Kerio WinRoute Firewall 6™

Administrator's Guide

Kerio Technologies

Printing Date: June 16, 2004

This product contains the following open-source libraries:

libiconv Libiconv converts from one character encoding to another through Unicode conversion.

Copyright (c)1999-2003 Free Software Foundation, Inc.

Author: Bruno Haible (`bruno@clisp.org`)

OpenLDAP Open source implementation of the *LDAP* (*Lightweight Directory Access Protocol*).

Copyright (c)1998-2004 The OpenLDAP Foundation.

OpenSSL Toolkit implementing the *Secure Sockets Layer* (SSL v2/v3) and *Transport Layer Security* (TLS v1) protocols.

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

zlib Zlib is a general purpose data compression library.

Copyright (c)1995-2003 Jean-Loup Gailly and Mark Adler.

Contents

Chapter 1

Quick Setup

In this chapter you can find a brief guide for a quick setup of *Kerio WinRoute Firewall* (called briefly *WinRoute* in further text). After this setup the firewall should be immediately available and able to share your Internet connection and protect your local network. For a detailed guide refer to the separate *WinRoute — Step-by-Step Configuration* guide.

If you are not sure how to set any of the *Kerio WinRoute Firewall* functions or features, look up the appropriate chapter in this manual. For information about your Internet connection (such as your IP address, default gateway, DNS server, etc.) contact your ISP.

Note: In this guide, the expression *firewall* represents the host where *WinRoute* is (or will be) installed.

1. The firewall must include at least two interfaces — one must be connected to the local network (i.e. the *Ethernet* or *Token Ring* network adapters), another must be connected to the Internet (i.e. analog modem, ISDN adapter, network adapter or USB Satellite adapter). TCP/IP parameters must be set properly at both/all interfaces.

   Test functionality of the Internet connection and of traffic among hosts within the local network before you run the *WinRoute* installation. This test will reduce possible problems with debugging and error detections.

2. Run *WinRoute* installation. Specify a username and password for access to the administration from the configuration wizard (for details refer to chapters 2.3 and 2.7).

3. Set basic traffic rules using the *Network Rules Wizard* (see chapter 5.1).

4. Run the *DHCP server* and set required IP ranges including their parameters (subnet mask, default gateway, DNS server address/domain name). Read more in chapter 4.4.

5. Check the *DNS Forwarder's* configuration. Define the local DNS domain if you intend to scan the `hosts` file and/or the DHCP server table. For details refer to chapter 4.3.

6. Create or import user accounts and user groups. Set access rights and sort accounts into groups. For details see chapters 10.1 and 10.4.

7. Define IP groups (chapter 9.1), time ranges (chapter 9.2) and URL groups (chapter 9.4), that will be used during rules definition (refer to chapter 9.2).

8. Create URL rules (chapter 6.1) and set the *Cobion* system (chapter 6.3). Set HTTP cache and automatic configuration of browsers (chapter 4.6). Define FTP rules (chapter 6.5).

9. Select an antivirus and define types of objects that will be scanned. If you choose the integrated *McAfee* antivirus application, check automatic update settings and edit them if necessary.

10. Using one of the following methods set TCP/IP parameters for the network adapter of individual LAN clients:

    - *Automatic configuration* — activate the *Obtain an IP address automatically* option. Do not set any other parameters.

    - *Manual configuration* — define IP address, subnet mask, default gateway address, DNS server address and local domain name.

    Use one of the following methods to set the Web browser at each workstation:

    - *Transparent configuration* — by default *WinRoute* will filter all outgoing HTTP traffic through the HTTP protocol inspector. This does not require any configuration to the Web browser of the workstations.

    - *Automatic configuration* — activate the *Automatically detect settings* option (*Microsoft Internet Explorer*) or specify URL for automatic configuration (other types of browsers). For details refer to chapter 4.6.

    - *Manual configuration* — select type of connection via the local network or define IP address and appropriate proxy server port (see chapter 4.5).

Chapter 2

Introduction

2.1 Kerio WinRoute Firewall 6.0

Kerio WinRoute Firewall 6.0 is a complex tool for connection of the local network to the Internet and protection of this network from intrusions. It is designed for Windows NT 4.0, 2000 and XP operating systems.

Basic Features

Transparent Internet Access With Network Address Translation (NAT) technology, the local private network can be connected to the Internet through a single public IP address (static or dynamic). Unlike proxy servers, with NAT technology all Internet services will be accessible from any workstation and it will be possible to run most standard network applications, as if all computers within the LAN had their own connection to the Internet.

