

# **ADOBE® UPDATE SERVER SETUP TOOL**

## **TECHNICAL NOTE**

For legal notices, see [http://help.adobe.com/en\\_US/legalnotices/index.html](http://help.adobe.com/en_US/legalnotices/index.html).

# Contents

<b>Adobe Update Server Setup Tool</b> .....	2
Overview .....	2
Using AUSST to set up your own update server.....	2
What's new in AUSST 3.0 .....	3
.Setting up an update server: at a glance .....	4
Location of the Adobe Update Server Setup Tool.....	6
Preparing a web server to use as the update server.....	6
Performing One-Time Setup.....	6
Verifying the set up.....	7
Setting up Client Machines .....	8
Generating client configuration files.....	9
Deploying client configuration files .....	10
Synchronizing with the Adobe Update Server .....	10
Incremental Synchronization .....	10
Fresh Synchronization .....	10
Setting Up IIS Server for Use with AUSST .....	10
Setting up IIS Server 6 .....	11
Setting up IIS Server 7 .....	15
Troubleshooting.....	18
Ensure that the web server is set up correctly .....	18
Check network connectivity .....	19
Ensure that there are no unwanted spaces in commands .....	19
Specify full URLs with protocol and port number .....	20
Ensure that the location for storing the updates has write permission .....	20
Ensure that the client configuration files are generated correctly on the in-house server .....	20
Ensure that the client configuration files are deployed correctly on the client machines .....	20
Ensure that paths specified are absolute paths .....	20
Use the fresh sync option if multiple updates are visible on client machines .....	21
Use the fresh sync option if other troubleshooting steps fail .....	21
Perform a fresh install as a last resort .....	21
Migrating from one in-house update server to another.....	21
Setup and synchronization .....	22
Updating client machines .....	22
Running AUSST command without specific parameters.....	22

# 1 Adobe Update Server Setup Tool

This document describes how to install and configure Adobe Update Server Setup Tool (AUSST), which is a utility to help you set up your own update server to manage the deployment of Adobe product updates in your enterprise.

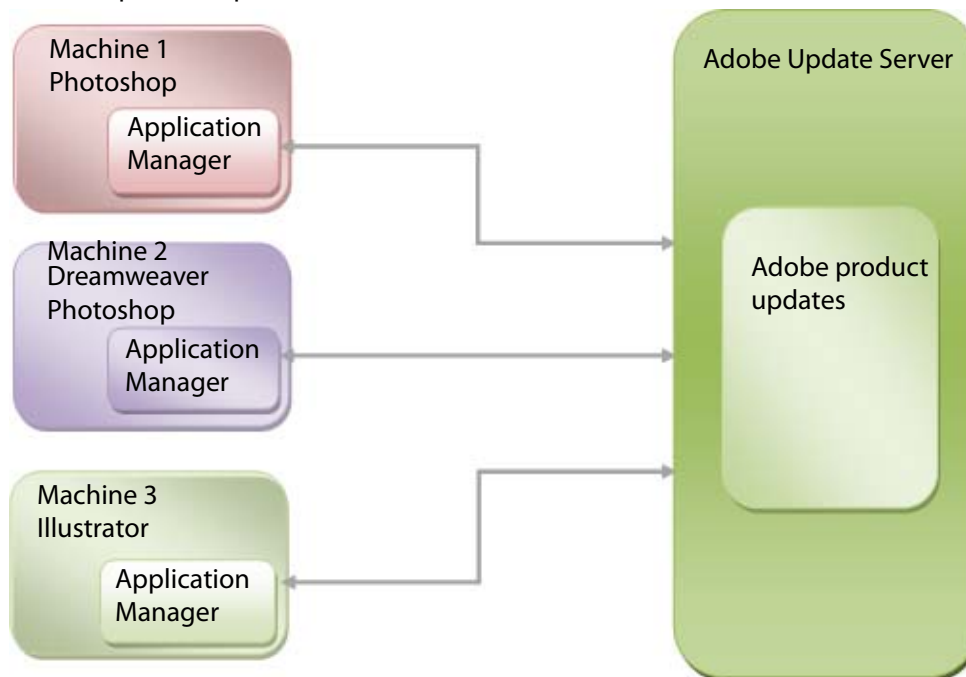
Updates are downloaded for both the Windows and the Mac OS platforms.

**NOTE:** AUSST is intended for the enterprise IT administrator who sets up and maintains the in-house update server.

## Overview

In an individual product installation, the Adobe Application Manager launches automatically every day at 2:00 am to check with the Adobe Update Server for updates to Adobe products. If a product update is found, the application displays a message. The user can then choose to download and install the update.

Users can also use the Help ->Updates menu command to invoke the Adobe Application Manager to check for product updates, and, if one is found, download and install it.

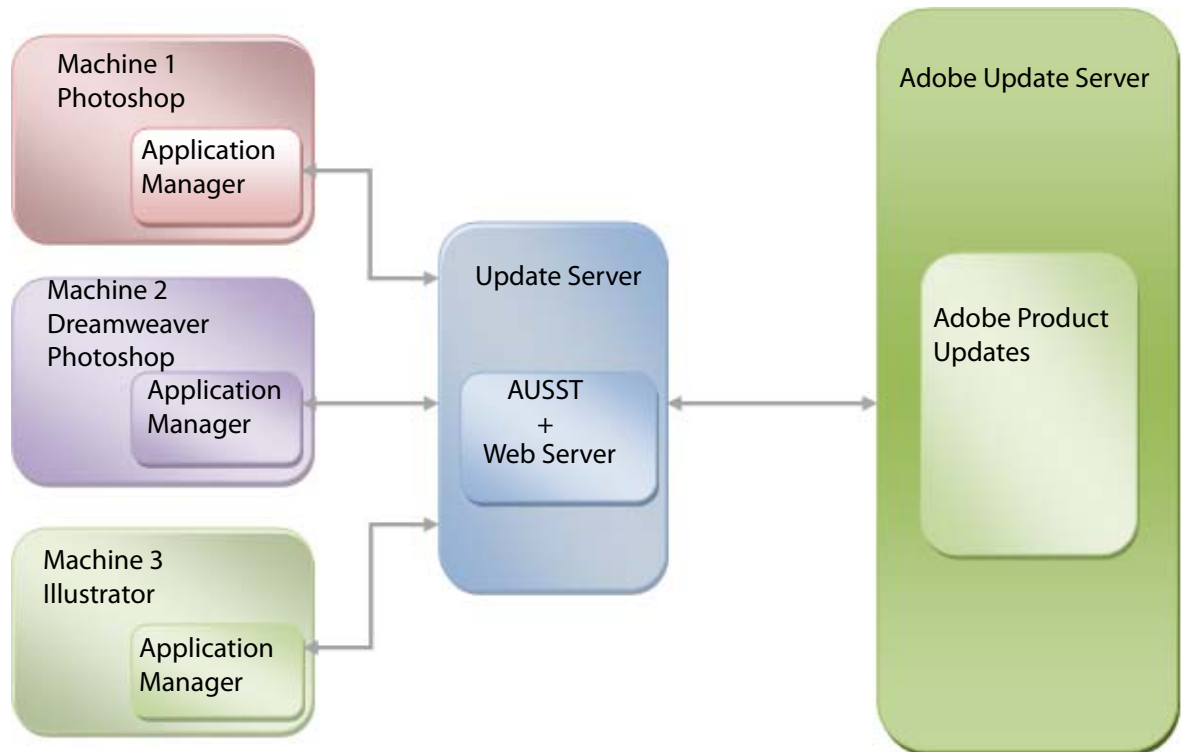


Client machines download product updates directly from the Adobe update server

## Using AUSST to set up your own update server

In an enterprise environment, you might prefer to host your own update server to download and store updates from the Adobe Update server.

