



Act! Web v25.1 Administrator's Guide

© 2023 Act! LLC. All Rights Reserved.

Act!, and the Act! product and service names mentioned herein are registered trademarks or trademarks of ACT! LLC, or its affiliated entities.

Microsoft® SQL Server® and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or in other countries.
All other trademarks are the property of their respective owners.

Released 06/2023 for Act! V25.1

Important Note

Review system requirements [here](#). You must purchase one license of Act! per user. Scalability varies based on hardware, size, and usage of your database. Visit the [Act! Apps Shop](#) or contact your add-on product provider to determine compatibility for your add-on products.

This material may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or computer-readable form without prior consent in writing from Act! LLC, 8800 N. Gainey Center Drive, Suite 200, Scottsdale, Arizona 85258 ATTN: Legal Department.

ALL EXAMPLES WITH NAMES, COMPANY NAMES, OR COMPANIES THAT APPEAR IN THIS MANUAL ARE FICTIONAL AND DO NOT REFER TO OR PORTRAY IN NAME OR SUBSTANCE ANY ACTUAL NAMES, COMPANIES, ENTITIES, OR INSTITUTIONS. ANY RESEMBLANCE TO ANY REAL PERSON, COMPANY, ENTITY, OR INSTITUTION IS PURELY COINCIDENTAL.

Every effort has been made to ensure the accuracy of this material. However, Act! LLC makes no warranties with respect to this documentation and disclaims any implied warranties of merchantability and fitness for a particular purpose. Act! LLC shall not be liable for any errors or for incidental or consequential damages in connection with the furnishing, performance, or use of this material or the examples herein. The information in this material is subject to change without notice.

End User License Agreement

This product is protected by an End User License Agreement. To view the agreement, go to the Help menu in the product, click About Act!, and then click the View End User License Agreement link.

Published by

Act! LLC

621 17th Street, Suite 500

Denver, CO 80293

10 9 8 7 6 5 4 3 2 1

Contents

Introduction	i
Getting Started	1
Planning Your Installation	1
Using Application Pools	2
Installing in Single- or Multiple-Server Configurations	2
Using Secure Sockets Layer (SSL) with Act! (Web)	3
Using Act! (Web) in a 64-bit Environment	4
Installation and Configuration Checklist	4
Setting Up Act! (Web)	5
Understanding the ASP.NET Process and Impersonation User Accounts	5
Creating a Windows User Account for ASP.NET Impersonation	5
Using the Web Site Administration Tool	6
Changing the Session Timeout Limit	8
How Users Access Act! (Web) and Act! Mobile	8
Access from Inside a Common Network	8
Access from Outside a Common Network	8
About User Preferences	8
Configuring Supported Browsers (Web Users)	10
Integrating With Other Applications	11
Editing the Outlook Service Web.config File to Use a Secured Service	11
Editing the Web.config File for Outlook Integration	12
Installing a Mail Merge Component to Use Microsoft Word	13
Setting Up Integration with Outlook or Google	13
Act! Webhooks	15
What are Act.WebHooks.Notifications (Act! Webhooks)?	15
Recycling an Application Pool	20
Troubleshooting Act! (Web)	21
Index	28

Introduction

Act! helps organize all your prospect and customer details in one place so you can prioritize your day and market your products and services more effectively, driving sales results while creating customers for life.

Act! lets users access the database using a supported browser. It is important that you review the full system requirements and supported compatibility list [here](#).

Whether you are installing Act! for the first time or upgrading from an earlier version, this guide provides the information to:

- Plan and organize your installation.
- Set up user access.
- Set up integration with supported applications, such as Microsoft® Outlook® or Google®.
- Troubleshoot the software.



Upgrade Web User. When you see this icon throughout the guide, it indicates information that is applicable to you.

IMPORTANT INFORMATION TO KNOW BEFORE YOU START

Use the following information to determine what guides to use when installing Act!.

If your current version is...	Then...
I do not have any version of Act! installed. I am a "New User."	<ol style="list-style-type: none">1. Use the Installing Act! for New Users guide to install, register, and activate Act!.2. Use the <i>Act! Web Administrator's Guide</i> to configure access via the web.
I have Act! v23 or below installed. My installation may or may not include the web.	<ol style="list-style-type: none">1. Contact Act! Technical Support. Upgrade assistance may be required due to Microsoft obsolescence policies.
I have Act! v24 installed. My installation may or may not include the web.	<ol style="list-style-type: none">1. How do I upgrade to Act! v25 from a previous version article.2. Use the <i>Act! Web Administrator's Guide</i> to configure access via the web.

The guides are available on the installation panel under Product Documentation. Adobe® Reader® is required to access and view product documentation.

Getting Started

In this section, you will learn about:

- Planning your installation, including common configuration suggestions.
- Using Application pools.
- Installing Act! (Web) in single- or multiple-server configurations.
- Using Secure Sockets Layer (SSL) with Act! (Web).

This section also includes an installation and configuration checklist.

Planning Your Installation

Act! (Web) requires Microsoft Internet Information Services (IIS) and ASP.NET to function. See Microsoft documentation to help you plan your installation and prepare your web server. You need to understand your operating system, IIS, ASP.NET, and user permissions required to run a web page.

To determine whether you will install Act! and the database together on one server or separate servers, consider the following:

- The number of users who require access to the database and the number of users who will access the Act! web site. For recommendations, review the system requirements at [here](#).
- How frequently this software and other services will access the server.
- Whether users need access from both inside and outside your network. If access is required from outside the network, two possible server/database configurations are:
 - Install using a single-server configuration, where the web server also hosts Act! databases.
 - Install using a multiple-server configuration, where you have a web server hosting Act! and Internet Information Services (IIS), and a database server hosting the Act! database (including the Microsoft® SQL Server® software to manage the database). You then connect to the database from the web server. (Recommended for more than 30 users.)

Configuration Suggestions

The following are suggestions for configuring Act! (Web). Actual results may vary depending on the size of your database and network configuration.

- Host the application server on a dedicated instance of Microsoft SQL Server.
- The database server should have 3 GB memory for every 50 users in the database. The hard drive speed is most important on this computer. With faster hard drive speeds, users may experience better performance. This computer's performance can benefit if you use 64-bit SQL Server and have quad core processors.
- The application server should have 3 GB memory for every 50 users in the database. CPU and memory are most important on this computer. With more memory, users may experience better performance.
- If possible, distribute mail merge and reporting tasks. On the client machine, mail merge can be the most performance consuming task. On the database server, reporting can be intensive. It is recommended that you run reports when users are not logged on, such as at night.
- If possible, put the Sync server on an out-facing machine and not on the same machine as the database. Internet sync is recommended because it is already running IIS, but you can have both Internet and Network sync set up to act as a safeguard or as a way to balance load. See "About Database Synchronization" in Help for more information about Internet and Network Sync.

Using Application Pools

An Internet Information Services (IIS) application pool is used to isolate ASP.NET web applications and help increase the stability and security of the applications. An application pool can include one or more applications. You can configure the level of isolation between them. For example, you can create an application pool for each application running on the same computer. If errors occur in one pool, they will not affect applications running in other pools. By default, the installation creates an application pool named ACT. To create a custom application pool for Act!, see this Knowledgebase Article for instructions.

Each application pool has one or more *worker processes*. A worker process is a process that runs the web site. The worker processes are unique to the application pool and are not shared across application pools. A worker process is responsible for handling requests sent to a web server for a specific application pool.

Application Pool Recycling

Application pools can be periodically recycled to avoid unstable states that can lead to application crashes, hangs, or memory leaks. By default, Microsoft sets application pool worker processes to recycle every 29 hours or 1,740 minutes.

