Infor PLM Accelerate 11

File Handling

Important Notices

The material contained in this publication (including any supplementary information) constitutes and contains confidential and proprietary information of Infor.

By gaining access to the attached, you acknowledge and agree that the material (including any modification, translation or adaptation of the material) and all copyright, trade secrets and all other right, title and interest therein, are the sole property of Infor and that you shall not gain right, title or interest in the material (including any modification, translation or adaptation of the material) by virtue of your review thereof other than the non-exclusive right to use the material solely in connection with and the furtherance of your license and use of software made available to your company from Infor pursuant to a separate agreement, the terms of which separate agreement shall govern your use of this material and all supplemental related materials ("Purpose").

In addition, by accessing the enclosed material, you acknowledge and agree that you are required to maintain such material in strict confidence and that your use of such material is limited to the Purpose described above. Although Infor has taken due care to ensure that the material included in this publication is accurate and complete, Infor cannot warrant that the information contained in this publication is complete, does not contain typographical or other errors, or will meet your specific requirements. As such, Infor does not assume and hereby disclaims all liability, consequential or otherwise, for any loss or damage to any person or entity which is caused by or relates to errors or omissions in this publication (including any supplementary information), whether such errors or omissions result from negligence, accident or any other cause.

Without limitation, U.S. export control laws and other applicable export and import laws govern your use of this material and you will neither export or re-export, directly or indirectly, this material nor any related materials or supplemental information in violation of such laws, or use such materials for any purpose prohibited by such laws.

Trademark Acknowledgements

Publication Information

Publication date: May 31, 2018

Contents

About this guide

This document provides information about the installation and the Graphical User Interface of the Infor PLM Accelerate Batch Loader tool.

Intended audience

This guide is intended for system administrators. This document assumes that you have at least some knowledge of:

- System architecture and functions for your Infor system
- SQL Server database

For the most up-to-date list of software and hardware requirements for Infor products, see the documentation for your system.

Related documents

You can find the documents in the product documentation section of the Infor Xtreme Support portal, as described in the "Contacting Infor" section.

Contacting Infor

If you have questions about Infor products, go to the Infor Xtreme Support portal at www.infor.com/inforxtreme.

If we update this document after the product release, we will post the new version on this Web site. We recommend that you check this Web site periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

Chapter 1. Overview

The Infor PLM Accelerate 10.11.0 release introduced a new way of handling physical files. The File Items in Infor PLM Accelerate 10.11.0 are immutable. The following sections describe the File permission model, versioning design, UI, and History tracking.

The permission of a File Item is defined directly by the user's access to the File's container. The File container is the Item that has a Property of type Item with "File" as the data source. This includes the related_id Property of Relationship Items with File as the related ItemType.

Version control of File Items is also directly tied to the File container. The File ItemType itself is not versionable and the **update** action is not allowed for the File ItemType. In order to create a new version, the File's container should be versioned and the File Item replaced with a new File Item.

The File UI is consistent in all locations: the Form, the main grid, and the relationships grid. This UI gives options for viewing the File, downloading the File, and deleting the File from the Item's Property. It is important to note that deleting the File from the Item's Property will not automatically delete the File Item.

Chapter 2. File Permissions

The permissions of a File Item are dependent on the Item containing that File, known as the File container. The File container is the Item which has a Property with a data source of the File ItemType. This includes relationship Items which have their `related_id` pointing to a File Item. This does not include Image properties used to display thumbnails on a Form. If a File has more than one container, the permissions will be based on the least restrictive access among all containers of the file.

There are two system File container objects that are used when a File is not referenced by any properties with a data source of the File ItemType. These are the “Orphaned Files Container” and the “Global Files Container”.

Files that are added with no container will automatically be placed in the Global Files Container, which allows **Get** for the World Identity. In other words, all Infor PLM Accelerate users will have access to these Files. In a standard implementation, this includes all Files stored in properties of type Image. To restrict access to these files, they can be added to another Item via a Property of type Item. Once added to another container, the File will be automatically removed from the Global Files Container and the containing Item will thereby define the permissions applied to the File.

Files that are originally added with a containing Item, but are later orphaned in some way will automatically be placed in the Orphaned Files Container and will only be accessible to members of the File Administrators Identity. Members of the File Administrators Identity have unconditional access to all File Items. To restore access to these Files they can be added to another Item via a Property of type Item. Once added to another container, the File is automatically removed from the Orphaned Files Container and the containing Item will thereby define the permissions applied to the File.

