INTEGRATION TO MICROSOFT EXCHANGE Installation Guide

V44.1

Last Updated: March 5, 2018

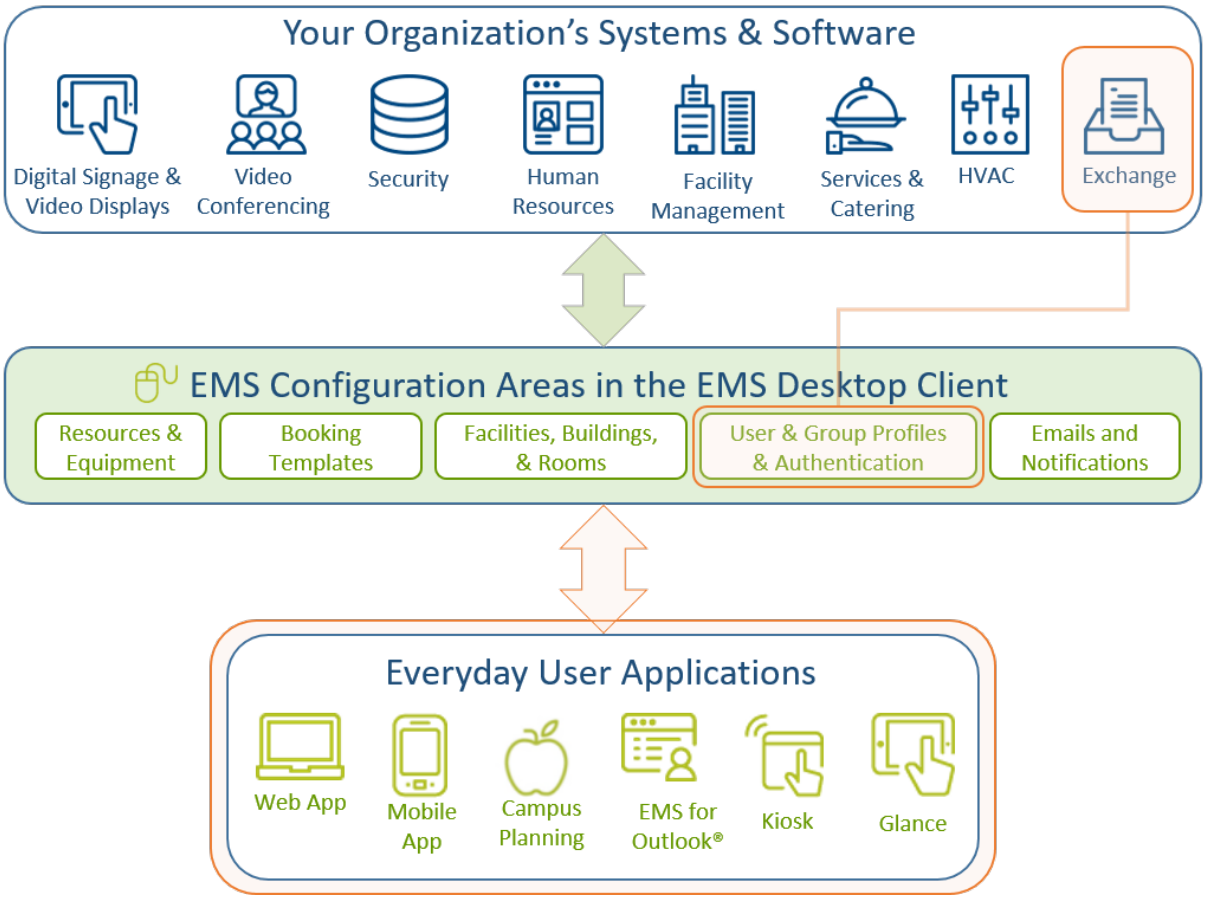
Table of Contents

CHAPTER 1: Introduction to Integration to Microsoft Exchange

This guide provides instruction in installing Integration to Microsoft Exchange for System Administration and IT users.

EMS Integration with Microsoft® Exchange is a component that integrates EMS Everyday User applications, such as EMS Mobile App, EMS for Outlook and EMS Web App, with Microsoft® Exchange. This module enables everyday users to view the availability of both meeting rooms *and* attendees, and send Outlook® meeting invitations, all from within EMS Everyday User applications.

EXCHANGE INTEGRATION FLOW



You must be licensed for EMS, EMS Web App, and EMS Integration with Microsoft® Exchange in order to configure and use this feature. If you are unsure if your organization is licensed for EMS Integration with

Microsoft® Exchange, or if you would like to learn more about it, please contact your Account Executive.