Note: We recommend that you reset the default recycling interval for the application pool assigned to Act! to recycle every 24 hours or 1,440 minutes. To do this, see ["Changing the Default Recycling Interval for an Application Pool" on page 20](#).

If desired, you can perform an on-demand recycle. An on-demand recycle is recommended after web users have installed and/or configured integration with Google or Outlook. For steps on how to perform an on-demand recycle, see ["Recycling an Application Pool" on page 20](#).

For more information about application pools, see Microsoft documentation.

Installing in Single- or Multiple-Server Configurations

The following explains the differences and recommendations for installing Act! (Web) in single- or multiple-server configurations.

Single-Server Configurations

You can install Act! (Web) using a single-server configuration, where the web server also hosts the database.

Note: If you are also using Act! (Desktop) version, you cannot install Act! (Web) on the same server.

To deploy Act! (Web) in a single-server configuration, the impersonation user account you create should:

- Be added to the local Administrators group.
- Have permission to access the Act! database and database supplemental files.

Multiple-Server Configurations

For scalability and enhanced performance, you can configure Act! (Web) to use multiple servers. The most common types of server configurations used are:

- One web server and one database server.
- Multiple web servers (with or without a database server).
- One or more web servers running multiple application pools. For more information about managing application pools in IIS, see Microsoft Help.

For a multiple-server configuration, the impersonation user account you create should:

- Be a standard domain user.
- Be added to the local Administrators group on all servers that are part of the Act! (Web) deployment.
- Have permission to access the database supplemental files for your Act! database on the database server.
- Have permission to access all Act! files on the web server.

You may have two or more instances of Act! (Web) hosted on one web server. In such cases, you can use virtual directories or application pools to distinguish and secure each Act! (Web) web site.

Multiple Virtual Directories

You can create multiple virtual directories within your default web site. For example, you may have two directories — one called Sales and one called Service. Users access the sites using a URL followed by the virtual directory name (for example, <http://domain.com/sales>). You point each virtual directory to the Act! (Web) installation directory. The installation directory differs for your operating system.

Recommendations When Installing in Multiple-Server Configurations

The following are recommendations when installing Act! (Web) in a multiple-server configuration:

- If you have more than 50 users on a single web server, use multiple application pools.
- You may have two or more instances of Act! (Web) hosted on one web server. In such cases, you can use virtual directories or application pools to distinguish and secure each Act! (Web) site. You should also isolate any supplemental data for each web site.
- Disable all unnecessary Windows services on the web server, such as Network News Transfer Protocol (NNTP) or File Transfer Protocol (FTP).
- Allow access through the firewall only to the web server. Do not allow access to the database server. See your firewall documentation for instructions.
- When using more than one web server, you can set user preferences to be shared across multiple web servers. For more information, see ["About User Preferences" on page 8](#).

Using Secure Sockets Layer (SSL) with Act! (Web)

Secure Sockets Layer (SSL) uses an encryption key to encrypt data between a web server and client so that others cannot access it. You may want to configure SSL in your version of IIS if you plan to integrate Act! with another application, such as Outlook. With SSL, communications between Act! (Web) and Outlook are secure. In IIS, you specify which portions of your Default Web Site to secure with SSL. You can secure the entire Default Web Site or only a specific virtual directory or page.

Consider the following scenarios:

- If your web server only hosts Act! (Web), you can use SSL to secure the entire Default Web Site. The Act! logon page and all child pages will be secured with SSL. You must precede all URLs with "https" rather than "http."
- If other web pages are hosted on your web server, you can use SSL to secure only the virtual directory where the Act! software logon page is set up. The logon page and all pages within Act! (Web) will be secured with SSL. You must precede all URLs with "https" rather than "http."

Best practices if you plan to use SSL with Act! (Web):

- Add an exception for port 443 (the default port for SSL).
- If you plan to integrate with another application, such as Outlook, open the Web.config located at \\<install folder>\ApfwoutlookSvc\ and comment out the UnSecured section and uncomment the Secured one. See ["Editing the Outlook Service Web.config File to Use a Secured Service" on page 11](#).
- Have caching of encrypted pages/files turned On. Users accessing the web site will need to check their browser settings to allow caching. See ["Browser Requirements for Act! \(Web\) Users" on page 10](#).

To configure SSL on IIS, see Microsoft documentation.

Installation and Configuration Checklist

The following checklist outlines the tasks to install and configure Act! (Web).

✓	Task	Page #
	Install or upgrade Act! using one of the following guides: <ul style="list-style-type: none">• Installing Act! for New Users• <i>How do I upgrade to Act! v25 from a previous version article</i>	
	Determine whether you will use a single-server or multiple-server configuration.	2
	Create a Windows user account for ASP.NET impersonation.	5
	Use the Web Site Administration tool to configure the web site. Provide users with access information.	6
	Set up shared preferences across web servers.	9
	Have users configure supported browsers.	10
	Have users integrate Act! (Web) with other applications. Important: Be sure to recycle the application pool after users have installed integration components.	11

Setting Up Act! (Web)

Administrators must perform several steps to set up and configure Act! (Web). Users must configure a supported browser and get the web address URL from the Administrator.

In this section, Administrators will learn how to:

- Understand the ASP.Net process and impersonation user accounts
- Create a Windows user account for ASP.NET impersonation.



Upgrade Web User. You will have a Windows user account for ASP.NET impersonation already set up for your existing web site. This impersonation user should not be affected during upgrade or previous version uninstall. You may want to test the Windows user account after upgrading.

- Use the Web Site Administration tool to set up access to Act! (Web).
- Provide access to the software from inside and outside a common network.
- Provide access to the software from a mobile device.
- Set up shared user preferences across web servers (optional).
- Instruct users how to configure supported browsers to allow access to Act!.

Understanding the ASP.NET Process and Impersonation User Accounts

An ASP.NET process account is required for ASP.NET to function. It is usually managed by Windows®, and it requires the permission: Impersonate a client after authentication. The default account name varies, depending on the operating system on your web server. The Web Site Administration tool lets you test the ASP.NET process account to verify it exists and is properly configured. For more information about configuring and troubleshooting the ASP.NET process account and service, see the Knowledgebase at <https://help.act.com/>.

The Act! (Web) software uses ASP.NET to access the database and other files. You create a Windows user account, and then use the Web Site Administration tool to configure the software to use that account. An impersonation account is required to let a user who does not have access to the network impersonate a user who does. This gives an ASP.NET application access to Windows resources using the account you provide.



Upgrade Web User. You may not need to create a new impersonation user when upgrading Act!. The impersonation user assigned in your previous version should be retained and usable after upgrading the software.

Creating a Windows User Account for ASP.NET Impersonation

The following section explains how an Administrator creates a Windows user account for ASP.NET impersonation. In a multiple-server configuration, the impersonation user account should:

- Be a standard domain user.
- Be added to the local Administrators group on all servers that are part of the Act! (Web) deployment.

Note: Depending on your operating system, the following steps may differ.

Create a Windows user account for ASP.NET impersonation

1. To create a new user account, open **Local Users and Groups**. Specific steps to do this depend on your operating system.
2. Expand **Local Users and Groups**, and then click the **Users** folder.
3. Use your operating system's menus to create a new user.

