



# **Tectia<sup>®</sup> Client/Server 7.0 (Windows)**

## **Quick Start Guide**

07 April 2026

---

# Tectia® Client/Server 7.0 (Windows): Quick Start Guide

07 April 2026

Copyright © 1995–2026 SSH Communications Security Corporation

This software and documentation are protected by international copyright laws and treaties. All rights reserved.

ssh® and Tectia® are registered trademarks of SSH Communications Security Corporation in the United States and in certain other jurisdictions.

SSH and Tectia logos and names of products and services are trademarks of SSH Communications Security Corporation. Logos and names of products may be registered in certain jurisdictions.

All other names and marks are property of their respective owners.

No part of this publication may be reproduced, published, stored in an electronic database, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, for any purpose, without the prior written permission of SSH Communications Security Corporation.

THERE IS NO WARRANTY OF ANY KIND FOR THE ACCURACY, RELIABILITY OR USEFULNESS OF THIS INFORMATION EXCEPT AS REQUIRED BY APPLICABLE LAW OR EXPRESSLY AGREED IN WRITING.

For Open Source Software acknowledgements, see appendix *Open Source Software License Acknowledgements* in the *User Manual*.

SSH Communications Security Corporation  
Karvaamokuja 2D, FI-00380 Helsinki, Finland

---

# Table of Contents

<b>1. About This Document</b> .....	5
1.1. Reference Documents .....	5
1.2. Component Terminology .....	5
1.3. Documentation Conventions .....	7
1.3.1. Operating System Names .....	8
1.4. Customer Support .....	8
<b>2. Installation</b> .....	11
2.1. Preparing for Installation .....	11
2.1.1. Hardware and Disk Space Requirements .....	11
2.1.2. Upgrading Previously Installed Tectia Software .....	11
2.1.3. License File .....	13
2.1.4. Creating Operating System User Accounts .....	13
2.2. Installing Tectia Software .....	14
2.2.1. Installing Tectia Client on Windows .....	14
2.2.2. Installing Tectia Server (and Client) on Windows .....	15
2.2.3. Installation Complete .....	17
2.3. Removing Tectia Software .....	17
2.3.1. Removing Tectia Client and Server from Windows .....	18
<b>3. Connecting to Remote Server</b> .....	19
3.1. First Connection with Password .....	19
3.2. Creating Connection Profiles .....	22
3.2.1. Defining Connection Profile Settings .....	23
<b>4. Configuring Authentication Methods</b> .....	25
4.1. Server Authentication Methods .....	25
4.2. User Authentication with Passwords .....	25
4.3. User Authentication with Public Keys .....	26
4.3.1. Creating Keys with Public-Key Authentication Wizard .....	27
4.3.2. Uploading Public Keys Automatically .....	30
4.4. Setting up Non-interactive Authentication for Automatic Scripts .....	32
<b>5. Using Secure File Transfer</b> .....	33
5.1. Using SFTP on Tectia Client .....	33
5.1.1. Using SFTP on Command Line .....	33
5.1.2. Using PrivX Desktop File Transfer GUI .....	33

---

5.2. Configuring Tectia Server for Automated Secure File Transfer .....	34
5.2.1. Opening Tectia Server Configuration GUI .....	35
5.2.2. Enabling Public-Key Authentication .....	35
5.2.3. Settings for the Admin Group .....	36
5.2.4. Settings for the SFTP-users Group .....	40
5.2.5. Settings for the Rest of Users .....	44
5.3. Automated Secure File Transfer Script .....	46
<b>6. Using Secure Application Connectivity .....</b>	<b>49</b>
6.1. Defining Automatic Tunnels .....	50
6.1.1. Settings in Tectia Client .....	50
6.1.2. Settings in the Tunneled Application .....	52
Index .....	53

# Chapter 1 About This Document

This guide gives quick instructions for getting started with Tectia Client and Server. There are alternative client/server products for different platform architectures:

- Tectia Client/Server for AIX, Linux, Solaris, and Windows platforms
- Tectia Server for IBM z/OS for IBM mainframes

The instructions in this guide are intended for a system where Tectia Client is used to connect to Tectia Server, and both are running on the Windows operating system.

The purpose of this quick start guide is to help you in getting the Tectia client/server solution up and running with the default settings so that you can evaluate the product.

The target audience of this guide are system administrators and other professionals who need to evaluate Tectia products. To be able to use the information presented in this document, you should have system-administrator-level knowledge and know what Tectia Client and Server are meant for.

## 1.1 Reference Documents

The Tectia client/server solution is described and more advanced user instructions are given in the following product-specific manuals:

- *Tectia Client/Server Product Description* contains general information about the product, its architecture, main features, and the product structure.
- *Tectia Client User Manual* contains detailed instructions on installing, configuring and using Tectia Client.
- *Tectia Server Administrator Manual* contains detailed instructions on installing, configuring and using Tectia Server.

Instructions for evaluating Tectia Client and Server on Unix are available in a separate quick guide *Tectia Client/Server (Unix) Quick Start Guide*.

## 1.2 Component Terminology

The following terms are used throughout the documentation.

**client computer**

The computer from which the Secure Shell connection is initiated.

**Connection Broker**

The Connection Broker is a component included in Tectia Client, and in the Tectia Server for IBM z/OS client tools. Connection Broker handles all cryptographic operations and authentication-related tasks.

**host key pair**

A public-key pair used to identify a Secure Shell server. The private hostkey file is accessible only to the server. The public key file is distributed to users connecting to the server.

**remote host**

Refers to the other party of the connection, [client computer](#) or [server computer](#), depending on the viewpoint.

**Secure Shell client**

A client-side application that uses the Secure Shell version 2 protocol, for example **sshg3**, **sftpg3**, or **scpg3** of Tectia Client.

**Secure Shell server**

A server-side application that uses the Secure Shell version 2 protocol.

**server computer**

The computer on which the Secure Shell service is running and to which the Secure Shell client connects.

**SFTP server**

A server-side application that provides a secure file transfer service as a subsystem of the Secure Shell server.

**Tectia Client**

A software component installed on a workstation. Tectia Client provides secure interactive file transfer and terminal client functionality for remote users and system administrators to access and manage servers running Tectia Server or other applications using the Secure Shell protocol. It also supports (non-transparent) static tunneling.

**Tectia client/server solution**

The Tectia client/server solution consists of Tectia Client, Tectia Server, and Tectia Server for IBM z/OS (including the Tectia Server for IBM z/OS client tools).

**Tectia Connections Configuration GUI**

Tectia Client has a graphical user interface for configuring the connection settings to remote servers. The GUI is supported on Windows and Linux.

**PrivX Desktop File Transfer GUI**

PrivX Desktop includes a graphical user interface (GUI) for handling and performing file transfers interactively. The GUI is supported on Windows and Linux.

## Tectia Server

Tectia Server is a server-side component where Secure Shell clients connect to. There are two versions of the Tectia Server product available: *Tectia Server* for Linux, Unix and Windows platforms, and *Tectia Server for IBM z/OS*.

## Tectia Server for IBM z/OS

Tectia Server for IBM z/OS provides normal Secure Shell connections and supports the enhanced file transfer (EFT) features and transparent TCP tunneling on IBM mainframes.

## Tectia Server Configuration tool

Tectia Server has a graphical user interface that can be used to configure the server instead of editing the configuration file. The GUI is supported on Windows.

## transparent FTP tunneling

An FTP connection transparently encrypted and secured by a Secure Shell tunnel.

## transparent TCP tunneling

A TCP application connection transparently encrypted and secured by a Secure Shell tunnel.

## tunneled application

A TCP application secured by a Secure Shell connection.

## user key pair

A public-key pair used to identify a Secure Shell user. The private key file is accessible only to the user. The public key file is copied to the servers the user wants to connect to.

## 1.3 Documentation Conventions

The following typographical conventions are used in Tectia documentation:

**Table 1.1. Documentation conventions**

Convention	Usage	Example
<b>Bold</b>	Tools, menus, GUI elements and commands, command-line tools, strong emphasis	Click <b>Apply</b> or <b>OK</b> .
→	Series of menu selections	Select <b>File</b> → <b>Save</b>
Monospace	Command-line and configuration options, file names and directories, etc.	Refer to <code>readme.txt</code>
<i>Italics</i>	Reference to other documents or products, URLs, emphasis	See <i>Tectia Client User Manual</i>
<i>Monospace Italics</i>	Replaceable text or values	<code>rename <i>oldfile</i> <i>newfile</i></code>
#	In front of a command, # indicates that the command is run as a privileged user (root).	<code># rpm --install package.rpm</code>

Convention	Usage	Example
\$	In front of a command, \$ indicates that the command is run as a non-privileged user.	<code>\$ sshg3 user@host</code>
\	At the end of a line in a command, \ indicates that the command continues on the next line, but there was not space enough to show it on one line.	<code>\$ ssh-keygen-g3 -t rsa \ -F -c mykey</code>



## Note

A Note indicates neutral or positive information that emphasizes or supplements important points of the main text. Supplies information that may apply only in special cases (for example, memory limitations, equipment configurations, or specific versions of a program).



## Caution

A Caution advises users that failure to take or to avoid a specified action could result in loss of data.

### 1.3.1 Operating System Names

When the information applies to several operating systems versions, the following naming systems are used:

- **Unix** refers to the following supported operating systems:
  - IBM AIX
  - Red Hat Linux, SUSE Linux
  - Solaris
  - IBM z/OS, when applicable; as Tectia Server for IBM z/OS is running in USS and uses Unix-like tools.
- **z/OS** is used for IBM z/OS, when the information is directly related to IBM z/OS versions.
- **Windows** refers to all supported Windows versions.

## 1.4 Customer Support

All Tectia product documentation is available at <https://www.ssh.com/manuals/>.

FAQ with how-to instructions for all Tectia products are available at <https://documents.ssh.com/>.

If you have purchased a maintenance agreement, you are entitled to technical support from SSH Communications Security. Review your agreement for specific terms and log in at <https://support.ssh.com/>.

Information on submitting support requests, feature requests, or bug reports, and on accessing the online resources is available at <https://support.ssh.com/>.



## Chapter 2 Installation

This chapter gives instructions for installing the Tectia client/server solution on the Windows 64-bit operating system running on x86-64 platform architecture.

Tectia products can also be run on other platforms. For a full list of supported operating systems and instructions for installing Tectia on them, see *Tectia Client User Manual* and *Tectia Server Administrator Manual*.

### 2.1 Preparing for Installation

Make the following preparations and check-ups before you start installing Tectia Client and Server.

#### 2.1.1 Hardware and Disk Space Requirements

Tectia products do not have any special hardware requirements. They can be installed on any computer capable of running the supported operating system versions and equipped with a functional network connection.

The Tectia Client installation requires about 400 MB of disk space. Note that Tectia Client will save each user's settings in that particular user's personal directory.

The Tectia Server installation requires 600 MB free disk space.

For general installation information, see *Tectia Client User Manual* and *Tectia Server Administrator Manual*.

#### Prerequisites

Before installing Tectia Server product on Windows platform, make sure the firewall is open for incoming connections to TCP port 22.

#### 2.1.2 Upgrading Previously Installed Tectia Software

If installed on the same machine, Tectia Client and Tectia Server should always be upgraded to be the same version, because there are dependencies between the common components.

Check if you have some Secure Shell software, such as earlier versions of Tectia products or third-party Secure Shell server or client, running on the machine where you are planning to install the new Tectia versions.

In the following cases you must uninstall the existing version of Tectia Client/Server before installing version 7.0.1:

- Your existing version is 6.0 or earlier.
- Your existing version of Tectia Client contains the transparent tunneling component.

In any other case you can upgrade to Tectia Client/Server 7.0.1 without first uninstalling the existing version. The existing version will be automatically removed from the host during the installation procedure.

### Configuration File Access Permissions

When upgrading a previously installed version of Tectia Server, the access permissions for existing configuration files will be checked during the upgrade installation.

The access permissions for the `ssh-server-config.xml` configuration file should be as follows:

- The owner of the file is a member of the Administrators group.
- Only Administrators and SYSTEM may have full control of the file.
- Users are not allowed to modify the file.
- Other accounts do not have access to the file.

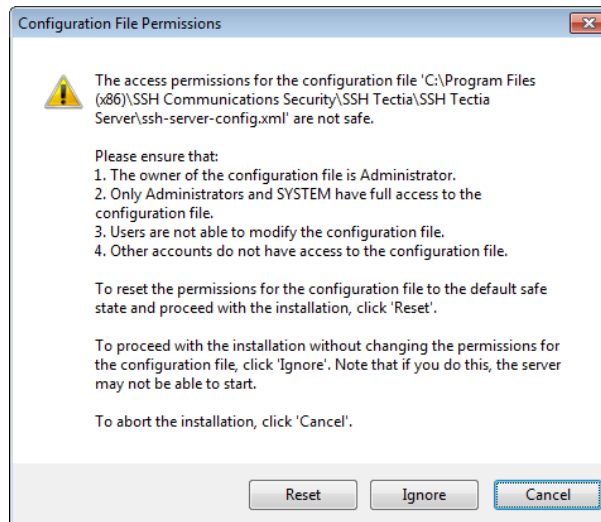
If the access permissions are not safe, you will see the **Configuration File Permissions** dialog box during the upgrade installation. Do one of the following:

- **Reset** the permissions for the configuration file to the default safe state and continue with the installation. (*Recommended*)
- **Ignore** the incorrect permissions and continue with the installation without fixing the permissions. Note that if you decide to do this, the server might not be able to start. You can fix the permissions manually later.
- **Cancel** the installation.



