where you want the Office Online Server files to be installed (for example, C:\Program Files\Microsoft Office Web Apps) and select **Install Now**. If the folder you specified doesn't exist, Setup creates it for you.

   We recommend that you install Office Online Server on the system drive.
5. When Setup finishes installing Office Online Server, choose **Close**.
6. If you're planning to use Kerberos Constrained Delegation with Excel Online, then, in **Services**, set the **Claims to Windows Token Service** to start automatically on this server.

If you plan to use Kerberos Constrained Delegation with Excel Online, be sure to add each server in the Office Online Server farm to the Active Directory Domain Services delegation list.

Step 3: Install language packs for Office Online Server

Office Online Server Language Packs let users view web-based Office files in multiple languages, whether they're opened from SharePoint document libraries or Outlook on the web.

To install the language packs, follow these steps.

1. Download the Office Online Server Language Packs from the Microsoft Download Center.

2. Run **wacserverlanguagepack.exe**.
3. In the Office Online Server Language Pack Wizard, on the **Read the Microsoft Software License Terms** page, select **I accept the terms of this agreement** and select **Continue**.
4. When Setup finishes installing Office Online Server, choose **Close**.

To **patch language packs**, deploy Office Online Updates after installing the Office Online Language Packs.

> **IMPORTANT**
> To install language packs after the Office Online Server farm is created, you must remove a server from the farm, install the language pack on it, and then add the server back to the farm.> For a language pack to work correctly, you'll need to install it on all servers in the farm.

Deploy the Office Online Server farm

Follow the procedures in one of the following three sections, based on what kind of Office Online Server farm you want to create.

> **TIP**
> If Microsoft PowerShell doesn't recognize the **New-OfficeWebAppsFarm** cmdlet when you run it, you may need to import the **OfficeWebApps** module. Use this command: `Import-Module -Name OfficeWebApps`

Deploy a single-server Office Online Server farm that uses HTTP

If you're only deploying Office Online Server for testing or internal use, and you don't need to provide Office Online Server functionality to Skype for Business Server 2015, this procedure is for you. Here, you'll install a single-server Office Online Server farm that uses HTTP. You won't need a certificate or load balancer, but you will need a dedicated physical server or virtual machine instance that isn't running any other server application.

You can use this Office Online Server farm to provide Office Online functionality to SharePoint Server 2016 and Exchange Server 2016.

> **NOTE**
> It is strongly recommended to use HTTPS (TLS) regardless of environment as Office Online Server uses OAuth tokens to communicate with external services, such as SharePoint or Exchange Server. OAuth tokens contain information that can potentially be intercepted and replayed by an attacker, granting the attacker the same rights as the user making the request to Office Online Server.

Step 1: Create the Office Online Server farm

Use the **New-OfficeWebAppsFarm** command to create a new Office Online Server farm that consists of a single server, as shown in the following example.

```
New-OfficeWebAppsFarm -InternalURL "http://servername" -AllowHttp -EditingEnabled
```

Parameters

- **-InternalURL** is the name of the server that runs Office Online Server, such as **http://servername**.
- **-AllowHttp** configures the farm to use HTTP.

- **-EditingEnabled** enables editing in Office Online when used with SharePoint Server. This parameter isn't used by Skype for Business Server 2015 or Exchange Server because those hosts don't support editing.

Step 2: Verify that the Office Online Server farm was created successfully

After the farm is created, details about the farm are displayed in the Microsoft PowerShell prompt. To verify that Office Online Server is installed and configured correctly, use a web browser to access the Office Online Server discovery URL, as shown in the following example. The discovery URL is the *InternalUrl* parameter you specified when you configured your Office Online Server farm, followed by **/hosting/discovery**, for example:

```
http://servername/hosting/discovery
```

If Office Online Server is working as expected, you should see a Web Application Open Platform Interface Protocol (WOPI)-discovery XML file in your web browser. The first few lines of that file should resemble the following example.

```
<?xml version="1.0" encoding="utf-8" ?>
- <wopi-discovery>
- <net-zone name="internal-http">
- <app name="Excel" favIconUrl="http://servername/x/_layouts/images/FavIcon_Excel.ico" checkLicense="true">
<action name="view" ext="ods" default="true" urlsrc="http://servername/x/_layouts/xlviewerinternal.aspx?
<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>" />
<action name="view" ext="xls" default="true" urlsrc="http://servername/x/_layouts/xlviewerinternal.aspx?
<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>" />
<action name="view" ext="xlsb" default="true" urlsrc="http://servername/x/_layouts/xlviewerinternal.aspx?
<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>" />
<action name="view" ext="xlsm" default="true" urlsrc="http://servername/x/_layouts/xlviewerinternal.aspx?
<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>" />
```

Step 3: Configure Secure Store access (optional)

If you're planning to use the Secure Store service in SharePoint Server in an HTTP environment, there's a parameter that you need to set to enable this. (If you're not planning to use Secure Store in SharePoint Server with Excel Online, you can skip this step.)

When Office Online Server attempts to refresh data in a workbook or ODC file that is stored in an HTTP path, that data refresh will fail if you have not configured Office Online Server to allow Secure Store connections over HTTP.

Use the Set-OfficeWebAppsFarm cmdlet to configure the Secure Store over HTTP settings:

```
Set-OfficeWebAppsFarm -AllowHttpSecureStoreConnections:$true
```

Keep in mind that the contents of the workbook or ODC file will be transmitted in clear text over HTTP. Data connected workbooks and ODC files contain database connection information, and can contain passwords.

Step 4: Configure the host

The farm is now ready to provide Office Online functionality to hosts over HTTP. Visit the following articles for more information about how to configure hosts.

- Configure Office Online Server for SharePoint Server 2016

> **NOTE**
>
> This also applies to SharePoint Server 2019.

- Office Online Server integration with Exchange

Deploy a single-server Office Online Server farm that uses HTTPS

For most production environments, we strongly recommend the use of HTTPS for its security features. Also, HTTPS is required if you want to provide Office Online Server functionality to Skype for Business Server 2015, which lets users view PowerPoint broadcasts in a browser. Here's how to install a single-server Office Online Server farm that uses HTTPS. You'll need to install a certificate on the server.

This Office Online Server farm will provide Office Online functionality to SharePoint Server, Skype for Business Server 2015, and Exchange Server 2016.

Step 1: Create the Office Online Server farm

Use the **New-OfficeWebAppsFarm** command to create a new Office Online Server farm that consists of a single server, as shown in the following example.

```
New-OfficeWebAppsFarm -InternalUrl "https://server.contoso.com" -ExternalUrl "https://wacweb01.contoso.com"
-CertificateName "OfficeWebApps Certificate" -EditingEnabled
```

Parameters

- **-InternalURL** is the fully qualified domain name (FQDN) of the server that runs Office Online Server, such as **http://servername.contoso.com**.
- **-ExternalURL** is the FQDN that can be accessed on the Internet.
- **-CertificateName** is the friendly name of the certificate.
- **-EditingEnabled** is optional and enables editing in Office Online when used with SharePoint Server. This parameter isn't used by Skype for Business Server 2015 or Exchange Server because those hosts don't support editing.

Step 2: Verify that the Office Online Server farm was created successfully

After the farm is created, details about the farm are displayed in the Microsoft PowerShell prompt. To verify that Office Online Server is installed and configured correctly, use a web browser to access the Office Online Server discovery URL, as shown in the following example. The discovery URL is the *InternalUrl* parameter you specified when you configured your Office Online Server farm, followed by **/hosting/discovery**, for example:

```
https://server.contoso.com/hosting/discovery
```

If Office Online Server works as expected, you should see a Web Application Open Platform Interface Protocol (WOPI)-discovery XML file in your web browser. The first few lines of that file should resemble the following example:

```
<?xml version="1.0" encoding="UTF-8"?>
<wopi-discovery><net-zone
name="internal-https"><app name="Excel" checkLicense="true"
favIconUrl="https://wac.contoso.com/x/_layouts/images/FavIcon_Excel.ico"><action
name="view"
urlsrc="https://wac.contoso.com/x/_layouts/xlviewerinternal.aspx?<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>"
default="true" ext="ods"/><action name="view"
urlsrc="https://wac.contoso.com/x/_layouts/xlviewerinternal.aspx?<ui=UI_LLCC&amp;><rs=DC_LLCC&amp;>"
default="true" ext="xls"/><action name="view"
```

> **NOTE**
> Depending on the security settings of your web browser, you might see a message that prompts you to select **Show all content** before the contents of the discovery XML file are displayed.

Step 3: Configure the host

The farm is now ready to provide Office Online functionality to hosts over HTTPS. Visit the following articles for more information about how to configure hosts.

- Configure Office Online Server for SharePoint Server 2016

> **NOTE**
> This also applies to SharePoint Server 2019.