Security The integrated firewall protects all the local network including the workstation it is installed on, regardless of whether the NAT function (IP translation) is used or *WinRoute* is used as a "neutral" router between two networks. *Kerio WinRoute Firewall* offers the same standard of protection found in much more costly hardware solutions.

Access Control All the security settings within *WinRoute* are managed through so-called traffic policy rules. These provide effective network protection from external attacks as well as easy access to all the services running on servers within the protected local network (e.g. Web Server, Mail server, FTP Server, etc.). Communication rules in the traffic policy can also restrict local users in accessing certain services on the Internet.

Protocol Maintenance (Protocol Inspectors) You may come across applications that do not support the standard communication and that may for instance use incompatible communication protocols, etc. To challenge this problem, *WinRoute* includes so-called protocol inspectors, which identify the appropriate application protocol and modify the firewall's behavior dynamically, such as temporary access to a specific port (it can temporarily open the port demanded by the server). FTP in the active mode, Real Audio or PPTP are just a few examples.

Content Filtering *WinRoute* can monitor all HTTP and FTP communication and block objects that do not match given criteria. The settings can be global or defined specifically for each user. Downloaded objects can also be transparently checked by an external anti-virus application.

Network Configuration *WinRoute* has a built-in DHCP server, which sets TCP/IP parameters for each workstation within your local network. Parameters for all workstations can be set centrally from a single point. This reduces the amount of time needed to set up the network and minimizes the risk of making a mistake during this process.

DNS forwarder module enables easy DNS configuration and faster responses to DNS requests. It is a simple type of caching nameserver that relays requests to another DNS server. Responses are stored in its cache. This significantly speeds up responses to frequent requests. Combined with the DHCP server and the system's `HOSTS` file, the *DNS forwarder* can be also used as a dynamic DNS server for the local domain.

Remote Administration All settings are performed in the *Kerio Administration Console*, an independent administration console used to manage all Kerio's server products. It can be run either on the workstation with *WinRoute* or on another host within the local network or the Internet. Communication between *WinRoute* and the administration console is encrypted and thus protected from being tapped or misused.

Various Operating Systems Within The Local Network *WinRoute* works with standard TCP/IP protocols. From the point of view of workstations within the local network it acts as a standard router and no special client applications are required. Therefore, any operating system with TCP/IP, such as Windows, Unix/Linux, Mac OS etc., can be run within the LAN.

Note: WinRoute can work with TCP/IP protocol sets only. It does not affect the functionality of other protocols (i.e. IPX/SPX, NetBEUI, AppleTalk, etc.).

Additional Features

Antivirus check *WinRoute* can perform antivirus check of transmitted files. For this purpose, either the built-in *McAfee* antivirus or an external antivirus program (e.g. *NOD32*, *AVG*, etc.) are available. Antivirus check can be applied to *HTTP*, *FTP*, *SMTP* and *POP3* protocols.

Email notifications *WinRoute* can send email notifications informing users about various events. This function makes firewall administration easier for the administrators since they need not connect to *WinRoute* frequently to check it through. All sent notifications are saved in a special log file.

User quotas A limit can be set for transmitted data per each user. This limit can be set for the amount of downloaded or/and uploaded data per day/month. These limits are called quotas. If any quota is exceeded, the connection to the Internet will be blocked for a corresponding user. Email notification can be optionally sent to the user.

Blocking of P2P networks *WinRoute* can detect and block so called peer-to-peer networks (networks used for sharing of files, such as *Kazaa*, *DirectConnect* etc.).

Statistics Detailed statistics of the firewall interface (current speed of transmitted data, amount of data transmitted in certain time periods) as well as of individual users (amount of transmitted data, used services, categories of connected Websites, etc.) can be viewed in *WinRoute*.

Proprietary VPN server and client *WinRoute* also provides a proprietary VPN solution which can be applied to the *server-to-server* and *client-to-server* modes. This VPN solution can perform NAT (even multiple) at both ends. The *Kerio VPN Client* client software is included in the *WinRoute* package that can be used for creation of *client-to-server* VPN types (connection of remote clients to local networks).

2.2 Conflicting Software

The *WinRoute* host can be used as a workstation, however it is not recommended as user activity can affect the functionality of the operating system and *WinRoute* in a negative way.

WinRoute can be run with most of common applications. However, there are certain applications that should not be run at the same host as *WinRoute* for this could result in collisions.

Collision of low-level drivers *WinRoute Firewall* may collide with applications that use low-level drivers with either identical or similar technology. The following applications are typical:

- Application for Internet connection sharing — e.g. *Microsoft Internet Connection Sharing*, *Microsoft Proxy Server*, *Microsoft Proxy Client*, etc.
- Network firewalls — i.e. *Microsoft ISA Server*, *CheckPoint Firewall-1*, *WinProxy* (by Ositis), *Sygate Office Network* and *Sygate Home Network*, etc.