#### Note

Your previous installation of Tectia Server has already been removed, so if you cancel the installation, your machine will be left with no version of Tectia Server installed.



**Figure 2.1. Unsafe configuration file permissions**

### 2.1.3 License File

Tectia Client and Server require a license to function. The license file for Tectia Client is named `stc70.dat` and the license file for Tectia Server is named `sts70.dat`.

Consider the following license-related issues:

- The installation wizard automatically copies the license file to the correct directory when installing from an extracted package.

After installation, the license file is located in the default installation directory:

- "C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia AUX \licenses" on 64-bit Windows versions
- In the commercial installation packages, the license file(s) are included in the compressed (.zip) files together with the release notes (.txt) files and the PDF-format documentation.
- The Tectia evaluation packages do not contain license files; the evaluation versions can be used for 45 days without a license file. A banner message will remind users of how many days are left until the license expires.
- When upgrading the evaluation version or standard commercial version to Tectia Quantum Safe Edition only license file(s) need to be copied to the license directory and Tectia Client and Server software restarted.

### 2.1.4 Creating Operating System User Accounts

Tectia Server does not have a user management program of its own - the user accounts are created with the standard operating system tools.

On Windows, user login requires the rights to log on locally and to access the remote computer from the network. Notice that on domain controllers, these rights are disabled by default. If Tectia Server is installed on a domain controller, permissions to log on locally and to access the computer from the network must be enabled on the domain controller for the intended group, for testing purposes for example the Domain Users group.

## 2.2 Installing Tectia Software

This section introduces how Tectia Client and Server are installed on Windows running on the 64-bit x86-64 platform architecture.

### 2.2.1 Installing Tectia Client on Windows

The Windows installation package is provided in the MSI (Windows Installer) format for Microsoft Windows versions running on the 64-bit (x86-64) platform architecture.

The installation package is a zip file containing the license file and the executable Windows Installer (MSI) packages.

You must have administrator rights to install Tectia Client on Windows.

The installation is carried out by a standard installation wizard. The wizard prompts you for information, copies the program files and sets up the client.

To install Tectia Client on Windows, do the following:

1. Extract the contents of the installation zip file to any temporary location.



#### Note

The license file will be automatically imported to the correct location, if you extract the contents of the `.zip` package before running the `.msi` installer.

If you run the `.msi` installer directly from the `.zip` package, you need to manually install the (`stc70.dat`) license file. The installation wizard will show an error message about missing license file, and when you attempt to start Tectia Client, you are prompted to install the license manually to the correct directory:

- On 64-bit Windows versions: "C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia AUX\licenses"
2. Locate the correct installation file `ssh-tectia-client-<version>-windows-<platform>.msi`, where:
    - `<version>` shows the Tectia Client/Server release version and build number, for example 7.0.1.123.
    - `<platform>` shows the platform architecture: `x86_64` for 64-bit Windows versions.

3. Double-click the installation file, and the installation wizard will start.
4. Follow the wizard through the installation steps and fill in information as requested.
5. Select **Typical** installation. For Tectia Client, it includes the **sshg3.exe**, **scpg3.exe**, and **sftpg3.exe** command-line tools, and the graphical user interface for terminal and file transfer.

To install all components, select **Complete** when the wizard prompts for the setup type.

6. When the installation has finished, click **Finish** to exit the wizard.

The default installation directory is:

- On 64-bit Windows versions: "C:\Program Files (x86)\SSH Communications Security\SSH Tectia"

## Desktop Icon

During Windows installation, a PrivX Desktop icon is added to your desktop. It opens the GUI for Tectia Client.



**Figure 2.2. The PrivX Desktop GUI icon**

## 2.2.2 Installing Tectia Server (and Client) on Windows

The Windows installation package is provided in the MSI (Windows Installer) format for Microsoft Windows versions running on the 64-bit (x86-64) platform architecture. Tectia Server installation package can be used to install also Tectia Client.

The installation package is a zip file containing the Tectia Client/Server license files and the executable Windows Installer (MSI) packages.

You must have administrator rights to install Tectia Client/Server on Windows.



### Note

Tectia Server cannot be installed on file systems that do not support permissions (for example, FAT16 or FAT32). The hard disk partition where Tectia Server is installed must use the NTFS file system.

The installation is carried out by a standard installation wizard. The wizard will prompt you for information and will copy the program files, install the services, and generate the host key pair for the server.

To install Tectia Server and (optionally) Tectia Client on Windows, do the following:

1. Extract the contents of the installation zip file to any temporary location.
2. Locate the correct installation file `ssh-tectia-server-<version>-windows-<platform>.msi`, where:
  - `<version>` shows the Tectia Client/Server release version and build number, for example 7.0.1.123.
  - `<platform>` shows the platform architecture `x86_64` for 64-bit Windows versions.
3. Double-click the installation file, and the installation wizard will start.



## Note

The license files for Tectia Client/Server will be imported automatically, if you extract the contents of the `.zip` package before running the `.msi` installer.

If you run the `.msi` installer directly from the `.zip` package, you need to manually import the license files (`sts70.dat` for Tectia Server and `stc70.dat` for Tectia Client) after completing the installation. The installation wizard will show an error message about missing license files, and when you attempt to start Tectia Client/Server, you are prompted to import the license(s) manually to the correct directory:

- "C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia AUX\licenses" on 64-bit Windows versions

4. Follow the wizard through the installation steps and fill in information as requested.

The installation wizard will display options **Typical**, **Custom** and **Complete**.

If you do not want to install both Tectia Server and Client, select **Custom** and choose which product components you wish to install.

The server host key is generated during the installation.

5. When the installation has finished, click **Finish** to exit the wizard.
6. For Tectia Server to be fully functional after installation, you must restart the computer.



## Note

If you do not restart the computer after installing Tectia Server, the server will run with the following limitations in the authentication of local users and domain users from one-way trusted domains:

- Public-key authentication will not work.
- Certificate authentication, keyboard-interactive submethods RADIUS and RSA SecurID, and host-based authentication will only work if the password cache is enabled and the user's password is stored in the cache.

- Authentication selectors of type **User group** (`user-group`) and **Administrator** (`user-privileged`) will not work.

Tectia Server will write warning messages into the Windows Event Log. Use the Windows Event Viewer to examine the log contents (On the **Tectia Server Configuration** tool's **Tectia Server** page, click the **View Event Log** button).

Tectia Server will start automatically every time the computer is started, and it stays running in the background. Tectia Server displays no icons on the desktop, but you can see it listed in the Windows **Start** → **(All) Programs** menu.

7. Usually there is no need to manually restart Tectia Server.

If you need to restart Tectia Server (for example because of a missing license or because some other secure shell software is running on port 22), use the Tectia Server Configuration GUI as follows:

- a. In the Windows **Start** menu, click **(All) Programs** → **Tectia Server** → **Tectia Server Configuration**.
- b. Under the Server Status, click the **Start Server** button.

The Server will start, and the status changes first to `Starting...` and then to `Running`.

- c. To exit the Tectia Server configuration GUI, click **OK**.

### 2.2.3 Installation Complete

After a successful installation, Tectia Client and Tectia Server are automatically started at reboot and they keep running in the background until you stop them manually, or shut the host down.

You can use Tectia Client and Tectia Server with the default settings to test their functions. For instructions on opening a secure connection for the first time, see [Chapter 3](#).

It is also possible to customize the behavior of the Tectia client/server solution according to your needs. To learn more about modifying the Tectia configuration for different purposes, refer to the later chapters in this manual:

- [Chapter 4](#) explains configuring of authentication methods
- [Chapter 5](#) explains secure file transfer
- [Chapter 6](#) explains securing application connections.

## 2.3 Removing Tectia Software

If you need to remove the Tectia Client and Server software, follow the instructions below.



## Note

The uninstallation procedure removes only the files that were created when installing the software. Any configuration files have to be removed manually from each user's %APPDATA%\SSH directory and from the installation directory:

- "C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server" on 64-bit Windows versions

### 2.3.1 Removing Tectia Client and Server from Windows

To remove Tectia Client and Server from a Windows environment, follow the instructions below:

1. From the Windows **Start** menu, open the **Control Panel** and click **Programs and Features**.
2. In the list of installed programs, select **Tectia Server** or **Tectia Client** and click **Uninstall**.



## Note

If you have installed Tectia Client together with Tectia Server, uninstalling Tectia Server will also remove Tectia Client.

3. Click **Yes** to confirm.
4. After you have uninstalled Tectia Server, the system will prompt you to restart Windows.

---

## Chapter 3 Connecting to Remote Server

This section explains how you can log in from Tectia Client to Tectia Server using password authentication with the default settings. The default settings on Tectia Client and Server allow login with passwords, public keys, GSSAPI, and keyboard-interactive. By default, passwords are used for user authentication, and public keys for server authentication.

You are expected to have a user account on the remote server where you will connect, and the server must have a Secure Shell server running. In the following example, you can also just connect within the local machine, to make sure that you know the server's address and that it has Tectia Server running.

### 3.1 First Connection with Password

You can connect to a remote host by using PrivX Desktop as follows:

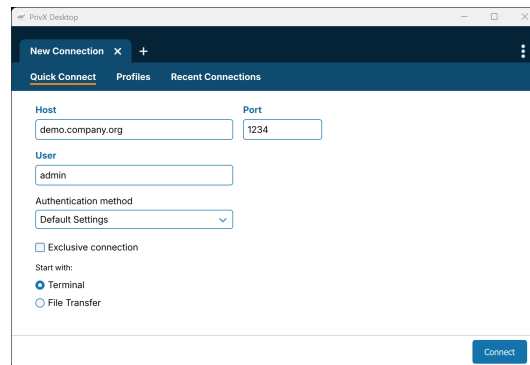
1. Open the Tectia Client GUI for example by clicking the PrivX Desktop icon on your desktop.



2. PrivX Desktop offers several ways to open a Secure Shell connection:
  - On **Quick Connect** page you can provide host name or profile name and change the most common connection settings for this connection without editing the profile.
  - If you already have an ongoing session, open a **New Connection** in a new tab by clicking the + button. You can connect to a new remote host computer and still keep the old connection to a different host open.
  - If earlier session has been disconnected, you can open a new session by pressing **Enter** or **Space** on the keyboard when the (still disconnected) terminal or file transfer window is active.
  - If you or the administrator has defined connection profiles, you can also connect from **Profiles** page of the **New Connection** tab by clicking the profile tile. From the alternate profile tile menu you can choose to `Open Terminal` or `Open File Transfer` initially instead of the default channel type.

In this case, the settings defined in the profile (hostname, port, user name etc.) are automatically used for the connection.

- On **Recent Connections** page of the **New Connection** tab, you can quickly reopen connections to the remote hosts that have been disconnected and closed since the Connection Broker has been started or open additional connections to the host with ongoing connections.
3. On **Quick Connect** page of the **New Connection** tab you can define the server host you want to connect to:



Define at least the Host and click **Connect**:

- **Host** – the FQDN, short host name, the IP address of the remote host or the connection profile name. (To test connecting to Tectia Server installed on the same machine, type in "localhost".)
- **User** – your user name on the remote host.
- **Port** – specify alternate port number if the default Secure Shell listener port 22 is not used on the remote host.
- **Authentication Method** – by default enabled user authentication methods from `Default Settings` are used unless you specify one of the `Quick Connect` user authentication methods. `Password` will attempt methods used commonly for password-based methods like `Password` and `Keyboard-interactive only`, and `Public-Key` will use available certificates and public keys only.
- **Exclusive connection** – by default a new connection to the same remote host will open a new channel. To open an additional secure shell connection or prevent opening additional channels later within the connection, enable `Exclusive connection` checkbox.
- **Start with** – `Terminal` or `File Transfer` as the initial channel in the secure shell connection. You can later open additional channels or new connections to the server.

Environment variables can be used to pre-fill the `Quick Connect` values. If undefined, the `Default Settings` will be used and the client prompts for required values if needed.

4. The server authentication phase starts. The remote server host will provide your local computer with its host public key. The host key identifies the server host.

Tectia Client checks if information on this key is already stored in your own host key directory. If not, the host key directory common to all users on your computer is checked next. If information on this host key is not found, you are asked to verify the new key.

When public-key authentication is used to authenticate the server, *the first connection is very important*. When Tectia Client receives a new server host key, it will display the host identification message.



**Figure 3.1. The host identification dialog – the first connection to a remote host**

The message displays the fingerprint of the host's public key in the SSH Babble format that is a series of pronounceable five-letter words in lower case and separated by dashes. By default also base64-encoded SHA-256 fingerprint is shown.

5. Verify the validity of the fingerprint, preferably by contacting the administrator of the remote host computer by telephone. After verifying the fingerprint, it is safe to save information on the host key for future use. You can also choose to cancel the connection, or to proceed with this connection without saving the host public key information.



### Caution

Never save a host public key without verifying its authenticity!

6. Click **OK** to close the host identification dialog.

Information on the server public key will be stored on the client-side machine so that the client can later validate the key. On Tectia Client, the public key information is stored in the user's hostkeys directory:

`%APPDATA%\SSH\HostKeys`, where `%APPDATA%` corresponds to `C:\Users\\AppData\Roaming`

After the first connection, only the locally stored information about the server public key will be used in server authentication.

