Reflection Desktop

Deployment Guide

Version 16.2

Copyrights and Notices

Reflection® Desktop

Copyright

Patents

This Attachmate software is protected by U.S. patents 6252607 and 6803914. Additional Patent Pending.

Trademarks

Attachmate, the Attachmate logo, and Reflection are registered trademarks of Attachmate Corporation in the USA. All other trademarks, trade names, or company names referenced in this product are used for identification only and are the property of their respective owners.

Third-Party Notices

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org (http://www.openssl.org)/).

Additional third-party notices, including copyrights and software license texts, can be found in a 'thirdpartynotices' file in the root directory of the software.

docsys (en) 5 January 2020

Contents

Introduction 7

1 Design and Prepare for Deployment 9

Best Practices for Planning a Reflection Deployment 9
1. Identify Technical and User Requirements, Subject Matter Resources, and Risk Mitigation 9
2. Inventory and Analyze User Requirements, Macros, Configuration Files, and Legacy Applications 10
3. Assess Which Existing Files To Carry Forward and Test 11
4. Package and Test 11
Installation 13
System Requirements 14
Upgrading from Previous Versions 15
Install Reflection on a Workstation 16
Features Selection Tab 17
Set Your User Data Directory 17
Advanced Tab 17
Setting up the Reflection Administrative Tools 18
About the Tools 18
Set up an Environment for Deployment Using Reflection Administrative Tools 20
Determining Customization Requirements 23
Create and Customize Reflection Sessions 23
Customize the Reflection Workspace 24
Customize to Protect Data and Information Privacy 25
Customize the Installation 26

2 Create and Customize Sessions 29

Create and Customize Session Documents 29
Walkthrough: Set up and Customize a Session 31
Create SSL/TLS or SSH Session Documents 32
Digital Certificates and Reflection Certificate Manager 32
Set up SSL/TLS Connections 34
Set up Secure Shell Connections 34
Set up Session Templates 35
Configure Reflection for PKI Auto Sign-on 36
Centrally Manage VBA Macros 37
Walkthrough: Set up a Session to Reference Macros in a Session Document File 39
Set up and Test References to Macros 39
Deploy Sessions that Reference Macros and Related Settings 40
Walkthrough: Set up and deploy a SharedMacros file 42
Set up and Test a SharedMacros file 43
Deploy the SharedMacros.rd3x file 43

3 Protect Data and Information Privacy 47

Add Trusted Locations 47

Configure Information Privacy 48
Configure API and Macro Security 49

4 Customize the Reflection Workspace 51

Configure Workspace Behavior and Appearance 51
Customize Workspace Settings Directly With Reflection 53
Configure and Automatically Package Workspace Settings 54
Update Only Specific Workspace Settings. 57
Walkthrough: Customize Reflection Appearance and Behavior 57
Control Access to Settings and Controls 60
Control Access to Settings and Controls with Reflection Administrative Tools 62
Specify Access Using Permissions Manager with the Installation Customization Tool 63
Specify Access Using Permissions Manager 66
Walkthrough: Restrict Access to Settings and Controls 67
Control Access to Settings and Controls with Microsoft Group Policy 68
Install Group Policy Templates 70
Set Access with Group Policy 71

5 Package Configuration Files 73

Package Sessions and Custom Settings Files 73
Walkthrough: Create a Package with the Installation Customization Tool 75
Customized Files and Where to Deploy Them 76
Workspace Settings 77
Shared Security Settings 78
Reflection FTP Client Settings 79
Regular Session Document Files and Session Settings 79
Custom Office Productivity Templates 80
VBA SharedMacros Files and the VBA Common Project 80

6 Modify the Installation 81

Create or Modify a Transform 81
Change the Installation Directory 82
Modify Setup Properties 82
Add/Modify Registry Data. 83
Select Features, Components, and Languages 84
Add (Chain) Installations and Run Programs 85
Install the Reflection Help 86
Predefined System Folders 87
Configure Shortcuts 88
Walkthrough: Create a Transform 89
Apply a Transform to Your Installation 90

7 Deploy Reflection 93

Deploy with the Reflection Setup program. 93
Deploy with MSI 94
Deploy Reflection from MSI Command Line 94
Deploy Companion MSI File from MSI Command Line. 96
Publish with Active Directory 96
Deploy with System Center Configuration Manager 96

Distribute Software Updates 97
Remove an Installation 97
Repair an Installation 99

8 Using a Centralized Management Server 101

Requirements and Best Practices for Using Centralized Management 102
Requirements for Centralized Management 102
Best Practices for using Centralized Management 102
Create and Deploy Sessions and Settings with the Administrative Console 103
Create or Modify a Centrally Managed Session 103
Make Centrally Managed Sessions Available to Users 103
Use Central Management to Deploy MSI Packages 103
Enable Certificate Management for IBM Terminals 104
Enable Usage Metering 104
How can you use metering? 104
Setting up Metering 105
Connect to Hosts using the Security Proxy Add-On 107
Requirements and Other Information for Using the Security Proxy 109
Connect using End-to-End Encryption in 3270 SSL/TLS Sessions 111
Connect using End-to-End Encryption in VT SSH Sessions 111
Set Up Terminal ID Management for Reflection Desktop Sessions 113
Set up Automated Sign-On for Mainframe Sessions 116
Implementing Automated Sign-On for Mainframe 117

Glossary 119

Introduction

This guide shows how to prepare for your Reflection Desktop deployment.

To get started with deployment and find answers to frequently asked questions, see Getting Started with Reflection Desktop Deployment.

This guide includes information for each part of your deployment process.

These articles	Show how to
Design and Prepare for Deployment	Follow best practices to plan your deployment. Determine whether you need to customize Reflection or modify how it is installed and make sure you meet systems requirements. Review the administrative tools and set up a system for customization, testing, and deployment.
"Create and Customize Sessions" on page 29	Create and customize session document files that you can deploy to users.
"Protect Data and Information Privacy" on page 47	Set up Reflection to add trusted locations, redact sensitive data (such as credit cared numbers), and control access to the Reflection API.
Customize Reflection	Customize Reflection to change the appearance and basic functionality of the main Reflection window and to control access to controls and settings.
"Package Configuration Files" on page 73	Create an MSI package to install custom files you created when you customized the Reflection sessions or the Reflection workspace.
Modify the Installation	Create a transform (MST file) to customize how Reflection is installed on user workstations.
Deploy Reflection	Deploy Reflection, session document files, and other configuration files. Instructions are included for using the Reflection setup.exe program and deploying directly with MSI as well as deploying with Microsoft Active Directory and Microsoft System Center Configuration Manager.
Use	Set up and centrally manage sessions on the Management and Security Server.

1 Design and Prepare for Deployment

You can choose from several different approaches for installing and deploying Reflection, ranging from “out-of-the-box” installations to heavily customized deployments.

Planning to make sure you meet user needs and technical requirements is a key to a successful deployment.

This article	Describes
“Best Practices for Planning a Reflection Deployment” on page 9	Best practices for designing a Reflection deployment to avoid common problems and make sure your deployment meets customer and technical requirements without disrupting users.
“Installation” on page 13	Hardware and software requirements, information about upgrading, and the Setup tool tabs.
“Setting up the Reflection Administrative Tools” on page 18	The Reflection administrative tools and how to set up an environment where you can use the tools to customize Reflection, package settings configuration files, and customize how Reflection is installed.
“Determining Customization Requirements” on page 23	How you can customize Reflection and customize how it is installed.

Best Practices for Planning a Reflection Deployment

Follow these best practices to avoid common problems and make sure your deployment meets customer and technical requirements without disrupting users.

- “1. Identify Technical and User Requirements, Subject Matter Resources, and Risk Mitigation” on page 9
- “2. Inventory and Analyze User Requirements, Macros, Configuration Files, and Legacy Applications” on page 10
- “3. Assess Which Existing Files To Carry Forward and Test” on page 11
- “4. Package and Test” on page 11

1. Identify Technical and User Requirements, Subject Matter Resources, and Risk Mitigation

Large-scale terminal emulation deployments have significant risks in terms of delays, cost, and user acceptance. Performing a high level assessment allows you to identify risks at an early stage, and plan mitigation strategies to address them. Be sure to:

✓**Develop communication plans**

Develop plans to communicate with user groups and key personnel throughout the process to avoid the communication problems that are common to many deployments. User organizations may not be aware of their licensing options, when they have to upgrade, or which resources and information to provide. IT staff are sometimes unaware of desktop macros and other customization files that are required for user groups.

✓**Coordinate to make sure all stakeholder needs are met**

Make certain all user groups, IT, and stakeholders are "on the same page" and are prepared regarding what is needed, schedules, and expectations.

✓**Define high level requirements**

Define high level requirements by collecting and analyzing all the information required to define and prioritize needs, address user concerns, and improve user acceptance.

✓**Define resource requirements**

Define resource requirements to determine how many and what type of resources your deployment requires.

✓**Assess customization requirements**

Assess your current environment to determine which customizations are required for user acceptance. Be sure to address special security requirements or other needs such as file transfer.

✓**Assess risks**

Assess the risks of application compatibility and user acceptance.

✓**Perform a high level analysis of your solution**

Make sure your deployment solution complies with new security mandates, reduces maintenance, meets productivity requirements and still has good user acceptance.

2. Inventory and Analyze User Requirements, Macros, Configuration Files, and Legacy Applications

Conduct a detailed inventory for each user group to determine which user applications and configurations are critical, used, or not used and whether your terminal emulator requires integration with custom applications. Performing an inventory of your current emulation collateral provides the data you need to make sound technical and business decisions about what to carry forward. It also helps identify needs for customization of Reflection and for integration with other applications. Be sure to:

✓**Define user requirements**

Define user requirements for each user group to determine the priorities for this group and special needs such as file transfer capability or security requirements.

✓**Inventory User Desktops**

Inventory desktops to assess how many vendor products and files are being used in the existing configuration.

✓**Identify integration needs**

Identify needs for integration with HLLAPI or other applications that use your terminal emulation software.

3. Assess Which Existing Files To Carry Forward and Test

The success of your deployment depends on careful assessment and planning. After you determine which files to carry forward, you'll need to convert them to Reflection Desktop files and then test these new files.

✓**Assess which existing files are required**

Assess which macros and configuration files are required for your new solution.

✓**Analyze inventory data**

Analyze your inventory data to make key technical and business decisions about what to carry forward and deploy.

✓**Convert Reflection 14 or Extra! files to Reflection Desktop files**

If you plan to carry forward any existing files, open the files in Reflection Desktop and save them (and any related supporting files) as Reflection Desktop files.

If any of your Reflection 14 or Extra! session files have macros that are locked and password-protected, these macros are lost when you open those files in Reflection Desktop. To import the macros, you'll need to unlock the macro projects before you open the files.

NOTE: Reflection Desktop cannot import legacy macros when they are locked because it needs to access the macro projects to convert them.

✓**Test all files in Reflection Desktop**

Make sure that the macros, configurations, and other files work with Reflection Desktop.

4. Package and Test

Package, test, and deploy to selective user groups and conduct pilots to minimize user disruption. Be sure to test for both technical issues and user acceptance.

Use this checklist to avoid common problems with Reflection deployments.

✓**Deploy files to trusted locations.**

The Trusted Locations feature provides a way to differentiate safe files from potentially harmful files. When a file is in a trusted location, it's files are assumed to be safe. Reflection enforces trusted locations by default, so if you want to save sessions in directories that are not default trusted locations, you'll have to define these directories or disable trusted locations. See "Add Trusted Locations" on page 47

✓**Deploy customized files to the correct locations**

Reflection looks for session files and supporting custom files in specific locations. If you deploy one of these files to another location, Reflection will be unable to find it. See "Customized Files and Where to Deploy Them" on page 76 in the Reflection Deployment Guide.

✓**Consider using the Reflection setup.exe**

Setup.exe is the recommended tool for installing and deploying Reflection. This tool uses the primary Reflection MSI file to install Reflection but it also installs prerequisite software (if needed) and has several other features that provide a smoother deployment than installing directly with the primary Reflection MSI file.

It determines whether each workstation has the required .NET Framework and Microsoft Windows Installer version and automatically installs them if necessary. It also automatically uses the correct language for the installation and removes previous versions of Reflection.

If the Visual Basic feature is selected, Setup.exe also installs the Visual Basic core MSI, along with the appropriate VB language MSI.

✓**If you are deploying with MSI directly, make sure all prerequisites are installed.**

If you use msiexec.exe for your install, prerequisites are not installed automatically. If the required prerequisites are not already on your users' workstations, you need to install them separately. You can find installers for the required prerequisites in the Prerequisites folder in the distribution media, or in your administrative installation. The prerequisites you need to install depend on which programs and features you are installing:

- All Reflection Workspace features require Microsoft .NET Framework 4.7.1. If you attempt an install using msiexec.exe and this prerequisite is not found, a message displays and the installer stops. To install the .NET Framework, run the executable file in `Prerequisites\DotNet471`.
- The Visual Basic for Applications feature requires Microsoft VBA 7.1. Use the core and language-specific *.msi packages in the `Prerequisites\VB71` folder.

✓**Ensure your user systems meet Reflection's Requirements**

To make sure users have the Reflection hardware and software requirements, see "System Requirements" on page 14.

✓**If your users have legacy macros, install legacy macro compatibility features**

To run a legacy macro, the Micro Focus compatibility feature for that type of macro must be installed. This feature is available on the Reflection Setup program Features tab, under 3270/5250 | Compatibility. See "Select Features, Components, and Languages" on page 84.

✓**Make sure that all VBA projects in referenced session documents and SharedMacros files have unique project names**

If you are referencing session documents or using SharedMacros files to share macros, make sure the projects in those files have unique project names. Each project name in the VBA Project editor must be unique to avoid errors caused by naming conflicts. You can change project names by modifying the project properties in the VBA editor or by creating and saving these files in Reflection Desktop 16.2 or greater.

✓**Follow guidelines for setting up security certificates**

You can configure certificate authentication for both Secure Shell and SSL/TLS connections.

All SSL/TLS sessions require certificates for host authentication; without the necessary certificate, you cannot make a host connection. Depending on the host configuration, you may also need to install certificates for user authentication.

Secure Shell sessions typically require both host and user authentication. Certificates can be used for either host and/or user authentication, but are not required by default.

For more, see "Digital Certificates and Reflection Certificate Manager" on page 32.

✓**If you are using Citrix or AppV, Follow best practices for these platforms**

Whenever possible, save sessions that have customized settings for QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons as compound session document files (see "Create and Customize Session Documents" on page 29). This prevents problems caused by sessions not being able to access or find configuration files.

If you specify a user data directory, do not use the Setup user interface to set the directory and do not use a drive letter as part of the directory. Setting the user data directory in this way on a Citrix server causes all users to have the same data directory. Instead, create a transform that sets the WRQ_USERDIR property to the location you want to use for the user data directory (see "Add/Modify Registry Data" on page 83) and install Reflection with the transform. Be sure to use a Reflection property (such as `[PersonalFolder]`) to specify the directory instead of a drive letter.

Installation

You can install Reflection from a downloaded distribution, an administrative installation image, or from a CD. If you are an administrator looking for instructions on how to create a customized installer, see the Reflection Deployment Guide.

NOTE: Reflection is typically distributed electronically. If your installation requires a CD, you will need to request it when you place your order.

- "System Requirements" on page 14
- "Upgrading from Previous Versions" on page 15
- "Install Reflection on a Workstation" on page 16
- "Features Selection Tab" on page 17
- "Set Your User Data Directory" on page 17
- "Advanced Tab" on page 17

System Requirements

Requirements for Reflection Desktop can vary, depending on your hardware and other software components present.

Hardware Requirements and Supported Operating Systems

Processor	2 GHz, 32-bit or 64-bit (1.5 GHz or higher multi-core, 32-bit or 64-bit recommended)
System memory (RAM)	2 GB (4 GB recommended)
Operating system and platform	One of the following:
	Microsoft Windows 10 Pro 32-bit and 64-bit
	Microsoft Windows 10 Enterprise 32-bit and 64-bit
	Microsoft Windows 8.1 Pro 32-bit and 64-bit
	Microsoft Windows 7 Enterprise 32-bit and 64-bit
	Microsoft Windows 7 Ultimate 32-bit and 64-bit
	Microsoft Windows Server 2019 with Remote Desktop Services (for multi-user environments)
	Microsoft Windows Server 2016 with Remote Desktop Services (for multi-user environments)
	Microsoft Windows Server 2012 R1 or R2 with Remote Desktop Services (for multi-user environments)
Additional software requirements	To use the Office integration features in Reflection, Microsoft Office 2013 or later must be installed.

NOTE: Where listed in the Hardware Requirements and Supported Operating Systems table, 64-bit operating systems are supported by the 32-bit Reflection Desktop product. The 64-bit .NET Framework Runtime is used on Windows 64-bit operating systems, which also supports native 64-bit applications calling the Reflection .NET API.

Prerequisite software — Microsoft .NET Framework

Reflection Desktop requires .NET Framework 4.7.1 or later. If the required .NET Framework isn't installed, the Reflection Setup program installs version 4.7.1 on all systems except Microsoft Windows Server 2008 R1 and later.

Notes:

- **If you install Reflection directly with MSI**and the product features you install do not require .NET 4.7.1, you can add the SKIPDOTNET=1 property to the command line to bypass this test condition.

Microsoft Windows Installer 4.5

The Microsoft Windows Installer (MSI) version 4.5 is distributed with Reflection Desktop.

Microsoft Visual Basic for Applications (VBA)

Reflection Desktop supports Microsoft VBA 7.1. If you select to install this feature in the Reflection Setup program, it is automatically installed. If you install Reflection Desktop directly with MSI or with a deployment tool and you want to install this feature, you must install it directly, using the core VBA 7.1 MSI and an appropriate language specific MSI (these MSIs are in the Prerequisites folder, in the distribution media).

Windows 10 “Windows as a Service” Support Statement

Reflection Desktop is supported on the following editions of Windows 10: Windows 10 Education, Windows 10 Pro, and Windows 10 Enterprise. Reflection Desktop is supported on the in-market supported servicing branches (https://technet.microsoft.com/en-us/windows/release-info.aspx) of Windows 10 including: Current Branch, Current Branch for Business, and Long-Term Servicing branches.

Supported Virtualization Platforms

Reflection Desktop is supported with the following virtualization platforms:

- Citrix XenApp

  NOTE: Note: Reflection Desktop TouchUx User Interface Mode is supported with Citrix 6.5 or higher. Reflection is supported only on Citrix versions currently supported by Citrix.

- Microsoft Application Virtualization (App-V)
- VMware Workstation/Player
- VMware ThinApp

Upgrading from Previous Versions

When you install Reflection Desktop, it upgrades the following Micro Focus products:

- Reflection for IBM 2007, 2008, 2011, and 2014
- Reflection for UNIX and OpenVMS 2008, 2011, and 2014
- Reflection for IBM v. 8.0 - 14.x
- Reflection for UNIX and OpenVMS v. 8.0 - 14.x
- Reflection for the Multi-Host Enterprise,Standard Edition v. 8.0 - 14.x
- EXTRA! X-treme v. 8.0 - 9.x
- myEXTRA! Enterprise v. 7.0
- myEXTRA! v. 7.x
- EXTRA! Enterprise 2000

NOTE: Reflection 2007, 2008, 2011, and 2014 are automatically removed when you upgrade. EXTRA! and the other Reflection products are automatically removed when you upgrade using setup.exe. If you upgrade these products by deploying the .msi file directly, you will get a message telling you to uninstall the older software first.

For detailed information about upgrading, see Technical Note 2808 (http://support.attachmate.com/techdocs/2808.html).

Supported Reflection Configuration Files

Reflection Desktop represents an upgrade over previous versions of both Reflection and EXTRA!. Most configuration files used with these products are supported by the new version.

For a complete list of legacy Reflection and EXTRA! files supported by Reflection Desktop, see Technical Note 2807 (http://support.attachmate.com/techdocs/2807.html).

Install Reflection on a Workstation

NOTE: You must log on with administrator privileges to install Reflection. If you do not have the necessary access rights, ask your system administrator to elevate your privileges.

To install on a workstation

1 Run the Reflection Setup program.

If you install from	Do This
A download site	Click the download link, and then run the download program. Select a location for the installer files, and then click **Next**. This extracts the files to the specified location and starts the Reflection Setup program.
An administrative installation image	From the administrative installation point, double-click the `setup.exe` file.

2 From the Reflection Setup program, click **Continue**, and then accept a license.

3 (Optional) To change the default installation folder, click the **File Location** tab and browse to the folder in which you want to install Reflection.

4 (Optional) To select which features, components, or languages are installed, click the **Feature Selection** tab.

5 Click **Install Now**.

NOTE: Use the **Advanced** tab of the installer only if you want to modify the installer log settings, or if you are an administrator configuring a Reflection deployment. An administrative installation does not actually install the product — instead, it creates an installation image that administrators can use to customize and deploy Reflection to end users.

Features Selection Tab

Use the **Feature Selection** tab to select which features you want to install.

Click the icon to the left of the feature name and select from the options below.

Option	Description
Feature will be installed on local hard drive	Installs the selected feature. **NOTE:** Some features listed under a selected feature may not be included when you select to install the higher-level feature. The features that are included are the recommended defaults. If you select the higher level feature a second time, all sub-features will be included.
Feature will be installed when required	Installs the feature when you first use it (for example, when you click the **Start**menu shortcut for this feature).
Feature will be unavailable	Leaves the feature uninstalled.

Set Your User Data Directory

You can specify any location as the user data directory. The default directory is Documents\Micro Focus\Reflection\.The directory you specify will be added as a Trusted Location.

To set the user data directory

1. Run the Reflection Setup program as shown in "Install Reflection on a Workstation" on page 16.
2. On the Reflection Setup program, click **File Location**, and then under **Default user data directory**, browse to the directory you want to use.

NOTE: This change only affects those features that you include in your initial installation. If you add a feature later using Programs and Features (or Add/Remove Programs), that application will use the original default directory (`Documents\Micro Focus\Reflection\`), not the value you specified for **Default user data directory**.

Advanced Tab

Use the **Advanced** tab of the installer only if you want to modify the installer log settings or you are an administrator configuring a Reflection deployment.

Install to this PC

Installs Reflection to your computer.

Create an Administrative install image on a server

NOTE: An administrative install image does not actually install the product — instead, it creates an installation image that administrators can use to deploy Reflection to end users.

When you create an administrative install image, an image of Reflection is copied to a network location for later installation to multiple workstations. This network location can be used by deployment tools to access and create packages that are deployed to workstations. Also, end users can perform installations by running `setup.exe` from this location.

Log file settings

By default, an installation log file is created and then deleted after installation successfully completes. (This configuration avoids accumulation of large log files after successful installations.) To save a log file for all installations, including successful ones, select **Create a log file for this installation**, and clear **Delete log file if install succeeds**.

The installation log file, which provides details about the installation, is saved in the user's Windows temporary folder `(%tmp%)` with a generated name that begins with `atm`. To open this directory, launch the **Start**menu **Run**command and enter `%tmp%.`

Setting up the Reflection Administrative Tools

Reflection uses the Microsoft Windows Installer application installation and configuration service. The Windows Installer gets installation information from a relational database, which is saved and deployed as a Microsoft MSI file. When you deploy an MSI file to a user workstation, the Windows Installer on the workstation accesses the information in the MSI file to perform the installation.

Use the following tools to customize and deploy Reflection. These tools are installed as part of the administrative install image.

About the Tools

Reflection Setup program

The Reflection Setup program (`setup.exe`) is the recommended tool for installing and deploying Reflection. This tool uses the primary Reflection MSI file to install Reflection but it also installs prerequisite software (if needed) and has several other features that provide a smoother deployment than installing directly with the primary Reflection MSI file. When the Setup Program installs Reflection, it determines whether each workstation has the required .NET Framework and Microsoft Windows Installer version and automatically installs them if necessary. It also automatically uses the correct language for the installation and removes previous versions of Reflection. (This is required to install the new version.) If the Visual Basic feature is selected, the Attachmate Installation Program also installs the Visual Basic core MSI, along with the appropriate Visual Basic language MSI.

NOTE: If you install with MSI directly, you will need to install the .NET Framework and Microsoft Windows Installer version directly and remove any previous versions of Reflection. The MSI installer uses English for the installation unless you specify another language on its command line. The Reflection MSI does not install Visual Basic. If you install with MSI directly, you must run the Visual Basic core and language MSIs (in the Prerequisites folder) directly.

The Setup program has a command-line interface that you can run from a command line, a batch file, or a deployment tool. You can set command-line options to preset installation parameters and limit user interaction as Reflection is installing. You can also suppress installation dialog boxes to provide an unattended installation or use command-line options to prepare Reflection for installation by users. In general, any of the MSI command-line options can be used from the Setup program command line.

To see a list of available command-line parameters, enter:

```
setup.exe /?
```

Installation Customization Tool

You can use the Installation Customization Tool (ICT) to customize Reflection or customize the way it is installed. The tool lets you create the following files:

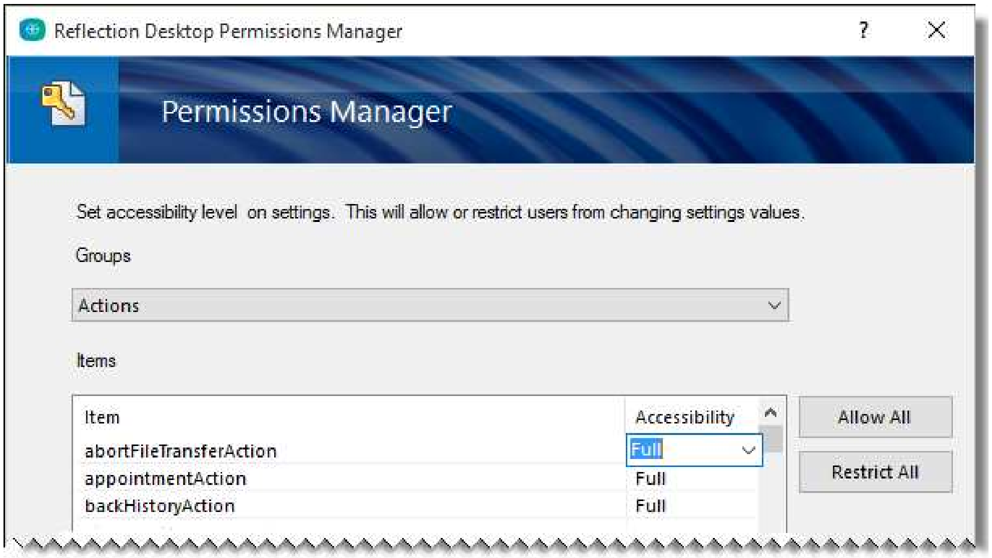
- **Companion installer package**(MSI). This package contains the customized configuration settings and files that you choose to deploy with the installation (or independently). Companion packages show up as independent entries in the Windows list of installed applications. To customize Reflection, see "Configure Workspace Behavior and Appearance" on page 51.
- **Transform file**(MST). This file modifies the default installation to change how Reflection is installed (For example, remove a command button in the Windows Uninstall or change a program list). When the transform is deployed along with the Attachmate Reflection MSI file, it changes the default installation settings to the settings specified in the MST file.

This tool is accessed by running `setup.exe` from the command line with the admin switch (`setup.exe /admin`). To simplify working with this tool, you can create a desktop shortcut to the Setup program and add the admin switch on the command line as shown in "Set up a Shortcut to the Installation Customization Tool" on page 22.

Permissions Manager

Permissions Manager is used to restrict access to Reflection Desktop settings and features. It creates special configuration (`.access`) files that can be deployed as part of an MSI package. There are different access files for mainframe, AS/400, UNIX/OpenVMS, and application-wide settings.

You can edit .access files by running Permissions Manager as shown in "Specify Access Using Permissions Manager" on page 66.



Set accessibility level on settings. This will allow or restrict users from changing settings values.

Groups

Actions

Items

Item	Accessibility
abortFileTransferAction	Full
appointmentAction	Full
backHistoryAction	Full

Set up an Environment for Deployment Using Reflection Administrative Tools

If you plan to use the Reflection administrative tools to customize session documents, the Reflection workspace, or the way that Reflection is installed, you'll need to set up an environment that you can use to access these administrative tools.

NOTE: There are two types of Reflection installations. The Administrative install image installs the files required for installation but does not install any values in the registry required to open and run Reflection. You cannot run Reflection from an Administrative installation image. The workstation (or PC) installation enters the values in the registry required to run the product.

Create an Administrative Installation Point

You create an administrative installation point by installing an administrative install image of Reflection on a network share (typically on a file server). An administrative install image is a source image of the application, similar to an image on a CDROM. It includes all the files required to install Reflection as well as the administrative tools used for customization.

Use this procedure to create an administrative installation point on a networked file server. This places all of the administrative tools and installation files you need to customize and install Reflection to a single location.

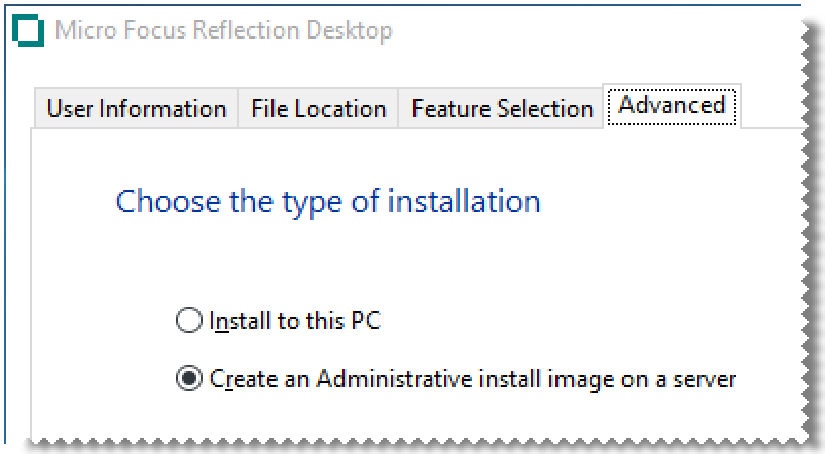
NOTE: Micro Focus recommends to create an administrative installation point before you install Reflection on a workstation. This allows you to use the administrative installation point for the workstation installation. If you are setting up the administrative point and the workstation installation on a single workstation for testing purposes, you must perform the administrative installation first.

To create an administrative installation point

1. Download the Reflection installation files.

2. Create a network share on a network file server.
3. From the root directory of the installation files, double-click `setup.exe`.

   This starts the Reflection Setup program.
4. Click **Continue**and accept the license.
5. From the the **Advanced**tab, click **Create an Administrative install image on a server**.



Choose the type of installation

○ Install to this PC

● Create an Administrative install image on a server

6. Click **Continue**.

   The **File Location** tab is selected.
7. Browse to the network share you want to use for the administrative install image.