- Office Online Server integration with Exchange

Deploy a multi-server, load-balanced Office Online Server farm that uses HTTPS

If you anticipate lots of traffic to your Office Online Server farm, and you want it to be available over the Internet as well as on your internal network, this type of topology is the way to go. This section shows how to install a multi-server Office Online Server farm that uses a load balancer and HTTPS.

Before you begin, make sure your load balancer is configured. Also, you'll need to install a certificate on the load balancer. This Office Online Server farm will provide Office Online functionality to SharePoint Server, Skype for Business Server 2015, and Exchange Server 2016.

Step 1: Create the Office Online Server farm on the first server

Use the **New-OfficeWebAppsFarm** command to create a new Office Online Server farm on the first server, as shown in the following example.

```
New-OfficeWebAppsFarm -InternalUrl "https://server.contoso.com" -ExternalUrl "https://wacweb01.contoso.com"
-SSLOffloaded -EditingEnabled
```

Parameters

- **-InternalURL** is the fully qualified domain name (FQDN) of the server that runs Office Online Server, such as **http://servername.contoso.com**.
- **-ExternalURL** is the FQDN name that can be accessed on the Internet.
- **-SSLOffloaded** enables offloading SSL termination to the load balancer.
- **-EditingEnabled** is optional and enables editing in Office Online when used with SharePoint Server. This parameter isn't used by Skype for Business Server 2015 or Exchange Server because those hosts don't support editing.

Step 2: Add more servers to the farm

After the first server is running Office Online Server, run the **New-OfficeWebAppsMachine** command on each server you want to add to the Office Online Server farm. For the **-MachineToJoin** parameter, use the computer name of a server that's already in the Office Online Server farm. For example, if server1.contoso.com is already in the farm, use the following:

```
New-OfficeWebAppsMachine -MachineToJoin "server1.contoso.com"
```

Step 3: Verify that the Office Online Server farm was created successfully

After the farm is created, details about the farm are displayed in the Microsoft PowerShell prompt. To verify that Office Online Server is installed and configured correctly, use a web browser to access the Office Online Server discovery URL, as shown in the following example. The discovery URL is the *InternalUrl* parameter you specified when you configured your Office Online Server farm, followed by **/hosting/discovery**. For example:

```
https://server.contoso.com/hosting/discovery
```

If Office Online Server works as expected, you should see a Web Application Open Platform Interface Protocol (WOPI)-discovery XML file in your web browser. The first few lines of that file should resemble the following example:

```
<?xml version="1.0" encoding="UTF-8"?>
<wopi-discovery><net-zone name="internal-https"><app name="Excel" checkLicense="true"
favIconUrl="https://officewebapps.contoso.com/x/_layouts/images/FavIcon_Excel.ico"><action name="view"
urlsrc="https://officewebapps.contoso.com/x/_layouts/xlviewerinternal.aspx?<ui=UI_LLCC&amp;>
<rs=DC_LLCC&amp;>" default="true" ext="ods"/><action name="view"
urlsrc="https://officewebapps.contoso.com/x/_layouts/xlviewerinternal.aspx?<ui=UI_LLCC&amp;>
<rs=DC_LLCC&amp;>" default="true" ext="xls"/><action name="view"
urlsrc="https://officewebapps.contoso.com/x/_layouts/xlviewerinternal.aspx?<ui=UI_LLCC&amp;>
<rs=DC_LLCC&amp;>" default="true" ext="xlsb"/>
```

> **NOTE**
>
> Depending on the security settings of your web browser, you might see a message that prompts you to select **Show all content** before the contents of the discovery XML file are displayed.

Step 4: Configure the host

The farm is now ready to provide Office Online functionality to hosts over HTTPS. Visit the following articles for more information about how to configure hosts.

- Configure Office Online Server for SharePoint Server 2016

> **NOTE**
>
> This also applies to SharePoint Server 2019.

- Office Online Server integration with Exchange

If you see "500 Web Service Exceptions" or "500.21 - Internal Server Error" messages

If features of the .NET Framework 4.6 were installed and then removed, you might see "500 Web Service Exceptions" or "500.21 - Internal Server Error" messages when you run OfficeWebApps cmdlets. To fix this, run the following sample commands from an elevated command prompt to clean up settings that could prevent Office Online Server from functioning correctly:

For Windows Server 2012 R2 or Windows Server 2016

```
Add-WindowsFeature NET-Framework-45-Core, NET-Framework-45-ASPNET, Web-Asp-Net45
```

See also

Apply software updates to Office Online Server

Office Online Server release schedule

Plan Office Online Server

Configure Office Online Server for SharePoint Server

3/24/2021 • 7 minutes to read

Summary: Learn how to configure SharePoint Server to use Office Online Server (the next version of Office Web Apps Server).

Audience: IT Professionals

This article picks up where Deploy Office Online Server left off. In that article, you set up the server that runs Office Online Server on-premises. In this one, you'll configure SharePoint Server to use Office Online Server. First, you'll need to run a few Microsoft PowerShell cmdlets from SharePoint Server 2016, after which users will be able to open Office files from SharePoint Server document libraries in a browser.

Before you configure SharePoint Server to use Office Online Server

A few things to check before getting started:

- These instructions also apply to SharePoint Server 2013, however SharePoint Server 2013 cannot use the Excel Online external data connectivity and data refresh functionality in Office Online Server.
- Install SharePoint Server 2016. See Install SharePoint Server for guidance.
- Make sure all SharePoint Server 2016 web applications use claims-based authentication. Office Online rendering and editing won't work on SharePoint Server 2016 web applications that use classic mode authentication.
- To enable users to edit (not just read) Office documents in a web browser, you'll need an editing license. Also, you'll need to enable editing on the Office Online Server farm.
- If you log on to SharePoint Server 2016 by using the System Account, you won't be able to test the connection between SharePoint Server 2016 and Office Online Server. Log on with a different account to test the connection.
- Low memory conditions can cause Office document previews to fail in Office Online.
- SharePoint Server 2013 cannot use the Excel Online external data connectivity and data refresh functionality in Office Online Server. That's only available with SharePoint Server 2016.
- Office Online Server uses OAuth tokens to communicate with SharePoint Server. These tokens can potentially be intercepted and replayed, providing an attacker with the same rights as the user making the request from SharePoint Server to Office Online Server. It is strongly recommended that you configure Office Online Server to use HTTPS (TLS) only.

Configure SharePoint Server to use Office Online Server

Choose one of the following sections depending on whether you want to use HTTP or HTTPS. HTTP is generally recommended only for test environments. In production environments, the more secure HTTPS protocol is the better choice.

In a test environment that uses HTTP

For this configuration, make sure you have set up Office Online Server by following the steps in Deploy a single-server Office Online Server farm that uses HTTP. Be sure to configure the Office Online Server farm to use an internal URL and HTTP.

Step 1: Create the binding between SharePoint 2016 and Office Web Apps Server

To get started, open an elevated SharePoint 2016 Management Shell. (Right-click **SharePoint 2016 Management Shell**, and then click **Run as Administrator**.)

Run the following command, where is the fully qualified domain name (FQDN) of the URL that you set for the internal URL. This is the point of entry for Office Online Server traffic. For this test environment, you need to specify the -AllowHTTP parameter to allow SharePoint Server 2016 to receive discovery information from the Office Online Server farm by using HTTP. If you don't specify -AllowHTTP, SharePoint Server 2016 will try to use HTTPS to communicate with the Office Online Server farm, and this command won't work.

```
New-SPWOPIBinding -ServerName <WacServerName> -AllowHTTP
```

After running this command, you should see a list of bindings displayed at the Microsoft PowerShell command prompt.

Step 2: View the WOPI zones for the SharePoint bindings

Office Online Server uses zones to determine which URL (internal or external) and which protocol (HTTP or HTTPS) to use when it communicates with the host, in this case, SharePoint Server 2016. By default, SharePoint Server 2016 uses the **internal-https** zone. Run the following command to see what your current zone is.

```
Get-SPWOPIZone
```

The WOPI zone displayed by this command should be **internal-http**. If it's displayed correctly, skip to step 4. If it isn't, see the next step.

Step 3: Change the WOPI zone to internal-http

If the result from Step 3 was **internal-https**, run the following command to change the zone to **internal-http**. You need to make this change because the zone of SharePoint Server 2016 must match the zone of the Office Online Server farm.

```
Set-SPWOPIZone -zone "internal-http"
```

Verify that the new zone is **internal-http** by running **Get-SPWOPIZone** again.

Step 4: Change the AllowOAuthOverHttp setting in SharePoint 2016 to True

To use Office Online with SharePoint Server 2016 over HTTP in a test environment, you need to set AllowOAuthOverHttp to **True**. Otherwise Office Online won't work. You can check the current status by running the following example.

```
(Get-SPSecurityTokenServiceConfig).AllowOAuthOverHttp
```

If this command returns **False**, run the following commands to set this to **True**.

```
$config = (Get-SPSecurityTokenServiceConfig)
```

```
$config.AllowOAuthOverHttp = $true
```

```
$config.Update()
```

Run the following command again to verify that the AllowOAuthOverHttp setting is now set to **True**.

```
(Get-SPSecurityTokenServiceConfig).AllowOAuthOverHttp
```

Step 5: Enable the Excel SOAP API

The Excel SOAP API is needed for scheduled data refresh with Excel Online, and for Excel Web Part rendering. To enable the Excel SOAP API, you need to add the WopiLegacySoapSupport property to the SharePoint Server farm properties using by PowerShell. The input parameter is the URL to ExcelServiceInternal.asmx. This URL can address multiple OOS servers via load balancing. Simply replace the with your Office Online Server path.