In addition to the above permissions model, the following Variable Item may be defined to effect the global handling of files. This variable is not defined out of the box and a new Variable Item will need to be created if used. The Variable Item can be added in the TOC Administration\\Variables.

- `Force.Delete.Orphaned.Files`

  a) If the value is set to 1, all files which get added to the Orphaned Files Container will automatically be deleted from Infor PLM Accelerate and the Vault.

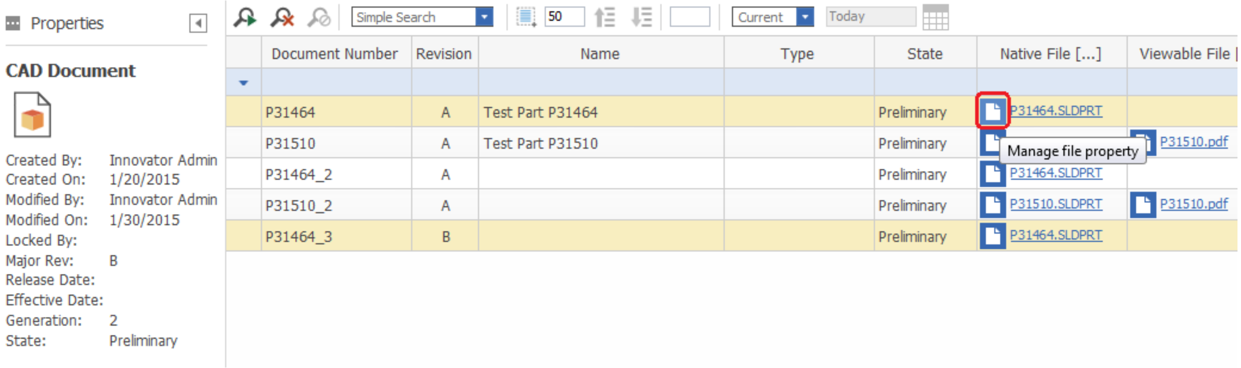
  b) If the value is set to 0 or if the Variable does not exist, all files which get added to the Orphaned Files Container will remain in the system.

Chapter 3. File Handling User Interface

This section describes the File Handling UI in all the locations it is seen and used. This UI is designed to be consistent across all locations where Files can be seen.

Main Grid

The following screenshot shows the properties with a data source of the File ItemType that are displayed in the main grid.

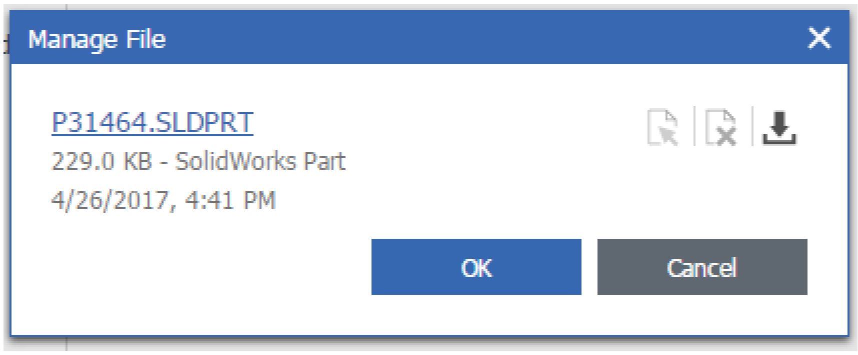


CAD Document

Created By: Innovator Admin
Created On: 1/20/2015
Modified By: Innovator Admin
Modified On: 1/30/2015
Locked By:
Major Rev: B
Release Date:
Effective Date:
Generation: 2
State: Preliminary

Document Number	Revision	Name	Type	State	Native File [...]	Viewable File
P31464	A	Test Part P31464		Preliminary	P31464.SLDPRT	
P31510	A	Test Part P31510		Preliminary		P31510.pdf
P31464_2	A			Preliminary	P31464.SLDPRT	
P31510_2	A			Preliminary	P31510.SLDPRT	P31510.pdf
P31464_3	B			Preliminary	P31464.SLDPRT	