   CAUTION: Important! Be sure to specify a UNC path for the network share. For example: `\\share_name\administrative_install_point`
8. Click **Install Now**.
9. The next step in preparing a test environment is to install Reflection on a workstation (page 21).

NOTE: Administrative install images are typically created in a file server folder but you can create them in any folder on a local hard drive. This is useful for testing purposes.

If you prefer to install Reflection on your workstation first, you cannot use the installation program graphical interface to create the administrative install image. Instead, you must install it from the command line as follows:

path_to_setup_file `\setup.exe /install /admin TARGETDIR=`
UNC_path_to_administrative_installation_point

Install Reflection on a Workstation

After you create an administrative installation point, you'll need to install Reflection on a workstation so that you can open and run Reflection. If you plan to customize Reflection, you will use this installation to create custom configuration files.

To install Reflection on a workstation

1 On the workstation, navigate to the network share where you have created the administrative installation point, and double-click `setup.exe`.

2 From the Reflection Setup program, click **Continue**, and then accept a license.

3 The following steps are optional:

 3a To personalize the installation, click the **User Information** tab and enter the name, organization, and Volume Purchase Agreement (VPA) number, if you have one. (VPA numbers, which are issued by Micro Focus, are used by customer support to expedite service requests.)

 3b To change the default installation folder or the default user data directory, click the **File Location** tab and browse to the folder you want to use.

 3c To select which features, components, or languages are installed, click the **Feature Selection** tab. The default installation does not include all features and components.

4 Click **Install Now**.

Set up a Shortcut to the Installation Customization Tool

The Installation Customization Tool is designed to be opened from a command line. However, you can create a desktop shortcut that opens this tool. This will save time when you are working with the tool.

NOTE: To start the Installation Customization Tool from a command line, change to the administrative installation point and enter:

<path_to_setup> `\setup.exe /admin`

To create a shortcut that opens the Installation Customization Tool

1 On your administrative installation point, right-click the setup.exe file, and choose **Create Shortcut**.

2 Right-click the shortcut and choose **Properties**.

3 In the **Target**box, add the /admin option to the end of the command line. For example:

 `\\` *myServer\adminInstallPoint* `\setup.exe /admin`

 CAUTION: Make sure that the path in the **Target**box conforms to the Uniform Naming Convention (UNC) format and does not include drive letters. Drive letters can cause problems if you try to use the shortcut on other workstations.

4 Rename the shortcut and save it on the desktops of your workstation and the server you use for your administrative installation point.

5 To start the Installation Customization Tool, double-click the shortcut, and then in the **Select Customization**dialog box, choose which mode you want to open.

Determining Customization Requirements

There are a number of ways to customize the Reflection sessions, settings, and installation:

- "Create and Customize Reflection Sessions" on page 23 that include settings required to connect to specific hosts. You can set up traditional telnet connections or secure connections for SSL/TLS, SSH, Kerberos, SOCKS or HTTP proxy servers. You can also configure Reflection Sessions to use custom settings such as custom keyboard and mouse maps.
- "Customize the Reflection Workspace" on page 24 to change it's appearance and functionality or to lock down access to settings and controls.
- "Customize to Protect Data and Information Privacy" on page 25 to define locations from which you can safely open (and store) documents, mask sensitive data (such as credit card numbers), or control access to the Reflection API.
- "Customize the Installation" on page 26 to create transforms (.mst files) that customize how Reflection is installed on user workstations.

NOTE: Reflection saves settings in custom configuration files. After you create the custom settings, you will need to package and deploy the configuration files you create as shown in "Package Sessions and Custom Settings Files" on page 73.

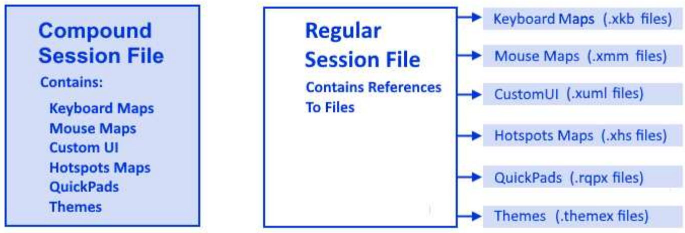
If you create a transform to customize the installation, you'll need to deploy the transform with the base installation.

Create and Customize Reflection Sessions

Reflection stores the information required to connect to hosts in configuration files called session document files. You can create session document files that have connection information and other settings and then deploy them independently of your installation.

You can also customize session document files to use custom keyboard maps, controls, themes, and other settings. To customize these settings, see "Create and Customize Session Documents" on page 29.

By default, Reflection saves custom settings for QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons in separate configuration files that you will need to deploy along with the session document files that reference them. But you can simplify your deployment by saving your session documents as compound session files, which include all of these settings.



Compound Session Document files include all of the customized settings for QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons. These files allow you to deploy your sessions without having to deploy dependant files for these settings.

Session Document files save these settings in separate files. If you save sessions using this default, you'll need to make sure that you deploy the custom files to the correct locations as shown in "Customized Files and Where to Deploy Them" on page 76.

Compound files are easier to deploy because you don't have to deploy the supporting files. Regular session files offer more flexibility for sharing common configurations. For example, you can reference one custom keyboard map from several regular session files. With compound session documents, you have to recreate the custom settings in each file.

NOTE: Not all custom settings are saved in compound session files. For example, settings such as ssh_config and ssh_known_hosts required for Secure Shell connections are not saved in these files.

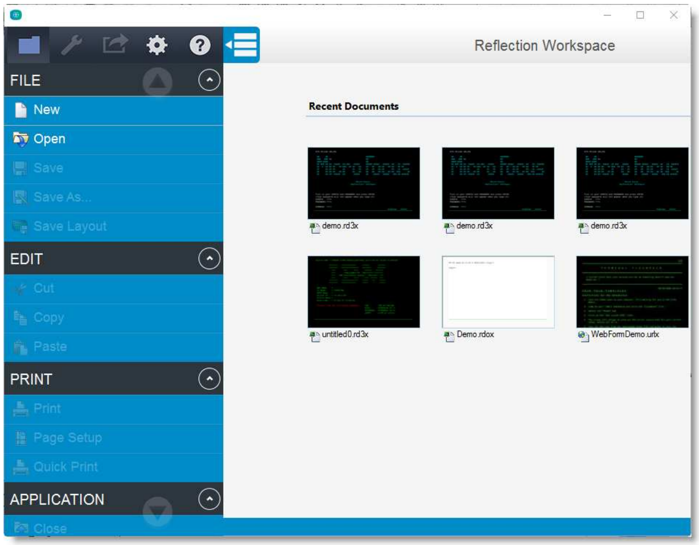
Customize the Reflection Workspace

You can customize Reflection by changing it's appearance and functionality and by locking down access to settings and controls.

Change Reflection Appearance and Functionality

You can change the appearance and basic functionality of the main Reflection window in a variety of ways, including specifying startup macros and actions.

You can also change the appearance of the window (for example, by setting Reflection to open in TouchUx mode).



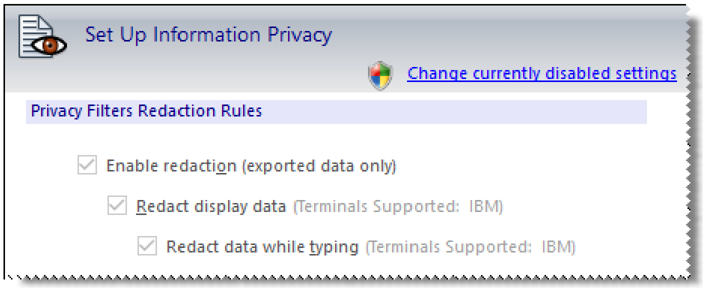
For more, see Chapter 4, "Customize the Reflection Workspace," on page 51

Locking Down Settings and Controls

You can lock down Reflection to limit access to settings and controls so that they are not available to users. This allows you to simplify support requirements and resolve security concerns.

To prevent a user from changing a setting, you set the permission level for that setting or control to “Restricted.” For example, you could restrict the users’s ability to modify security settings.

As shown below, the restricted settings are grayed out. The security shield and the **Change currently disabled settings** link indicate administrative access is required to change them.



Set Up Information Privacy

Change currently disabled settings

Privacy Filters Redaction Rules

☑ Enable redaction (exported data only)

☑ Redact display data (Terminals Supported: IBM)

☑ Redact data while typing (Terminals Supported: IBM)

For more, see:

- “Control Access to Settings and Controls with Reflection Administrative Tools” on page 62
- “Control Access to Settings and Controls with Microsoft Group Policy” on page 68

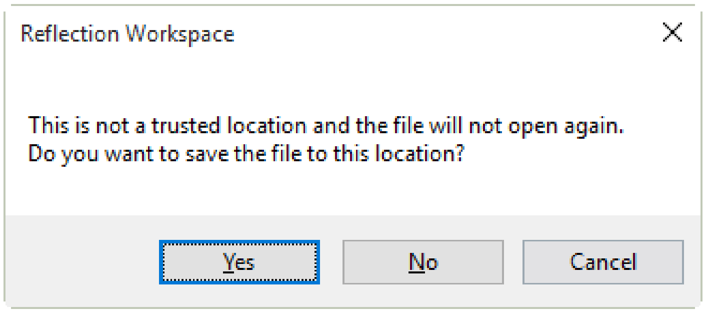
Customize to Protect Data and Information Privacy

From the Reflection Trust Center, you can set up Reflection to protect your working environment from information theft, and your data from potential damage caused by opening documents from non-trusted sources, using the following methods:

- “Set up Trusted locations” on page 25, from which you can safely open (and store) documents.
- “Set up Information Privacy” on page 26 to mask sensitive data (such as credit card numbers) with privacy filters.
- “Set up macro and API security” on page 26 to control access to the Reflection API and control the execution of actions invoked by a macro or API call.

Set up Trusted locations

Define Trusted Locations to differentiate safe files from potentially harmful files. When a file is in a trusted location, it’s files are assumed to be safe. If you try to save a file in a location that is not trusted, Reflection warns that it will not be able to reopen the file.

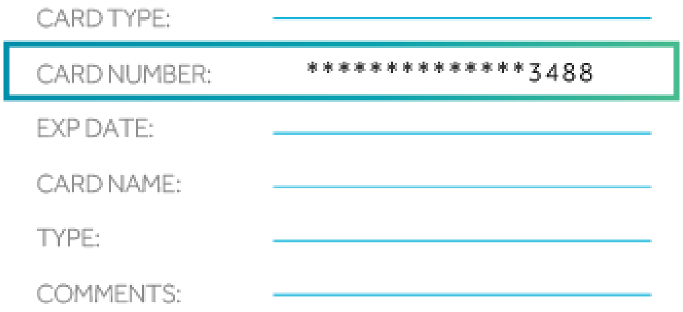

Reflection Workspace

This is not a trusted location and the file will not open again.
Do you want to save the file to this location?

Reflection enforces trusted locations by default, so if you want to save sessions in directories that are not default trusted locations, you'll have to define these directories or disable the Trusted Locations feature. See "Add Trusted Locations" on page 47.

Set up Information Privacy

Set up Information Privacy to help comply with PCI-DSS requirements, including redaction of sensitive data such as credit card or social security numbers.


CARD TYPE:

CARD NUMBER: **************3488

EXP DATE:

CARD NAME:

TYPE:

COMMENTS:

You can also set up the Reflection API to log access to unredacted data. See "Configure Information Privacy" on page 48.

Set up macro and API security

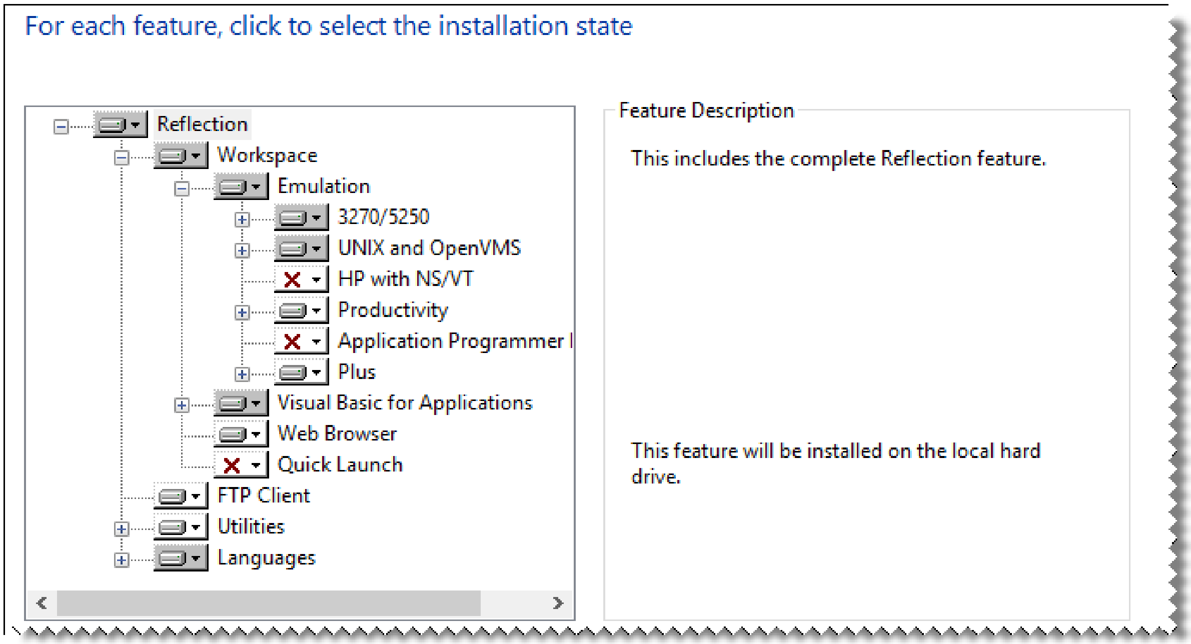
You can enable or disable the Reflection .NET API, determine whether Reflection legacy macros are supported, and determine which legacy API has preference for the GetObject() method used to retrieve API COM objects.

You can also specify whether to run restricted actions that are initiated through a macro or API call without elevating permissions.

Customize the Installation

Create and deploy a transform to customize how Reflection is installed on user workstations. As defined by Microsoft, "a transform is a collection of changes applied to an installation. By applying a transform (*.mst) to a base installation package, the installer can add or replace data in the installation database."

For example, by deploying the transform with the Reflection base installation package, you can select which features to install.



For each feature, click to select the installation state

This includes the complete Reflection feature.

This feature will be installed on the local hard drive.

You can also specify the installation directory or the user data location, change the Remove or Add commands from the Windows Uninstall or change a program list, and change other default settings.

For more about creating transforms, see Chapter 6, “Modify the Installation,” on page 81

2 Create and Customize Sessions

Reflection provides a number of options for customizing and deploying sessions.

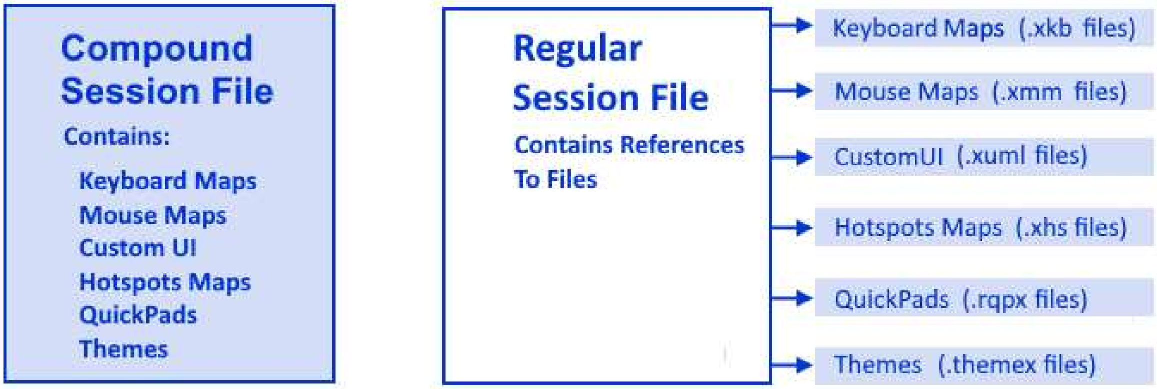
This article	Describes
Create and Customize Session Documents	How to create session document files and customize them to configure text input, appearance, macros, and other settings. This article also describes the advantages and of saving sessions in compound session document files that include all of these settings in a single file.
Create SSL/TLS or SSH Session Documents	How to create secure session document files that include settings for SSL/TLS or SSH connections.
Set up Session Templates	How to create a custom session template configured to use specific settings (for example, an SSL/TLS terminal session with a specific TLS level of encryption).
Configure Reflection for PKI Auto Sign-on	How to use the PKI Auto Sign-on Add-On Client product with Reflection to allow the use of a Common Access Card (CAC) or other smart card for authentication.
"Centrally Manage VBA Macros" on page 37	Shows two approaches for centrally managing VBA macros.

Create and Customize Session Documents

Basic connection and terminal settings for Reflection sessions are saved in session document files that you can configure and deploy independently of the product installation.

After you create a session document, you can customize it to configure text input, appearance, macros, and other settings. You can also specify or customize the files referenced in a session document, such as QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons.

You can save these session settings as compound session files or as standard session document files (the default).



Compound Session Document files include all of the customized settings for QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons. When you Save a session as a compound session document file, you can deploy the session without having to deploy dependant files for these settings.

Session Document files save these settings in separate files. When you save a session in a regular session document file, all custom settings for keyboard maps, themes, mouse maps, hotspots, and other items are saved in separate files that must be deployed with the session file. You'll need to make sure that you deploy these custom files to the correct locations as shown in "Customized Files and Where to Deploy Them" on page 76.

Compound files are easier to deploy because you don't have to deploy the supporting files. Regular session files offer more flexibility for sharing common configurations. For example, you can reference one custom keyboard map from several regular session files. With compound session documents, you have to recreate the custom settings in each file.

Use these procedures in the Reflection Desktop Guide to create and customize a session document file (that is, change its default settings for text input, appearance, macros, Hotspots and other settings).

To	See
Create and customize session settings that specify whether to automatically connect sessions, encrypt sessions, and configure other host connection options.	Set up Sessions and Connections Connect Using Kerberos Set up a SOCKS or HTTP Proxy Server Session
Create custom keyboard or mouse maps	Select and Map the Keyboard and Mouse
Set options for using Microsoft Office Tools such as Recent Typing, Auto Complete, or Screen History.	Set up Productivity Features
Create custom themes and set background and foreground colors, cursor shape, and other display options.	Change the Look and Feel of a Session
Create custom Quick Access Toolbars or context menus, set up Micro Focus Plus or Hotspots features, and import legacy toolbars	Set up Custom Controls for your Program Screens

To	See
Configure Reflection to initiate Reflection actions, such as Reflection macros, and menu and terminal commands when an event is encountered during a host session.	Set up Actions for Reflection Events
Add or remove tabs, groups, buttons, and menus to the Ribbon to create custom Ribbons.	Customize the Ribbon
Perform common tasks such as editing the translation table or setting up customized host files.	Perform Other Common Tasks

Walkthrough: Set up and Customize a Session

The following example shows how to create a custom keyboard map file and set up a session document file to use this keyboard map. This keyboard map assigns the Ctrl+E key combination to the EraseEOF Send Key. The session is saved as a regular session file so you can see the additional work you need to perform to deploy it (as opposed to saving it as a compound session file).

To walk through the process for creating and deploying a compound session file, see Setting up a Simple Deployment in The Getting Started with Reflection Desktop Deployment Quick Start Guide.

First, create and save the session.

Create a session

1. Open Reflection on your workstation.
2. In the Create New Document dialog box, under **Built-in Templates**, choose a type of session (for our example, we will choose 3270 terminal) and click **Create**.
3. In the Create New Terminal Document dialog box, in the **Host name/IP address** box, enter the host name (for example, IBM390 or 10.9.1.151).
4. From the Reflection Quick Access toolbar, click the Save button and save the session as `mySession.rd3x`.

   This file is saved in the `My Documents\Micro Focus\Reflection` folder.

Then configure the session with a keyboard map.

Customize the session to reference a custom keyboard map

1. With a session open in Reflection (for example, mySession.rd3x), on the Quick Access toolbar, click the Document Settings button.
2. In the Settings dialog box, under **Input Devices**, click **Manage Keyboard Map**.
3. In the Manage Keyboard Map dialog box, click **Create a new keyboard map from an existing keyboard map file**.
4. In the Create a New Keyboard Map file dialog box, select a keyboard map file to use as a template for your new file (for example, `Default 3270.xkb`) and then select **Use the new file in the current session document** and click **OK**.
5. In the Keyboard Mapper dialog box, click in the **Press the key or key combination that you want to map** box. Then hold down the Ctrl key and press the E key to enter Ctrl+E in this box.

6. Click the **Select Action** button.
7. In the Select Action dialog box, in the **Action** list, select **Send Key**.
8. Under **Action parameters**, in the **Key** list, select **Erase EOF** and click **OK**. (The Ctrl+E key combination is displayed as a key combination in the Keyboard Mapper table.)
9. Click **OK**. When prompted, save the new keyboard map in the Keyboard Maps folder as `myKeyboardMap.xkb`.
10. If you are prompted that this is not a secure location, accept to save it in the default location. (The file is saved in your `Documents\Micro Focus\Reflection\Keyboard Maps` folder.)
11. To package the session file and the keyboard map, see "Walkthrough: Create a Package with the Installation Customization Tool" on page 75.

Create SSL/TLS or SSH Session Documents

When you create a Reflection session document, configure it to use the security protocols your organization requires.

This connection type	Supports these protocols
3270 terminal or printer	SSL/TLS, SOCKS
5250 terminal or printer	
6530 terminal*	SSL/TLS, Secure Shell
VT terminal or FTP Client	SSL/TLS, Secure Shell, Kerberos, SOCKS, HTTP

- "Digital Certificates and Reflection Certificate Manager" on page 32
- "Set up SSL/TLS Connections" on page 34
- "Set up Secure Shell Connections" on page 34

Digital Certificates and Reflection Certificate Manager

You can configure certificate authentication for both Secure Shell and SSL/TLS connections.

- All SSL/TLS sessions require certificates for host authentication; without the necessary certificate, you cannot make a host connection. Depending on the host configuration, you may also need to install certificates for user authentication.
- Secure Shell sessions typically require both host and user authentication. Certificates can be used for either host and/or user authentication, but are not required by default.

Certificate authentication solves some of the problems presented by public key authentication. For example, for host public key authentication, the system administrator must either distribute host keys for every server to each client's known hosts store, or count on client users to confirm the host identity correctly when they connect to an unknown host. When certificates are used for host authentication, a single CA root certificate can be used to authenticate multiple hosts. In many cases the required certificate is already available in the Windows certificate store.

Digital certificates are maintained on your computer in certificate stores. A certificate store contains the certificates you use to confirm the identity of remote parties, and may also contain personal certificates, which you use to identify yourself to remote parties. Personal certificates are associated with a private key on your computer.

You can use digital certificates located in all of the following stores:

- **The Windows Certificate Store**

  This store can be used by a number of applications, web browsers, and mail clients. Some certificates in this store are included when you install the Windows operating system. Others may be added when you connect to internet sites and establish trust, when you install software, or when you receive an encrypted or digitally signed e-mail. You can also import certificates manually into your Windows store. Manage the certificates in this store using the Windows Certificate Manager.
- **The Reflection Certificate Manager Store**

  This store is used only by Micro Focus applications. To add certificates to this store, you must import them manually. You can import certificates from files and also use certificates on hardware tokens such as smart cards.
- **Centralized Management Server**

  The Centralized Management Server provides an administrator the means to centrally manage, secure, and monitor users' access to host applications. Administrators can deploy centrally managed sessions and certificates to the user. Digital certificates through the centralized management server can only be enabled if the centralized management server is configured to provide users' access to host applications.

Reflection Certificate Manager

Use the Reflection Certificate Manager to manage certificates for use exclusively by Reflection. You can deploy certificates and settings per-user or for all users of the system.

- User-specific location: [PersonalFolder]\Micro Focus\Reflection\.pki
- Global location: [CommonAppDataFolder]\Micro Focus\Reflection\.pki

NOTE: These settings are not included in compound documents.

The procedures for opening the Certificate Manager depend on your product and session type.

To open the Reflection Certificate manager from the Secure Shell Settings dialog box

1. Open the **Reflection Secure Shell Settings** dialog box.
2. On the **PKI**tab, click **Reflection Certificate Manager**.

To open the Reflection Certificate manager from the Security Properties dialog box

1. Open the **Security Properties** dialog box.
2. On the **SSL/TLS** tab, select **Use SSL/TLS Security**.
3. Click **Configure PKI**.
4. Click **Reflection Certificate Manager**.

Set up SSL/TLS Connections

SSL/TLS connections use digital certificates for authentication. Depending on how your certificate was issued and the way your host is configured, you may need to install a host and/or personal certificate before you can connect using SSL/TLS.

- In 3270, 5250, and VT sessions, SSL/TLS connection settings are saved to the session document.
- In the FTP Client, SSL/TLS connection settings are saved to the FTP Client settings file (`*.rfw`).

To configure SSL/TLS in 3270, 5250, or VT terminal sessions

1. Open the **Create New Document** dialog box, select a session template and click **Create**.
2. Select **Configure additional settings,**and then click **OK**.
3. Do one of the following:
   - If you are setting up a 3270 and 5250 terminal session, under **Host Connection**, click **Set Up Connection Security**. Then, in the Configure Advanced Connection Settings dialog box, click **Security Settings**.
   - If you are setting up a VT terminal session, click **Configure Connection Settings**, confirm Network Connection Type is set to **Telnet**, and click the Back arrow button. Then, under **Host Connection**, click **Set Up Connection Security**.
4. From the **Security Properties** dialog box, select the **SSL/TLS** tab, and select **Use SSL/TLS security**.
5. Click **Configure PKI** to configure certificate settings.

To configure SSL/TLS in FTP Client Sessions

1. Start the FTP Client.
2. In the **Connect to Site** dialog box, select a site and click **Security**.
3. Click the **SSL/TLS** tab and select **Use SSL/TLS security**.
4. Click **Configure PKI** to configure certificate settings.

Set up Secure Shell Connections

Secure Shell connections are available for VT terminal sessions and to configure SFTP transfers using the FTP Client.

By default, Secure Shell connections use public key authentication for the host and username/password authentication for the user. If you configure non-default settings, they are saved for each host (or ssh configuration scheme) to the ssh configuration file. This file is used for all connections (VT sessions and the FTP Client). You can deploy these settings per-user or for all users of the system. These settings are not included in compound documents.

- User-specific configuration: [PersonalFolder]\Micro Focus\Reflection\.ssh\config
- Global configuration: [CommonAppDataFolder]\Micro Focus\Reflection\ssh_config

To configure a secure terminal session using Secure Shell (SSH)

1. Open the **Create New Document** dialog box, select the **VT Terminal** template and click **Create**
2. In the **Create New** dialog box, under **Connection**, select **Secure Shell** and click **OK**.
3. Click **OK**.

To configure non-default Secure Shell settings

1. Open a session that you have configured to use Secure Shell. Disconnect if you are connected.
2. Open the **Document Settings** dialog box.
3. Under **Host Connection**, click **Set up Connection Security**.
4. In the **Reflection Secure Shell Settings** dialog box, configure any non-default settings and then click **OK**.

   When you click **OK**, changes to the default settings are saved in the Secure Shell `config`file in [PersonalFolder] `\Micro Focus\Reflection\.ssh`

To configure username and password prompts to appear in the terminal window

1. Open a session that you have configured to use Secure Shell. Disconnect if you are connected.
2. Under **Host Connection**, click **Configure Connection Settings**.
3. Under **Connection Options**, select **Handle SSH user authentication in terminal window**.

Known Hosts

Host authentication (performed with public key authentication) enables the Secure Shell client to reliably confirm the identity of the Secure Shell server. If the host public key is not installed on the client, the host fingerprint is displayed and users are prompted to contact the system administrator to verify the fingerprint. This confirmation prevents risk of a "man-in-the-middle" attack, in which another server poses as the host. If you select Always in response to this prompt, the host key is saved in a file called `known_hosts`, which is created in [PersonalFolder] `\Micro Focus\Reflection\.ssh`. After the host key is added, Micro Focus Reflection Desktop can authenticate the server without requiring user confirmation, and the unknown host prompt does not appear again.

To prevent end-users from seeing the unknown host message you can deploy a known hosts file per-user or for all users of the system. These settings are not included in compound documents

- User-specific file: [PersonalFolder] `\Micro Focus\Reflection\.ssh\known_hosts`
- Global file: [CommonAppDataFolder]\ `Micro Focus\Reflection\ssh_known_hosts`

Set up Session Templates

After you configure a session document, you can share and reuse your settings by saving the document as a template. Templates provide an untitled copy of the original, giving you a quick and easy way to create pre-configured documents, while ensuring that your original file remains unchanged.

You can also use templates to control the types of sessions that users can create. For example, you can create templates that have pre-configured SSL/TLS settings. These templates appear in the **Create New Document** dialog box, under **User-defined**.

To create a session template

1. Open the session document that you've configured.

2 Save the session as a template.

The steps depend on your user interface mode.

Ribbon	On the **File** menu or the Reflection button (if using the Office 2007 Look and Feel), choose **Save As** and then **Save Template.**
Classic MDI	From the **File** menu, select **Save As Template**.

3 Name the template file with an `.rsft` extension.

To make changes to the template, you must replace the template file — save the file that contains your changes using the same filename and extension as the template.

NOTE: To deploy templates, install the .rsft files to the `templates` folder:

[AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 `\templates`\

Templates saved to this location appear in the **Create New Document** dialog box, under **User-defined**.

Configure Reflection for PKI Auto Sign-on

You can configure a Reflection Desktop session to use the PKI Auto Sign-on Add-On Client product, which allows the use of a Common Access Card (CAC) or other smart card for authentication.

To use PKI Auto Sign-on, the PKI Auto Sign-on host module must be installed on your host server. This module can be used to verify that a client is in control of a CAC or other smart card, and to extract the Distinguished Name (DN) from the certificate used for authentication. The DN, or some substring contained in the DN, can then be used to provide service to the authorized user. PKI Auto Sign-on is designed to provide a validated identity even via a shared host login, that is, the identity comes from the smart card itself, not from the host user ID.

When a session is configured to use PKI Auto Sign-on:

- System administrators can set up an OpenVMS or UNIX session to use a shared log-on that provides the host application with a strongly validated identity directly from a CAC.
- Host programmers can get the strongly validated DN of a user in control of a CAC. The programmers can then extract information from the DN and use it as an identifier to authorize access (for example, to the CAC-bearer's health records).

Prerequisites

- The Reflection PKI Auto Sign-on host module must be installed on the host server.
- You can use PKI Auto Sign-on with Reflection Desktop or Reflection 2014 R1 SP1 VT terminals using the SSH protocol. All of the client-side functionality required for PKI Auto Sign-on is included only in these product versions.