To enable the Excel SOAP API, run the following PowerShell where is the URL of your Office Online Server farm. (For example, http://OfficeOnlineServer.contoso.com.)

```
$Farm = Get-SPFarm
$Farm.Properties.Add("WopiLegacySoapSupport", "<URL>/x/_vti_bin/ExcelServiceInternal.asmx");
$Farm.Update();
```

Step 6: Verify that Office Web Apps is working

In SharePoint Server 2016, make sure you're not logged on as System Account because you won't be able to edit or view the documents with Office Online. Go to a SharePoint Server 2016 document library that contains Office documents and view a Word, PowerPoint, Excel, or OneNote file. The document should open in a browser that displays the file by using Office Online.

In a production environment that uses HTTPS

Before you start the following procedures, make sure that you have set up Office Online Server by following the steps in Deploy a single-server Office Online Server farm that uses HTTPS or Deploy a multi-server, load-balanced Office Online Server farm that uses HTTPS.

Step 1: Create the binding between SharePoint 2016 and Office Online Server

To get started, open an elevated SharePoint 2016 Management Shell. (Right-click **SharePoint 2016 Management Shell**, and then click **Run as Administrator**.)

Run the following command, where is the fully qualified domain name (FQDN) of the URL that you set for the internal URL. This is the point of entry for Office Online Server traffic.

```
New-SPWOPIBinding -ServerName <WacServerName>
```

Step 2: View the WOPI zone of SharePoint 2016

Office Online Server uses zones to determine which URL (internal or external) and which protocol (HTTP or HTTPS) to use when it communicates with the host, which in this case is SharePoint Server 2016. By default, SharePoint Server 2016 uses the **internal-https** zone. Verify that this is the current zone by running the following command.

```
Get-SPWOPIZone
```

Take note of the WOPI zone that is displayed.

Step 3: Change the WOPI zone if necessary

Depending on your environment, you might have to change the WOPI zone. If you have a SharePoint farm that's both internal and external, specify external. If you have a SharePoint farm that's internal only, specify internal.

If the results from Step 2 show that **internal-https** and the SharePoint farm is internal only, you can skip this step. If you have a SharePoint farm that's internal and external, you need to run the following command to change the zone to **external-https**.

```
Set-SPWOPIZone -zone "external-https"
```

Step 4: Enable the Excel SOAP API

The Excel SOAP API is needed for scheduled data refresh with Excel Online, and for Excel Web Part rendering. To enable the Excel SOAP API, you need to add the WopiLegacySoapSupport property to the SharePoint Server farm properties using by PowerShell. The input parameter is the URL to ExcelServiceInternal.asmx. This URL can address multiple OOS servers via load balancing. Simply replace the with your Office Online Server path.

To enable the Excel SOAP API, run the following PowerShell where is the URL of your Office Online Server farm. (For example, https://OfficeOnlineServer.contoso.com.)

```
$Farm = Get-SPFarm
$Farm.Properties.Add("WopiLegacySoapSupport", "<URL>/x/_vti_bin/ExcelServiceInternal.asmx");
$Farm.Update();
```

Step 5: Verify that Office Web Apps is working

In SharePoint Server 2016, make sure you aren't logged on as System Account because you won't be able to edit or view the documents with Office Online. Go to a SharePoint Server 2016 document library that contains Office documents and view a Word, PowerPoint, Excel, or OneNote file. The document should open in a browser that displays the file by using Office Online.

Disconnect SharePoint Server 2016 from Office Online Server

If, for any reason, you want to disconnect SharePoint Server 2016 from Office Online Server, use the following command example.

```
Remove-SPWOPIBinding -All:$true
```

Configure server-to-server authentication between Office Online Server and SharePoint Server 2016

3/24/2021 • 5 minutes to read

Summary: Configure server-to-server authentication between Office Online Server and SharePoint 2016.

Server-to-server authentication between Office Online Server and SharePoint Server 2016 establishes trust between the two servers. This trust is a necessary prerequisite for some Excel Online features, such as Office Data Connection (ODC) file support and the IT Management Dashboard, which is part of SQL Server Power Pivot for SharePoint. This article guides you through the steps to set up this trust.

To configure server-to-server authentication, your Office Online Server farm and SharePoint Server farm must be in the same Active Directory forest. You also must have a User Profile service application configured on the SharePoint Server farm.

The basic actions required for configuring server-to-server authentication are:

1. Import the certificate to Office Online Server
2. Export the certificate for use on SharePoint Server
3. Configure Office Online Server to use the certificate for server-to-server authentication
4. Configure SharePoint Server to use the certificate for server-to-server authentication

You'll start with importing the certificate to Office Online Server.

Import the certificate to Office Online Server

The first step is to import your certificate for use by Office Online Server. Follow the *Import your certificate* and *Grant network service permission to use the key* procedures on each server in your Office Online Server farm.

Import your certificate

You can use either a private key SSL certificate or a self-signed certificate. We highly recommend using a private key SSL certificate. The following sections provide procedures for both. Choose the one the you are using.

Use a private key SSL certificate

Install the certificate on each server running Office Online Server.

To install the certificate on Office Online Server

1. On the server running Office Online Server, open IIS Manager.
2. In the left pane, click the server name.
3. Double-click **Server Certificates**.
4. In the **Actions** pane, click **Import**.
5. Type the *path* and *file name* of the SSL certificate that you want to use.
6. In the **Password** box, type the *password* for the certificate.
7. In the **Select Certificate Store** drop-down list, make sure **Personal** is selected.
8. Click **OK**.

Repeat this procedure on each server running Office Online.

Use a self-signed certificate

If you're using a self-signed certificate, you need to add it to the Trusted Root Certification Authorities.

To import the certificate to the Trusted Root Certification Authorities

1. Open the Microsoft Management Console.
2. On the **File** menu, choose **Add/Remove Snap-in**.
3. Choose **Certificates**, and then click **Add**.
4. Choose the **Computer account** option, click **Next**, and then click **Finish**.
5. Click **OK**.
6. Expand **Certificates (Local Computer)**, right-click **Trusted Root Certification Authorities**, click **All Tasks**, and then click **Import**.
7. Click **Next**.
8. Browse to the location of the certificate, select it, and then click **Next**.
9. Type the certificate password, click **Next**, and then click **Finish**.

Keep the Microsoft Management Console open for the next procedure.

Grant network service permission to use the key

Next, use the Microsoft Management Console (MMC) to grant the network service permissions to use the private key.

To grant network service permissions to use the private key

1. Open the Microsoft Management Console.
2. On the **File** menu, choose **Add/Remove Snap-in**.
3. Choose **Certificates**, and then click **Add**.
4. Choose the **Computer account** option, click **Next**, and then click **Finish**.
5. Click **OK**.
6. Expand **Certificates (Local Computer)**, expand **Personal**, and then click **Certificates**.
7. Right-click the certificate that you just imported, click **All Tasks**, and then click **Manage Private Keys**.
8. In the **Permissions** dialog box, click **Add**.
9. Type Network Service, and then click **OK**.
10. Click **OK**.

Be sure you follow the *Import your certificate* and *Grant network service permission to use the key* procedures on each server in your Office Online Server farm.

Keep the Microsoft Management Console open for the next procedure.

Export the certificate for use on SharePoint Server

The next step is to export the certificate so that you can use it to register Office Online Server as a trusted token issuer.

To export the certificate for use with SharePoint Server 2016

1. Right-click the certificate that you just imported, click **All Tasks**, and then click **Export**.
2. On the Welcome page, click **Next**.
3. Choose the **No, do not export the private key** option, and then click **Next**.
4. Choose the **DER encoded binary X.509 (.CER)** option, and then click **Next**.
5. Type the *path* and *name* of the file that you want to export, and then click **Next**.
6. Click **Finish**, and then click **OK**.

Copy the certificate file that you created to a location where you can access it from SharePoint Server.

Next, you need to specify this certificate as the S2S certificate for Office Online Server.

Configure Office Online Server to use the certificate for server-to-server authentication

To specify the S2S certificate for Office Online Server

1. Open a Microsoft PowerShell window as Administrator.
2. Type the following where is the friendly name of the certificate that you're using.

```
Set-OfficeWebAppsFarm -S2SCertificateName "<friendlyName>" -Confirm:$false -Force
```

Using HTTP with Office Online Server

If you're using HTTP instead of HTTPS for your Office Online Server farm, you have to allow outbound HTTP connections from Office Online Server. (If you're using SSL, you can skip this procedure.)

