Evaluating Reflection for Secure IT

Micro Focus®

Reflection for Secure IT®

Reflection for Secure IT…
…Secure Shell backed by Service

You're ready to get serious about security, and Reflection for Secure IT can help. By replacing nonsecure Telnet and FTP with a reliable encrypted alternative, administrators can access any TCP/IP-based application through a secure transmission tunnel, and safely transmit sensitive data and manage remote servers — even over untrusted networks.

Why use Secure Shell?

The Secure Shell (SSH) protocol is a flexible, dependable way to guarantee the safety of your data in motion. The following vital features provide reliable safeguards that are increasingly important in today's world:

- Server authentication ensures that your clients communicate with the correct server.
- Client authentication ensures that only authorized client users can connect to your server.
- Data encryption assures that data in transit is indecipherable — the client and server establish a unique key for each Secure Shell session, and this key is required to decipher the data.
- Data integrity checking verifies that your data has not been altered during transit.
- Port forwarding protects TCP/IP communications sent over an untrusted network.

Why use Reflection for Secure IT?

To build a security solution you can trust, you need to work with software and people you can trust. With Reflection for Secure IT and Attachmate you can count on:

- Rock solid, supported software.
  Our developers use the latest techniques in secure software design to ensure that Reflection for Secure IT products are optimized for stability and security.
- Cross-platform support.
  Reflection for Secure IT clients and servers are available for Windows and UNIX operating systems, and run on both 32-bit and 64-bit hardware.
- Responsive technical support.
  Our technical support experts work closely with you and with our development team to make sure that your questions and concerns are answered quickly and correctly.
- Security updates.
  Our security specialists watch for potential security vulnerabilities. In the event that we learn of a new vulnerability, we keep you informed and make it our top priority to resolve your security concerns.
- Comprehensive documentation.
  Our documentation team is committed to providing you with complete, technically accurate information about all facets of our products.

See for yourself!

The evaluation software is a fully-functional, time-limited copy of Reflection for Secure IT Windows Client. If you haven't yet downloaded the evaluation software, go to http://www.attachmate.com/Evals/rsit/rsit-eval.htm and fill out the evaluation request form. Shortly after you submit the form, you'll receive an e-mail message with a link to the evaluation download page.

1. From the evaluation download page, click the **Download Now** link, and run the program.
2. Select a location for the installer files and click **Next**.
   The files are extracted to the specified location and the Reflection installer starts automatically.

Before you get started…

- You'll need access to a running Secure Shell server. You can use any server that is already available to you, or you can download and install an evaluation copy of the Reflection for Secure IT Windows or UNIX server.
- Supplement this document by viewing the Reflection for Secure IT Windows Client 8.0 user guide. This guide is installed with your evaluation copy.

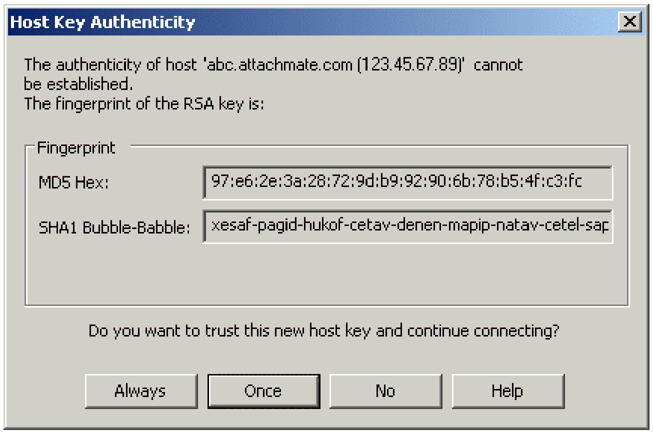
Getting started with Reflection for Secure IT

This guide walks you through a few key tasks that are familiar to typical Reflection for Secure IT users. After you try the procedures, you can review the ideas under "Do more..." There you'll find suggestions that can help you use the rich feature set in Reflection to maximize the highest levels of security, while saving time and money.