To install and configure EMS Integration with Exchange, you will:

- Install the Exchange Integration Web Service
- Configure EMS Integration to Exchange
- Configure EWS Impersonation for Exchange Online (Office 365)

SYSTEM REQUIREMENTS

You must be licensed for EMS, EMS Web App, and Integration with Exchange in order to configure and use this module. If you are unsure if your organization is licensed for Integration with Exchange, or if you would like to learn more about it, please contact your Account Executive.

The following requirements must be met to install and configure Integration to Microsoft® Exchange:

- EMS and/or EMS Web App Installed

EMS must be installed and operational.

- Valid Outlook Integration License

You must be licensed for EMS, EMS Web App and Integration with Exchange in order to configure and use this module. If you are unsure if your organization is licensed for Integration with Exchange, or if you would like to learn more about it, please contact your Account Executive.

WEB APPLICATION REQUIREMENTS

DESKTOP BROWSER
Internet Explorer 11 (please see Tip below)
Microsoft Edge (latest)
Firefox (latest)
Chrome (latest)
Safari (Mac) (latest)

*= varies per application

> **TIP:** EMS Web App V44.1 has been optimized for Internet Explorer 11 and

does not require compatibility with previous versions of Internet Explorer. EMS recommends *disabling compatibility mode* when using Web App V44.1.

EMS WEB APP (MOBILE)

MOBILE BROWSER	PLATFORM
Internet Explorer for Mobile 8.1	Windows
Internet Explorer for Mobile 10	Windows
Chrome	Android, 4.4, 6.0, 7.0, 7.1
	iOS 9.x, iOS 10.x
Safari	iOS 9.x, iOS 10.x

IMPORTANT: Integration with Exchange configuration issues often relate to access rights with this account. Please ensure that the account has the necessary permissions.

WEB SERVER REQUIREMENTS

OPERATING SYSTEM	IIS APP POOL
Windows Server 2008 R2	7/7.5
Windows Server 2012	8
Windows Server 2012 R2	8.5
Prerequisites	
Application Pool Running 4.0*	
.NET Framework 4.6.1*	
Minimum System Requirements	
Processor: 2.0 GHz and 4 cores or faster	
Memory: 8 GB or more*	
Hard-Disk Space: 1 GB or more	

OPERATING SYSTEM	IIS APP POOL

*For up to 100 concurrent users. Increased specs required for 100+ concurrent users.

*= varies per EMS Software Application

EXCHANGE INTEGRATION REQUIREMENTS

Microsoft® Office	365
Outlook	2010, 2013, 2016
.NET Framework	4.6.1
Microsoft® Visual Studio 2010 Tools for Office Runtime	VSTOR 2010

Prerequisites

EMS Web App	Latest

On User Workstations	Desktop requirements for Microsoft® Outlook Windows 7, 8, or 10

EMS PLATFORM SERVICES

OPERATING SYSTEM	IIS
Windows Server 2008 R2	7/7.5
Windows Server 2012	8
Windows Server 2012 R2	8.5
.NET Framework	4.6.1
Application Pool	4.0
Prerequisites	
HTTPPlatformHandler IIS Module	Download Version 1.2 here OR download the installer here.

OPERATING SYSTEM	IIS
PowerShell	5+ Version
ASP.NET Version 4.6	Under Web Server (IIS)->Web Server->Application Development: » ISAPI Extensions » ISAPI Filters » .NET Extensibility 4.6

EMS INTEGRATION FOR EXCHANGE

Microsoft® Exchange	2010 SP3, 2013, 2016
Microsoft® Office	365

Configure EWS Impersonation for Microsoft® Exchange

CHAPTER 2: System Requirements for Integration to Microsoft® Exchange

You must be licensed for EMS, EMS Web App, and Integration with Exchange in order to configure and use this module. If you are unsure if your organization is licensed for Integration with Exchange, or if you would like to learn more about it, please contact your Account Executive.

The following requirements must be met to install and configure Integration to Microsoft® Exchange:

- EMS and/or EMS Web App Installed

EMS must be installed and operational.

- Valid Outlook Integration License

WEB APPLICATION REQUIREMENTS

DESKTOP BROWSER
Internet Explorer 11 (please see Tip below)
Microsoft Edge (latest)
Firefox (latest)
Chrome (latest)
Safari (Mac) (latest)

*= varies per application

> **TIP:** EMS Web App V44.1 has been optimized for Internet Explorer 11 and does not require compatibility with previous versions of Internet Explorer. EMS recommends disabling compatibility mode when using Web App V44.1.

