Customizing the Outlook Security Features Administrative Package

The Microsoft® Outlook Security features guard against most viruses that are spread via attachments to e-mail messages, in addition to protecting users from worm viruses that replicate through Microsoft Outlook®. The security features are installed by default with Microsoft Outlook 2002, which means that a standard installation will contain the locked-down settings established by the Outlook security template.

While the security features provide a higher level of protection, they do limit certain functionality with Outlook. Restrictions include limits to specific types of attachments, heightened default security settings, and controlled access to the Outlook automation code.

As an administrator, you can customize the Outlook security settings to meet your organization's needs. For example, you can control the types of attached files blocked by Outlook, modify the Outlook object model security and warning levels, and specify user or group security levels.

At this time, to enable custom security settings, your clients must be using Outlook with Microsoft Exchange Server and have either the Mailbox (MDB) or Offline folders (OST) as the default e-mail delivery location. You cannot modify the settings if a client is using a local PST file for a mailbox, or if your organization is using Outlook with a third-party e-mail service.

Note Lowering any default security settings may increase your risk of virus execution or propagation. Use caution and read the documentation before you modify these settings.

How custom security settings work

When you create custom security settings for Outlook, the settings are stored in messages in a top-level folder in the Public Folders tree. Every user who needs customized security settings must have a special registry key set on his or her computer in order to gain access to the modified settings. When the key is present, Outlook will look on the server for any custom security settings that apply to that user. If customized security settings are found, they will be used. Otherwise, the default security settings will be applied to the computer. Users without the special key will have the default Outlook security settings. In some cases, administrator-defined security settings may interact with security settings defined by the user. For more information, see the section “Administrator-controlled settings versus user-controlled settings,” later in this topic.

Tools for the Outlook Security features administrative package

The administrative tools for the Outlook Security features consist of four files, packaged into one self-extracting executable. This executable, Admpack.exe, can be installed separately from the Office XP Resource Kit CD or Enterprise editions of Microsoft Office. It is not installed by default from the Office Resource Kit Setup program. The four administrative files are as follows:

- OutlookSecurity.oft is an Outlook template that enables you to customize the security settings on the Microsoft Exchange server.

  The template does not actually implement security — it is simply the storage location for the customized security settings.
- Hashctl.dll is the file for the Trusted Code control, a tool used by the template to specify trusted COM add-ins.
- Comdlg32.ocx is a file used for the Trusted Code control. It provides a user interface for selecting the trusted COM add-in.
- Readme.doc is a document that provides information on the values and settings available in the template and describes how to deploy the new settings on Exchange Server.

To install the Outlook Security features administrative package, run Admpack.exe from the \Files\PFiles\ORKTools\ORK10\Tools\Admpack\ folder on the Office Resource Kit CD. If you are installing the Outlook Security features administrative package from an Office Enterprise edition CD, the path is \ORK\Files\PFiles\ORKTools\ORK10\Tools\Admpack\. When you run this executable, it copies the four administrative files to a working directory you specify on your computer.

Note The administrator must use a computer with the Windows 2000 operating system in order to run the administrative tools for Outlook Security features.

Installing the Trusted Code control

In Outlook 2002, administrators are now able to specify COM add-ins that are trusted by the security features and can be run without encountering the Outlook Object Model security blocks. In order to specify a COM add-in, administrators must first install a control on the computer they are using to modify the security settings. The control does not need to be installed on end-user computers, just the administrator’s machine.

Installing the Trusted Code control is a good first step in the customization process, as it will enable you to see all options available on the Outlook Security template. After you install the control, you must register it on the administrative computer. If you do not register the control, you will get an error when you try to view the **Trusted Code** tab on the template.

To install and register the Trusted Code control

1. Copy the file Hashctl.dll from your working directory to the \winnt\system32 folder on your administrative computer.

   If your operating system is installed in a directory other than \Winnt, substitute the appropriate path name.