To enable outbound HTTP connections from Office Online Server, run the following PowerShell command:

```
Set-OfficeWebAppsFarm -AllowOutboundHttp:$True
iisreset
```

> **IMPORTANT**
> It is strongly recommended to use HTTPS (TLS) regardless of environment as Office Online Server uses OAuth tokens to communicate with external services, such as SharePoint or Exchange Server. OAuth tokens contain information that can potentially be intercepted and replayed by an attacker, granting the attacker the same rights as the user making the request to Office Online Server.

Using HTTP paths with ODC files

If you plan to store ODC files in an HTTP path, you have to configure Office Online Server to allow Secure Store connections over HTTP.

> **IMPORTANT**
> When you use Secure Store connections over HTTP, the contents of the ODC file are passed in clear text. ODC files contain database connection information and can contain passwords. Microsoft recomments using HTTPS.

To enable Office Online Server to use HTTP paths with Secure Store, run the following PowerShell command:

```
Set-OfficeWebAppsFarm -AllowHttpSecureStoreConnections:$true
iisreset
```

Configure SharePoint Server to use the certificate for server-to-server authentication

You need to register SharePoint Server and SQL Server as trusted token issuers. This is done through PowerShell. Here are the parameters that you'll use:

- - The URL of your top-level site collection.
- - The name of the certificate issuer. You can find this by viewing the **Details** tab of the certificate in IIS Manager.

- - The path and file name of the certificate file that you exported.
- - The GUID of the trusted token issuer. SharePoint Server is 67e3df25-268a-4324-a550-0de1c7f97287@bd2372e4-0a11-495c-9541-8377c6def195 and SQL Server is 67e3df25-268a-4324-a550-0de1c7f97287@ffab2d74-c6ae-4375-819a-8555d49b699a

Perform the following procedure twice - once for each GUID.

To register a trusted token issuer

1. Open the SharePoint 2016 Management Shell as Administrator.
2. Run the following script using the parameters noted above:

```
$issuer = New-SPTrustedSecurityTokenIssuer -Name <CertificateIssuer> -Certificate <X509Certificate> -
RegisteredIssuerName <RegisteredIssuer>
$app = Get-SPAppPrincipal -Site <SPSiteURL> -NameIdentifier $issuer.NameId
$site = Get-SPSite <SPSiteURL>
Set-SPAppPrincipalPermission -appPrincipal $app -Site $site.RootWeb -Scope SiteSubscription -Right
FullControl -EnableAppOnlyPolicy
```

3. If you're using a self-signed certificate, run the following command:

```
New-SPTrustedRootAuthority -Name <CertificateIssuer> -Certificate <X509Certificate>
```

See Also

New-SPTrustedSecurityTokenIssuer

New-SPTrustedRootAuthority

Get-SPAppPrincipal

Set-SPAppPrincipalPermission

Configure Excel Online administrative settings

3/24/2021 • 4 minutes to read

Summary: Configure administrative settings for Excel workbooks rendered in Excel Online.

There are several settings that you can use to customize Excel Online. These settings help you adjust the resource usage of your Office Online Server farm and enforce some of your organization's governance policies.

In Office Online Server, most of these settings are available as parameters for the New-OfficeWebAppsFarm and Set-OfficeWebAppsFarm Microsoft PowerShell cmdlets, and there is an additional cmdlet (OfficeWebAppsExcelBIServer) that configures access to SQL Server Analysis Services (SSAS) servers. (Note that this is a subset of the settings that were available in Excel Services in SharePoint Server 2013.)

Here's what you can do:

- Prevent a workbook from loading in Excel Online if data refresh fails
- Set the Excel Online cache time for volatile functions
- Set the number of Excel Online data requests per session
- Set the Excel Online workbook calculation mode
- Set the maximum Excel Online image size
- Configure an Analysis Services (data model) server for Excel Online
- Configure Analysis Services EffectiveUserName in Excel Online

Prevent a workbook from loading in Excel Online if data refresh fails

By default, Excel Online doesn't load Excel files if an automatic data refresh operation fails when someone opens the file. This helps prevent users from seeing out-of-date information or possibly information that they should not have access to.

The load fails only in the following conditions:

- The user has read-only permissions for the file in SharePoint Server.
- There are data connections in the workbook file that are automatically refreshed when someone opens the workbook.

Syntax: Set-OfficeWebAppsFarm -ExcelAbortOnRefreshOnOpenFail

Default: True

Example:

```
Set-OfficeWebAppsFarm -ExcelAbortOnRefreshOnOpenFail:$false
```

Set the Excel Online cache time for volatile functions

You can specify the maximum time, in seconds, that a computed value for a volatile function is cached for

automatic recalculations. Valid values are:

- -1: Calculates once when the workbook loads.
- 0: Always calculates.
- 1 to 2073600: Caches 1 second to 24 days.

The value must be an integer from -1 to 2073600.

Syntax: Set-OfficeWebAppsFarm -ExcelAutomaticVolativeFunctionCacheLifetime

Default: 300

Example:

```
Set-OfficeWebAppsFarm -ExcelAutomaticVolativeFunctionCacheLifetime:500
```

Set the number of Excel Online data requests per session

You can specify the maximum number of concurrent external data requests allowed in each session. If a session must issue more than this number of requests, additional requests are queued. The scope of this setting is the logical server. The value must be a positive integer.

Syntax: Set-OfficeWebAppsFarm -ExcelConcurrentDataRequestsPerSessionMax

Default: 5

Example:

```
Set-OfficeWebAppsFarm -ExcelConcurrentDataRequestsPerSessionMax:10
```

Set the Excel Online workbook calculation mode

You can specify the calculation mode of workbooks rendered in Excel Online. The available values are: *File*, *Manual*, *Auto*, and *AutoDataTables* (automatic except data tables). Settings other than *File* override the workbook settings.

Syntax: Set-OfficeWebAppsFarm -ExcelDefaultWorkbookCalcMode

Default: File

Example:

```
Set-OfficeWebAppsFarm -ExcelDefaultWorkbookCalcMode:Auto
```

Set the maximum Excel Online image size

You can specify the maximum size, in megabytes, of a chart or image that can be opened by Excel Online. The value must be an integer greater than 0.

Syntax: Set-OfficeWebAppsFarm -ExcelChartAndImagesSizeMax

Default: 1

Example:

```
Set-OfficeWebAppsFarm -ExcelChartAndImagesSizeMax:5
```

Configure an Analysis Services (data model) server for Excel Online

You can configure Analysis Services servers to work with Excel Online by using the OfficeWebAppsExcelBIServer cmdlets:

- **New-OfficeWebAppsExcelBIServer** Adds an Analysis Services server location to the Allow List for Excel Calculation Services in Office Online Server for advanced BI functionality.
- **Get-OfficeWebAppsExcelBIServer** Gets the Analysis Services servers in the Allow List.
- **Remove-OfficeWebAppsExcelBIServer** Removes a server from the Allow List.

To use this feature, you must also configure each computer in your Office Online Server farm as an Analysis Services administrator.

The New and Remove cmdlets take a parameter of -ServerID, which is the server name of the Analysis Services server that you want to add or remove.

Examples:

```
New-OfficeWebAppsExcelBIServer -ServerID "SSAS01"
```

```
Remove-OfficeWebAppsExcelBIServer -ServerID "SSAS01"
```

The OfficeWebAppsExcelBIServer cmdlets also support common parameters.

Configure Analysis Services EffectiveUserName in Excel Online

EffectiveUserName is a SQL Server Analysis Services connection string property that contains the name of the user who is accessing a report. In Office Online Server, you can use this property in conjunction with Excel Online to pass the identity of the user who is viewing the report to Analysis Services. This allows per-user identity without the need to configure Kerberos constrained delegation.

To enable this option, you need to use the SQL Server 2016 version of SQL Server Management Studio. The actual data source can be in an earlier version of Analysis Services.

To configure this option, you have to do the following:

- Configure each computer in your Office Online Server farm as an Analysis Services administrator.
- Use PowerShell to enable EffectiveUserName in Excel Online (described below).

The Set-OfficeWebAppsFarm is used to enable or disable EffectiveUserName in Excel Online.

To enable EffectiveUserName in Excel Online, run the following command:

```
Set-OfficeWebAppsFarm -ExcelUseEffectiveUserName:$True
```

To disable EffectiveUserName in Excel Online, run the following command:

```
Set-OfficeWebAppsFarm -ExcelUseEffectiveUserName:$False
```

Working with large workbooks

When opening a workbook in Excel Online, there is a time limit of one minute before Excel Online will time out and fail to load the workbook. Occasionally, this time limit may not be enough to load large workbooks. If you run into problems loading large workbooks, you can adjust the timeout value.

To change the timeout value, you must update the settings.xml file on each computer running Office Online Server. (This file is normally located at C:\ProgramData\Microsoft\OfficeWebApps\Data\FarmState\settings.xml.)