EMS WEB APP (MOBILE)

MOBILE BROWSER	PLATFORM
Internet Explorer for Mobile 8.1	Windows
Internet Explorer for Mobile 10	Windows
Chrome	Android, 4.4, 6.0, 7.0, 7.1
	iOS 9.x, iOS 10.x
Safari	iOS 9.x, iOS 10.x

IMPORTANT: Integration with Exchange configuration issues often relate to access rights with this account. Please ensure that the account has the necessary permissions.

WEB SERVER REQUIREMENTS

OPERATING SYSTEM	IIS APP POOL
Windows Server 2008 R2	7/7.5
Windows Server 2012	8
Windows Server 2012 R2	8.5
Prerequisites	
Application Pool Running 4.0*	
.NET Framework 4.6.1*	
Minimum System Requirements	
Processor: 2.0 GHz and 4 cores or faster	
Memory: 8 GB or more*	
Hard-Disk Space: 1 GB or more	

OPERATING SYSTEM	IIS APP POOL

*For up to 100 concurrent users. Increased specs required for 100+ concurrent users.

*= varies per EMS Software Application

EXCHANGE INTEGRATION REQUIREMENTS

Microsoft® Office	365
Outlook	2010, 2013, 2016
.NET Framework	4.6.1
Microsoft® Visual Studio 2010 Tools for Office Runtime	VSTOR 2010

Prerequisites

EMS Web App	Latest

On User Workstations	Desktop requirements for Microsoft® Outlook Windows 7, 8, or 10

EMS PLATFORM SERVICES

OPERATING SYSTEM	IIS
Windows Server 2008 R2	7/7.5
Windows Server 2012	8
Windows Server 2012 R2	8.5
.NET Framework	4.6.1
Application Pool	4.0
Prerequisites	
HTTPPlatformHandler IIS Module	Download Version 1.2 here OR download the installer here.

OPERATING SYSTEM	IIS
PowerShell	5+ Version
ASP.NET Version 4.6	Under Web Server (IIS)->Web Server->Application Development: » ISAPI Extensions » ISAPI Filters » .NET Extensibility 4.6

EMS INTEGRATION FOR EXCHANGE

Microsoft® Exchange	2010 SP3, 2013, 2016
Microsoft® Office	365

Configure EWS Impersonation for Microsoft® Exchange

CHAPTER 3: Install or Upgrade the Exchange Integration Web Service

PRIOR TO INSTALL OR UPGRADE

> **IMPORTANT:** Before beginning the installation process, complete the following steps.

1. Install or upgrade your EMS databases as outlined in the Desktop Client Installation Instructions.
2. Manually uninstall any previous versions of the Exchange Integration Service on your web server.
3. If you are upgrading from previous versions, update your parameter settings for "PAM Web Service URL" to "Exchange Integration Web Service URL", i.e. http://server/ExchangeIntegrationWebService. See Also: EMS Web App Parameters.

INSTALL OR UPGRADE INSTRUCTIONS

1. Verify that the requirements outlined in the System Requirements section have been met.
2. Download **ExchangeIntegrationWebService.msi** onto the web server that will be running the service.
3. Run **ExchangeIntegrationWebService.msi**.
4. The first screen welcomes you to the Exchange Integration Service Setup Wizard. Click **Next** to begin the installation process. The Destination Folder screen will appear.
5. Select the destination folder. The installation process will create a new physical directory on your web server based on the destination folder path entered ("ExchangeIntegrationService" in the example above.) Click **Next**.

> **NOTE:** The Exchange Integration Service should not be installed in the same physical directory as other EMS web-based products.

6. The SQL Server and database information screen will appear.
7. Enter your EMS SQL Instance Name.
8. Enter your EMS Database Name, typically named "EMS".
9. Click **Next**. The Virtual Directory information screen will appear.

10. The Virtual Directory Name will default to the destination folder specified in Step 5. It is recommended that you keep the default setting. The installation process will create a virtual directory on your web server based on the virtual directory entered ("ExchangeIntegrationWebService" in the example above.) Click Next.

> **NOTE:** The Exchange Integration should not be installed in the same virtual directory as other EMS web-based products.