The Adobe Update Server Setup Tool (AUSST) helps you to configure your own update server and download product updates from the Adobe Update Server to it. Once you have set up your own update server, you can redirect the Adobe Application Manager to your own server, rather than the Adobe Update Server. When you do this, the Adobe Application Manager on the client machines checks for updates on your server, and when instructed, downloads the updates from your server.



Using AUSST, you can set up your own update server, from which the client machines download updates.

**NOTE:** The purpose of AUSST is to help you host an in-house update server so that the client machines check for updates on the in-house server instead of the Adobe update server. AUSST is not currently intended for remotely deploying updates on machines in your network.

## What's new in AUSST 3.0

AUSST 3.0 provides the following enhancements over the previous version (AUSST 2.0) that was shipped with Adobe Application Manager Enterprise Edition 3.1.

- ◆ There is a major performance improvement—for example, AUSST 3.0 synchronizes the updates up to four times faster than the previous version for fresh synchronization.
- ◆ Support for proxy servers has been added.
- ◆ You can automate or schedule AUSST runs -- AUSST now includes options to run the command without any manual intervention at run time.
- ◆ Download progress is displayed real-time on the console.

- ◆ AUSST now provides a log file that includes status and troubleshooting information.

To run AUSST, you will need to make the following change to the AUSST 2.0 setup:

- ◆ The name of the tool has changed from `AdobeUpdateServerSetupTool2.0` to `AdobeUpdateServerSetupTool`. You should, therefore, update your previous scripts accordingly.

## Setting up an update server: at a glance

Here are the main steps for setting up your in-house update server using AUSST. You will need an already up and running http server to use as the update server.

**NOTE:** AUSST is available in the `utilities` folder of Adobe® Creative Cloud™ Packager installation.

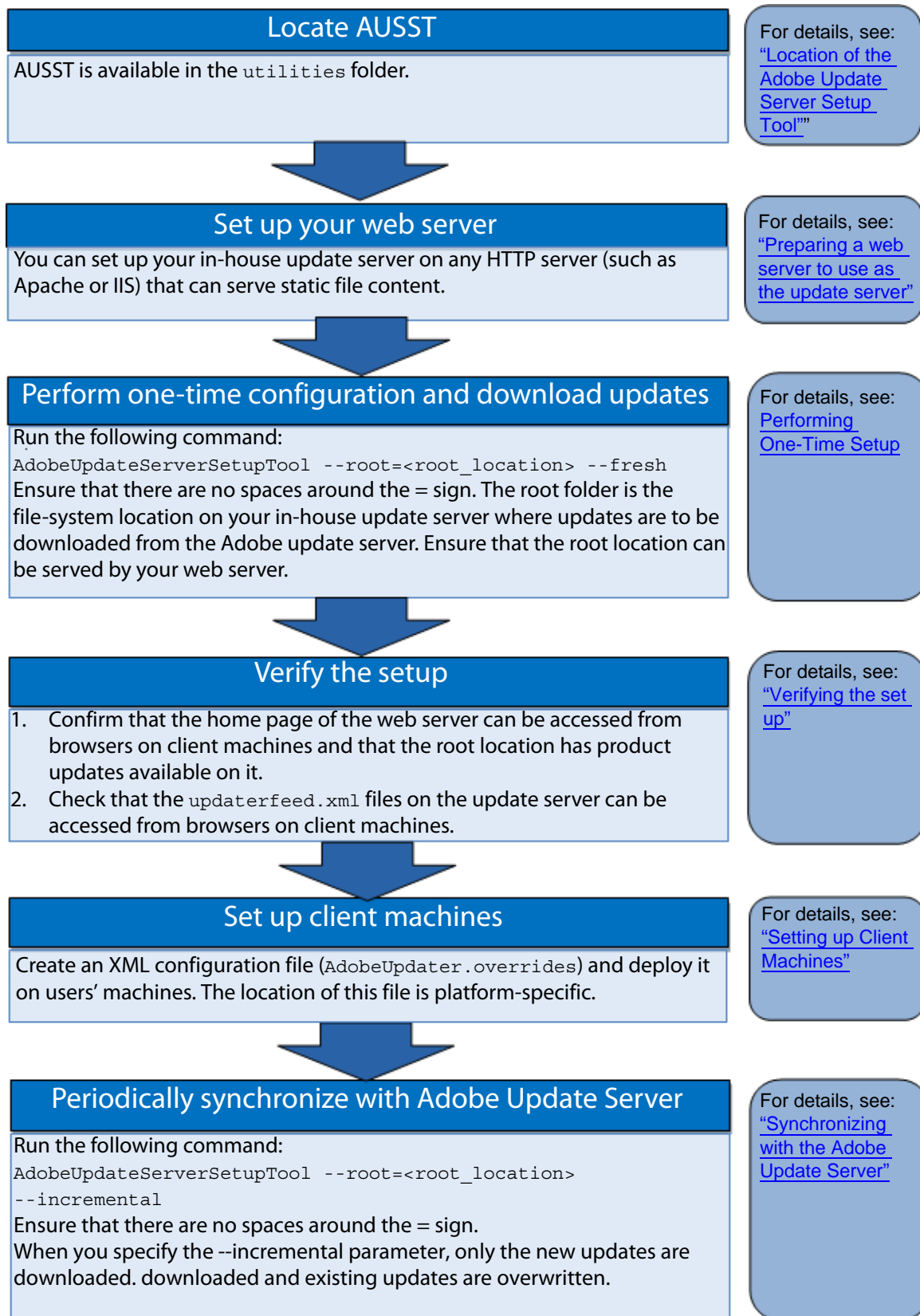
1. Ensure that a web server is available
2. Perform one-time setup using AUSST:

As part of the one-time setup, AUSST does the following:

- Performs the initial configuration
  - Sets up a directory structure similar to the one on the Adobe update server
  - Copies updates from Adobe update server to the web server. Both Windows and Mac OS updates will be downloaded.
3. Periodically, synchronize the latest updates from the Adobe update server to the web server. Perform this step regularly to ensure that your in-house server hosts the latest updates.

These steps are described in the next sections. The following diagram illustrates the process and provides links to the sections in this document where the corresponding steps are explained.

**NOTE:** The paths provided to all the command-line options should be absolute paths. AUSST does not support relative paths.



## Location of the Adobe Update Server Setup Tool

AUSST is available as .exe (Windows) or UNIX executable (Mac OS) file.

If you use Creative Cloud Packager to deploy Adobe products, AUSST is made available as part of the Creative Cloud Packager installation. Here are the locations:

In Windows 32-bit	<system drive>:\Program Files\Common Files\Adobe\OOBE\PDApp\CCP\utilities\AUSST
In Windows 64-bit	<system drive>:\Program Files (x86)\Common Files\Adobe\OOBE\PDApp\CCP\utilities\AUSST
In Mac OS	/Library/Application Support/Adobe/OOBE/PDApp/CCP/utilities/AUSST

AUSST is a command line tool and needs no separate installation steps. There are no restrictions on where AUSST should be located on the machine.