2. From the **Start** menu, choose **Run**, then type the following command line in the box to register the control:

   regsvr32 hashctl.dll

3. Copy the file Comdlg32.ocx from your working directory to the \winnt\system32 folder on your administrative computer.

   If your operating system is installed in a directory other than \Winnt, substitute the appropriate path name.

4. From the **Start** menu, choose **Run**, then type the following command line in the box to register the control:

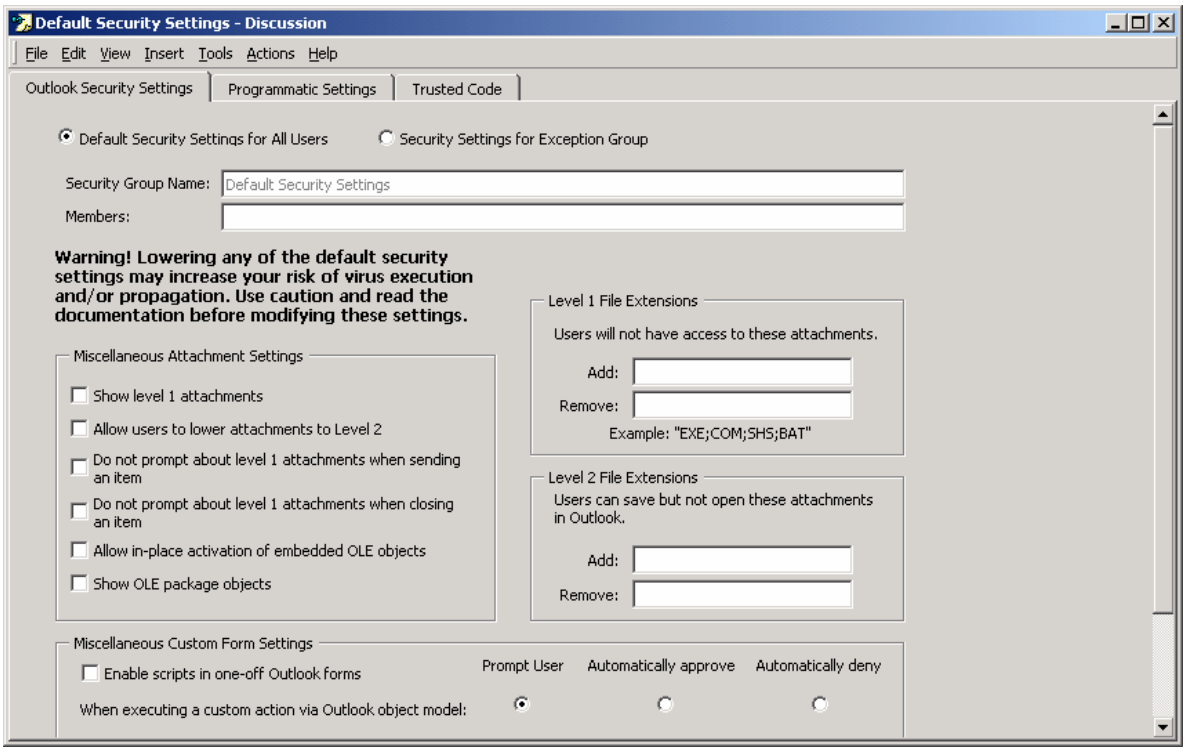
   regsvr32 comdlg32.ocx

Customizing the Outlook security settings

You can modify the default settings on the Outlook Security template to configure the level of security enforced by Exchange Server. The Outlook Security template is an Outlook item template that you run through Microsoft Outlook. The template contains three tabs:

- Outlook Security Settings
- Programmatic Settings
- Trusted Code

When you first load the template, the settings show the default values for the Outlook Security features.