7. The user authentication phase starts. You will be prompted to authenticate yourself to the server with your password. The required authentication method depends on the server settings.


After the server has successfully authenticated you, the Secure Shell connection to the server is opened.

## 3.2 Creating Connection Profiles

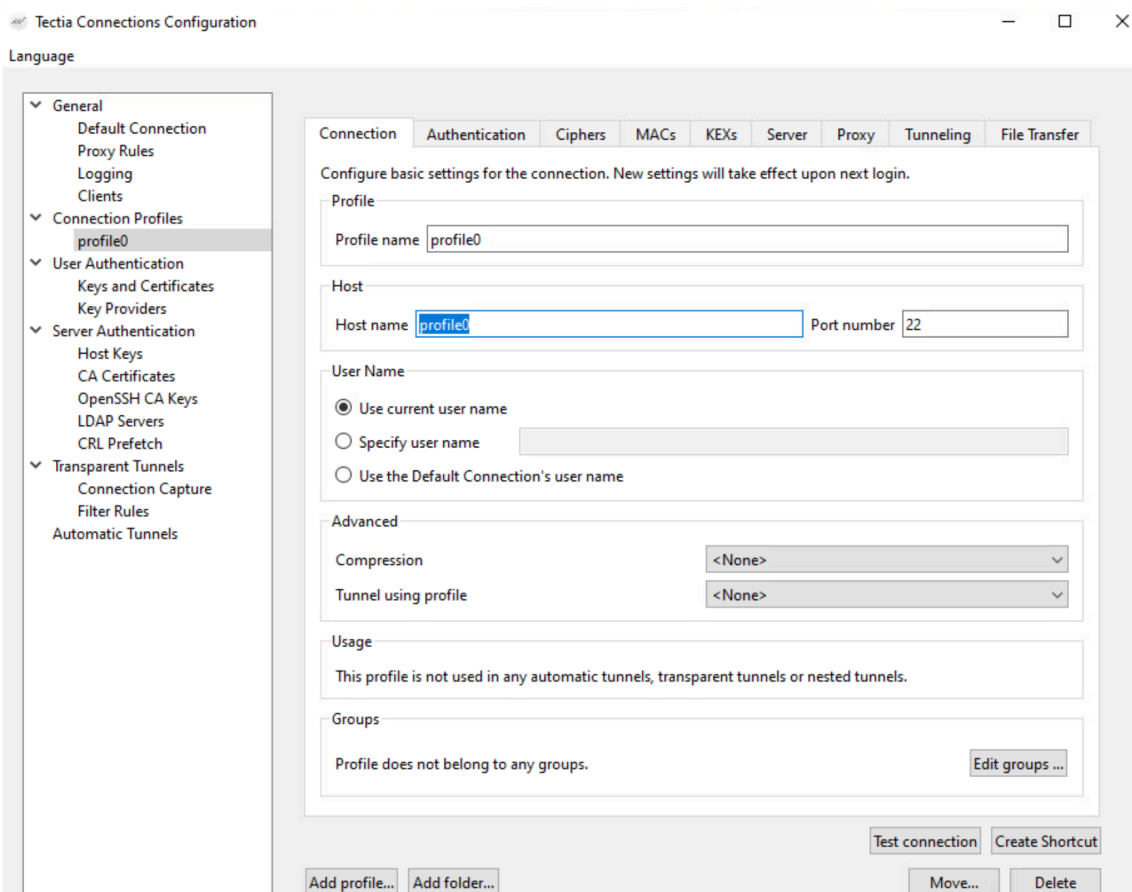
On Tectia Client on Windows, you can configure separate connection settings for each Secure Shell server you connect to. You can also create several profiles for the same server, for example, with different user accounts.

You can add connection profiles via the following views:

- Start PrivX Desktop GUI and open the **Tectia Connections Configuration GUI** by selecting **Configuration** from the menu icon.

(Alternatively, you can open the Tectia Connections Configuration GUI by right-clicking the PrivX Desktop tray icon  in the Windows taskbar notification area and selecting **Configuration** from the shortcut menu.)

In the Tectia Connections Configuration GUI, go to the **Connection Profiles** page (as shown below) and click **Add profile**.



**Figure 3.2. Adding connection profiles**

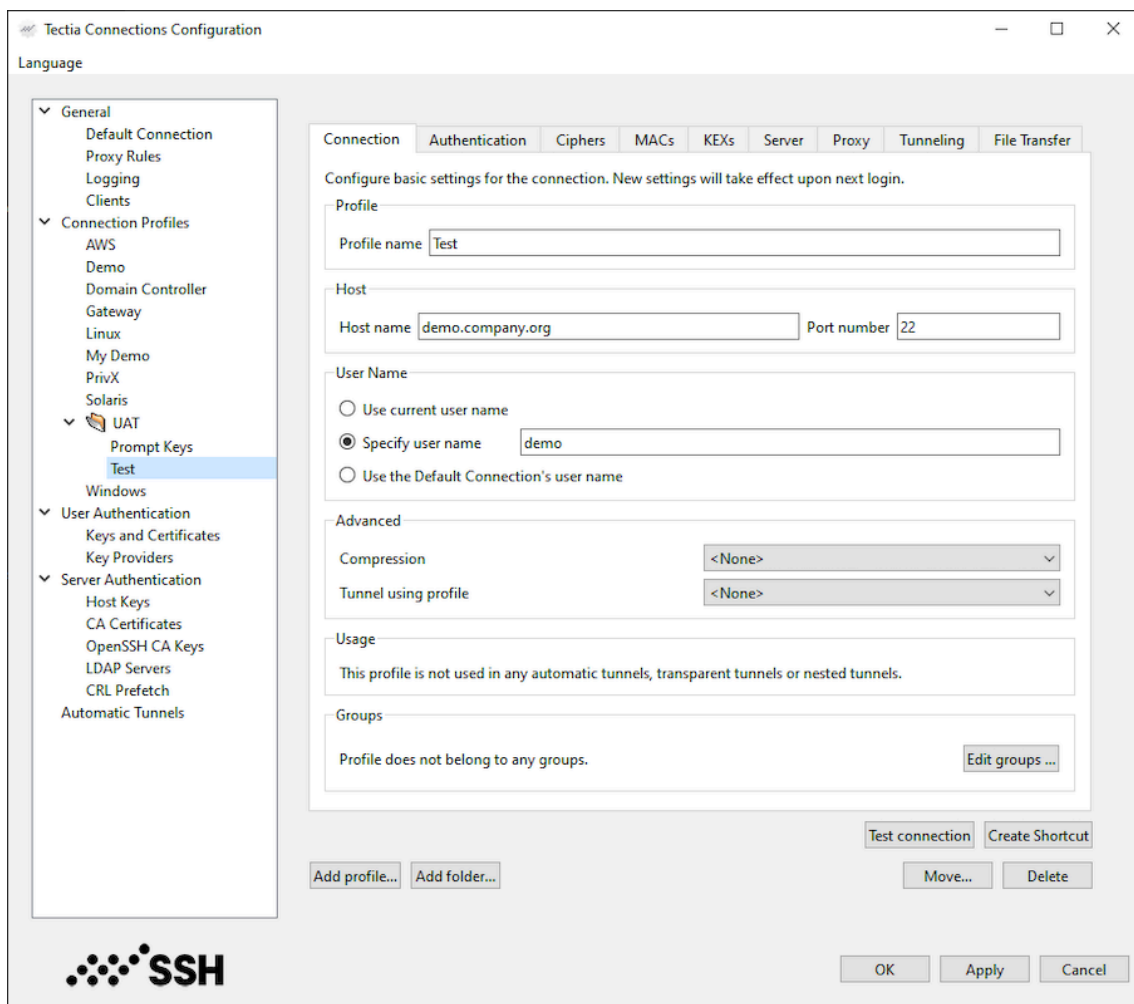
Newly created connection profiles will inherit the default values for authentication, ciphers, MACs, KEXs, tunneling, and advanced server settings defined under the **General** → **Default Connection** page. The values can be customized on the profile-specific tabbed pages, see [Figure 3.3](#).

To rename a connection profile, right-click the profile name in the **Connection Profiles** list and click **Rename**. Type in the new name.

To remove a connection profile, select the profile and click **Delete**. You will be asked for confirmation. Click **Yes** to proceed with the deletion.

### 3.2.1 Defining Connection Profile Settings

Under the **Connection Profile** page, on the **Connection** tab, you can define the protocol settings used in the connection. Any changed connection settings will take effect the next time you log in.



**Figure 3.3. Configuring connection profiles**

#### Profile

In **Profile name**, type a name for the profile.

## Host

In **Host name**, enter the name of the remote host computer to which you want to connect with the profile.

In **Port number**, enter the port number you want to use for the Secure Shell connection. The default port is 22.

### **Note**

A Secure Shell server program must be listening to the specified port on the remote host computer or the connection attempt will not succeed. If you are unsure which port the remote host computer is listening to, contact the system administrator of the remote host.

## User Name

Select **Use current user name** if the connection should always be made using the currently logged in Windows user name. This is similar to defining %USERNAME% (note the percent signs) as the user name.

Select **Specify user name** and enter the user name, if you want to define the user name to be used when connecting to the remote host computer. If you specify %USERNAME% (note the percent signs) as the user name, it will be replaced with the name of the current Windows user account upon connecting.

## Advanced

*Not needed now:* In **Compression**, select the desired compression setting from the drop-down menu. Valid choices are **zlib** and **none**. Compression is disabled by default.

*Not needed now:* In **Tunnel using profile**, select the desired connection profile from the drop-down menu. Any nested tunnels will be created through the profile. For information on the tunneling features, refer to the *Tectia Client User Manual* .

## Chapter 4 Configuring Authentication Methods

The Tectia client/server solution has separate authentication procedures for authenticating the servers and the users. The authentication is mutual; the client authenticates the server and the server authenticates the user.

The server configuration defines which authentication methods are allowed, and the client configuration defines the order in which the methods will be tried.

In this guide we introduce how public-key authentication is used in authenticating the remote Tectia Server host. For user authentication, we introduce both the password authentication method, as it is set up by default, and public keys, which provide stronger security and make it possible to use non-interactive login securely.

### 4.1 Server Authentication Methods

The server is authenticated with a digital signature based on an RSA, DSA, ECDSA or Ed25519 public-key algorithm.

During the server installation process, one RSA key pair (with the file names `hostkey` and `hostkey.pub`) is generated and stored on the server host in directory:

- "C:\Program Files (x86)\SSH Communications Security\SSH Tectia\SSH Tectia Server" on 64-bit Windows versions

By default, this key pair is used for server authentication.

For information on connecting to a remote server for the first time, see [Chapter 3](#).

### 4.2 User Authentication with Passwords

The password and public-key authentication methods are set up by default on both Tectia Client and Server. Passwords are the easiest method for authenticating users because no configuring is required on the server side. The passwords are protected from eavesdroppers, since all communication is encrypted.

On Windows, password authentication uses the Windows password to authenticate the user at login time.

For information on the differences in user name handling on local and domain accounts, see *User Authentication with Passwords* in *Tectia Server Administrator Manual*.

### 4.3 User Authentication with Public Keys

Public-key authentication is based on the use of digital signatures and provides very good authentication security.

To use public keys in user authentication, you must first create a key pair on the client. One of the created key files is your public key, and the other is your secret private key.

The security level of the key pair depends on the complexity (or bit length) of the key. Larger keys are more secure, but generating and using them takes a longer time.



#### Note

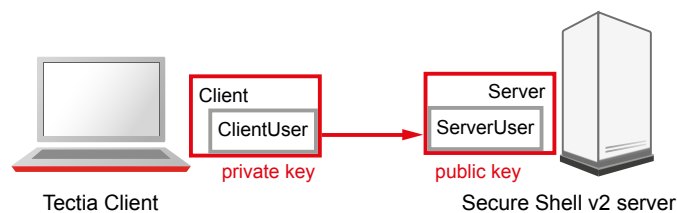
The default RSA key size (3072 bits) provides 128-bit security and default ECDSA key size (384 bits) provides 192-bit security. We do not recommend generating RSA or DSA keys smaller than 2048 bits even for interoperability with 3rd party implementations.



#### Note

We recommend you to replace your SSH keys with new ones at a minimum frequency of every two years.

The server must know the user's public key, so you need to upload the public key to the server, but the private key must remain only in your possession.



**Figure 4.1. User public-key authentication**

When you start logging in to a remote server, the client sends a signature to the server, and the server checks for matching public keys. If the key is protected with a passphrase, the client requests you to enter the passphrase.

Remember that your private key is used to authenticate you. Keep your private key in a secure place and make sure that no one else has access to it. If anyone else can access your private key, they can attempt

to log in to the remote host computer pretending to be you. Define a passphrase to protect your private key, whenever possible.



## Caution

Generate keys only on your personal computer that no one else can access! Do not store your private key on a computer that is shared with other users.

When you start using public-key authentication, do the following:

1. Generate a key pair. You can generate your own key files with the help of a built-in **Public-Key Authentication Wizard** (see [Section 4.3.1](#)).

You can also import existing keys on the **Keys and Certificates** page of the **Tectia Connections Configuration GUI**.