- a. Type a **User name** (this would be the user's logon name, for example, "actimpersonator") and a **Full name** (this can be the same as the User name).
 - b. Type a password and confirm it.
 - c. Clear the **User must change password at next logon** check box.
 - d. Click **Create**. Then, click **Close**.
 - e. Write down the User name and password because you will need it later when you use the Web Site Administration tool to test the impersonation user.
- Next, add the new user to a group that has access permissions. For the impersonation account, add it to the Administrators group.
4. Right-click on the name of the user you created. Click **Properties**. (The step might differ for your operating system.)
 - a. In the **Properties** dialog box, click the **Member Of** tab. Click **Add**.
 - b. In the **Select Groups** dialog box, in the **Enter the object names to select** box, type the word **Administrators**.
 - c. Click **Check Names** to resolve the Administrator's group. (The group name will appear underlined.)
 - d. Click **OK**.
 5. Close all dialog boxes.

Using the Web Site Administration Tool

Use the Web Site Administration tool to:

- Validate that an ASP.NET process account exists and is correctly configured on the web server.
- Specify and test the Windows user account for ASP.NET impersonation.
- Assign an Act! database to a virtual directory.

You do not need to exit the tool as you move through the tabs to complete the setup. For more information about using the features in the Web Site Administration tool, see "About the Web Site Administration Tool" in Help.



Upgrade Web User. If you removed the database from your Act! web site before upgrading, you will need to assign the upgraded database to the web site so that users can access it.

For the first task, an Administrator must validate that an ASP.NET process account exists and is correctly configured on the web server.

Note: To access this feature, you must be running Windows as an administrator or disable User Access Control (UAC). See your operating system's help for how to do these tasks first.

Test the ASP.NET process account

1. On the **Tools** menu, click **Web Site Administration**.
 2. On the **Web Server** tab, click **Test**. A message appears.
 - If the test is successful, click **OK**.
 - If the test is not successful, an error message appears. This may indicate that an ASP.NET process account does not exist or is incorrectly configured. Click **OK** to close the message. For help, see ["Troubleshooting Act! \(Web\)" on page 21](#).
- Correct any errors and repeat step 2 until the test is successful. The test must be successful before you can access the User Account tab.

The second task is to specify and test the Windows user account for ASP.NET impersonation.



Upgrade Web User. Your existing Windows user account for use in ASP.NET impersonation that you set up for your Web version should be available after installation. You will not need to click Edit or make changes. Proceed to the test step. If the test is not successful, you may have to create a new Windows user account for ASP.NET impersonation.

Specify and test the Windows user account for ASP.NET impersonation

1. Click the **User Account** tab.
2. Click **Edit**.
 - a. From the **Windows Domain** list, select the domain for your impersonation user. (Your domain may already appear in the list.) If you receive an error message that the Web Site Administration tool is unable to access the Windows domain, the Windows User Name does not exist, or you entered an invalid password, correct the errors, and try again.
 - b. Type the user name and password you set up for the Windows user account. Click **OK**.
3. Click **Test**. A message appears.
 - If the test is successful, click **OK**.
 - If the test is not successful, or the ASP.NET process account does not have the required access to the registry key, an error message appears. This may indicate that the impersonation user does not have the required access to files and folders. Click **OK** to close the message. For help, see ["Troubleshooting Act! \(Web\)" on page 21](#).

If necessary, correct errors and repeat the steps until the test is successful. The test must be successful before you can access the Add/Remove Database tab.

During installation, Act! (Web) creates a default virtual directory called "apfw". Users will need this default virtual directory name and the URL of your web server to access the product.

On the Add/Remove Database tab of the Web Site Administration tool, you assign Act! databases to this virtual directory. Then, users can open their web browser and type in the web address to open the product and log on to an assigned database.

Assign a database to a virtual directory

1. On the **Add/Remove Database** tab, in the **Web site** list, the name **Default Web Site** appears. This means the application is looking in the Default Web Site directory of IIS. If you created a custom web site, select it from the list.

A list of available virtual directories for the selected web site appears in the Virtual directories box.
2. In the **Virtual directories** box, by default, **APFW** appears for Default Web Site. If you have a custom web site and virtual directory, select the virtual directory from the list.
3. To add a database to the virtual directory, click **Add**.
 - a. Browse to the database to associate with your selected virtual directory. Click **OK**.
 - b. When prompted to log on, enter your user name and password. Click **OK**.
 - c. A Test is Successful message should appear. If it does not appear, click **TestDB**.
 - If the test is successful, click **OK**.
 - If the test is not successful, an error message appears. This may indicate that the impersonation user account does not have the required access. Click **OK** to close the message. For help, see ["Troubleshooting Act! \(Web\)" on page 21](#).

Changing the Session Timeout Limit

If desired, you can change the default session timeout limit which controls how long users can be idle and remain logged on to Act! (Web). The default setting is 20 minutes. For security purposes, we recommend a maximum limit of two hours.

Change the session timeout limit

1. On the **Options** tab, from the **Session timeout** list, select a timeout limit. Click **Apply**.
2. If a message appears, click **Yes** to continue.
3. Click **Close**.

Microsoft IIS services are automatically restarted so the new session timeout limit can take effect.

How Users Access Act! (Web)

For users to have access to Act! (Web) using a browser or mobile device, an Administrator must add them to the database and assign them a user role. The number of users is dependent on the purchased licenses and how many users have "Active" status. For more information on adding users, see "Adding, Editing, or Deleting Users" in Help.

After adding the user to the database, provide the web address for Act! (Web). This address consists of the virtual directory name appended to the web site URL.

Access from Inside a Common Network

If the web server and users are on a common network, users can access the software using the network computer name of the web server. In the two examples below, that name is "salesweb" and "teamwebserver."

Examples are:

- `http://salesweb/apfw`, where "salesweb" is your site URL, and "apfw" is the virtual directory name
- `http://teamwebserver/apfw`

Access from Outside a Common Network

Users cannot use the network machine name to access the software from outside the common network. Instead, provide an external IP address or a URL for a registered domain name. In the examples below, that information is "12.129.78.35" or "www.yourcompanyname.com."

Examples are:

- `http://www.xxx.com:8080/apfw`, where "www.xxx.com" is your site domain name and "8080" is the port number used to access the site
- `https://www.xxx.com:443/apfw`, where "www.xxx.com" is your site domain name and "443" is the secured port used to access the site
- `http://12.129.78.35/apfw`, where "12.129.78.35" is the external IP Address
- `http://www.yourcompanyname.com/apfw`

Setting up external Internet access to Act! (Web) may require additional configuration. Refer to Microsoft IIS Help or contact your Internet Service Provider for assistance.

About User Preferences

Act! (Web) stores user preferences in a WebPreferences.xml file on the web server that hosts the product. Preferences let users customize name and salutation settings, change background colors and fonts, create custom spelling check dictionaries, and set other options for how they use Act! (Web). In some cases, you may want to distribute web requests across multiple web servers. If you do this, each web server will have a unique set of stored user preferences. If a user changes a preference when logged onto server 1, that change will not be reflected on server 2.

To make sure users have the same settings, you can set up a shared location to store user preferences. Then, when a user logs on, their user preferences will be the same no matter what server responds to their logon request.

Depending on your operating system, user preferences are stored at:

- \\Windows\System32\inetsrv\ACT\ACTforWeb\Preferences
- \\Users[*Impersonation User*]\AppData\Roaming\ACT\ACTfor Web\Preferences.

Setting Up Shared User Preferences Across Web Servers

The following section explains how to set up a shared location to store user preferences for web users. You would want to do this if you distribute web requests across multiple web servers.

Note: If the shared preference storage location is offline when a web user logs on, the preferences that are stored locally on the user's system are accessed for the duration of the session.