Creating a public folder for the security settings

Before you begin to modify the security settings, you must create a public folder named "Outlook Security Settings" or “Outlook 10 Security Settings” on Exchange Server. The administrator must create this folder, using that exact name, in the root folder of the Public Folder tree. You must set the folder Access Control Lists (ACLs) so all users can read all items in the folder. However, only those users who you want to create or change security settings should have permission to create, edit, or delete items in the folder.

If you want multiple users to be able to edit or create items, and if the list of users can change at anytime, then you must create a security group that includes all users who you want to be able to create or change security settings. This security group should have owner permissions on the security folder. After you create the folder, you can use the template to make the changes you need.

Note If you are publishing a new security form over a previous version, you should overwrite the older form with the new copy, using the same name and message class. This will install the new form in place of the old one in the security folder. When you replace the security form, you will also need to open any other forms in the security folder and close them by using the Close button to correctly register the change.

Modifying the default security settings

Use the following procedure to modify the default security settings established by the security features.

To use the Outlook Security template to modify settings on Exchange Server

1. On a computer running Outlook, open OutlookSecurity.oft from the working directory where you installed the Outlook security tools.
2. When asked to select a folder, select the Outlook Security Settings or Outlook 10 Security Settings public folder that you created on Exchange Server. The template will then open in Compose mode.
3. On the **Tools** menu of the template, point to **Forms**, and then click **Publish Form**. (The folder selected should be your current folder, Outlook Security Settings or Outlook 10 Security Settings.)
4. In the **Form Name** box, type **Outlook Security Form**. If you are currently using the security form from the e-mail security patch, and if you are publishing the form to the Outlook Security Settings folder, then in the **Form Name** box, type the same name as the previous security form (i.e., overwrite previous security form).
5. Click the **Publish** button to publish the security template in the Security Settings folder.

   You can now close the Outlook Security template. Do **not** save when prompted to save while closing the template.
6. Switch to Microsoft Outlook, click the drop-down arrow next to the **New** button on the toolbar, and select the **Choose Form** command from the list.
7. Navigate to the template you just created in the previous steps then select the new template name and click the **Open** button.
8. Create either a default security setting or custom settings for a specific set of users.

   To create a default security setting that will be used by all users, click **Default Security Settings for All Users**, and then scroll to the bottom of the template and click the **Close** button.

   To create custom security settings for a specific set of users, click **Security Settings for Exception Group**, and then type a name in the **Security Group Name** box that describes the group. In the **Members** box, type the name of each user who must have custom security settings. If the Exchange server you are running against is an Exchange 2000 or later server, then you can use distribution lists (only server created security groups) in the **Members** box. Otherwise, you cannot use distribution lists. Adding users from the Contact Address Book is not supported. Specify the settings you need and then click the **Close** button.

Note Administrators to the public folder must have their names added explicitly to one of the member lists of a security setting if you have created any security settings. A distribution list of Administrators will not be sufficient—you must explicitly add each administrator's name to a security setting. If you are only using a single default security group, then you do not need to add administrator's names.

If a user's name is entered as a member of more than one security group, the settings of the most recently created group will apply, because Outlook looks for the first item that has the user's name in the **To** field.

Details on all fields, values, and settings for the template can be found in the section "Outlook Template Security Settings" later in this topic.

Creating security settings

Every time a Default Security Settings for All Users or Security Settings for Exception Group is created, the administrator will be prompted twice for credentials. If no credentials are entered or if the wrong credentials are entered, an "Operation Failed" error message will appear. At this point, the security settings item has been created but will not work correctly. The administrator must delete the security item that was created and recreate it. If the item created is not deleted, then these settings will be applied to everybody, and users not intended to get this security/permission setting will receive it.

Editing security settings

If the administrator adds a user to the Members field of an existing security form, the administrator should make sure that all aliases already present in the form are current and active.

Note If you add the alias of a new member to an existing security form, the change may not be correctly registered unless you make other changes to the form as well. For example, you might toggle another setting on and off, or otherwise activate the form through some interaction. After you have added the new alias and activated the form, you can choose Save from the File menu of the form to save your changes.