11. The Ready to Install Exchange Integration Web Service screen will appear. Click **Install** to install the Exchange Integration.
12. The Completed the Exchange Integration Web Service Setup Wizard screen will appear. Click **Finish**.
13. After following the steps above, verify your installation by opening a browser and entering the following:

    http://[ServerName]/ExchangeIntegrationWebService/Service.asmx
    (replace [ServerName] with the name of your web server)

> **IMPORTANT:** A standard installation requires that the Exchange Integration be published without any authentication methods in place (e.g.

Integrated Windows Authentication or Portal Authentication). If you require the Exchange Integration to be secured with authentication, additional configuration is necessary. Contact your implementation consultant for further details.

CHAPTER 4: Configure Integration to Exchange

Configuring EMS to work with Exchange Online (Office 365) or Exchange 2013 is the same as configuring EMS to work with a 2007/2010 Exchange environment that is hosted on your network. See Configure EWS Impersonation for Microsoft® Exchange for information on configuring impersonation on Exchange Online (Office 365). If you need additional assistance configuring this, please contact support@emssoftware.com.

> **Note**: Integration with Microsoft Exchange requires the use of a mail-enabled service account that has the Application/Impersonation role in Exchange for all users who will be accessing EMS. See Also Configuring Exchange Web Service Impersonation.

This topic provides information on the following:

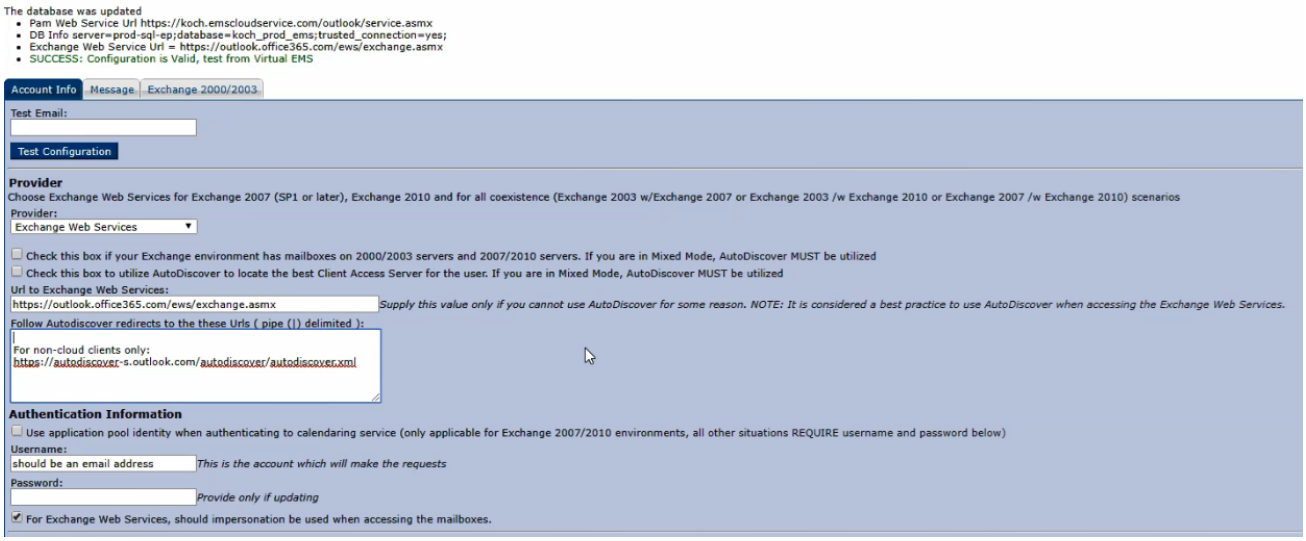
CONFIGURE INTEGRATION TO EXCHANGE INSTRUCTIONS

1. After following the installation instructions, access the Integration with Exchange configuration area by opening a browser and entering the following:

   http://[ServerName]/ExchangeIntegrationWebService/PamConfig.aspx
   (replace [ServerName] with the name of your web server)

2. Go the **Account Info** tab.

Office 365 Configuration Example



3. Select your email system in the Provider dropdown using the instructions provided on the page.
4. Check the box "... utilize AutoDiscover to locate the best Client Access Server for the user..."

> **NOTE:** If you do not check this box, you **MUST** fill in the Url to Exchange Web Services field.

5. Within the Authentication Information section, enter your Integration with Exchange Account User Name and Password. The User Name should be prefixed with your domain (example – YourDomain\Integration with Exchange Account, or Integration with Exchange Account@YourDomain) .

