Colleague

Setting Up Colleague Web API

Release 1.12
May 2016

Notices

In preparing and providing this publication, Ellucian is not rendering legal, accounting, or other similar professional services. Ellucian makes no claims that an institution's use of this publication or the software for which it is provided will guarantee compliance with applicable federal or state laws, rules, or regulations. Each organization should seek legal, accounting, and other similar professional services from competent providers of the organization's own choosing.

Ellucian
4375 Fair Lakes Court
Fairfax, VA 22033
United States of America

Contents

Prepare for Colleague Web API.......................... 5
Overview.......................... 5
Business model layer.......................... 6
Data layer.......................... 7
Prepare for the installation.......................... 7
Hardware and software requirements.......................... 7
Obtain the installer.......................... 8
Colleague Web API deployment scenarios.......................... 8
Single web server - non-production environment.......................... 8
Multiple web servers - production environment.......................... 9
Component naming conventions.......................... 10
Web server preparation steps.......................... 11
Windows Server 2012 R2/IIS8.5 and 2012/IIS8: Install the ASP.NET Role Service and the IIS 6 Management Compatibility components.......................... 11
Windows Server 2008 R2/IIS7.5 and 2008/IIS7: Install the ASP.NET Role Service and the IIS 6 Management Compatibility components.......................... 12
Install .NET framework.......................... 13
Register ASP.NET 4.x.......................... 13
Re-register older versions of ASP.NET.......................... 14
Determine which .NET version an application uses.......................... 15
Re-register the correct .NET version for an application.......................... 15
Allow the ASP.NET extension to run on the web server.......................... 16
Limit IIS logging.......................... 16
Install software updates.......................... 17
Install Colleague Web API.......................... 18
Run the Colleague Web API installer.......................... 18
Upgrade Colleague Web API installation and configuration.......................... 20
Uninstall Colleague Web API.......................... 20
Configure and Administer Colleague Web API.......................... 22
Configure web services parameters.......................... 22
Configure Colleague time zone.......................... 22
Colleague Web API performance considerations.......................... 22
Configure the Colleague Web API.......................... 23
Generate machine keys for security.......................... 24
Configure the authentication services for your API site.......................... 25
Configure .NET 4.X for the website.......................... 25
Configure the Colleague connection settings.......................... 25
DAS Data Reader option.......................... 30
Using the DAS Data Reader.......................... 30
DAS Data Reader security considerations.......................... 30
Configure the Web API logging settings.......................... 31
Configure the Web API settings.......................... 31

Photo setup scenario examples....34
Colleague Web API and SSL....35
Security concepts....37
Public and private keys....37
Certificate overview....37
Install an SSL certificate for the website....37
Enter a secure port number and turn on SSL for a new website....38
Colleague Web API security....39
Colleague security....39
Security notifications....39
Enable security notifications....40
Security notification text....40
Disable individual security notifications....41
Re-enable a disabled security notification....41
Remove security notifications....42

Understand Colleague Web API Caching and the Warm Up Script....43

Caching and caching timeouts....43
Run the warm up script....44

Prepare for Colleague Web API

This manual is intended for anyone who installs and administers Ellucian web solutions at your institution. It describes the Colleague Web API, and tells you how to prepare for, install, and configure the software.

Overview

Colleague Web API provides a RESTful web interface to Colleague data and business logic, allowing Ellucian solutions and third-party products to communicate with the Colleague ERP in a consistent manner.

The Colleague Web API infrastructure is comprised of two main component layers: the Business Model and Data layers. This separation of concerns allows for individual enhancement and maintenance of each component, and the ability to scale the components independently of each other as needed.

Figure 1: Architecture

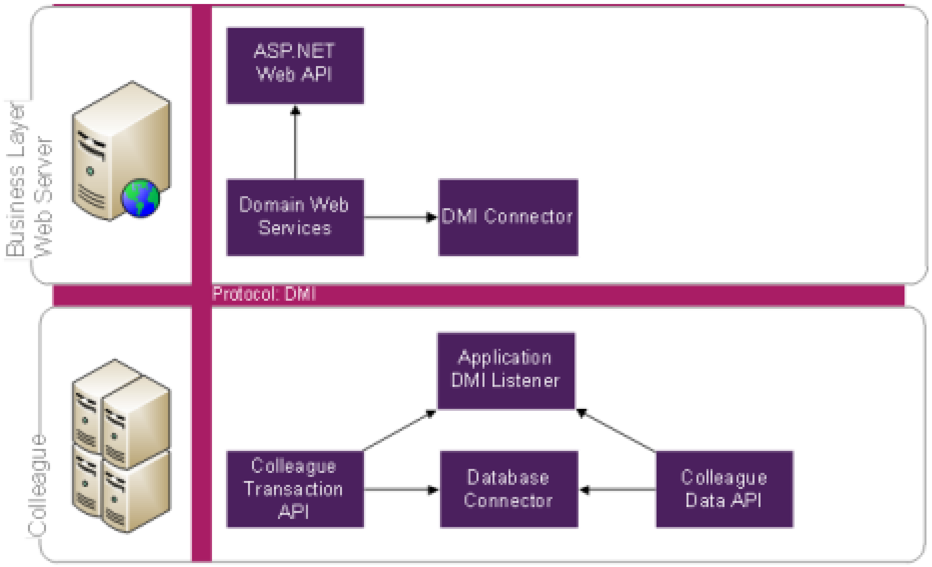
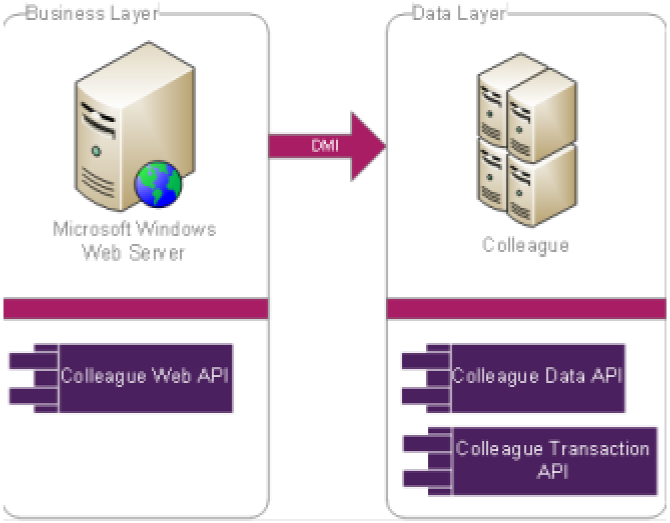


Figure 2: System deployment



Business model layer

The business model layer is composed of a business model web server that contains well-formed, standardized business data models.

Data is retrieved from the Colleague database and is passed to the requesting presentation web servers. This layer leverages Microsoft ASP.NET Web API, which is an extension of ASP.NET MVC. The Web API services communicate using a REST-style format.[1] This layer can include one or more web servers.

Data layer

The data layer is your existing Colleague data infrastructure.

This layer is logically part of the business model web server, yet physically comprised of two additional servers: Colleague Application Server (CAS) and Database Access Server (DAS). The CAS leverages DMI to execute the business logic and data retrieval through the Colleague Transaction API and the Colleague Data API. The data retrieved from Colleague is passed to the business layer.

Prepare for the installation

Review and complete the steps in this section to prepare your environments and servers for Colleague Web API.

Hardware and software requirements

Before you install the Colleague Web API, you must meet the hardware and software requirements and download the Colleague Web API installer.

For Colleague Web API hardware requirements, refer to the *Ellucian Hardware Recommendations – Colleague* manual. For a list of browsers and operating systems supported in the Colleague Web API, refer to the Product Certifications and Cessations page on the client website.

To run the Colleague Web API, each web server must meet the following requirements:

- Internet Information Services (IIS). One of the following:
  - 8.5 (Windows Server 2012 R2)
  - 8 (Windows Server 2012)
  - 7.5 (Windows Server 2008 R2)
  - 7 (Windows Server 2008)

  If you have a later version, note that some of the configuration steps may be different.
- IIS 6 compatibility pack.

The Colleague Web API installation requires:

- NET Framework 4.5.1

[1] REST is the acronym for a web-based architectural style called Representation State Transfer; a common transmission format of REST-style data is JavaScript Object Notation (JSON).

- ASP.NET 4.x

Obtain the installer

The installer can run on either 32-bit or 64-bit servers. The file is named `ColleagueWebAPI`*`xx`*`Setup.exe`, where *xx* is the version number.

Before you begin

Before you download the installer, verify that you have also downloaded and installed the prerequisite software updates.

Procedure

1. Using SA Valet 2.8 or later, download the installer release packages.

   For the procedure, see the “Viewing and Downloading Installer Releases” section of the Updating Colleague Software manual.
2. Save the installer on the web server where you plan to use that component.