Deploying the customized Outlook security settings to client computers

After you configure the security features on Exchange Server, you must enable the customized settings for your users. To enable the changed settings, you may need to deploy a new registry key to the client computers, depending upon whether or not Microsoft Office was initially deployed with system policies.

- If Office was deployed with system policies, the Outlk10.adm file will automatically pass your customized security settings to client computers each time users log on to the system.
- If Office was deployed without system policies, you must create a new registry key on the client computers. Outlook will respect this new registry key, even if you are not using policies.

The registry key is of type **DWORD** and the value is as follows:

HKCU\SW\Policies\Microsoft\Security\CheckAdminSettings

The following table describes the key states.

Key state	Description
No key	Outlook uses default administrative settings.
Set to **0**	Outlook uses default administrative settings.
Set to **1**	Outlook looks for custom administrative settings in the Outlook Security Settings folder.
Set to **2**	Outlook looks for custom administrative settings in the Outlook 10 Security Settings folder.
Set to anything else	Outlook uses default administrative settings

To create a new registry key for distribution to client computers

1. Start the registry editor and expand the following subkey:

   HKEY_CURRENT_USER\Software\Policies\Microsoft\Security

2. From the **Edit** menu, choose **New**, then click **DWORD value** to add a new registry key.

   The value name for the key must be **CheckAdminSettings**.

3. Select the new key name, and then from the **Registry** menu, choose **Export Registry File**.

4. In the **Export Registry File** dialog box, type a name for the registry file and select the option for **Selected Branch** under the **Export Range** group, then click **Save** to create the registry file.

   Registry files have a REG extension.

After you create the new key, you distribute it to the client computers. You can distribute the key by adding it to a logon script, copying it to a shared server for users to run, or attaching it as a shortcut to an e-mail message. You cannot attach the file itself to a message, since REG files are restricted by the Outlook Security features.

Registry key files are a registered file type, which means that the key will be automatically installed on a client computer when a user double-clicks the file name.

Outlook template security settings

The following sections describe the configurations you can specify on the Outlook Security template.

Outlook Security Settings tab

The **Outlook Security Settings** tab enables you to configure settings related to attachments, the types of files to which users can gain access, and scripting.

General settings

The following table describes the settings that specify security groups and members.

Item	Description
Default Security Settings for All Users	Applies the default e-mail security settings to everyone.
Security Settings for Exception Group	Enables you to create custom e-mail security settings for some users.
Security Group Name	Specifies a name for the security group to which these customizations will apply. For example: "Object model access approved."
Members	Lists the names of members in this security group. If you are using Exchange 2000 or later server then you can use Distribution Lists (i.e., server-based security groups). You must type names individually, separating each name by a semicolon. If a user's name is entered as a member of more than one security group, the settings of the most recently created group will apply, because Outlook looks for the first item that has the user's name in the **To** field. Administrator should not use the address book to enter an alias into the Members field when creating a security form. The only way to enter an alias into the Members field is by directly entering it into the field.

Miscellaneous attachment settings

The following table describes the security options for e-mail attachments.

Item	Description
Show Level 1 attachments	Enables users to gain access to attachments with Level 1 file types.
Allow users to lower attachments to Level 2	Enables the end user to demote a Level 1 attachment to Level 2.
Do not prompt about Level 1 attachments	Prevents users from receiving a warning when they send an item containing a