To create an SSH-enabled Reflection session that uses PKI Auto Sign-on

1 Create a new VT session document.

2 Click **Configure additional settings**and then click OK.

3 In the Settings dialog box, under **Host Connection**, select **Set up Connection Security**.

4 On the Reflection Secure Shell Settings dialog box General tab, under **User authentication**, deselect **Public Key**.

5 On the PKI tab, click **Reflection Certificate Manager**.

6 On the Reflection Certificate Manager dialog box PKCS #11 tab, click **Add**.

7 In the PKCS #11 Provider dialog box, browse to the Provider DLL required to access your CAC.

8 In the `.ssh/config` file for this session document, add the appropriate PKIC prompt string configured on the server. The following example shows an entry for a prompt “Starting PKI Validation...”

```
PKICPrompt "Starting PKI Validation..."
```

When you are done, the file should look like this:

```
Host myHostName
RSAAuthentication no
PubkeyAuthentication no
connectionReuse no
PKICPrompt "Starting PKI Validation..."
#EndHost
```

Centrally Manage VBA Macros

With Reflection Desktop 16.2 (and greater), you can use two different approaches to centrally control and manage Reflection Desktop VBA macros.

- You can save VBA macros in the `SharedMacros.rd3x`, `SharedMacros.rd5x`, or `SharedMacros.rdox` files and deploy them to the user data folder. When a session opens, it automatically loads the macros from the `sharedMacros` file that corresponds to it’s file type. (See “Walkthrough: Set up and deploy a SharedMacros file” on page 42)
- You can save VBA macros in session document files that reside on a network share or other location and then configure local session documents to reference these shared documents. When the local session documents open, they load the macros from the documents that they reference. (See “Walkthrough: Set up a Session to Reference Macros in a Session Document File” on page 39)

Reflection Desktop supports using SharedMacros files and referencing session documents that contain VBA macros for IBM (`.rd5x` and `.rd3x`) and VT (`.rdox`) session documents. You’ll need to create at least one session document file for each type of terminal you are using. Session document files can only reference macros in the same type of session document. For example, IBM3270 sessions can only reference session documents with an `.rd3x`extension.

NOTE: Reflection does not support nested references to session documents. If a session references a session document that in turn references other session documents, the references in the second session document are ignored.

If you are referencing session documents or using SharedMacros files to share macros, make sure the projects in those files have unique project names. Each project name in the VBA Project editor must be unique to avoid errors caused by naming conflicts. You can change project names by modifying the project properties in the VBA editor or by creating and saving these files in Reflection Desktop 16.2 or greater.

	Using SharedMacros files	**Referencing macros in shared session document files**
Use this approach when	You want to share macros with every session file of a specific terminal type (for example, all IBM3270 session files).	◆ You want to share macros only with specific session document files. ◆ You want to centrally manage macros by saving them in session document files that reside on a network share or other location. Saving the session file on a share allows you to change the macros on the user's machines without deploying any files. **NOTE:** When a local session with a reference to a session document file opens, it downloads the latest versions of the macros from the document that it references.
You'll need to	◆ Develop the macros you want to share and save them in one of the following session document files: SharedMacros.rd3x SharedMacros.rd5x SHaredMacros.rdox ◆ Deploy the `SharedMacros` file to the user data folder.	◆ Develop the macros you want to reference and save them in a session document file on a network share or another location. ◆ Set up custom session documents that reference the documents that contain the macros and deploy them to users. ◆ If the location you save the session documents on is not a trusted location, you'll need to add it to the list of trusted locations and deploy that setting to users.
Considerations	If you want to modify any of the macros in the `SharedMacros` files, you'll need to deploy these files to users.	To prevent users from editing the macro source in session documents referenced by other sessions, consider locking and password-protecting the macro projects or using some other method to control access to these files. **NOTE:** When you save a macro that was downloaded from a referenced session in a local session, the modified macro source is automatically uploaded to the referenced session document.

NOTE: Prior to Reflection Desktop version 16-2, Visual Basic code was shared by exporting and distributing code module (`.bas`), class (`.cls`), and form (`.frm`) files that could then be imported into user session document files. This method for sharing macros is still supported. However, for new implementations, it is recommended to share macros either by including them in the `SharedMacros` files or by referencing session documents that contain the shared macros.

Walkthrough: Set up a Session to Reference Macros in a Session Document File

This walkthrough shows how to set up a local session document file to reference macros in a session document file on a network share and how to add the share as a trusted location. It also shows how to create an MSI file that can be used to deploy the local sessions and their related settings to users.

NOTE: You can set up session document files to reference session document files that contain shared macros on any location that is a trusted location. These locations are not limited to network shares. For example, you could choose to reference macros in a session document file that resides on a folder on user workstations that is available to all users.

Set up and Test References to Macros

To set up and test your references to macros, first you'll need to develop the macros you want to reference and save them in a session document on the network share. Then you'll need to specify the network share as a trusted location. And then set up a local session document to reference this session document and configure the local session with buttons or menu items that run the macros.

Make sure the VBA projects in the session document files you are referencing have unique project names. Each project name in the VBA Project editor must be unique to avoid errors caused by naming conflicts. You can change project names by modifying the project properties in the VBA editor or by creating and saving these session document files in Reflection Desktop 16.2 or greater.

To develop and test a reference

1. Set up a network share to save the macros on.
2. Create a session document and develop the macros you want to reference. Make sure the macro projects in this document have unique names to prevent naming conflicts with VBA projects in any sessions that reference this session. Then save that session document file on the network share.

   NOTE: You'll need to create at least one session document file for each type of terminal you are using. Session document files can only reference macros in the same type of session document. For example, IBM3270 sessions can only reference session documents with an `.rd3x`extension.

3. Open the Reflection Workspace Settings window as follows:
   - If you're using the Ribbon interface, choose **Reflection Workspace Settings**from the **File** menu or the Reflection button (if using the Office 2007 Look and Feel).

- For an IBM session in the Classic interface, choose **Settings** from the **Options** menu to open the Settings window and then choose **Reflection Workspace Settings**.
- For a VT Session in the Classic interface, choose **View Settings** on the **Setup** menu to open the Settings window and then choose Reflection **Workspace Settings**.

4 In the Specify Trusted Locations dialog box, make sure the location you are saving the shared session documents on is on the list of trusted locations and add it if necessary. You must use a UNC path to specify this location. If you are using a network share, you'll also need to select **Allow trusted location on my network**.

5 Open each session you want to use these macros in and configure them as follows:

5a In the Settings window, select **Set Up VBA References**.

5b Under **Referenced Session Documents**, enter the fully qualified paths to the session documents that have the macros you want to reference.

NOTE: If you are saving the referenced session documents on a network share, you'll need to specify the location with a UNC path. Mapped letter drives are not supported.

5c If you want to set up a toolbar button or menu item to run a referenced macro, open the UI Designer on the Appearance tab and set the action for the button (or menu item) to run the macro.

NOTE: For more about setting up a button or menu item, see "Customize the Ribbon" or "Customize Classic Menus and Toolbars" in the Reflection Help.

6 Run and test the macros in your local session documents and modify them as needed. When you save your local sessions, any changes you made to the macros are automatically uploaded and saved in the document on the share.

7 To prevent users from modifying the macro source in the session document on the share, consider locking and password protecting the macros or using some other method to control access.

Deploy Sessions that Reference Macros and Related Settings

After you have developed and tested your macros, you'll need to deploy your local session files and the related settings files to users. This typically includes creating a companion MSI file or using another method to deploy the following files:

- *The session files*(the `.rd5x`, `.rdox`, or `. rd3x` files) that are configured with references to the files that contain the macros. (For this walkthrough, we'll deploy the `mySession.rd3x`file.)
- *A custom ribbon (`.xuml`)* file that includes any buttons or menu items you added to run the macros.
- *The `Application.Settings`*file that includes the trusted locations you added.

To create an MSI file for session documents that reference shared macros

1 Make sure you know where the session files and their related settings you want to add are located. For this example, we are packaging the files in the following locations:

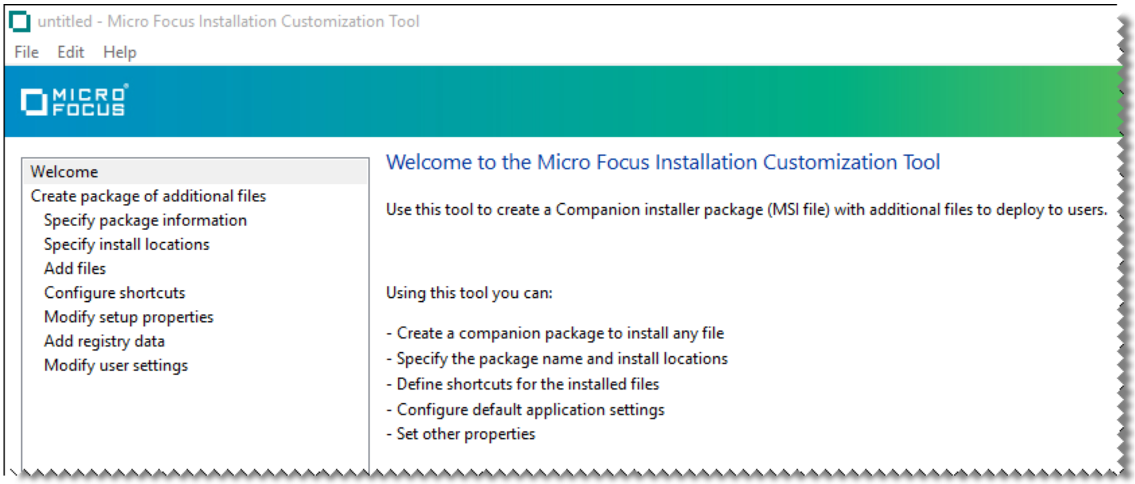
Add this file	In this directory
mySession.rd3x	`%USERPROFILE%\Documents\Micro Focus\Reflection`
myRibbon.xuml	`%USERPROFILE%\Documents\Micro Focus\Reflection\CustomUI`

2 On your administrative workstation, open the Installation Customization Tool from a desktop shortcut or from a command line as follows:

```
path_to_setup\setup.exe /admin
```

3 In the Select Customization dialog box, choose **Create a new Companion installer**.

The Installation Customization Tool opens in the mode used to create companion installer packages.



untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Welcome to the Micro Focus Installation Customization Tool

Use this tool to create a Companion installer package (MSI file) with additional files to deploy to users.

Using this tool you can:

- Create a companion package to install any file
- Specify the package name and install locations
- Define shortcuts for the installed files
- Configure default application settings
- Set other properties

4 From the Navigation pane, click **Specify package information**.

- In the **Add/Remove name** box, enter the name for the package that you want to appear in the Windows **Uninstall or change a program** list.
- In the **Organization name** box, enter the name of your department.

5 From the Navigation pane, click **Specify install locations**. Then, under **Installation type**, **select Installs only for the user who installs it**.

6 In the Navigation pane, select **Add Files**. Then, in the **Add files to** list, select `[PersonalFolder]` and type in:

```
\Micro Focus\Reflection\CustomUI
```

When you are done, the list entry is:

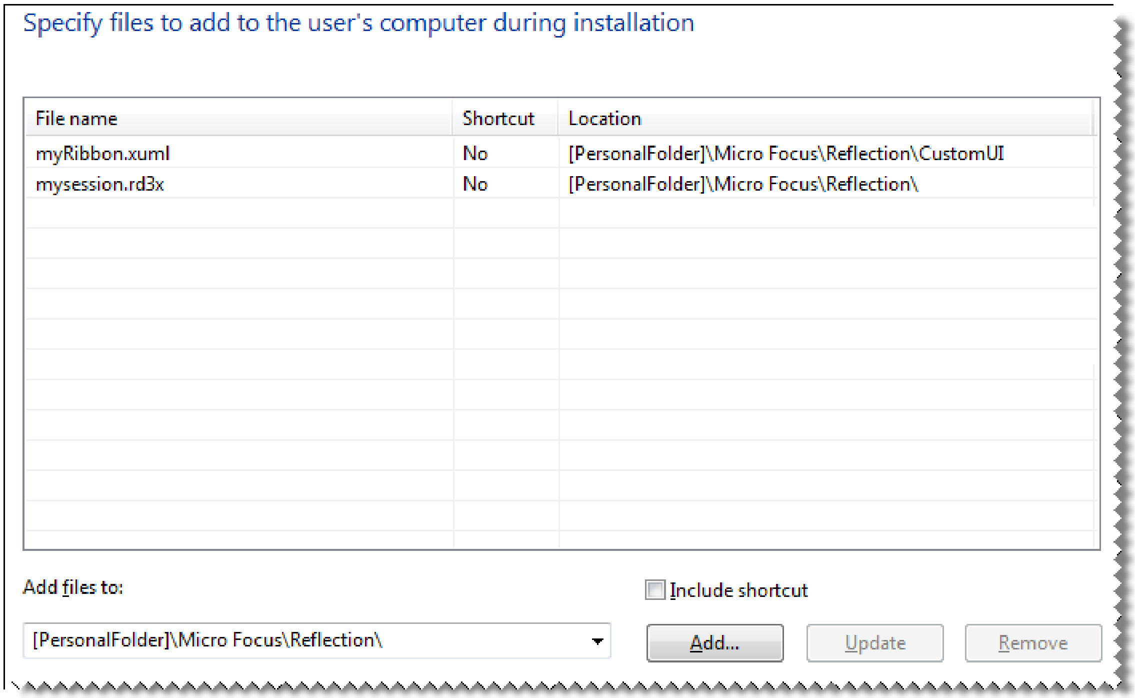
```
[PersonalFolder]\Micro Focus\Reflection\CustomUI
```

7 Click **Add**. Then browse to the configuration file (`myRibbon.xuml`) and click **Open**.

8 Repeat steps 6 and 7 to add the `mySession.rd3x` file to the `[PersonalFolder]\Micro Focus\Reflection` location.

NOTE: Be sure to specify the correct location for each file you add. Reflection looks for configuration files in specific locations. To find the correct file locations for any type of file see "Customized Files and Where to Deploy Them" on page 76.

When you are finished adding files, the panel should look similar to the following:



Specify files to add to the user's computer during installation

File name	Shortcut	Location
myRibbon.xuml	No	[PersonalFolder]\Micro Focus\Reflection\CustomUI
mysession.rd3x	No	[PersonalFolder]\Micro Focus\Reflection\

Add files to: [PersonalFolder]\Micro Focus\Reflection\

□ Include shortcut

9. On the Navigation pane, click Modify user settings. Then, in the **Make changes to user settings** panel, under **Application – Settings**, select **Reflection Desktop -Workspace Settings** and then click **Define**.
10. In the Reflection Workspace Settings dialog box, under **Workspace Settings**, select **Specify Trusted Locations**.
11. Add the path to the shared macros to the list of trusted locations and select **Allow trusted locations on my network**.

NOTE: Be sure to specify the path in the same way (either drive letter or UNC path) that you used in your session files.

12. Click **OK** to add the settings to the package and return to the main Installation Customization Tool window. The `Application.Settings`file is automatically added to the package in the appropriate location.
13. From the **File** menu, save the package as an `.msi` file on the administrative installation point.

Walkthrough: Set up and deploy a SharedMacros file

This walkthrough shows how to set up a `SharedMacros.rd3x` file and create an MSI file that can be used to deploy that file to users.

NOTE: Reflection Desktop supports shared VBA macros for IBM (`.rd5x` and `.rd3x)` and VT (`.rdox`) session documents.The `SharedMacros` settings files have the same extensions as their corresponding session document files:

IBM 5250: `SharedMacros.rd5x`

IBM 3250: `SharedMacros.rd3x`

VT: `SharedMacros.rdox`

Set up and Test a SharedMacros file

First, you'll need to set up and test your SharedMacros file. After you test the macros in this file, consider locking them and password protecting them so that users cannot change them.

Make sure the VBA projects in the SharedMacros files have unique project names. Each project name in the VBA Project editor must be unique to avoid errors caused by naming conflicts. You can change project names by modifying the project properties in the VBA editor or by creating and saving the SharedMacros files in Reflection Desktop 16.2 or greater.

To develop and test a SharedMacros file

1. Develop the macros you want to share.
2. Make sure the macro projects have unique names and save them in a session document named `SharedMacros.rd3x` in the user data directory.

   NOTE: The default user data directory is `%USERPROFILE%\Documents\Micro Focus\Reflection`

3. If you want to set up a toolbar button or menu item to run a shared macro, open the UI Designer on the Appearance tab and set the action for the button (or menu item) to run the macro.

   NOTE: For more about setting up a button or menu item, see "Customize the Ribbon" or "Customize Classic Menus and Toolbars" in the Reflection Help.

4. Run and test the macros in your session documents and modify the macros as needed.
5. To prevent users from modifying the shared macros, consider locking and password protecting them.

Deploy the SharedMacros.rd3x file

After you have developed and tested your macros, you'll need to deploy the `SharedMacros.rd3x` file to users, along with related settings, such as toolbar or ribbon buttons. This includes:

- *The SharedMacros files*(the `.rd5x`, `.rdox`, or `. rd3x` files) that contain the shared macros.
- *A custom ribbon (`.xuml`) file* that includes any buttons or menu items you added to run the macros.

To create an MSI file for a SharedMacros file

1. Verify the location of the user data folder and make sure you know where the session files and their related settings you want to add are located. For this example, we are packaging the files in the following locations:

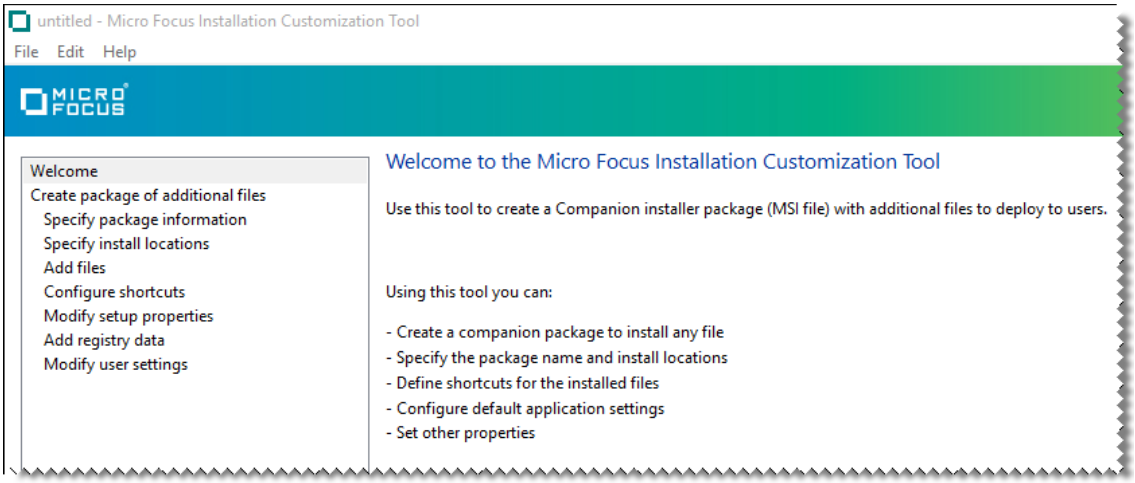
Add this file	In this directory
SharedMacros.rd3x	`%USERPROFILE%\Documents\Micro Focus\Reflection\`
myRibbon.xuml	`%USERPROFILE%\Documents\Micro Focus\Reflection\CustomUI`

2. On your administrative workstation, open the Installation Customization Tool from a desktop shortcut or from a command line as follows:

   ```
   path_to_setup\setup.exe /admin
   ```

3. In the Select Customization dialog box, choose **Create a new Companion installer**.

   The Installation Customization Tool opens in the mode used to create companion installer packages.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Welcome to the Micro Focus Installation Customization Tool

Use this tool to create a Companion installer package (MSI file) with additional files to deploy to users.

Using this tool you can:

- Create a companion package to install any file
- Specify the package name and install locations
- Define shortcuts for the installed files
- Configure default application settings
- Set other properties

4. From the Navigation pane, click **Specify package information**.
   - In the **Add/Remove name** box, enter the name for the package that you want to appear in the Windows **Uninstall or change a program** list.
   - In the **Organization name** box, enter the name of your department.
5. From the Navigation pane, click **Specify install locations**. Then, under Installation type, select **Installs only for the user who installs it**.
6. In the Navigation pane, select **Add Files**. Then, in the **Add files to** list, select `[PersonalFolder]` and type in:

   ```
   \Micro Focus\Reflection\CustomUI
   ```

   When you are done, the list entry is:

   ```
   [PersonalFolder]\Micro Focus\Reflection\CustomUI
   ```

7. Click **Add**. Then browse to the configuration file (`myRibbon.xuml`) and click **Open**.
8. In the Add file to list, select [PersonalFolder] and type in:

`\Micro Focus\Reflection\`

When you are done, the list entry is:

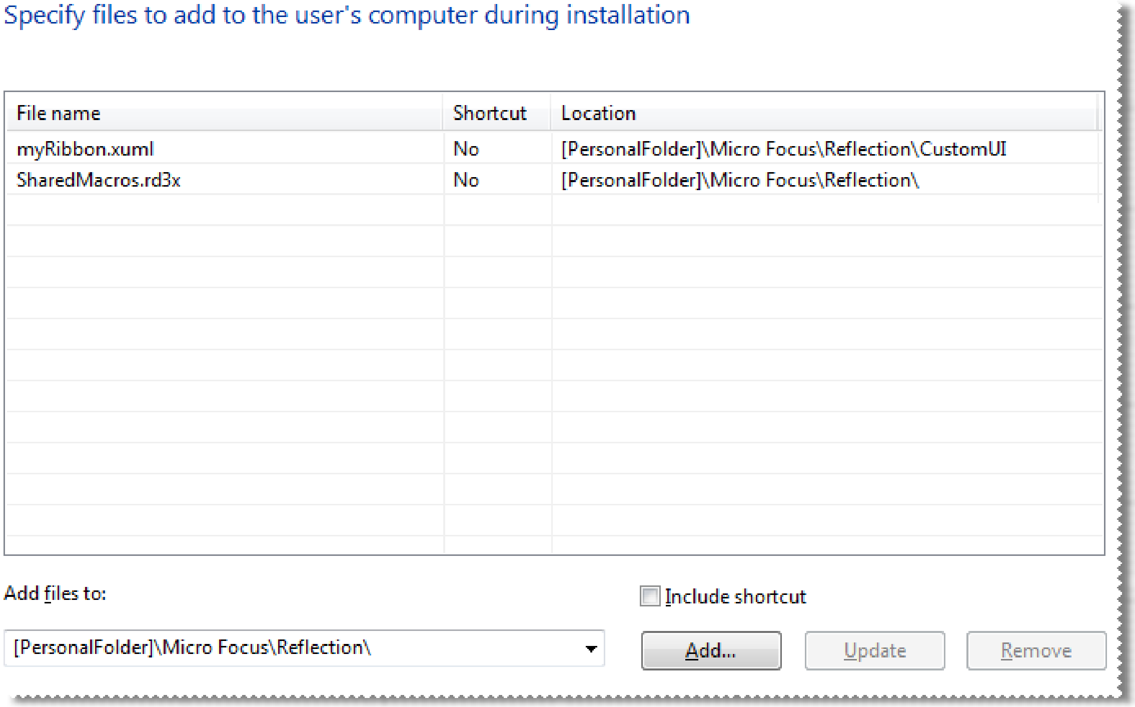
`[PersonalFolder]\Micro Focus\Reflection`

NOTE: Be sure to specify the correct location for each file you add. Reflection looks for configuration files in specific locations. To find the correct file locations for any type of file see "Customized Files and Where to Deploy Them" on page 76.

---

9 Click **Add**. Then browse to the `SharedMacros.rd3x`file.

When you are finished adding files, the panel should look similar to the following:



Specify files to add to the user's computer during installation

File name	Shortcut	Location
myRibbon.xuml	No	[PersonalFolder]\Micro Focus\Reflection\CustomUI
SharedMacros.rd3x	No	[PersonalFolder]\Micro Focus\Reflection\

Add files to: [PersonalFolder]\Micro Focus\Reflection\

10 From the **File** menu, save the package as an `.msi` file on the administrative installation point.

3 Protect Data and Information Privacy

Use the Trust Center to protect your working environment from information theft, and your data from potential damage caused by opening documents from non-trusted sources.

This article	Describes
Add Trusted Locations (page 48)	How to define locations from which you can safely open (and store) documents.
"Configure Information Privacy" on page 48	How to mask sensitive data (such as credit card numbers) with privacy filters.
"Configure API and Macro Security" on page 49	How to control access to the Reflection API and control the execution of actions invoked by a macro or API call.

Add Trusted Locations

A trusted location is a directory that is designated as a secure source for opening files. By default, Reflection Desktop allows users to open documents only in directories specified as trusted locations in the Reflection settings. Reflection Desktop specifies three trusted locations in the workspace `Application.settings` file in the program directory.

When you add other locations, these locations are saved in a custom `Application.settings` file in the user data directory folder. If you add trusted locations, you will need to deploy this file.

To set up a trusted location

1. From the Reflection File menu or Reflection button (if using the Office 2007 Look and Feel), select **Reflection Workspace Settings**.
2. Under Trust Center, click **Specify Trusted Locations**.
3. Click **Add new location** and then, under **Path**, browse to the location you want to add.

   NOTE: You can use Windows environment variables to define the trusted location.

4. To trust all folders within the trusted location, click **Subfolders**.
5. To package this file for deployment, see Package Configuration Files.

   NOTE:

   These settings are saved in the `Application.settings` file. You can deploy this file to one of the following locations:

   Location for a single user: [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

   Location for all users: [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

Configure Information Privacy

With Reflection Information Privacy, you can protect sensitive data such as credit card Primary Account Numbers (PANs), phone numbers, and US Social Security numbers. Information Privacy allows you to configure Reflection so that the sensitive data is not displayed on the screen or in productivity features, such as Screen History. It also allows you to require secure connections.

You can configure Information Privacy with the Installation Customization Tool or with Group Policy.

To set up Information Privacy in Reflection

1. From the Reflection File menu or the Reflection button ⊕ (if using the Office 2007 Look and Feel), select **Reflection Workspace Settings**.
2. Under Trust Center, click **Set Up Information Privacy**.
3. Configure privacy settings as shown in Setting Up Information Privacy.

If you need to...	Do this...
Redact certain patterns of data that are outside the realm of credit card formats (e.g., US Social Security numbers).	Set up Privacy Filter Redaction Rules and Privacy Filters.
Redact credit card Primary Account Numbers (PANs) to meet PCI DSS requirements. PCI DSS (Payment Card Industry Data Security Standard) is a worldwide standard comprising technology requirements and process requirements designed to prevent fraud and is published by PCI Security Standards Council, LLC (`https://www.pcisecuritystandards.org/`). All companies who handle credit cards are likely to be subject to this standard.	Set up Primary Account Number (PAN) Redaction Rules and Primary Account Number (PAN) Detection Rules.
Require secure connections (as may be required for PCI DSS compliance).	Set up PCI DSS Rules.

4. To package this file for deployment, see Package Configuration Files.

NOTE: Privacy filter settings are saved in the `PrivacyFilters.xml`file. All other Information for Privacy settings is saved in the `PCIDSS.settings` file. You can deploy these files to one of the following locations:

Location for a single user: [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

Location for all users: [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

To set up Information Privacy with Group Policy

1. Copy the following files to the central store as follows:

Copy these files	To

`ReflectionPCIDSS.admx and ReflectionWorkspace.admx in:`
`…\install_dir\Configuration\GroupPolicy\ADMX`
	`%systemroot%\PolicyDefinitions`

`ReflectionPCIDSS.adml and ReflectionWorkspace.adml in:`
`…\install_dir\Configuration\GroupPolicy\ADMX\en-us`
	%systemroot%\PolicyDefinitions\ <locale>

2. Open the Group Policy Object Editor (`gpedit.msc`).
3. Under either the Computer Configuration or User Configuration branch, browse to **Administrative Templates** | **Reflection Workspace** | **Information Privacy**.
4. In the **Information Privacy** panel, select and edit the policy settings.

NOTE: If you want to include the default regular expressions used for Custom Detection Rules and Custom Exception Expressions, you must add these expressions through the Group Policy editor. For detailed instructions, see Technical Note 2576: Adding Regular Expressions for Custom Detection Rules and Custom Exception Expressions to Group Policy (http://support.attachmate.com/techdocs/2576.html).

Configure API and Macro Security

You can enable the Reflection Desktop .NET API, and specify corresponding settings.

To set up API and macro and security

1 From the Reflection File menu or the Reflection button (if using the Office 2007 Look and Feel), select **Reflection Workspace Settings**.

2 Under **Trust Center**, click **Set Up API and Macro Security**.

3 Configure the API settings as follows:

To	Select
Prevent custom applications from accessing this installation.	**Disable .Net API**
Determine if Reflection legacy macros are supported, and to determine which legacy API has preference for the GetObject() method used to retrieve API COM objects. (Reflection supports multiple APIs, but can accept GetObject() calls for only one type of legacy API object at a time.)	**Legacy API preference**

4 Under **Action Permissions**, specify what you want to happen if an action that has been restricted through Group Policy or the Permissions Manager is initiated through a macro or API call.