- Personal firewalls — i.e. *Kerio Personal Firewall*, *Internet Connection Firewall* (included in Windows XP), *Zone Alarm*, *Sygate Personal Firewall*, *Norton Personal Firewall*, etc.

- Software designed to create virtual private networks (VPN) — i.e. software applications developed by the following companies: CheckPoint, Cisco Systems, Nortel, etc. There are many such applications and their features vary from vendor to vendor.

  Under proper circumstances, use of the VPN solution included in *WinRoute* is recommended (for details see chapter 12). Otherwise, we recommend you to test a particular VPN server or VPN client with *WinRoute* trial version or to contact our technical support (`http://www.kerio.com/`).

  Note: VPN implementation included in Windows operating system (based on Microsoft's PPTP protocol) is supported by *WinRoute*.

Port collision Applications that use the same ports as the firewall cannot be run at the *WinRoute* host (or the configuration of the ports must be modified). If all services are running, *WinRoute* uses the following ports:

- `53/UDP` — *DNS Forwarder*
- `67/UDP` — *DHCP server*
- `1900/UDP` — *SSDP Discovery* service
- `2869/TCP` — *UPnP Host* service

  The two recently mentioned services belong to the UPnP support (see chapter 11.6).

- `3128/TCP` — HTTP proxy server (see chapter 4.5)
- `44333/TCP+UDP` — traffic between *Kerio Administration Console* and *WinRoute Firewall Engine*. This service cannot be stopped.

The following services use corresponding ports by default. Ports for these services can be changed.

- `3128/TCP` — HTTP proxy server (see chapter 4.5)
- `4080/TCP` — Web administration interface (refer to chapter 8)

- `4081/TCP` — secured (SSL-encrypted) version of the Web administration interface (see chapter 8)
- `4090/TCP+UDP` — proprietary VPN server (for details refer to chapter 12)

Antivirus applications If an antivirus application that scans files on the disc is run on the *WinRoute* host, the HTTP cache file (see chapter 4.6, usually the `cache` subdirectory under the directory where *WinRoute* is installed) and the `tmp` subdirectory (used to scan HTTP and FTP objects) must be excluded from inspection. If the antivirus is run manually, there is no need to exclude these files, however, *WinRoute Firewall Engine* must be stopped before running the antivirus (this is not always desirable).

Note: If *WinRoute* uses an antivirus to check objects downloaded via HTTP or FTP protocols (see chapter 7), the cache directory can be excluded with no risk — files in this directory have already been checked by the antivirus.

2.3 Installation

System Requirements

Requirements on minimal hardware parameters of the host where *WinRoute* will be installed:

- CPU Intel Pentium II or compatible; 300 MHz
- 128 MB RAM
- 2 network interfaces
- 50 MB disc space free for the installation
- Free memory for logs (depends on traffic load and selected logging level)

The product supports for the following operating systems:

- Windows 2000
- Windows XP
- Windows Server 2003

Note: The *Client for Microsoft Networks* component must be installed for all supported operating systems, otherwise *WinRoute* will not be available as a service and NTLM authentication will not function. The component is included in installation packages of all supported operating systems.

Steps to be taken before the installation

Install *WinRoute* on a computer which is used as a gateway connecting the local network and the Internet. This computer must include at least one interface connected to the local network (Ethernet, TokenRing, etc.) and at least one interface connected to the Internet. You can use either a network adapter (Ethernet, WaveLAN, etc.) or a modem (analog, ISDN, etc.) as an Internet interface.

We recommend you to check through the following items before you run *WinRoute* installation:

- Time of the operating system should be set correctly (for timely operating system and antivirus upgrades, etc.)
- The latest service packs and any Microsoft recommended security updates should be applied.
- TCP/IP parameters should be set for all available network adapters
- All network connections (both to the local network and to the Internet) should function properly. You can use for example the `ping` command to detect time that is needed for connections.

These checks and pre-installation tests may protect you from later problems and complications.