Results

The Colleague Self-Service components are 32-bit applications, so they are installed into the x86 program file location on a 64-bit system. For example, `C:\Program Files (x86)\Ellucian \`*`website name`*.

Colleague Web API deployment scenarios

Deployment of the Colleague Web API components will vary based on deployment practices and security setup at your site.

If you have multiple environments that you want to set up, you will need to repeat the installation and configuration process for each component.

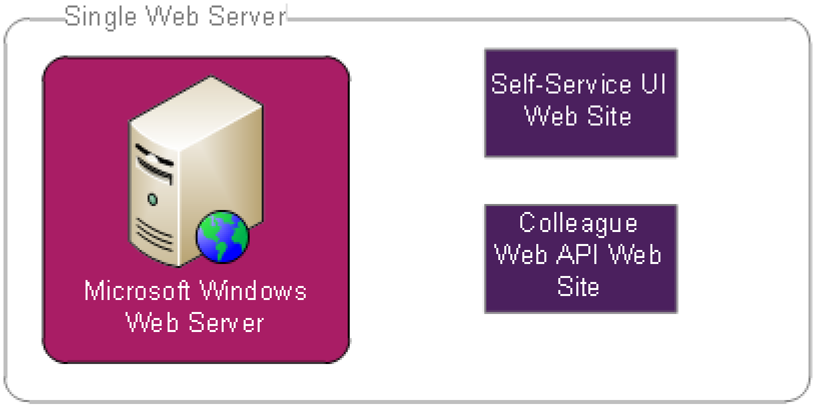
Ellucian recommends using different web servers for each environment, if possible. At the very least, the servers for your production environment should be separate from the servers used for non-production environments.

Each instance of the Colleague Web API should be installed in a different folder with a name that reflects the name of the environment in which the components will be used.

Single web server - non-production environment

For a non-production environment, you can deploy Colleague Web API to a single web server behind the internal network firewall. With the server positioned behind the firewall, Colleague Self-Service and Colleague Web API can be installed on the same server.

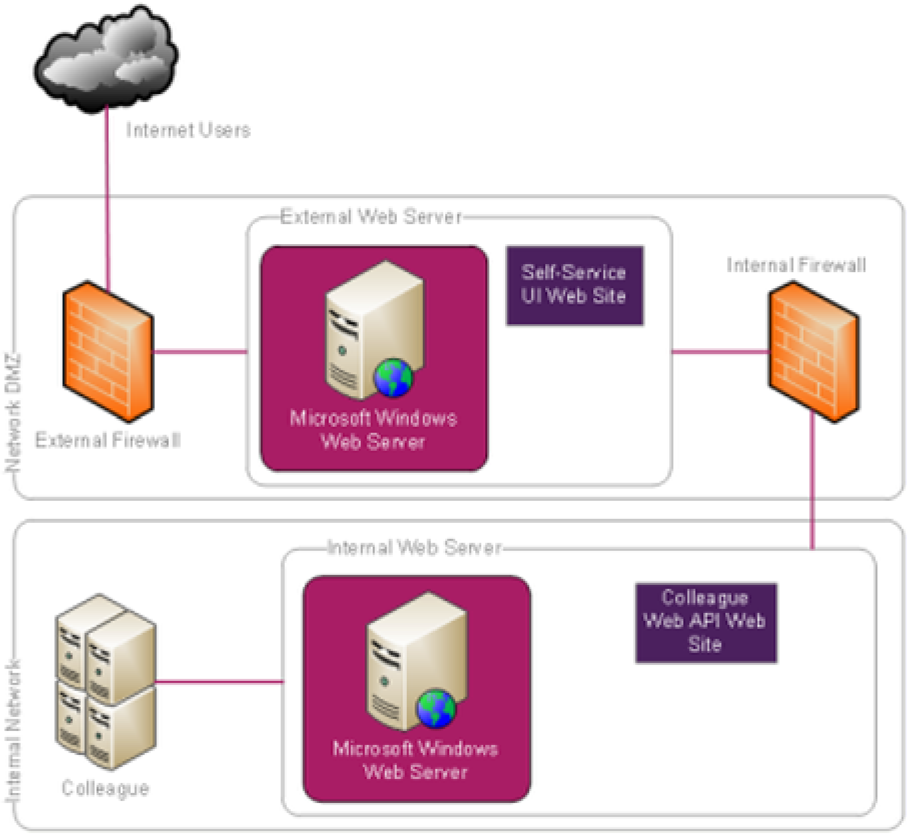
This setup is simpler than deploying multiple web servers, but it does not provide the scalability and security that is necessary for a production application. The figure below depicts the deployment of components on a single web server. In this deployment scenario, the client computer communicates with the web server, and the ASP.NET MVC website communicates with the Colleague Web API on the same machine. Communication with the Colleague Application Server (CAS) is direct because the web server is behind the internal network firewall.



Multiple web servers - production environment

For a production Colleague environment, you should have a web server in the network demilitarized zone (DMZ) for external network access and an internal web server behind the internal network firewall.

The external web server delivers browser content and must be accessible to end users. The internal web server communicates with Colleague and should not be accessible to end users. This setup is a little more complex than deploying a single web server, but it offers the scalability and security that is necessary for a production application. The figure below depicts the deployment of components on multiple web servers. In this deployment scenario, the external web server permits connections from external networks, such as the Internet. Requests from the external web server to the internal web server are routed through the internal network firewall. The region of the network that accepts external connections and filters internal network connections is the network DMZ. This network topology is commonly used for secure server deployment. Because the internal web server is behind the firewall with the Colleague Application Server (CAS), no additional firewall filtering is needed.



Component naming conventions

The web server hosting the Colleague Web API must be able to connect to the application listener port on the Colleague Application Server. Ellucian suggests that you use distinct naming conventions for your environments and websites.

The suffix WebApi (Colleague Web API) will help to ensure that you know which components are being installed.

The Colleague Web API must be installed as the only application within the website to ensure correct functionality.

Note: The installers create the application pool in IIS Manager using the same name used for the website; therefore, the website and application pool names should be identical.

			Suggested port numbers	
Environment	Installation folder name	Installation website name	Unsecure	Secure
Development	development_WebApi[2]	development_WebApi	8082	8182
Production	production_WebApi	production_WebApi	8083	8183
Test	test_WebApi	test_WebApi	8084	8184
Training	training_WebApi	training_WebApi	8085	8185

Web server preparation steps

Before installing and configuring Colleague Web API, make sure your web servers are ready. Review the preparation information in this section and install any web server components needed.

The required web server components and actions depend on which components will be installed on that server. The following components are required before you can install and use Colleague Web API:

- Windows Server 2012 R2, 2012, 2008 R2, or 2008 web servers, which use IIS 8.5, 8, 7.5, or 7 respectively, require the IIS 6 compatibility components and specific role services enabled.
- All web servers require .NET Framework 4.5.1.
- ASP.NET 4.x must be registered on the web server hosting the Colleague Web API. You can also optionally re-register older versions of ASP.NET as necessary.
- ASP.NET 4.x extensions must be set to “allowed” on the web server hosting the Colleague Web API so that they will run.

Windows Server 2012 R2/IIS8.5 and 2012/IIS8: Install the ASP.NET Role Service and the IIS 6 Management Compatibility components

If you are using Windows Server 2012 running IIS 8 or 8.5 and the ASP.NET Role Service and the IIS 6 Management Compatibility components are not already installed, follow these steps to install them.

Procedure

1. On the Windows desktop, start **Server Manager**.
2. On the **Dashboard** page, click **Add roles and features**.
3. On the **Before you begin** page, verify that your destination server and network environment are prepared for the role and feature you want to install. Click **Next**.

[2] The name listed here is the physical folder name created when running the Installer.

4. On the **Select installation type** page, select **Role-based or feature-based installation** to install all parts of roles or features on a single server, and then click **Next**.
5. On the **Select destination server** page, select a server from the server pool, and then click **Next**.
6. On the **Select server roles** page, scroll down to **Web Server (IIS)**, and then expand the node. Under this node, expand the **Web Server** node.
7. Select the **Application Development** checkbox, and then expand the node. Select all options under the **Application Development** checkbox.

   When prompted, click **Add Features to add each of these features:**

   - Add features that are required for .NET Extensibility 3.5?
   - Add features that are required for ASP?
   - Add features that are required for ASP.NET 3.5?
8. Expand the **Management Tools** node and select all options under it. Also, expand the **IIS 6 Management Capability** node and select all options under it.

   When prompted, click **Add Features to add each of these features:**

   Add features that are required for IIS 6 Scripting Tools?
9. If all of the features are already installed, click **Cancel**. Otherwise, click **Next**.
10. On the **Select features** page, click **Next**.
11. On the **Confirm Installation Selections** page, click **Install**.