when sending an item	Level 1 attachment. This option affects only the warning. Once the item is sent, the user will not be able to see or gain access to the attachment. If you want users to be able to post items to a public folder without receiving this prompt, you must select both this check box and the **Do not prompt about Level 1 attachments when closing an item** check box.
Do not prompt about Level 1 attachments when closing an item	Prevents users from receiving a warning when they close a mail message, appointment, or other item containing a Level 1 attachment. This option affects only the warning. Once the item is closed, the user will not be able to see or gain access to the attachment. If you want users to be able to post items to a public folder without receiving this prompt, you must select both this check box and the **Do not prompt about Level 1 attachments when sending an item** check box.
Allow in-place activation of embedded OLE objects	Allows users to double-click an embedded object, such as a Microsoft Excel spreadsheet, and open it in the program. If you are using Microsoft Word as your e-mail editor, clearing this check box will still allow OLE objects to be opened when the embedded object is double-clicked.
Show OLE package objects	Displays OLE objects that have been packaged. A package is an icon that represents an embedded or linked OLE object. When you double-click the package, the program used to create the object either plays the object (for example, if it's a sound file) or opens and displays the object. Caution should be used in displaying OLE package objects, because the icon can easily be changed and used to disguise malicious files.

Level 1 file extensions

Level 1 files are hidden from the user in all items. The user cannot open, save, or print a Level 1 attachment. The InfoBar at the top of the item will display a list of the blocked files. The Infobar does not appear on a custom form. For information on a default list of

Level 1 file types, see the section "Level 1 file types blocked by Outlook," later in this topic.

The following table describes how to add or remove Level 1 file extensions from the default list.

Item	Description
Add	Specifies the file extension (usually three letters) of the file types you want to add to the Level 1 file list. Do not enter a period before the file extension. If you enter multiple extensions, separate them with semicolons.
Remove	Specifies the file extension (usually three letters) of file types you want to remove from the Level 1 file list. Do not enter a period before the file extension. If you enter multiple extensions, separate them with semicolons.

Level 2 file extensions

With a Level 2 file, the user is required to save the file to disk before opening it. A Level 2 file cannot be opened directly from an item. The following table describes how to add or remove Level 2 file extensions from the default list.

Item	Description
Add	Specifies the file extension (usually three letters) of the file types you want to add to the Level 2 file list. Do not enter a period before the file extension. If you enter multiple extensions, separate them with semicolons.
Remove	Specifies the file extension (usually three letters) of file types you want to remove from the Level 2 file list. Do not enter a period before the file extension. If you enter multiple extensions, separate them with semicolons.

Miscellaneous Custom Form Settings

The following table describes the security settings for scripts, custom controls, and custom actions. (Scroll down in the Outlook Security template to see the full set of options.)

Item	Description

Enable scripts in one-off Outlook forms	Select this check box to run scripts in forms where the script and the layout are contained in the message itself.
When executing a custom action via the Outlook object model	Specifies what happens when a program attempts to run a custom action using the Outlook object model. A custom action can be created to reply to a message and circumvent the programmatic send protections described above. Select one of the following: **Prompt user** enables the user to receive a message and decide whether to allow programmatic send access. **Automatically approve** always allows programmatic send access without displaying a message. **Automatically deny** always denies programmatic send access without displaying a message.
When accessing the ItemProperty property of a control on an Outlook custom form	Specifies what happens when a user adds a control to a custom Outlook form and then binds that control directly to any of the Address Information fields. By doing this, code can be used to indirectly retrieve the value of the Address Information field by getting the Value property of the control. Select one of the following: **Prompt user** enables the user to receive a message and decide whether to allow access to Address Information fields. **Automatically approve** always allows access to Address Information fields without displaying a message. **Automatically deny** always denies access to Address Information fields without displaying a message.

Programmatic Settings tab

The **Programmatic Settings** tab enables you to configure settings related to your use of the Outlook object model, Collaboration Data Objects (CDO), and the Simple Messaging Application Programming Interface (Simple MAPI). These technologies are defined as follows:

- **Outlook object model** — The Outlook object model allows you to programmatically manipulate data stored in Outlook folders.
- **CDO** — Collaboration Data Object libraries are used to implement messaging and collaboration functionality in a custom application. CDO is a COM wrapper of the MAPI library and can be called from any development language that supports automation. CDO implements most but not all MAPI functionality (but more than Simple MAPI).
- **Simple MAPI** — Simple MAPI enables developers to add basic messaging functionality, such as sending and receiving messages, to their Windows®-based applications. It is a subset of MAPI, which provides complete access to messaging and information exchange systems.