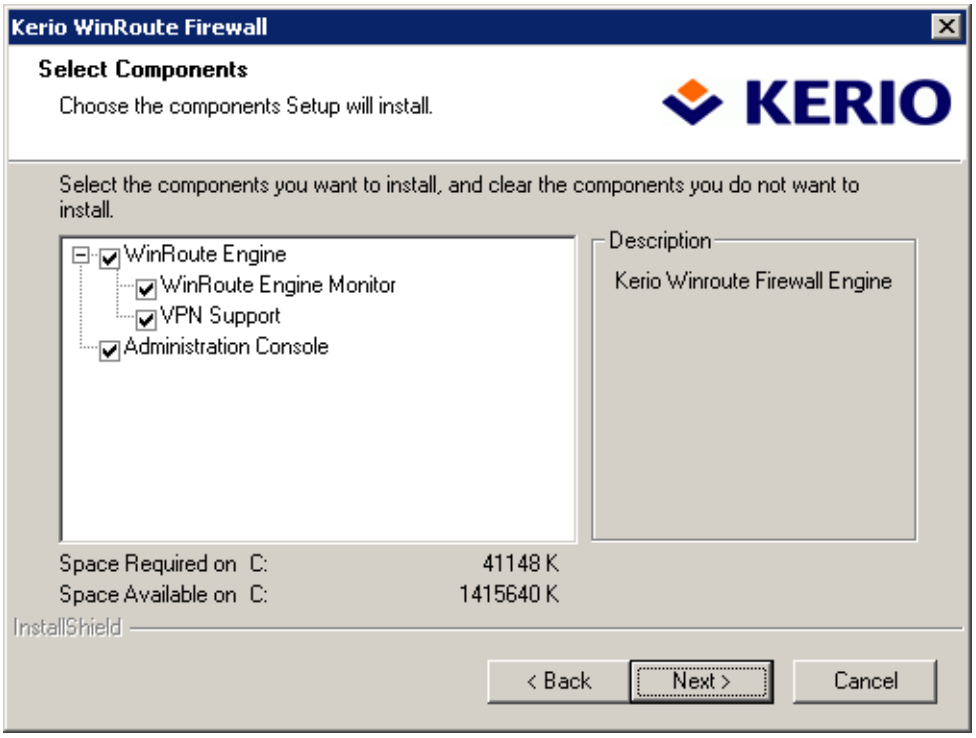
Note: Basic installation of all supported operating systems include all components required for smooth functionality of *WinRoute*.

Installation and Basic Configuration Guide

Once the installation program is launched (i.e. through `kerio-kwf-mcafee-6.0.0-win.exe`), a guide will take you through setting the basic firewall parameters.

You will be asked to choose between two types of installation — *Full* or *Custom*. Choosing the custom mode will let you select *WinRoute's* individual components (see also chapter 2.4).

- *WinRoute Firewall Engine* — core of the application
- *WinRoute Engine Monitor* — utility for *WinRoute Firewall Engine* control and monitoring its status (icon in the system's notification area)



- *VPN Support* — proprietary VPN solution developed by Kerio Technologies,
- *Kerio Administration Console* — the *Kerio Administration Console* application (universal console for all server applications of Kerio Technologies)

Go to chapter 2.4 for a detailed description of all *WinRoute* components. For detailed description on the proprietary VPN solution, refer to chapter 12.

Note: If you selected the *Custom* installation mode, the behavior of the installation program will be as follows:

- all checked components will be installed or updated
- all checked components will not be installed or will be removed

During an update, all components that are intended to remain must be ticked.

Having completed this step, you can start the installation process. All files will be copied to the hard disk and all the necessary system settings will be performed. The initial Wizard will be run automatically after your first login (see chapter 2.7).

Restart the machine when the installation has completed. This will install the *WinRoute* low-level driver into the system kernel. *WinRoute Engine* will be automatically launched after restart. The engine runs as a service. The *WinRoute Engine Monitor* will be launched

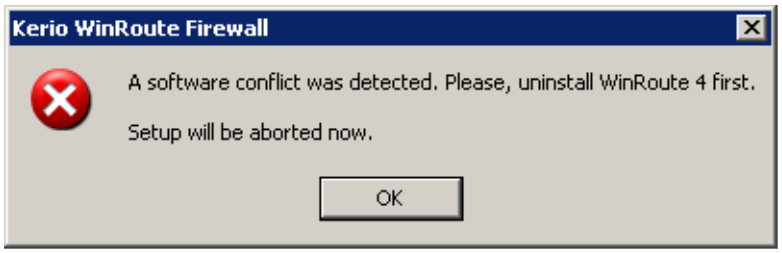
after a user login. This utility monitors the *Engine* status and is used to start or stop the engine. *WinRoute Engine Monitor* icon is displayed in the system's notification area (system tray).

Conflicting Applications and System Services

The *WinRoute* installation program detects applications and system services that might conflict with the *WinRoute Firewall Engine*.