Results

After you click **Install**, the Installation progress page displays installation progress, results, and messages such as warnings, failures, or post-installation configuration steps that are required for the roles or features that you installed.

You can close the **Add Roles and Features** wizard while installation is still in progress, and view installation results or other messages in the **Notifications** area at the top of the **Server Manager** console. Click the **Notifications** flag icon to see more details about installations or other tasks that you are performing in **Server Manager**.

Windows Server 2008 R2/IIS7.5 and 2008/IIS7: Install the ASP.NET Role Service and the IIS 6 Management Compatibility components

If you are using Windows Server 2008 running IIS 7+ and the ASP.NET Role Service and the IIS 6 Management Compatibility components are not already installed, follow these steps to install them.

Procedure

1. Click **Start** > **Administrative Tools** > **Server Manager**.

   Note: It may take a few seconds for the Server Manager to start.
2. In the navigation pane, expand **Roles**, right-click **Web Server (IIS)**, and then click **Add Role Services**.

The **Select Role Services** dialog box is displayed.

3. In the **Select Role Services** pane, scroll down to **Application Development**.
4. Select all Application Development options if they are not already selected.
5. Scroll down to **Management Tools**.
6. Under **Management Tools**, make sure that **Management Service** and all **IIS 6 Management Compatibility** role services are installed, or install them if they are not. To do this, select all options under **IIS 6 Management Compatibility**.
7. Ont the **Confirm Installation Selections** pane, click **Next**, and then click **Install** .
8. Click **Close** .

   The **Add Role Services** wizard closes.

Install .NET framework

The web server requires .NET Framework 4.5.1.

Only Windows Server 2012 R2 comes with .NET Framework 4.5.1 pre-installed. For all earlier versions of Windows Server, you will need to install .NET Framework 4.5.1 from the following link: http://www.microsoft.com/en-us/download/details.aspx?id=40773.

To check whether .NET Framework 4.5.1 is installed, use **Programs and Features** in the **Control Panel** to confirm that it is installed.

After installing .NET Framework 4.5.1, restart your web server.

Note: If you have applications installed on your web server that require earlier versions of .NET, review and record the .NET version needed for each application. You will need to re-register the earlier version of .NET for these other applications in the next section of this document.

If no other applications on the web server require earlier versions of .NET, you do not need to record this information or complete the procedure for re-registering the earlier version of .NET for those applications.

Register ASP.NET 4.x

If you have not already done so, you must register ASP.NET 4.x on the web server hosting the Colleague Web API. Additionally, if your web server contains applications that require earlier versions of .NET, you may need to re-register the appropriate version of .NET for each application.

Procedure

1. At the Command Prompt, go to the directory just above the `Config` folder that contains the `machine.config` file.

   If you are running 64-bit Windows, navigate to %WINDIR%\microsoft.net \Framework64\v4.0.30319.

   If you are running 32-bit Windows, navigate to %WINDIR%\microsoft.net\Framework \v4.0.30319.

Note: The Command Prompt needs to be opened as an administrator for the commands to function properly.

2. If your web server contains any applications that require earlier versions of .NET, you must make note of each application and the required .NET version before continuing. See Determine which .NET version an application uses on page 15 for more details.

   Warning! The command in the next step registers ASP.NET version 4.x for all websites on the web server you are working with. If you have other applications on your web server that require earlier versions of ASP.NET you must identify those applications and their respective .NET versions before proceeding so that you can re-register the correct .NET version after running the command in the next step.

   After running the command in the next step, users may experience some disruption of the other applications until the correct .NET version is reregistered.

3. For Windows Server 2008 R2 and 2008 only, enter `aspnet_regiis -i` at the Command Prompt to register ASP.NET version 4.x for all websites on the web server:

   For additional information about this command, see http://msdn.microsoft.com/en-us/library/k6h9cz8h%28v=vs.80%29.aspx.

   For Windows Server 2012 R2 and 2012, the aspnet_regiis -i command is not supported.

   The aspnet_regiis utility does not apply, because NET 4.x is now a Windows component and must be administered as such. For more information, see http://support.microsoft.com/kb/2736284.

4. If your web server contains applications that require earlier versions of .NET, re-register the appropriate version for each application.

   See Re-register older versions of ASP.NET on page 14 for details, then return here to continue.

   Note: If you are running New Atlanta ServletExec® on your web server, make sure that .NET version for that application is set to 2.0 after running the command in Step 3 on page 14. If this version has changed, change it back to .NET 2.0 by following the procedure in Re-register older versions of ASP.NET on page 14.

5. Restart IIS using the iisreset.exe command at the Command Prompt.

   Note: The Command Prompt needs to be opened as an administrator for the commands to function properly.

Re-register older versions of ASP.NET

If you are using Windows Server 2012, skip this topic.

The aspnet_regiis utility does not apply, because NET 4.x is now a Windows component and must be administered as such. For more information, see http://support.microsoft.com/kb/2736284.

The default command used in these procedures to register ASP.NET (aspnet_regiis –i) does so for all websites on the web server. However, if you have other applications running on the same web server that require an earlier version of ASP.NET, you may need to specifically register that earlier version of ASP.NET for one or more individual websites after running the aspnet_regiis -i command.

To do so, you must first determine which .NET version the other applications are using if you do not know already.

Determine which .NET version an application uses

Follow these steps to determine which .NET version other applications use in Windows Server/IIS.

Procedure

1. On the appropriate web server, start IIS Manager.
2. Click the **Application Pools** node.
3. Highlight the application pool associated with the web server application you are checking.
4. In the **Actions** pane (right side of the **IIS Manager** window), select **Basic Settings**.

   The **Edit Application Pool** dialog box appears.
5. Check and record the .NET Framework version specified in the **.NET Framework** version field.
6. Click **Cancel**.
7. Repeat these steps for all other applications installed on the web server.

Re-register the correct .NET version for an application

Follow these steps to re-register the correct .NET version for an application in Windows Server/IIS.

Procedure

1. On the appropriate web server, start IIS Manager.
2. Click the **Application Pools** node.
3. Highlight the application pool associated with the application you are working with.
4. In the **Actions** pane (right side of the **IIS Manager** window), select **Basic Settings**.

   The **Edit Application Pool** dialog box appears.
5. In the **.NET Framework version** field, select the correct .NET Framework version for this application from the information you recorded earlier.

   See Determine which .NET version an application uses on page 15 for details.
6. Click **OK** to save your changes.
7. Repeat these steps for all other applications installed on the web server.

Allow the ASP.NET extension to run on the web server

Set the ASP.NET 4.x extension to be allowed in the ISAPI and CGI Restrictions list, so that ASP.NET 4.x can be run on the web server.

Procedure

1. On the appropriate web server, start IIS Manager.
2. In the left navigation pane, highlight the web server name.
3. Double-click **ISAPI and CGI Restrictions**.
4. Locate any ASP.NET 4.x extensions in the list. More than one may be listed.
5. Right-click each ASP.NET 4.x extension and select **Allow**.

   The restriction changes from Not Allowed to Allowed.
6. Restart IIS using the iisreset.exe command at the command prompt.

   Note: The command prompt needs to be opened as an administrator for the commands to function properly.

Limit IIS logging

You must limit the volume of IIS logging on all web servers by setting the logging levels to capture only errors and not successful requests. If you do not limit the IIS logs, the hard drive space on the web server will eventually fill up.

IIS 7.5/7

http://support.microsoft.com/kb/930909

IIS 8.5/8

To reduce logging overhead, follow these steps:

1. Access IIS Manager.
2. In the **Connections** pane on the left, select the server.
3. In the center pane, open the Configuration Editor.
4. In the section drop-down, select system.webServer/httpLogging.
5. Change the selectiveLogging value to LogError.
6. In the Actions pane, click **Apply**.

Install software updates

Before running the installer, you must install software updates in the environments with which you want to use the Colleague Web API.

About this task

Note: The release system automatically requires you to install any dependent software updates. Depending on which software updates are already installed, you might have more software updates to load in addition to the ones listed above.

The steps below provide an overview of the software update installation process. See the Updating Colleague Software manual for detailed procedures on installing software updates.

Procedure

1. Start SA Valet.

   You must right-click the **SA Valet** icon or executable file and click **Run as Administrator** even if you are logged in to the server as an administrative user.
2. Retrieve the software updates.
3. Review the release summaries for the software updates, particularly any pre-installation or post-installation steps and any implementation information.
4. Install the software updates into the environments that you are using for the Colleague Web API.

Install Colleague Web API

The Colleague Web API is an application that you install on a web server and provides a RESTful web interface for accessing Colleague data and business logic.

Run the Colleague Web API installer

Follow these steps to install this version of the Colleague Web API for the first time.