Clicking on the name of the File will open the file within the browser and prompts you to “Open, Save, or Save As”. Clicking on the File icon 🗋 to the left of the file name in the main grid opens the **Manage File** dialog:

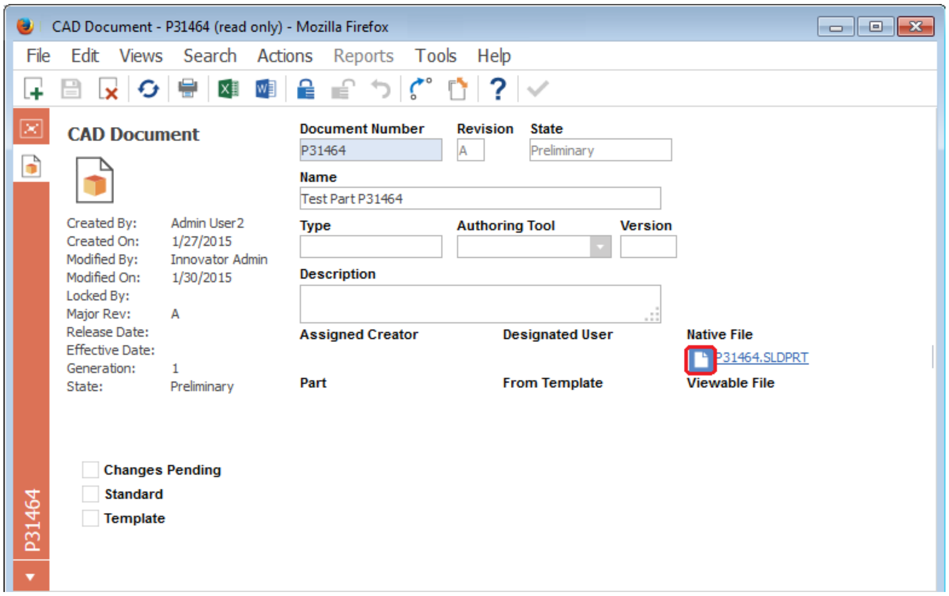


P31464.SLDPRT
229.0 KB - SolidWorks Part
4/26/2017, 4:41 PM

This dialog presents a similar File link as seen in the main grid as well as some additional information and buttons. You can choose to download the file to their downloads directory by clicking on the rightmost icon.

Item Window

The properties with a data source of the File ItemType that are displayed on the Item Form are seen as demonstrated by the Native File Property here:



CAD Document

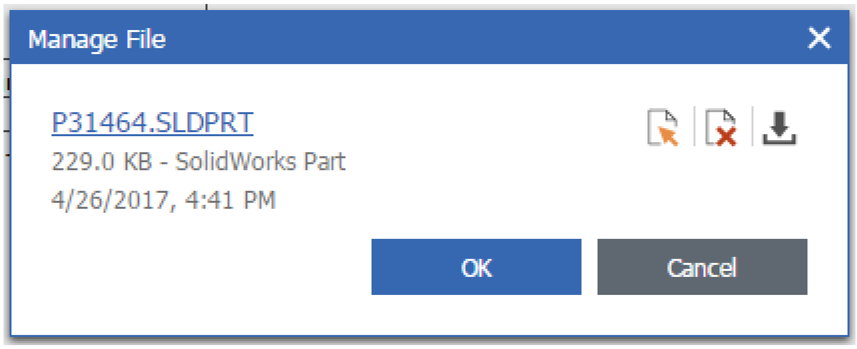
Created By: Admin User2
Created On: 1/27/2015
Modified By: Innovator Admin
Modified On: 1/30/2015
Locked By:
Major Rev: A
Release Date:
Effective Date:
Generation: 1
State: Preliminary

Document Number: P31464
Revision: A
State: Preliminary
Name: Test Part P31464
Type:
Authoring Tool:
Version:
Description:
Assigned Creator:
Designated User:
Native File: P31464.SLDPRT
Part:
From Template:
Viewable File:

□ Changes Pending
□ Standard
□ Template

This display is similar to the one seen in the main grid. Both the direct link to the File and the File icon are available. Clicking on the link streams the physical file to the browser to be handled while clicking on the File icon displays the Manage File UI.