To	Select
On a computer running Windows 10, Windows 8, or Windows 7, select to control restricted actions with User Account Control (UAC).	**Require elevated rights**
Select to run restricted actions that are initiated through a macro or API call as expected. The same actions won't run if they are initiated through the user interface.	**Execute the action**

5 To package this file for deployment, see Package Configuration Files.

NOTE: These settings are saved in the `Application.settings` file. You can deploy this file to one of the following locations:

Location for a single user: [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

Location for all users: [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

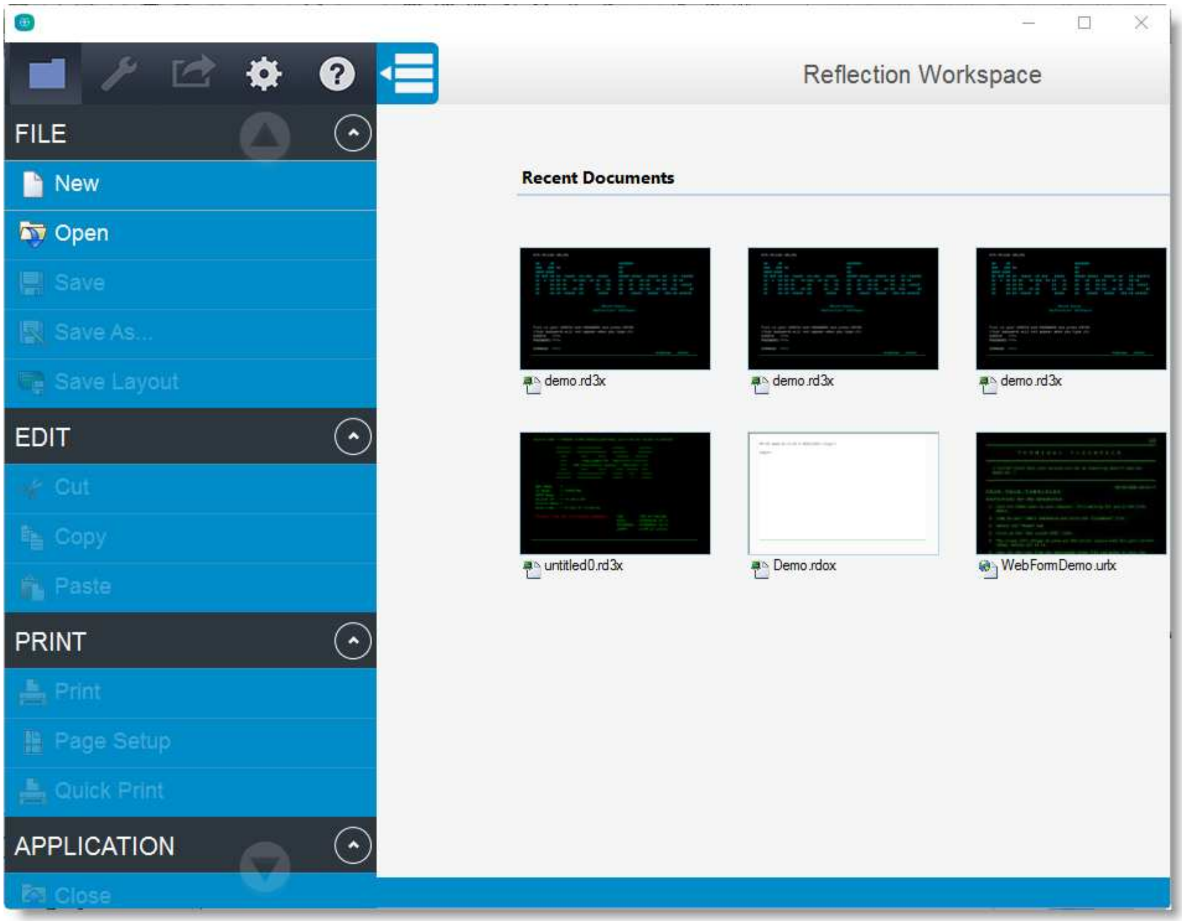
4 Customize the Reflection Workspace

You can customize the Reflection workspace to control its appearance and behavior or to lock down access to Reflection settings and controls.

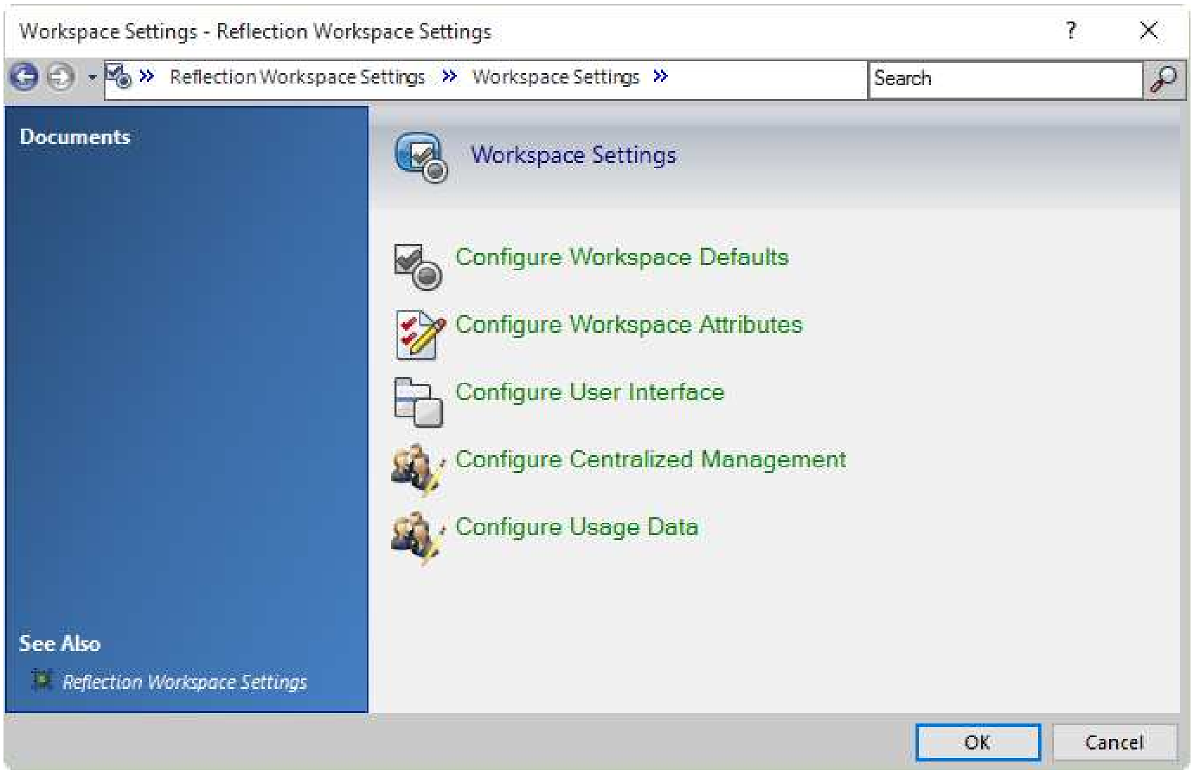
This article	Describes
Configure Workspace Behavior and Appearance	How to configure Reflection to change its appearance and functionality.
Walkthrough: Customize Reflection Appearance and Behavior	An example that shows how to use the Installation Customization Tool to configure Reflection so that multiple sessions are displayed in separate windows instead of in a single window with tabs.
Control Access to Settings and Controls	How access to settings and controls on the Reflection interface can be restricted so that administrative access is required to use them.
Control Access to Settings and Controls with Reflection Administrative Tools	How to control access using the Permissions Manager administrative tool.
Walkthrough: Restrict Access to Settings and Controls	An example that shows how to use the Permissions Manager administrative tool to restrict access to the Auto Complete control.
Control Access to Settings and Controls with Microsoft Group Policy	How to install the Microsoft Group Policy templates provided for Reflection and use them to restrict access.

Configure Workspace Behavior and Appearance

You can change the basic functionality of the main Reflection window in a variety of ways. For example, you can specify startup macros and actions. You can also change the appearance of the window (for example, by setting Reflection to open in the TouchUx interface mode as shown below).



Workspace settings include all of the settings you can access from the Workspace Settings dialog box.


Workspace Settings - Reflection Workspace Settings
?
Reflection Workspace Settings
Workspace Settings
Search
Documents
Workspace Settings
Configure Workspace Defaults
Configure Workspace Attributes
Configure User Interface
Configure Centralized Management
Configure Usage Data
See Also
Reflection Workspace Settings
OK
Cancel

You can customize workspace settings in several different ways:

- "Customize Workspace Settings Directly With Reflection" on page 53 to configure the settings in Reflection. Then save the settings in the application.access file. It's easier to verify your settings with this approach but you'll need to package the setting as shown in "Package Sessions and Custom Settings Files" on page 73.
- "Configure and Automatically Package Workspace Settings" on page 54, using the Installation Customization Tool to configure the settings. When you use this approach, the settings are automatically packaged in an MSI file.
- "Update Only Specific Workspace Settings" on page 57 to change a specific workspace setting without changing the other settings on a user desktop.

Customize Workspace Settings Directly With Reflection

To customize workspace settings from Reflection

1. Open the Workspace Settings dialog box.

   The steps depend on your user interface.

User Interface Mode	Steps
Ribbon	On the **File** menu or the Reflection button (if using the Office 2007 Look and Feel), choose **Reflection Workspace Settings**.
Reflection Browser	On the Reflection menu, choose **Settings** and then **Reflection Workspace Settings**.
Mobile UI	Tap the Gear icon and then select **Reflection Workspace Settings**.

2. Under Workspace Settings, select the type of workspace setting you want to configure:

Select this option	To
Configure Workspace Defaults	Configure the actions to perform when the Reflection workspace opens or closes and preferences for automatically saving session document files.
Configure Workspace Attributes	Configure options for logging, running remote sessions, and displaying Help. You can also specify the user data directory, in which session documents and other related files are saved.
Configure User Interface	Configure which type of user interface to use (Reflection provides four interfaces), its look and feel, and other user interface options.
Configure Centralized Management	Set up the workspace to access sessions that are centrally managed on a Micro Focus Management and Security Server.
Configure Usage Data	Choose whether to participate in the Product Experience Improvement program.

3. Change the settings as needed and then save them.
4. Close and reopen Reflection and then verify the settings.
5. To package this file for deployment, see Package Configuration Files.

NOTE: These settings are saved in the `Application.settings` file. You can deploy this file to one of the following locations:

Location for a single user: [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

Location for all users: [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

To find out more about configuring workspace settings, see Set up Workspace Settings in the Micro Focus Reflection Desktop Help.

Configure and Automatically Package Workspace Settings

When you use the Installation Customization Tool to configure workspace settings, the custom configuration files that include the settings are automatically added to an MSI.

CAUTION: Do not use the Installation Customization tool to configure Trust Center settings. This approach should be used only to configure settings in the Configure Workspace Defaults, Configure Workspace Attributes, Configure User Interface, Configure Centralized management, or Configure Usage Data dialog boxes.

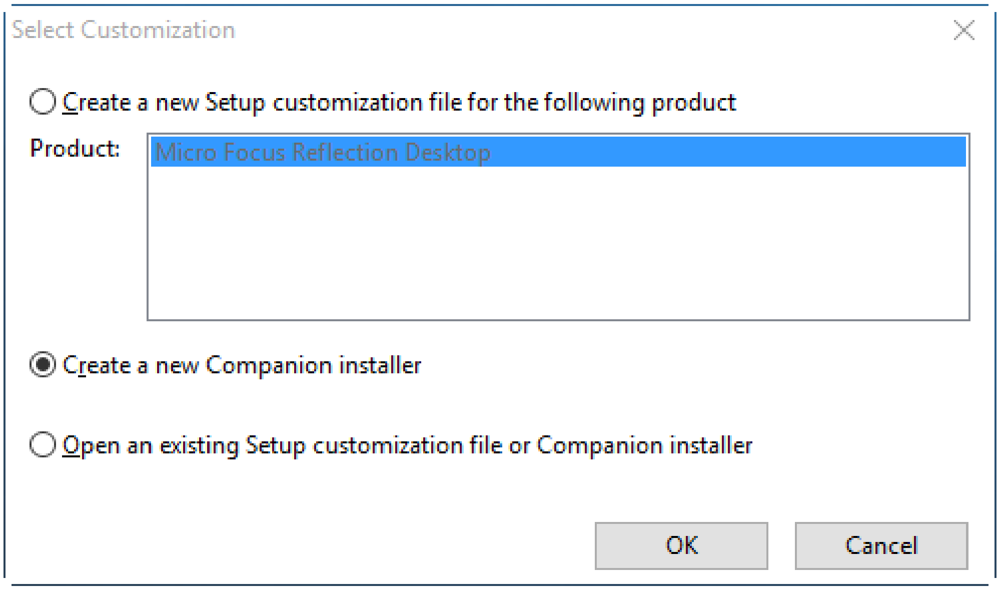
You'll probably want to configure these settings on your workstation first to make sure you get the results you want. Then reconfigure these settings as shown below.

To create a custom workspace with the Installation Customization Tool

1. From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2. From the **Select Customization** dialog box, select **Create a new Companion installer**, and then click **OK**.



Select Customization

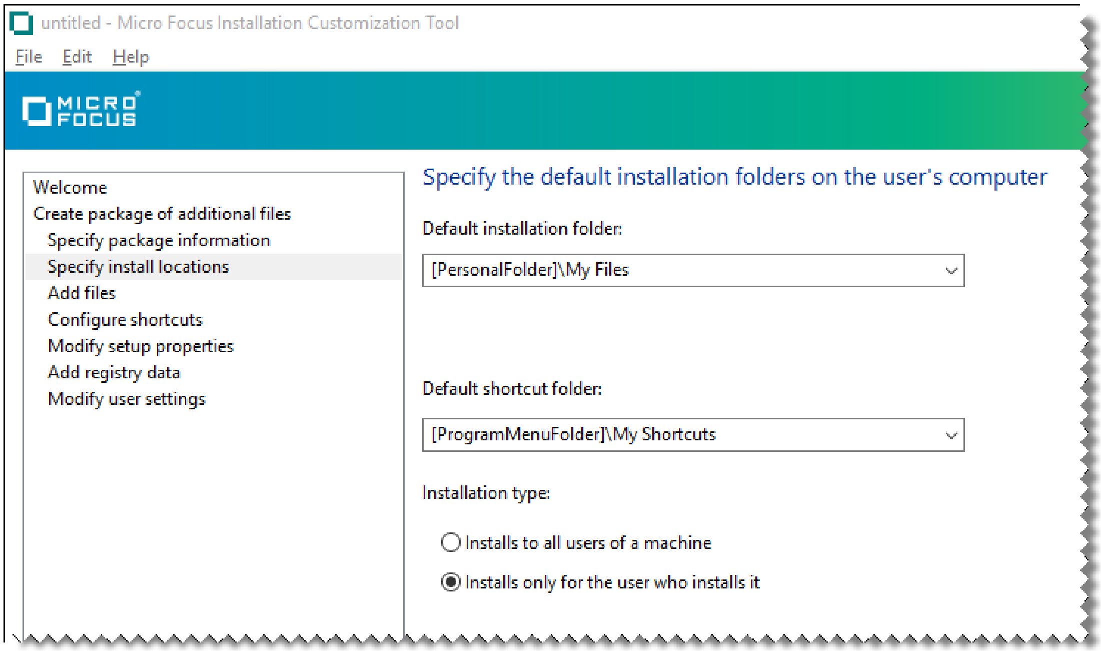
○ Create a new Setup customization file for the following product

Product: Micro Focus Reflection Desktop

● Create a new Companion installer

○ Open an existing Setup customization file or Companion installer

3. From the navigation pane, click **Specify install locations** and choose whether to install the workspace settings for all users of a machine or only for specific users.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Specify the default installation folders on the user's computer

Default installation folder:

[PersonalFolder]\My Files

Default shortcut folder:

[ProgramMenuFolder]\My Shortcuts

Installation type:

○ Installs to all users of a machine

● Installs only for the user who installs it

4. On the navigation pane, click **Modify user settings**.
5. From the list of **Application - Settings**, select **Reflection Desktop - Workspace Settings** and then click **Define**.

   The **Reflection Workspace Settings**dialog box opens in a separate window.
6. Under **Workspace Settings**, click the type of setting to configure (for example, **Configure Workspace Defaults**).
7. Configure the settings just as you would if you opened the workspace settings dialog box from the product.
8. Save the package (`.msi`) file and close the Installation Customization Tool. You can deploy the package file as it is, or you can edit it to add additional files.

NOTE: These settings are automatically saved in the `Application.settings` file. The package `.msi` file is automatically configured to deploy this file to one of the following locations, depending on which option you selected when you specified install locations.

- If you selected **Installs only for the user who installs it**, the file is deployed to:

  [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
- If you selected **Installs to all users of a machine**, the file is deployed to:

  [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

Update Only Specific Workspace Settings

You can change a specific workspace setting without changing the other settings on a user desktop. This allows you to change a setting after you deploy Reflection without the risk of overwriting other settings that users have configured.

To deploy a specific setting, you'll use the Installation Customization Tool (ICT) to create and package an Update.settings file in an MSI that you can deploy to users. When Reflection starts, it applies the settings in this file to the workspace.

If you want to modify a setting that you have updated in this way or change it back to the default setting, you'll use the ICT to modify the Update.settings file in the MSI you deployed and then deploy the new version of the MSI to users.

NOTE: You can update only workspace settings that are set in the Application.Settings file. You cannot update Information Privacy or Access file settings.

To create and package a workspace settings update

1. On a workstation on which you have installed Reflection, open the Installation Customization Tool from a desktop shortcut.
2. From the Select Customization dialog box, select **Create a new Companion installer**.
3. From the navigation pane, click **Specify install locations**.
4. Under **Installation type**, select how to install the configuration file (the `Update.settings` file) that specifies your workspace setting.
   - **Installs to all users of a machine** installs the file in a common folder so that the settings apply to all users of the machine.
   - **Installs only for the user who installs it**installs the file in a personal folder so that the settings apply only to a single user.
5. On the navigation pane, click **Modify user settings**.
6. In the Make changes to user settings... panel, under **Application – Settings**, select **Reflection Desktop - Update Workspace Settings** and then click **Define**.
7. Configure the settings you want to change and then close the Workspace Settings dialog box.
8. Save the companion file and close the Installation Customization Tool.

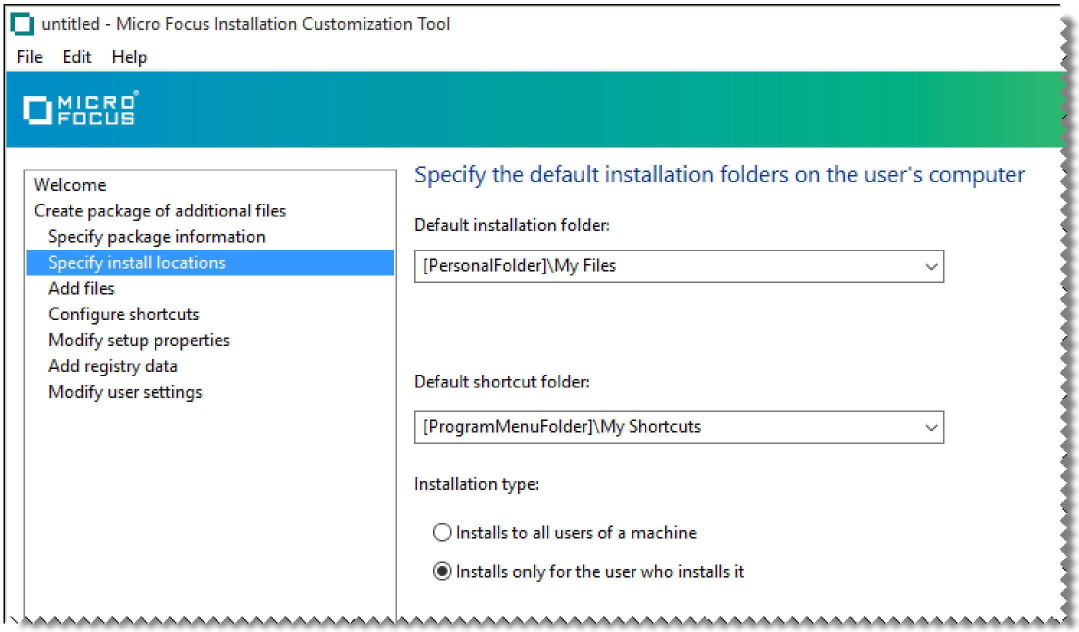
NOTE: When you deploy the new version of the MSI to users, the new Update.settings file is installed with the setting you modified.

Walkthrough: Customize Reflection Appearance and Behavior

You can change the appearance and basic functionality of the main Reflection window in a variety of ways. You can also specify startup macros, startup actions, whether to open Reflection with the Ribbon open or closed, and other options. You will use the Installation Customization Tool (ICT) to configure most of the custom workspace settings. The following example shows how to configure a workspace so that multiple sessions are displayed in separate windows instead of in a single window with tabs (the default).

To create a custom workspace

1. On a workstation on which you have installed Reflection, open the Installation Customization Tool from a desktop shortcut.
2. From the Select Customization dialog box, select **Create a new Companion installer**.
3. From the navigation pane, click **Specify install locations**.
4. Under **Installation type**, select how to install the configuration file (the `Application.settings` file) that specifies your workspace settings.
   - **Installs to all users of a machine** installs the file in a common folder so that the settings apply to all users of the machine.
   - **Installs only for the user who installs it**installs the file in a personal folder so that the settings apply only to a single user.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Specify the default installation folders on the user's computer

Default installation folder: [PersonalFolder]\My Files

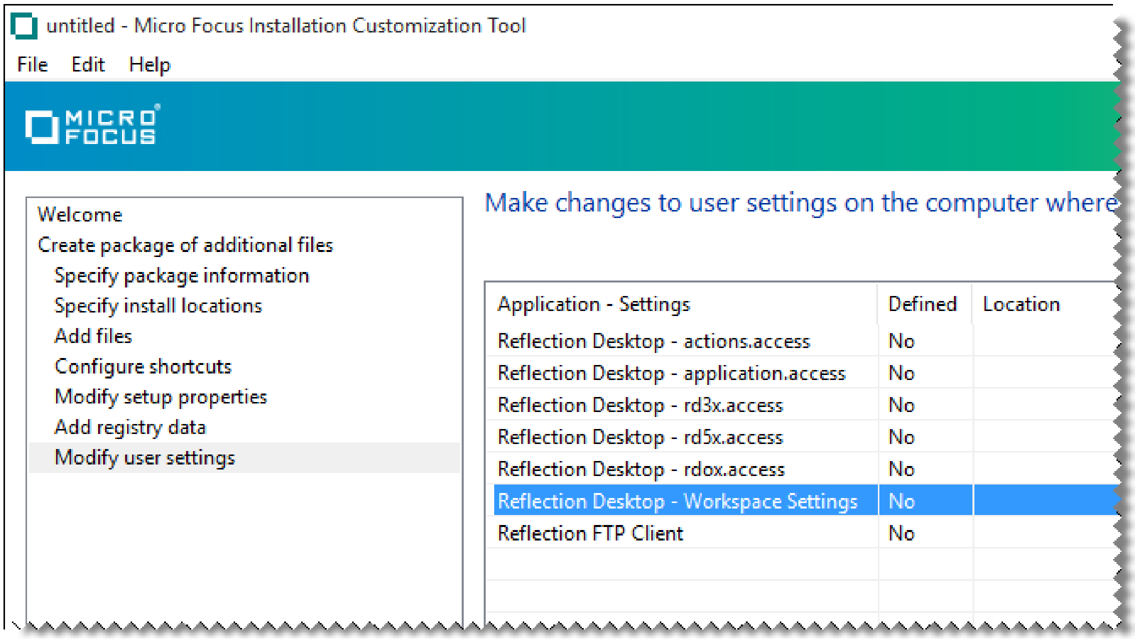
Default shortcut folder: [ProgramMenuFolder]\My Shortcuts

Installation type:

○ Installs to all users of a machine

● Installs only for the user who installs it

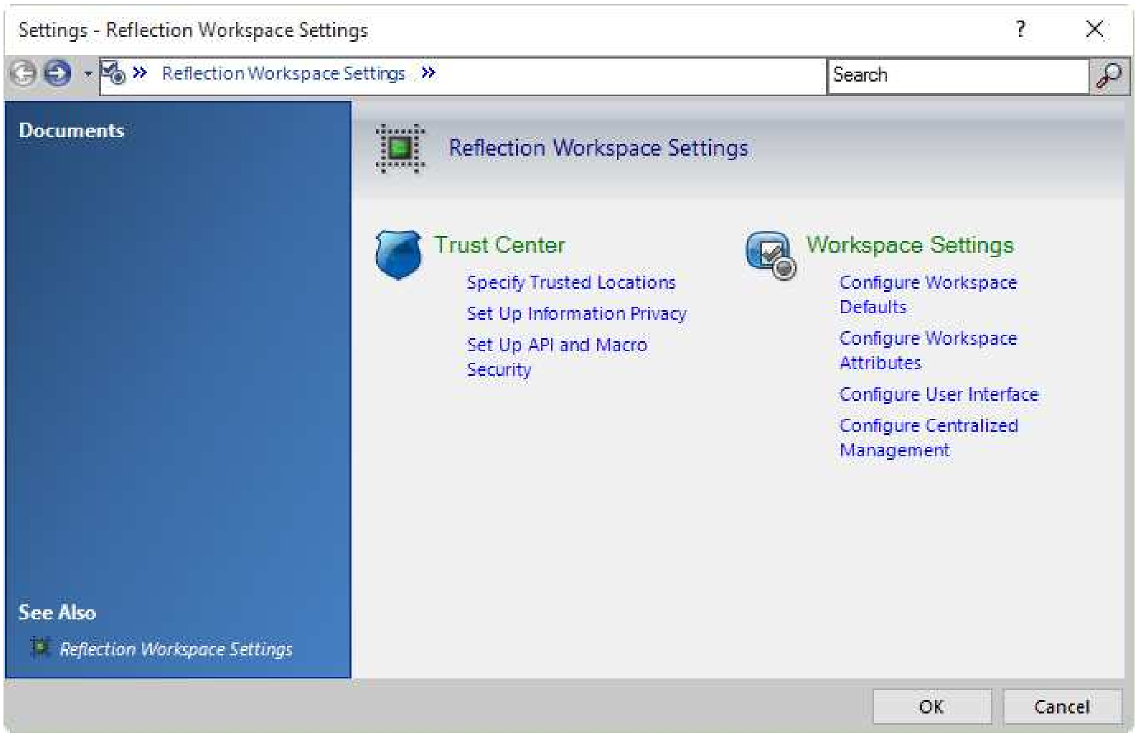
5. On the navigation pane, click **Modify user settings**.
6. In the Make changes to user settings... panel, under **Application – Settings**, select **Reflection Desktop -Workspace Settings** and then click **Define**.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

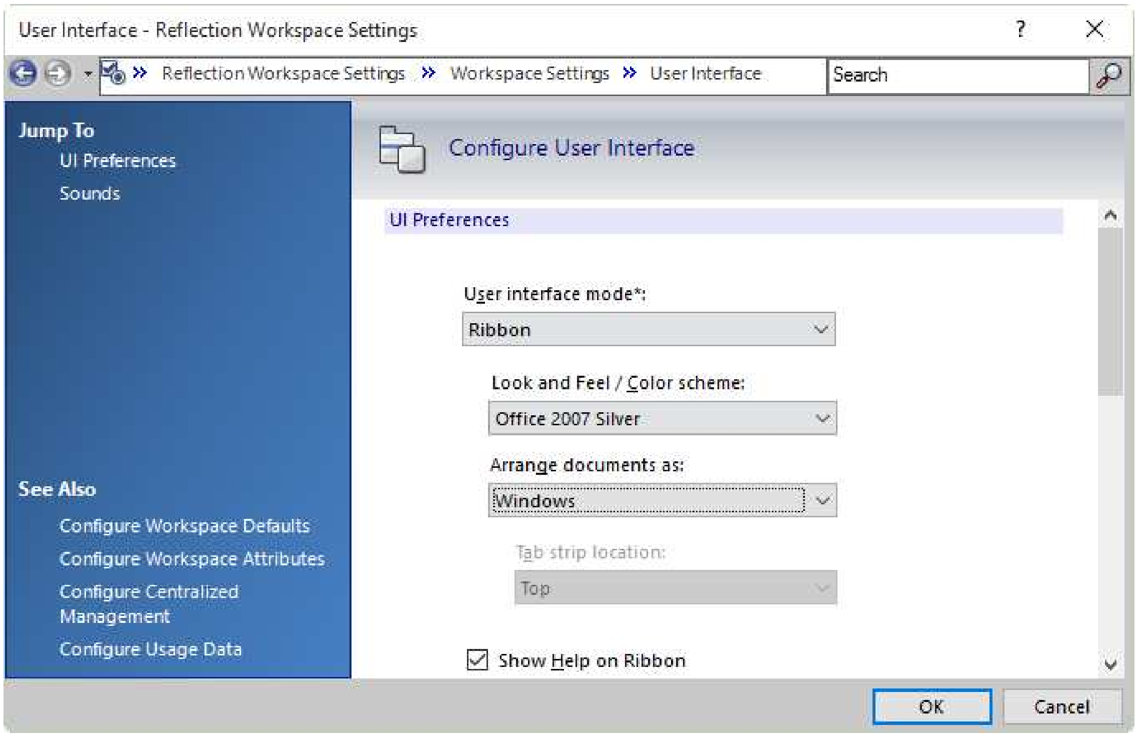
Make changes to user settings on the computer where

Application - Settings	Defined	Location
Reflection Desktop - actions.access	No	
Reflection Desktop - application.access	No	
Reflection Desktop - rd3x.access	No	
Reflection Desktop - rd5x.access	No	
Reflection Desktop - rdox.access	No	
Reflection Desktop - Workspace Settings	No	
Reflection FTP Client	No	

7 In the Reflection Workspace Settings dialog box, under **Workspace Settings**, click **Configure User Interface**.


Settings - Reflection Workspace Settings
?
Reflection Workspace Settings
Search
Documents
Reflection Workspace Settings
Trust Center
Specify Trusted Locations
Set Up Information Privacy
Set Up API and Macro Security
Workspace Settings
Configure Workspace Defaults
Configure Workspace Attributes
Configure User Interface
Configure Centralized Management
See Also
Reflection Workspace Settings
OK
Cancel

8 In the Configure Workspace Settings dialog box, under **UI Preferences**, in the **Arrange documents as** list, select **Windows**.


User Interface - Reflection Workspace Settings
?
Reflection Workspace Settings » Workspace Settings » User Interface
Search
Jump To
UI Preferences
Sounds
See Also
Configure Workspace Defaults
Configure Workspace Attributes
Configure Centralized Management
Configure Usage Data
Configure User Interface
UI Preferences
OK
Cancel

User interface mode*: Ribbon

Look and Feel / Color scheme: Office 2007 Silver

Arrange documents as: Windows

Tab strip location: Top

☑ Show Help on Ribbon

9 Save the companion file and close the Installation Customization Tool.

NOTE: These settings are automatically saved in the `Application.settings` file. The package `.msi` file is automatically configured to deploy this file to one of the following locations, depending on which option you selected when you specified install locations.

- If you selected **Installs only for the user who installs it**, the file is deployed to:

  [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

- If you selected **Installs to all users of a machine**, the file is deployed to:

  [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

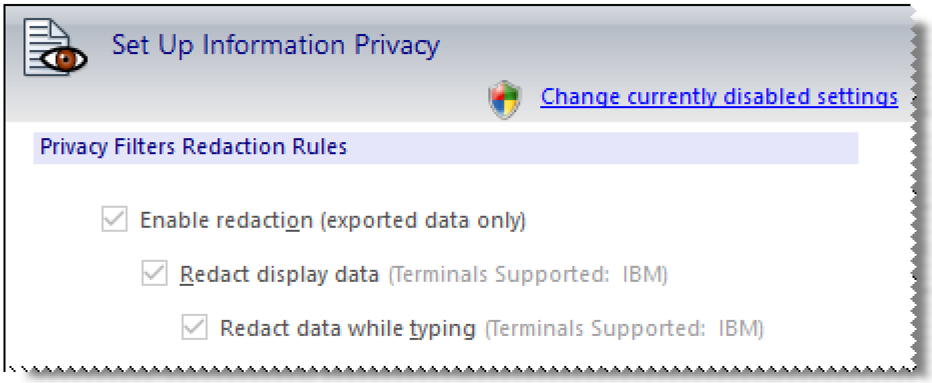
Control Access to Settings and Controls

You can restrict access to almost any of the Reflection settings or controls. For example, you can prevent users from changing the host address that a session connects to, or from running a macro. This allows you to simplify support requirements and resolve security concerns.