> **TIP:** Make a note of this URL for use later in this topic.

6. (*Optional*) The "Use application pool identity…" option allows you to set the Integration with Exchange Account credentials at the Application Pool level instead of storing the credentials in the EMS database. See the Use Application Pool Identity for Integration for Exchange Service Account topic for more information about this option. If this option is selected, you must check the box to use Impersonation.
7. If you selected "Exchange Web Services" as your Provider, select the checkbox if the account specified has Exchange Impersonation access to all mailboxes in your Exchange mailbox store.
8. Select the Authentication Type:
   - **Anonymous** – No authentication
   - **Specify Account** – Relies on a custom account (not the Integration with Exchange Account) that you create and manage. Please contact Customer Support (or a member of the Professional Services group if you are working with one) to discuss the configuration process for this option.
   - **Default Credentials** – Relies on security context of EMS application calling the Integration with Exchange Web Service. If using this option, Integrated Windows Authentication should be enabled for the Integration with

Exchange Web Service.

» For MS Exchange 2007/2010 environments, click **Save**.

> **NOTE:** When testing Integration with Exchange, the email account that is being used (either on the Test Settings tab or in the "Testing Integration with Exchange" section below) MUST exist in the Exchange environment being tested. If you are testing Integration with Exchange in a development environment please verify that a mailbox for the email being used exists in that domain/environment.

Click **Test Configuration**. If any errors are encountered, please verify your configuration. Otherwise, your Integration with Exchange configuration is complete.

TEST YOUR EXCHANGE INTEGRATION

To test your configuration, you will need to log into EMS Web App with a user account (configured with the user's primary email address) belonging to a Everyday Application Process Template (within the EMS client application) that has the Enable Integration to Microsoft Exchange option checked.

1. Log into EMS Web App. Begin making a reservation and selecting a room.
2. Select the **Add to my calendar** checkbox. If this option is not available, please verify (within the EMS client application) that your user account belongs to a Every-day User Process Template that has the **Allow Invitations** option checked.
3. Find and add an attendee using the Find Attendee field.
4. Complete necessary information on the **Details** tab and click **Submit Reservation**.
5. Verify that an appointment was added to your Outlook Calendar and that your attendee received an invitation.

OPTIONAL MESSAGING SETTINGS

The options on the **Message** tab (as reached above in Step 2) shown below guide you in further configuring your integration.

Message Tab Fields

FIELD	DESCRIPTION
Message To Append	Message appended to the bottom of the appointment body. This message is seen by all users.
To view the details of this	Message added to the appointment body, above a link that takes a user to a view-only EMS Web App page for the

FIELD	DESCRIPTION
reservation click the below link	appointment. This message is seen by all users.
If you are the meeting organizer click the below link to edit the reservation	Message added to the appointment body, above a link that takes the meeting organizer to the EMS Web App Reservation Summary page for that reservation. This message is seen by all users, but only the meeting organizer can access the Reservation Summary page to make changes.
Allow Attachments	Allows users to add attachments within EMS Web App when making an appointment.
Maximum Attachment Size	If attachments are allowed, set the maximum file size allowed for an attachment.

Concept: The default installation allows file attachments up to 4MB. ***Click for more...***

If your implementation needs file attachments that are larger, follow the two procedures below:

1. Update the config file.
2. Update the database.

> **NOTE:** File sizes larger than 2 GB are not allowed at this time.

ENABLE LARGER FILE ATTACHMENTS ON THE CONFIG FILE

By default, Exchange Integration attachments will only accept files 4MB or less. If your implementation needs to allow files of larger sizes to be attached to reservations, the following config updates will be required, both in EMS Web App and in the Exchange Integration Web Service.

> **IMPORTANT:** The maximum file size is 2 GB.

1. In the <system.webServer> section, include this xml node:

   <security>

```
<requestFiltering>

 <requestLimits maxAllowedContentLength="51200000"/> <!--
maxAllowedContentLength in bytes, 50MB=51200000-->

</requestFiltering>

</security>
```

2. In the <httpRuntime element, add these highlighted attributes with the end result looking like this:

```
<httpRuntime targetFramework="4.5" requestLengthDiskThreshold=-
"2147483644" maxRequestLength="51200"  />  <!—-
requestLengthDiskThreshold in bytes, & maxRequestLength in KB,
50MB-->
```

3. Under the <appSettings> look for the “MaximumUploadSizeInBytes” key. Update this value to the number of bytes allowed. For instance, 50MB would look like this:

```
<add key="MaximumUploadSizeInBytes" value="52428800000"/>
<!—- in bytes50MB-->
```

ENABLE LARGER FILE ATTACHMENTS IN THE EXCHANGE INTEGRATION WEB SERVICE

By default Exchange Integration attachments will only accept files 4MB or less. If your implementation needs to allow for Exchange message attachments larger than 4MB, the config updates above will need to be applied in the Exchange Integration Web Service.