If the Item is not locked, you see the same dialog as when you clicked the file icon in the main grid. When the containing Item is locked, additional buttons become available:



P31464.SLDPRT
229.0 KB - SolidWorks Part
4/26/2017, 4:41 PM

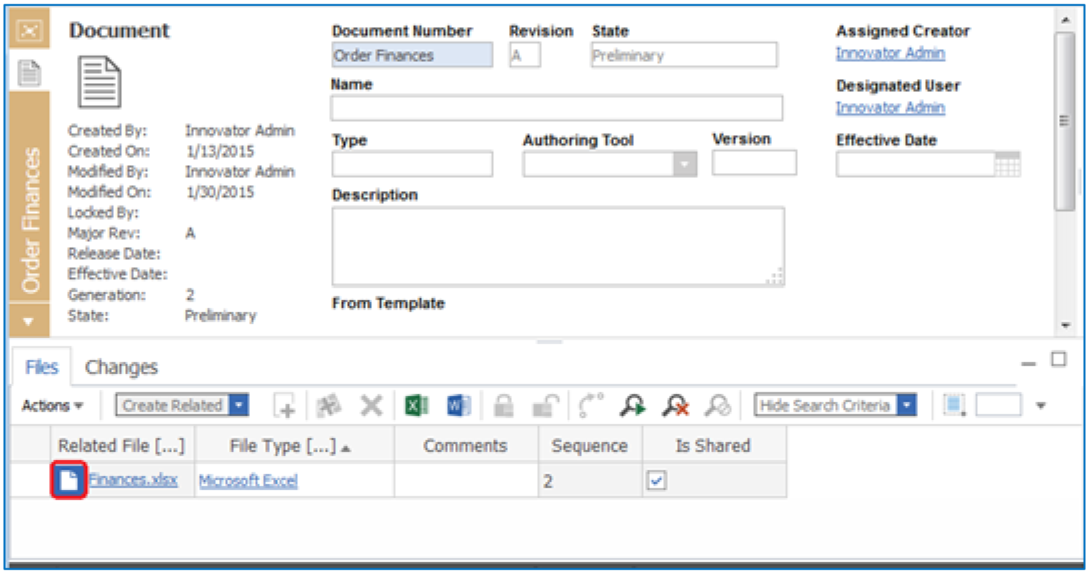
Clicking on will clear the file from the Property. The File is not deleted unless the Variable Item `Force.Delete.Orphaned.Files` is set to 1 and the File is not included in any other Items.

Clicking on opens a file browser to select a new file for the Property. Any existing file stored in the property is replaced.

When you have finished working with the File, click on **OK** to close the dialog.

Relationship Grid

Files displayed in the Relationships grid work in the same way as those displayed in the main grid. By default, the `related_id` property is directly displayed in the relationships grid. This property acts as any other Property with a data source of the File ItemType.



Created By: Innovator Admin
Created On: 1/13/2015
Modified By: Innovator Admin
Modified On: 1/30/2015
Locked By:
Major Rev: A
Release Date:
Effective Date:
Generation: 2
State: Preliminary

Document Number: Order Finances
Revision: A
State: Preliminary
Name:
Type:
Authoring Tool:
Version:
Description:
From Template
Assigned Creator: Innovator Admin
Designated User: Innovator Admin
Effective Date:

Related File [...]	File Type [...]	Comments	Sequence	Is Shared
Finances.xlsx	Microsoft Excel		2	☑

In the **Related File […]** column, clicking on the file link streams the physical file to the browser to be handled while clicking on the File icon displays the Manage File UI. The available buttons in the UI are dependent on the current lock status of the Item as well as on the RelationshipType definition.

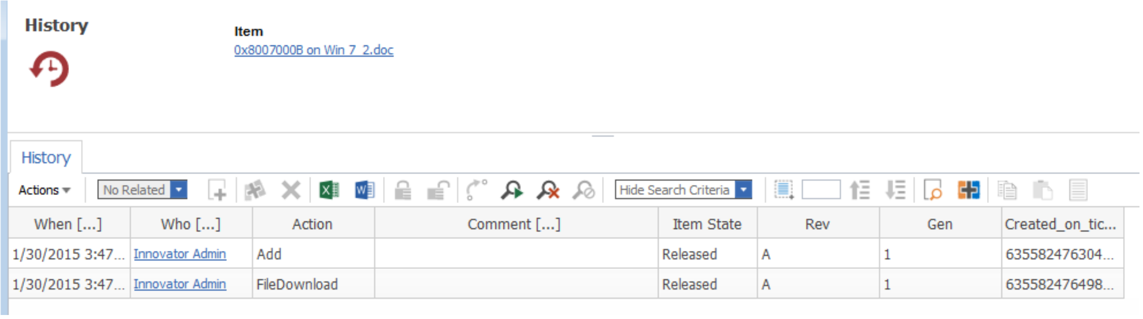
If the relationship does not allow the `related_id` to be null, then the **clear** icon is disabled.