Getting connected

As soon as you've completed your installation, Reflection for Secure IT is ready to make a secure connection to your host. The following procedure will get you connected using default security settings:

1. Start Reflection for Secure IT Client (**Start** > **Programs** > **Micro Focus Reflection** > **SSH Client**).
2. From the toolbar, click the **Connect/ Disconnect** button:
3. From the **Connection Setup** dialog box, enter your host name (or IP address) and user name, and then click **Connect**.
   Note: The first time you connect to your host, you'll see a **Host Key Authenticity** dialog box like the one shown here:



Host Key Authenticity

The authenticity of host 'abc.attachmate.com (123.45.67.89)' cannot be established.
The fingerprint of the RSA key is:

Fingerprint

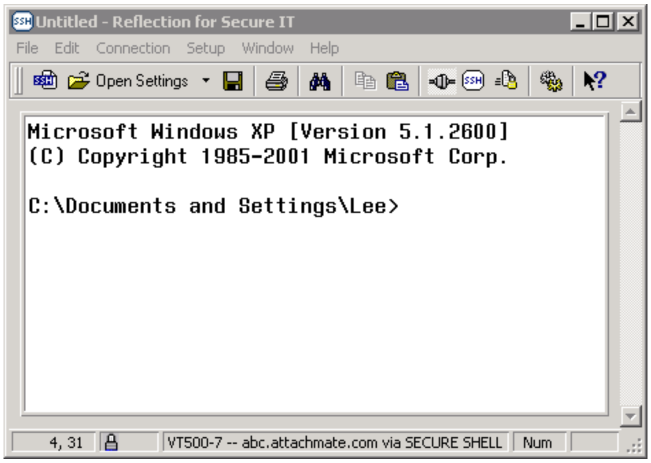
MD5 Hex: 97:e6:2e:3a:28:72:9d:b9:92:90:6b:78:b5:4f:c3:fc

SHA1 Bubble-Babble: xesaf-pagid-hukof-cetav-denen-mapip-natav-cetel-sap

Do you want to trust this new host key and continue connecting?

If you're used to Telnet connections, this prompt will be new to you. The fingerprint shown here (in two formats) can be used to authenticate your server. You can confirm the validity of the host key by contacting the system administrator of the server.

4. Click **Always** to add this host key to your known hosts list.
   Once the host key is added to this list, the client can authenticate the server without requiring user confirmation, so you won't see the unknown host prompt again when connecting to this host.
5. Enter your password for this host, and then click **OK**.
   After you've authenticated successfully, the terminal window provides a shell session that you can use to execute commands on the server — either UNIX shell commands or Windows DOS commands depending on your server.



Untitled - Reflection for Secure IT
File
Edit
Connection
Setup
Window
Help
Open Settings

```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Lee>
```

4, 31
VT500-7 -- abc.attachmate.com via SECURE SHELL
Num

A successful connection to a Windows server

6. Click **File** > **Save** to save a settings file with this session configuration.

To connect to the same host again, just open your saved settings file. Reflection for Secure IT automatically initiates the connection.

Do more...

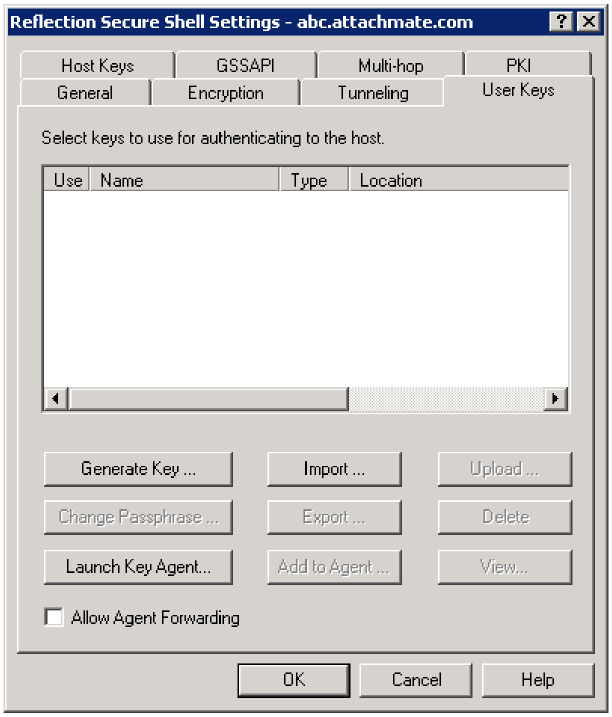
Now that you've made your first connection, you're ready to do more. Here are some ideas:

- **Customize the display**

  Go to **Setup** > **Display** to customize screen colors, fonts, and other display options.
- **Get quick access to settings using the Configuration toolbar**
  Click the **Configure session settings** button:

- **Configure public key authentication**

  With Public key user authentication, the Secure Shell server uses a unique digital signature to authenticate the client user. To configure this, you create a private key on your workstation and upload the corresponding public key to the remote server. Public key authentication improves security because no authentication secret is ever sent over the network. (With password authentication, the encrypted password must be sent over the network each time the user authenticates.)
  To configure public key authentication, click the **Configure session setting**s toolbar button. In the left pane of the terminal window, click **User Keys** to open the tab shown in the following graphic. The *Reflection for Secure IT Windows Client User Guide* includes detailed procedures to help you configure your settings.



Reflection Secure Shell Settings - abc.attachmate.com

Select keys to use for authenticating to the host.

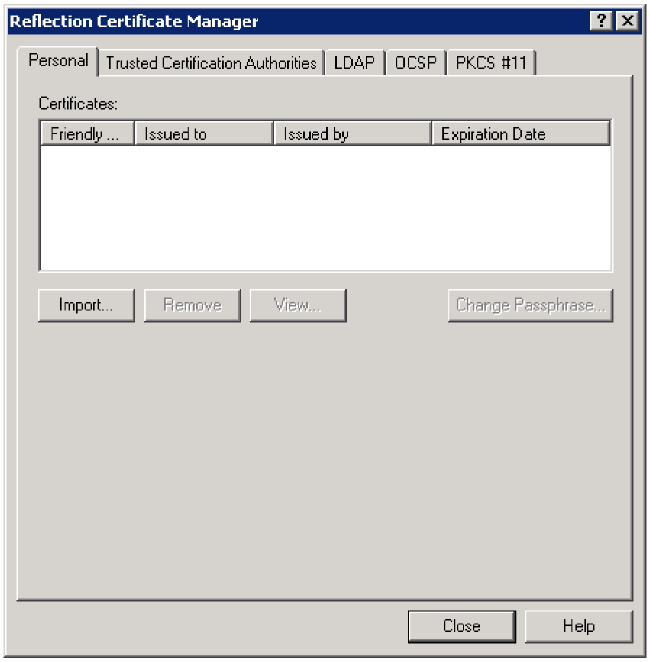
Configuring user keys

- **Easily upload keys to the server**

  Use the **Upload** button, shown in the dialog box above, to simplify uploading your user key to the server. Reflection automatically uploads keys using the correct location and format for your server.

- **Configure certificate authentication**

  Certificate authentication solves some of the problems presented by public key authentication. For example, when public keys are used for client authentication, each client public key must be uploaded to the server and the server needs to be configured to recognize that key. When certificate authentication is used, a single CA (Certificate Authority) root certificate can be used to authenticate multiple client users. By default, Reflection for Secure IT uses certificates that you install in a Reflection-specific certificate store. You can also configure Reflection to use certificates in the Microsoft certificate store in addition to those in the Reflection store. Using the Microsoft store enables you to use certificates that were installed with the operating system. Using the Reflection store enables you to enforce a higher level of security for your Reflection connections.
  To open the Reflection Certificate Manager, click the **Configure session setting**s toolbar button. In the left pane of the terminal window, click **PKI**, then click the **Reflection Certificate Manager** button.



Reflection Certificate Manager

The Reflection Certificate Manager

- **Configure Secure Shell settings for particular hosts or groups of hosts**

  Whether you want to share the same Secure Shell settings for multiple hosts or use different settings for particular hosts, you can use configuration schemes to store appropriate settings for each connection.

To create a new configuration scheme, open the **Connection Setup** dialog box (**Connection** > **Connection Setup**) and specify a host name. You can leave **SSH config scheme** blank (in this case, the scheme name defaults to the current host name), or specify a scheme name that describes the security settings you are configuring. Next, click **Security**, configure your Secure Shell settings, and then click **OK**. Any changes you make to your Secure Shell settings are saved to the client configuration file under the current scheme name. You can now apply this scheme to subsequent connections.