2. Upload your public key to the remote host computer (running Tectia Server) automatically (see [Section 4.3.2](#)).



## Note


Tectia Server supports also user public keys generated with OpenSSH. Tectia Server can be configured to check the OpenSSH `authorized_keys` file in addition to the Tectia `authorized_keys` directory and/or `authorization` file. Public keys defined in the Tectia locations have precedence over the keys in the OpenSSH file if the same key is defined in both.

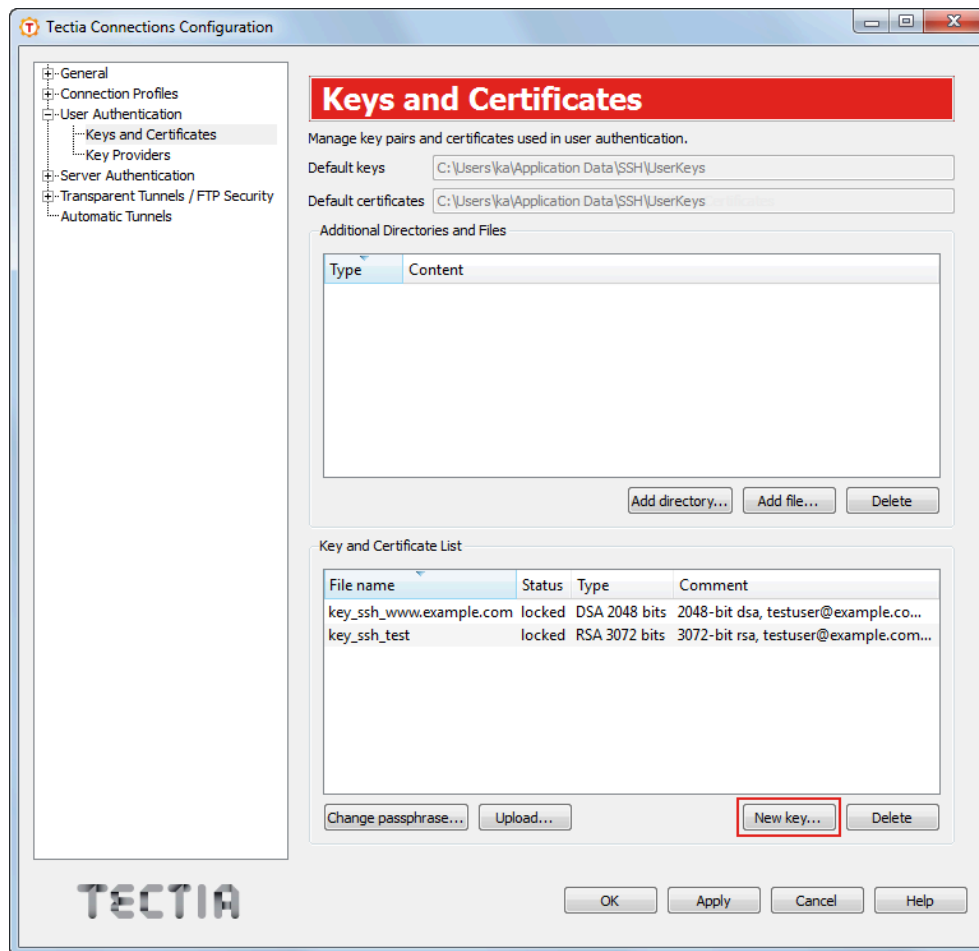
These instructions assume that the client user is allowed to log in to the remote host, where Tectia Server is running, using password authentication.

### 4.3.1 Creating Keys with Public-Key Authentication Wizard

On Windows, you can use the Tectia **Public-Key Authentication Wizard** to generate a key pair. The wizard will generate two key files, your private key and your public key, and store them in the `%APPDATA%\SSH\UserKeys` directory on your local computer. The public key has `.pub` as the file extension, and the private key file has the same base file name as the public key but no file extension.

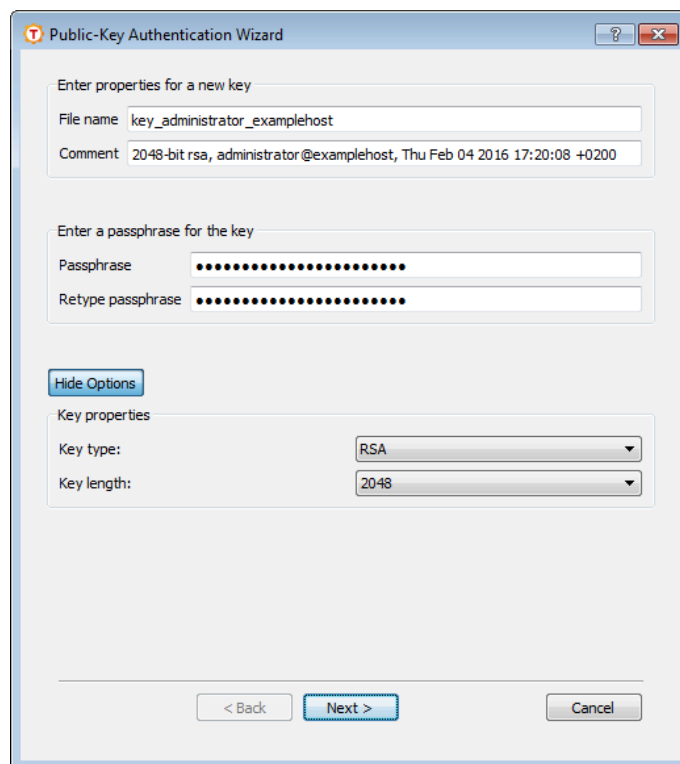
Public key pairs can also be generated with the command line tool **ssh-keygen-g3**. For instructions, see *Client User Manual*.

1. Open the **Tectia Connections Configuration GUI** by clicking the Tectia icon  in the Windows taskbar notification area or on the Tectia Client toolbar.
2. Go to **User Authentication** and select the **Keys and Certificates** page. Click **New key**.



**Figure 4.2. Tectia Connections Configuration GUI, Keys and Certificates view**

3. The **Public-Key Authentication Wizard** starts.



**Figure 4.3. The Public-Key Authentication Wizard**

4. Define the key properties and the required passphrase to protect your key pair.

#### **File Name**

Type a unique name for the key file. The wizard suggests a name consisting of your user name and the host name.

#### **Comment**

Write a short comment that describes the key pair. For example, describe the connection the key is used for. The wizard suggests a comment consisting of the key length and type, your user name and the host name, and the current date and time. This field is not obligatory, but it helps to identify the key later.

#### **Passphrase**

Type a phrase that is difficult to guess. Use ideally at least 20 characters, both letters and numbers. Any punctuation characters can be used as well. While the passphrase or private key is never sent over the network, a dictionary attack can be used against a private key if it is accessible locally. For ease of use, an authentication agent is recommended instead of leaving the passphrase empty. By default ssh-broker-g3 functions as an authentication agent.

 **Note**

In FIPS mode, due to a FIPS regulation which forbids exporting unencrypted private keys out of the FIPS module, it is not possible to generate user keys without a passphrase.

If the key pair will be used for automated jobs, you can leave the passphrase field empty to generate the key without a passphrase.

You will be requested to enter the passphrase always when using the keys to authenticate yourself. The passphrase works in a way similar to a password and gives some protection for your private key.

Memorize the passphrase carefully, and do not write it down.

**Retype passphrase**

Type the passphrase again. This ensures that you have not made a typing error.

5. Click the **Advanced Options** if you want to define the type and/or length of the key to be generated to be different from the defaults. By default, Tectia Client generates a pair of 3072-bit RSA keys.

In the **Key Properties** area, you can define the following:

**Key Type**

Select the type of the key to be generated. Available options are Ed25519, RSA, ECDSA and DSA.

 **Note**

In FIPS mode (conforming to FIPS 186-5) RSA, ECDSA and Ed25519 are supported. DSA has been deprecated.

**Key Length**

Select the length (complexity) of the key to be generated. Available options are:

- DSA/RSA keys: 2048, 3072, 4096, 5120, 6144, 7168, 8192 bits
- ECDSA keys: 256, 384, 521 bits
- Ed25519 keys: 256 bits

Larger keys of the same key type are more secure, but also slower to generate. A 256-bit ECDSA key and a 3072-bit RSA key provide equivalent security.

6. Click **Next** to proceed to uploading the key. The wizard continues with Step 3 in [Section 4.3.2](#).

Uploading existing public keys to new remote servers is instructed in [Section 4.3.2](#).

## 4.3.2 Uploading Public Keys Automatically

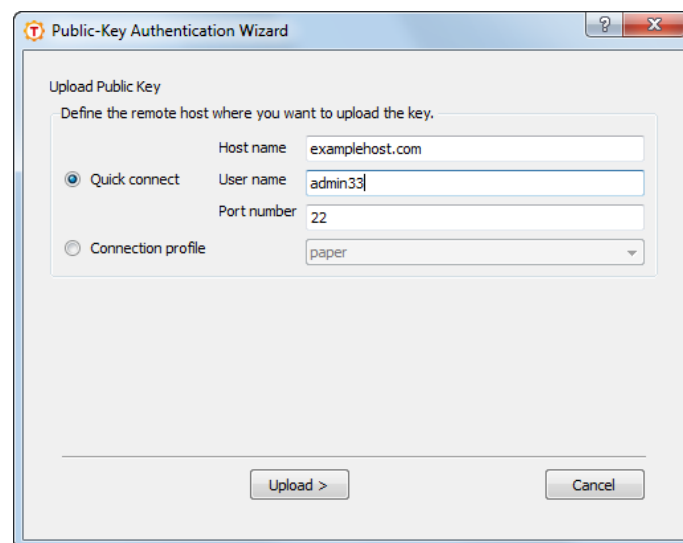
Public keys can be automatically uploaded to servers that have the SFTP subsystem enabled, and by default, SFTP is enabled on Tectia Servers. The **Public-Key Authentication Wizard** automatically uploads each new public key to a remote host of your choice. All existing keys are also listed on the **Keys and Certificates** page of the **Tectia Connections Configuration GUI**, and you can select a key to upload it to a remote server at any time.

The public key will be uploaded to the default user home directory (`%USERPROFILE%\ .ssh2` on Windows) on the remote server.

## Note

The key user is required to have `write` permissions to the key directory on the server, otherwise the automatic upload will fail. The administrator of the remote host computer may have restricted user access so that users are not able to configure public-key authentication for themselves even if public-key authentication is allowed in the server configuration.

1. To access the **Public-Key Authentication Wizard**, click **User Authentication** → **Keys and Certificates** on the tree view.
2. Select a key from the **Key and Certificate List** and click **Upload**.
3. The **Upload Public Key** view of the wizard appears.



**Figure 4.4. Uploading a key**

Define the remote host where you want to upload the key:

### Quick connect

Select this option to define the remote **Host name** and your **User name** there. The default Secure Shell **Port number** is 22.

### Connection profile

Select a **Connection profile** from the drop-down list that specifies the desired remote host and user name.

4. Click **Upload** to transfer the key to the selected server. If you are already connected to the remote server host, the key upload starts immediately. If you are not connected, you will be prompted to authenticate on the server (by default with password).

## 4.4 Setting up Non-interactive Authentication for Automatic Scripts

When Tectia Server is used for automated file transfer, you can create separate user accounts for file transfer purposes. When such user accounts are used only for non-interactive file transfers, it is advisable to disable terminal access on the server side. See instructions in [Section 5.2.5](#).

Non-interactive authentication with public keys and scripted commands can be set for the SFTP accounts. For non-interactive batch jobs, you can use public-key authentication without a passphrase.

Running the client non-interactively requires that you have already saved the server's public host key on the client, and set up a non-interactive method for user authentication. Batch mode should be used non-interactively with command-line tools.

1. Generate an RSA key pair and leave the passphrase field empty. See instructions in [Section 4.3.1](#).
2. For uploading the keys, see instructions in [Section 4.3.2](#).



### Caution

Make sure your private key is not accessible to others. This is especially important when the key is stored without a passphrase.

For more information on other non-interactive authentication methods, see Chapter *Authentication* in *Tectia Server Administrator Manual*.

---

## Chapter 5 Using Secure File Transfer

Secure File Transfer Protocol (SFTP) is a secure replacement for the plain-text FTP service. The SFTP service encrypts all files during the transfer.

This chapter shows how secure file transfer is used and describes a use case plus the required configuration changes.

### 5.1 Using SFTP on Tectia Client

On Tectia Client, the default settings for SFTP are applicable in most cases, so you can start experimenting with file transfers immediately. The SFTP service can be used on the command line or via PrivX Desktop File Transfer GUI. The GUI includes tooltips to guide you.

#### 5.1.1 Using SFTP on Command Line

Command **sftpg3** is used on the command line to connect to any host that is running a Secure Shell version 2 server with the SFTP server subsystem enabled.