If the relationship does not allow using **Pick Related**, then the **upload file** button is disabled.

Both buttons are disabled if the containing Item is locked.

File History

File Items are handled in different ways in Infor PLM Accelerate. Because of this, the History Actions associated with Files are also different:



Item: 0x8007000B on Win 7_2.doc

When [...]	Who [...]	Action	Comment [...]	Item State	Rev	Gen	Created_on_tic...
1/30/2015 3:47...	Innovator Admin	Add		Released	A	1	635582476304...
1/30/2015 3:47...	Innovator Admin	FileDownload		Released	A	1	635582476498...

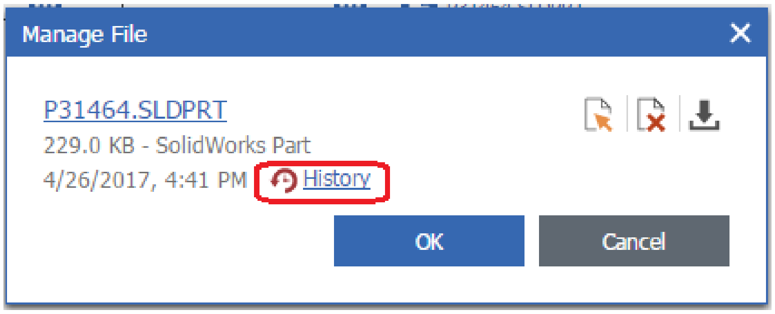
The **FileDownload** History Action records a History entry whenever the File is requested from the server to be downloaded to the client. The standard **Add** History Action records a History entry whenever a new File Item is added to Infor PLM Accelerate.

Additional information on File Activity can be followed using the **Track** History option for the Properties themselves.

Enabling File History

In a standard installation of Infor PLM Accelerate, History for File Items is disabled.

To enable history you need to login as Super User, login name **root**, and set the File History template on the File ItemType:



P31464.SLDPRT
229.0 KB - SolidWorks Part
4/26/2017, 4:41 PM

Once enabled, access the File History from the Manage File dialog:

Chapter 4. File API

Item.setFileProperty()

The Item function `setFileProperty()` takes in 2 parameters: property and file path.

- ***Property* –** The property of type file for which you want to set the file item for add
- ***File path* –** The file path to the file on the file system

Example:

```
Item cadItem = inn.newItem("CAD", "add");

cadItem.setFileProperty("native_file", filePath);

...
```

Item.fetchFileProperty()

The function `fetchFileProperty()` takes in 3 parameters; property, file path, and mode. It returns *Item* of type *File*.

- ***Property* –** The property of type file for which you want to download the file
- ***Target path* –** The file path on the file system where to place the file
  - A file name can be specified as to change the name of the file
    i.e. C:\InforTest\MyNewFileName.txt
  - ***Mode* –** The mode of operation for checkout
    - `FetchFileMode.Normal –` In this mode, method performs file downloading to a defined target path. The return item has the `checkedout_path` property set to the value of the path where the file is downloaded.
    - `FetchFileMode.Dry –` In this mode, the method performs a “dry” run without downloading the physical file, but checks if a file with same name exists in the specified target path. The return item has the `checkedout_path` property set to the value of the path to where the file would be downloaded. You can then use this path to determine if a file already exists in this path for further logic handling. No file is downloaded in this mode.

Example:

```
Item cadItem = inn.newItem("CAD", "add");

Item myFile = cadItem.fetchFileProperty("native_file", filePath,
FetchFileMode.Normal);

...
```

replaceFile method

A server method called replaceFile can be executed by File Administrators and replaces all references of the specified old file with the new file you specify.

This server method can be called with the following syntax:

```
<AML>

  <Item type="File" action="replaceFile" id="oldFileID" newId="newFileID"
/>

</AML>
```

JavaScript functions

To upload a File using method code on the client-side, the following examples may be referenced.