Set up shared user preferences

1. Identify the web servers that host Act! (Web).
2. Verify that the impersonation user is the same for each web server in the cluster.
3. Select a computer that all of the web servers can access (for example, Preference_Server).
 - a. On this computer, create and name a folder to store the shared preferences. (For example, SharedPrefs.)
 - b. Share the folder you created. Grant Change and Read permission to your impersonation user account.
4. On a web server in the cluster, locate and open the **web.config** file for the APFW virtual directory. By default, this file is located at \\Program Files\ACT\ACT for Web\APFW.

Note: You can open the web.config file in Notepad. If you have the Folder Option set to Hide Extensions for known file types, this file appears as "web".

- a. In the web.config file, locate the following Preferences Shared Network Folder entry.

```
...
<!--
Shared network folder to store preferences (cluster setup)Leave this value
empty if single web server
-->
<add key="PreferencesSharedNetworkFolder" value="" />
```

- b. Edit the entry by inserting the UNC path to your shared folder between the quotation marks following value=. For example, if your server name is Preference_Server, and your folder name is SharedPrefs, then you would insert \\Preference_Server\SharedPrefs, as shown below.

```
... <!-- Shared network folder to store preferences (cluster setup) Leave this
value empty if single web server --> <add key="PreferencesSharedNetworkFolder"
value="\\Preference_Server\SharedPrefs" /> ...
```

- c. Save and close the **web.config** file.
5. Repeat Steps 4a-c on the remaining web servers in the cluster. If all web servers in the cluster use the same configuration, you can copy the modified web.config file to the other web servers.

Tip: You can use Windows Distributed File Services (DFS) to replicate one web.config file across all servers. See Windows documentation for help.

Configuring Supported Browsers (Web Users)

To open Act! (Web), users must have a supported web browser. To see supported versions, go to <https://www.act.com/products/system-requirements/>.

Required Browser Settings

For Act! (Web) to function properly, the following settings are required on the client computer.

- Turn On **JavaScript**.
- Turn Off **Popup Blocker**.
- Accept **Cookies** "Always" or "From Visited".
- Turn Off **Private Browsing**.
- Add the Act! (Web) server to the **Trusted Sites** zone(1).

Browser Limitations

Internet Explorer has the following limitations when used with Act!.

- View Source function is not available.
- Right-click shortcut menu is not available

(1) Required for Web Help to open on Windows Server® 2008. The Help file is an uncompiled Web Help system which includes the Mark of the Web (MOTW) setting in each Help topic (.htm). The MOTW setting has the "about:internet" text embedded into each Help topic.

Integrating With Other Applications

You can integrate Act! (Web) with:

- Microsoft Word, Outlook, and Excel®.
- Google.

Integration with Word, Outlook, and Google requires configuration by an Administrator and users. Some of the Administrator tasks must be done before users can set up integration with other applications. Integration with Excel does not require any configuration; however, all Microsoft applications must be installed on the computer before installing Act!.

This section explains the Outlook and Google requirements as well as the tasks for an Administrator and users to set up integration with other applications.

Tasks an Administrator can do before users set up integration are:

- Edit the Outlook Service web.config file to use a secured service if you have configured IIS to Secure Sockets Layer (SSL).
- Edit the web.config file to allow users access to Outlook from outside the network.

Tasks a user can do are:

- Install the Microsoft Word component.
- Install the integration component to use with Outlook integration (email, contacts and calendar sync) or Google integration (contacts and calendar sync and Gmail record history).

After users have installed the integration component and set up their integration preferences with either Outlook or Google, an Administrator can recycle the Act! application pool so that the application refreshes and integration tools and menus are available.

Editing the Outlook Service Web.config File to Use a Secured Service

The following steps explain how to comment out the unsecured section and uncomment the secured section of the web.config file so that you can securely use Outlook integration. You must have already configured IIS to use Secure Sockets Layer (SSL).

Edit the web.config file to use a secured service

1. Navigate to the web.config file located under <install folder>\ApfwoutlookSvc.

The default location is \\Program Files\ACT\Act for Web\ApfwoutlookSvc.

2. Select the web.config file and open it in Notepad.
3. To comment out the unsecured section, locate the following text:

```
<!-- Following is UnSecured Configuration. -->  
<services>
```

4. Comment out <services> so that it looks like this:

```
<!--<services>
```

5. Locate the following text:

```
</services>  
<!-- End of Unsecured Configuration. -->
```

6. Change `</services>` so that it looks like this:
`</services> -->`
7. To uncomment the secured section, locate the following text:
`<!-- Following is SSL Secured Configuration. Please comment Unsecured
Configuration above and un-comment SSL secured configuration below in order to
enable Security over SSL -->`
`<!--<services>`
8. Uncomment `<!--<services>` so that it looks like this:
`<services>`
9. Save the file.
10. Reset IIS.

Editing the Web.config File for Outlook Integration

The following steps explain how to edit the web.config file located under the APFW directory to allow users access to Outlook from outside the network.

Note: You must also configure your firewall to permit access from outside the network. For more information, see your firewall documentation.

Edit the web.config file

1. Navigate to the APFW folder on the server where Act! (Web) is installed. By default, the file is located at `\\Program Files\ACT\Act for Web\APFW`.
2. Select the **web.config** file. Open it in Notepad.
3. Locate the **Act.Web.OutlookWebServiceAddress** parameter, which looks similar to the following:
`<add key="Act.Web.OutlookWebServiceAddress" value="http://
/www.xxx.com:8080/apfw"/>`
4. Replace the text following "value=" with the IP address or domain name for the Act! (Web) site.
5. On the **File** menu, click **Save**.

After editing the web.config file, let users know they can install the integration component.

Installing a Mail Merge Component to Use Microsoft Word

The following steps explain how users can install a mail merge component to use Microsoft Word for creating letters and other documents, and performing mail merges. If users plan to use the Act! Word Processor instead of Word, skip this installation.



Upgrade Web User. To record history and edit Word documents, users must first uninstall the existing Act! (web) integration, and then download and install the current version of the component. Use Programs and Features to uninstall the existing version.

Install the Mail Merge component to use Microsoft Word

1. Close Word.
2. Open Act!.
3. On the **Tools** menu, click **Preferences**.
4. Click the **Word Processor** hyperlink.
5. In the **Word Processor Preferences** dialog box, select **Microsoft Word** from the list.
6. Click the **Download** button.

Note: If the integration component cannot be downloaded, check to see if the Act! site URL is added to the Trusted Sites zone.

7. When prompted to run or save the file, click **Run**. Follow all remaining prompts to install the integration component. Click **Finish**.
8. Click **OK**.

Setting Up Integration with Outlook or Google

Users can integrate Act! (Web) with Microsoft Outlook or Google. Integration allows Act! to add to and use features of Outlook or Google, such as:

- Outlook email including Act! address books (contacts) and Quick Attach for emails.
- Outlook calendar and/or contacts synchronization.
- Google integration including calendar and/or contact synchronization and recording history of Gmail messages to contacts.

Users can install the integration component once to use it with either Outlook or Google.

Note: Users should synchronize their calendar and contacts with either Outlook or Google, but not both. Synchronizing with both applications may cause duplicate data.

The following sections describe the integration requirements and how users can install the integration component.

Outlook Integration Requirements

To set up integration between Act! and Outlook, users need:

- A compatible version of Microsoft Outlook. For supported versions, see system requirements at: <https://www.act.com/products/system-requirements/>.
- Outlook installed and working correctly.
- An Outlook profile set up.

Google Integration Requirements

To set up the integration connection between Act! and Google, users need:

- Internet access to a Google account.
- A Google account username and password.
- Authorization for Act! to access the Google account.
- To have logged on to their Google calendar at least once before setting up calendar integration with Act!.
- (Recommended) Google added to their web browser's Trusted Sites list.