Before you begin

Complete these tasks before you install the Colleague Web API:

1. Ensure that you have completed all necessary preparations. See Prepare for Installation for details.
2. Make sure that you have a unique name for the website. See Component Naming Conventions for more information on naming environments and websites.

   Note: You will create a new website for each Colleague environment, so Ellucian recommends that the website name match the associated Colleague environment name.
3. Make sure that the port you want to use is open in your firewall. You can use any port number; however, it must be a port number that is not already being used.
   a. The Website Port Number that you enter here is a non-SSL port. If you need to run Colleague Web API using SSL, for example, you must enter a non-SSL port here and change it to an SSL port later.
   b. If your institution uses eTranscripts, see Using Colleague eTranscripts for the range of ports that will allow Colleague Web API to be accessible.
   c. The web server hosting the Colleague Web API must be able to connect to the application listener port on the Colleague Application Server.
   d. External web applications, such as Colleague Self Service, must be able to connect to the web server hosting Colleague Web API on the IIS website port for the Colleague environment.

Procedure

1. Log on to your server as an administrator.
2. Disable any antivirus software that may be running on the server on which you plan to run the installer.

   The Colleague Web API installer might not install correctly if antivirus software is running.
3. From the location on your server where you downloaded the Colleague Web API installer, run the executable file with Administrator privileges.

Note: For Windows Server installations, you must right-click the installer file and click **Run as Administrator** even if you are logged in to the server as an administrative user.

4. On the Welcome page, click **Next** to continue to the Customer Information page.
5. On the Customer Information page, enter your institution's name, ID, and password. Click **Next** to continue to the License Agreement page.
6. On the License Agreement page, read the license agreement and select **I accept the terms in the license agreement**. Click **Next** to continue to the New or Existing Website page.
7. On the **New Website** page, select **Create New Website** to create a new IIS website during the installation. Click **Next** to continue to the Destination Folder page.
8. On the Destination Folder page, review the path where the new website folder will be created.

   A new folder for the website will be created under the folder shown, and the name that you enter for the website in the next step will be used as the folder name.

   If needed, you can click **Change** to select a new folder location. Ellucian recommends that you accept the default path.

   Click **Next** to continue to the Website Information page.
9. On the Website Information page, enter the name and port number to use for the new website that you want to create, and enter a name for your web application. Click **Next** to continue to the Ready to Install page.

   Note: If the installer fails at this point on Windows Server, it is most likely because the correct role services are not installed on the server. See Web server preparation steps on page 11 for more information, and then re-run the installer.
10. On the Ready to Install page, click **Install** to begin the installation process.
11. When the installation is complete, click **Finish** from the final installer page.
12. Restart your antivirus software.
13. After installing the components, verify that the `settings.config` file is contained in the `\ColleagueApi\App_Data` folder.
14. Within IIS Manager for the website where the Colleague Web API components were installed, check that the website was installed with the following node and application: *website name* > ColleagueApi.

Results

The Colleague Web API installer delivers the following components:

- The Colleague Web API (installed for the selected website).
- The Colleague Web API configuration page (installed for the selected website), which allows you to manage applications settings, such as logging and Colleague environment connection settings related to Colleague Web API..

Upgrade Colleague Web API installation and configuration

You can upgrade an existing Colleague Web API installation with one exception: *Under no circumstances* should you upgrade an existing Student Finance Views website (Student Finance Views 1.0 or Colleague Application web services 1.0) using the new installers.

Procedure

1. Disable any antivirus software that may be running on the server on which you plan to run the installer.

   The Colleague Web API installer might not install correctly if antivirus software is running.
2. From the location on your server where you downloaded the Colleague Web API installer, run the executable file with Administrator privileges.

   Note: For Windows Server installations, you must right-click the installer file and click **Run as Administrator** even if you are logged in to the server as an administrative user.
3. On the Welcome page, click **Next** to continue to the Customer Information page.
4. On the Customer Information page, enter your institution's name, ID, and password. Click **Next** to continue to the License Agreement page.
5. On the License Agreement page, read the license agreement and select **I accept the terms in the license agreement**. Click **Next** to continue to the New or Existing Website page.
6. On the **New Website** page, select **Use Existing Website**. Click **Next** to continue.
7. On the Website Selection page, select the existing website you want to use. Click **Next** to continue.

   Note: When you select the existing website, the Application Pool parameters on the existing Application Pool are reset. You will need to reconfigure the Application Pool parameters using the steps in Colleague Web API performance considerations on page 22.
8. On the Ready to Install page, click **Install** to begin the installation process.
9. When the installation is complete, click **Finish** from the final installer page.
10. Restart your antivirus software.

Uninstall Colleague Web API

Use caution when uninstalling a Colleague Web API website.

Procedure

1. On the appropriate web server, start IIS Manager.
2. Double-click the **Sites** node to view the list of websites.
3. Right-click on the website you want to uninstall and click **Manage Website** > **Stop**.

4. Righ-click on the website you want to uninstall and click **Manage Website** > **Advanced Settings**, then make note of these properties to be used later in this procedure:
   - The name of the application pool (**Application Pool** field).
   - The path to the website contents (**Physical Path** field).
5. Click the **Application Pools** node.
6. Using the application pool value from above, right-click on the application pool in the list of application pools and click **Stop**.
7. Under the **Sites** node, right-click on the website and select **Remove**.
8. When prompted, confirm the website removal.
9. In Windows Explorer, delete the site contents for the website by using the specified physical path.

   Delete the entire folder specified by the physical path. For example, if the physical path is `C:\Program Files (x86)\Ellucian\test_WebApi`, you would delete the `test_WebApi` folder within the `C:\Program Files (x86)\Ellucian` folder.
10. In IIS Manager, determine if you can remove the application pool used by the website. Right-click on the specified application pool and select **View Applications**.

If...	Then...
the list is empty	remove the application pool by returning to the list of application pools, right-clicking on the application pool and selecting **Remove**. Confirm the application pool removal when prompted.
the applications are listed	do not remove the application pool. Before removing the application pool, research these applications to determine if it is safe to remove the application pool.

Warning! Be extremely careful when removing the application pool. Removing an application pool that still has associated applications will cause those applications to stop working.

Configure and Administer Colleague Web API

Configure web services parameters

You will need to configure the Colleague Web API.

Procedure

Complete the Web Services Parameters (WSPD) form in the environment where you installed the Colleague Web API.

Note: The shared secret will be encrypted and used to secure Colleague Web API transactions between, for example, Colleague Self Service and Colleague. You will enter the same shared secret when configuring the Colleague Web API configuration page.

Configure Colleague time zone

You must define the Colleague time zone to ensure that time data communicated by APIs are correctly converted between Universal Time Coordinated (UTC) and Colleague local time.

Procedure

Complete the Colleague Time Zone Settings (CTZS) form in the environment where you installed the Colleague Web API.

Note: The Colleague time zone is not necessarily the same as the time zone of the Colleague application server, database server, or web server. It is the time zone in which your institution operates.

Colleague Web API performance considerations

Some of the IIS application pool default settings are less than ideal for Colleague Web API's caching strategy. The following procedure details which application pool settings should be changed for the best caching performance.

About this task

It is recommended that you change the following settings of the application pool running Colleague Web API for optimal cache performance.

Note: The configuration settings in this task assume that the Colleague Web API warm-up script is being run at least once every 24-hours. See Run the warm up script for details on how to setup the Colleague Web API warm-up script to run automatically. Ellucian recommends running the Colleague Web API warm-up script to prevent users from experiencing cache warm-up delays.

Procedure

1. In IIS Manager, select the application pool for Web API.
2. Right-click **Advanced Settings**.
3. Under the **Process Model** settings:
   a) Change the **Idle Time-out** (minutes) to 1440 minutes.
   b) Ensure that the **Maximum Worker Processes** is set to 1.
4. Under the **Recycling** settings:
   a) Change the **Regular Time Interval** (minutes) from 1740 to 0.
   b) Ensure there are no **Specific Times** listed by clicking on the **TimeSpan Collection Editor** button (with the ellipses) and making sure the members list is empty.
   c) Ensure that the **Private Memory Limit**, **Request Limit**, and **Virtual Memory Limit** values are set to 0.
5. Click **OK**.

Configure the Colleague Web API

Complete the procedures in this section to configure the new website and the Colleague Web API application.

Procedure

1. To configure the website that you added during installation and connect to the corresponding Colleague environment, click the name of the website in IIS Manager to select it. For example, dev_WebApi.
2. Generate machine keys for security.
3. Configure the Authentication Services for your API site.
4. Configure .NET 4.X for the website.
5. Configure the Colleague connection settings.
6. Configure the Web API logging settings.
7. Configure the Web API settings.

Generate machine keys for security

Unique machine keys are automatically generated with IIS default settings.