NOTE: You can restrict access to Reflection controls only on systems on which the Windows User Access Control (UAC) is enabled. If the UAC is disabled, the configuration settings that restrict access do not apply.

Restricting Access to Settings

As shown below, the restricted settings are grayed out. The security shield and the **Change currently disabled settings** link indicate administrative access is required to change them. Users cannot change these options unless they elevate their access level to administrator.



Set Up Information Privacy

Change currently disabled settings

Privacy Filters Redaction Rules

☑ Enable redaction (exported data only)

☑ Redact display data (Terminals Supported: IBM)

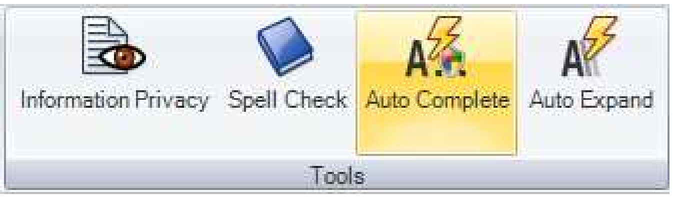
☑ Redact data while typing (Terminals Supported: IBM)

The security shield and the Change currently disabled settings link indicate that settings are disabled and administrative access is required to enable them.

NOTE: The customized files that include these settings can be bundled in a companion installer package (`.msi`) and deployed with the installation or later.

Restricting Access to Controls

You can prevent the user from using a particular control by restricting access to the control. In the following example, the security shield over the Auto Complete icon indicates that the "action" associated with this option is "restricted".



Tools for Restricting Access

You can restrict access to Reflection controls and settings with the Permissions Manager tool or through Microsoft Windows Group Policy settings.

- "Control Access to Settings and Controls with Reflection Administrative Tools" on page 62. The Permissions Manager tool is a Reflection administrative tool that is automatically installed on your administrative installation point when you create an administrative install image. It can be accessed through the Installation Customization Tool or independently. With this tool, you can create and deploy .access files that restrict access to controls or settings.
- "Control Access to Settings and Controls with Microsoft Group Policy" on page 68. Microsoft Group Policy provides another method for restricting access. If you are using Group Policy, you can import the Reflection ADM or ADMX files into your environment and modify the settings in the Group Policy Editor.

Control Access to Settings and Controls with Reflection Administrative Tools

To prevent a user from changing a setting, you set the permission level for that setting or control to “Restricted.” When a setting is restricted, administrative access is required to change the setting. For example, you could restrict the user’s ability to modify security settings.

NOTE: You can restrict access to Reflection controls only on systems on which the Windows User Access Control (UAC) is enabled. If the UAC is disabled, the configuration settings that restrict access do not apply.

The following access file templates are distributed with Reflection Desktop:

This File	Controls access to...
`actions.access`	Actions (for example, Auto Complete)
`application.access`	Reflection workspace settings
`rd3x.access`	3270 terminal settings
`rd5x.access`	5250 terminal settings
`rdox.access`	VT terminal settings
`security.access`	TLS and PKI settings

Individual permissions are merged in the following order (from highest to lowest):

- Group Policy – user
- Group Policy – machine
- Local permissions file (.access)

Deploying local permissions (.access) files

Use the Reflection Permissions Manager tool to set local permissions and save them in `.access` files that you can deploy.

You can deploy user-specific access settings for any type of `.access` files. To deploy user-specific files, install the `.access` files to [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

You can install some types of access configuration (`actions.access, application.access`, and `security.access` files) for all users of the system. To deploy these settings for all users, install these files in [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

NOTE: Settings files in the [CommonAppDataFolder] location are copied to the [AppDataFolder] location when the user opens the Workspace.

You can set permissions and create `.access`files by using the Permissions Manager with or without the Installation Customization Tool. When you use Permissions Manager with this tool, the tool automatically determines the correct location in which to install the required files. When you open Permissions manager and create `.access` files outside of this tool, you'll need to be sure the files are installed in the correct directory.

Specify Access Using Permissions Manager with the Installation Customization Tool

You can open Permissions Manager from the Installation Customization Tool to lock down access. When you use this approach, the resulting `.access`files are automatically added to the correct directory in the package (MSI file).

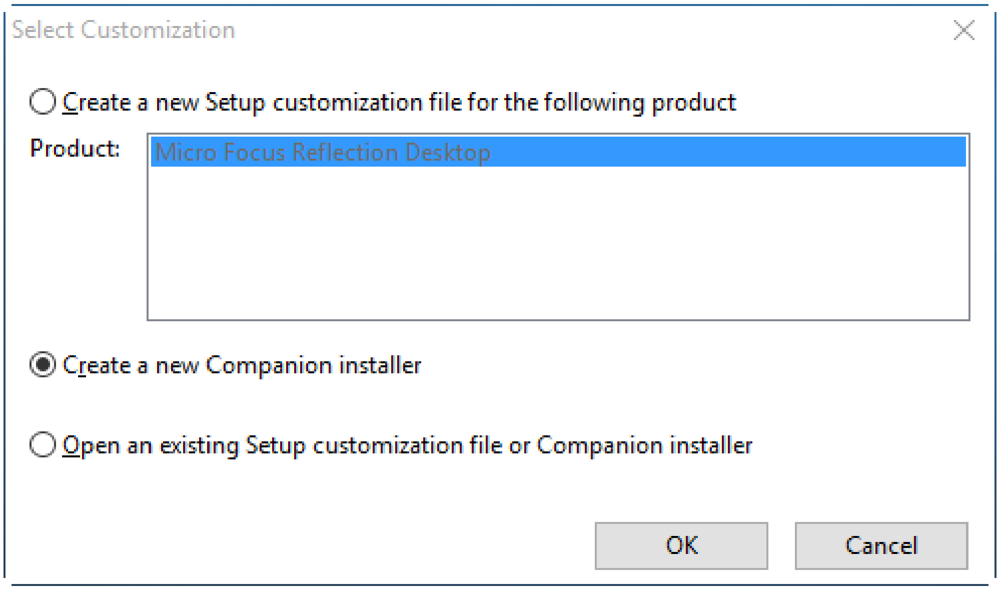
NOTE: These files are not saved to your local machine. They are saved only in your MSI database. To make changes to these files, you will need to use the same approach to edit them as you used to create them. You'll need to open the MSI file in the Installation Customization Tool and then open Permissions Manager from the tool.

To set user and group access with the Installation Customization Tool

1 From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

<path_to_setup> `\setup.exe /admin`

2 In the **Select Customization** dialog box, select **Create a new Companion installer**.


Select Customization

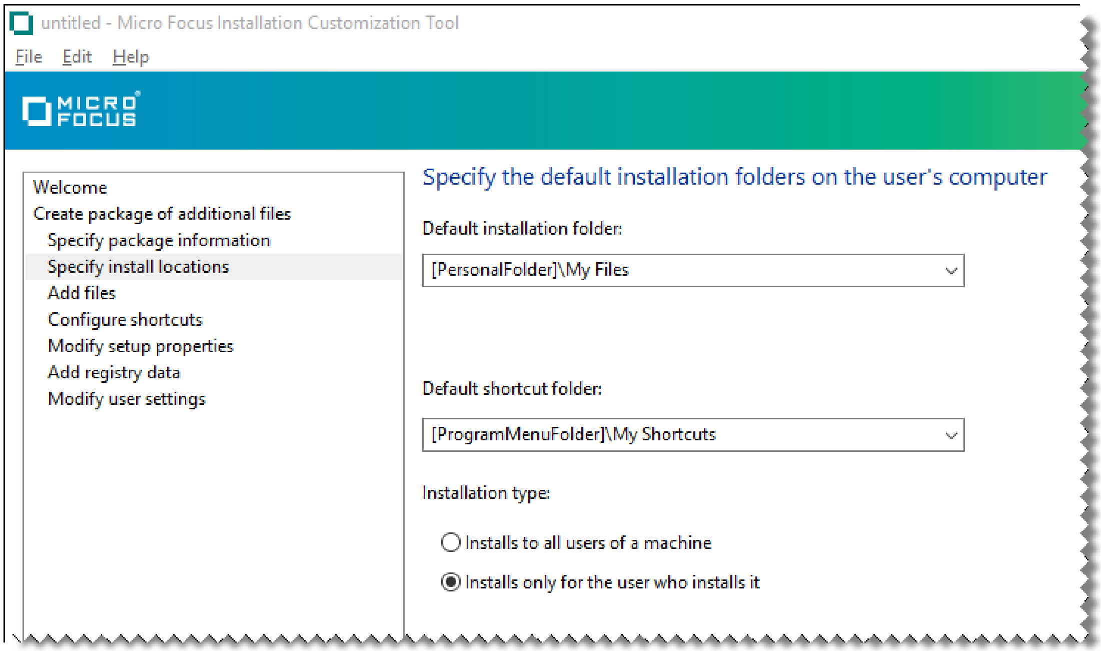
○ Create a new Setup customization file for the following product

Product: Micro Focus Reflection Desktop

● Create a new Companion installer

○ Open an existing Setup customization file or Companion installer

3 On the left pane, select **Specify install locations**.


untitled - Micro Focus Installation Customization Tool
File Edit Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Specify the default installation folders on the user's computer

Default installation folder:

[PersonalFolder]\My Files

Default shortcut folder:

[ProgramMenuFolder]\My Shortcuts

Installation type:

○ Installs to all users of a machine

● Installs only for the user who installs it

4 Under **Installation type**, select whether to install the settings to all users of a machine or only for the user who installs it.

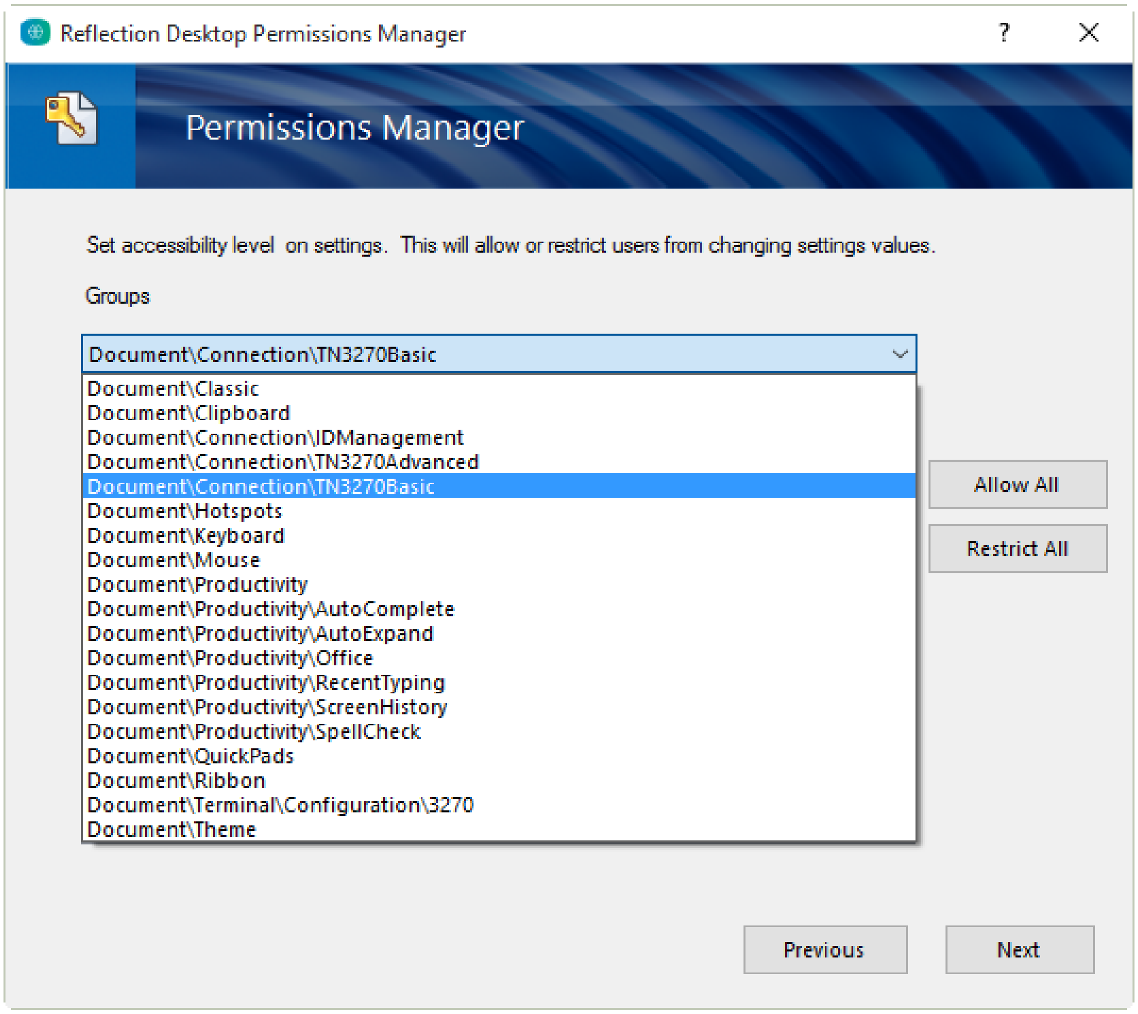
NOTE: Only `actions.access`, `application.access`, and `security.access` files can be deployed to all users.

5 In the left pane, select **Modify user settings**.

NOTE: Under **Application - Settings**, the Permissions Manager displays groups of configurable items. These items are listed by their internal names, which may not exactly match the user interface item. The item's **Accessibility**indicates whether the user can configure the item (**Full**) or if administrator assistance is required to configure the item (**Restricted**).

6 In the **Make changes to user settings** pane, select one of the .access options and click **Define**.

7 In Permissions Manager, under **Groups**, select the group of settings you want to control access to (for example, `Document\Connection\TN3270Basic`).


Reflection Desktop Permissions Manager
?
Permissions Manager
Document\Classic
Document\Clipboard
Document\Connection\IDManagement
Document\Connection\TN3270Advanced
Document\Connection\TN3270Basic
Document\Hotspots
Document\Keyboard
Document\Mouse
Document\Productivity
Document\Productivity\AutoComplete
Document\Productivity\AutoExpand
Document\Productivity\Office
Document\Productivity\RecentTyping
Document\Productivity\ScreenHistory
Document\Productivity\SpellCheck
Document\QuickPads
Document\Ribbon
Document\Terminal\Configuration\3270
Document\Theme
Allow All
Restrict All
Previous
Next

Set accessibility level on settings. This will allow or restrict users from changing settings values.

Groups: Document\Connection\TN3270Basic

8 In the Itemsbox, in the Accessibility column for the item (or items) you want to restrict, click Full and then select Restricted from the drop down menu.

NOTE: The Accessibility drop down menu includes three items:

- **Full**: All users can configure the item.
- **Restricted**: Only administrators of the system can configure the item. These items have the Windows access shield added to their icons.
- **Read-only**: No users of the system can configure the item. These items are grayed out.

9 Under Additional security options, select how to control session file encryption:

To do this	Select
Configure all sessions so that users can open only encrypted display session files.	User can open only encrypted session files
Configure all sessions so that users can save a display session only if it is encrypted.	User can save only encrypted session files

10 From the Filemenu, choose Save As and save the companion installer package.

If you selected **Installs only for the user who installs it** when you specified install locations, the companion installer package automatically specifies to deploy this `.access` file to [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

If you selected **Installs to all users of a machine**, it specifies to deploy the `.access` file to [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

NOTE

- Make sure to set file access rights on `.access` files to prevent users from deleting, replacing, or editing them.
- To deploy files to the [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 folder, you will need to use a deployment tool that allows you to install the companion installer package as the user.
- When accessing a setting via an API, such as executing a macro, a setting with restricted access cannot be modified. (When attempting to set a restricted setting via an API, an error is logged.)
- Setting session encryption options in an *.access file affects only the associated session type. For example, limiting users to opening only encrypted session files in rd3x.access affects only 3270 terminal session files, and not 5250 session files.

Specify Access Using Permissions Manager

To prevent a user from changing a setting, you set the permission level for that setting or control to “Restricted.” When a setting is restricted, administrative access is required to change the setting. For example, you could restrict the user’s ability to modify security settings.

You can lock down access by running Permissions Manager (without using the Installation Customization Tool) to edit `.access` files. If you use this approach, be sure to deploy the customized `.access` files to the correct directory.

NOTE: Important: Be sure to set file access rights on `.access` files that you deploy to prevent users from deleting, replacing, or editing them.

To set access with Permissions Manager

1. On a workstation to which you have installed Reflection, log on as administrator and in the Reflection Desktop install folder, run `AccessConfig.exe`.
2. When prompted to create a new permission file, or edit an existing one, choose **Create new permission file**.
3. When prompted with a list of access file templates, choose the type of permission file to create.
4. Under **Groups**, select the type of setting to control access to (for example, the **Document\Connection\TN3270Basic** group).
5. In the **Items**box, in the **Accessibility**field for the item (or items) you want to restrict, click **Full**and then select **Restricted** from the drop down menu.
6. If you are configuring `rd3x.access`, `rd5x.access`, or `rdox.access` files, under **Additional security options**, select how to control session file encryption:

To do this	Select
Configure all sessions so that users can open only encrypted display session files.	**User can open only encrypted session files**
Configure all sessions so that users can save a display session only if it is encrypted.	**User can save only encrypted session files**

7 Be sure to deploy the `.access` files to the correct directory:

To deploy settings that are user-specific, deploy the `.access` files to [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

To deploy settings for all users of a machine, deploy the `.access` files to [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2.

NOTE: Only `actions.access`, `application.access`, and `security.access` files can be deployed to all users.

IMPORTANT

- To deploy files to the [AppDataFolder] folder, your deployment tool must allow you to install the companion installer package as the user.
- Setting session encryption options in an `.access` file affects only the associated session type. For example, limiting users to opening only encrypted session files in `rd3x.access` affects only 3270 terminal session files, and not 5250 session files..
- When accessing a setting via an API, such as executing a macro, a setting with restricted access cannot be modified. (When attempting to set a restricted setting via an API, an error is logged.)

Walkthrough: Restrict Access to Settings and Controls

The following example shows how to use the Permissions Manager tool to restrict access to the Auto Complete control.

To use Permissions Manager to restrict access to Auto Complete

1 On a workstation on which you have installed Reflection, open the Installation Customization Tool from a desktop shortcut or from a command line as follows:

```
path_to_setup\setup.exe /admin
```

2 In the Select Customization dialog box, choose to either create a new companion installer or edit an existing one.

The Installation Customization Tool opens in the mode used to create or edit companion installer packages (MSI files). This mode is also used to open the Permissions Manager tool that is used to restrict access.

3 From the navigation pane, click **Specify install locations**.

4 Under **Installation type**, select **Installs only for the user who installs it**.

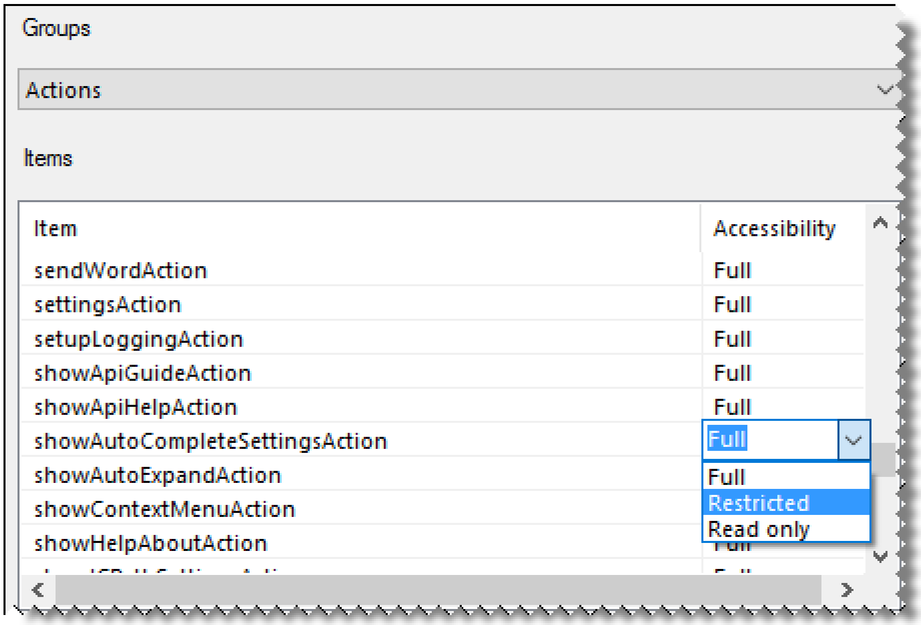
5 From the navigation pane, click **Modify user settings**.

6 Under **Application – Settings, select Reflection Desktop – actions.access** and then click **Define**.

NOTE: The access file templates in the **Application-Settings** list are grouped by function. Features such as Auto Complete are in the **actions.access** template. Workspace settings are in the **application.access** template. Session Settings are in the `rd3x.access`, `rd5x.access`, and `rdox.access` terminal templates.

The Permissions Manager Tool opens in a separate window. This tool displays all of the controls (actions) and settings that you can restrict.

7 In the items list, scroll down to **showAutoCompleteSettingsAction**. Then, under **Accessibility**, on the drop-down menu, choose **Restricted**.



Groups

Actions

Items

Item	Accessibility
sendWordAction	Full
settingsAction	Full
setupLoggingAction	Full
showApiGuideAction	Full
showApiHelpAction	Full
showAutoCompleteSettingsAction	Full
showAutoExpandAction	
showContextMenuAction	
showHelpAboutAction	Full

8 Click **Next**, accept the default values, and then click **Finish**.

The settings are automatically saved in the `actions.access` file and the companion installer package is automatically configured to deploy this file to [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

9 From the ICT File menu, choose **Save As** and save the companion installer package file on the administrative installation point.

Control Access to Settings and Controls with Microsoft Group Policy

As an administrator, you can limit users' ability to modify their workspace or session documents by setting permissions from the Microsoft Group Policy Management Console using group policy templates.

NOTE: You can restrict access to Reflection controls only on systems on which the Windows User Access Control (UAC) is enabled. If the UAC is disabled, the configuration settings that restrict access do not apply.

Reflection installs a set of group policy templates (ADM and ADMX files) to the following directory:

```
\Program Files\Micro Focus\Reflection\Configuration\GroupPolicy
```

NOTE: To use this feature, you must be running Windows 7, Windows 8.1, or later on an administrative machine. For more information about managing group policy, see *Managing Group Policy ADMX Files Step-by-Step Guide* (http://technet.microsoft.com/en-us/library/cc709647(v=ws.10).aspx).

ADMX files

ADMX files are divided into language-neutral files (.admx) and language-specific resource files (.adml), available to all Group Policy administrators. These factors allow Group Policy tools to adjust their UI according to the administrator's configured language.

Reflection `setup.exe` installs ADMX files to:

```
... \ install_dir \Configuration\GroupPolicy\ADMX
```

It installs ADML files to the following directory:

```
...\
    install_dir \Configuration\GroupPolicy\ADMX\en-us
```

Reflection provides the following ADMX Group Policy files. Each of these files has a corresponding ADML language file.

This file	Controls access to
`ACTIONS.admx`	Actions
`APPLICATION.admx`	Reflection Workspace
`RD3X.admx`	3270 terminal settings
`RD5X.admx`	5250 terminal settings
`RDOX.admx`	VT terminal settings
`ReflectionWorkspace.admx`	Root-level ADMX file
`ReflectionSecurity.admx`	TLS and PKI settings

NOTE: This directory also includes the `ReflectionPCIDSS.admx` file. This file is used to configure information privacy through Group Policy and is not used to control access.

ADM files

ADM files contain the Group Policy definitions and resource strings in the same file.

Reflection `setup.exe` installs ADM files to:

```
...
\
 install_dir
\Configuration\GroupPolicy\ADM\
```

ADM Group Policy files:

This file	Controls access to
`ACTIONS.adm`	Actions
`APPLICATION.adm`	Reflection Workspace
`RD3X.adm`	3270 terminal settings
`RD5X.adm`	5250 terminal settings
`RDOX.adm`	VT terminal settings

NOTE: This directory also includes the `ReflectionPCIDSS.adm` file. This file is used to configure information privacy through Group Policy and is not used to control access.

Install Group Policy Templates

Before you deploy group policy definitions, set and test them on a local test machine.

To deploy ADMX & ADML files on a local test machine

1. Copy the `.admx` files from …\ *`install_dir`* `\Configuration\GroupPolicy\ADMX` to the central store (`%systemroot%\PolicyDefinitions`)
2. Copy all required locale `.adml` files to: `%systemroot%\PolicyDefinitions\<locale>`
3. Open the Group Policy Object Editor (`gpedit.msc`)
4. Under either **Computer Configuration** or **User Configuration**, browse to **Administrative Templates** | **Reflection Desktop** or **Administrative Templates** | **Reflection Security**.
5. In the Group Policy Management Editor, navigate to the setting or feature you want to configure.
6. Enable the Group Policy settings you want to restrict access to.

   NOTE: For more about using ADMX files to set group policy, see *Managing Group Policy ADMX Files Step-by-Step Guide* (http://technet.microsoft.com/en-us/library/cc709647(v=ws.10).aspx)
   .

To install ADM files on a local test machine

1. Copy all .adm files

   From:

   `...\ ` *`install_dir`* `\Configuration\GroupPolicy\ADM\`

   to:

`C:\Windows\inf`

2 Open Group Policy Object Editor (`gpedit.msc`)

3 Under either **User Configuration** or **Computer Configuration**, Right-click on **Administrative Templates** and select **Add/Remove Templates**.

4 Click **Add**, select the Reflection ADM files you need to add, and then click **Open**.

The Reflection ADM files are listed in the Add/Remove Templates dialog box, in the **Current Policy Templates** list.

5 Under either **Computer Configuration**or **User Configuration**, browse to **Administrative Templates** | **Classic Administrative Templates (ADM)** | **Reflection Desktop**.

6 In the Group Policy Management Editor, navigate to the setting or feature you want to configure.

7 Enable the Group Policy settings to which you want to restrict access.

NOTE: Registry keys are added when policy settings are **Enabled**. When **Not Configured**, no key is present. When a setting is **Disabled**, the key is still present, and it's data is set to 0x00000000. The data is 0x00000004 when enabled.

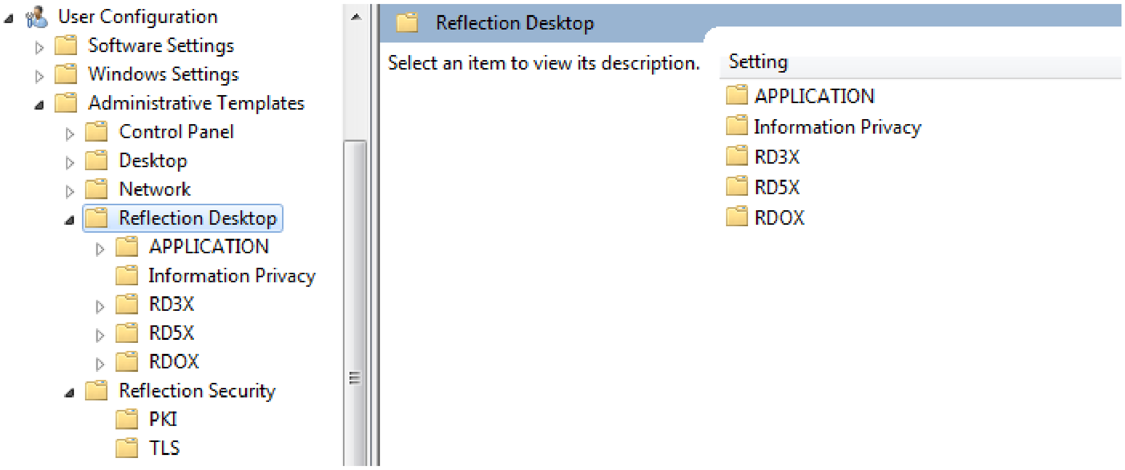
For more about using ADM files to set group policy, see *Add or remove an Administrative Template (.adm file)* (http://technet.microsoft.com/en-us/library/cc739134.aspx).

Set Access with Group Policy

To set access with Group Policy Object Editor

1 In the Group Policy Management Editor, navigate to the setting or feature you want to configure.

The following example shows all shipping ADMX files loaded into the GPO Editor under User Configuration. Group Policies can be set at the machine (Computer Configuration) or user (User Configuration) levels.



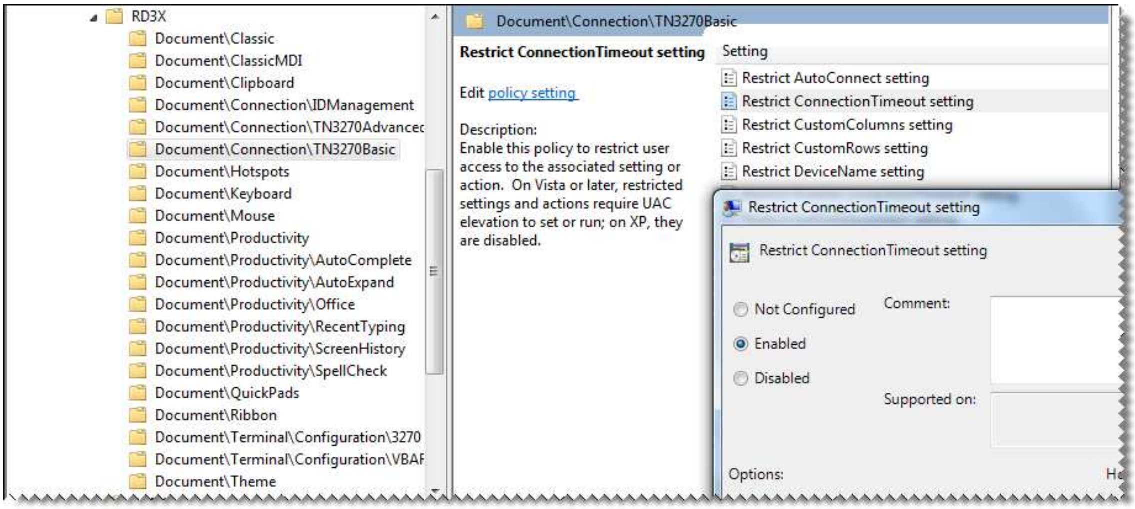
Select an item to view its description.

2 Enable the Group Policy settings you want to use.

The following example shows the following:

- The current node is the RD3X Document\Connection\TN3270Basic group.

- All the settings for this group are listed in the right-hand panel.
- The Restrict ConnectionTimeout setting policy is **Enabled**. This setting for 3270 display sessions is restricted.