- .NET Framework. If .NET Framework is not detected, it will be installed with the Google Integration components.

Installing the Integration Component

Users only need to install the integration component once to use it with either Outlook or Google.

Install the integration component

1. Close the application to integrate with. For example, if users plan to integrate with Outlook, close it.
2. Open Act! (Web).
3. Depending on the task users are attempting, have them do one of the following:
 - If users are setting up Outlook email: On the **Tools** menu, click **Preferences**. Click the **E-mail** hyperlink. Then, in the **E-mail Integration** box, click the **Download** button.
 - If users are setting up Outlook contacts/calendar sync: On the **Tools** menu, click **Preferences**. Click the **Outlook Synchronization** hyperlink.
 - If users are setting up Google integration: On the **Tools** menu, point to **Integrate with Google**, and then click **Google Integration Preferences**.

Note: If the integration component cannot be downloaded, check to see if the Act! (Web) site URL is added to the Trusted Sites zone.

4. If a message appears, read it and click **Install**.
5. When prompted to run or save the file, click **Run**. Follow all remaining prompts to install the integration component. Click **Finish**.

Important: After the first user downloads and installs the integration component, the Administrator must now reset IIS on the web server to complete Google or Outlook integration setup. Any subsequent downloads and installs of the integration component by other web users do not require an IIS reset.

6. If users are setting up Outlook contacts/calendar sync and have installed the integration component using the Outlook Synchronization hyperlink, users must then install the APFWOutlookDetect add-on.
 - a. Click the browser information bar.

If the browser information bar does not display, go to the Tools menu and click Preferences. Click No when prompted to install additional software.
 - b. Click **Run Add-on**.
 - c. When the security warning appears, click **Run**.

Next Steps

- Open Act! (Web) and set up email and calendar/contacts synchronization with Outlook or integration with Google. See Help for steps on how to do this.
- An Administrator must recycle the application pool so that integration tools and menus are available in Act!.

Increasing size limit of emails that can be attached to a database

The following steps explain how to increase the size limit of emails that can be attached to an Act! (Web) database.

Note: This information is provided as a convenience. Act! LLC does not provide support for Microsoft products. If you require assistance, contact Microsoft or an IT professional.

1. On your web server, ensure that you have the runtime component for Web Services Enhancements (WSE) 2.0 SP3 for Microsoft .NET installed.

Note: If you do not have this installed, you can download that by going to this link: [Web Services Enhancements \(WSE\) 2.0 SP3 for Microsoft .NET](#)

2. Browse to the APFWOutlookSrc folder

Note: By default, this folder is located C:\Program Files\ACT\Act for Web\APFWOutlookSrc

3. Locate the web.config file and open it in a text editor
4. Go to the <configSections> line
5. Uncomment, or add if not present, the following line:

```
<section name="microsoft.web.services2"
type="Microsoft.Web.Services2.Configuration.WebServicesConfiguration, Microsoft.Web.Services2,
Version=2.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" />
```

6. Below the <configuration> line, add the following section:

```
<microsoft.web.services2>
<messaging>
<maxRequestLength>-1</maxRequestLength>
</messaging>
</microsoft.web.services2>
```

7. Save and close the Web.config file
8. Restart IIS

Act! Webhooks

What are Act.WebHooks.Notifications (Act! Webhooks)?

Act.WebHook.Notifications is a simple service library built to perform user-defined HTTP callbacks. These callbacks can be created, modified, and managed using the Act! Web API, supported in API v1.0.375 or greater.

How do I get started?

First, install and configure Act.WebHook.Notification service. You can install this service using the installer tool. This is a command-line utility that allows you to install and uninstall server resources by executing the installer components in specified assemblies. The tool is located in the C:\Windows\Microsoft.NET\Framework64\v4.0.30319 directory.

Note: If you don't have Visual Studio development command prompt installed you might need to add InstallUtil.exe to your environment variable path so you can use the default command prompt. You can also copy the three files: InstallUtil.exe, InstallUtil.exe.config, and InstallUtilLib.dll, to C:\Program Files (x86)\ACT\act.webhook.notifications\bin.

Tenant file:

This is for cloud configuration only. Create this file if you want to the web-hooks service to manage multiple Act! instances or tenant, otherwise do not create this file. This file consists of the Act! virtual directory and the API symmetric key (BearerKey). The file is just a name value pair delimited by a colon. The name is the virtual directory located in the Act! admin.xml and the value is the symmetric key located in the Act.Web.Api web configuration file (AppSetting: BearerKey).

APFW:	0F6TijJ4EIY4bjk_Cv51WRW4I7phvMhpHJDu8NX-nVRKVWCnGyvYKcF5jypDbG6yaFOMmPChHDck6x85QdOhJg
Tenant01:	JxoW2It5jujWxmFvvBjqZlffC0UBFSSL_xtc2JgYpkQiGzH5j4ZIORNhgV-BG6ozKw5rya100Nr3903FT_lwSg
Tenant02:	jlrOu5785t3wJPAEk06F-1GkyvqAMQWYh5QJkktXELocsdzsEJkYac62VB0M2atKBeybyFlk_Ruv6x2e8IPXzw

Installation | Uninstallation:

Installation:	C:\>installutil -i act.webhook.notification.exe
Uninstallation:	C:\>installutil -u act.webhooknotification.exe

Configuration:

The Act.Webhook.Notifications service has an act.webhook.notifications.exe.config file. In this file there is an <appSettings> section with a few configurable settings that can be modified if the default settings do not suit your needs. Each settings is as follows:

IsManualConfigure

This indicator informs the web-hooks service to ignore the default location for the API configuration file(s) and use the tenant file that was indicated above. This should be set to false until hosting this in a multi-tenant configuration (one web-hooks service to many Act! instances).

BearerKey

This allows web-hooks to communicate with the Act! API. This will automatically be setup when the service is started, however if the API BearerKey is modified in the Act! API, you will need to update this key by stopping and restarting the web-hooks service. **Note:** Do not manually modify this setting.

PollingIntervalSeconds

The service using Long-Polling to broadcast notifications, this defines the number of seconds elapsed between each polling interval. The default is **900 seconds (15 minutes)**. **60 seconds** is the minimum number of seconds that can be configured.

PollingRetryIntervalSeconds

This defines the number of seconds elapsed between retries. If a notification fails for any reason, the action will be retried. **Note:** retries are loaded into a queue, so this defines when the queue will be processed, not when the notification will be sent (small delay). This setting cannot be less than **30 seconds** or greater than the **polling interval**. Default is **120 seconds (2 minutes)**.

CachingIntervalSeconds

All webhook registrations are maintained in cache to avoid calling the data store every polling interval. However, when new webhook registrations are add through the API this cache will automatically be refreshed. As a backup, this setting interval will force a refresh regardless if the API updates any registrations. default is **7200 seconds (2 hours)**.

AlarmQueueSize

This defines the size of the in-memory queue cache for alarms. The size of this is determined by how many alarms will reside in memory for the set polling interval time. To calculate, determine how many possible alarms that might sound within the polling interval time. Number of users times polling interval. **Note:** memory is freed when the alarm sounds so you don't have to make this to big. The default size is **6000 alarms** cached in memory

RetryAttempts

This is the number of attempts that a failed webhook will be retried before the webhook is suspended. Once a webhook registration has been suspended it will no longer send notifications out, until the registration is unsuspended. This is done through the Act.Web.Api. The default retry attempts is **3**.