About this task

These values change with each application pool recycle and can result in lost sessions if the application pool is recycled with users in the system. To provide consistent, yet secure, behavior for the application pool, the machine keys must be generated within IIS and saved for future sessions. The following procedure takes you through the process of generating and saving the keys.

Note: Even though the machine key can be set up at the server, website, or application level within IIS, you should perform the following tasks only at the website level, on the Colleague Web API website node. Altering the machine key at the server level affects all websites hosted on the server and can cause those websites to stop working correctly. Altering the machine key at the application level will not preserve the generated keys when an update installation is performed for this site.

Procedure

1. Open IIS Manager on the web server.
2. Click on the Colleague Web API website to view the website configuration options.
3. Double-click the **Machine Key** icon.
4. Clear the check boxes on the **Machine Key** form.

   This will use specified keys instead of generating new keys for each instance of the website.
5. On the right side of the **Machine Key** form, click on the **Generate Keys** link to generate unique keys for validation and decryption.

   The **Validation Key** and **Decryption Key** fields will now have values for each required key.
6. On the right side again, click on the **Apply** link to save the changes.
7. Now that the key values have been generated, recycle the IIS application pool for the website.

   If you accidentally generated machine keys at the web application level (e.g., at the “ColleagueApi” node instead of the “Colleague Web API” website node), you will need to follow these steps to correct this:

   a) In IIS Manager, select the ColleagueApi web application node, and view the **Machine Key** form. Save the keys you generated here so you can apply them at the website level.

   b) Right-click the ColleagueApi web application, and then select **Explore**.

   c) Locate the web.config file. In this file, remove the entire `<machineKey ... />` section. Otherwise, the machine key settings at the web application level will override the machine key settings at the website level.

   d) In IIS Manager, select the Colleague Web API website node, and view the **Machine Key** form. Clear all check boxes, paste the generated keys, which you saved above, and then click **Apply**. If you generate new machine keys at the website level instead of using the saved keys, you will need to re-enter the shared secret on the Colleague Connection Settings form.

Configure the authentication services for your API site

Colleague Web API requires that anonymous authentication be enabled for proper user authentication. All other authentication methods must be disabled. You must complete these steps at the application level.

Procedure

1. For your WebAPI site, in IIS Manager, go to your ColleagueApi application.
2. Select **Authentication**.
3. Change the Anonymous Authentication status to Enabled if it is not already set to Enabled.
4. Ensure that all remaining authentication methods listed are disabled. To disable, select each authentication method and click Disable in the action pane.

Configure .NET 4.X for the website

Apply the following procedure to your Colleague Web API configuration, as you need to make sure it is using .NET4.x.

Procedure

1. On the appropriate web server, start IIS Manager, and click the **Application Pools** node.
2. Highlight the application pool associated with the website that you are working with.
3. In the **Actions** pane (right side of the IIS Manager window), select **Basic Settings**.
4. In the **Edit Application Pool** dialog box, in the **.NET Framework version** field, select .NET Framework v4.x. If version 4.x is already selected, click **Cancel**. There is no need to reselect it.

   Leave the **Managed pipeline mode** field as it is. Click **OK** to save your changes.

Configure the Colleague connection settings

You must complete the step of generating machine keys for the website before continuing with this section.

About this task

Note: The links and pages used to maintain the connection setting are only available when browsing the Web API application in a web browser on the server.

Procedure

1. Browse to the Web API website from within IIS by selecting the ColleagueApi application node under the website where the Web API has been installed.
2. Go to **Manage Application** > **Browse Application**.
3. Click the **API Administration** link.

Note: If you are using Internet Explorer and experience trouble viewing the API Administration page, ensure that the hostname in the URL is localhost and not the name of the server. This will ease browser security restrictions that can interfere with page interactivity.

4. Click the **Connection Settings** link.
5. Enter the Colleague connection settings as shown below.

Field	Description
Account DMI Registry Name	Enter the name of the Colleague environment to which you are associating the Colleague Web API. Be sure to include the _rt suffix. Typically, this value is the same as value displayed in the Account DMI Registry Name field on the Define Account WebConfig (DWEB) form in Colleague.
DMI Application Listener IP Address	Enter the host name or IP address of the server where the DMI application listener is installed.
DMI Application Listener Port	Enter the port on which the DMI application listener is listening. If this a secure port also check the Connect to DMI Application Listener Securely checkbox and, if needed, enter the Certificate Host Name Override.
Connect to DMI Application Listener Securely	When checked, communicate with the DMI application listener using secure sockets (TLS 1.0+). The DMI Application Listener Port number must represent a secure port. In addition, you may also need to enter a Certificate Host Name Override in order to achieve secure communication. **Note:** You can view the DMI Listener properties in SA Valet to check whether a port is secure. See “Securing a DMI Listener” in the *Managing Colleague Software Environments* manual for more information
Certificate Host Name Override	Enter the host name of the security certificate if it differs from the DMI Application Listener IP Address field. In most cases you can leave this field blank. This field allows you to override the host name that will be used to verify the server’s security certificate. For example, if your security certificate was issued to `colleague.mycollege.edu` but you use the IP address or an alternate

Field	Description
	host name in the DMI Application Listener IP Address then you would need to enter `colleague.mycollege.edu` in this field for the security certificate to be trusted by the secure connection.
Connection Pool Size	Enter the maximum number of persistent DMI client socket connections that can be created between the API and the DMI Application listener. Current recommendations are to enter a number no greater than 50 in this field. If you are using the DAS Data Reader option then the recommended value for this field is 20. **Note:** The connection pool values suggested are based on internal performance benchmarking and working with clients on performance tuning. These values are provided as reasonable starting points based on experience, but, with any tuning parameters, they should be verified by running load tests to ensure they produce the expected results. The Colleague Self-Service Hardware and Software Guide provides further explanation of these topics and includes topics on tuning and sizing the Colleague Web API.
Use DAS Data Reader	When checked, enables the use of the DAS Data Reader option. The Colleague Web API will perform read-only data operations using the DAS listener and DAS credentials specified on the Connection Settings page. This option provides faster data access and frees-up the DMI application listener connections for transactional processing.
DAS Registry Name	Enter the database registry name used with the DAS listener. Unlike the Account DMI Registry Name specified for the DMI application listener connection, this value typically does not end with the _rt suffix. This value is typically found at the end of the Connection String on the Environment Accounts Setup dialog in SA Valet. This field is only maintainable if the Use DAS Data Reader filed is enabled.

Field	Description
DAS Listener IP Address	Enter the host name or IP address of the server where the DMI DAS listener is installed. This field is only maintainable if the Use DAS Data Reader filed is enabled.
DAS Listener Port	Enter the port on which the DMI DAS listener is listening. If this is the secure port also check the Connect to DAS Listener Securely checkbox and, if needed, enter the DAS Certificate Host Name Override. This field is only maintainable if the Use DAS Data Reader filed is enabled.
Connect to DAS Listener Securely	When checked, communicate with the DAS listener, using secure sockets (TLS 1.0+). The DMI Listener Port number must represent a secure port. In addition, you may also need to enter a DAS Certificate Host Name Override. **Note:** You can view the DMI Listener properties in SA Valet to check whether a port is secure. See “Securing a DMI Listener” in the *Managing Colleague Software Environments* manual for more information. This field is only maintainable if the Use DAS Data Reader field is enabled.
DAS Certificate Host Name Override	Enter the host name of the security certificate if it differs from the DAS Listener IP Address field. In most cases you can leave this field blank. This field allows you to override the host name that will be used to verify the server’s security certificate. For example, if your security certificate was issued to `colleague.mycollege.edu` but you use the IP address or an alternate host name in the DAS Listener IP Address, you would need to enter `colleague.mycollege.edu` in this field for the security certificate to be trusted by the secure connection. This field is only maintainable if the Use DAS Data Reader filed is enabled.

Field	Description
DAS Connection Pool Size	Enter the maximum number of persistent DAS client socket connections that can be created between the API and the DAS listener. Current recommendations are to enter a number no greater than 30 in this field. **Note:** The connection pool values suggested are based on internal performance benchmarking and working with clients on performance tuning. These values are provided as reasonable starting points based on experience, but, with any tuning parameters, they should be verified by running load tests to ensure they produce the expected results. The Colleague Self-Service Hardware and Software Guide provides further explanation of these topics and includes topics on tuning and sizing the Colleague Web API. This field is only maintainable if the Use DAS Data Reader field is enabled.
DAS Username	Enter the user name that should be used for establishing DAS sessions. Connections with the DAS listener utilize a single set of credentials for establishing DAS sessions like the Colleague Application Server uses for transactions that use the DMI connection pools. This is the same user name that would be entered on the Environment Accounts Setup dialog in SA Valet. This field is only maintainable if the Use DAS Data Reader field is enabled.
DAS Password	Enter the password that should be used for establishing DAS sessions. Connections with the DAS listener utilize a single set of credentials for establishing DAS sessions just like the Colleague Application Server uses for transactions which utilize the DMI connection pools. This is the same password that would be entered on the Environment Accounts Setup dialog in SA Valet. This field is only maintainable if the Use DAS Data Reader field is enabled.