> **NOTE:** Due to the size of the xml sent, we recommend adding 5MB to the desired file upload size. (i.e., if you want to allow a max of 20MB files, calculate a total of 25MB worth of Kilobytes and bytes.

In addition to these web.config settings above, a web administrator will need to update the file size in the Exchange Integration Web Service as follows:

1. Navigate to the Exchange Integration Web Service/PAMConfig.aspx.
2. Click the **Message** tab.
3. Update the **Maximum Attachment Size** text box and **Save**.

> **WARNING: FOR EXTERNALLY EXPOSED WEB APP SITES**

If your EMS Web App site is externally exposed, some of the web.config settings above could make the site vulnerable to DoS site attacks. We highly recommend setting network-level protection to prevent DoS attacks.

CHAPTER 5: Use Application Pool Identity for Integration for Exchange Service Account

Rather than entering the Integration for Exchange account credentials on the PAMConfig.aspx page (as in V44 and previous releases), credentials can be maintained at the Application Pool level. This allows your organization to maintain absolute control—**only** IIS applications running in the newly created application pool can run as the Integration to Exchange Account.

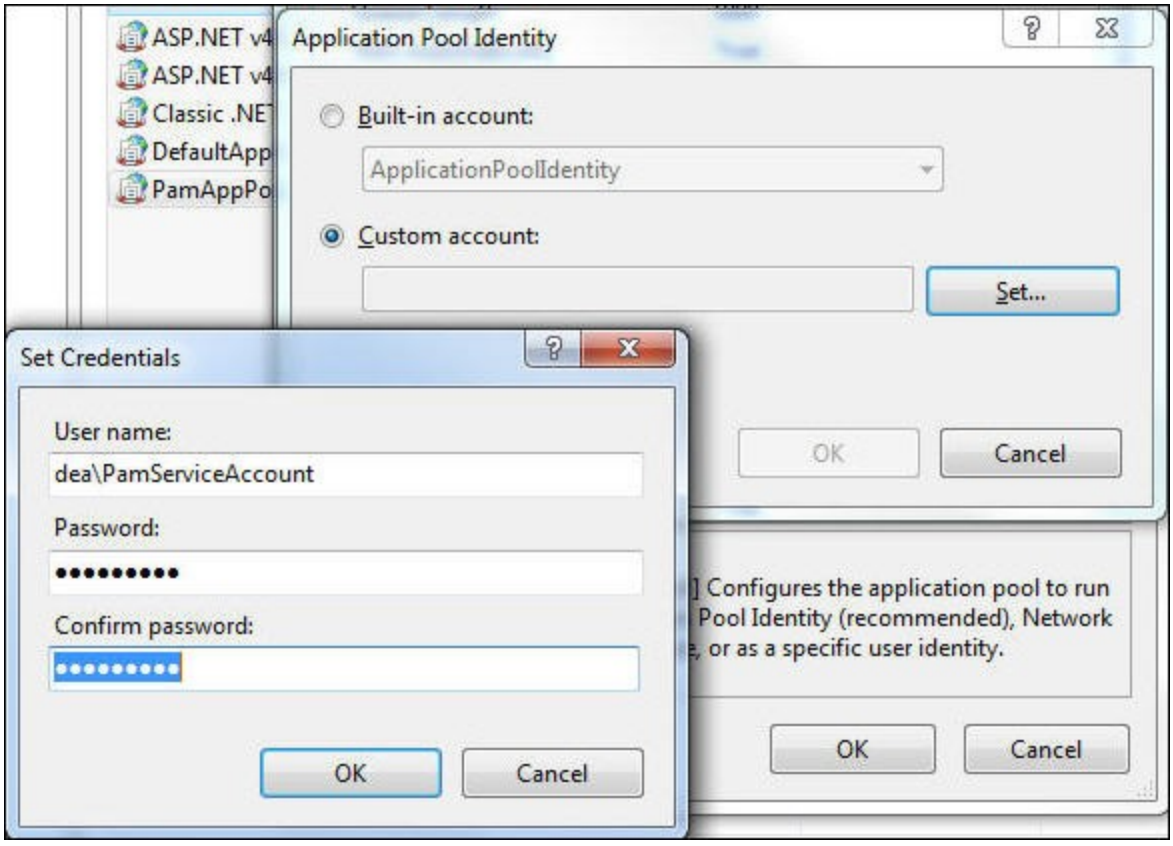
This functionality requires the following:

- Microsoft Exchange 2007 (SP1) or Exchange 2010.
- Microsoft Exchange Impersonation Account (your EMS Integration to Exchange account). This account **must** be using Exchange Web Services (EWS) Impersonation, not full access to the mailbox store.