The basic syntax of **sftpg3** is:

```
sftpg3 username@remotehost
```

This logs you in to the remote host. For example, after a successful login you can fetch a file from the remote host to your local host with a command like this:

```
sftp> get filename
```

To view the commands available with **sftpg3**, type `help` at the SFTP prompt:

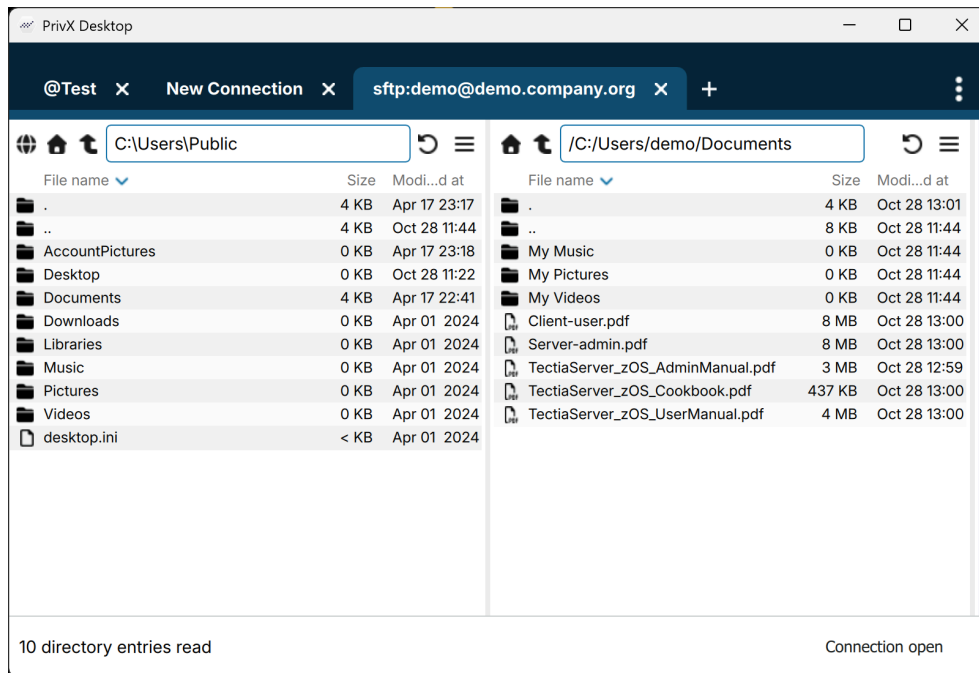
```
sftp> help
```

For more information on **sftpg3**, see the *Tectia Client User Manual*.

#### 5.1.2 Using PrivX Desktop File Transfer GUI

You can transfer files via PrivX Desktop.

You can open a connection to a remote host using a connection profile defined in the Connection Broker configuration (click **Profiles**), or by using the **Quick Connect** option.



**Figure 5.1. PrivX Desktop File Transfer GUI**

PrivX Desktop makes it easy to download files from a remote host computer into your local computer and to upload files to a remote host.

## 5.2 Configuring Tectia Server for Automated Secure File Transfer

Tectia Server can be used for automated secure file transfer. This use case shows how to configure Tectia Server for it. Tectia Client does not require any configuration changes.

The goal of changing the Tectia Server configuration is to improve the security of the system for automated file transfers. This requires some user restrictions on the SFTP usage. In this use case, the following restrictions are defined on Tectia Server:

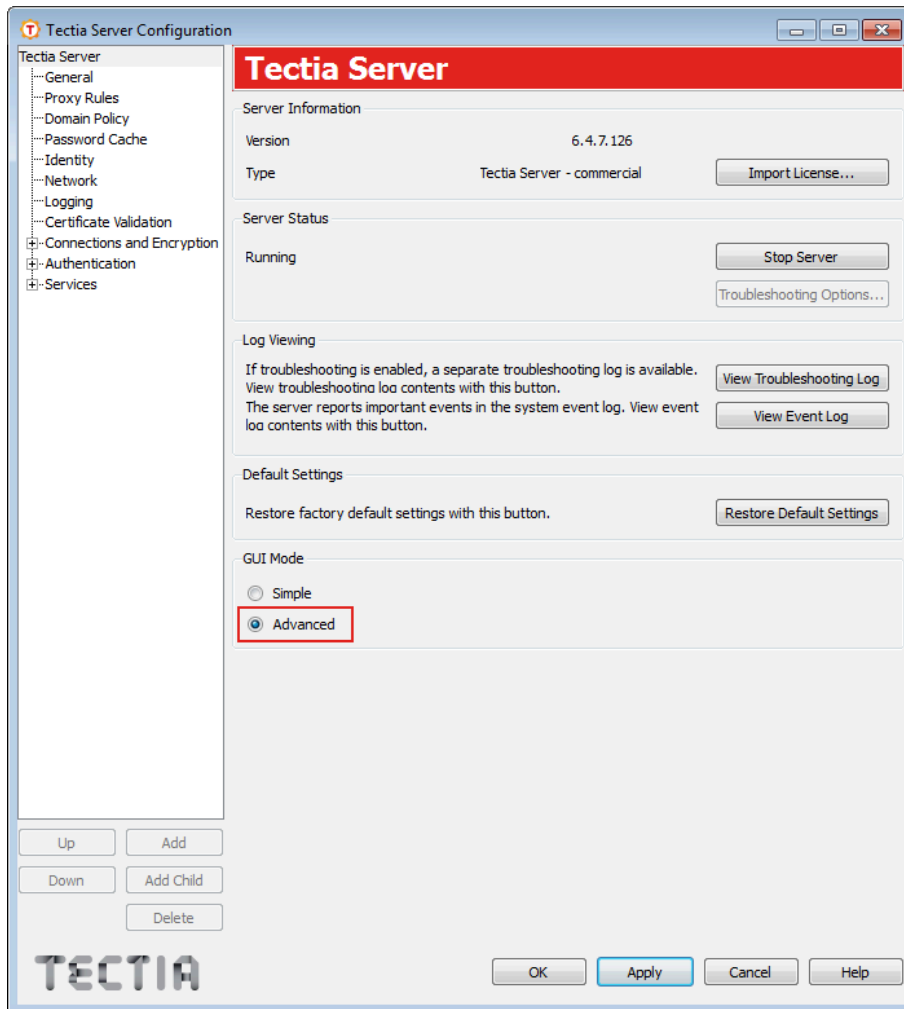
1. Public keys are the only allowed authentication method. See instructions in [Section 5.2.2](#).
2. SFTP service is allowed only for specially created user groups `SFTP-users` and `admin`. SFTP service is denied from all other users. See instructions in [Section 5.2.3](#), [Section 5.2.4](#) and [Section 5.2.5](#).
3. Members of `SFTP-users` have access to their user-specific home folders only. This can be defined with virtual folders. See instructions in [Section 5.2.4](#) and [Figure 5.14](#).
4. Terminal access is allowed only for administrators; from everyone else, it is denied. See instructions in [Section 5.2.3](#) and [Section 5.2.5](#).

## 5.2.1 Opening Tectia Server Configuration GUI

On Windows, Tectia Server is configured through a graphical user interface.

Open the **Tectia Server Configuration GUI** by clicking **Start** → **(All) Programs** → **Tectia Server** → **Tectia Server Configuration**.

To access the necessary Tectia Server configuration settings, enable the advanced settings by clicking **Advanced** under **GUI Mode** on the **Tectia Server** view:

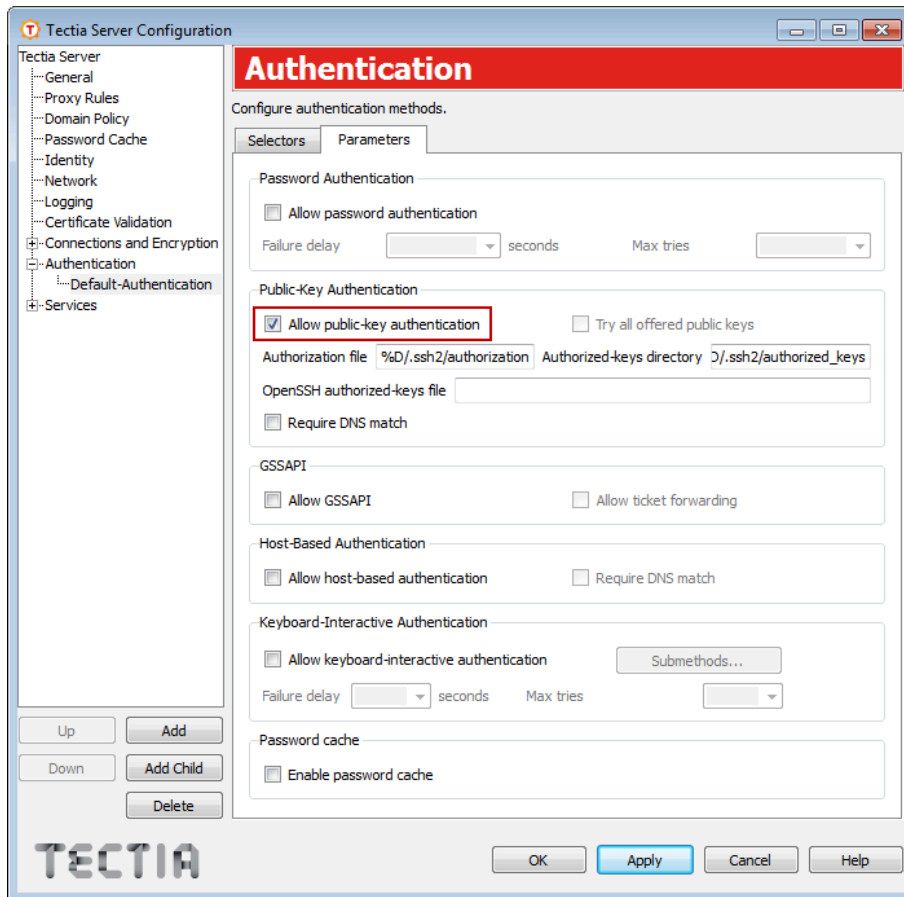


**Figure 5.2. Enable Advanced GUI Mode**

Now proceed to the actual configuration settings. See the example views below.

## 5.2.2 Enabling Public-Key Authentication

Define public-key authentication as the only allowed authentication method under the **Authentication - Default-Authentication** page, on the **Parameters** tab.

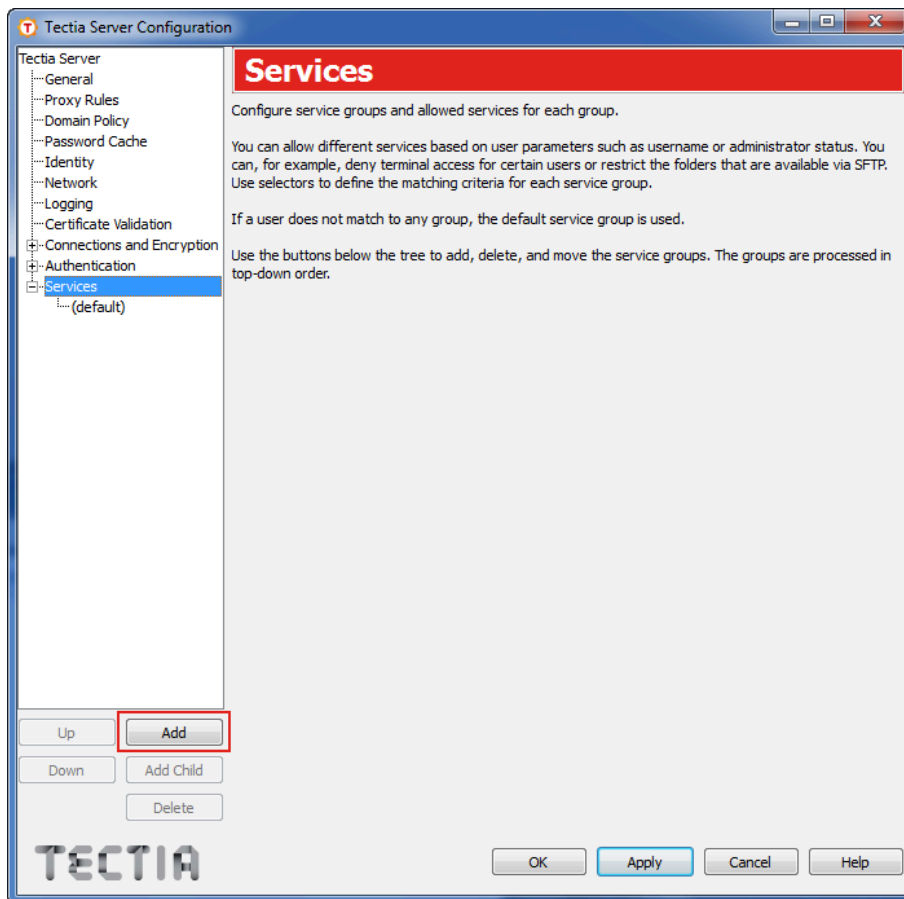


**Figure 5.3. Enable only public-key authentication**

### 5.2.3 Settings for the Admin Group

Create a user group with administrator rights and allow all actions and services for the members of the group.