RD3X
Document\Classic
Document\ClassicMDI
Document\Clipboard
Document\Connection\IDManagement
Document\Connection\TN3270Advancec
Document\Connection\TN3270Basic
Document\Hotspots
Document\Keyboard
Document\Mouse
Document\Productivity
Document\Productivity\AutoComplete
Document\Productivity\AutoExpand
Document\Productivity\Office
Document\Productivity\RecentTyping
Document\Productivity\ScreenHistory
Document\Productivity\SpellCheck
Document\QuickPads
Document\Ribbon
Document\Terminal\Configuration\3270
Document\Terminal\Configuration\VBAF
Document\Theme
Document\Connection\TN3270Basic
Setting
Restrict AutoConnect setting
Restrict ConnectionTimeout setting
Restrict CustomColumns setting
Restrict CustomRows setting
Restrict DeviceName setting
Restrict ConnectionTimeout setting
Comment:
Supported on:
Options:
He

Restrict ConnectionTimeout setting

Edit policy setting

Description:
Enable this policy to restrict user access to the associated setting or action. On Vista or later, restricted settings and actions require UAC elevation to set or run; on XP, they are disabled.

Restrict ConnectionTimeout setting

○ Not Configured
● Enabled
○ Disabled

Registry keys are added when policy settings are **Enabled**. These keys remain in the registry when policy settings are **Disabled**. No key is present when policy settings are **Not Configured**.

5 Package Configuration Files

Create a companion installer package (also called a "companion database") to install any custom files you created when you customized Reflection. A companion installer package is a standalone MSI file that is independent of any Reflection installer package.

You can deploy companion installer packages separately or you can add them to a customized installation, so that the companion installer packages are automatically installed when the Reflection installer package completes.

You can also create and install packages at any time after the initial installation. Because companion installer packages are installed independently of Reflection, you can upgrade the product without removing these support files and you can deploy additional support files without re-installing the product.

If you support several business units that require their own customized configuration files, you can create a companion installer package for each business unit.

Companion installer packages have no built-in user interface except a standard progress bar. They are displayed as independent entries in the Windows **Add\Remove Programs** list and can be installed or uninstalled independently of Reflection.

If you use centralized management, you can also use "Use Central Management to Deploy MSI Packages" on page 103.

This article	Describes
Package Sessions and Custom Settings Files	How to bundle configuration files into Microsoft MSI files that you can deploy using standard Windows deployment tools.
Walkthrough: Create a Package with the Installation Customization Tool	An example that shows how to create and deploy a companion installer package to install a keyboard map file and a session document file.
Customized Files and Where to Deploy Them	Directory locations for custom configuration files that you create when you customize Reflection sessions or other settings. Reflection cannot find these files unless they are in these specific locations.

Package Sessions and Custom Settings Files

After you customize Reflection, bundle the customized session or settings files into a deployable companion installer package (.msi).

To create a companion installer package

1 Note the locations of custom configuration files you've created.

2 From your administrative installation point, open the Installation Customization Tool from a shortcut or by typing the following command line:

```
<path_to_setup>\setup.exe /admin
```

3 In the **Select Customization** dialog box, open the companion installer view:

- If you created a companion installer package earlier, select **Open an existing Setup file or Companion installer**, click **OK**, and then browse to select the file.
- If you have not created a companion installer package, select **Create a new Companion installer**.

4 From the navigation pane, select **Specify package information**.

5 In the **Add/Remove name** box and the **Organization name** box, type a name for the installation and the publisher that you want to be displayed in the Microsoft Windows **Uninstall or change a program** (**Add or remove programs**) list.

6 From the navigation pane, click **Specify install locations**, and specify the **Installation type**. Select whether to install the files for all users or for only one user:

- **Installs to all users of a machine** makes files available for every user who logs onto the computer. Use this option for settings files, macros, and other configuration files that you want to be available to all users.
- **Installs only for the user who installs it** makes files available only for the user who installs them.

7 In the **Default installation folder**list, select the folder in which to install the files. (Files are deployed to this folder unless you specify another folder when you add a file.)

NOTE: The folders available in this list depend on which Installation type you chose. Installation Type folder options specify the access for the files (after the installation). These options affect only the folders to which you can install — you cannot change these options after you add a file.

8 In the **Default shortcut folder** list, select the folder in which to install program shortcuts. (Shortcuts are deployed to this folder unless you specify another folder when you add a file.)

9 From the navigation pane, select **Add files** and then click **Add** and browse to the files you want to include.

10 Specify the destination directory for each file as follows:

10a In the table of files that you added, select the file.

10b In the **Add files to** list (at the bottom of the panel), enter the destination directory. (You can choose a location from the list or edit the path by typing. Use the "Customized Files and Where to Deploy Them" on page 76 table as a guide for where to install the custom files you have created.)

NOTE: The folder location you add must already exist on all target computers.

10c Click **Update**. (The destination directory for the file is displayed in the table's Location column.)

11 If you want to create a shortcut for the file, select **Include shortcut** and then click **Update**.

12 From the **File**menu, save the `.msi` file on the administrative installation point.

Walkthrough: Create a Package with the Installation Customization Tool

The following example shows how to create and deploy a companion installer package to install a keyboard map file and a session document file.

To create a companion installer package MSI file that includes configuration files

1 Make sure you know where the files you want to add are located. For this example, we are packaging the files in the following locations:

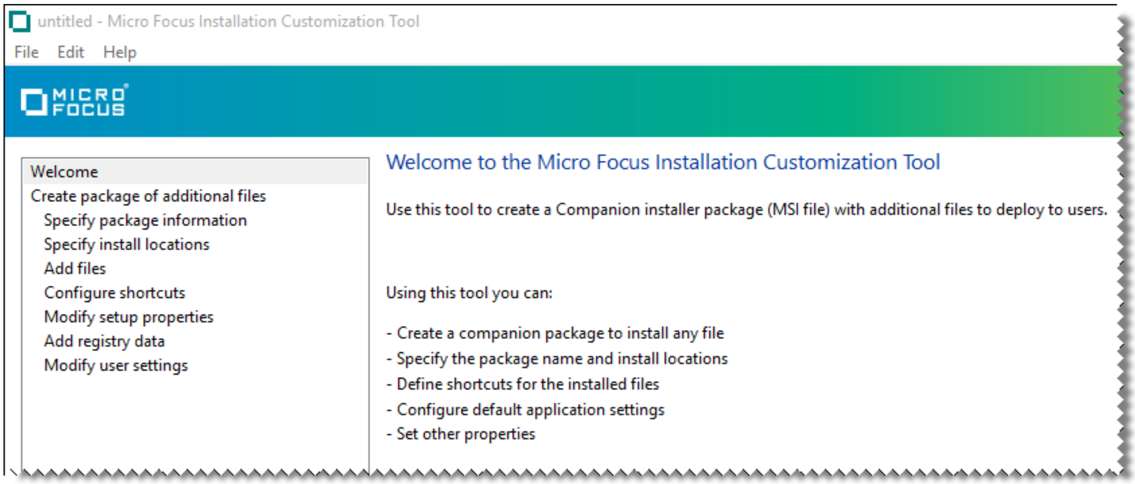
Add this file	In this directory
mySession.rd3x	`C:\Users\yourUserName\Documents\Micro Focus\Reflection`
myKeyboardMap.xkb	`C:\Users\yourUserName\Documents\Micro Focus\Reflection\Keyboard Maps`

2 On your administrative workstation, open the Installation Customization Tool from a desktop shortcut or from a command line as follows:

```
path_to_setup\setup.exe /admin
```

3 In the Select Customization dialog box, choose **Create a new Companion installer**.

The Installation Customization Tool opens in the mode used to create companion installer packages.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Create package of additional files
Specify package information
Specify install locations
Add files
Configure shortcuts
Modify setup properties
Add registry data
Modify user settings

Welcome to the Micro Focus Installation Customization Tool

Use this tool to create a Companion installer package (MSI file) with additional files to deploy to users.

Using this tool you can:

- Create a companion package to install any file
- Specify the package name and install locations
- Define shortcuts for the installed files
- Configure default application settings
- Set other properties

4 From the ICT navigation pane, click Specify install locations. Then, under Installation type, **select Installs only for the user who installs it**.

5 In the Navigation pane, select **Add Files**.

6 in the **Add files to** list, select `[PersonalFolder]`. Then type in:

```
\Micro Focus\Reflection\Keyboard Maps
```

When you are done, the list entry is:

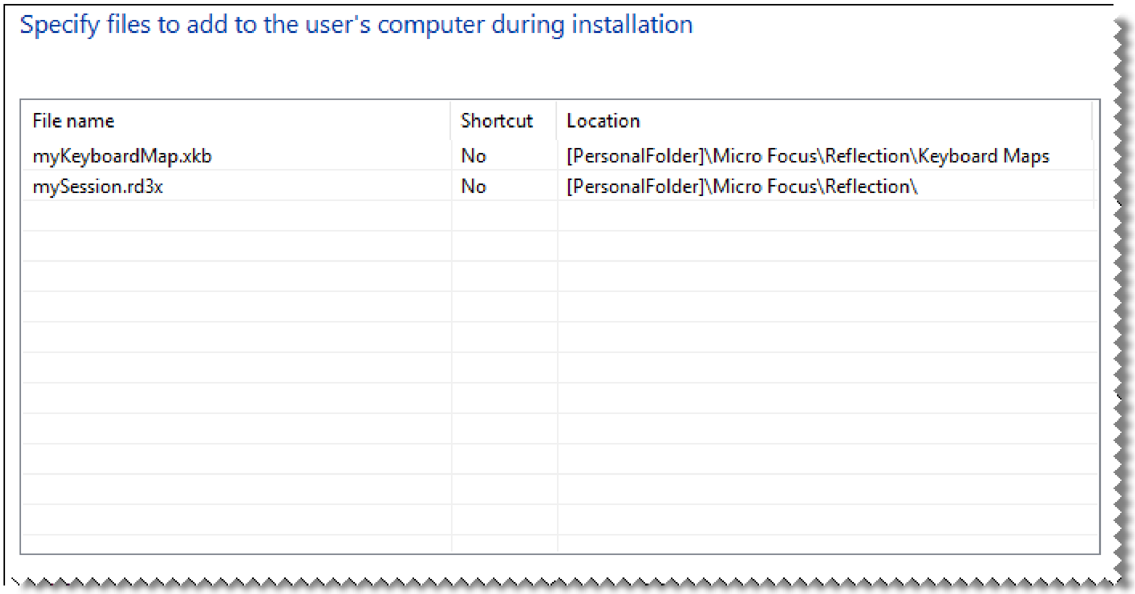
```
[PersonalFolder]\Micro Focus\Reflection\Keyboard Maps
```

7 Click **Add**. Then browse to the configuration file (`myKeyboardMap.xkb`) and click **Open**.

8 Repeat steps 6 and 7 to add the `mySession.rd3x` file to the `[PersonalFolder]\Micro Focus\Reflection` location.

NOTE: Be sure to specify the correct location for each file you add. Reflection looks for configuration files in specific locations. To find the correct file locations for any type of file see "Customized Files and Where to Deploy Them" on page 76.

When you are finished adding files, the panel should look similar to the following:



Specify files to add to the user's computer during installation

File name	Shortcut	Location
myKeyboardMap.xkb	No	[PersonalFolder]\Micro Focus\Reflection\Keyboard Maps
mySession.rd3x	No	[PersonalFolder]\Micro Focus\Reflection\

9 From the navigation pane, click **Specify package information**.

- In the **Add/Remove name** box, enter the name for the package that you want to appear in the Windows **Uninstall or change a program** list.
- In the **Organization name** box, enter the name of your department.

10 From the File menu, save the package as an `.msi` file on the administrative installation point.

Customized Files and Where to Deploy Them

Be sure to deploy the configuration files you create when you customize Reflection to the following locations. Reflection looks for custom files in these specific locations.

Workspace Settings

Deploy these files	To these folders
Customized workspace settings Application. settings Update.settings	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
Customized information privacy settings PrivacyFilters.xml (includes all Privacy Filter settings) PCIDSS.settings (includes all other Information Privacy settings)	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
Customized (or minimized) ribbon frame.settings	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
Customized Quick Access toolbar Application. settings frame.settings Reflection2007.Application.Ribbon.xuml	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
Restricted feature settings .access	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2
User templates .rsft	**For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2\templates

Shared Security Settings

File Type	Deployment location
Secure Shell settings *Global* ssh_config ssh_known_hosts *User-specific* config known_hosts	Note that the global filenames are different than the user-specific filenames, although the content may be the same. **For all users** [CommonAppDataFolder]\Micro Focus\Reflection . **For the user who installs** [PersonalFolder]\Micro Focus\Reflection\.ssh
Reflection Trusted Certificate Authorities trust_store.p12	**For all users** [CommonAppDataFolder]\Micro Focus\Reflection\.pki **For the user who installs** [PersonalFolder]\Micro Focus\Reflection\.pki
Reflection Certificate Manager settings pki_config	**For all users** [CommonAppDataFolder]\Micro Focus\Reflection\.pki **For the user who installs** [PersonalFolder]\Micro Focus\Reflection\.pki
Kerberos settings rsckrb5.xml	Settings in this location are migrated to the Windows registry for each Windows user the first time the user runs Kerberos Manager or any Reflection client configured to use Reflection Kerberos. **For all users** [CommonAppDataFolder]\Micro Focus\Reflection **For the user who installs** [AppDataFolder]\Micro Focus\Reflection

Reflection FTP Client Settings

File Type	Deployment location
rftp.xml	Settings in the .xml settings file are migrated to a settings.rfw file for each user the first time the user runs the FTP Client. Note: You can also use Modify User Settings to deploy FTP Client settings. **For all users** [CommonAppDataFolder] \Micro Focus\Reflection **For the user who installs** [AppDataFolder]\Micro Focus\Reflection
settings.rfw	**For the user who installs** The default user folder is: [PersonalFolder]\Micro Focus\Reflection

Regular Session Document Files and Session Settings

Deploy these files	To these folders
Session document files (.rdox, .rd3x, or .rd5x, .urlx)	Regular session document files and session settings files are installed under the location that is defined as the user data directory. The following locations are based on the default location of this directory: [PersonalFolder]\Micro Focus\Reflection **For only the user who installs:** [PersonalFolder]\Micro Focus\Reflection
Layout file (.rwsp)	[PersonalFolder]\Micro Focus\Reflection
keyboard map (.xkb)	[PersonalFolder]\Micro Focus\Reflection\Keyboard Maps
mouse map (.xmm)	[PersonalFolder]\Micro Focus\Reflection\Mouse Maps
Hotspots (.xhs)	[PersonalFolder]\Micro Focus\Reflection\Hotspots Maps
Custom Ribbon Interfaces (.xuml)	[PersonalFolder]\Micro Focus\Reflection\CustomUI
Themes (.themex)	[PersonalFolder]\Micro Focus\Reflection\Themes
Quickpads (.rqpx)	[PersonalFolder]\Micro Focus\Reflection\QuickPads

Custom Office Productivity Templates

Deploy these files	To these folders
Word and PowerPoint templates (.dotx or. ppt)	**For only the user who installs:** [PersonalFolder]\Micro Focus\Reflection

VBA SharedMacros Files and the VBA Common Project

Deploy these files	To these folders
Shared macros files sharedmacros.rd3x sharedmacros.rd5x sharedmacros.rdox	**For only the user who installs:** The location that is defined as the user data directory. The default location is: [PersonalFolder]\Micro Focus\Reflection
VBA Macros in the Common project `vbaProject.bin`	**For all users:** [CommonAppDataFolder]\Micro Focus\Reflection\Desktop\v16.2 **For only the user who installs:** [AppDataFolder]\Micro Focus\Reflection\Desktop\v16.2

NOTE: The default values for the [AppDataFolder], [PersonalFolder], and [CommonAppDataFolder] Reflection properties are:

```
[AppDataFolder]: C:\Users\username\AppData\Roaming
```

```
[PersonalFolder]: C:\Users\username\Documents
```

```
[CommonAppDataFolder]: C:\ProgramData
```

6 Modify the Installation

Create and deploy a transform to customize how Reflection is installed on user workstations. As defined by Microsoft, "a transform is a collection of changes applied to an installation. By applying a transform (*.mst file) to a base installation package, the installer can add or replace data in the installation database."

By deploying the transform with the Reflection base installation package, you can specify the installation directory, change the user data location, change the Remove or Add commands from the Windows Uninstall or change a program list, and change other default settings.

This article	
Create or Modify a Transform	Create a Microsoft transform (.mst file) that you can deploy with your product MSI file to change how the product is installed.
Walkthrough: Create a Transform	Example that shows how to add a desktop shortcut and disable the Change button on the Uninstall or change a program list.
Apply a Transform to Your Installation	How to add a transform to installs started with setup.exe

Create or Modify a Transform

You can use the Installation Customization Tool to create a transform (`.mst`) file.

- "Change the Installation Directory" on page 82
- "Modify Setup Properties" on page 82
- "Add/Modify Registry Data" on page 83
- "Select Features, Components, and Languages" on page 84
- "Add (Chain) Installations and Run Programs" on page 85
- "Install the Reflection Help" on page 86
- "Predefined System Folders" on page 87
- "Configure Shortcuts" on page 88

To create a transform with the Installation Customization Tool

1. From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2. In the Select Customization dialog box, select **Create a new Setup customization file for the following product** or **Open an existing Setup customization file**.

3 Select items from the list in the left panel to open configuration panels on the right, and then make your customizations.

4 Save your transform as an .mst file.

> **NOTE:** Micro Focus recommends that you save transform files in the same folder as the Reflection base installation package .msi file.

Change the Installation Directory

You can change the Reflection installation directory.

To change the installation directory

1 From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

<path_to_setup> `\setup.exe /admin`

2 From the **Select Customization** dialog box, do one of the following:

- Select **Create a new Setup customization file for the following produc**t.
- Select **Open an existing Setup customization file or Companion installer** and, in the **Open** dialog box, select an `.mst` file.

3 On the left panel of the Installation Customization Tool, click **Install location and organization name**.

4 In the **Default installation folder** box, specify where to install the product files.

> **NOTE:** List items that refer to folders (for example, `[ProgramMenuFolder]`) are pre-defined folder keywords. You can create customized directories by adding new folder names with typical directory syntax (such as, `[ProgramFilesFolder]\My Folder`). Alternatively, you can enter a fully qualified path (for example, `C:\Program Files\My Folder`), as long as that location is known to exist on the target machine.

Modify Setup Properties

In some cases you may want to customize your installation using Windows Installer (MSI) properties that support features not configurable on other Installation Customization Tool panels. For example, the installer property ARPHELPLINK, sets the URL used by the support link in the Programs and Features list.

> **NOTE:** If you want to create a custom (non-default) user data location that will be used by all installed Reflection products, specify the product-specific property WRQ_USERDIR. This property affects only those applications whose default user data directory is [PersonalFolder]\Micro Focus\Reflection\. Reflection applications that use this default data location include: Reflection Workspace, Reflection FTP Client, Reflection Kerberos Manager, Reflection Key Agent, Reflection IBM Printer, and Reflection X (legacy). Reflection X Advantage does not use this user data location.
>
> To set the location where Reflection X Advantage artifact files are saved, specify the file path in the RXA_ARTIFACTS_DIR property.

To modify installation properties

1. From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2. From the **Select Customization** dialog box, do one of the following:
   - Select **Create a new Setup customization file for the following produc**t.
   - Select **Open an existing Setup customization file or Companion installer** and, in the **Open** dialog box, select an `.mst` file.

3. From the Installation Customization Tool navigation pane, select **Modify setup properties**.

4. Click **Add** to open the **Add/Modify Property Value** dialog box.

5. In the **Name**box, use the drop-down list to select commonly-used public properties that are standard to the Windows Installer. Select an item in the list to see a brief description of the selected property. For additional information about Windows Installer properties, see the Microsoft Windows Installer Guide (http://msdn.microsoft.com/en-us/library/windows/desktop/aa372845(v=vs.85).aspx). Some Micro Focus products support additional properties that do not appear in the drop-down list. You can configure these properties by manually entering the property name.

6. From the **File**menu, select **Save As**and save the file to the same folder as the installer package file for Reflection (`setup.exe`).

Add/Modify Registry Data

To modify registry data

1. From your administrative installation point, open the Installation Customization Tool from a shortcut or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2. From the **Select Customization** dialog box, do one of the following:
   - Select **Create a new Setup customization file for the following produc**t.
   - Select **Open an existing Setup customization file or Companion installer** and, in the **Open** dialog box, select an `.mst` file.

3. From the navigation pane, click **Add registry data**.

4. To add a new registry value, click **Add**.

   -or-

   To modify a registry value in the table, select the value, and then click **Modify**.

5 Specify registry keys and values to add or modify during the installation process. By modifying registry values, you can change the way the application operates. For example, for certain Micro Focus applications, you can add a value that specifies to never save settings on exit.

In this field	Enter or select
Key	The complete Registry path from the root, for example: `HKEY_LOCAL_MACHINE\SOFTWARE\Reflection\Rwin\Reflection`
Name	The registry value name, for example: `SaveChanges` If the **Name**box is blank, the data entered into the **Value**box are written to the Default registry key.
Type	The data type of the value. For example: `DWORD` Types include strings, integers (DWORD), or binary values.
Value	The value. For example: `0x00000000 (0)`

Select Features, Components, and Languages

You can select which features, components, and languages to install for your end users. In addition, you can make features available to users for a later installation or hide them from view.

To select features, components, and languages to install

1 From your administrative installation point, open the Installation Customization Tool from a shortcut or by typing the following command line:

<path_to_setup> `\setup.exe /admin`

2 From the **Select Customization** dialog box, do one of the following:

- Select **Create a new Setup customization file for the following product**.
- Select **Open an existing Setup customization file or Companion installer** and select an .mst file.

3 From the Installation Customization Tool navigation pane, select **Set feature installation states**, and for each feature, choose from the following states:

Choose	To do this
Feature will be installed on local hard drive	Add a feature to the installation. **NOTE:** Some features listed under a selected feature may not be included when you select to install the higher-level feature. The features that are included are the recommended defaults. If you select the higher level feature a second time, all sub-features will be included.
Feature will be installed when required	Advertise a feature.
Feature will be unavailable	Leave a feature uninstalled. End users will still be able to select and install the item from the Windows **Program and Features** or the **Add or Remove Programs** control panel.

Related Topics

- "Apply a Transform to Your Installation" on page 90

Add (Chain) Installations and Run Programs

Reflection makes it easy to "chain" installs. You can set up an install to run companion install packages automatically before or after the primary installation. You can also specify to run other scripts or programs. For example, the Reflection product Help is available as a separate program (`.msi`) that you can add to the base product installation. See "Install the Reflection Help" on page 86.

NOTE: This method of chaining installations applies only to installs performed with `setup.exe`. It does not apply to installs that use the MSI command-line method.

To chain installations and programs

1 From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

<path_to_setup> `\setup.exe /admin`

2 From the **Select Customization** dialog box, do one of the following:

- Select **Create a new Setup customization file for the following product**.
- Select **Open an existing Setup customization file or Companion installer** and select an .mst file.

3 From the navigation pane, select **User Interface** and then select **Use this customization with interactive installs using setup.exe**.

4 From the navigation pane, select **Add installations and run programs**.

5. Click **Add**.

   The **Add/Modify Program Entry** dialog box opens.

6. In the **Target** list, enter or select the folder where the program `.exe` file or the `.msi` file resides, and then enter the executable to run. For example:

   ```
   msiexec.exe
   ```

7. Under **Arguments**, enter the command-line arguments to execute. For example:

   ```
   /i my_installation.msi
   ```

8. To specify when to run the program, select one of the following:
   - **Run this program after base product has been installed**
   - **Run this program before the base product has been installed**.

   NOTE: For most cases, select **Run this program after the base product has been installed**. If you select **Run this program before the base product has been installed** and the program fails, Reflection is not installed.

9. Repeat these steps to add other programs or `.msi` files.

10. To change the execution sequence, use the arrows next to **Move**in the lower-left area of the pane. To remove a program from the list, select it in the list and click **Remove**.

Install the Reflection Help

By default, the Reflection Help browser opens the Help from the Micro Focus website. To allow users to open the Help from the local hard drive you must install the product Help after installing Reflection and change the Reflection Workspace (page 51) settings to use the installed help system.

To configure the Reflection Workspace to open Help locally

1. Open the **Reflection Workspace Settings > dialog box**.

   The steps depend on your user interface mode.

User Interface Mode	Steps
Ribbon (Office 2007)	On the Reflection button (if using the Office 2007 Look and Feel), choose **Reflection Workspace Settings**.
Ribbon (Office 2010)	On the **File** menu, choose **Reflection Workspace Settings**.
Reflection Browser	On the Reflection menu, choose **Settings** and then **Reflection Workspace Settings**.
Mobile UI	Tap the Gear icon and then select **Reflection Workspace Settings**.

2. Under **Workspace Settings**, select **Configure Workspace Attributes**.
3. Under **Help System**, select **Use installed help system**.

   If Reflection can't find the Help files on the local drive, it will start your default browser and open the help files from Micro Focus website.

To install the Help locally

Do one of the following:

- On a workstation where Reflection is already installed, from the installation image, open the HelpInstaller folder and double-click `setup.exe`.

  - or -

- Chain the Help installer to the main installation, selecting **Run this program after base product has been installed**. For more information, see "Add (Chain) Installations and Run Programs" on page 85.

To set up Help on a Windows server

NOTE: If you install Reflection on a Windows Server and Enhanced Security (IE ESC) is enabled, viewing the Help will result in security prompts each time you open a page. To resolve this problem, do the following:

---

1. Turn off **Internet Explorer Enhanced Security for users and administrators**.
2. Turn Off **Internet Explorer Enhanced Security** on Windows 2008 Server.
3. Enable scripting for Internet Explorer. See Technical Note 2293 (http://support.attachmate.com/techdocs/2293.html).

Predefined System Folders

When you configure destination locations using the Installation Customization Tool, your options include Reflection properties that correspond to Windows system folder properties (http://msdn.microsoft.com/en-us/library/aa372057.aspx?ppud=4). During installation, the Windows installer expands these to show the appropriate location for your operating system.

The list of available folders for adding files to a companion installer depends on whether you are installing for all users (the default) or for individual users.

All-user installations

Reflection Property name	Default Windows location	Default path using Windows variables
[CommonAppDataFolder]	`C:\ProgramData`	`%Programdata%`
[CommonDocumentsFolder]	`C:\Users\Public\Documents`	`%Public%\Documents`
[CommonFilesFolder]	`C:\Program Files\Common Files`	`%ProgramFiles%\Common Files`
[ProgramFilesFolder]	`C:\Program Files`	`%ProgramFiles%`
[RootDrive]	`C:\`	`\`
[WindowsFolder]	`C:\Windows`	`%windir%`

Individual user installations

Reflection Property name	Default Windows location	Path using Windows variables
[AppDataFolder]	`C:\Users\<user>\AppData\Roaming\`	`%appdata%`
[LocalAppDataFolder]	`C:\Users\<user>\AppData\Local\`	`%localappdata%`
[PersonalFolder]	`C:\Users\<user>\Documents\`	`%userprofile%\documents`
[RootDrive]	`C:\`	`\`
[%UserProfile]	`C:\Users\<user>`	`%userprofile%`

Configure Shortcuts

You can change the attributes associated with the pre-defined Reflection shortcuts. Also, you can configure shortcuts associated with files you've added to a custom install package.

NOTE: This method of chaining installations applies only to installs performed with `setup.exe`. It does not apply to installs that use the MSI command-line method.

To configure shortcuts

1. On a workstation on which you have installed Reflection, open the Installation Customization Tool from a desktop shortcut (if you set up a shortcut as shown on page 11) or from a command line as follows:

   ```
   path_to_setup\setup.exe /admin
   ```

2. From the **Select Customization** dialog box, do one of the following:
   - Select **Create a new Setup customization file for the following product**.
   - Select **Open an existing Setup customization file or Companion installer** and select an .mst file.
3. From the Installation Customization Tool navigation pane, choose Configure shortcuts.
4. Select the shortcut that you want to configure, and then click **Modify**.
5. In the Modify Shortcut dialog box, in the **Location** list, enter or select the folder where you want the shortcut to reside.

   NOTE: List items that refer to folders (for example, [ProgramMenuFolder]) are pre-defined folder keywords. You can create customized directories by adding new folder names with typical directory syntax (such as, [ProgramFilesFolder]\My Folder). Alternatively, you can enter a fully qualified path (for example, C:\Program Files\My Folder), as long as that location is known to exist on the target machine.

6. Enter a descriptive name and tooltip for the shortcut.
7. Enter the command-line arguments for the shortcut in the Arguments window.

Walkthrough: Create a Transform

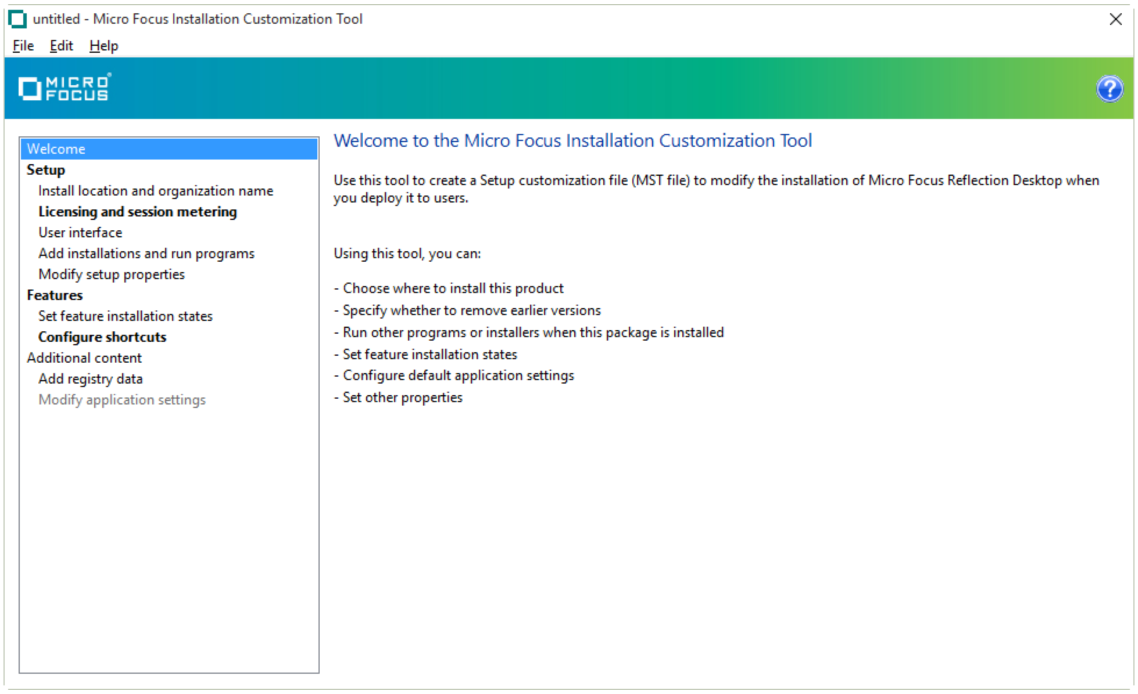
This example shows how to set up a transform that can be applied to an install on a command line. To automatically deploy the transform in an installation that uses the Reflection Setup program, see “Apply a Transform to Your Installation” on page 90.