1. *Kerio WinRoute Pro* and*Kerio WinRoute Lite*

   WinRoute is no longer compatible with versions *4.x*. If *WinRoute Pro 4.x* or *WinRoute Lite 4.x* is detected during the installation, the following error will be reported:

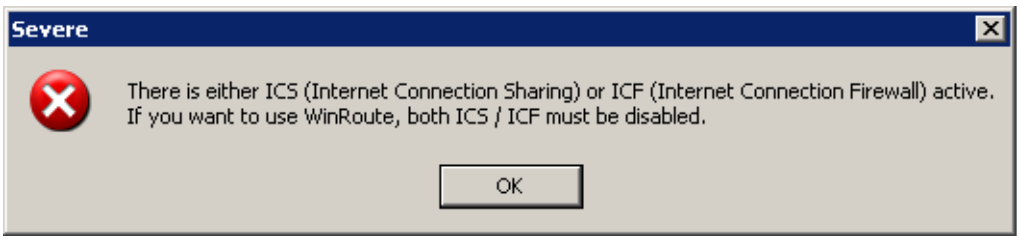
   

   Click *OK* to close the installation. Uninstall *WinRoute Pro/Lite* using the *Add/Remove programs* option in the *Control Panels*), restart the system and start the installation again.

   Note: If your *WinRoute Pro* configuration is to be used in *Kerio WinRoute Firewall 6.x* (e.g. when there are many user accounts), upgrade your firewall to the version *5.x* first, then perform the upgrade from *Kerio WinRoute Firewall 5.x* to *Kerio WinRoute Firewall 6.x* (for details, refer to chapter 2.6).

2. *Internet Connection Sharing* and *Internet Connection Firewall*

   If *Internet Connection Sharing* or *Internet Connection Firewall* (Windows XP or 2003 Server) is running at any interface of the *WinRoute* host, the following warning will reported:

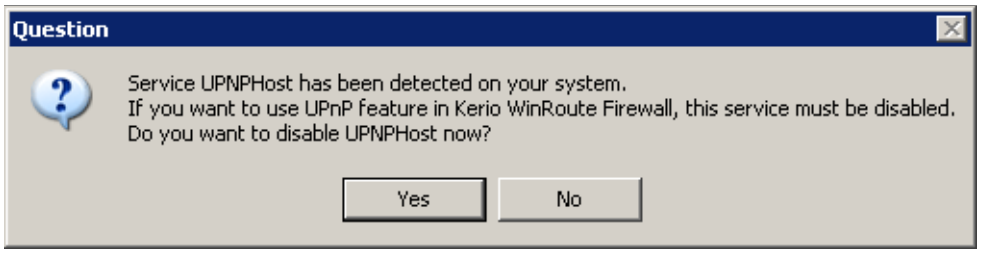
   

Do not proceed unless both services are disabled on all interfaces. *WinRoute* will not function correctly if this condition is not met.

3. *Universal Plug and Play Device Host* and *SSDP Discovery Service*

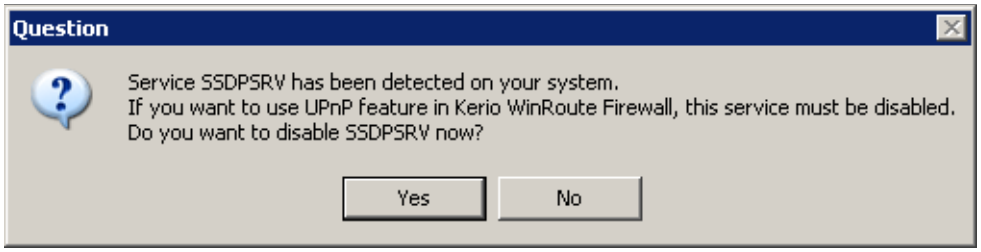
   These two services support the *UPnP* (Universal Plug and Play) protocol on the Windows XP and 2003 Server operating systems. Both services must be disabled if you intend to use *UPnP* in *WinRoute* (see chapter 11.6).

   - If the *Universal Plug and Play Device Host* service is detected, the following dialog will appear:



     Click on *Yes* to stop the *Universal Plug and Play Device Host* service and to disable its automatic startup when the system is started. Select *No* to keep existing service status and parameters.

   - If the *SSDP Discovery Service* service is detected, the following dialog will appear:



     The action of the buttons is the same as described above.

   Note: To read more about *UPnP* refer to chapter 11.6.

2.4 WinRoute Components

Kerio WinRoute consists of the three following components:

WinRoute Firewall Engine The core of the program executing all services and functions. It is running as a service in the operating system (the service is called *Kerio WinRoute Firewall* and it is run automatically within the system account by default).

WinRoute Engine Monitor With this application you can monitor the *Engine* and/or *Monitor* applications, you can switch the engine's on/off status, edit startup preferences or launch the administration console. For more info see chapter 2.5.