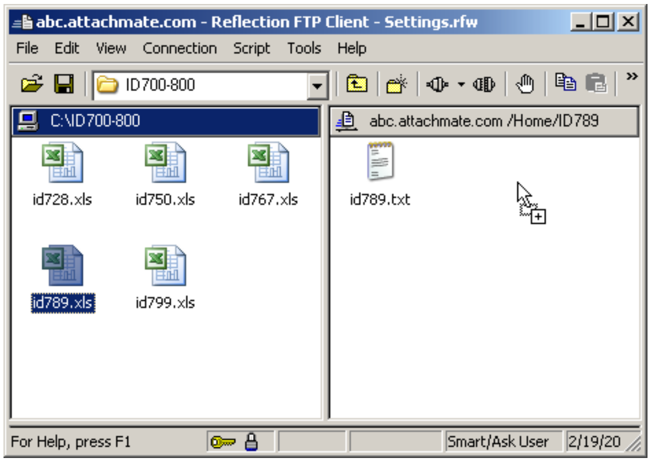
- **Save window layouts**
  If you frequently work with several terminal sessions at a time, or if you like to open an FTP client session automatically with each terminal session, you can use layouts to save your Reflection sessions together as a group. Open the Reflection settings files that you want to include in your layout, and then click **File** > **Layout**.
- **Work from the command line**
  Reflection for Secure IT client includes a complete set of command line utilities that you can use to establish connections, configure forwarding, transfer files, and manage public keys. The following commands are supported: **ssh**, **scp**, **sftp**, and **ssh-keygen**.

Transferring files securely

You can transfer files securely with the Reflection FTP Client using a simple drag-and-drop operation. You can drag individual files, multiple files, and entire folders. Let's try it:

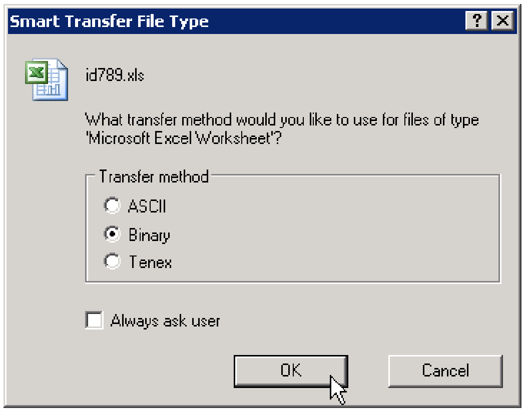
1. From the Windows **Start** menu, click **Micro Focus Reflection** > **FTP Client**.
   The **Connect to FTP Site** dialog box opens automatically.
2. Click **New**.
   The **Add FTP Site** wizard starts. The wizard is configured to create secure SFTP connections by default. For SFTP connections, your host needs to be running a Secure Shell server.
3. Step through the wizard, entering your host and user name when prompted.
4. On the last panel, you are asked if you want to connect to the FTP site. Select **Yes** (the default), and then click **Finish** to exit the wizard and make the connection.

   Note: If you are connecting to the same server you used for your terminal session, you won't see the **Host Key Authenticity** dialog box. If you are connecting to a different server, you'll need to add this server to your known hosts list.
5. Browse to locate the files or folders you want to transfer, and to the destination location for the transfer. To browse your local folders use the left pane. To browse the server directories, use the right pane.
6. Select the files or folders you want to transfer and drag them from the source location to your desired destination.



Using drag-and-drop to transfer from the PC to the server

Note: The client maintains a list of which transfer method to use for each file type. The first time you transfer a file with a file extension that's not on this list, you'll be given the option to set the transfer method as shown here:


Smart Transfer File Type

id789.xls

What transfer method would you like to use for files of type 'Microsoft Excel Worksheet'?

Transfer method
○ ASCII
● Binary
○ Tenex

□ Always ask user

Setting the file transfer type for .xls files

7. Click **File** > **Save** to save your changes to the settings file.

Your saved site will appear in the connection list each time you launch the FTP Client. Saving your settings also saves other changes you make during a session (such as setting a new transfer method).