The following table lists descriptions for each option on the **Programmatic Settings** tab. For each item, you can choose one of the following settings:

- **Prompt user** — Users receive a message allowing them to choose whether to allow or deny the operation. For some prompts, users can choose to allow or deny the operation without prompts for up to 10 minutes.
- **Automatically approve** — The operation will be allowed and the user will not receive a prompt.
- **Automatically deny** — The operation will not be allowed and the user will not receive a prompt.

The following table describes the available options. You will need to scroll down in the template to see the full set of options.

Item	Description
When sending items via Outlook object model	Specifies what happens when a program attempts to send mail programmatically using the Outlook object model.
When sending items via CDO	Specifies what happens when a program attempts to send mail programmatically using CDO.
When sending items via Simple MAPI	Specifies what happens when a program attempts to send mail programmatically using Simple MAPI.
When accessing the address book via Outlook object model	Specifies what happens when a program attempts to gain access to an address book using the Outlook object model.
When accessing the address book via CDO	Specifies what happens when a program attempts to gain access to an address book using CDO.
When resolving names via Simple MAPI	Specifies what happens when a program attempts to gain access to an address book using Simple MAPI.
When accessing address information via	Specifies what happens when a program

Outlook object model	attempts to gain access to a recipient field, such as **To**, using the Outlook object model.
When accessing address information via CDO	Specifies what happens when a program attempts to gain access to a recipient field, such as **To**, using CDO.
When opening messages via Simple MAPI	Specifies what happens when a program attempts to gain access to a recipient field, such as **To**, using Simple MAPI.
When responding to meeting and task requests via Outlook object model	Specifies what happens when a program attempts to send mail programmatically using the Respond method on task requests and meeting requests. This method is similar to the Send method on mail messages.
When executing Save As via the Outlook object model	Specifies what happens when a program attempts to programmatically use the **Save As** command on the **File** menu to save an item. Once an item has been saved, a malicious program could search the file for e-mail addresses.
When accessing the Formula property of a UserProperty object in the Outlook object model	Specifies what happens when a user adds a Combination or Formula custom field to a custom form and binds it to an Address Information field. By doing this, code can be used to indirectly retrieve the value of the Address Information field by getting the Value property of the field.
When accessing address information via UserProperties.Find in the Outlook object model	Specifies what happens when a program attempts to search mail folders for address information using the Outlook object model.

Trusted Code tab

The **Trusted Code** tab is used to specify which COM add-ins are trusted and can be run without encountering the Outlook object model blocks. The following procedure describes how to use this feature.

To specify a trusted add-in

1. Copy the DLL (or other file) that is used to load the COM add-in to a location where the administrator creating the security setting has access to it.

   This file must be the same file used on the client computers that will run the COM add-in.

2. On the **Trusted Code** tab, click the **Add** button and select the name of the DLL you want to add.
3. Click the **Close** button on the form when you have finished.

The COM add-in will now run without prompts for Outlook 2002 users who use this security setting. To remove a file from the Trusted list on the **Trusted Code** tab, select the file name and click the **Remove** button.

Administrator-controlled settings versus user-controlled settings

In general, security settings defined by the user work as if they were added to the settings defined by the administrator. When there is a conflict between the two, the settings with a higher security level will override settings with a lower level of security. The following list describes some specific interactions between administrator and user security settings.

Show Level 1 attachments. When this option is set on the Outlook Security Settings tab, all file types are set to Level 2 security. The user will then need to customize the list to block access to specific types of attachments.