Field	Description
Shared Secret/Confirm Shared Secret	For this environment, confirm the code that you entered in the Web Services Parameters (WSPD) form.

6. Click **Test App Listener Connection** and follow the prompts to test the connection to the DMI Application listener.
7. Click **Test DAS Listener Connection** to test the connection to the DMI DAS listener.
8. Click **Save**.

   The application will restart automatically.

DAS Data Reader option

The DAS Data Reader option allows Colleague Web API to fetch read-only data directly from a DAS listener. This provides the fastest path possible for reading data in the Colleague architecture and can do so using fewer resources when compared to using the DMI application listener for reading data. Given that the majority of data provided through the Colleague Web API is from read-only sources, enabling this feature removes data read requests from the DMI application listener, freeing those connections for data updates and login/logout activities

Using the DAS Data Reader

The DAS Data Reader option is currently an opt-in feature. To use, access the Colleague Web API's Colleague Connection Settings page, check the Use DAS Data Reader checkbox, and fill in the remaining DAS fields.

While the use of a dedicated DAS listener is not required, under many circumstances it will be advantageous to use a dedicated DAS listener. Reasons for doing so include the number of DAS connections that will be made from the Web API(s) when load balancing, and not creating a single failure point for all Colleague DAS connections (API and UI).

DAS Data Reader security considerations

- Since this is potentially a new connection path between the Colleague Web API server and the server hosting the DAS listener, you may need to open firewall ports to use this feature.
- You may, optionally, secure the communication between the Colleague Web API and DAS listener. Typically, the DAS listener does not have a secure port setup/enabled as the Colleague Application Server cannot communicate with the DAS over a secure connection. You can add a secure port to the primary or a dedicated DAS listener the same way you would a DMI application listener. If you add a secure port to your primary DAS listener, be sure to leave the non-secure port in place and enabled otherwise the Colleague Application Server will not be able to communicate with the primary DAS listener.
- The connections made to the DAS listener from the Colleague Web API do set a much lower connection timeout (less than an hour) than traditional DAS connections made from the Colleague Application Server.

Configure the Web API logging settings

The links and pages used to maintain the logging setting are only available when browsing the Web API application in a web browser on the server.

Procedure

1. Browse to the Web API website from within IIS by selecting the ColleagueApi application node under the website where the Web API has been installed.
2. Go to **Manage Application** > **Browse Application**.
3. Click the **API Administration** link.
4. Click the **Logging** link.
5. On the Logging page, select the level at which you want logging to occur for web server debugging.

   Ellucian recommends that you enable debugging only when actively troubleshooting. The log file is saved to the following location: `*Colleague Web API xx InstallationFolder*\App_Data\Logs\ColleagueWebApi.log` (where *xx* is the latest version) This option indicates the level of detail provided in the web server debug logs when debugging is enabled. Options include: Off, Error, Warning, Information, and Verbose.
6. Click **Save**.

   The application will restart automatically.

Configure the Web API settings

The links and pages used to maintain the Web API setting are only available when browsing the Web API application in a web browser on the server.

About this task

Note: Public access must be enabled for the WEB.API.CONFIG entity on the Web Services Parameters (WSPD) form for the API Settings to be read at runtime. Public access to this entity has been pre-delivered so there is no need visit the WSPD form before configuring the API Settings. The settings are not exposed through any endpoints.

The API Settings page allows you to maintain settings that are used by various services within the Colleague Web API. The API Settings are stored in Colleague within a named Configuration Profile (WEB.API.CONFIG entity). This allows one or more Colleague Web API instances, attached to the same Colleague environment, to easily share the same settings. You do not have to enter the same setting in each Colleague Web API instance.

Procedure

1. Browse to the Web API website from within IIS by selecting the ColleagueApi application node under the website where the Web API has been installed.
2. Go to **Manage Application** > **Browse Application**.

3. Click the **API Administration** link.
4. Click the **API Settings** link.
5. If you have not logged in, you will prompted to log in.

   Note: If this is the first time accessing the API Settings page on a new Web API installation, you will be taken to the API Settings Profile page before being taken to the API Settings page.

6. Complete the API Settings Profile page if necessary, then click **Save** to return to the API Settings page.

   The API Settings Profile page allows you to choose or create a new Configuration Profile which will be used by this Web API instance.

   The settings from the Configuration Profile you specify on this page will be maintainable on the API Settings page and used at runtime by the Web API. The API Settings Profile page will be shown the first time you access the API Settings page for a new Web API installation, or by clicking the **Change** link at the top of the API Settings page where the current Configuration Profile name is displayed.

Field	Description
Current Profile	Displays the current Configuration Profile this Web API instance is using.
Change Profile To	Displays all available Configuration Profiles that are defined in the Colleague Environment associated with this Web API Instance. You may only choose a value from this drop-down or define a new value in the Create a New Profile field.
Create a New Profile	Allows you to define a new Configuration Profile name. The profile name should only contain characters, numerals, and underscores.Text entered in this field will automatically be converted to uppercase and have all spaces removed after the field has been edited. You may only enter a new value in this field or select an existing profile from the Change Profile To drop-down.

7. Complete the API Settings page.

   Table 1: API Settings - Photos

   The web API endpoints that provide person photos use the fields on the Photos tab; they tell the Web API where to and how to fetch the photos from your image server using HTTP(S) requests. This setup is very similar to person photo configuration used by Colleague UI but does not use the existing setup forms or configuration used by Colleague UI.

Field	Description
Image Server Base URL	Enter the base URL needed by the image server to form a request for an image. This field will be combined with the person's ID and the Image Extension (if entered) to form the full URL used in the request to the image server. Use the URL Preview field to preview what the full URL will look like when sent to the image server.
Image Type	Select the type (MIME/content type) of the image that is returned by the image server request. This will subsequently become the content type returned by the Web API when a photo request is made.
Image Extension	Enter the image file extension of the image on the image server. You do not have to enter a leading period when specifying the file extension, just enter the extension text, such as “jpg”. If the URL sent to the image server does not require an extension, leave this field blank. Use the URL Preview field to preview what the full URL will look like when sent to the image server.
URL Preview	Provides a preview of how the Image Server Base URL and Image Extension (if entered) will be combined with a Person ID to form a request to the image server.
Custom Headers	Use the Custom Headers fields to define additional request headers that should be sent with the HTTP(S) request to the image server. A common usage for this field is to provide basic authentication credentials to the image server. • To add a new header name/value pair, click the **Add** button. • To remove an existing header name/value pair, click the **Remove** button to the right of the header name/value pair you want to remove.

Table 2: API Settings - Reports

The fields on the Reports tab are used by the Web API endpoints where a logo is used on reports or when a watermark is needed on an unofficial transcript; they tell the Web API where to fetch the photos from your image server using HTTP(S) requests.

Field	Description
Report Logo Path	Enter the relative path on the web server to the institution's report header image. (Example: /content/images/logo.jpg). If no valid path is specified, no logo will be displayed on the report/document. Example: /content/images/logo.jpg
Unofficial Watermark Path	Enter the relative path on the web server to the institution's watermark image for marking reports/documents as “Unofficial”. If no valid path is specified, no watermark will be displayed on the report/document. Example: /content/images/watermark.jpg

Photo setup scenario examples

The following are sample photo setup scenarios.

Simple web server with image files

URL needed by the web server: `http://www.school.edu:8080/images/000140.jpg`

Field	Value
Image Server Base URL	`http://www.school.edu:8080/images/`
Image Type	`JPEG`
Image Extension	`jpg`
Custom Headers	none

Website that requires HTTP basic authentication using a Webuser user ID

URL needed by the web server: `http://www.school.edu:8080/images/000140.jpg`

Header needed by the web server: `Authentication`

To create the basic authorization header value for this example, Base64 encode the username and password (concatenated with a colon) and prepend `Basic` and a space to the Base64 encoded value.

- Plain text username/password: `webuser:u141m4g3`
- Base64 encoded: `d2VidXNlcjp1MTQxbTRnMw==`
- Final value: `Basic d2VidXNlcjp1MTQxbTRnMw==`

For more information on HTTP authentication options, see http://en.wikipedia.org/wiki/Basic_access_authentication or the original RFC for HTTP Authentication.

