Edit Permissions tab

Specifies how you want Internet Explorer to handle all content and permissions requested by signed and unsigned Java applets.

The following permissions are affected by the settings for Unsigned and Signed permissions:

Access to all files

Access to all Network Addresses

Execute

Dialogs

System Information

Printing

Protected Scratch Space

User Directed File I/O

Unsigned Permissions

You can specify permissions individually by setting **Run Unsigned Content** to **Run in sandbox** Then you can reset each permission individually to **Disable** or **Enable**. If you specify **Disable** or **Enable** under **Run Unsigned Content**, all permissions under **Additional Unsigned Permissions** will use that setting.

Select one of the following:

- To run unsigned content with only the permissions that are allowed in the sandbox, click **Run in sandbox**. If you choose **Run in sandbox** for **Run Unsigned Content**, you can reset each permission individually to **Disable** or **Enable**.
- To automatically refuse unsigned content without being prompted, click **Disable**. All permissions under **Additional Unsigned Permissions** are set to **Disable**, and you cannot reset any permission individually to **Enable**.
- To automatically accept unsigned content without being prompted, click **Enable**. All permissions under **Additional Unsigned Permissions** are set to **Enable**, and you cannot reset any permission individually to **Disable**.

Signed Permissions

You can specify permissions individually by setting **Run Signed Content** to **Prompt**, which sets all permissions under **Additional Signed Permissions** to **Prompt**. Then you can reset each permission individually to **Disable** or **Enable**. If you specify **Disable** or **Enable** under **Run Signed Content**, all permissions under **Additional Signed Permissions** will use that setting.

Select one of the following:

- To be prompted for approval before proceeding, click **Prompt**. If you choose **Prompt** for **Run Signed Content**, all permissions under **Additional Signed Permissions** are set to **Prompt**, but you can reset each permission individually to **Disable** or **Enable**.
- To automatically refuse signed content without being prompted, click **Disable**. All permissions under **Additional Signed Permissions** are set to **Disable**, and you cannot reset any permission individually to **Prompt** or **Enable**.
- To automatically accept signed content without being prompted, click **Enable**. All permissions under **Additional Signed Permissions** are set to **Enable**, and you cannot reset any permission individually to **Prompt** or **Disable**.

Closes this dialog box and saves any changes you have made.

Click this to reset all Java permissions. Select one of the following, and then click **Reset**.

- **Saved Permissions** Resets to the last known saved permissions. Any changes made since the last saved settings will be lost.
- **High Security** Resets to High Security permissions (most restrictive; applets run in safe mode). This resets all permissions under **Run Signed Content** to **Prompt** and **Additional Unsigned Permissions** to **Disable**.
- **Medium Security** Reset to Medium Security permissions (applets run with some restrictions). This resets all permissions (except Scratch Space and User Directed File I/O) under **Run Signed Content** to **Prompt** and **Additional Unsigned Permissions** to **Disable**.
- **Low Security** Reset to Low Security permissions (least restrictive; applets run with all permissions). This resets all permissions under **Run Signed Content** to **Enable** and **Additional Unsigned Permissions** to **Disable**.

View Permissions tab

These permissions are what Java permissions have been specified by your network administrator.

In order for a Java applet to run, it may require file access and other resources on your computer. Each of these types of actions requires a specific permission to be granted before the action can be carried out. Your network administrator may have already specified which permissions to allow. For those which are allowed, your network administrator can specify whether to notify you when those permissions are requested. Otherwise, you will only be notified when a Java applet requests more permissions than are automatically allowed by your network administrator.

There are three sets of permission:

Permissions Enabled for Unsigned Content Permissions granted to unsigned downloaded content

Permissions Enabled for Signed Content Permissions that do not require user approval

Permissions Disabled for Signed Content Permissions that require user approval or are absolutely denied

You can double-click each of these permission headings to view the specific permissions and settings specified.

The following permissions may be assigned to these sets:

System Information

Reflection

User Interface Access

File I/O

Net I/O

Threads

User Directed File I/O

Client Storage

Property

Execution

Printing

Registry

Security

Multimedia

Custom

A permission that controls read, write, and delete access to files.

A permission that controls the ability to perform network operations or a network-related action.

A permission that controls the ability to create and manipulate threads and thread groups.

A permission that controls the ability to access or manipulate global system properties.

A permission that controls the ability to run other programs.

A permission that controls the ability to use the Reflection API to gain access to members of a specified class.

A permission that controls access to the printing APIs.

A permission that controls the ability to gain access to the registry.

A permission that controls access to the JDK security classes **java.lang.security**.

A permission for controlling access to client-side storage that is available through the **ClientStore** class.

A permission that controls the ability to use some of the enhanced functionality of the AWT.

A permission that controls access to system information.

A permission that controls the ability to display file dialog boxes to perform file operations. For example, if the applet wanted to open a file, it must present the standard File Open dialog box and let the user select the file to be opened. The applet will not be able to perform file operations on its own. As such, this operation is considered safer than code having direct file access since there is direct user involvement. This permission is a "Medium" level permission.

A permission that controls the use of enhanced multimedia functionality.

A permission that controls the ability to perform custom operations.

A permission that controls the ability to create a scratch space area that can be used to store temporary information. The applet will not be allowed to read or write any other files on the users hard drive. A signed applet can only access its own scratch area. This permission is a "Medium" level permission.

A permission that controls the ability to present dialog boxes.