Level 1 file extensions – Add. When set by the administrator, this list overrides the user's settings. For example, if the user wants to remove EXE, REG, and COM, but the administrator explicitly adds EXE into the Level 1 Add category, then the user would only have access to REG and COM files.

Level 1 file extensions – Remove. The user's list is combined with the list set by the administrator to determine which Level 1 items are set to Level 2.

Level 2 file extensions – Add. If a user turns Level 1 files into Level 2 files, and those file types are listed in the Add box, the files are treated as Level 2 attachments.

Level 2 file extensions – Remove. There is no interaction with this setting.

Allow users to lower attachments to Level 2. This setting allows a user to demote a Level 1 attachment to Level 2. If this option is unchecked, the user's list is ignored and the administrative settings control the security.

How security settings are controlled within the registry

The option to prevent end-user customization of the security settings is controlled by a registry key. The value of the key is as follows:

```
HKCU\Software\Policies\Microsoft\Office\10.0\Outlook
Value: DisallowAttachmentCustomization
```

If the registry key is present, end-user customization is disallowed. If key not installed on the computer, end-user customization is allowed. The value of the key has no effect.

The registry key to set the exception list contains a semi-colon delimited list of file extensions. The value of the key is as follows:

```
HKCU\Software\Microsoft\Office\10.0\Outlook\Security
Key: Level1Remove
```

If the value of the key is formatted incorrectly, the restrictions are ignored.

Additional information

The following topics address questions you might encounter when customizing Outlook security settings.

PIM only mode

In PIM (Personal Information Manager) mode, Outlook uses the default security settings. No administrator settings are looked for or used in this mode.

Need to restart Outlook

The first time a user starts Outlook after the security settings have been applied, the user will see default administrative settings and not the exception or default form that has been set. The user needs to close Outlook and then restart Outlook again to get the correct security settings and permissions.

Level 1 file types blocked by Outlook

The following table lists the Level 1 file types that are blocked under a default installation of Outlook 2002. You can add or remove items from the default list through the **Outlook Security Settings** tab of the Outlook Security template.

File extension	File type
.ade	Microsoft Access project extension
.adp	Microsoft Access project
.asx	Windows Media Audio / Video shortcut
.bas	Microsoft Visual Basic® class module
.bat	Batch file
.chm	Compiled HTML Help file
.cmd	Microsoft Windows NT® command script
.com	Microsoft MS-DOS program
.cpl	Control Panel extension
.crt	Security certificate
.exe	Executable program
.hlp	Help file
.hta	HTML program
.inf	Setup information

.ins	Internet naming service
.isp	Internet communication settings
.js	JScript file
.jse	Jscript-encoded script file
.lnk	Shortcut
.mda	Microsoft Access add-in program
.mdb	Microsoft Access program
.mde	Microsoft Access MDE database
.mdz	Microsoft Access wizard program
.msc	Microsoft Common Console document
.msi	Windows Installer package
.msp	Windows Installer patch
.mst	Visual Test source files
.pcd	Photo CD image or Microsoft Visual Test compiled script
.pif	Shortcut to MS-DOS program
.prf	Microsoft Outlook Profile Settings
.reg	Registration entries
.scf	Windows Explorer Command
.scr	Screen saver
.sct	Windows script component
.shb	Shortcut into a document
.shs	Shell scrap object
.url	Internet shortcut
.vb	VBScript file
.vbe	VBScript-encoded script file
.vbs	VBScript file
.wsc	Windows script component
.wsf	Windows script file
.wsh	Windows script host settings file

Level 2 file types restricted by Outlook

Level 2 file types are restricted under a default installation of Outlook 2002. You must first save a Level 2 file to disk before you can open it – Level 2 file types cannot be

opened directly from an item. You can add or remove attachment types from the default list through the Outlook Security Settings tab of the Outlook Security template. Also, you can demote attachments from Level 1 to Level 2 via the **Level1Remove** registry key. By default, there are no Level 2 file types.