Do more...

Now that you have transferred your first file, you're ready to take advantage of Reflection features that can help you simplify and automate your file transfer tasks. Here are some ideas to get you started:

- **Drag-and-drop entire folders**
  When you drag a folder, all the files within that folder are transferred.
- **Set site-specific home directories**
  To save time navigating to files and directories, you can configure site-specific startup directories. From the **Connect to FTP Site** dialog box click **Properties**, then go to the **Directories** tab. You can configure startup directories for both the PC and the server; however, your access to server directories may be limited by the server configuration.
- **Use the command window to see detailed command information**
  Click **View** > **Command Window** to see the communication sent between the client and server.
- **Use the script recorder to create scripts**
  You can easily automate transfers by recording and replaying transfer scripts. Use **Script** > **Start recording**.
- **Automate transfers using the scp and sftp command line utilities**
  You can use the **scp** and **sftp** command line utilities to create batch files for automating secure file transfer. Reference information for these utilities is available in the application help.
- **Use the Automation API to transfer files from other applications**
  Using the FTP Client Automation API, you can script FTP transfers from Visual Basic programs and other external applications, including Microsoft Office applications. For a complete reference, open the FTP Client application help (**Help** > **Help Topics**), and find "Help for the FTP Client Automation API" in the table of contents.

Securing communications using port forwarding

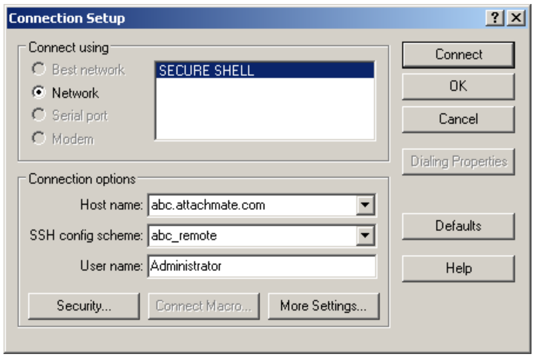
Port forwarding, also known as tunneling, is a powerful feature that redirects communications through the Secure Shell channel of an active session. When port forwarding is configured, all data sent to a specified port is redirected through the secure channel. This capability enables you to route all your traffic through one secured port in the firewall (similar to the way SOCKS works, except with encryption). The next exercise shows how you can use port forwarding to accomplish secure remote server administration.

Configure secure remote administration

One of the protocols that can be sent through a Secure Shell tunnel is RDP (Remote Desktop Protocol). RDP is the protocol underlying what Microsoft calls Remote Desktop in Windows Server 2003 and Terminal Services in Windows 2000 Server. It enables you to log on remotely to a Windows computer and work as if you were seated at the local console.

Remote Desktop requires opening another port in the firewall — typically port 3389 — which most security-conscious organizations are reluctant to do. Running RDP through a Secure Shell tunnel encrypts your communication and enables you to keep tighter control over which ports are open. Here's how it works:

1. Install your Secure Shell server on the target Windows server, and the Reflection for Secure IT client on your local workstation.
2. Confirm that forwarding is enabled on your server; the default for Reflection for Secure IT servers.
3. Enable Remote Desktop/Terminal Services on the Windows server, and confirm that your account is allowed remote access.
4. Launch the Reflection for Secure IT Client (**Start** > **Programs** > **Micro Focus Reflection** > **SSH Client**).
5. From the **Connection** menu, select **Connection Setup**, and then enter your host and user name. **Note:** You may also want to specify a value for **SSH config scheme**. Doing this enables you to select this scheme whenever you want to initiate a Remote Desktop connection, and use a different scheme for other Secure Shell connections to the same host.