CONFIGURE THE APPLICATION POOL

1. Open IIS Manager
2. Open the Application Pools panel

3. Click **Add Application Pool…**
4. The Add Application Pool window opens. Enter a unique name and ensure the correct .NET Framework is selected. Managed pipeline mode should be **Integrated**. Click **OK**
5. Find the Application Pool you just created. Right-click it and select **Advanced Settings**.
6. The third section in the list is Process Model. Highlight **Identity** and then click the (**…**) button to configure.
7. Choose **Custom Account** and then click **Set**. Enter the username and password for your EMS Integration to Exchange account. Confirm the password and click **OK** on any remaining dialogs (see following image).



8. Within IIS Manager, navigate to the Virtual Directory containing the Integration for Exchange Web Service. This is under the Default Web Site by default, but may be installed to a different website.
9. With the **IntegrationExchangeWebService** Virtual Directory highlighted in the left pane, select **Basic Settings…** under Actions in the right pane.
10. Click the **Select** button and then choose your newly created application pool from the list.

11. Click **OK** on all remaining dialogs.

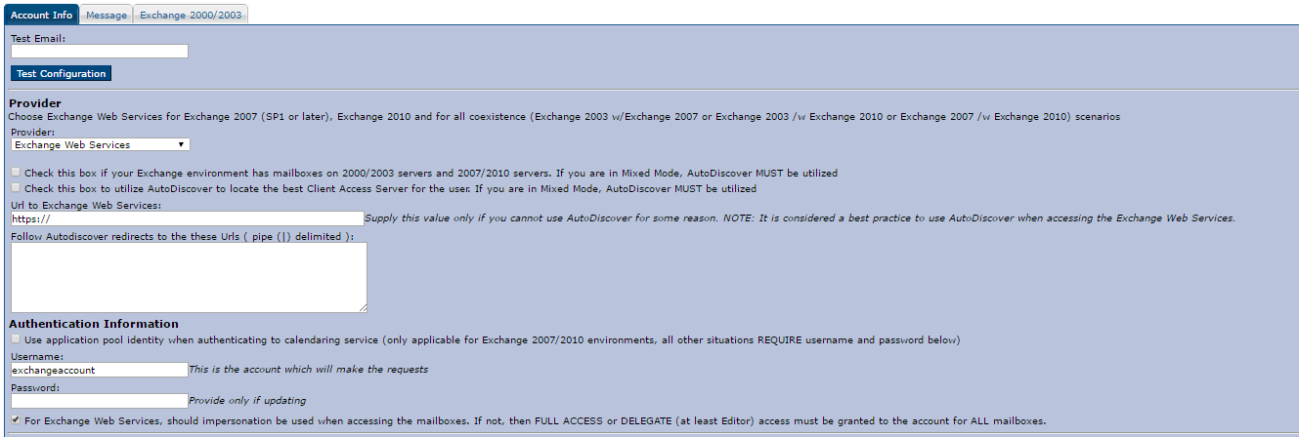
CONFIGURE INTEGRATION FOR EXCHANGE TO USE THE APPLICATION POOL ACCOUNT

1. Navigate to the Integration for Exchange configuration area by opening a browser and entering the following:

   http://[ServerName]/PAMWebService/PAMConfig.aspx (replace [Server-Name] with the name of your web server)

2. From the **Account Info** tab, find the Authentication Information section, check the box for **Use application pool identity when authenticating to calendaring service** (see following image).
3. With this option enabled, you can leave the Username and Password fields blank in the Authentication Information section.

4. Click **Save** button at the bottom of the page.



CHAPTER 6: Configure EWS Impersonation for Microsoft® Exchange

> **TIP:** See Also: What is EWS Impersonation?

1. Log in to the Office 365® Exchange Administration Center.
2. Create a Service Account User within your Office 365 Environment.