Field	Value
Image Server Base URL	`http://www.school.edu:8080/images/`
Image Type	`JPEG`
Image Extension	`jpg`
Custom Headers (name/value)	`Authorization/Basic d2VidXNlcjp1MTQxbTRnMw==`

CGI script that requires only person ID

URL needed by the web server: `http://www.school.edu:8888/cgi-bin/GetImage?IMAGE.ID=000140`

Image Type returned: `JPEG`

Field	Value
Image Server Base URL	`http://www.school.edu:8888/cgi-bin/GetImage?IMAGE.ID=`
Image Type	`JPEG`
Image Extension	leave blank
Custom Headers	none

Colleague Web API and SSL

It is recommended that the Colleague Web API be hosted using a secure website in IIS as other, possibly external, applications may be accessing functionality exposed by the Colleague API.

Note: Apply the following information to your Colleague Web API configuration, as you need to make sure it is using HTTPS.

The following communication paths can be secured:

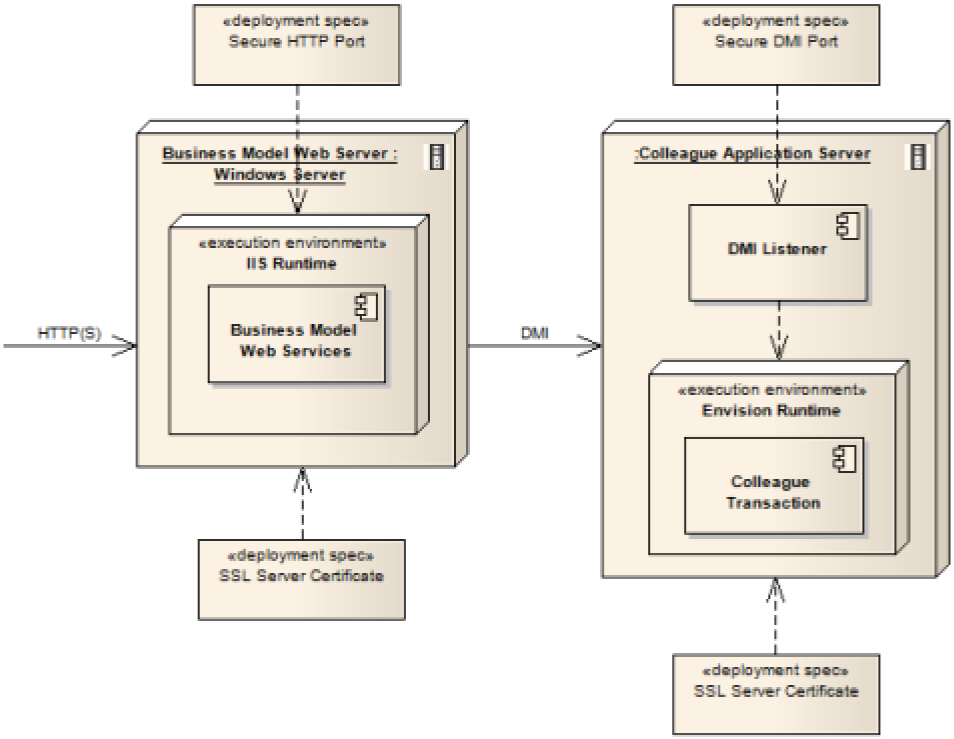
- Between other applications, such as Colleague Self Service and the web server hosting the Colleague Web API.
- Between the web server hosting the Colleague Web API and the Colleague Application Server.

Security between Colleague Web API and the Colleague Application Server through DMI is discussed in the “Securing Colleague Connections” chapter of the Managing Colleague Software Environments manual.

The certificate of the IIS website needs to be issued for the hostname of the server. For instance, if your IIS server’s hostname is “myiisserver.institution.edu,” the certificate also needs to be issued to “myiisserver.institution.edu.” If your SSL configuration is incorrect, the user will not be able to connect to the Colleague Web API.

To install server certificates in IIS, refer to the following Microsoft documents: http://technet.microsoft.com/en-us/library/cc732230(WS.10).aspx.

The figure below shows how security should be enabled between components.



Note: Self-signed certificates are not supported. The SSL certificate must be signed by a valid Certificate Authority (CA).

The certificate of the IIS website needs to be issued for the hostname of the server. For instance, if your IIS server's hostname is “myiisserver.institution.edu,” the certificate also needs to be issued to “myiisserver.institution.edu.” If your SSL configuration is incorrect, the user will not be able to connect to the Colleague Web API.

To install server certificates in IIS, refer to the following Microsoft documents: http://technet.microsoft.com/en-us/library/cc732230(WS.10).aspx.

Security concepts

Colleague Web API uses SSL technology to secure communications between servers. If you choose to deploy security, you must have one certificate for each server that you intend to secure.

All Colleague environments that run on a server with a certificate are capable of using SSL-secured communications. Though SSL is enabled at the environments level, certificates are managed for all environments on the server.

Public and private keys

A server employing SSL encryption requires two keys: a private key and a public key.

- *Private key* - Kept protected and is known only to the owner.
- *Public key* - Distributed openly to all devices that request a connection to the server through an SSL-secure port.

As part of the public key's certification process, the key is packaged up, with the key owner's information, into a certificate-signing request. The public key, with information identifying the key pair owner, becomes the certificate after a trusted independent authority (known as a Certificate Authority) validates the information regarding the key owner. When the key pair owner's identity is verified, the CA approves the public key with a digital signature. The signed public key is known as the certificate.

Certificate overview

To enable SSL on the web server, you must purchase a certificate from a certificate authority (CA) and install the certificate on your web server. Ellucian recommends purchasing, at minimum, a 128-bit certificate.

The web server certificate must first be combined with its corresponding private key into a PKCS #12 vault file. When the vault file has been generated, the certificate can be imported into the web server's OS key store for the "local machine." When importing the vault file, it is important that you use the local PC's personal store and mark the vault file private key as "exportable."

Failure to do these will result in SSL not working for the web server.

Install an SSL certificate for the website

You need an SSL certificate installed in the following locations.

- On the web server hosting the Colleague Web API to secure communication with other Ellucian applications and third-party products.
- On the Colleague Application Server to secure the communications between the web server hosting Colleague Web API and the DMI Listener.

Note: Self-signed certificates are not supported. The SSL certificate must be signed by a valid Certificate Authority (CA).

Ellucian requires an SSL-enabled internal server with IIS installed to support the Colleague Web API components, for example. If you do not have a certificate from a Certificate Authority (CA) for the internal server, then you can use IIS to create a certificate request. You can use this certificate request capability to contact a CA (such as VeriSign) to obtain the CA's digital signature on your certificate for that particular instance of the server.

The CA requires you to fill out a web form with a variety of information about your site including the name of the website, your organization, organizational unit, city/location, state/province, country, and technical contact information. The certificate must be renewed every year to remain in force.

The CA conducts an investigation of your application and then affixes its public key and digital signature to your certificate. The signed certificate includes a public key that, together with the public key of the CA, allows communication with the client using an encrypted communications link (for example, an Ellucian application, such as Self-Service, to the server hosting the Colleague Web API).

The certificate of the IIS website also needs to be issued for the Fully Qualified Domain Name of the server. For instance, if your IIS server's Fully Qualified Domain Name is "myiisserver.institution.edu," the certificate also needs to be issued to "myiisserver.institution.edu." If your SSL configuration is incorrect, then other applications will not be able to connect to Colleague Web API.

To install server certificates, please refer to the following Microsoft documents: http://techhttp://technet.microsoft.com/en-us/library/cc732230(WS.10).aspxnet.microsoft.com/en-us/library/cc732230(WS.10).aspx.

Enter a secure port number and turn on SSL for a new website

If you created a new website during the Colleague Web API installation, for example, the port entered was a non-SSL port.

About this task

You can either change the existing port number to be a secured port, or leave it as is and add a secure port number. You can then turn on SSL for the website.

Follow these steps to change or add a secure port number and turn on SSL for a new website.

Procedure

1. On the appropriate web server, start IIS Manager.
2. Right-click the website that you are working with, and click **Edit Bindings**. The **Site Bindings** dialog box is displayed.
3. If you want to change the existing port to be a secure port, select it and click **Edit**. In the **Edit Site Bindings** dialog box, select https. If you are not allowed to change the port to be a secure port, see Step 4.

4. If you want to add an SSL port (or if you are not allowed to edit the existing port in Step 3), click **Add** from the **Site Bindings** dialog box. In the **Add Site Bindings** dialog box, create a new binding with a type of https and select the certificate that you purchased for the servers.
5. When you ran the installer and created a new website, a temporary binding was also created. You should remove this temporary binding by clicking on it, and then clicking **Remove**. Click **Close**.
6. Save your changes and restart the website. To restart the website in IIS Manager, right-click the website name, select **Manage Web Site**, and then click **Restart**.
7. Double-click the website name. In the **Features View**, click the **SSL Settings** icon. The SSL settings are displayed. Select the **Require SSL** check box.

Colleague Web API security

The Colleague Web API is secured using permissions that are defined and assigned to roles in Colleague UI.