Add the following value to the settings.xml file, where *TimeoutValue* is the timeout value in milliseconds:

```
<Setting Name="FBDirectReadTimeoutInMilliseconds" DataType="System.Int32">
    <StringValue>TimeoutValue</StringValue>
</Setting>
```

Note that a timeout value of 0 will make the timeout indefinite. This is not recommended as it increases the risk of a denial-of-service attack.

Data authentication for Excel Online in Office Online Server

3/24/2021 • 9 minutes to read

Summary Learn how Excel Online supports connections with SQL Server Analysis Services (SSAS), SQL Server databases, and OLE DB and ODBC data sources.

Retrieving data from a data source requires a user to be authenticated by the data source and then authorized to access the data that is contained therein. In the case of a workbook, Excel Online authenticates to the data source on behalf of the user who is viewing it in order to refresh the data to which the workbook is connected.

Which authentication method Excel Online can use to retrieve data depends on the type of the underlying data source, as outlined in the following table. For data sources that support more than one authentication method, data connections must specify which one to use.

Data sources and authentication methods for Excel Online

DATA SOURCE	AUTHENTICATION METHOD
Analysis Services	Windows authentication (integrated security) using Kerberos Constrained Delegation using Secure Store using the EffectiveUserName connection string property
SQL Server	One of: Windows authentication (integrated security) using Kerberos Constrained Delegation using Secure Store SQL Server Authentication
Custom data providers	Varies per data source, typically a user-name and password pair stored in the connection string.

The following data sources are supported in Excel but not in Excel Online:

- Access databases
- Web content
- XML data
- Microsoft Azure Marketplace
- Text files

Connecting to external data with Excel Online

Excel Online can connect to various external data sources, including SQL Server, Analysis Services, and custom OLE DB/ODBC data providers. To connect to the data source, Excel Online uses a specific data provider for each data source.

Connecting to a SQL Server data source can be done by using either:

- Windows authentication

- SQL Server Authentication

Connecting to an Analysis Services data source is done by using Windows authentication.

Other data sources use a connection string usually consisting of a user name and password.

Data connections for Excel Online workbooks

Excel Online workbooks use one of two kinds of connections:

- Embedded connections
- Linked connections

Embedded connections are stored as part of the Excel workbook. Linked connections are stored externally to a workbook in Office Data Connection (ODC) files. To use a linked connection, a workbook must reference an .odc file that is also stored in the same SharePoint Server farm as the workbook. Each data connection consists of:

- A connection string
- A query string
- An authentication method
- Optionally, some metadata required to retrieve external data

Each kind of connection has its advantages and drawbacks discussed here. Choose the one that best suits your scenario.

Comparison of data connections for Excel Online

CONNECTION TYPE	EMBEDDED CONNECTIONS	ODC FILES
Advantages	All connection information is stored in the workbook. Embedded connections require little administrative overhead to support. Embedded connections are easy to create.	Linked connections can be centrally stored, managed, audited, shared and access to them can be controlled by using a SharePoint document library. Workbook authors can use existing connections without having to create queries and connection strings. If the data connection details for a data source change, an administrator only need update one ODC file. With that change, all workbooks that refer to the ODC file will use the updated connection information when the next refresh occurs. (An example of this scenario is when the database server is moved or the database name is changed.)
Drawbacks	If the data connection details for a data source change, all workbooks with embedded connections to that data source will have to be republished with updated connection information. Embedded data connections are more difficult to audit by SharePoint administrators.	Linked connections may require the help of a SharePoint administrator to share, manage and secure. Linked connections are saved in clear text and may contain database passwords. Extra care must be taken to help secure these files. Requires server-to-server authentication between Office Online Server and SharePoint Server. This adds configuration and administration overhead.

Choose a linked data connection, by using an ODC file, for scenarios in which you must have a data connection to an enterprise-scale data source such as SQL Server or Analysis Services. Linked data connections are most useful in scenarios in which they will be shared across many users and in which administrator control of the connection is important.

Choose an embedded connection for scenarios where you need a data connection that will not be widely used.

ODC files can be centralized in a data connection library. Doing so has several advantages:

- Administrators can restrict write access to a data connection library to trusted data connection authors to ensure that only well tested and secure data connections are used by workbook authors.
- Administrators have a single location to manage data connections for a large group of users.
- Administrators can easily approve, audit, revert and manage data connection files by using document library versioning and workflow features.
- Data connection libraries can be reused across other Office applications such as Visio and Visio Services.
- Workbook authors only have a single location to find workbook data connections, reducing confusion and user training.

Windows authentication

Windows authentication requires that Excel Online present to the data source a set of Windows credentials. This kind of credential is common on Windows networks and is the same credential used to log on to computers on a Windows domain. Windows credentials are considered the most secure and manageable means of controlling access to SQL Server databases. However, one obstacle to using Windows authentication with Excel Online is the Windows double hop security measure, wherein a user's credentials cannot be passed across more than one computer in a Windows network. Given that Excel Online used with SharePoint Server is a multi-tiered system, special authentication methods are required for Excel Online to retrieve data on behalf of the end-user.

The authentication method to choose depends on various factors as outlined in the following table. Choose the one that best suits your scenario.

Comparison of authentication methods

AUTHENTICATION METHOD	KERBEROS DELEGATION	SECURE STORE	EFFECTIVE USER NAME
Description	Using Kerberos constrained delegation, the workbook viewer's Windows credentials are sent to the data source directly.	Using the Secure Store Service, the viewer's Windows credentials are mapped to another set of credentials specified in a Secure Store target application.	Using the EffectiveUserName Global Setting, the user's domain user name is passed to Analysis Services data sources.
Data connection credentials	The Windows credentials of the workbook viewer.	The credentials specified in the Secure Store target application.	The credentials of the Office Online Server process identity.

AUTHENTICATION METHOD	KERBEROS DELEGATION	SECURE STORE	EFFECTIVE USER NAME
Advantages	The Kerberos protocol is an industry standard in credentials management. Kerberos ties into the existing Active Directory infrastructure. Kerberos delegation permits auditing of individual accesses to a data source. Given that the workbook viewer's identity is known, workbook creators can embed personalized database queries into workbooks.	The Secure Store Service is part of SharePoint Server and is easier to configure than Kerberos. Mappings are flexible: a user can be mapped either 1-to-1 or many-to-1. Non-Windows credentials can be used to connect to data sources that do not accept Windows credentials. Mappings created for Excel Online can be re-used by other business intelligence applications such as Visio Services.	Per-user data security without the need to configure Kerberos delegation. Minimal configuration and administrative overhead.
Drawbacks	Additional administrative effort required to configure SharePoint Server and Excel Online.	Establishing and managing mapping tables requires some administrative overhead. Secure Store permits limited auditing. In the many-to-1 scenario, individual incoming users are mapped into the same credentials through a target application, effectively blending them into one user.	Only works with Analysis Services data sources.
For the authentication operation to succeed ...	Kerberos delegation must be set up on the Office Online Server.	The Secure Store Service must be provisioned and configured on the SharePoint Server farm. It must also contain appropriate mapping information for a particular incoming user. Additionally the mapping information may need to be updated periodically to reflect password changes on the mapped account.	The **EffectiveUserName** option must be enabled in Office Online Server. The user must be a member of the appropriate Analysis Services role.

Kerberos delegation

Choose Kerberos delegation for secure and fast authentication to enterprise-scale relational data sources that support Windows authentication. If you plan to configure Kerberos constrained delegation, the following requirements are specific to using Kerberos constrained delegation with Excel Online in Office Online Server:

- The Claims to Windows Token Service must be running on each server in the Office Online Server farm and set to run as Local System.
- Each server in the Office Online Server farm must be allowed to delegate to each back-end data source as shown in the Active Directory Domain Services delegation list.
- The c2wtshost.exe.config file (located at \Program Files\Windows Identity Foundation\v3.5) must be updated and the comment tags removed from *NT AUTHORITY\Network Service* allowedCallers list:

Secure Store

Choose Secure Store for authentication to enterprise-scale relational data sources that may or may not support Windows Authentication. Secure Store is also useful in scenarios in which you want to control user credential mappings.

For information about using Secure Store with Excel Online, see:

- Configure Excel Online data refresh by using embedded data connections in Office Online Server
- Configure Excel Online data refresh by using external data connections in Office Online Server

SQL Server Authentication

SQL Server Authentication requires that Excel Online present a SQL Server user name and password to a SQL Server data source to authenticate. Excel Online passes the connection string to the data source. The connection string must contain the user name and password.

If the user name and password are stored in a Secure Store target application (recommended for best security), then Excel Online will impersonate the Office Online Server network service account and when the connection is made, the SQL credentials are set as properties of the connection.

Authentication against OLEDB/ODBC data sources

Authentication to third party data sources typically requires that Excel Online present a user name and password to a data source.

If the user name and password are stored in the workbook or in the ODC file, then Excel Online impersonates a Windows identity dependent on which option has been selected for **Excel Services authentication settings**, either in the workbook or in the ODC file.