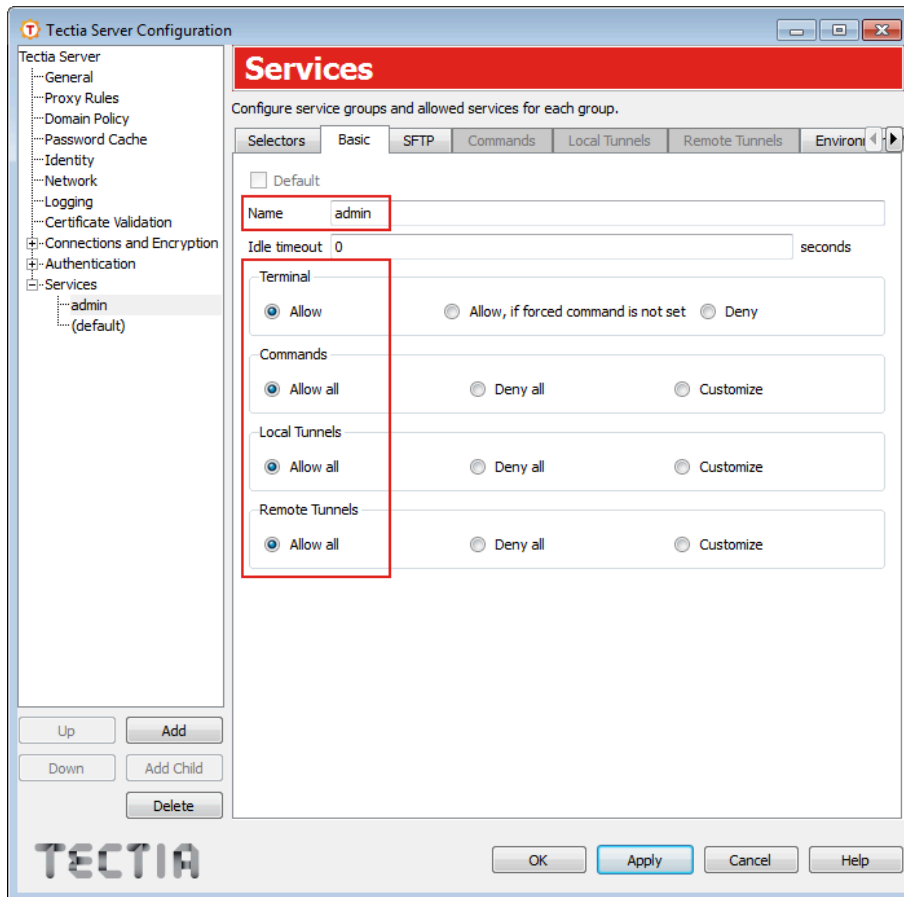
1. Under the **Services** page, click **Add** to create a group for administrators.



**Figure 5.4. Start creating a user group**

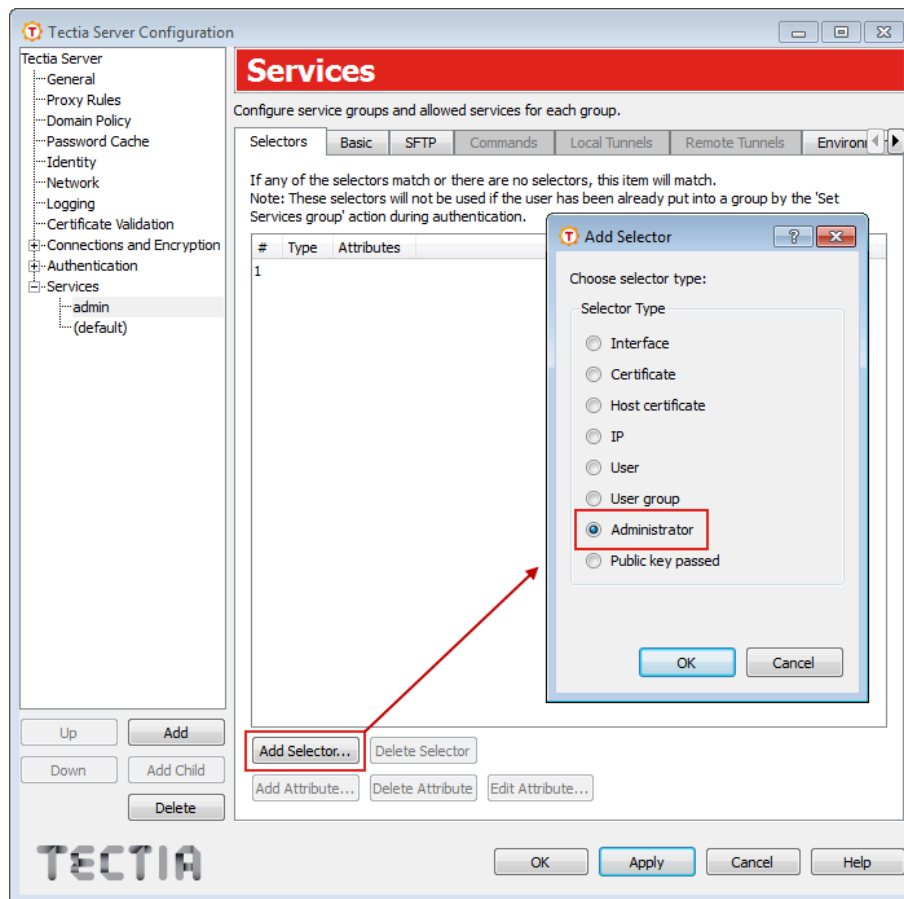
Tectia Server will use a placeholder name `group1` for a newly created group.

2. On the **Basic** tab, name the group `admin` and choose **Allow** or **Allow all** for all services, **Terminal**, **Commands**, **Local Tunnels**, and **Remote Tunnels**.



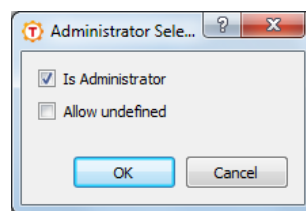
**Figure 5.5. Name the group 'admin' and allow all services**

3. Go to the **Selectors** tab, and click **Add Selector**. On the **Add Selector** tab, choose selector type **Administrator**, and click **OK**.



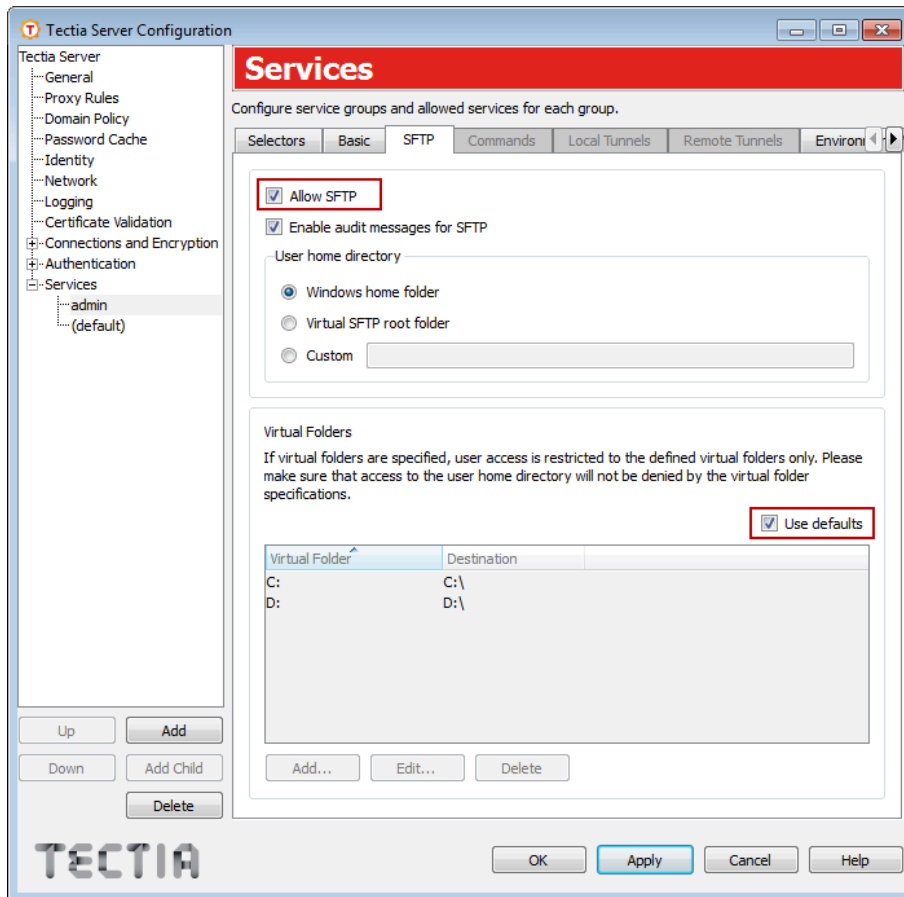
**Figure 5.6. Define the group selector as administrator**

4. When the **Administrator Selector** view opens, select **Is Administrator**, and click **OK**.



**Figure 5.7. Define user group as administrator group**

5. On the **SFTP** tab, allow the SFTP service for the `admin` group, and keep the default settings.

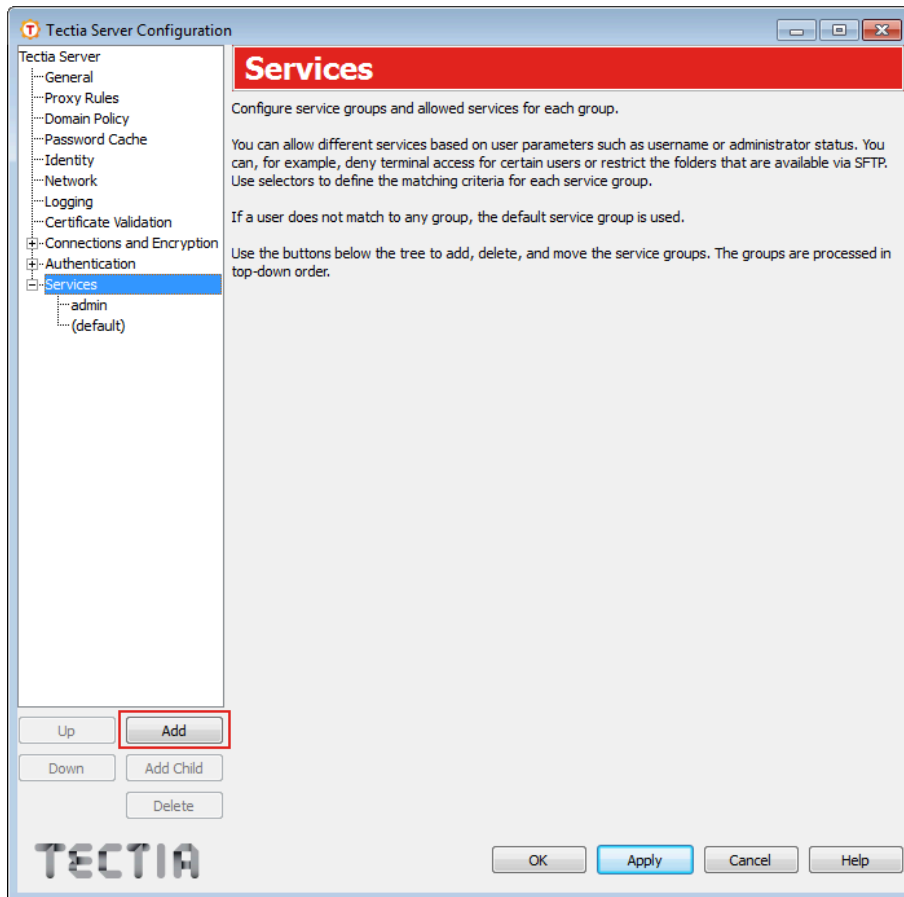


**Figure 5.8. Allow SFTP for the group 'admin'**

## 5.2.4 Settings for the SFTP-users Group

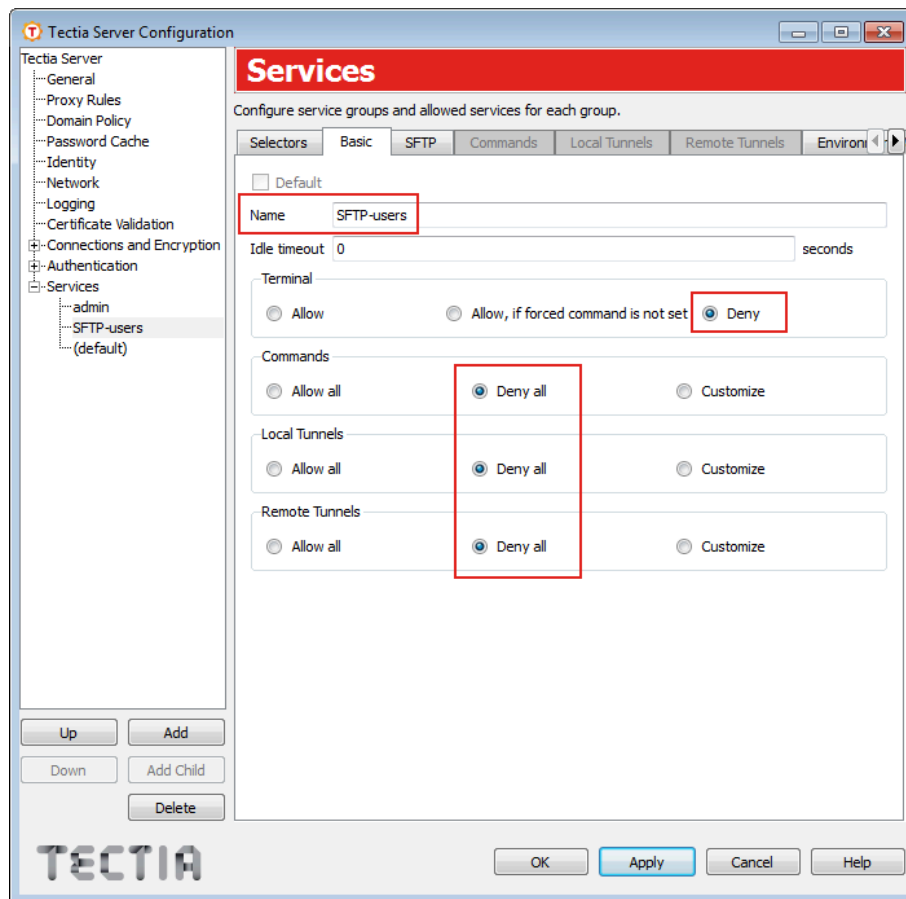
Create a dedicated user group for secure file transfer users. An existing operating-system-related user group is attached to the Tectia SFTP group, and they are allowed access only to their user-specific home folders.