Note: WinRoute Firewall Engine is independent on the *WinRoute Engine Monitor*. This means that the *Engine* can be run even when the icon is not displayed on the toolbar.

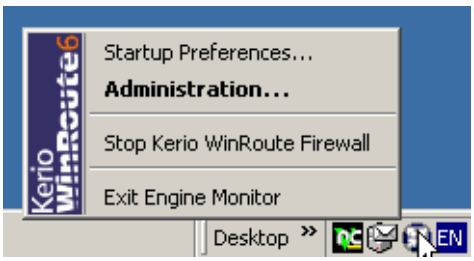
Kerio Administration Console It is a versatile console for local or remote administration of Kerio server products. For successful connection to an application you need a plug-in with an appropriate interface. *Kerio Administration Console* is installed hand-in-hand with the appropriate module during the installation of *Kerio WinRoute*. Refer to the *Kerio Administration Console — Help* document to see how *Kerio Administration Console* can be used for *Kerio WinRoute* administration.

2.5 WinRoute Engine Monitor

WinRoute Engine Monitor is a utility used to control and monitor the *WinRoute Engine* status. The icon of this component is displayed on the systray.

If *WinRoute Engine* is stopped, a white crossed red spot appears on the icon. Under different circumstances, it can take up to a few seconds to start or stop the *WinRoute Engine* application. Meanwhile, the icon gets grey and is inactive — does not respond to mouse clicking.

By double-clicking on the icon with the left mouse button, *Kerio Administration Console* (see below) will be launched. By clicking the right mouse button on the icon, a menu offering the following functions will be displayed:



Startup Preferences With these options *WinRoute Engine* and/or *WinRoute Engine Monitor* applications can be set to be launched automatically when the operating system is started. In default settings (after the installation) both functions are on.

Administration This option starts *Kerio Administration Console*. The application can be also started by double-clicking on the *WinRoute Engine Monitor* icon with the left mouse button.

Start / Stop WinRoute Engine Switches between the Start and Stop modes. The text displays the current mode status.

Exit Engine Monitor An option to exit *WinRoute Engine Monitor*. It does not affect status of the *WinRoute Engine* application (this will be announced by a report).

2.6 Upgrade and Uninstallation

In this chapter you can find a description of *WinRoute* upgrade within the versions `5.x` and `6.x` (i.e. upgrade from the `5.1.10` version to the `6.0.0` version or from `6.0.0` to `6.0.1`). Direct upgrade from `4.x` versions or earlier to the `6.x` version is not supported.

Simply run the installation of a new version to upgrade *WinRoute* (i.e. to get a new release from the *Kerio* Web pages — `http://www.kerio.com/`).

All windows of the *Kerio Administration Console* must be closed before the (un)installation is started. All of the three *WinRoute* components will be stopped and closed automatically.

The installation program detects the directory with the former version and updates it by replacing appropriate files with the new ones automatically. All logs and user defined settings are saved.

Warning: We strongly recommend you not to change the installation directory!

To uninstall *WinRoute*, stop all three *WinRoute* components. The *Add/Remove Programs* option in the *Control Panel* launches the uninstallation process. All files under the *WinRoute*

(`C:\Program Files\Kerio\WinRoute Firewall`, by default)

directory can be optionally deleted.

Upgrade from WinRoute Pro 4.x

To import your configuration used in *WinRoute Pro 4.x* to the *Kerio WinRoute Firewall 6.x*, follow these steps:

1. Upgrade the *WinRoute Pro 4.x* to the *Kerio WinRoute Firewall 5.x*. Version 5.x includes a tool for initial configuration, which is able to read and translate the configuration from the *WinRoute Pro 4.x*.
2. Upgrade version 5.x to version 6.x (see above).

Note: This method of upgrade is not recommended. Do not use it unless necessary (e.g. a great amount of user accounts to be imported). Configuration parameters of the *WinRoute Pro 4.x* have crucial differences and only some of the parameters can be imported. Later revisions and error removals might be more exigent than a brand new configuration.

Update Checker

WinRoute enables automatic checks for new versions of the product at the *Kerio Technologies* website. Whenever a new version is detected, its download and installation will be offered automatically.

For detailed information refer to chapter 11.8.

2.7 Configuration Wizard

Using this Wizard you can define all basic *WinRoute* parameters. It is started automatically by the installation program.

Note: In any language version, the configuration wizard is available in English only.

Administrator Account Password

Definition of the administration password is essential for the security of the firewall. Do not use the standard (blank) password, otherwise unauthorized users may be able to access the *WinRoute* configuration.