To create a transform that adds a desktop shortcut and disables the Change button

1. From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2. Select **Create a new setup customization file for the following product**, and then click **OK**.


untitled - Micro Focus Installation Customization Tool
File
Edit
Help
MICRO FOCUS
Welcome
Setup
Install location and organization name
Licensing and session metering
User interface
Add installations and run programs
Modify setup properties
Features
Set feature installation states
Configure shortcuts
Additional content
Add registry data
Modify application settings

Welcome to the Micro Focus Installation Customization Tool

Use this tool to create a Setup customization file (MST file) to modify the installation of Micro Focus Reflection Desktop when you deploy it to users.

Using this tool, you can:

- Choose where to install this product
- Specify whether to remove earlier versions
- Run other programs or installers when this package is installed
- Set feature installation states
- Configure default application settings
- Set other properties

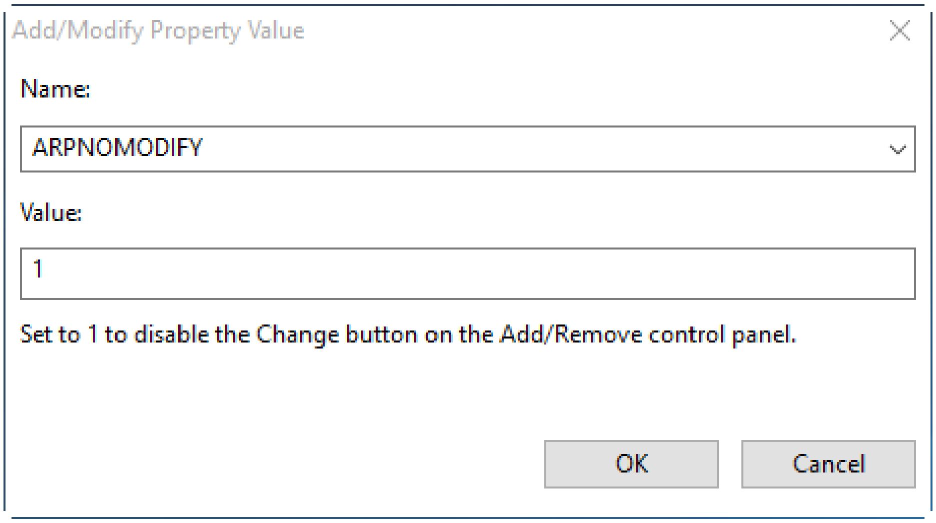
3. From the navigation pane, click **Licensing and session metering** and then select **I accept the terms of the Software License Agreement**.

4. The following optional steps describe how to create a Reflection Workspace shortcut to the user’s desktop and remove the **Change**button from the Windows **Uninstall or change a program** list.

   4a From the navigation pane, click **Modify setup properties**.

   4b In the lower-right corner, click **Add**.

   4c In the **Add/Modify Property Value** dialog box, for **Name**select ARPNOMODIFY and for **Value**enter 1. This step removes the **Change**button.


Add/Modify Property Value

Name: ARPNOMODIFY

Value: 1

Set to 1 to disable the Change button on the Add/Remove control panel.

4d From the navigation pane, click **Configure shortcuts**.

4e Under **Modify shortcuts for this product**, select **Reflection workspace**and click **Modify**.

4f In the **Modify shortcut**dialog box, in the **Location** field, select **[DesktopFolder]**and click **OK**. This step creates a desktop shortcut.

5 From the **File**menu, save the transform on the administrative installation point as `myCustomInstall.mst`.

Apply a Transform to Your Installation

If you have created a transform to customize how Reflection is installed, you need to deploy the transform with the primary installation. (This is in contrast to companion installation packages, which can be chained with the primary installation or installed separately.)

You can include a transform by modifying the setup.ini file or by adding it to the setup.exe command line. Any install started with `setup.exe`or with command-line installs can include a transform.

To apply a transform by modifying the setup.exe program

1 From your administrative installation point, open the Installation Customization Tool from a shortcut (page 22) or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`

2 From the **Select Customization** dialog box, do one of the following:

   - Select **Create a new Setup customization file for the following product**.
   - Select **Open an existing Setup customization file or Companion installer** and select an .mst file.

3 Go to the **User interface** panel.

4 Select **Use this customization with interactive installs using setup.exe**.

   When you save your transform with this option selected, the Installation Customization Tool automatically updates the `setup.ini` file to apply the transform to the Reflection installation by adding the following line to the `[Setup]` section:

```
CustomTransform=<your_transform.mst>
```

This modification to `setup.ini` means that any install using `setup.exe` (using either the interactive user interface or using `setup.exe` on a command line) will automatically apply this transform.

5 Save the transform to the default location (the folder that includes `setup.exe`).

The transform can now be deployed to end users via the `setup.exe` file. (Users can run `setup.exe` directly or `setup.exe` can be called from a script or initiated from a command line.)

NOTE: If the `setup.ini` file already specifies the transform file you want to install (as noted in step 3 of the previous procedure) do not specify the transform on the command line.

To add the transform to a setup.exe command-line install

- From the command line, use the following syntax:

```
<path_to_setup>\setup.exe /install TRANSFORMS=<transform.mst>
```

7 Deploy Reflection

This chapter provides instructions for deploying Reflection, session document files, and other configuration files.

This article	Describes
Deploy with the Reflection Setup program	How to deploy using the setup.exe command line.
Deploy with MSI	How to deploy using the MSI command line and handle prerequisites that must be installed before you deploy with MSI directly.
Publish with Active Directory	Requirements for assigning and publishing your product installation using Microsoft Active Directory.
Deploy with System Center Configuration Manager	How to deploy Reflection with Microsoft Systems Center Configuration Manager (or Microsoft Systems Management Server).
Distribute Software Updates	How to install software updates with the Micro Focus Patch utility included with your distribution.
Remove an Installation	How to remove Reflection or a package of configuration files.
Repair an Installation	How to use the Repair option, which automatically searches for and replaces missing or corrupted files.

Deploy with the Reflection Setup program

The Reflection Setup program (`setup.exe`) is the recommended tool for installing and deploying Reflection.

`Setup.exe` has a command-line interface that you can run from a command line, a batch file, or a deployment tool. You can type command-line options to set installation parameters and limit user interaction as Reflection is installing, or use command-line options to prepare Reflection for installation by users. Command-line installations may have additional "System Requirements" on page 14.

To see a list of available command-line parameters, type:

```
setup.exe /?
```

To install a companion installer package with the Setup program, add it to a transform and then apply that transform to `setup.exe`. This "chains" the companion installer package to the main installation. You can set the package to deploy before or after the main installation.

The following procedures show command-line options commonly used for deployment, including switches for silent installations.

To deploy Reflection "out-of-the-box"

Use this command to deploy Reflection with default settings.

- At a command prompt on a test workstation, enter:

  path_to_administrative_installation_point `\setup.exe /install`

To perform a silent installation

- At a command prompt, change to the directory in which the `setup.exe` file resides and do one of the following:

To perform	Type
A silent install that displays a progress bar and disables the Cancel button	`setup.exe /install /passive`
A silent install with no display	`setup.exe /install /quiet`

To deploy Reflection and a transform

- At a command prompt, enter:

  path_to_administrative_installation_point `\setup.exe /install TRANSFORMS=` *myCustomInstall* `.mst`

NOTE: You can also set up the Reflection Setup program to deploy a transform automatically. See "Add (Chain) Installations and Run Programs" on page 85.

Deploy with MSI

You can deploy Reflection directly from the MSI command line. You can also deploy companion .msi files that you have created to contain your custom configuration files.

Deploy Reflection from MSI Command Line

Use these procedures to install Reflection from a command line with MSI.

To customize your installation, specify Windows Installer properties on the command line or pass them in a transform file. For a list of public properties that are standard to the Microsoft Windows Installer, see the "Modify Setup Properties" on page 82 pane in the Installation Customization Tool, or refer to the Microsoft Windows Installer Guide (http://msdn.microsoft.com/en-us/library/windows/desktop/aa372845(v=vs.85).aspx).

If you want to create a custom (non-default) user data location that will be used by all installed Reflection products, specify the product-specific property WRQ_USERDIR. This property affects only those applications whose default user data directory is `[PersonalFolder]\Micro Focus\Reflection\`. Reflection applications that use this default data location include: Reflection Workspace, Reflection FTP Client, Reflection Kerberos Manager, Reflection Key Agent, Reflection IBM Printer, and Reflection X (legacy). Reflection X Advantage does not use this user data location.

Handling prerequisites in command line installs

If you use setup.exe for your install, the setup program checks for any prerequisites required by the features you have selected and installs them automatically.

If you use msiexec.exe for your install, prerequisites are not installed automatically. You need to install them separately if they are not already on your users' workstations. You can find installers for the required prerequisites in the Prerequisites folder in the distribution media, or in your administrative installation. The prerequisites you need to install depend on which programs and features you are installing:

- All Reflection Workspace features require Microsoft .NET Framework 4.7.1. If you attempt an install using msiexec.exe and this prerequisite is not found, a message displays and the installer stops. To install the .NET Framework, run the executable file in Prerequisites\DotNet471.
- The Visual Basic for Applications feature requires Microsoft VBA 7.1. Use the core and language-specific *.msi packages in the Prerequisites\VB71 folder.

Handling upgrades in command line installs

If you are upgrading a previous version of Reflection, the prior version must be uninstalled before you can install the current version.

If you use setup.exe for your install, the setup program checks for a previous versions of Reflection and uninstalls it automatically if one is found.

If you use msiexec.exe for your install, you must first manually uninstall any earlier versions. If you upgrade these products by deploying the .msi file directly and have not removed the earlier version, a message displays telling you to uninstall the older software first.

To deploy Reflection "out-of-the-box" directly with MSI

- At a command prompt on a test workstation, change to the directory in which the `.msi` file resides and enter:

  `msiexec /i` *path_to_administrative_installation_point* `\` *yourVersion* `.msi`

  where *yourVersion* `.msi` is the specific version of the reflection MSI that you downloaded (for example, `ReflectionV16.msi`).

To deploy Reflection and a transform directly with MSI

- At a command prompt, enter:

  `msiexec /i` *path_to_administrative_installation_point* `\` *yourVersion* `.msi TRANSFORMS=` *yourCustomInstall* `.mst`

  where yourVersion `.msi` is the specific version of the Reflection MSI that you downloaded (for example, `ReflectionV16.msi`).

Deploy Companion MSI File from MSI Command Line

You can deploy configuration files that are “packaged” in a companion installer package separately from Reflection. This allows you to deploy and maintain these files between Reflection software updates without removing Reflection.

If you use the Host Access Management and Security Server, you can upload MSI files to the Package Manager and silently deploy them to users' workstations. See Use Central Management to Deploy MSI Packages.

To deploy a companion installer package directly with MSI

- At a command prompt, enter:

  `msiexec /i` *path_to_administrative_installation_point* \ *your_companion_file* `.msi`

To remove a companion installer package directly with MSI

- At a command prompt, change to the directory in which the companion installer package file resides and enter:

  `msiexec /x` *your_companion_file* `.msi`

Publish with Active Directory

To assign and publish your product installation using Microsoft Active Directory, you must meet the following requirements:

- Windows Administrative Tools are installed on your workstation.
- You are a member of **Domain Admins** and **Group Policy Creators and Owners**. (This is required to publish software.)

For more information, see "Active Directory groups" in the Microsoft Management Console help.

To install with Active Directory

1. From the **Active Directory User and Computers Console**, advertise your product installation to members of any organizational units in your Active Directory using appropriate transform modifications.
2. If multiple transforms are specified, make sure that the listed order of the transforms is correct, and click **OK**. (If you need to change the order for any reason after you click **OK**, you will have to start over again.)

   NOTE: For more information about assigning and publishing, see "assigning applications" and "publishing applications" in the Microsoft Management Console help.

Deploy with System Center Configuration Manager

You can deploy Reflection with Microsoft Systems Center Configuration Manager (or Microsoft Systems Management Server).

To deploy with System Center Configuration Manager

1. Create an administrative install image on your site server.

   This serves as the administrative installation point for deployment.

2. Use the product Package Definition File (`.sms`) to create the product installation package.

   NOTE: The Package Definition File (`.sms`) is created during the administrative installation and can be found at the root of the administrative installation point. Alternatively, you can reference the `.msi` file directly — consult the Microsoft SMS documentation for more information.

3. Advertise the installation packages to your users.

Distribute Software Updates

Software updates are distributed as Microsoft `.msp` files. You can deploy these updates with the Micro Focus Patch utility included with your distribution. Patch log files are saved in the user's Windows temporary folder(`%tmp%`) with a generated name, using the form a `tmpatch xxxxxx .log`.

To distribute updates with the Micro Focus Patch utility

1. From the distribution image, double-click the self-extracting executable update file.

   The Micro Focus Patch utility opens.

2. Follow the instructions provided by the utility.

   For detailed instructions on upgrading your Micro Focus products, refer to the Micro Focus Patch Utility Help.

To distribute updates from the command line

- To Install to a (clean) workstation and apply a patch:

  `msiexec /i <`*path_to_original*`.msi> PATCH=<`*path_to_patch*`.msp> /qb`

- To apply a patch to an administrative installation:

  `msiexec /p <`*path_to_patch*`.msp> /a <`*path_to_admin*`.msi> /qb`

- To reinstall to a workstation (for example, after applying patch to admin):

  `msiexec /i <`*path_to_admin*`.msi> REINSTALLMODE=vomus REINSTALL=ALL /qb`

Remove an Installation

To remove Reflection, you can use the Windows Control Panel, the Reflection Setup program user interface, or a command line. To remove a companion installation, you can use the Windows Control Panel or a command line.

NOTE: You must log on with administrator privileges to remove Reflection.

To remove an installation using the Windows control panel

1. To open the **Programs and Features** control panel go to **Start** > **Control Panel** > **Programs and Features**. (On older Windows systems, this Control Panel is called Add or Remove Programs.)
2. Select the name of the installation that you want to remove.
3. Click **Uninstall**(or **Remove**).

To remove an installation with the Reflection Setup program user interface

1. From an administrative installation image, click the `setup.exe` file.
2. From the tab, select **Remove**, and then click **Continue**.

To remove an installation from the Reflection Setup program command line

CAUTION: If you use the following instructions to find the product code in the registry, make sure you do not change any registry values. Changing these values can damage an installation. If you prefer not to use the registry, you can get the product code by contacting Technical Support.

1. Open the registry editor (`regedit.exe`) and find this key:

   For 32-bit platforms

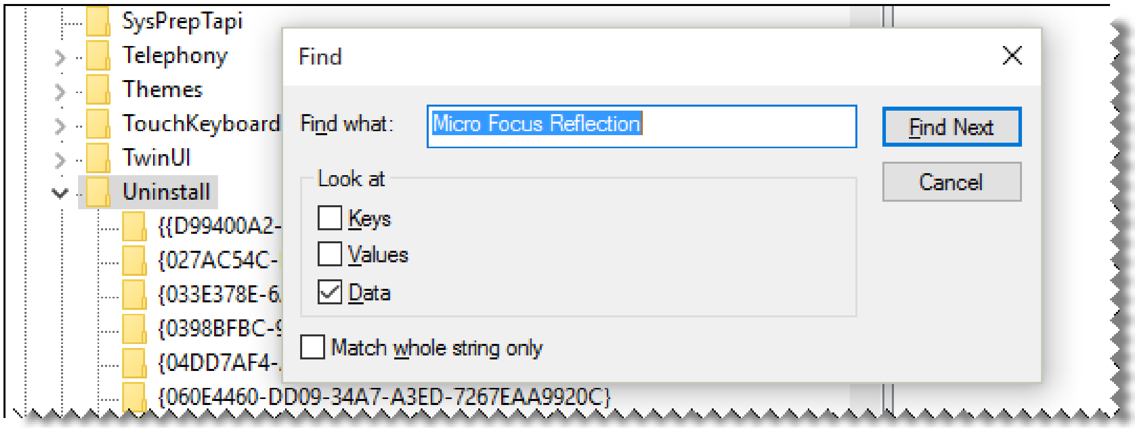
   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall
   ```

   For 64-bit platforms

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersi
   on\Uninstall
   ```

   Each key under the `Uninstall`key is the product code or Globally Unique Identifier (GUID) for a product installed on the computer.

2. In the `Uninstall`key, search for `Micro Focus Reflection` to locate the GUID associated with the product.



Find what: Micro Focus Reflection

Look at

☐ Keys

☐ Values

☑ Data

☐ Match whole string only

3. At the command prompt or the Start menu Run command, change to the directory in which the `setup.exe` file resides and enter:

```
setup.exe /uninstall ProductCode
```

where *ProductCode* is the Globally Unique Identifier (GUID) that is the principal identifier for the product.

To remove a companion installer package with MSI directly

- At a command prompt, change to the directory in which the companion installer package file resides and enter:

```
msiexec /x your_companion_file.msi
```

Repair an Installation

If you are experiencing problems with your installation, you can use the **Repair** option, which automatically searches for and replaces missing or corrupted files.

To repair an installation with the Reflection Setup program user interface

1. From an administrative installation image, click the `setup.exe` file.
2. Click **Repair,** and then follow the installer instructions.

To repair an installation using Windows Add/Remove

1. From the Windows **Programs and Features** (or the **Add or Remove Programs**) control panel, select the name of the installation that you want to repair, and then click **Change**.
2. From the Reflection Setup program, select **Repair**, and then click **Continue**.

8 Using a Centralized Management Server

You can centrally manage, secure, and monitor users' access to host connections with the Micro Focus Host Access Management and Security Server (MSS), a separately available product that is designed to provide centralized management for Reflection sessions.

Using this centralized management server, you can grant or deny access based on group or role, quickly apply security updates and configuration changes to align with changing regulatory or business needs, and make post-install adjustments on the fly. MSS allows you to configure and lock down large numbers of desktops with ease.

MSS Servers and Add-Ons

MSS includes two servers that you can use to configure and monitor your sessions:

- **The Administrative Server**

  Using the Administrative Server's central console, you can define terminal emulation sessions, configure and save session settings, and then manage and configure secure settings for those sessions. You can also use directory services, such as Active Directory, to authorize access to host applications—without changing your LDAP schema or data. Sessions that you create in this way are saved to the server and can be made available to users from the server and modified at any time. See *"Create and Deploy Sessions and Settings with the Administrative Console" on page 103*.
- **The Metering Server**

  Use the Metering Server to track Reflection sessions and determine how many client workstations use the product. See *"Enable Usage Metering" on page 104*.

You can enhance your ability to manage sessions and reinforce security with MSS Add-Ons. You'll gain additional critical functionality when you pair MSS with these products:

- **Security Proxy Add-On**

  The Security Proxy acts as a proxy for terminal sessions and provides token-based access control, routing encrypted network traffic to and from user workstations. The Security Proxy Server can be installed on the same server as the Administrative Server or on another system.

  To set up the Security Proxy for client authorization, pass through, end-to-end SSL/TLS, and end-to-end SSH security connections, see *"Connect to Hosts using the Security Proxy Add-On" on page 107*.
- **Terminal ID Manager Add-On**

  You can use the Terminal ID Management Add-On to monitor a pool of resource IDs that a client can use to establish a host session, thereby eliminating the need to create configurations for every client. The Terminal ID Manager enables you to pool terminal IDs, track ID usage, and manage inactivity timeout values for specific users, thus conserving terminal ID resources and significantly reducing operating expenses. The Terminal ID Manager can be installed on the same server as the Administrative Server or on another system using either the automated installer or a manual installation. See *“Set Up Terminal ID Management for Reflection Desktop Sessions” on page 113*.

- **Automated Sign-On for Mainframe Add-On**

  The Automated Sign-On for Mainframe Add-On enables users to authenticate to a front-end system using a modern form of authentication (such as a smart card, certificate, LDAP password, Kerberos, etc.) and then be automatically logged on to a z/OS mainframe application. To add Automated Sign-On for Mainframe, you need to install the activation file and configure settings using the Administrative Console. Some configuration is also needed on the mainframe. See *“Set up Automated Sign-On for Mainframe Sessions” on page 116*.

Requirements and Best Practices for Using Centralized Management

Requirements for Centralized Management

To use centralized management, your environment must meet the following requirements.

- Reflection Desktop must be installed on the Administrative workstation and on user computers.
- The users' Reflection Desktop clients must be configured to use Centralized Management as shown in *Set up Reflection to Access a Centralized Management Server* in the *Reflection Help & HowTo Guide*. If user workstations are not configured with this setting, you will need to configure and save this workspace setting and then deploy it to users (see Package Sessions and Custom Settings Files).
- Java must be enabled in the browser you use to run the Administrative Console.
- Management and Security Server (MSS) must be installed on one or more servers available over the network. You will need to know the administrative credentials to log onto the Management Server.

Best Practices for using Centralized Management

- If your session uses associated settings (such as a theme or keyboard map file), be sure to select Send as compound session to include these settings. When you save the session as a compound file, all of the QuickPads, keyboard maps, themes, mouse maps, hotspots, and ribbons that apply to that session are saved in the session file. Compound files simplify the deployment process because you do not have to deploy these settings in separate files.

  NOTE: Custom workspace settings are not saved in compound session document files. These settings are saved in other files that you will need to package and deploy separately.

- Be sure to choose to store all settings files in a trusted location.

Create and Deploy Sessions and Settings with the Administrative Console

Using the Administrative Console, you can define terminal emulation sessions, configure and save session settings, and then manage and configure secure settings for those sessions.

Create or Modify a Centrally Managed Session

You can create and manage Reflection sessions from the Administrative Console. Sessions that you create in this way are saved to the server and can be made available to users from the server and modified at any time.

For instructions that show how to add or modify sessions, see the *Management and Security Server Administrator Guide*, in the *Host Access Management and Security Server Documentation*.

Make Centrally Managed Sessions Available to Users

After you have saved sessions to the centralized management server, you can make these sessions available to users who have Reflection installed on their workstation. Users can launch these sessions in the same way they launch locally created sessions. When Centralized Management is configured, each time a user opens the workspace:

- Reflection contacts the server and prompts for user credentials (if required by the server).
- Sessions that are available to the user are downloaded to the user data directory.

For instructions that show how to make sessions available to users, see the *Management and Security Server Administrator Guide*, in the *Host Access Management and Security Server Documentation*.

Use Central Management to Deploy MSI Packages

Use the Administrative Console to upload companion install packages (.msi) to the Administrative Server for deployment to specified users. Companion install packages can be created in the Installation Customization Tool or other MSI creation tools, and may include toolbars, macros, keyboard maps, and settings files.

NOTE: You can simplify deployment of MSI packages by saving your customized sessions as compound session document files.

These packages are automatically deployed to a user's desktop when the user logs on to the Management and Security Server or starts a Reflection Workspace session with Centralized Management enabled.

For instructions that show how to configure, upload, or update a package, see the *Management and Security Server Administrator Guide*, in the *Host Access Management and Security Server Documentation*.

Enable Certificate Management for IBM Terminals

You can import certificates to the centralized management server and then make them available to Reflection users on their workstations. When configured to use certificate management, Reflection uses these managed certificates instead of the Reflection or Windows Certificate store.

To Upload a Certificate

1. Log on to the Management and Security Server Administrative Console as an administrator.
2. From the **Configure Settings** sidebar menu, select **Trusted Certificates**.
3. Select **Import** and upload your certificate to the Management and Security Server.
4. To configure Reflection Desktop to use certificate management, select the **Enable Certificate Management**setting on the Configure Centralized Management dialog box and then deploy this setting to users.

   When launching the workspace with this setting enabled, Reflection uses the managed certificates instead of the Reflection or Windows Certificate store.

Enable Usage Metering

Usage metering allows you to track Reflection sessions and determine how many client workstations use the product.

When a user starts the Windows-based client session, the session notifies the Metering server, which begins logging the product usage. The session then sends updates to the Metering server at regular intervals until the user shuts down the client.

How can you use metering?

You can set up metering to monitor host sessions using the following features.

- ***License Pool Configuration*** A license pool comprises the licenses for a given product, type (production, evaluation, pre-release), and VPA number. When the server receives a metering message from a new product/type/VPA, a license pool is automatically created in the server's list of license pools.
- ***Usage Logging*** Connection activity is recorded in daily log files. You can specify the number of days the log files are stored before they are automatically deleted. The centralized management server uses the log data to generate reports summarizing usage information, such as connection activity, hourly usage levels, and connections from a specific client or to a specific host. You can use the Metering Reports page to create a variety of reports that use the logged data.

- ***Connection Monitoring*** When the client and server are configured for metering, you can monitor the number of user workstations connecting to host computers. You can also choose to receive e-mail notification when the concurrent number of workstations using metered clients increases beyond a specified limit. This notification feature also allows you to monitor concurrent product usage without enforcing a license limit.
- ***Concurrent License Enforcement*** The concurrent license enforcement feature allows you to control the total number of user workstations using Reflection to make host connections at the same time. When you configure metering, you can enter the maximum number of computers allowed to concurrently run Reflection; all attempted license use beyond this number can be blocked.

Setting up Metering

To enable metering, you must configure client workstations to report to the Metering server. You can do this by configuring metering with a transform as part of the initial installation or by setting group policy after Reflection is installed.

Requirements

- Micro Focus Management and Security Server (MSS) with a Metering server must be installed and configured for Reflection as shown in the *Management and Security Server Installation Guide*, available in the *Host Access Management and Security Server Documentation*.
- You must know the complete URL for the Metering server. The syntax is: `http://[ `*`host name`*`]:[ `*`port`*`]/[ `*`metering server context name`*`]/meter.do`. (If you used the default port, you can omit the colon and port number.)

  For example:

  ```
  http://Myserver.com:80/meter/meter.do
  ```

To enable metering with a transform

1. From your administrative installation point, open the Installation Customization Tool from a shortcut or by typing the following command line:

   <path_to_setup> `\setup.exe /admin`
2. In the Select Customization dialog box, select **Create a new Setup customization file for the following product**.
3. On the left panel, select **Licensing and session metering**.
4. Select to accept the terms of the license agreement on behalf of users.
5. In the **Meter As** list, select the product you want to meter.
6. In the **Metering URL**box, enter the URL of your metering server.

   Syntax:

   ```
   http://[host name]:[port number]/[metering server context name]/
   meter.do
   ```

   For example:

   ```
   http://Myserver.com:80/meter/meter.do
   ```

NOTE: If you use the default port, you can omit the colon and port number.

7 If you want to prevent users from launching Reflection when the metering server is not available, Select **Require metering**. (Enabling this setting can be useful when you are creating a trial installation and want to test to see if the metering server is running and available.)

8 If you are going to deploy using the `setup.exe` file in your administrative installation point, click **User Interface** in the left pane, and then select **Use this customization with interactive installs using setup.exe**.

9 Save your transform as an .mst file.

When you deploy this transform with your installation, the Reflection client is configured to use metering.

NOTE: Micro Focus recommends that you save transform files in the same folder as the Reflection base installation package .msi file.

To enable metering via group policy

1 Download and install the administrative template file for Reflection (http://download2.attachmate.com/fileinfo.asp?filename=ReflectionPolicy.zip).

2 Open the Windows Group Policy Editor (`gpedit.msc`).

3 Under **Computer Configuration**, right-click on **Administrative Templates** and select **Add/Remove Templates**.

4 Click **Add**, select the Reflection.adm file you need to add, and then click **Open**. The added ADM file is listed in the Add/Remove Templates dialog box, in the **Current Policy Templates** list.

5 Open **Computer Configuration**> **Administrative Templates**> **Classic Administrative Templates**(**ADM**) > **Reflection Settings**> **Client Metering**.

6 Select the **Configure Client Metering** policy setting and then select to edit this setting.

7 In the Configure Client Metering dialog box, select **Enabled**.

8 In the **Metering web server**box, enter the URL of your metering server.

Syntax:

```
http://[host name]:[port number]/[metering server context name]/
meter.do
```

For example:

```
http://Myserver.com:80/meter/meter.do
```

NOTE: If you use the default port, you can omit the colon and port number.

9 Select **Require connection to metering server** only if you want to prevent users from launching Reflection when the metering server is not available. (Enabling this setting can be useful when you are creating a trial installation and want to test to see if the metering server is running and available.)

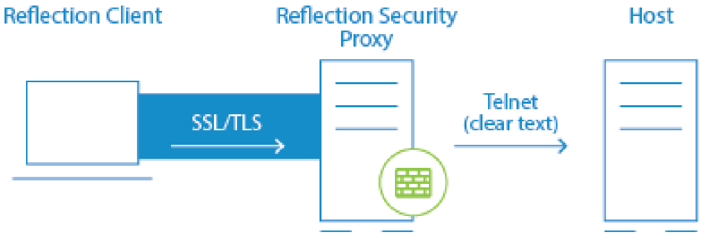
Connect to Hosts using the Security Proxy Add-On

The Security Proxy Add-On acts as a proxy for terminal sessions and provides token-based access control, routing encrypted network traffic to and from user workstations.

NOTE: The Security Proxy Add-On requires the base installation of Host Access Management and Security Server. It is not included with the Management and Security Server license. To activate this product, you must purchase a separate license.

Using the Security Proxy Add-On, you can set up the following types of centrally managed secure connections.

Connect using...	Description
Client Authorization	When using the default configuration for the Security Proxy, users are authorized using security tokens. Transmitted data between the client and the Security Proxy is encrypted; transmitted data between the Security Proxy and the host is not. The Security Proxy server should be installed behind a corporate firewall when used in this mode. Before you create a connection, review the "Requirements for connections through the Security Proxy" on page 109 and "Getting to the Reflection Desktop Security Settings" on page 110. Then see the instructions for creating and assigning secure sessions in the *Management and Security Server Administrator Guide*, in the *Host Access Management and Security Server Documentation*.



Connect using... **Description**

Pass Through

When configured as a Pass Through Proxy, the Security Proxy passes data to the destination host without regard to content (that is, it ignores any SSL handshaking data) and does not provide client/server authentication or encryption. If SSL is used in this mode, the SSL session is created between the client and destination host and encrypted data simply passes through the Security Proxy.

You can secure data traffic using SSL between the client and the destination host by enabling SSL user authentication on the destination host. When using a Pass Through proxy, client authorization is not an option.

Before you create a connection, review the "Requirements for connections through the Security Proxy" on page 109 and "Getting to the Reflection Desktop Security Settings" on page 110.