The same roles can be used to secure the Colleague Web API as to secure, for example, Colleague Self-Service. See the *Self-Service Installation and Administration* manual for related information.

Colleague security

A security boundary between the Web API in .NET and the Colleague application server is accessed using the DMI protocol.

This boundary is secured with a shared secret and message hash technique. The shared secret is configured on both sides of the boundary: in Colleague and in .NET using the Colleague Web API application.

Each user must have a record in the resource database. This includes Colleague User Interface users who will use single sign on to access administrative views of student data within, for example, Colleague Self-Service. See "Set up Colleague Self-Service users" in the *Self-Service Installation and Administration* manual for related information.

Security notifications

When a user's personal information, such as a phone number, is changed in an application that uses the Colleague Web API, an email can be sent to users to notify them of the change.

Colleague Web API security notifications make use of Envision Data Exchange (EDX). From the perspective of EDX, the Colleague Web API is a subscriber. See EDX Administration for additional information about EDX.

Enable security notifications

You can enable all available security notifications, then if necessary, you can disable specific triggers.

Procedure

Run the EDX Interface Load Utility (EDIL) process to install the Colleague Web API security notification components.

Field	Entry
Clear all existing trigger definitions	`No`
Subscriber Interfaces to Load	`SEC.NOTIFY`

Results

You can view the individual triggers that are associated with the SEC.NOTIFY subscriber on the EDX Subscriber Definition (EDXS) form.

What to do next

Define the text of the security notification using the appropriate form.

Security notification text

For all situations in which security notifications can be sent, you can specify the text of the notification.

The form used to specify the text depends on the area to which the notification pertains.

Area	Form used to specify the security notification text
Banking information	Use the Banking Information Web Parameters (BIWP) form to define the text of the security notifications that can be sent to users when their payroll direct deposit or refund/ reimbursement/ payment bank account information is changed.
Contact information (email, phone)	Use the Core Data Security Notifications (CDSN) form to define the text of the security notifications that are sent to users when their email addresses or phone numbers are changed.

Due to privacy and security concerns, Ellucian recommends certain best practices with regard to all security notifications.

Do	Don't
Tell users what to do if they did not initiate the changes to their personal information.	Include any personal information about the user in your notifications.
Indicate that your institution will never ask for a user's password in an email.	Include a link to a login page or form.

Disable individual security notifications

You can disable specific security notification triggers for the SEC.NOTIFY subscriber.

Before you begin

Before disabling a particular security notification, use the EDX Subscriber Definition (EDXS) form to view the list of triggers (see the EDX Documents field) and their associated RFSPECS file (see the Triggered Files field).

Procedure

1. Access the EDX Trigger Entry (EDXT) form.
2. At the EDX Subscribers LookUp prompt, enter `SEC.NOTIFY`.
3. At the RFSPECS file LookUp prompt, enter the name of the triggered file for the trigger you want to disable.
4. Detail from the Opc field to the EDX Direct Trigger Definition (ED09) form.
5. On ED09, detail from the Trigger Condition field to the EDX Trigger Conditions (EDXC) form.
6. On EDXC, in the Criteria/Field Name field, enter any constant surrounded by quotes to replace the current value.

   Entering a constant, such as `"Y"`, makes the trigger condition always false.
7. Click **Save All**.

Re-enable a disabled security notification

If you have previously disabled specific security notifications for the SEC.NOTIFY subscriber, you can re-enable them later if necessary.

Procedure

1. Access the EDX Trigger Entry (EDXT) form.
2. At the EDX Subscribers LookUp prompt, enter `SEC.NOTIFY`.
3. At the RFSPECS file LookUp prompt, enter the name of the triggered file for the trigger you want to re-enable.
4. Detail from the Opc field to the EDX Direct Trigger Definition (ED09) form.
5. On ED09, detail from the Trigger Condition field to the EDX Trigger Conditions (EDXC) form.

6. On EDXC, enter one of the following in the Trigger Condition fields, depending on the triggered file with which you are working:

RFSPECS file **(trigger condition)**	**Trigger condition**

PERSON
(SEC-PERSON-PEOPLE.EMAIL)
	`WITH`
	` SUBR("S.NOTIFY.PERSON.EMAIL",`
	` @ID) EQ "never matches"`
	`OR SUBR("S.NOTIFY.PERSON.PHONE",`
	` @ID) EQ "never matches"`

PERSON.ADDR.BNK.INFO
(SEC-PERSON.ADDR.BNK.INFO-TIME)
	`WITH`
	` SUBR("S.NOTIFY.PERSON.ADDR.BNK.INFO",`
	` @ID) EQ "never matches"`

EMPLOYES
(SEC-EMPLOYES-DIR.DEP.CHGTIME)
	`WITH`
	` SUBR("S.NOTIFY.EMPLOYES.DIR.DEP",`
	` @ID) EQ "never matches"`

7. Click **Save All**.

Remove security notifications

You can remove all available security notifications if necessary.

Procedure

Run the EDX Interface Load Utility (EDIL) process to remove the Colleague Web API security notification components.

Warning! Be very careful when using EDIL. If you enter Yes with no subscriber ID, all existing EDX triggers are removed.

Field	**Entry**
Clear all existing trigger definitions	`Yes`
Subscriber Interfaces to Load	`SEC.NOTIFY`

Understand Colleague Web API Caching and the Warm Up Script

Data caches are not retained when the Colleague Web API application pool is recycled. Therefore, Ellucian recommends that IT administrators consider running the delivered warm up script after the scheduled recycle time for the application pool.

Caching and caching timeouts

Products that use Colleague Web API will cache relatively static data for a specified period of time, which saves having to retrieve this data from the database and decreases processing time.

Colleague Web API uses caching for data such as validation code tables, code files, and parameter records, but also caches other Colleague data such as grades and buildings.

The following list shows the caching duration of domain entities that are cached in Colleague Web API, grouped according to their cache duration.

- Cached for 24 hours
  - Books
  - Catalog (Catalog Years)
  - Counties
  - Countries
  - CourseEquates
  - CourseParameters
  - CourseStatuses
  - DueDateOverrides
  - EducationGoals
  - FinancialPeriods
  - GeneralLedgerConfiguration (parameters from ACCT.STRUCTURE)
  - InternationalParameters (parameters from INTL.PARMS)
  - InvoiceTypes
  - NonCourseCategories
  - PlanningConfiguration (parameters from STWEB.DEFAULTS)
  - ProgramStatuses
  - RegistrationDefaults
  - ResidenceLifeConfiguration (parameters from RL.EXT.SYS.DEFAULTS)
  - SampleDegreePlans (curriculum tracks)

  - SectionStatuses
  - SessionCycles
  - StaffStatuses
  - StaffTypes
  - States
  - StudentAcademicCreditStatuses
  - StudentConfiguration
  - StudentProgramStatuses
  - WaitlistStatuses
  - YearlyCycles
- Cached for 4 hours: EventTypes
- Cached for 20 minutes: ArchivedSections
- Cached for 15 minutes
  - Permissions
  - Roles
- Cached for 5 minutes: ImageMimeTypes
- Cached for 1 minute
  - StudentPeriodActivity
  - StudentTermActivity

Run the warm up script

The data caches maintained by Colleague Web API are not retained when its application pool is recycled.

About this task

Accordingly, there can be a noticeable delay in the responsiveness of applications that make use of the Colleague Web API. This may result in a poor user experience for whoever happens to access the application first. To alleviate this, IT administrators should consider running this script at least one time every 24 hours, during off-peak time or just after daily Colleague backup activities. The script can be run more frequently as well.

The script can be used with or without an option that preforms a recycle of the Colleague Web API's IIS application pool. When using the -recycleAppPool option, the traditional recycling settings within IIS can be set to not recycle on periodic bases (on regular time interval or specific times) and to never time out (idle time-out) and a scheduled run of this script with the -recycleAppPool can be used instead to ensure that the application pool recycle and warm-up happen at the same time rather than trying to coordinate a scheduled task just after a periodic application pool recycle or worse not warming-up due to an idle time-out shutting-down the application pool. The suggested usage of the -recycleAppPool option is to create a scheduled task that runs one time a day, during off-peak time or just after Colleague backup activities are finished that uses the -recycleAppPool option and then, optionally if you want to run the warm-up periodically throughout the day, create another scheduled task that does not use the -recycleAppPool so that the application pool is not being recycled during the normal hours of the day.

To access the warm up script, follow these steps:

Procedure

1. Run the following InstallShield file (if you have not already done this), to set up or update your Colleague API web application: `ColleagueWebAPIxxSetup.exe` (where *xx* is the latest version)
2. In IIS, right-click the ColleagueApi web application that you have just created or updated, and then select **Explore**.
3. Locate the script named `WarmUp.ps1` in the bin directory. Review important instructions and information in the header comment block within the script.