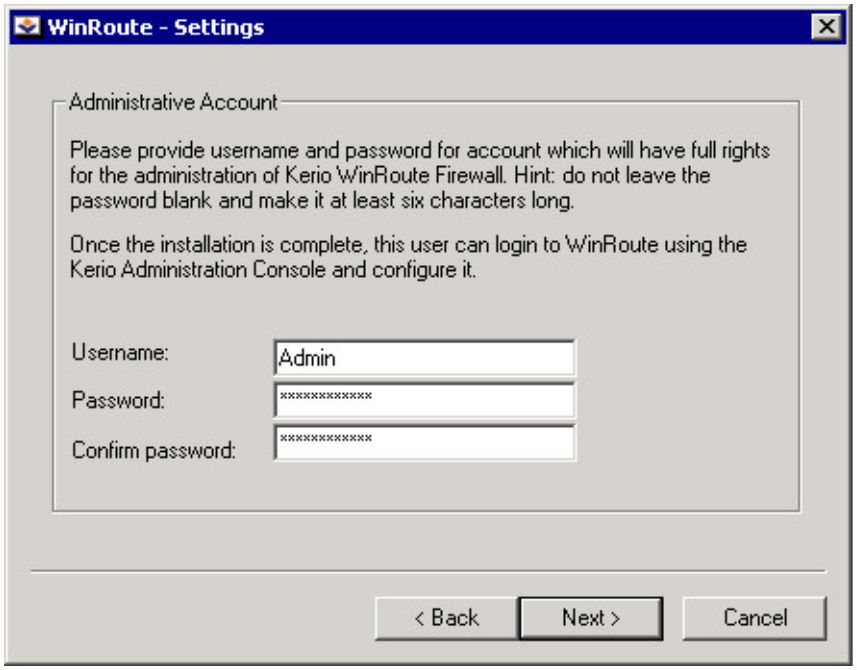
In the dialog window for the administrative account settings define your *Password* and confirm the definition in the *Confirm Password* text field. The administrator's username (`Admin` is used as default) can be edited in the *Username* text field.

Note: If *WinRoute* is upgraded from *WinRoute Pro 4.x*, skip this step and import the administrative account from *WinRoute Pro 4.x* (see below).

Remote Access

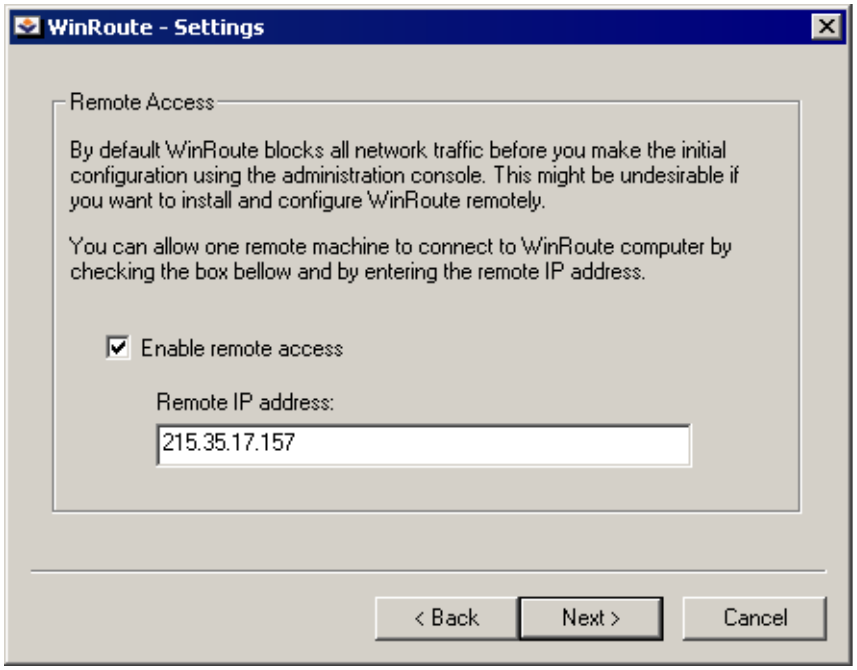
Immediately after the first *WinRoute Firewall Engine* startup all network traffic will be blocked (desirable traffic must be permitted by traffic rules — see chapter 5). If *WinRoute* is installed remotely (i.e. using terminal access), communication with the remote client will be also interrupted immediately (*WinRoute* must be configured locally).

Within Step 2 of the configuration wizard specify the IP address of the host from which the firewall will be controlled remotely (i.e. using terminal services) to enable remote



installation and administration. Thus *WinRoute* will enable all traffic between the firewall and the remote host.

Note: Skip this step if you install *WinRoute* locally.