ThreadParallelism

This setting controls the number of threads used when sending notifications and can be configured based on the servers configuration and performance levels. Normally, this setting should be configured to be two times the number of processors on the server, however take into account how many other applications are running on the system. To be safe, we can default this to **4 threads**, but you can optimize this setting to what works best for your server. **Note:** Recommend this setting remain default, unless you understand threading.

Tenants

Defines the full path and file name of the tenant file created. If this file doesn't exist, the web-hooks service will default to APFW. This tenant file is only needed if the web-hooks service will manage 2 or more tenant API's.

Note: Remember if using this file, the IsManualConfigure should be set to true.

CacheNotificationUrl

The Act.Webhook.Notifications service host a small web API so that the Act! API can notify the web-hooks service if the cache needs to be reloaded. This URL can be changed to what ever you wish to host this server, but it must be accessible from the Act! API. The Act! API has this web configuration setting as well (webhookUrl) and they must match. **Note:** You would only need to modify this if the port is already being used or you plan to hosting the Act! API and Act! Web-hooks service on different servers (not supported at this time).

FileAndPathToApiWebConfig

If the Act! API is not located in its default folder (C:\Program Files\ACT\Act.Web.API), it would be required to provide the full file path and file to the Act! API configuration file (web.config).

FileAndPathToActWebApiHostingExe

If the Act! self-hosting service is not located in its default folder (C:\Program Files\ACT\Act.Web.API\bin), it would be required to provide the full file path and file to the Act! API self-hosting service (act.web.api.hosting.exe). **Note:** This is the hosting service not its config file. Should be something like: C:\NonDefaultPath\act.web.api.hosting\bin\act.web.api.hosting.exe

Sample Configuration:

```
<add key="PollingIntervalSeconds" value="900" />
<add key="PollingRetryIntervalSeconds" value="120"/>
<add key="CachingIntervalSeconds" value="7200"/>
<add key="AlarmQueueSize" value="6000"/>
<add key="RetryAttempts" value="3"/>
<add key="ThreadParallelism" value="4"/>
<add key="Tenants" value="C:\Program Files (x86)\ACT\tenants.config" />
<add key="CacheNotificationUrl" value="http://localhost:17531"/>
<add key="FileAndPathToApiWebConfig" value="" />
<add key="FileAndPathToActWebApiHostingExe" value="" />
```

Database

When the Act! WebHook Messenger service starts it will build a web-hooks registration database (ActWebHooks) in the ACT7 SqlServer instance as well as create a new SQL Database Login Account (ACTWEBHOOKS) with limited access. The password for this log-in is auto-generated each time the started (stopping and starting the service will modify the password and update its' and the Act! APIs' connection string in the configuration file(s)).

ConnectionString are added/modified to these files:

C:\Program Files\ACT\act.webhook.notifications\bin\act.webhook.notifications.exe.config

C:\Program Files\ACT\Act.Web.API\web.config

C:\Program Files\ACT\Act.Web.API\bin\act.web.api.hosting.exe.config

WebhooksDBConnectionString

```
Server=localhost\ACT7;Database=ActWebHooks;User Id=ACTWEBHOOKS;Password=XXXXXXXX-XXXX-XXXXXXXX-XXXXXXXXXXXX;" providerName="System.Data.SqlClient" />
```

Testing Webhook Callbacks

To test outside of product, you can use an UI based testing tool to visual see your registered callback responses. It is very simple to use, Navigate to this site <https://webhook.site> (<https://webhook.site>) to create a temporary API endpoint that will listen for any registered callbacks. Once the site is opened, create your webhook registration(s) in the Act! Web API using the unique URL generated by the test site.

API Sample: POST:api/webhooks

```
{
  "monitor": "Contacts",
  "triggerEvent": "Created",
  "queryOption": "$select=fullName",
  "callbackUrl": "https://webhook.site/374380ba-a348-4026-870d-09dec5a79d49",
  "callbackToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ0wNS0zMVQyMDozODowMS41OTI1MTcxWiJ9",
  "description": "Webhook to capture newly created contacts."
}
```

Note: For more information one how to create the webhook registration, refer to the Act! Web API documentation. Open the Act! Web API and click on WebHooks.

Monitoring Log Files

The simplest way to monitor log files on windows is to use powershell's version of Linux tail. Open Powershell and run:

```
Get-Content "C:\Program Files\ACT\Act.Webhook.Notifications\bin\webhook-notifications.log" -Tail 0 -Wait
```

The "-tail 0" ignores all the existing entries in the log file and waits for new entry to arrive. If you want to see what is already in the file, just change the 0 to how many lines from the end of the file you wish to display.

Troubleshooting

All critical exceptions are written to the Windows Event Logs "ACT! Services" under the Applications and Services Log. It also writes more verbose logging to a file defined in the application configuration file. To find more information on how to configure logging, see <https://logging.apache.org/log4net/release/config-examples.html> (<https://logging.apache.org/log4net/release/config-examples.html>)

Warning: Start up failures are not always able to write to the application log files, but you should be able to find more information about these types of failures in the Windows Logs under Application.

The more common error is that the Web hosting can't start because another process is already running. This is not common, but most likely it is because the Act.Web.Api was started before this service and will not release the port. Trying recycling the Act.Web.Api process. If that doesn't work it might require a reboot. However, is extremely rare.

Exceptions and Message Codes

Fatal Exceptions (Range: 1 to 5) :

We recommend that you monitor these events

Event id	Message
1	Symmetric Key Failure: Symmetric key file is corrupted.
2	Symmetric Key Failure: Symmetric key not found.
3	Token Exception: Unable to generate token.
4	Failed to update registration that a successful notification had been sent.
5	Failed to update registration that a failed notification had not been sent.

Task related exceptions (Range: 100 to 400)

These maybe important because they could inactivite a registration after retries are exhausted.

Event id	Message
100	Alarm Sounds Retrieval Failure: Unable to get alarm data.
200	Created Entity Retrieval Failure: Unable to get newly created data.
300	Edited Entity Retrieval Failure: Unable to get recently modified data.
400	Added failed registration to the retry queue.

Database related exceptions and messages (Range: 1000 to 1500)

Event id	Message
1000	Database has not been created.
1001	Unable to access that webhook's registration database.
1500	Database has been created. (Information Message)

Service related messages (Range: 2000 to 2004) :

We recommend that you monitor most of these events (Pause, Shutdown, and Stop).

Event id	Message
2000	OnContinue: Act! Webhook Messaging Service was re-started after a pause.
2001	OnPause: Paused the the Act! Webhook Messaging Service.
2002	OnShutdown: Shutdown the Act! Webhook Messaging Service.

2003	OnStart: Started the Act! Webhook Messaging Service.
2004	OnStop: Stopped the Act! Webhook Messaging Service.

Recycling an Application Pool

An Administrator may have to immediately recycle an application pool instead of waiting for the next configured recycle. Rather than abruptly stopping it, which can cause service interruptions, you can use on-demand recycling. You would do an on-demand recycle after users have installed the Act! Scratchpad and/or installed integration components.

The following steps explain how to use the IIS Manager to recycle an application pool on demand.

Recycle an application pool using the IIS Manager

- To open IIS Manager using the Run dialog box, do one of the following for your operating system:
 - On the **Start** menu, click **All Programs**, click **Accessories**, and then click **Run**.
 - On the **Start** menu, click **Run**.

In the **Open** dialog box, type **inetmgr**. Click **OK**.
- Expand the local computer and click **Application Pools**.
- Select the Act! (Web) application pool. By default, it is ACT. If you have created a custom application pool, select it.
- On the **Actions** menu, or on the right pane, under **Application Pool Tasks**, click **Recycle**. If a confirmation message appears, click **Yes**.
- Close IIS Manager.