Connection Setup

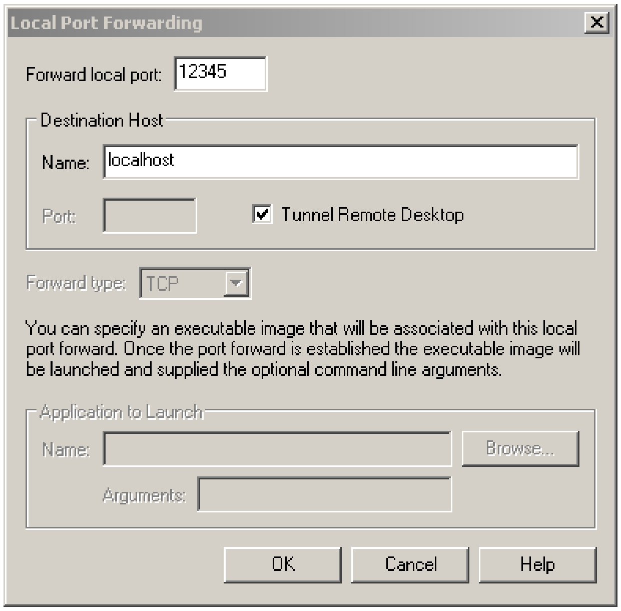
Connect using
○ Best network
● Network
○ Serial port
○ Modem

SECURE SHELL

Connection options
Host name: abc.attachmate.com
SSH config scheme: abc_remote
User name: Administrator

Configuring the connection to the Secure Shell server

6. Click **Security**.
7. From the **Tunneling** tab, under **Local Forwarding**, click **Add**.
   The **Local Port Forwarding** dialog box opens.



Local Port Forwarding

Forward local port: 12345

Name: localhost

☑ Tunnel Remote Desktop

Forward type: TCP

You can specify an executable image that will be associated with this local port forward. Once the port forward is established the executable image will be launched and supplied the optional command line arguments.

Configuring secure Remote Desktop forwarding

8. For **Forward local port**, enter an arbitrary port (using any available value greater than 1024). Data sent to this port will be forwarded through the secure tunnel to the server.
9. Under **Destination Host**, select **Tunnel Remote Desktop**.
   This setting configures the client to forward the RDP protocol using the specified port, and also to launch Remote Desktop automatically as soon as the Secure Shell connection is established.
10. Under **Destination Host**, set **Name** equal to localhost. This value specifies that the destination for the forwarded data is the same computer that is running the Secure Shell server.
11. Click **OK** to return to the **Connection Setup** dialog box.
12. Click **Connect** and log on to the Windows server using your credentials.
    A Remote Desktop/Terminal Services session starts automatically as soon as the Secure Shell session is established.
13. Log on to complete your secure remote session.
14. Click **File** > **Save** to save this session configuration.

Now you can open your saved settings file whenever you need to administer the remote server. Reflection automatically launches a secure remote session using your saved configuration.

Do more…

Port forwarding is a powerful and flexible feature that can help you secure your data in motion. Here are some ideas to get you started:

- **Forward data exchanged between any TCP/IP client and server**

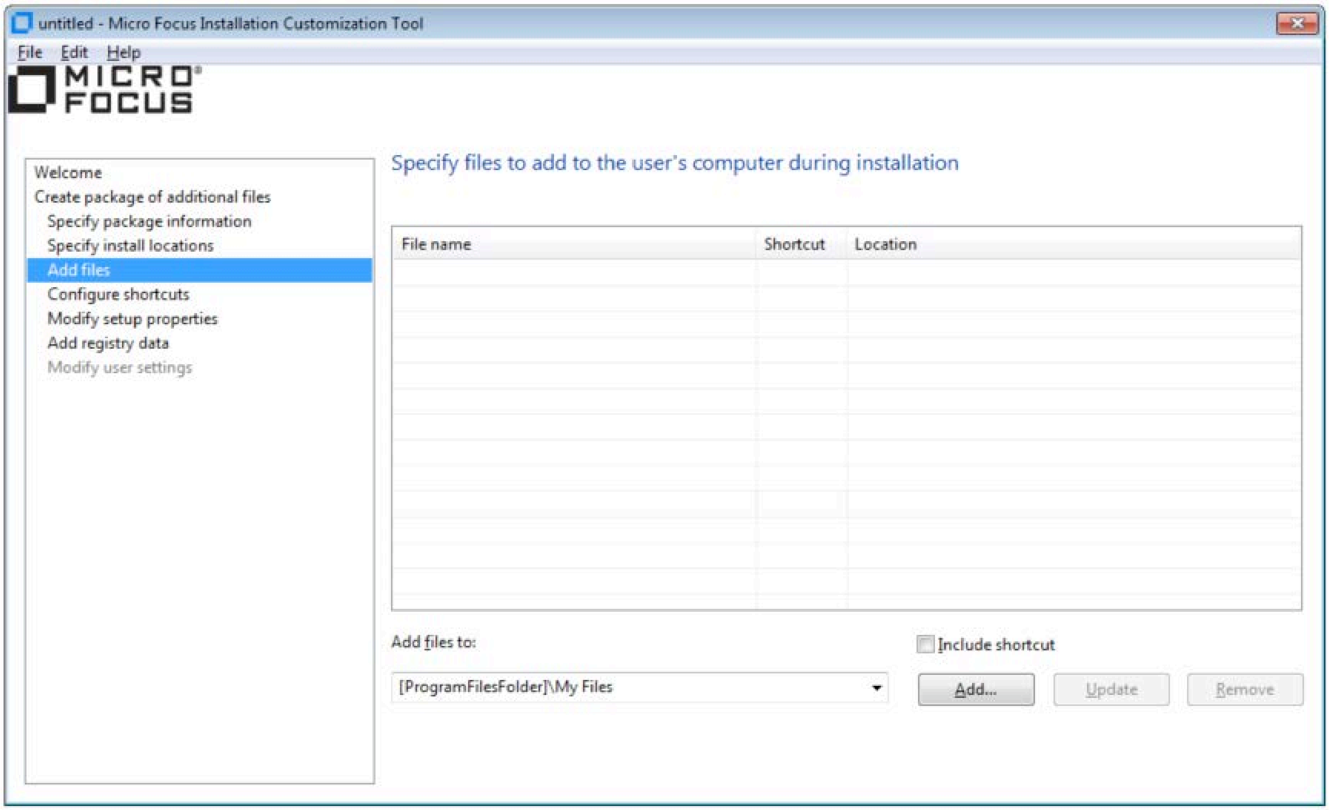
  You can secure the data exchanged between any client and server applications that use the TCP/IP protocol. This means that you can securely forward Telnet, HTTP, SMTP, POP, and IMAP communications over an untrusted network. Once you've configured a forwarded port, just set your TCP/IP application to connect to the forwarded port. For help setting up forwarding, see the *Reflection for Secure IT Windows Client User Guide*.

- **Use forwarding to secure FTP protocol transfers**

  By default, Reflection for Secure IT uses the SFTP protocol for secure file transfer. In some cases, you may prefer to use the FTP protocol, which supports additional command options. With Reflection for Secure IT, you can easily configure FTP protocol forwarding. When you do, your data is fully protected because both the command and data channels are forwarded through the secure tunnel. From the FTP Client **Connect to Site** dialog box select a site, and then go to **Security** > **Secure Shell** > **Tunnel FTP using port forwarding**.

- **Configure forwarding using the command line or the configuration file**

  Reflection for Secure IT offers flexible options for configuring forwarding. Besides configuring it from the user interface, you can configure forwarding using the **ssh** command line utility or by editing the client configuration file.



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

Add files to: [ProgramFilesFolder]\My Files

Using the Installation Customization Tool to install a known hosts file

Customizing the Installation

In addition to supporting easy default installations, the Attachmate Setup program includes the Attachmate Customization Tool, which enables administrators to customize Reflection for Secure IT installations.

- To run the Installation Customization Tool, type the following on the command line:

  setup.exe /admin

You can create a new Setup customization file for your product, or create a Companion installer that, for example, uses the **Add Files** feature of this tool to install a known hosts file to ensure that users connect securely to known servers without needing to respond to the unknown host prompt.

For more details about the Installation Customization Tool**,** see the *Reflection for Secure IT Windows Client User Guide***.**

For More Information on Reflection for Secure IT Windows Client

For more information about Reflection for Secure IT Windows Client, visit the Product Documentation site at: http://support.attachmate.com/manuals/rsit_win_client.html

For further assistance regarding evaluation software and product updates, visit our Technical Support site at http://support.attachmate.com/.

For additional office locations, partners, and resellers, visit our Web site at www.microfocus.com.