1. Under the **Services** page, click **Add** to create a group for SFTP users.



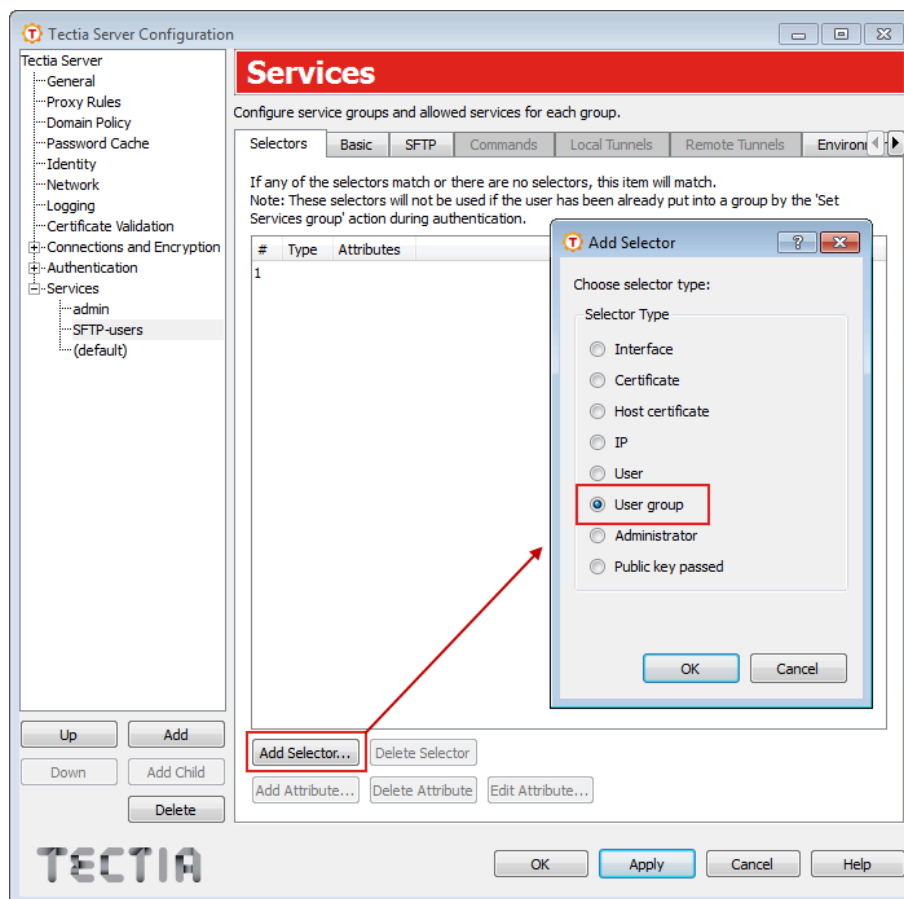
**Figure 5.9. Start creating the SFTP user group**

2. On the **Basic** tab, name the group `SFTP-users` and choose **Deny** or **Deny all** for all the listed services, **Terminal**, **Commands**, **Local Tunnels**, and **Remote Tunnels**. For more information on restricting terminal access, see [Section 5.2.5](#).



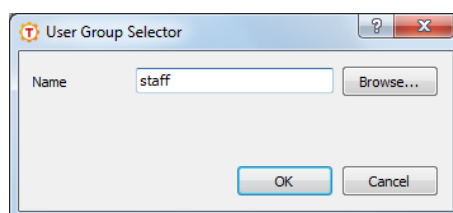
**Figure 5.10. Name the group 'SFTP-users' and deny all services**

3. On the **Selectors** tab, click **Add Selector** and choose the selector type **User Group**, and click **OK**.



**Figure 5.11. Define the group selector as user group**

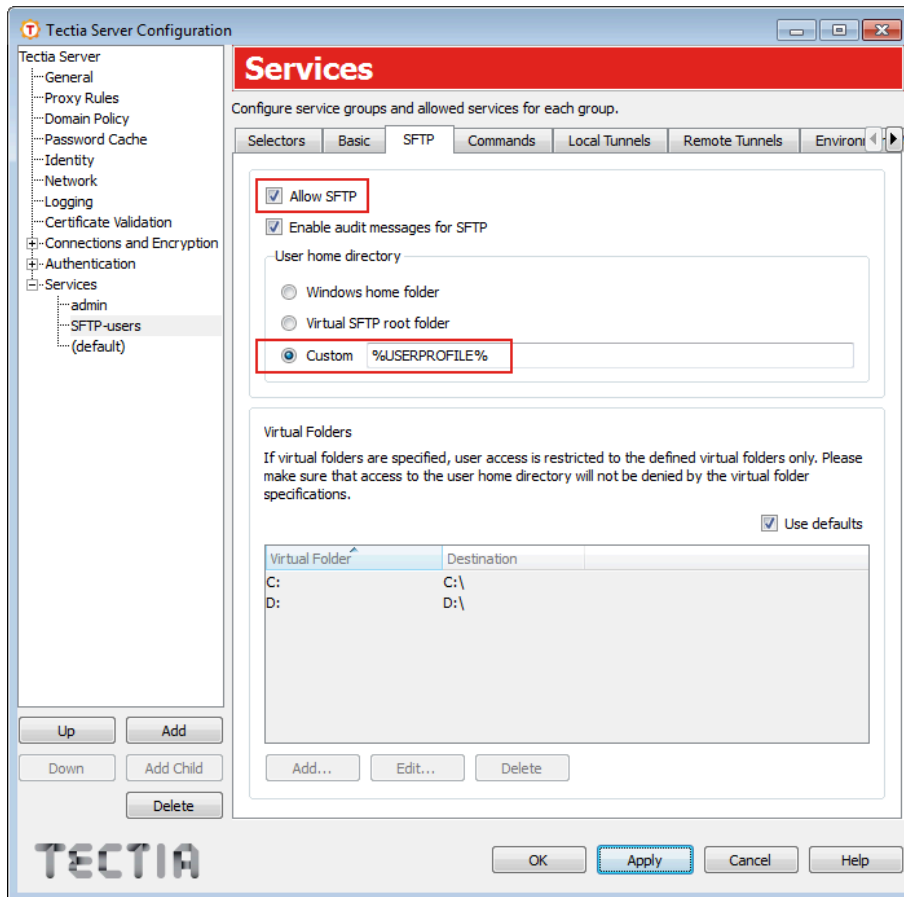
4. When the **User Group Selector** view opens, attach the relevant existing operating-system-related user group (named `staff` in this example) to the group.



**Figure 5.12. Attach user group 'staff'**

Data on the newly created group selectors appears on the **Selectors** tab.

5. On the **SFTP** tab, allow the SFTP service for the `SFTP-users` and define the **User Home Directory** for the user group. This is the SFTP starting directory. Use the default `%USERPROFILES%`, as shown in the following figure.

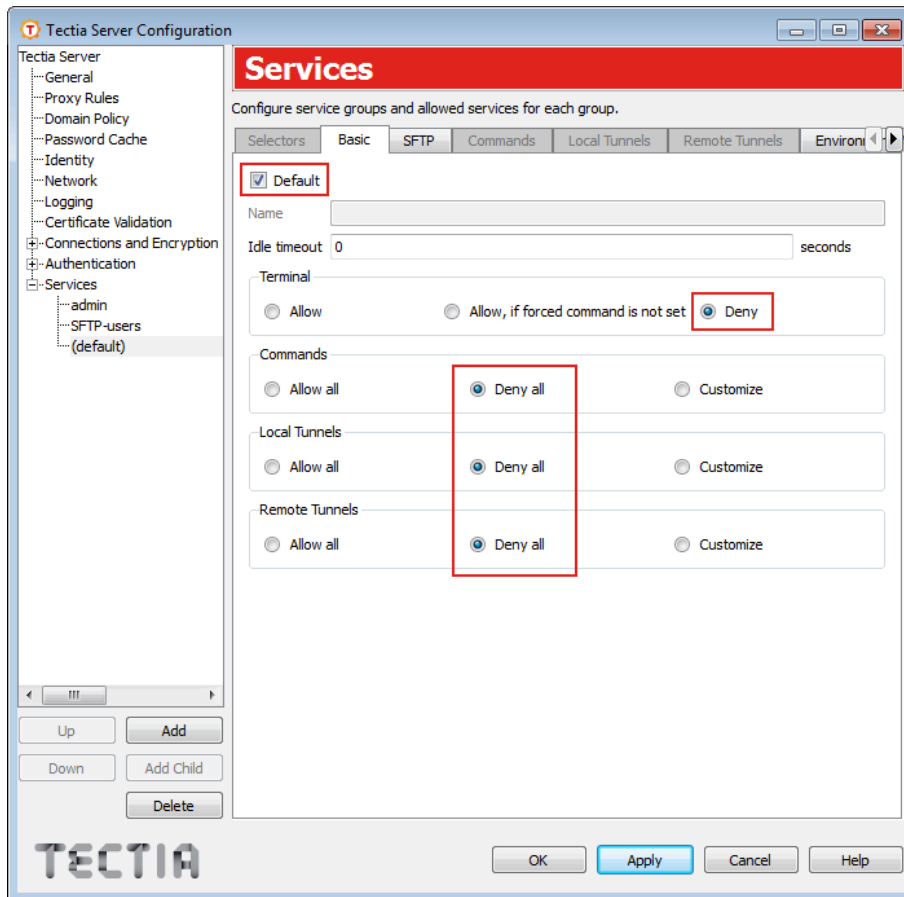


**Figure 5.13. Allow SFTP service for group SFTP-users**

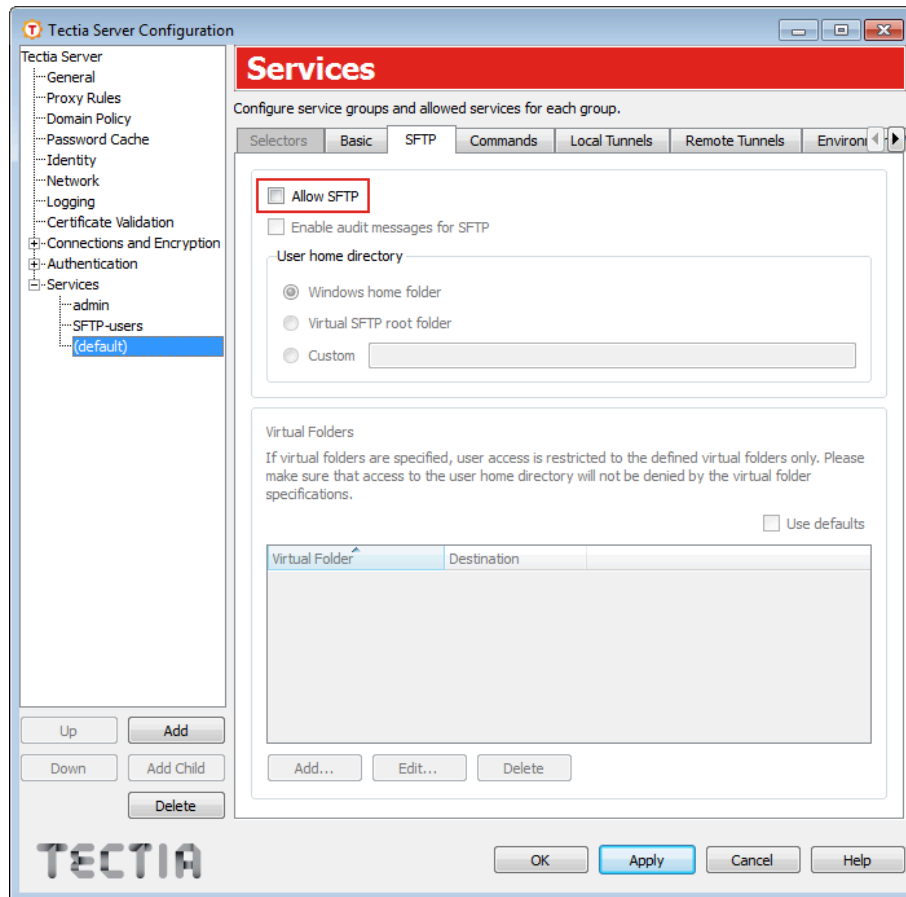
6.

### 5.2.5 Settings for the Rest of Users

The `default` service settings are applied to all users who do not belong to the `admin` group or the `SFTP` group. Deny all services from them on the **Basic** and **SFTP** tabs.



**Figure 5.15. All services denied from default groups**



**Figure 5.16. SFTP service denied from default groups**

Notice that denying the terminal service, denies also X11 and agent forwarding and shell commands for the specified group (unless some commands are explicitly allowed).

## 5.3 Automated Secure File Transfer Script

You can set up automated file transfer between Tectia Client and Server hosts using scripts.

When Tectia Server is used for automated file transfer, separate user accounts can be created for the file transfer users. Non-interactive authentication with public keys and scripted commands are then set for these accounts, and the file transfers are carried out as the current user.