If desired, you can change the default recycling interval.

Changing the Default Recycling Interval for an Application Pool

Microsoft sets the default recycling interval at 1,740 minutes (29 hours) to restart an application pool. We recommend you change the interval to 1,440 minutes (24 hours). A shorter recycling interval may help with application crashes and recovery of system resources.

Change the default recycling interval

- To open IIS Manager using the Run dialog box, do one of the following for your operating system:
 - On the **Start** menu, click **All Programs**, click **Accessories**, and then click **Run**.
 - On the **Start** menu, click **Run**.

In the **Open** dialog box, type **inetmgr**. Click **OK**.
- Expand the local computer and click **Application Pools**.
- Do one of the following:
 - Right-click the Act! application pool, and select **Properties**. By default, Act! (Web) uses the ACT application pool. On the **Recycling** tab, for **Recycle worker processes (in minutes)**, change 1740 to **1440**. Click **OK**.
 - On the right pane, under **Edit Application Pool**, click **Recycling**. Under **Fixed Intervals**, for **Regular time intervals**, change 1740 to **1440**. Click **Next**. Click **Finish**.
- Close IIS Manager.

Troubleshooting Act! (Web)

This section helps you identify solutions to:

- Web Site Administration tool errors.
- General configuration issues or errors.

Troubleshooting Web Site Administration Tool Errors

This section describes how to troubleshoot error messages received while configuring your software. When troubleshooting, you may also refer to the Web Site Administration tool log file (AFWAdminLog.log) and the validation log file (validate.txt). These files may provide information about user permissions and errors displayed in the tool, such as ASP.NET and HTTP errors. The Web Site Administration tool must be closed to create an updated log file. Log files are located in: <Act! installation path>\ACTACT for Web\Logs.

Additional information for troubleshooting the Web Site Administration tool is provided in the Act! Knowledgebase.

Tip: To bypass errors in the Web Site Administration tool, hold down the SHIFT or CTRL key and click another tab. For example, you may want to view information on another tab before completing the configuration.

Web Server and User Account Tabs

Microsoft SharePoint software detected. Additional configuration may be required to enable Act! (Web) to run.

Cause:

- SharePoint was detected on the web server. Some installations of SharePoint modify the default web site and ASP.NET service configuration.

Solution:

- You may need to perform additional steps for the Act! (Web) web site and other applications to function. The following commands may help resolve issues that may occur if you install some Microsoft products after installing Act!.

REM - Enable Act! for Web to run on SBS server 2011 which has 64 bit Web apps Exchange and Outlook Web Access installed. It should also apply to situations where Exchange or Outlook has been installed on other versions of server OS before or after installing Act!.

REM - This only applies to Act! (Web) 2013 Service Pack 1 and later. Act! uses a new AppPool separate from the Default App Pool to allow 32 and 64-Bit apps to run on the same server.

REM - Make a backup of the IIS config file.

```
%SystemRoot%\System32\inetsrv\appcmd.exe add backup ACTapphostconfigbackup
```

Microsoft SharePoint® software detected. Additional configuration may be required to enable Act! (Web) to run.

REM - Several modules on a 64-Bit server cause conflicts with 32-Bit web apps. They are each handled differently. Unlock the Static and Dynamic compression modules globally, so that they can be configured individually for each virtual site.

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/modules /[name='DynamicComp
(accelerationModule)'].lockItem:false /commit:apphost
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/modules /
[name='StaticCompressionModule'].lockItem:false /commit:apphost
```

REM - Remove the Static and Dynamic compression modules from each Act! virtual site/virtual directory.

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
dynamiccompressionmodule /app.name:"Default Web Site/APFW"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
staticcompressionmodule /app.name:"Default Web Site/APFW"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
dynamiccompressionmodule /app.name:"Default Web Site/AFWValidationSrcv"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
staticcompressionmodule /app.name:"Default Web Site/AFWValidationSrcv"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
dynamiccompressionmodule /app.name:"Default Web Site/APFWMailMergeSrcv"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
staticcompressionmodule /app.name:"Default Web Site/APFWMailMergeSrcv"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
dynamiccompressionmodule /app.name:"Default Web Site/APFWOutlookSrcv"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe delete module
staticcompressionmodule /app.name:"Default Web Site/APFWOutlookSrcv"
```

REM - Add a pre-Condition for 64-bit only on these modules, so that they will not load with the ACT appPool. These are authentication modules for Exchange Server.

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/globalModules /
[name='PasswordExpiryModule'].preCondition:bitness64 /commit:apphost
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/globalModules /
[name='kerbauth'].preCondition:bitness64 /commit:apphost
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/globalModules /
[name='exppw'].preCondition:bitness64 /commit:apphost
```

REM - Add a pre-Condition for 64-bit only on these isapiFilters. Be sure to include the "Default Web Site" on the end of the command.

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/isapiFilters /["name='Exchange OWA Cookie
Authentication ISAPI Filter'"].preCondition:bitness64 /commit:apphost
"Default Web Site"
```

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -
section:system.webServer/isapiFilters /["name='Exchange ActiveSync ISAPI
Filter'"].preCondition:bitness64 /commit:apphost "Default Web Site"
```

REM - Disable Rapid Fail Protection mode for the ACT appPool. This protection will terminate the appPool if it throws a number of errors. This can cause users to receive a 505/503 Service Unavailable response from the web with no useful information as to why.

Microsoft SharePoint® software detected. Additional configuration may be required to enable Act! (Web) to run.