## Preparing a web server to use as the update server

You can set up your in-house update server on any HTTP server (such as Apache or IIS) that can host and serve static file content. An already up and running http server is a pre-requisite for setting up AUSST.

To use a specific port, configure the port number while setting up the client machines. You will only need to provide the port number in the client configuration file (overrides file).

If you use Internet Information Services (IIS) Server as your web server, refer to the following section for configuring the IIS server for use as the update server.

- ◆ [Setting Up IIS Server for Use with AUSST](#)

## Performing One-Time Setup

To set up your in-house update server for the first time, you use the Adobe Update Server Setup tool to create an update directory structure at the root folder location you created. If any files or folders exist at that location, the tool removes them. It then creates the folder structure that matches that of the Adobe update server, and performs the initial synchronization that downloads all available updates from the Adobe update server to your in-house server.

Here are the steps to run the tool for first-time configuration:

1. Run the tool in a command shell or terminal, specifying your root update folder. For example:

```
AdobeUpdateServerSetupTool
  --root=<root_folder_location>
  --fresh
  [--proxyusername=<proxy_username> --proxypassword=<proxy_password>]
  [--silent]
```

- The `--fresh` parameter removes any existing files from the root location (if it exists) and then downloads all the updates present on the Adobe Update Server.



- The `[--proxyusername=<proxy_username> --proxypassword=<proxy_password>]` parameter is optional and specifies the proxy username and password if a proxy server is required to connect to the network.

The `--silent` parameter *suppresses* the warning that is displayed by default in case the root folder specified already exists. **NOTE:** Ensure that there are no spaces around the `=` sign.

---

The root folder is the file-system location on your in-house update server where updates from the Adobe update server are stored. The root folder location must map to a valid HTTP URL.

**NOTE:** Ensure that the root folder lies within the access of the server root location so that update content can be served by the web server.

As an example, suppose:

- ◆ The root update folder on your web server is at the file-system location `/serverroot/updates/`
- ◆ The web server's URL is `http://serverabc.example.com:80`
- ◆ Within your web server, you set up the in-house update server at

`http://serverabc.example.com:80/Adobe/CS`

In this case (which we will use as an example throughout this document), the root folder location is

```
--root="/serverroot/updates/Adobe/CS"
```

For this example, the command to run the updates will be:

```
AdobeUpdateServerSetupTool --root="/serverroot/updates/Adobe/CS" --fresh
```

---

AUSST creates a directory structure for the updates in the root folder and then copies the updates from the Adobe update server to your in-house update server.

**NOTE:** Both Windows and Mac OS updates are downloaded. Currently, you cannot selectively download updates for only one platform.

As soon as you have completed this initial setup, your in-house update server is ready to update client machines.

**NOTE:** Errors, warnings, and troubleshooting information are recorded in the log file. The log file is located in the `%temp%` folder in Windows and in the `~/Library/Logs` in Mac OS.

## Verifying the set up

To confirm that the first-time configuration is successful, check the following

1. Ensure that the web server is running correctly — confirm that the home page of the web server can be accessed from a client machine.
2. Confirm that the root location has the Adobe product updates available on it. If not, check that the root location has the correct write permissions.
3. Confirm that you can view/download the updates from the client machines through a browser.
4. Check that the `updaterfeed.xml` files on the update server can be accessed from browsers on the client machines.

The `updaterfeed.xml` files are located at the following path, determined by the parameters in the overrides file:

```
http://<Domain>:<Port>/<URL>/updaterfeed.xml
```

For example, let us assume that the override file has the following entries

```
<Overrides>
  <Application appID="webfeed">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="webfeed20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
</Overrides>
```

In this case, the location of the `updaterfeed.xml` file will be as follows:

```
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam10/win/updaterfeed.xml
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam20/win/updaterfeed.xml
```

The location on Mac OS will be:

```
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam10/mac/updaterfeed.xml
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam20/mac/updaterfeed.xml
```

**NOTE:** When you view the `updaterfeed.xml` file in a browser, you will not be able to view the content in the file. This is OK; you only need to ensure that the file is accessible through the `http://` path. You can view the content of the `updaterfeed.xml` file by right-clicking and viewing the source.

## Setting up Client Machines

The Adobe Application Manager on client machines, by default, checks for updates on the Adobe update server. When you host your own update server, you must configure the Adobe Application Manager on each user's machine to check for updates on your own server instead.

To do this, you must create an XML configuration file (`AdobeUpdater.overrides`) and deploy it on the users' machines. You can create the client configuration files in a text editor, or use the tool to generate them automatically as explained in the next section. The configuration file provides the domain, URL and port information for your in-house update server; the URL is different for Windows and Mac OS platforms. The following shows the format of the `AdobeUpdater.Overrides` file, using our example server information (see ["Performing One-Time Setup"](#) for the example server information):

**NOTE:** If you migrated from one version of AUSST to another, you must update the XML configuration files on the client machines.

#### IN WINDOWS:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Overrides>
  <Application appID="webfeed">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="webfeed20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
</Overrides>
```

#### IN MAC OS:

In Mac OS, the URL specifies the `mac/` subfolder instead of the `win/` subfolder:

```
...
<URL>/Adobe/CS/webfeed/oobe/aam10/mac/</URL>
...
<URL>/Adobe/CS/webfeed20/oobe/aam20/mac/</URL>
...
<URL>/Adobe/CS/updates/oobe/aam10/mac/</URL>
...
<URL>/Adobe/CS/updates20/oobe/aam20/mac/</URL>
...
```

## Generating client configuration files

You can generate the client configuration files automatically with AUSST. You can only do this after you have set up your server and synchronized it with the Adobe update server.

To generate the configuration files, enter this command in a command shell or terminal (using your own server information):

```
AdobeUpdateServerSetupTool
--genclientconf="/serverroot/config/AdobeUpdaterClient"
--root="/serverroot/updates/Adobe/CS"
--url="http://serverabc.example.com:1234/Adobe/CS"
```

This command creates two client configuration files (one for Windows platform, and one for Mac OS platform), and writes them into platform-specific folders under the path given in `--genclientconf`.

In this example, the new files will be:

```
/serverroot/config/AdobeUpdaterClient/win/AdobeUpdater.Overrides  
/serverroot/config/AdobeUpdaterClient/mac/AdobeUpdater.Overrides
```

## Deploying client configuration files

To deploy the client configuration file for the Adobe Application Manager Updater on each client machine, you must write the correct platform version of the file to the following platform-specific location:

### IN WINDOWS XP:

```
\Documents and Settings\All Users\Application Data\Adobe\AAMUpdater\1.0\AdobeUpdater.Overrides
```

### IN WINDOWS 7/VISTA:

```
\ProgramData\Adobe\AAMUpdater\1.0\AdobeUpdater.Overrides
```

### IN MAC OS X:

```
/Library/Application Support/Adobe/AAMUpdater/1.0/AdobeUpdater.Overrides
```

## Synchronizing with the Adobe Update Server

After you initial setup, you will need to regularly synchronize your in-house update server with the Adobe Update Server, to make sure you have all the latest updates. You do this with the following command:

```
AdobeUpdateServerSetupTool --root="/serverroot/updates/Adobe/CS" {--incremental | --fresh}
```

### Incremental Synchronization

The `--incremental` parameter downloads only those updates for which a more recent version is available.

If there are new updates posted on the Adobe update server since your last synchronization, specifying the `--incremental` parameter pulls only the new updates to the local server. If there are no new updates available, the command does nothing.

In normal course, this is the option that you should use. However, if you find that if for some reason the latest updates are not being reflected on the client machines, you can perform a fresh synchronization as explained next.

### Fresh Synchronization

The `--fresh` parameter downloads all updates present on the Adobe Update Server.

Any updates present on the machine are first removed. Then, all the updates available on the Adobe Update Server are downloaded to the machine.

In either case, the result is that your in-house update server is up-to-date, and ready to update client machines.

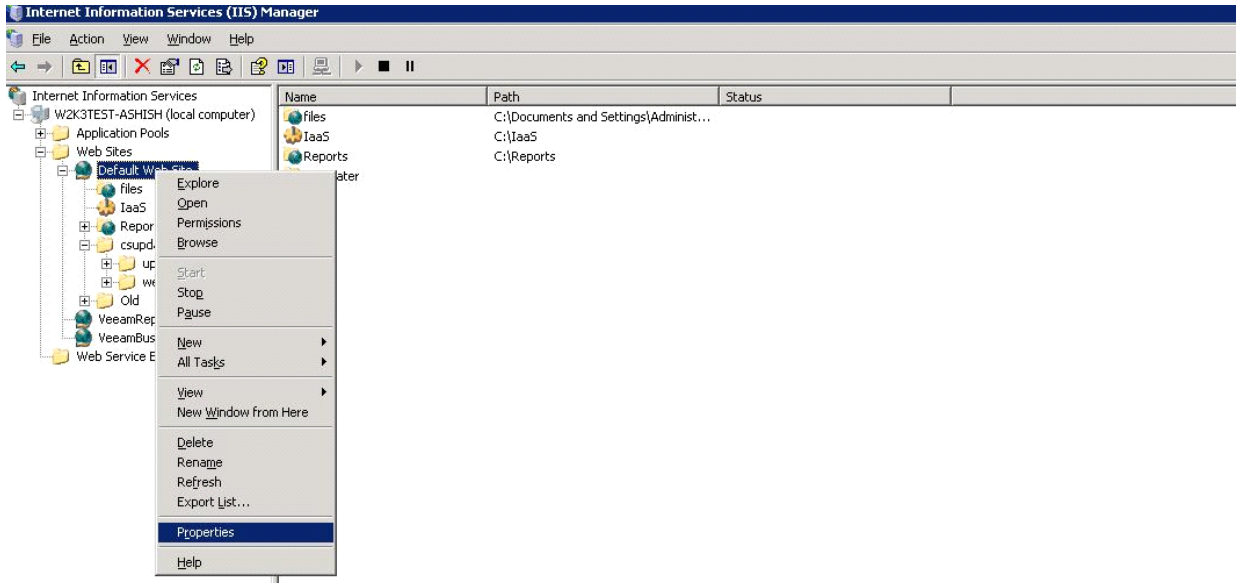