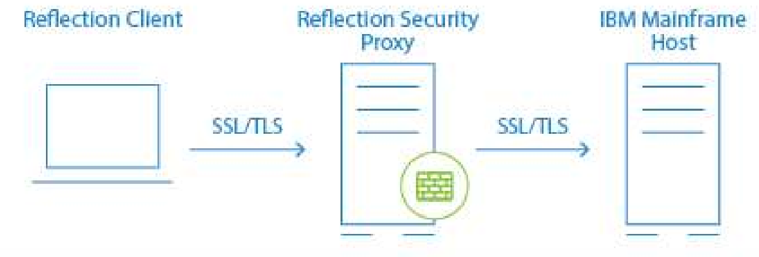
For instructions that show how to configure the Security Proxy, see the *Management and Security Server Administrator Guide*, in the *Host Access Management and Security Server Documentation*.

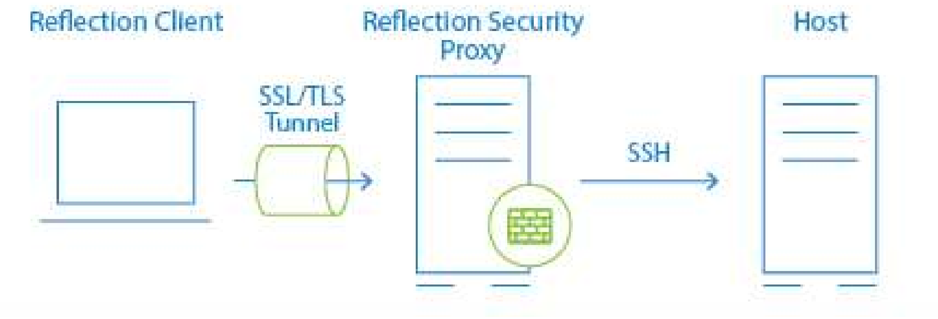
NOTE: If you want to establish an SSL-secured connection between Reflection and the destination host using the Security Proxy in Pass Through mode, you may need to unselect **Host name must match certificate** or, preferably, add the Security Proxy as the Subject Alternate name in the host server certificate.

To create a Reflection Desktop session that connects to the Security Proxy, see the instructions for creating secure sessions in the *Management and Security Server Administrator Guide*.

End-to-End SSL/TLS Security

This option, available for 3270 sessions only, combines user authorization with SSL/TLS security for the entire connection. Single sign-on capability using the IBM Express Logon Feature (page 119) is also supported, provided the host supports SSL/TLS. See Connect using End-to-End Security in 3270 Sessions.



Connect using...	Description
End-to-End SSH Security	In a standard configuration for a secure Reflection session, the connection between the client and security proxy server is encrypted using SSL/TLS, but the connection between the security proxy and the host uses unencrypted Telnet. By sending an SSH-encrypted connection through the security proxy tunnel, you can configure a secure Reflection session so that the entire communication path is encrypted from the client, through the proxy server, and on to the host. See "Connect using End-to-End Encryption in VT SSH Sessions" on page 111  Reflection Client SSL/TLS Tunnel Reflection Security Proxy SSH Host

Requirements and Other Information for Using the Security Proxy

Requirements for connections through the Security Proxy

Requirements for connections through the Security Proxy include:

- SSL/TLS must be enabled on the hosts to which you want to establish SSL/TLS connections. See the documentation included with the hosts for instructions.
- A centralized management server with the Security Proxy Add-On installed must be available. The Security Proxy must be configured to require client authorization. (It can optionally be configured to require client authentication. For client authentication, you can use a single certificate or two separate client certificates on each server (Security Proxy and destination host).
- You may need multiple digital certificates to successfully establish the SSL/TLS sessions between the client and the Security Proxy, and the client and the destination host.

Getting to the Reflection Desktop Security Settings

This section includes references to the *Management and Security Server Administrator Guide* in the *Host Access Management and Security Server Documentation*. The procedures referenced in this guide does not include product-specific information for setting up SSL/TLS connections. When you create a session in Reflection Desktop, use the following steps to go to these security settings:

1. Enter the **Host name** and **Port**.
2. Select **Configure additional settings** to open the Settings window and then choose **Set Up Connection Security**.
3. In the **Configure Advanced Connection Settings** dialog box, click **Security Settings**.

About Certificates

Server Certificates

Destination SSL/TLS hosts and Security Proxy servers typically have server certificates already installed. Each of these server certificates must be trusted by the client. The client will trust a server certificate if:

- It is signed by the certificate authority that is trusted by the client, or
- It is self-signed and imported into the trusted root certificate store where Reflection can find it.

To use a single server certificate for both the destination host and the Security Proxy, do one of the following:

- In the Reflection session, unselect the **Verify Server Identity**check box on the **Connection Editor**dialog box.
- (Recommended) Create a certificate that uses the destination host address for the **Subject Common Name** and the Security Proxy address for the **Subject Alternative Name**.

Client certificates

Certificates used for client authentication must be signed by a certificate authority that is trusted by both the Security Proxy and the destination host’s SSL server.

Express Logon also requires that the client certificate used to authenticate on the TN3270 server be registered with RACF. (For details, see the documentation that came with the 3270 server.)

For more details on configuring SSL and creating certificates on the host, see Technical Note 1760 (https://support.microfocus.com/kb/doc.php?id=7022182)

Connect using End-to-End Encryption in 3270 SSL/TLS Sessions

End-to-end encryption tunnels a direct SSL/TLS connection to the host, while still connecting through the Security Proxy Server. These connections require two certificates and SSL/TLS handshakes — one for the client/proxy server connection and another for the client/host connection. Without end-to-end encryption, only data between the client and proxy server is encrypted.

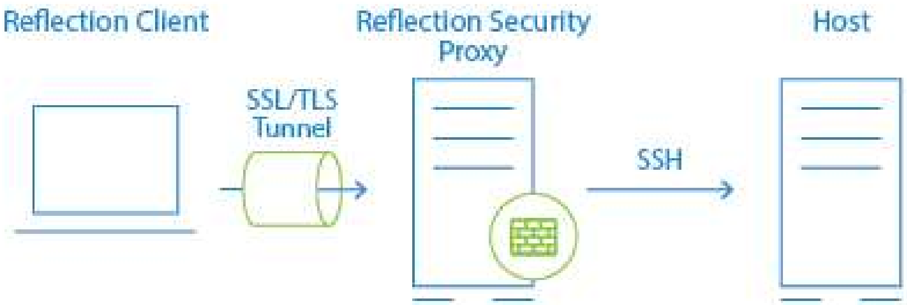
Before you create a connection, review the “Requirements for connections through the Security Proxy” on page 109 and “Getting to the Reflection Desktop Security Settings” on page 110.

To configure a session with end-to-end encryption, see the instructions for creating secure sessions in the *Management and Security Server Administrator Guide* in the *Host Access Management and Security Server Documentation*.

NOTE: You can modify the level of security by adjusting the SSL protocol version and encryption key-strength setting. Click **PKI Manager** to add the Certificate Revocation List (CRL) and Online Certificate Status protocols (OCSP) to certificate validation.

Connect using End-to-End Encryption in VT SSH Sessions

You can configure a Reflection Desktop session to send an SSH-encrypted connection through the Security Proxy Server.



In a standard Administrative Console configuration for a secure Reflection session, the connection between the client and security proxy server is encrypted using SSL/TLS, but the connection between the security proxy and the host uses unencrypted Telnet. By sending an SSH-encrypted connection through the security proxy tunnel, you can configure a secure Reflection session so that the entire communication path is encrypted from the client, through the proxy server, and on to the host.This feature has the following advantages:

- Encryption is used for the entire connection.
- The IP addresses and names of your secure hosts are not exposed outside of the internal network.
- Only clients with a valid authorization token can launch a secure session.

- The authorization token contains connection information. This enables the security proxy to send all secure host connections through a single port, eliminating the need to open multiple firewall ports.
- All settings required for a connection (such as the trusted certificate, the personal certificate, user keys, and host keys) reside on the Administrative Console and are downloaded to users' workstations when they start sessions.

You can set up this configuration using the Reflection VT Terminal type (used for UNIX and OpenVMS sessions).

Prerequisites

To make these SSH connections through the Security Proxy, you must have the following:

- The host must have an SSH server installed.
- Reflection Desktop v16 or Reflection 2014 R1 SP1 must be installed locally on your workstation. Note: Earlier versions and other products (such as Reflection 2014 R1, Reflection 2011, Reflection 14.x, or Extra!) are not supported.
- You must have access to the centralized management server Security Proxy and Administrative Console.
- Host Sessions must be opened from the Administrative Console or the Links List page.
- Sessions must be configured from the Administrative Console in the centralized management server.
- The Security Proxy must be running with Client authorization enabled.

NOTE: Management and Security Server is not licensed for connections from Reflection Desktop clients. You must have a Management and Security Server activation file installed to configure Reflection Desktop sessions.

To connect your VT session SSH connection through the Security Proxy Add-On

1. Follow the instructions for creating a new session in the *Management and Security Server Administrator Guide* in the *Host Access Management and Security Server Documentation* to create a new VT session.
2. When you configure the session, select **Secure Shell** for the connection type.
3. Enter the host name and user name (optional; users are otherwise prompted when they connect). Then select **Configure additional settings**and click **OK** to open the Settings dialog box.
4. Under **Host Connection**, click **Set up Connection Security**.
5. In the Reflection Secure Shell Settings dialog box, on the Reflection Security Proxy tab, select **Use Security Proxy**, and then choose a Security proxy and a Proxy port.

   NOTE: The Destination host values you entered in step 6 should be entered automatically here. If you don't see them, select the Security proxy name from the drop-down list to populate these fields.

   ---

6. Configure SSH connection settings such as the trusted certificate, the personal certificate, user keys, and host keys as required for your connection. For more information about configuring your SSH-specific settings, refer to the Reflection Help topic Reflection Secure Shell Settings Dialog Box.

7. Click **OK** to close the open dialog boxes and initiate the connection. Select **Always** to import the host key for these sessions.

   NOTE: If you do not want to include the user name in the configuration, cancel the connection. If you cancel, you will be unable to import the host key for the session.

8. Save the session. When prompted, choose to send the settings for this session to the Administrative Console, and then exit the Reflection workspace.

   All the files required for your configuration are uploaded to the Administrative Console. When a user launches the session, these files are downloaded to their workstation so that Reflection has access to all configuration data required to establish a connection.

NOTE: All non-default SSH settings required to establish a connection are saved in three files:

- The `sessionname.rssh` file contains the public key (if public key authorization is used), the host key (if a host key is accepted while in administrative mode), and the settings normally stored in both the `pki_config` file and the `config` file. It also includes all SSL/TLS settings such as the TLS version, cipher suites, and applicable proxy data.
- The `sessionname.ps` file stores any personal certificates included for the connection.
- The `sessionname.ts` file includes any trust certificates.

When you send settings for the session to the Administrative Console, these files are uploaded along with the session document file.

Set Up Terminal ID Management for Reflection Desktop Sessions

The Terminal ID Management Add-On configures and monitors a pool of resource IDs that a client can use to establish a host session, thereby eliminating the need for administrators to create configurations for every client. Depending on the type of terminal or printer, these resource IDs may represent addresses or identifiers as required.You can use ID Manager with the following types of Reflection terminals and printers:

- IBM 3270 Terminal
- IBM 3270 Printer
- IBM 5250 Terminal
- IBM 5250 Printer

NOTE: The Terminal ID Management Add-On requires the base installation of Host Access Management and Security Server. It is not included with the Management and Security Server license. To activate this product, you must purchase a separate license.

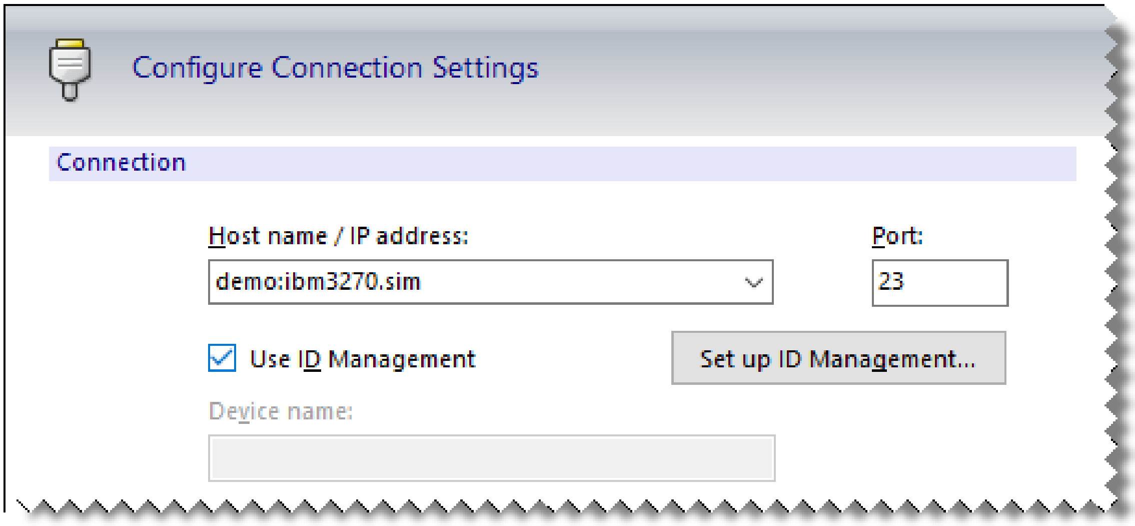
Prerequisites

To use ID Manager, you need access to a Management and Security Server with the Terminal ID Management Add-On configured.

For instructions that show how to configure ID Manager, see the *Management and Security Server Installation Guide* in the *Host Access Management and Security Server Documentation*:

To create a terminal session that accesses IDs from the Terminal ID Management Server every time it connects

1. Make sure that you have the following information for the Terminal ID Management Server:
   - The complete URL. For example, `http://server.name/tidm`, where `tidm` is typically case sensitive, but `server.name` is not.
   - The parameters required by the Terminal ID Management Server to allocate an ID, such as a pool name).
2. Open the Document Settings dialog box by going to **File** > **Settings** > **Document Settings**.Under **Host Connection**, select **Configure Connection Settings**.Under **Connection**, select **Use ID Management** and then click **Set up ID Management**.



Configure Connection Settings

Connection

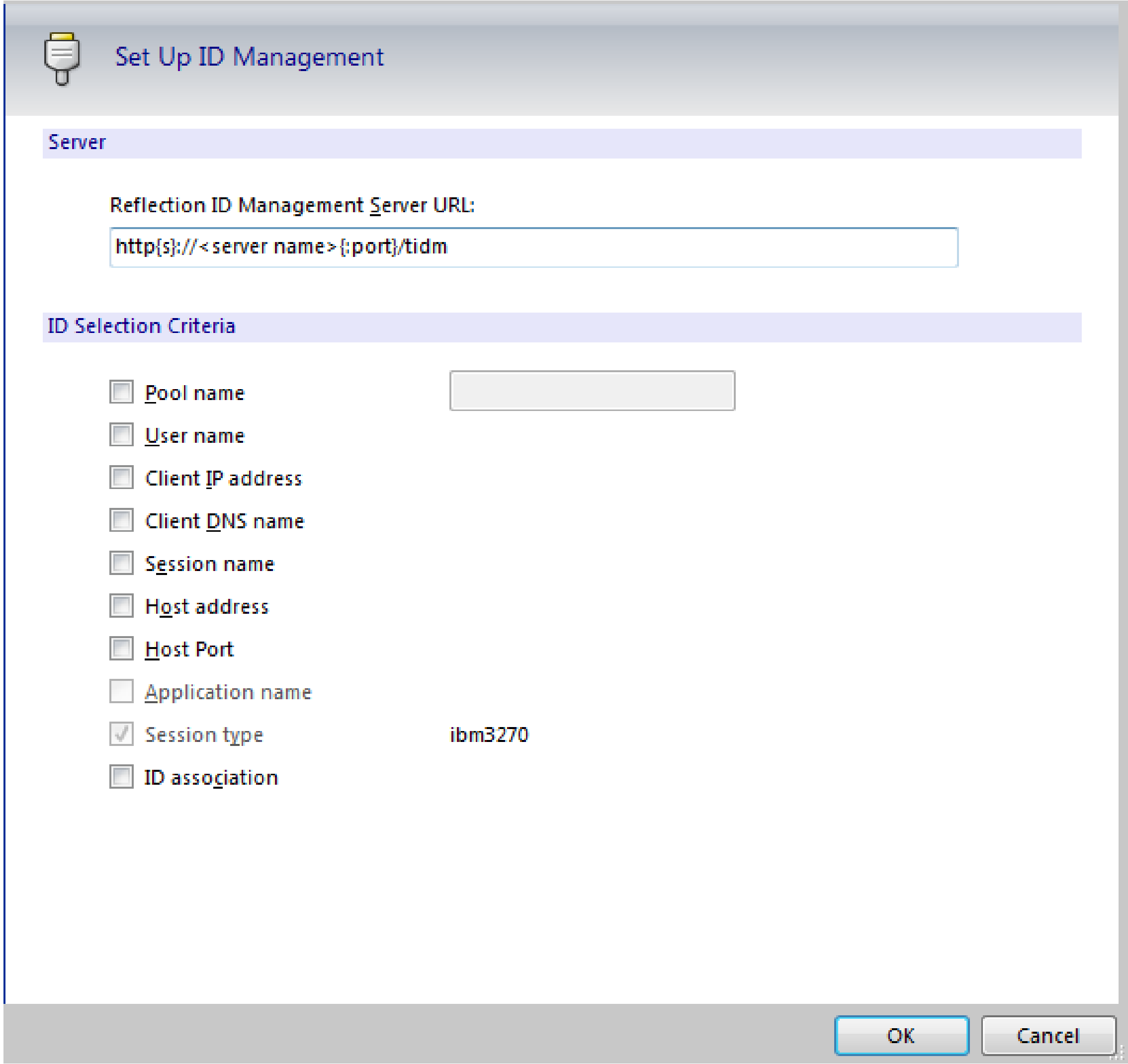
Host name / IP address: demo:ibm3270.sim

Port: 23

☑ Use ID Management

Device name:

3. Under Server, enter the **Reflection ID Management Server URL**.



Set Up ID Management

Server

Reflection ID Management Server URL:

http{s}://<server name>{:port}/tidm

ID Selection Criteria

- □ Pool name
- □ User name
- □ Client IP address
- □ Client DNS name
- □ Session name
- □ Host address
- □ Host Port
- □ Application name
- ☑ Session type ibm3270
- □ ID association

4 Select the **ID Selection Criteria**, such as **Pool name**.

To configure a printer session document to access IDs from the Terminal ID Management Server every time it connects

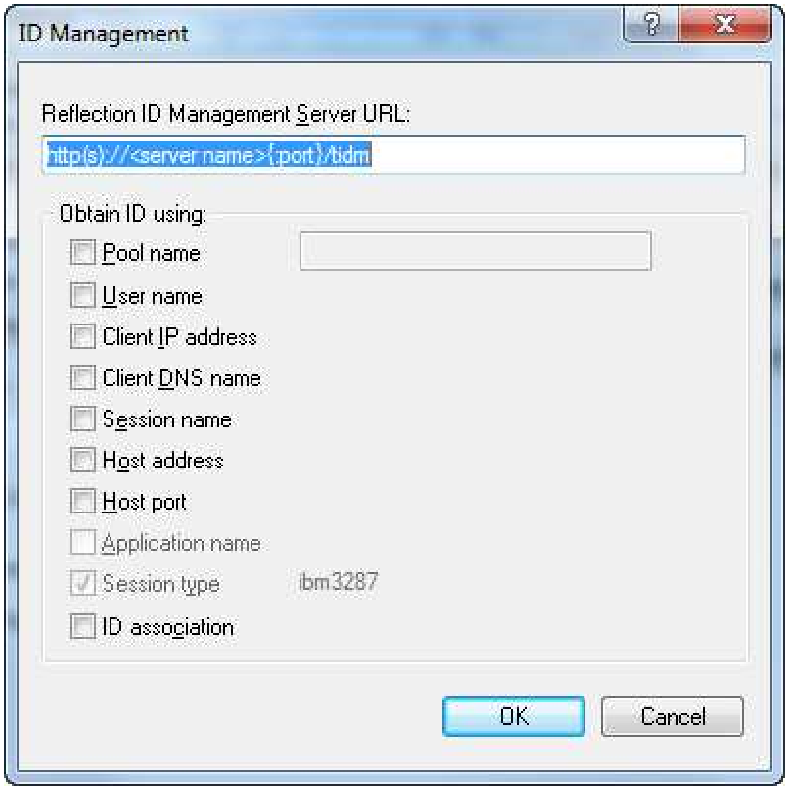
1 Make sure that you have the following information for the Terminal ID Management Server:

- The complete URL. For example, http://server.name/tidm, where tidm is typically case sensitive, but server.name is not.
- The parameters required by the Terminal ID Management Server to allocate an ID, such as a pool name).

2 Open a Reflection Printer Session Document File.

3 From the **Connection** menu, choose **Session Setup**.

4 Under **Transport**, select **Use ID Management** and then click **Set up ID Management**.



ID Management

Reflection ID Management Server URL:

http(s)://<server name>{:port}/tidm

Obtain ID using:

- □ Pool name
- □ User name
- □ Client IP address
- □ Client DNS name
- □ Session name
- □ Host address
- □ Host port
- □ Application name
- ☑ Session type ibm3287
- □ ID association

5 In the Set up ID Management dialog box, enter the **Reflection ID Management Server URL**.

6 Select the **ID Selection Criteria**, such as **Pool name**.

Set up Automated Sign-On for Mainframe Sessions

Using the Automated Sign-On for Mainframe Add-On, you can enable a user to authenticate to a front-end system using a modern form of authentication (such as a smart card, certificate, LDAP password, Kerberos, etc.) and then be automatically logged on to a z/OS mainframe application.

NOTE: The Automated Sign-On for Mainframe Add-on requires the base installation of the Management and Security Server (MSS), which provides the Administrative Server. This add-on is not included with the Management and Security Server license. To activate this product, you must purchase a separate license.

Automated Sign-On solves problems associated with credentials typically required for mainframe applications. Mainframe applications prompt for traditional credentials (a user name and password) and are typically hard-coded to accept a maximum of 8 characters for these credentials. Changing the password to match the user enterprise password is often not practical because of the mainframe limits on password character length and coordination of password changes. Because of this limitation, logging on to mainframe applications requires an identify that is separate from the user enterprise identity.

This add-on solves this problem by providing middleware that maps the user enterprise identity to the user mainframe identity. When using a Reflection session configured to use Automated Sign-On, the user authenticates to the front-end system using a modern authentication method. After authentication through the front-end system, the user is automatically logged into the host application.

Implementing Automated Sign-On for Mainframe

To implement Automated Sign On, you'll need to configure the Administrative Server, the Reflection emulation client session, and the z/OS mainframe as shown in the *Automated Sign-On for Mainframe Administrator Guide* in the *Host Access Management and Security Server Documentation*.

The process for setting up automated sign-on depends on your environment

Process for Centrally Managed Environments

If your environment is set up to centrally manage and control all session document files on an MSS server and you do not allow users to create their own session documents, use the following instructions in the *Automated Sign-On for Mainframe Administrator Guide* in the *Host Access Management and Security Server Documentation*.

- *Reflection or InfoConnect Desktop - Managed Sessions* in the *Enable your emulator for automated sign-on* section

Process for Unmanaged Environments

If your environment is not set up to centrally manage session document files with MSS and you want to use MSS only for automated sign-on, use the following instructions in the *Automated Sign-On for Mainframe Administrator Guid*e in the *Host Access Management and Security Server Documentation*.

- *Reflection or InfoConnect Desktop - Workspace Automated Sign-on* in the *Enable your emulator for automated sign-on* section

Glossary

Reflection database. The Reflection database (`ic32.cfg`) contains connection settings information for ALC, T27, and UTS terminal sessions. The database contains information about all the Reflection packages, path templates and libraries that have been installed, as well the paths that have been created. The Reflection packages, path templates and libraries are included based on which product features (emulations and transports) are installed.

Reflection global application data folder. Settings here apply to all users of the system. The location is version-specific: `\ProgramData\Micro Focus\Reflection\Desktop\v16.2`.

Reflection global ssh data folder. Reflection stores global Secure Shell information in the Windows common application data folder. The default is \ `ProgramData\Micro Focus\Reflection`.

Reflection program folder. The default on English language systems is `C:\Program Files (x86)\Micro Focus\Reflection` on 64-bit systems and `C:\Program Files\Micro Focus\Reflection` on 32-bit systems.

Reflection user application data folder. The default is `\Users\` *`username`* `\AppData\Roaming\Micro Focus\Reflection\Desktop\v16.2`.

Reflection user data folder. This folder location is configurable using the **Data Location** tab during installation. The default is `C:\Users\` *`username`*`\Documents\Micro Focus\Reflection`.

Reflection user ssh folder. Reflection stores Secure Shell information for individual users in the following location in the Windows personal documents folder. The default is `\Users\` *`username`* `\Documents`\`Micro Focus\Reflection\.ssh`.

AppDataFolder property. The full path of the Roaming folder for the current user. The default is `C:\Users\username\AppData\Roaming\`.

authentication. The process of reliably determining the identity of a communicating party. Identity can be proven by something you know (such as a password), something you have (such as a private key or token), or something intrinsic about you (such as a fingerprint).

Auto Expand. Use the Auto Expand feature to add acronyms or shortcuts for long words, phrases, or complex repeat commands. The shortcut, when typed and followed by the Spacebar, automatically expands to the full word or phrase.

cipher. A cipher is an encryption algorithm. The cipher you select determines which mathematical algorithm is used to obscure the data being sent after a successful Secure Shell connection has been established.

Client authentication. Client authentication (also referred to as *user authentication*) requires users to prove their identity using digital certificates (the default setting for the Reflection Security Proxy).

Client authentication is typically required when an SSL session is first established. It will also be required by a TN 3270 server if the user is using the Express Logon Feature provided by some mainframe systems.

Client authorization. Used in connections secured by the Reflection Security Gateway to ensure that access to host systems is approved before the connection can proceed.

When a user logs into the Reflection Security Gateway, he or she only has access to terminal session files and other features for which he has been explicitly authorized to use.

CommonAppDataFolder property. The full path to application data for all users. The default is `C:\ProgramData`.

CRL (Certificate Revocation List). A digitally signed list of certificates that have been revoked by the Certification Authority. Certificates identified in a CRL are no longer valid.

digital certificate. An integral part of a PKI (Public Key Infrastructure). Digital certificates (also called X.509 certificates) are issued by a certificate authority (CA), which ensures the validity of the information in the certificate. Each certificate contains identifying information about the certificate owner, a copy of the certificate owner's public key (used for encrypting and decrypting messages and digital signatures), and a digital signature (generated by the CA based on the certificate contents). The digital signature is used by a recipient to verify that the certificate has not been tampered with and can be trusted.

digital signature. Used to confirm the authenticity and integrity of a transmitted message. Typically, the sender holds the private key of a public/private key pair and the recipient holds the public key. To create the signature, the sender computes a hash from the message, and then encrypts this value with its private key. The recipient decrypts the signature using the sender's public key, and independently computes the hash of the received message. If the decrypted and calculated values match, the recipient trusts that the sender holds the private key, and that the message has not been altered in transit.

Express Logon Feature (ELF). Also referred to as *single sign-on (SSO),* express logon is an IBM mainframe feature that lets users log on and connect to the host without entering a user ID and password each time. Express Logon authenticates the user on the mainframe by using her SSL client certificate in lieu of entering a user ID and password.

FCC. Field Control Character. A UTS terminal field attribute.

hash. Also called a message digest, a hash or hash value is a fixed-length number generated from variable-length digital data. The hash is substantially smaller than the original data, and is generated by a formula in such a way that it is statistically unlikely that some other data will produce the same hash value.

KDC (Key Distribution Center). The security server that maintains the database of principal information, uses the information in the database to authenticate users, and controls access to kerberized services in a realm.

keyboard map. A keyboard map is a configuration file that allows you to use your PC keyboard as a host terminal keyboard. Keyboard maps also include definitions for keyboard shortcuts.

OCSP (Online Certificate Status Protocol). A protocol (using the HTTP transport) that can be used as an alternative to CRL checking to confirm whether a certificate is valid. An OCSP responder responds to certificate status requests with one of three digitally signed responses: "good", "revoked", and "unknown". Using OCSP removes the need for servers and/or clients to retrieve and sort through large CRLs.

passphrase. A passphrase is similar to a password, except it can be a phrase with a series of words, punctuation, numbers, white space, or any string of characters. Passphrases improve security by limiting access to secure objects, such as private keys and/or a key agent.

PCI DSS. PCI DSS (Payment Card Industry Data Security Standard) is a worldwide standard comprising technology requirements and process requirements designed to prevent fraud and is published by PCI Security Standards Council, LLC. All companies who handle credit cards are likely to be subject to this standard.

PersonalFolder property. The full path to the `Documents` folder for the current user. The default is `C:\Users\username\Documents`.

port forwarding. A way to redirect unsecured traffic through a secure SSH tunnel. Two types of port forwarding are available: local and remote. Local (also called outgoing) port forwarding sends outgoing data sent from a specified local port through the secure channel to a specified remote port. You can configure a client application to exchange data securely with a server by configuring the client to connect to the redirected port instead of directly to the computer running the associated server. Remote (also called incoming) port forwarding sends incoming data from a specified remote port through the secure channel to a specified local port.

product installation folder. The default on English language systems is C:\Program Files (x86)\Micro Focus\Reflection on 64-bit systems and C:\Program Files\Micro Focus\Reflection on 32-bit systems.

public key/private key. Public keys and private keys are pairs of cryptographic keys that are used to encrypt or decrypt data. Data encrypted with the public key can only be decrypted with the private key; and data encrypted with the private key can only be decrypted with the public key.

Screen History. Screen History creates recordings of host screens as you navigate to them. (VT screens are not recorded automatically; they can be recorded using manual capture.) You can view and/or verify the information from those screens, and send multiple host screens to Microsoft Word, PowerPoint, and Outlook (Email Message and Note only), if they are installed on your computer.

socket. The combination of a host name (IP address or DNS name) and a port number. This creates a unique identifier that a client application uses as an end point of communications.

trusted host. A trusted host is one for which you hold the public key.

trusted locations. A trusted location is a directory that's designated as a secure source for opening files. By default, Reflection allows you to open documents only in directories specified as trusted locations using the Specify Trusted Locations dialog box.

Windows common application data folder. The application data folder is hidden by default. The default is \ `ProgramData\`.

Windows personal documents folder. The default on English systems is `\Users\` *`username`* `\Documents\`.

Workspace Menu. The Workspace menu contains layout options, application and document settings, and a list of recent documents. It is accessed by clicking the **File**menu or Reflection button ⊕ (when using the Ribbon user interface).