```
%SystemRoot%\System32\inetsrv\appcmd.exe set config -  
section:system.applicationHost/applicationPools /  
[name='ACT'].failure.rapidFailProtection:False /commit:apphost  
REM - Use the following command to undo the changes made to the IIS config  
file if necessary.  
REM - %SystemRoot%\System32\inetsrv\appcmd.exe restore backup  
ACTapphostconfigbackup
```

The ASP.NET process account does not exist or is not configured properly.

Causes:

- The ASP.NET service is not available.
- Non-default IIS settings, such as an assigned IP address or host headers, are configured.
- Read permissions are not set.

Solutions:

- Verify that the ASP.NET service is available.
- Verify the configuration of the ASP.NET process account.
- Manually configure users and permissions as described in Knowledgebase Article #14867.

User Account Tab

Unable to retrieve Windows domain information at this time. This information is required to edit the Windows User Account. Please check with your network administrator and try again later.

Cause:

- The Windows domain information is unavailable.

Solution:

- See your network administrator to verify that you have access to the domain.

The ASP.NET process does not have permission to access or run Act! (Web).

Cause:

- The ASP.NET process account does not have the permissions on the web server to run Act! (Web).

Solution:

- This account requires Read permission for the registry key that contains the encrypted impersonation user credentials. Verify the correct account for your web server configuration.

The impersonation user account does not have access to files and folders required to run Act! (Web) on your web server.

Cause:

- The Access to program files and folders or registry permissions required to run Act! (Web) has not been granted to the impersonation user account.

Solution:

- Verify that the user is at least a domain user and part of the local Administrators group, and then set the appropriate permissions to the Act! files and folders on the server.

The Windows user account information you have entered is invalid. Please check the user name and password and try again.

Causes:

- The Windows user name or password you entered does not exist in the selected Windows domain.
- The option to force users to change their password at logon has been selected.

Solutions:

- Verify that you entered the correct user name for the selected domain. If you did, you may need to create a user. For more information, see "[Creating a Windows Impersonation User Account](#)" on page 5.
- For the impersonation user account, clear the User must change password at next logon check box.

Add/Remove Databases Tab

The impersonation user account does not have access to the database you selected.

Cause:

- The impersonation user specified on the User Account tab does not have permission to access the program files and folders or registry permissions required to run Act! (Web).

Solution:

- Verify that the user has the appropriate permission to access all Act! files and database supplemental files on the server.

The web page fails to load after selecting a web site and virtual directory.

Causes

- Various.

Solutions:

- If you are testing a web site that was working previously, execute the tests on the Web Server and User Account tabs to determine the cause of the issue. Your web server configuration may have changed, for example:
 - The password for the impersonation user account was changed or has expired.
 - New software or services, such as SharePoint or Active Directory, were installed on the web server.
 - The application pool where the Act! (Web) virtual directory runs was modified.
- If you just created the impersonation user account, you may not have cleared the option to force the user to change the password at the next logon. Clear the User must change password at next logon check box.

The web page fails to load after selecting a web site and virtual directory.

- If you just successfully executed the tests on the Web Server and User Accounts tabs, restart IIS on the web server. See Help for how to do this.

Troubleshooting General Errors and Issues

This section describes how to troubleshoot general issues and errors with regard to installation, configuration, security, and so on.

General Errors and Issues

Act! Help is not opening when you click Help topics or the Help menu.

Reason:

- Act! (Web) Help is a Web Help help system containing the Mark of the Web (MOTW) setting in each Help topic (.htm). On some server operating systems using Internet Explorer, a blank page appears when Help is accessed.

Solution:

- Add the about:internet site to your Trusted Sites zone or your local intranet zone. The about:internet setting refers to the MOTW security setting in each help topic.

You receive the message: "Web site is not running".

Reason:

- IIS may not be running.

Solution:

- Verify that IIS and the default web site are running.

Errors appear on a remote browser, but not while using the browser on the server.

Reasons:

- Pop-up blockers are enabled.
- The web site is not a trusted site on client computers.
- Your Internet security software does not allow access.

Solution:

- Disable pop-up blockers in your browser.
- Add the web site to the list of trusted sites on all client computers.
- Shut down the Internet security software to allow access.

You receive the message: "400 - Bad Request" or "404 - File not found".

Reason:

- The Web Site Administration tool tests the site using "localhost."

Solution:

- If the Act! (Web) site is configured to use an IP address, it will not use "localhost".

You can access the web site internally, but not externally.

Reason:

- Your router may not be configured correctly.

Solution:

- Ensure your router is forwarded to port 80. For more information, see your router documentation.

You receive a message saying "... If you continue to see this message, please recycle your IIS application pool, restart IIS, or contact your Administrator".

Reason:

- You have just installed the Act! Scratchpad or the integration component to use supported applications with Act! (Web) and this message appears.

Solution:

- An Administrator needs to recycle the application pool as it is less disruptive than resetting IIS. See ["Recycling an Application Pool"](#) on page 20.

You receive the message: "403 - Forbidden".

Reason:

- You do not have permission to access a file.

Solutions:

- Ensure the impersonation user is a Windows Administrator.
- If you created a custom virtual directory, ensure the Read and Run Scripts (such as ASP) are enabled. See Knowledgebase Article #14868.

You are accessing Act! (Web) using a secured connection (https://) and settings are not being retained. For example, you are scheduling an activity and as you move through the Schedule Activity tabs, settings are changing back to the defaults.

Reason:

- Your browser may not have the correct setting to allow caching of pages/files when using Secure Sockets Layer (SSL). It is a best practice to have caching of encrypted pages/files turned on when using Act! (Web) over SSL.

Solution:

- For Internet Explorer: Click Tools>Internet Options. Click the Advanced tab. Under Security, clear Do not save encrypted pages/files to disk.

Index

A

- Act! Premium Mobile
 - accessing 8
- Act! Scratchpad
 - installing 12
- Act! Webhooks 15
- ActiveX security 10
- Add/Remove Database tab
 - assigning databases to virtual directories 7
- application pools
 - changing the recycling interval 20
 - overview 2
 - recycling on demand 20
 - using to isolate Web applications 2
- ASP.NET impersonation
 - creating a Windows user account 5
- ASP.NET impersonation user account
 - overview 5
 - testing 6
- ASP.NET user account
 - testing 6

B

- browsers
 - Internet Explorer limitations 10
 - Internet Explorer required settings 10

C

- changing session timeout limit 8
- copying
 - user preferences for ACT! for Web 2005 -2006 users 9
- creating a Windows user account for ASP.NET impersonation 5

D

- databases
 - assigning to virtual directories 7
- default virtual directory 7

E

- e-mail
 - installing components to use Outlook 14

- examples
 - URLs and IP addresses for web site access 8

G

- Google
 - installing integration components 14
 - integration requirements 13

I

- impersonation
 - for single-server configurations 2
 - in multiple-server configurations 2
- impersonation user account
 - creating 5
 - overview 5
 - testing 6
- Increasing size limit of emails that can be attached to a database 14
- installation
 - configuration suggestions 1
 - planning 1
- installing
 - integration components 14
 - mail merge component to use Word 13
 - multiple-server configurations 2
 - single-server configuration 2

- Internet Explorer
 - browser settings 10
 - limitations 10

M

- Mark of the Web (MOTW) setting 10
- mobile
 - accessing Act! Premium Mobile 8
- mobile access
 - examples 8
- multiple-server configurations 2
 - multiple virtual directories 3
 - recommendations 3

O

- Options tab
 - changing the session timeout limit for a web site 8
- Outlook
 - allowing access outside network 12
 - installing integration components 14
 - integration requirements 13
- Outlook integration
 - editing the web.config file when using a secured service 11

P

- planning your installation [1](#)
- preferences
 - for users [8](#)
 - locations of [9](#)
 - sharing across web servers [9](#)

R

- recycling application pools
 - changing the interval [20](#)
 - on demand [20](#)
- requirements
 - Google integration [13](#)
 - Outlook integration [13](#)
 - supported browsers [10](#)

S

- Secure Sockets Layer (SSL)
 - editing the Outlook service web.config file [11](#)
 - overview [3](#)
 - troubleshooting caching pages [27](#)
- setting up
 - Word integration [13](#)
- single-server configuration
 - impersonation user account recommendations [2](#)
- specifying the Windows user account for ASP.NET
 - impersonation [6](#)
- SSL
 - editing the Outlook Service web.config file [11](#)
 - overview [3](#)
 - troubleshooting caching pages [27](#)

T

- testing
 - Windows user account for ASP.NET impersonation [6](#)
- timeout limit
 - changing [8](#)
- troubleshooting
 - general errors and issues [25](#)
 - SSL and caching pages [27](#)
 - Web Site Admin tool errors [21](#)
- Trusted Sites zone [10](#)

U

- User Account tab
 - specifying and testing Windows user account for ASP.NET
 - impersonation [6](#)
- users
 - accessing the web site [8](#)

- adding to databases and assigning roles [8](#)
- allowing access to Outlook (outside network) [12](#)
- installing Act! Scratchpad [12](#)
- installing integration components [14](#)
- preferences for [8](#)

V

- virtual directories
 - assigning databases to [7](#)
 - default for Act! Premium (access via web) [7](#)
 - in multiple-server configurations [3](#)

W

- Web Server tab
 - testing your ASP.NET user account [6](#)
- web site access
 - changing session timeout limit [8](#)
 - changing the session timeout limit [8](#)
 - examples of URLs and IPs [8](#)
 - for users [8](#)
 - from inside common network [8](#)
 - from outside common network [8](#)
 - using a mobile device [8](#)
- web.config file
 - editing for Outlook access [12](#)
- Word
 - installing the mail merge component [13](#)
- word processor
 - installing mail merge component for Word [13](#)