The following example script first transfers `testfile` from Tectia Client to Tectia Server and then transfers the file back. The script logs the command and the return values to a file.

```
@echo off
REM Transfer file from localhost to sftpserver.example.com and back

set SRV=sftpserver.example.com
set logfile=C:\SCP-logs\scpg3_%SRV%
```

```
echo Script started %date% %time% >> %logfile%

REM This 'scp3 put' command transfers the file from client to server.
echo scp3.exe -B -q testfile.dat %SRV%:test >> %logfile%
scp3.exe -B -q testfile.dat %SRV%:test
echo Result: %ERRORLEVEL% >> %logfile%

REM This 'scp3 get' command fetches the file from server to client.
echo scp3.exe -B -q %SRV%:test test >> %logfile%
scp3.exe -B -q %SRV%:test test
echo Result: %ERRORLEVEL% >> %logfile%

echo Script ended %date% %time% >> %logfile%
echo *** >> %logfile%
```



---

## Chapter 6 Using Secure Application Connectivity

This chapter shows how to set up easy application tunneling with pre-configured automatic tunnels for secure e-mail server access. The client machine where the e-mail application is running requires Tectia Client.

The tunneling capability of Tectia is a feature that allows, for example, company employees to access their e-mail, company intranet pages, and shared files securely even when working outside the office.

Tunneling, or port forwarding, is a way of forwarding otherwise unsecured TCP application traffic through Tectia in secure encrypted format. You can secure, for example, POP3, SMTP, and HTTP connections that would otherwise be unsecured.

Tunneling makes it possible to access e-mail from any type of Internet service, whether accessed via modem, GPRS, 3G, a DSL line or a cable connection, or a hotel Internet service. As long as the users have a TCP/IP connection to the Internet, they can get their e-mail and access other resources from anywhere in the world securely.

The Tectia Connection Broker takes care of the tunneling in the background. When the Connection Broker starts up, it opens the listeners for the defined automatic tunnels and asks the user to enter the password or passphrase. If the connections are authenticated with public keys that have empty passphrases, the user does not need to take any actions. The actual tunnel is formed the first time a connection is made to the listener port.

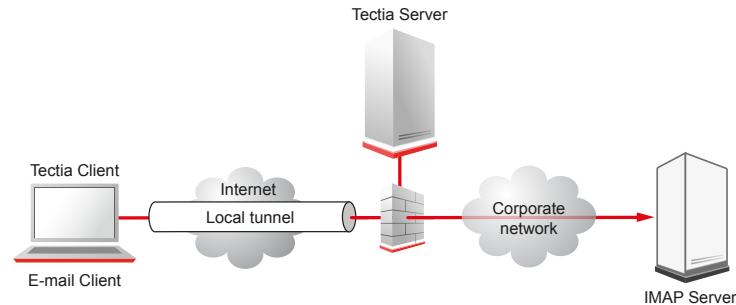


### Note

The user applications using the tunnel will carry out their own authentication procedures (if any) the same way they would without the encrypted tunnel.

The automatic tunnels are local (outgoing) tunnels, which means that they protect TCP connections that your local computer forwards from a specified local port to a specified port on the remote host computer where you are connecting to. It is also possible to forward the connection beyond the remote host computer, but the connection is encrypted only between Tectia Client and the Secure Shell server.

Figure 6.1 shows an example where the Secure Shell server resides in the DMZ network. The connection is encrypted from Tectia Client to the Secure Shell server and continues unencrypted within the corporate network to the IMAP server.



**Figure 6.1. Local tunnel to an IMAP server**

## 6.1 Defining Automatic Tunnels


Automatic tunnels are pre-configured secure connections to servers and the connections are opened automatically when Tectia Client starts up (usually when the session is started). The actual tunnel is formed the first time an application connects to the listener port. If the connection to the server is not open at that time, it will be opened automatically as well.

Automatic tunneling requires settings on Tectia Client and on the application. For instructions on defining the automatic tunnels on Tectia Client, see [Section 6.1.1](#).

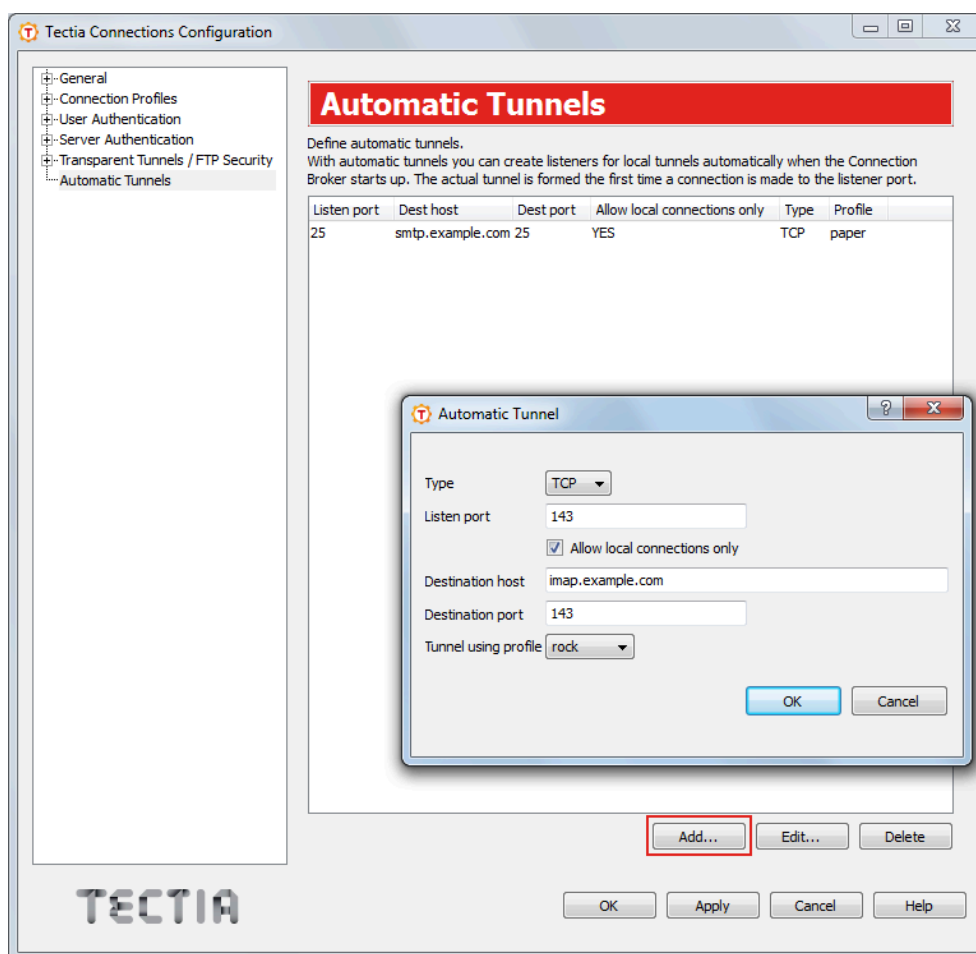
For instructions on defining the automatic tunnels on the application to be tunneled, see [Section 6.1.2](#).

### 6.1.1 Settings in Tectia Client

Automatic tunnels are configured with the **Tectia Connections Configuration GUI**.

Open the tool from the Windows taskbar notification area by right-clicking the Tectia icon  and selecting **Configuration**.

Select **Automatic Tunnels** in the tree menu and click **Add** to open the **Automatic Tunnel** dialog box.



**Figure 6.2. Defining an automatic tunnel**

Fill in the fields as follows:

- **Type:** Select the type of the tunnel from the drop-down list. Available types are TCP and FTP.
- **Listen port:** Define the number of a local port that Tectia Client listens to and that the applications connect to. Do not use a reserved port number.

### Note

The protocol or application for which you wish to create the tunnel may have a fixed port number (for example, 143 for IMAP and 25 for SMTP) that it needs to use to connect successfully. Other protocols or applications may require an offset (for example, 5900 for VNC) that you will have to take into account.

- **Allow local connections only:** Leave this option selected if you want to allow only local connections to be made. This means that other computers will not be able to use the tunnel you created. By default, only local connections are allowed. This is the right choice for most situations. You should carefully consider the security implications if you decide to also allow outside connections.
- **Destination host:** This field defines the destination host for the tunnel.

## Note

The destination host address is resolved after the Secure Shell connection has been established, so here `localhost` means to the Tectia Server host you have connected to.

- **Destination port:** The destination port defines the port to which the tunneled connection is made on the destination host.
- **Tunnel using profile:** Select a connection profile through which the tunnel will be created. See [Section 3.2](#) for instructions on creating connection profiles.

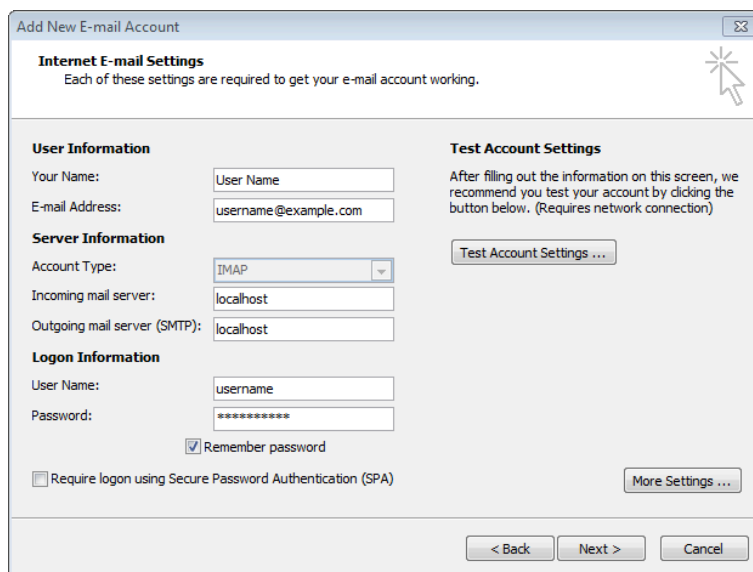
To edit an automatic tunnel, select the tunnel from the list and click **Edit**.

To delete an automatic tunnel, select the tunnel from the list and click **Delete**.

## 6.1.2 Settings in the Tunneled Application

The application (for example, an IMAP and SMTP e-mail, such as Microsoft Outlook) must be configured to connect to the `localhost` port instead of the application server port.

[Figure 6.3](#) shows an example of e-mail account settings in Microsoft Outlook 2007.



**Add New E-mail Account**

**Internet E-mail Settings**  
Each of these settings are required to get your e-mail account working.

**User Information**  
Your Name:   
E-mail Address:

**Server Information**  
Account Type:   
Incoming mail server:   
Outgoing mail server (SMTP):

**Logon Information**  
User Name:   
Password:   
 Remember password  
 Require logon using Secure Password Authentication (SPA)

**Test Account Settings**  
After filling out the information on this screen, we recommend you test your account by clicking the button below. (Requires network connection)

**Figure 6.3. Defining e-mail settings in Microsoft Outlook 2007**

When the tunneled application connects to the `localhost` port, the connection is forwarded in encrypted format to Tectia Server, and from there unencrypted to the application server.

# Index

## A

- application connectivity, 49
- application tunneling, 49
- authentication, 25
  - of server, 25
  - of user, 25, 26
  - with passwords, 25
  - with public keys, 26
- automated file transfer, 46
- automatic tunnels, 50

## C

- client
  - installation, 14
  - uninstallation, 18
- connection profiles, 22
- customer support, 8

## D

- default installation directory, 15
- denying terminal access, 44
- directories
  - virtual,
- documentation, 5
- documentation conventions, 7

## E

- event log, 17

## F

- file transfer, 33
  - automated, 46
- folders
  - virtual,

## G

- generating keys, 27

## H

- home folder, 43
- host, 20
- hostname, 20

## I

- icons, 15
- installation
  - directory, 15
  - preparations, 11
  - removing Tectia products, 17
  - upgrading, 12
- installing
  - client on Windows, 14
  - server on Windows, 15
  - Tectia products, 14

## K

- key file, 29

## L

- license file, 13
- licensing, 13

## M

- Microsoft Windows, 14, 15
- MSI package, 14, 15

## N

- nested tunnel, 24

## O

- online purchase, 13

## P

- port, 20, 24
- port number, 20
- prerequisites, 11
- profile settings, 22
- program group, 15
- program icon, 15
- Programs menu, 15
- public key
  - uploading, 31
- Public-Key Authentication Wizard, 27

## R

- related documents, 5
- removing

- old versions, 12
- software, 17

## S

- secure file transfer, 33
  - configuring it, 34
  - using it, 33

- server
  - installation, 15
  - uninstallation, 18

- settings
  - profile, 22

- SFTP, 33
  - use case, 34

- Start menu, 15
- static tunnels, 50
- support, 8

## T

- technical support, 8
- Tectia Client, 6
- Tectia Server, 7
- Tectia Server Configuration tool, 7
- Tectia Server for IBM z/OS, 7
- terminal access
  - restricting, 44
- terminology, 5
- tunneling, 49

## U

- uninstalling, 17, 18
- upgrading, 11
- uploading a public key, 30
- user, 20
- user account, 13
- user authentication, 25, 26
- username, 20
- user name, 20

## V

- virtual folders, 44

## W

- Windows
  - desktop, 15

- Event Log, 17
- installation, 14, 15
- Start menu, 15
- uninstallation, 18