   OR

   Configure a already migrated account.
3. Select **Exchange** > **Admin Roles** from the navigation tree.
4. Click the **+** icon to add a new role
5. In the role group dialog box, provide a name for your Role Group (e.g. "EMS_ Exchange_Impersonation"). It is also helpful to enter a Description.
6. Under Role, click the **+** icon to add the "Application Impersonation" Role.
7. Under Members, click the **+** icon and find your Exchange Service Account.

CHAPTER 7: Exchange Web Services (EWS) Impersonation

EMS offers two Exchange integration options to enable seamless room, resource, and attendance scheduling:

1. **EMS Integration to Exchange** offers users the convenience of scheduling rooms, resources, and services, confirming attendee availability, and managing Outlook invitations via EMS Web App (our web-based reservation tool). See Also: Installation Overview.
2. **EMS for Outlook** lets users find available rooms, review their details, reserve them and book any necessary resources (equipment, etc.) without ever leaving Microsoft® Outlook.

To achieve this seamless interaction between everyday users, Outlook hosts, and EMS administrators, an account with Exchange impersonation access to all mailboxes in your Exchange mailbox store is required.

See Also: Configure EWS Impersonation for Microsoft® Exchange

FAQS

Why is this account necessary?

Meetings created via EMS Integration for Exchange either on EMS Web App or EMS for Outlook are owned by the host and associated with a specific Exchange account. That Exchange user can move, update, or cancel the event. However, these meetings can also be moved, changed, or canceled by IT admins and expert users in EMS Desktop Client. When a reservation is moved, changed or canceled in the client, EMS must be able to update the record on the host's Exchange account. Co-ownership of events between the meeting host and the EMS administrators necessitates an account that can read and write to all Exchange accounts being used for booking.

What do we lose if we don't allow impersonation?

Without the impersonation account turned on, you can only make "hanger "reservations from EMS for Outlook: meetings made in EMS for Outlook will be locked in EMS Web App and EMS Desktop Client.

Can we exclude people from impersonation? (For example, remove CEO, Board of Directors, etc. from being impersonated.)

Microsoft Exchange Server supports a CustomRecipientScope parameter when defining the impersonation role. You can define a scope of included users by implementing this parameter.

Is there any way that we could use a delegation feature (like allowing office admins delegate rights) instead of impersonation to notify hosts of updates/changes?

Delegation is possible, here are some things you should know:

- The account needs Editor w/Folder owner (so a custom rights set).
- Custom rights, at least through exchange 2010, are not scriptable. This means the delegation account will get set to owner, which is the only built in (read scriptable) option that has all the necessary permissions.
- EMS for Outlook creates a custom property on the Calendar folder, which allows you to programmatically search the folder for items that have the custom property. Once that custom property is created, then Editor will be enough. It is the creation of the custom property at the folder level that requires owner permission.
- While you can use PowerShell to script the permissions and loop through the users and set the permissions (owner), you would need to make sure that the script got applied to any new users and reapplied to any users that have changed the permissions of the delegation account

- Rights are granted to ANY mail client (Outlook, OWA, etc): when using the impersonation account, rights are only granted to Exchange Web Services, so nobody could type in the service account into Outlook and gain the same permissions.
- These rights are visible to the end user. For example, if an account, "EMSExchangeAccount", has been granted, delegation rights (any level) to User1's calendar, and User1 goes to the Permissions tab of his calendar, he will see the EMSExchangeAccount and the rights it is assigned. Additionally, User1 would be able to change the rights, which would essentially disable the Exchange integration.
- This restricts access only to the calendar

By contrast, EWS impersonation provides the following alternatives to delegation:

- Allows access ONLY through Exchange Web Services
- Does grant permission to do anything the impersonated user could do (assuming it is available as part of EWS)
- End users do not see (and cannot change) the permissions

ADDITIONAL READING

The links below provide additional information from Microsoft® about Exchange Web Services (EWS).

- The Importance of EWS Impersonation
- Authentication and EWS in Exchange
- Impersonation and EWS in Exchange

With Impersonation, a service account has full access to a defined set of mailboxes. What it can access in those mailboxes (such as specific folders) cannot be filtered or defined. Only an Exchange Admin can configure an EWS Impersonation account for impersonation and configure its mailboxes to allow the impersonation.

- Delegate Access and EWS in Exchange

Delegate access allows a user to access certain folders in another user's mailbox Delegate permissions can be set by a mailbox owner or administrator using an app or other app code.