If the user name and password are stored in a Secure Store target application (recommended for best security), then Excel Online impersonates the Office Online Server network service account and when the connection is made, the SQL credentials are set as properties of the connection.

Data refresh in Excel Online

Excel Online supports refreshing workbooks connected to one or more of the following data sources:

- SQL Server
- SQL Server Analysis Services (SSAS)

> **NOTE**
> If the data source that you plan to connect to is not in the list above, you can add support for it by creating a Custom Data Provider. This technology enables you to wrap your existing data sources into one that Excel Online can consume.

External data refresh is the result of the following set of steps through Excel Online.

1. **Creating a workbook:** A workbook author uploads a data-connected workbook to SharePoint Server.
2. **Triggering Refresh:** The workbook viewer triggers refresh on a data-connected workbook.
3. **Data Connections:** Excel Online retrieves data connection information for each external data source in the workbook.
4. **Trusted Data Providers:** Excel Online checks to see if there is a trusted data provider it can use to retrieve data.
5. **Authentication:** Excel Online authenticates into the data source and retrieves the requested data on

behalf of the workbook viewer.

6. **Workbook Refresh:** Excel Online updates the workbook based on the data source data and returns it to the viewer.

Refresh can be triggered in one of following ways from within the browser:

- The end-user opens the workbook (if the workbook is configured to refresh on open).
- The end-user clicks on the refresh button on an already open workbook.

If there are no previously cached versions of this workbook, any of these actions will trigger a refresh and update the workbook.

See also

Configure Analysis Services and Kerberos Constrained Delegation (KCD)

Configure Excel Online data refresh by using embedded data connections in Office Online Server

3/24/2021 • 5 minutes to read

Summary: Configure Excel Online to use a Secure Store target application for external data refresh.

Excel Online supports two methods of using Secure Store Service to connect to external data:

- You can specify a Secure Store target application in a workbook. (This is known as an embedded connection.) This article describes how to do this.
- You can use an Office Data Connection (ODC) file that specifies a Secure Store target application. For more information, see Configure Excel Online data refresh by using external data connections in Office Online Server.

To configure Excel Online data access to use embedded data connections, you use the following process:

1. Configure a data access account
2. Create a Secure Store target application
3. Configure an Excel workbook to use an embedded data connection

Note that you must have installed Office Online Server and configured SharePoint Server to use it to render documents for this to work.

Configure a data access account

You must have an account that can be granted access to the data source to which you want to connect your Excel workbook. This can be a Windows Active Directory account, a SQL Server logon, or other set of credentials as required by your data source. This account will be stored in Secure Store.

Once you have created the account, the next step is to grant that account read access to the required data. (In this article, we use the example of accessing a SQL Server database through an Active Directory account. If you are using a data source other than SQL Server, see the instructions for your data source to create a logon with data-read permissions for the data access account.)

Follow these steps to create a SQL Server logon and grant Read access to the database.

To create a SQL Server logon for the data access account

1. In SQL Server Management Studio, connect to the database engine.
2. In Object Explorer, expand **Security**.
3. Right-click **Logins**, and then click **New Login**.
4. In the **Login name** box, type the name of the Active Directory account that you created for data access.
5. In the **Select a page** section, click **User Mapping**.
6. Select the **Map** check box for the database that you want to provide access to, and then, under **Database role membership for:** , select the **db_datareader** check box.
7. Click **OK**.

Now that you have created a data access account and granted it access to a data source, the next step is to create a Secure Store target application.

Create a Secure Store target application

You must create a target application in Secure Store that contains the credentials that you created for data access. This target application can then be specified in data-connected Excel workbooks and will be used by Excel Online when it refreshes data in the workbook.

When you create the target application, you have to specify which users will be authorized to use the credentials stored in Secure Store. You can list users individually, or you can use an Active Directory group. We recommend that you use an Active Directory group for ease of administration.

> **NOTE**
> The users that you list in the target application do not have direct access to the stored credentials. Instead, Excel Online uses the credentials on their behalf to refresh data in data-connected workbooks that specify this target application.

Use the following procedure to create a Secure Store target application.

To create a target application

1. On the Central Administration home page, in the **Application Management** section, click **Manage service applications**.
2. Click the Secure Store service application.
3. On the ribbon, click **New**.
4. In the **Target Application ID** box, type a unique identifier for this target application (for example,ExcelOnlineDataAccess).
5. In the **Display Name** box, type a friendly name or short description.
6. In the **Contact E-mail** box, type the e-mail address for a contact for this target application.
7. In the **Target Application Type** drop-down list, select **Group**.
8. Click **Next**.
9. On the Credential Fields page, if you are using Windows credentials, leave the default credential fields. If you are using credentials other than Windows credentials, modify the **Field Type** drop-down lists to comply with the credentials that you are using. Click **Next**.
10. On the Specify the membership settings page:

- In the **Target Application Administrators** box, type the account of the user who will administer this target application.

  > **NOTE**
  > You can specify multiple users or an Active Directory group.

- In the **Members** box, type the users to whom you want to grant the ability to refresh data.

> **NOTE**
> You can specify multiple users or an Active Directory group.

11. Click **OK**.

Use the following procedure to set the credentials for the target application.

To set the credentials for the target application

1. On the Secure Store Service Application page, in the **Target Application ID** column, point to the target application that you just created, click the arrow that appears, and then click **Set Credentials**.
2. Type the user name and password of the data access account.
3. Click **OK**.

Once you have set the credentials for the target application, the target application is ready to use. The next step is to specify this target application in the Excel Online authentication settings in your data-connected Excel workbook.

Configure an Excel workbook to use an embedded data connection

You must configure the Excel Online authentication settings in the workbook before you publish it to SharePoint Server 2016. Doing so enables the workbook to use a Secure Store target application to refresh data that is rendered using Excel Online. Use the following procedure to configure the authentication settings.

To configure Excel Online authentication settings

1. In a data-connected Excel workbook, on the **Data** tab, click **Connections**.
2. On the **Workbook Connections** dialog box, select the data connection that you want to update, and then click **Properties**.
3. On the **Connection Properties** dialog box, on the **Definition** tab, click **Authentication Settings**.
4. On the **Excel Services Authentication Settings** dialog box, select the **Use a stored account** option, type the Application ID of the target application in the text box, and then click **OK**.

   > **NOTE**
   > If you are using Excel 2010, select the **SSS** option.

5. On the **Connection Properties** dialog box, click **OK**.

   > **NOTE**
   > If you see a warning that the link to the external connection file will be removed, click **Yes**.

6. On the Workbook Connections dialog box, click **Close**.

With the target application specified in the Excel Online authentication settings, Excel Online uses the credentials associated with that target application to refresh the data in the workbook after you have published it to SharePoint Server 2016.

See also

Configure the Secure Store Service (SharePoint Server 2013)

Configure Excel Online data refresh by using external data connections in Office Online Server

3/24/2021 • 7 minutes to read

Summary: Configure Excel Online data refresh by using Secure Store and an external Office Data Connection (ODC) file.

Excel Online provides two methods of using Secure Store Service to refresh the external data source in a workbook:

- You can specify a Secure Store target application in a workbook. (This is known as an embedded connection.) For more information, see Configure Excel Online data refresh by using embedded data connections in Office Online Server.
- You can use an Office Data Connection (ODC) file that specifies a Secure Store target application. This article describes how to do this.

By using an ODC file for your data connection, you separate your Excel workbooks from the data connection information. This allows you to share a single ODC file among multiple workbooks and also allows you to centrally manage your data connections.

Before you can use Excel Online with an ODC file, you must have installed Office Online Server and configured it to work with SharePoint Server. To use an ODC file, you must also configure server-to-server authentication between Office Online Server and SharePoint Server.

Using Excel Online with an ODC file consists of the following steps:

1. Configure a data access account
2. Create a Secure Store target application
3. Create and publish an ODC file
4. Configure an Excel workbook to use the published ODC file as a data connection

Configure a data access account

You must have an account that can be granted access to the data source to which you want to connect your Excel workbook. This can be an Active Directory account, a SQL Server logon, or other set of credentials as required by your data source. This account will be stored in Secure Store.

After you have created the account, the next step is to grant that account read access to the required data. (in this article, we use the example of accessing a SQL Server database through an Active Directory account. If you are using a data source other than SQL Server, see the instructions for your data source to create a logon with data read permissions for the data access account.)

Follow these steps to create a SQL Server logon and grant Read access to the database.

To create a SQL Server logon for the data access account

1. In SQL Server Management Studio, connect to the database engine.
2. In Object Explorer, expand **Security**.
3. Right-click **Logins**, and then click **New Login**.

4. In the **Login name** box, type the name of the Active Directory account that you created for data access.
5. In the **Select a page** section, click **User Mapping**.
6. Select the **Map** check box for the database that you want to provide access to, and then, in the **Database role membership for:** section, select the **db_datareader** check box.
7. Click **OK**.