## Setting Up IIS Server for Use with AUSST

This section describes how to set up Internet Information Services (IIS) Server for use with AUSST.

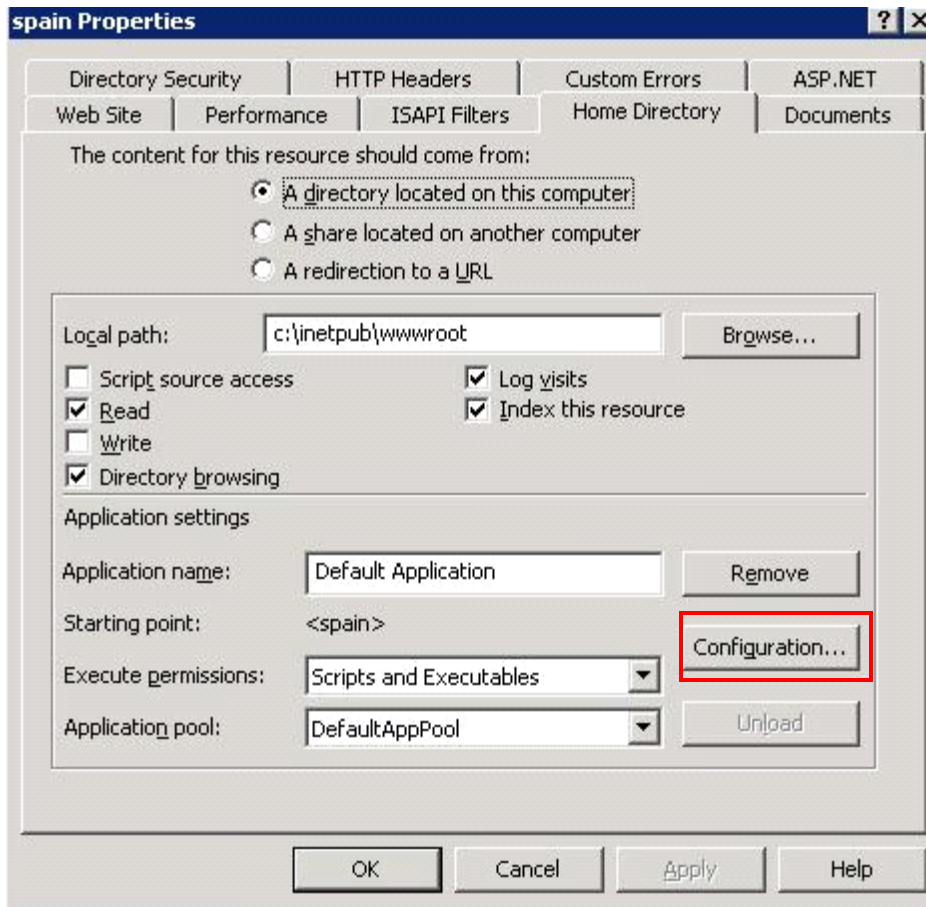
- ◆ [Setting up IIS Server 6](#)
- ◆ [Setting up IIS Server 7](#)

## Setting up IIS Server 6

1. Run AUSST and synchronize the updates available on the Adobe update server at default website root.
2. Configure website properties as shown here:



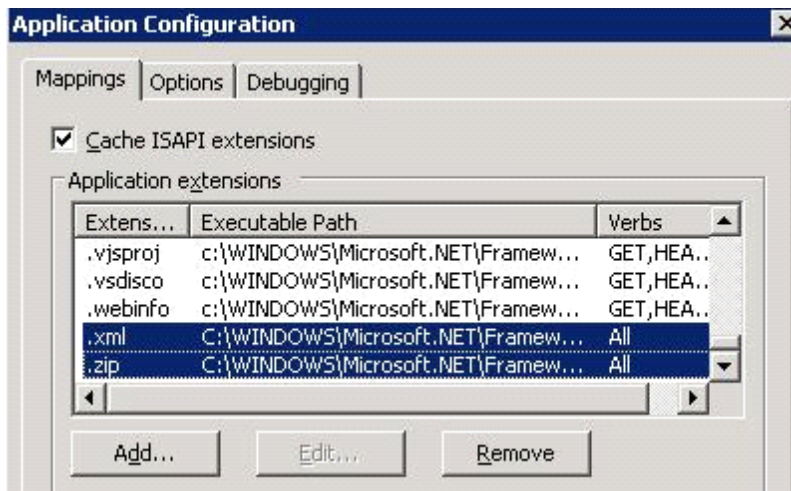
3. Select the HomeDirectory tab and then click Configuration



**IMPORTANT:** The configuration changes are applied to all data referring to this (default in this example) website (for example, laas, Reports, Old). You should, therefore, create a separate website for updaterelevant data and apply these configuration related changes to this separate website, so that the other sites are not affected.

4. . Add the ISAPI extension for following extensions:

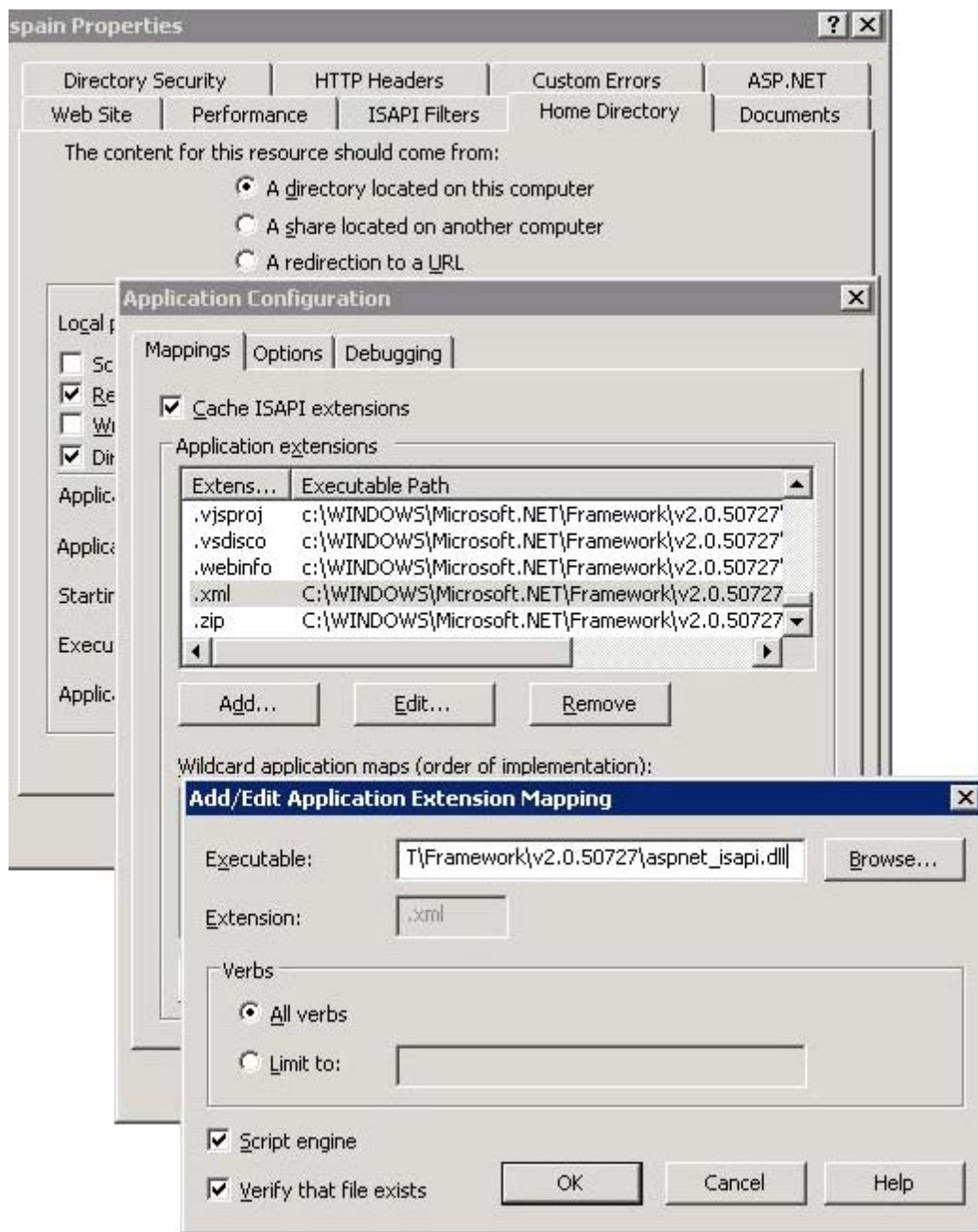
- xml
- zip
- dmg
- sig
- crl



5. Perform extension mapping for the following executable:

C:\Windows\Microsoft.NET\Framework\v2.0.50727\aspnet\_isapi.dll

as shown here:

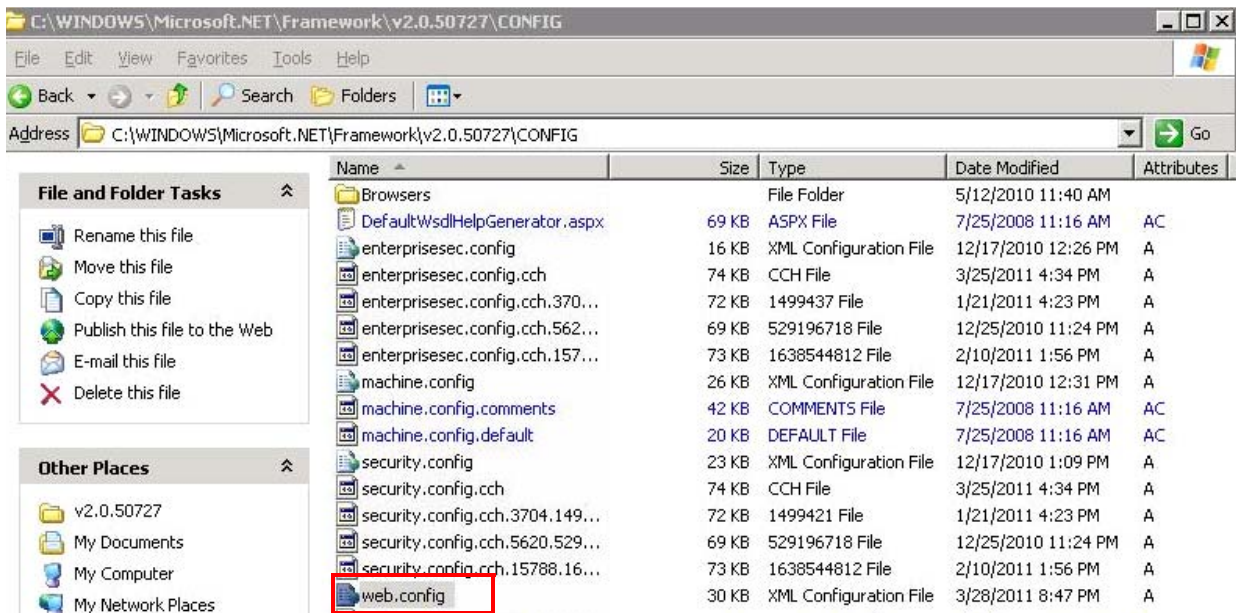


**NOTE:** Note: For 64bit-OS, perform the mapping in the 64-bit framework path as follows:

C:\Windows\Microsoft.NET\Framework64\v2.0.50727\aspnet\_isapi.dll.

6. Open web.config file in the framework directory as shown below:





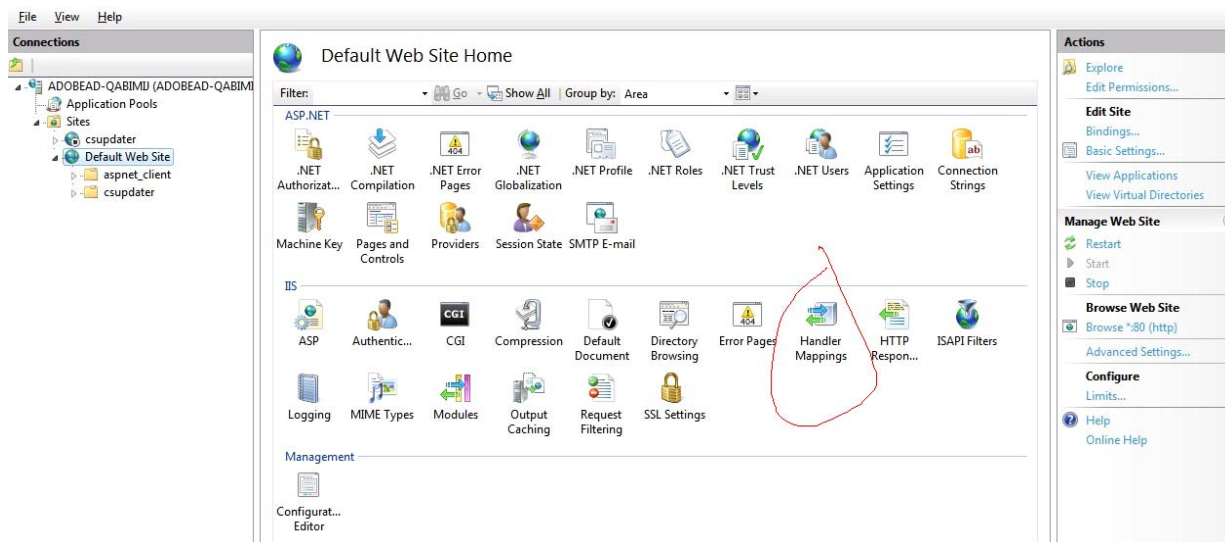
7. Add entries for the xml, crl, dmg, zip, and sig extensions as shown here:

```
<httpHandlers>
  <add path="*.zip" verb="*" type="System.Web.StaticFileHandler" />
  <add path="*.xml" verb="*" type="System.Web.StaticFileHandler" />
```

8. Restart the website and run AUSST.

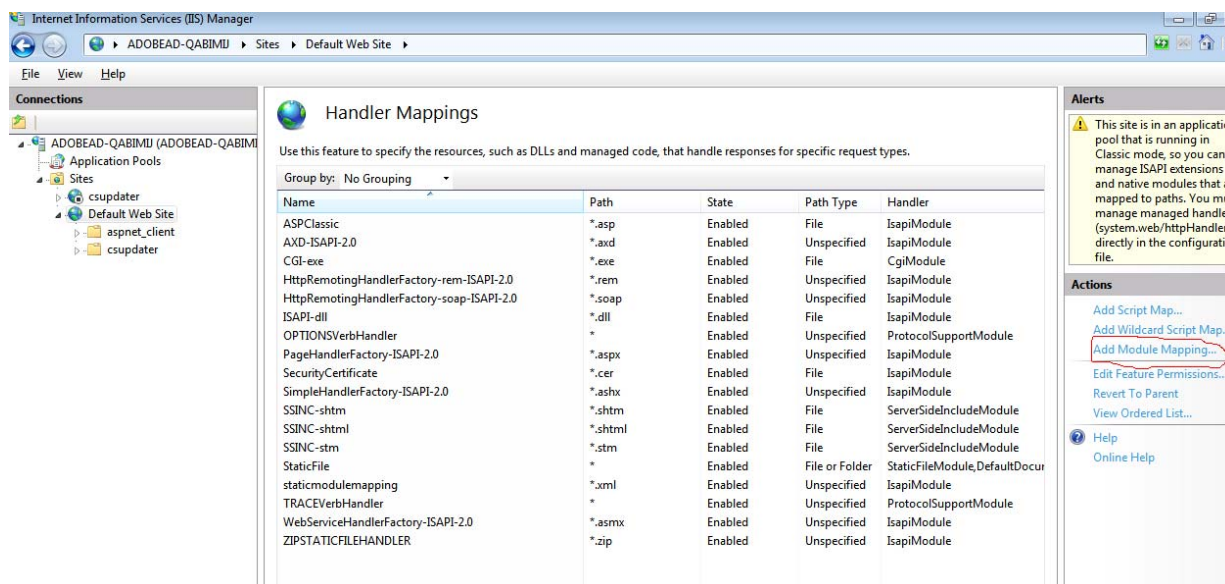
## Setting up IIS Server 7

1. Run AUSST and synchronize the updates available on the Adobe update server at default website root.
2. Open inetmgr and click the handler Mapping of the required website as shown here



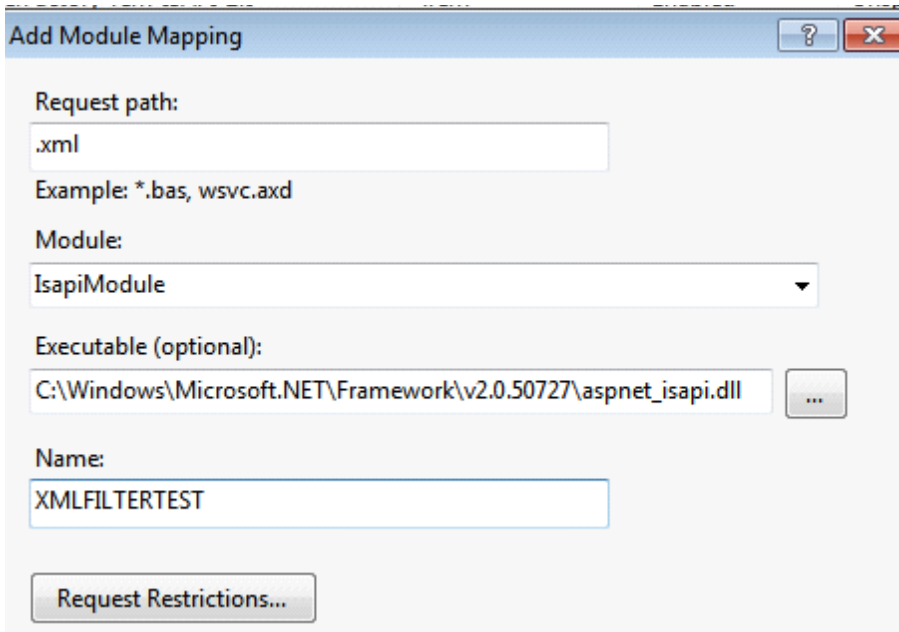
**IMPORTANT:** The configuration changes are applied to all data referring to this (default in this example) website. You should, therefore, create a separate website for updater- related data and apply these configuration related changes to this separate website, so that the other sites are not affected.

3. Select the Add Module Mapping option as shown here:

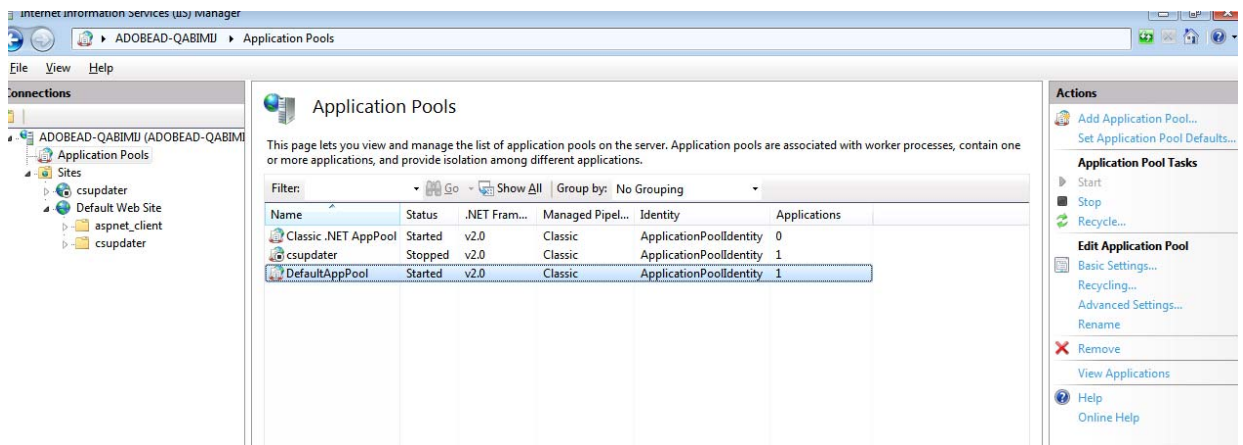


4. Add Module mapping for xml, crl, zip, dmg and sig extensions. A sample dialog box for .xml is shown here.

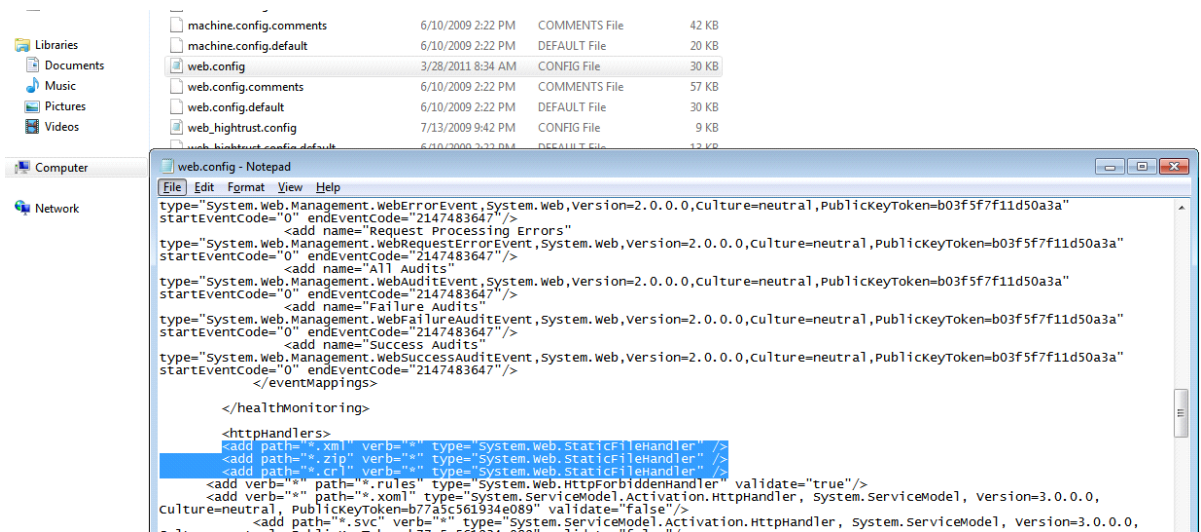
**NOTE:** The mapping for crl, zip, dmg and sig extensions can be performed similar to as shown here for the .xml extension.



5. In the Application pool section change the App Pool Manager Pipeline mode to Classic as shown here:



6. Add the httpHandles for .the zip, xml, crl, dmg, and sig extension in the web.config file as shown here:



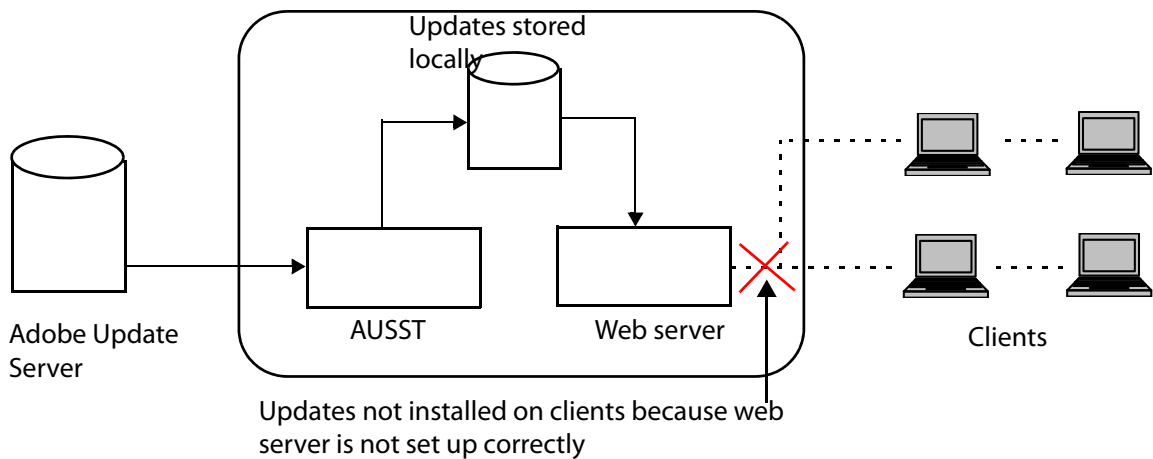
- Restart the website and run AUSST.

## Troubleshooting

Here are some common troubleshooting steps to follow if you face problems in deploying updates through Adobe Update Server Setup Tool.

### Ensure that the web server is set up correctly

If the web server that is used to distribute the updates is not set up correctly, updates might get downloaded from the Adobe update server (Adobe Update Server Setup Tool does that automatically), but will not be distributed to the clients.



As a simple test to check that the web server is running correctly, check that the `updaterfeed.xml` files on the update server can be accessed from browsers on the client machines. Essentially, you need to ensure that the web server is working, independent of the Adobe Update Server Setup Tool.

The `updaterfeed.xml` files are located at the following path, determined by the parameters in the `overrides` file:

```
http://<Domain>:<Port>/<URL>/updaterfeed.xml
```

For example, let us assume that the override file has the following entries

```
<Overrides>
  <Application appID="webfeed">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="webfeed20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam10/win/</URL>
    <Port>1234</Port>
  </Application>
  <Application appID="updates20">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/aam20/win/</URL>
    <Port>1234</Port>
  </Application>
</Overrides>
```

In this case, the location of the `updaterfeed.xml` files will be as follows:

```
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam10/win/updaterfeed.xml
```

```
http://serverabc.example.com:1234/Adobe/CS/webfeed/oobe/aam20/win/updaterfeed.xml
```

## Check network connectivity

Ensure that there are not problems related to network connectivity and that the in-house server can connect to the Adobe update server. For example, check that you can access the default welcome page of the in-house server.

## Ensure that there are no unwanted spaces in commands

When you run the `AdobeUpdateServerSetupTool` command, ensure that there are no spaces in the following cases:

- ◆ Between parameters and the `=` sign
- ◆ Between the `=` sign and arguments

For example, the following command has an incorrect extra space between `--root` and the `=` sign

```
--root ="/serverroot/updates/Adobe/CS"
```




Extra space that should be removed

- ◆ Anywhere in a path.

For example, the following command has an extra space between the quotation mark (") and the forward slash (/)

```
--root=" /serverroot/updates/Adobe/CS"
```


  
Extra space that should be removed

## Specify full URLs with protocol and port number

The URLs for the server should include the protocol (such as `http://`). If the port number is different from the default value of 80, the port number should also be specified.

For example, the following section in the `AdobeUpdater.Overrides` file is incorrect because it does not specify the `http://` protocol:

```
<Domain>serverabc.example.com</Domain>
```

  
`http://` should be added here  
<URL>/Adobe/CS/webfeed/oobe/aam10/win/</URL>  
<Port>1234</Port>

Note that in this example, the port number (1234) has been specified, which is required if the port number does not have the default value of 80.

## Ensure that the location for storing the updates has write permission

Ensure that the location on the server where the updates are to be stored has the correct write permission. Otherwise, updates might not be synced/downloaded on the in-house server. The server setup process is not complete unless all updates are synced/downloaded on the in-house server

## Ensure that the client configuration files are generated correctly on the in-house server

The client configuration files are generated at a location determined by the `-genclientconf` parameter of the `AdobeUpdateServerSetupTool` command as explained in the section [“Generating client configuration files”](#) of this document. Two files are generated, one each for Windows and Mac-OS clients. Check that the files are available in the respective locations on the in-house server.

## Ensure that the client configuration files are deployed correctly on the client machines

After the client configuration files are generated on the in-house server, they are deployed on each client machine. The files are different for the Windows and the Mac-OS platforms.

The location for the file on each client machine is platform-specific. Ensure that the configuration files are deployed on the client machines at the locations as explained in the section [“Deploying client configuration files”](#) of this document.

## Ensure that paths specified are absolute paths

Check that the paths provided to all the command-line options are absolute paths. AUSST does not support relative paths.

## Use the fresh sync option if multiple updates are visible on client machines

In exceptional cases, multiple updates of the same package might get stored on the in-house server if the option of incremental update sync is specified. These will then be deployed to the client machines, and the users will see multiple copies of the packages on their machines. If this happens, perform the fresh update sync (one time) by running the following command:

```
AdobeUpdateServerSetupTool --root="/<update_folder>" --fresh
```

This ensures that the correct single copy of the packages get downloaded to your in-house server, from where they are deployed to the client machines.

## Use the fresh sync option if other troubleshooting steps fail

If you are using the incremental sync option and continue to get errors even after trying all the previous steps, perform fresh update sync (one time) by running the following command:

```
AdobeUpdateServerSetupTool --root="/<update_folder>" --fresh
```

This will ensure that all updates from the Adobe update server are downloaded to your in-house server.

## Perform a fresh install as a last resort

If all other troubleshooting steps fail, as a last resort perform a fresh install by running the following command:

```
AdobeUpdateServerSetupTool --root="/<update_folder>" --fresh
```

**NOTE:** When you perform a fresh install, the update folder and all its subfolders on your in-house server will be first deleted and then recreated with the updates from the Adobe update server. If you had created any additional files or folders in the update folder, they will be deleted. You should, therefore, back up any such files or folders before performing a fresh install.

# Migrating from one in-house update server to another

Occasionally, you might find it necessary to migrate from one server that is already set up as your in-house update server to a different server.

In this case, you should not need to repeat the download of all the update data that had already been downloaded to the original server. To save network bandwidth and download time, the Adobe Update Server Setup Tool provides an override facility to copy the data from your original server to your new server.

For example, suppose the original in-house HTTP server was set up according to our example:

- Server root hosted at file system location `"/serverroot/updates/"`
- URL is `http://serverabc.example.com:80`
- Update server set up at `http://serverabc.example.com:80/Adobe/CS`

Suppose you want to switch to a new in-house HTTP server with:

- Server root hosted at file system location `"/newserverroot/newupdates/"`,
- URL is `http://newserverabc.example.com:80`

- Update server set up at `http://newserverabc.example.com:80/Adobe/CS`

In the initial setup and synchronization commands, you will now specify the root folder location as:

```
--root="/newserverroot/newupdates/Adobe/CS"
```

For initial setup of the new server, you must supply the `--overrides` option, providing the path to an XML file that identifies the previous in-house update server. This could be, for example, a file named `UpdateServerMigration.xml` placed at `/newserverroot/newupdates/migrationfiles/`.

Although this file is similar to the Adobe Application Manager updater configuration file, it is not identical. In addition to having a different name and being used in a different context, it is not platform-specific; the URLs do not have the `/win` or `/mac` subfolder at the end.

For our example, the `UpdateServerMigration.xml` file should look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Overrides>

  <Application appID="webfeed">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/webfeed/oobe/</URL>
    <Port>80</Port>
  </Application>
  <Application appID="updates">
    <Domain>http://serverabc.example.com</Domain>
    <URL>/Adobe/CS/updates/oobe/</URL>
    <Port>80</Port>
  </Application>
</Overrides>
```

## Setup and synchronization

To perform the initial setup of this new server, you would invoke this command:

```
AdobeUpdateServerSetupTool
--root="/newserverroot/newupdates/Adobe/CS"
--overrides="/newserverroot/newupdates/migrationfiles/UpdateServerMigration.xml"
--fresh
```

The tool will automatically migrate the data to the new server, which will then be ready to use. All subsequent synchronization operations will use the new root value, but will be otherwise identical. That is, invoke this command:

```
AdobeUpdateServerSetupTool --root="/newserverroot/newupdates/Adobe/CS" --incremental
```

## Updating client machines

To ensure that user machines begin updating from the new in-house update server, you must deploy a new version of the `AdobeUpdater.overrides` file containing the information for the new server. See ["Setting up Client Machines"](#).

## Running AUSST command without specific parameters

You can run AUSST from the command line by specifying only the `--root` parameter. In this case, a menu is displayed from which you can make your choice:

```
AdobeUpdateServerSetupTool.exe --root=<root location>
```



Choose one of the following options :

1. Fresh Synchronization.
2. Incremental Synchronization.
3. Generate Client Configuration XML.
4. Migrate to an In-House Server.
5. Exit.

Your choice:

You can select one of the options to proceed further. The details of these options are available earlier in the article.