To download a File through javascript, use the downloadFile function. For example:

```
   aras.downloadFile(myFileItem.node);
```

Client-side Action to prompt for File:

```
      function processAddingFile(file) {

            var newFile = aras.newItem("File", file);

            if (newFile) {
                  aras.itemsCache.addItem(newFile);
                  var res = aras.saveItemEx(newFile);
                  aras.unlockItemEx(res);
            }
      }
      aras.vault.selectFile().then(processAddingFile.bind(this));
```

Form onClick event for Button:

```
      function processAddingFile(file) {
            var newFile = aras.newItem("File", file);

            if (newFile) {
                  aras.itemsCache.addItem(newFile);
                  window.handleItemChange("myFileProperty", newFile);
            }
      }
      aras.vault.selectFile().then(processAddingFile.bind(this));
```

Client-side setFileProperty Example:

```
      aras.vault.selectFile().then(function (fileObject)
{
    var d = aras.IomInnovator.newItem('CAD', 'add');
    d.setProperty('item_number', 'CAD_001');
    d.setFileProperty('native_file', fileObject);
    d.apply();
});
```

Chapter 5. File Representations

Starting in Infor PLM Accelerate 10.11.12, it is possible to represent files with the contents of another file, which is available to the end user. The most common use cases for this feature are:

- Providing a web-capable image for another, less available file type. For example, a CGM file cannot be displayed by browsers, but could be represented by an SVG file.
- Providing multiple resolutions of an image. For example, a high resolution JPEG could be used as the main image on an item. This image then could have multiple representations at different resolutions, which could be used as a thumbnail or a mobile view.

AML API

It is possible to use standard AML to operate on File Representations.

AML to Add a File Representation to a File.

```
   <AML>
      <Item type="fr_Representation" action="add">
            <source_id>{source file ID}</source_id>
            <representation_type_id>
                  <Item type="fr_RepresentationType" action="get">
                        <name>XML</name>
                  </Item>
            </representation_type_id>
            <Relationships>
                  <Item type="fr_RepresentationFile" action="add">
                        <kind>js</kind>
                        <related_id>{ID of representation JS
   file}</related_id>
                  </Item>
                  <Item type="fr_RepresentationFile" action="add">
                        <kind>html</kind>
                        <related_id>{ID of representation HTML
   file}<</related_id>
                  </Item>
            </Relationships>
      </Item>
</AML>
```

AML to Get a File Representation to a File.

```
<AML>
   <Item type="fr_Representation" action="get" select="id">
         <source_id>{source file ID}</source_id>
         <representation_type_id>
               <Item type="fr_RepresentationType" action="get">
                     <name>XML</name>
               </Item>
         </representation_type_id>
         <Relationships>
               <Item type="fr_RepresentationFile" select="related_id(id,
filename)">
               </Item>
         </Relationships>
   </Item>
</AML>
```

AML to Delete a File Representation to a File.

The clearest way to delete representations is to first get the representations, and then delete them in a separate action.

```
<AML>
   <Item type="fr_Representation" action="get" select="id">
         <source_id>0DDDD9189B4742EDB99452D45B3C9E3B</source_id>
         <representation_type_id>
               <Item type="fr_RepresentationType" action="get">
                     <name>XML</name>
               </Item>
         </representation_type_id>
   </Item>
</AML>
```

```
<AML>
   <Item type="fr_Representation" action="delete"
idlist="%idlist%"></Item>
</AML>
```

Representation Characteristics

File Representations make use of the new XProperty model to distinguish between different representations of the same RepresentationType. For example, a resolution characteristic may be defined to allow different values for different representations within a RepresentationType.

Note that there is a known issue that modifications to the RepresentationType can take a significant amount of server processing (see the Release Notes for details on IR-053121). This will cause the server to appear frozen to end users.

We recommend performing any modifications to characteristics as an offline action. Additionally, the database should be backed up prior to modifying the characteristics.

Chapter 6. File UI Security

The Vault Server web.config file contains a Content Security Policy (CSP) that prevents scripts contained in html and xml files from executing before displaying the files in a browser. The configuration adds the following parameters to the web.config file:

```
    </headers>
```

This security feature is designed to prevent both Injection and Cross-Site Scripting attacks.