Now that you have created a data access account and granted it access to a data source, the next step is to create a Secure Store target application.

Create a Secure Store target application

You must create a target application in Secure Store that contains the credentials that you created for data access. This target application can then be specified in an ODC file and will be used by Excel Online when it refreshes data in the workbook.

When you create the target application, you have to specify which users will be authorized to use the credentials stored in Secure Store. You can list users individually, or you can use an Active Directory group. We recommend that you use an Active Directory group for ease of administration.

> **NOTE**
> The users that you list in the target application do not have direct access to the stored credentials. Instead, Excel Online uses the credentials on their behalf to refresh data in data-connected workbooks that specify this target application.

Use the following procedure to create a Secure Store target application.

To create a target application

1. On the the SharePoint Central Administration website home page, in the **Application Management** section, click **Manage service applications**.
2. Click the Secure Store Service service application.
3. On the ribbon, click **New**.
4. In the **Target Application ID** box, type a unique identifier for this target application (for example,ExcelOnlineDataAccess).
5. In the **Display Name** box, type a friendly name or short description.
6. In the **Contact E-mail** box, type the email address for a contact for this target application.
7. In the **Target Application Type** drop-down list, select **Group**.
8. Click **Next**.
9. On the Credential Fields page, if you are using Windows credentials, leave the default credential fields. If you are using credentials other than Windows credentials, modify the **Field Type** drop-down list to comply with the credentials that you are using. Click **Next**.
10. On the Specify the membership settings page:

- In the **Target Application Administrators** box, type the account of the user who will administer this target application.

> **NOTE**
> You can specify multiple users or an Active Directory group.

- In the **Members** box, type the users to whom you want to grant the ability to refresh data.

> **NOTE**
> You can specify multiple users or an Active Directory group.

11. Click **OK**.

Use the following procedure to set the credentials for the target application.

To set the credentials for the target application

1. On the Secure Store Service Application page, in the **Target Application ID** column, point to the target application that you just created, click the arrow that appears, and then click **Set Credentials**.
2. Type the user name and password of the data access account.
3. Click **OK**.

Once you have set the credentials for the target application, the target application is ready to use. The next step is to create an ODC file that specifies this target application for Excel Online data refresh.

Create and publish an ODC file

Now that the Secure Store target application is configured, the next step is to create the ODC file and publish it to a SharePoint Server 2016 library. Use the following procedure to create an ODC file that specifies the target application that you just created.

To create and publish an ODC file

1. In Excel, on the **Data** tab, in the **Get External Data** section, click **From Other Sources**, and then select your data source.
2. Complete the wizard to create a data connection to your data source.
3. On the **Data** tab, click **Connections**.
4. On the **Workbook Connections** dialog box, select the connection that you just created, and then click **Properties**.
5. On the **Connection Properties** dialog box, on the **Definition** tab, click **Authentication Settings**.
6. On the **Excel Services Authentication Settings** dialog box, select the **Use a stored account** option, and in the **Application ID** box, type the Application ID of the Secure Store target application that you created.

> **NOTE**
> In Excel 2010, select the **SSS** option.

7. Click **OK**.
8. On the **Connection Properties** dialog box, click **Export Connection File**.

9. Save the ODC file to a data connection or document library on your farm.

Configure an Excel workbook to use the published ODC file as a data connection

In order for a workbook to use the ODC file that you just created, you must connect to it as a data source. Once it is connected, you can publish the workbook to a SharePoint Server 2016 document library and it will maintain its connection to the ODC file. Excel Online then uses the connection information specified in the ODC file when it refreshes data in the workbook.

Use the following procedure to connect to the ODC file in Excel.

To use an ODC file as a data source in Excel

1. In Excel, on the **Data** tab, in the **Get External Data** section, click **Existing Connections**.
2. On the **Existing Connections** dialog box, click **Browse for More**.
3. On the **Select Data Source** dialog box, in the URL box, type the URL for the library where you saved the ODC file, and then press Enter.

   > **NOTE**
   > It may take several moments for the list to refresh with content from the specified location.

4. On the list of **Data Connections**, select the ODC file that you saved, and then click **Open**.
5. On the **Import Data** dialog box, select the **PivotTable Report** or **PivotChart and PivotTable Report** option, and then click **OK**.
6. On the **Data** tab, click **Connections**.
7. On the **Workbook Connections** dialog box, select the connection that you just opened, and then click **Properties**.
8. On the **Connection Properties** dialog box, on the **Definition** tab, select the **Always use connection file** check box, and then click **OK**.

   > **NOTE**
   > This ensures that the connection file that you connected to will be used rather than the embedded connection information.

9. Click **Close**.

Once you have completed the data connection wizard, you can create your report and then publish it to a document library. When the workbook is rendered using Excel Online, Excel Online uses the connection information specified in the ODC file to refresh the data.

See also

Configure the Secure Store Service (SharePoint Server 2013)

Office Online Server release schedule

7/31/2019 • 2 minutes to read

Summary: Learn about the Office Online Server release schedule and upgrade requirements.

Microsoft plans to support the most current Office Online Server build with security and critical updates until the next one is released.

Previous builds will continue to function. However, if you request support that ultimately requires a hotfix, you will be required to upgrade to the latest build.

The following table shows the Office Online Server build release history and current build.

(You can determine your current build by looking at the **Version** number in Add/Remove Programs.)

BUILD	AVAILABILITY DATE	SUPPORT END DATE
16.0.6814.2226	4-May-2016	18-November-2016
16.0.7601.6800	18-November-2016	18-April-2017
16.0.7766.8550	18-April-2017	8-November-2017
16.0.8471.8525	8-November-2017	30-November-2018
16.0.10338.20039	30-November-2018	TBD

See also

Apply software updates to Office Online Server

Apply software updates to Office Online Server

9/20/2021 • 6 minutes to read

Summary: Explains how to apply software updates or new versions to an Office Online Server farm.

Audience: IT Professionals

On a regular basis, Microsoft makes a series of software updates and new versions available to help improve server security, performance, and reliability. This article describes how to apply software updates or new versions to individual servers in an Office Online Server farm.

> **IMPORTANT**
>
> **Are you looking for help with Office Online on your desktop or mobile device?** You can find this information by searching for "Office Online" on Office.com.

Caution

Applying Office Online Server updates or new versions by using the automatic updates process isn't supported with Office Online Server. This is because updates to an Office Online Server must be applied in a specific way, as described in this article. If Office Online Server updates are applied automatically, users may be unable to view or edit documents in Office Online. If this happens, you have to rebuild your Office Online Server farm. To rebuild a farm, you must remove the Office Online Server from the farm by using Remove-OfficeWebAppsMachine, uninstall Office Online Server by using Add or remove programs, and then reinstall Office Online Server by following the steps that are described in Deploy Office Online Server. After you have reinstalled, apply the update by following the steps that are described in this article. > It is important that you review the guidelines in Planning updates for Office Online Server and establish an update process for the Office Online Server farm.

Before you begin

Updates that are released for Office Online Server will update Office Online Server and any Office Online Server language packs that are installed. There are no separate updates for Office Online Server language packs.

As part of the update process, you'll have to re-create the Office Online Server farm. To prepare to re-create the Office Online Server farm, review your current Office Online Server farm properties by running the Microsoft PowerShell cmdlet **Get-OfficeWebAppsFarm** and review the parameters for New-OfficeWebAppsFarm. The parameters that you use for **New-OfficeWebAppsFarm** should be the same parameters that you used when you first set up the Office Online Server farm.

Note that when you've completed updating your farm, you need to re-add any data model servers that you had configured for Excel Online.

Apply software updates or new versions to a single server Office Online Server farm

To apply software updates or new versions to a single server Office Online Server farm, remove the Office Online Server from the farm, apply the software update or new version, and re-create the Office Online Server farm. If you have only one server in your Office Online Server farm, users won't be able to use Office Online while you are updating the server. So consider updating the Office Online Server during either non-critical or

non-business hours.

To apply software updates or new versions to a single server farm

1. On the Office Online Server that you want to apply the software update to, open the Microsoft PowerShell prompt as an administrator and run the following command.

```
Remove-OfficeWebAppsMachine
```

2. If this is a new release of Office Online Server that you downloaded from the Volume Licensing Service Center (VLSC), then you must uninstall the existing version before you install the new version.

   (If this is an update from Microsoft Update, then there's no need to uninstall Office Online Server.)

3. Install the Office Online Server update or new version on that server. If prompted, restart the server.

4. Open the Microsoft PowerShell prompt as an administrator and run the **New-OfficeWebAppsFarm** cmdlet to re-create an Office Online Server farm. The URL you specify for **-InternalURL** is the name of the server that runs Office Online Server, such as **https://oos.contoso.com**. In this case, you would use the same name that you used for the previous Office Online Server farm. Use the same additional parameters that you used when you first created the Office Online Server farm. For example, the the **-EditingEnabled** parameter enables editing in Office Online when it is used together with SharePoint Server.

   The code in the following example creates a new Office Online Server farm named https://oos.contoso.com.

```
New-OfficeWebAppsFarm -InternalURL "https://oos.contoso.com" -EditingEnabled
```

Additional parameters that configure translation services, proxy servers, clipart support, and Online Viewers are described in New-OfficeWebAppsFarm.

Apply software updates or new versions to a multiple Office Online Server farm

To apply software updates or new versions to a multiple Office Online Server farm, you first remove one of the servers from the load balancer pool and from the farm, apply the software update or new version, and create an updated Office Online Server farm. Then you remove and update the remaining servers in the Office Online Server farm, and join them to the new updated farm. You load balance traffic to the new farm when you have enough servers in the new farm to support the current traffic. By using this update process, users can open and edit documents in Office Online without disruption.

To apply software updates to a multiple server farm

1. Remove the Office Online Server that you want to apply the software update to from the load balancer pool.

2. On that Office Online Server, open the Microsoft PowerShell prompt as an administrator and run the following command.

```
Remove-OfficeWebAppsMachine
```

3. If this is a new release of Office Online Server that you downloaded from the Volume Licensing Service Center (VLSC), then you must uninstall the existing version before you install the new version.

(If this is an update from Microsoft Update, then there's no need to uninstall Office Online Server.)

4. Install the Office Online Server update or new version on that server. If prompted, restart the server.

5. Open the Microsoft PowerShell prompt as an administrator and create an updated Office Online Server farm by using the cmdlet **New-OfficeWebAppsFarm**. The URL you specify for **-InternalURL** contains the DNS A record of the Office Online Server farm, such as **https://oos.contoso.com**. In this case, you use the same name as the existing Office Online Server farm. Use the same additional parameters that you used when you first created the Office Online Server farm. For example, the **-EditingEnabled** parameter enables editing in Office Online when it is used together with SharePoint Server.

   The code in the following example creates a new Office Online Server farm named https://oos.contoso.com.

```
New-OfficeWebAppsFarm -InternalURL "https://oos.contoso.com"  -EditingEnabled
```

Additional parameters that configure translation services, proxy servers, clipart support, and Online Viewers are described in New-OfficeWebAppsFarm.

6. Depending on how many servers that you have in the Office Online Server farm, load balance traffic to the new farm. You may delay this step until you have more updated servers to join the farm.

7. For each remaining server in the farm, follow these steps.

8. Remove the next Office Online Server from the load balancer pool.

9. Install the Office Online Server update on that server. If prompted, restart the server.

10. Open the Microsoft PowerShell prompt as an administrator and run the following command. The **-MachineToJoin** parameter adds the current server to an existing Office Online Server farm. In this case, you want to add the server to the updated Office Online Server farm. So use the computer name of one of the servers in the updated Office Online Server farm.

```
New-OfficeWebAppsMachine -MachineToJoin "server1.contoso.com"
```

See also

Remove-OfficeWebAppsMachine

New-OfficeWebAppsMachine

New-OfficeWebAppsFarm

Get-OfficeWebAppsFarm

Office Online Server release schedule

Enable TLS 1.1 and TLS 1.2 support in Office Online Server

7/31/2019 • 3 minutes to read

Summary: This article describes how to enable Transport Layer Security (TLS) protocol versions 1.1 and 1.2 in Office Online Server.

To enable TLS protocol versions 1.1 and 1.2 in your Office Online Server environment, you need to configure settings on each server in your Office Online Server farm.

The configuration process involves setting a number of registry keys to turn security protocols on or off. While you can make these updates to the registry manually or by using a .reg file, we recommend that you create group policy objects to manage these settings, particularly if you are configuring these protocols across your organization.

The basic steps covered in this article are:

- Enable strong cryptography in .NET Framework
- (Optional) Disable earlier versions of SSL and TLS

Note that using TLS 1.1 and TLS 1.2 with Office Online Server requires that TLS 1.1 and TLS 1.2 be enabled on Windows Server for each computer in your Office Online Server farm. They are enabled by default for Windows Server 2012 R2.

Follow these steps on each server in your Office Online Server farm.

Enable strong cryptography in .NET Framework 4.5 or higher

Using TLS 1.1 and TLS 1.2 with Office Online Server requires strong cryptography in .NET Framework 4.5 or higher. To enable strong cryptography in .NET Framework 4.5 or higher, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\.NETFramework\\v4.0.30319]
"SchUseStrongCrypto"=dword:00000001

[HKEY_LOCAL_MACHINE\\SOFTWARE\\Wow6432Node\\Microsoft\\.NETFramework\\v4.0.30319]
"SchUseStrongCrypto"=dword:00000001
```

Disable earlier versions of SSL and TLS in Windows Schannel

SSL and TLS support are enabled or disabled in Windows Schannel by editing the Windows Registry. Each SSL and TLS protocol version can be enabled or disabled independently. You don't need to enable or disable one protocol version to enable or disable another protocol version.

> **IMPORTANT**
>
> Microsoft recommends disabling SSL 2.0 and SSL 3.0 due to serious security vulnerabilities in those protocol versions. > Customers may also choose to disable TLS 1.0 and TLS 1.1 to ensure that only the newest protocol version is used. However, this may cause compatibility issues with software that doesn't support the newest TLS protocol version. Customers should test such a change before performing it in production.

The **Enabled** registry value defines whether the protocol version can be used. If the value is set to 0, the protocol version cannot be used, even if it is enabled by default or if the application explicitly requests that protocol version. If the value is set to 1, the protocol version can be used if enabled by default or if the application explicitly requests that protocol version. If the value is not defined, it will use a default value determined by the operating system.

The **DisabledByDefault** registry value defines whether the protocol version is used by default. This setting only applies when the application doesn't explicitly request the protocol versions to be used. If the value is set to 0, the protocol version will be used by default. If the value is set to 1, the protocol version will not be used by default. If the value is not defined, it will use a default value determined by the operating system.

To disable SSL 2.0 support in Windows Schannel, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\SSL
2.0\\Client]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\SSL
2.0\\Server]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000
```

To disable SSL 3.0 support in Windows Schannel, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\SSL
3.0\\Client]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\SSL
3.0\\Server]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000
```

To disable TLS 1.0 support in Windows Schannel, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\TLS
1.0\\Client]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\TLS
1.0\\Server]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000
```

To disable TLS 1.1 support in Windows Schannel, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\TLS
1.1\\Client]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Control\\SecurityProviders\\SCHANNEL\\Protocols\\TLS
1.1\\Server]
"DisabledByDefault"=dword:00000001
"Enabled"=dword:00000000
```

Windows PowerShell for Office Online Server

3/24/2021 • 2 minutes to read

Summary: Find articles about Office Online Server Microsoft PowerShell cmdlets.

Audience: IT Professionals

The following table lists and describes the articles for each OfficeWebApps Microsoft PowerShell cmdlet for Office Online Server.

> **TIP**
> If Microsoft PowerShell doesn't recognize these cmdlets when you run them, you may need to import the **OfficeWebApps** module. Use this command: `Import-Module -Name OfficeWebApps`

ARTICLE	DESCRIPTION
Get-OfficeWebAppsExcelBIServer	Returns the instance of Analysis Services that has been configured as a data model server in Office Online Server.
Get-OfficeWebAppsExcelUserDefinedFunction	Returns a list of currently configured UDF definitions.
Get-OfficeWebAppsFarm	Returns details about the OfficeWebAppsFarm object that the current server is a member of.
Get-OfficeWebAppsHost	Returns the list of host domains that are on the Allow List for an Office Online Server farm.
Get-OfficeWebAppsMachine	Returns details about the current server that is in an Office Online Server farm.
New-OfficeWebAppsExcelBIServer	Configures Analysis Services servers to work with Excel Online.
New-OfficeWebAppsExcelUserDefinedFunction	Creates a definition for a UDF binary.
New-OfficeWebAppsFarm	Creates a new Office Online Server farm on the local computer.
New-OfficeWebAppsHost	Adds a host domain to the Allow List for an Office Online Server farm.
New-OfficeWebAppsMachine	Adds the current server to an existing Office Online Server farm.
Remove-OfficeWebAppsExcelBIServer	Removes an instance of Analysis Services from the Allow List of BI servers to be used with Excel Online.
Remove-OfficeWebAppsExcelUserDefinedFunction	Removes an existing UDF definition.

ARTICLE	DESCRIPTION
Remove-OfficeWebAppsHost	Removes a host domain from the Allow List for an Office Online Server farm.
Remove-OfficeWebAppsMachine	Removes the current server from the Office Online Server farm.
Repair-OfficeWebAppsFarm	Removes all servers flagged as unhealthy from an Office Online Server farm.
Set-OfficeWebAppsExcelUserDefinedFunction	Sets properties on existing UDF definitions.
Set-OfficeWebAppsFarm	Configures the settings of an existing Office Online Server farm.
Set-OfficeWebAppsMachine	Changes the settings of the current server that is in an Office Online Server farm.