Enable remote access

This option enables full access to the *WinRoute* computer from a selected IP address

Remote IP address

IP address of the computer from where you will be connecting (e.g. terminal services client). This field must contain an IP address. A domain name is not allowed.

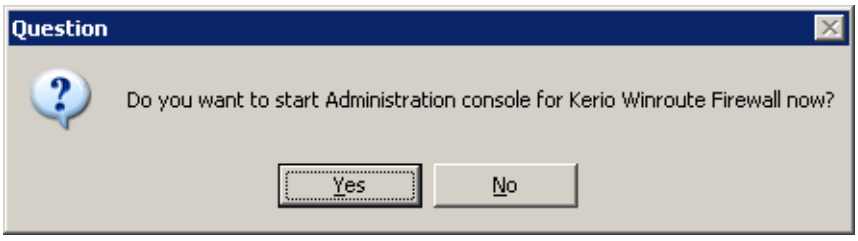
Notice: After *WinRoute* has been remotely configured, the rule allowing remote access will be removed.

Restart of the operating system

When the installation is completed successfully, the operating system must be restarted for the *WinRoute* low-level driver to be implemented (`wrdrv.sys`).

After the restart, the *WinRoute Firewall Engine* service and the *WinRoute Engine Monitor* will be launched automatically.

After the *WinRoute Firewall Engine* is started for the first time (immediately after the installation), the user will be asked whether the *Kerio Administration Console* should be started. It is recommended, since it is necessary to perform at least the basic configuration of the console (see chapter 5.1), otherwise all network traffic of the *WinRoute* host will be blocked.



Chapter 3

WinRoute Administration

All Kerio products including *WinRoute* are administered through the *Kerio Administration Console* application. Using this program you can access *WinRoute Firewall Engine* either locally (from the *Engine* host) or remotely (from another host). Traffic between *Kerio Administration Console* and *WinRoute Firewall Engine* is encrypted. This protects you from tapping and misuse.

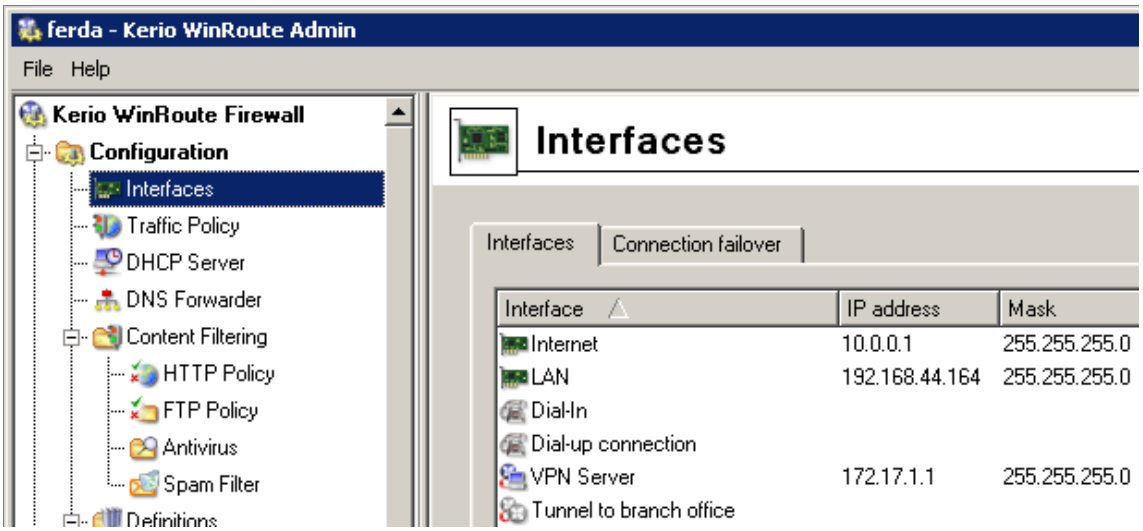
The *Kerio Administration Console* is installed along with *WinRoute* (see chapters 2.3 and 2.4). Refer to the *Kerio Administration Console — Help* document to see how it works.

The following chapters of this guide provide descriptions on individual sections of the *WinRoute* administration dialog window which is opened upon a successful login to the *WinRoute Firewall Engine*.

Note: Upon the first login to *WinRoute* after a successful installation, the traffic rules wizard is run so that the initial *WinRoute* configuration can be performed. For a detailed description on this wizard please refer to chapter 5.1.

3.1 Administration dialog window

The main *WinRoute* administration dialog window ("administration window") will be opened upon a successful login to the *WinRoute Firewall Engine* through the *Kerio Administration Console*. This window consists of two sections:



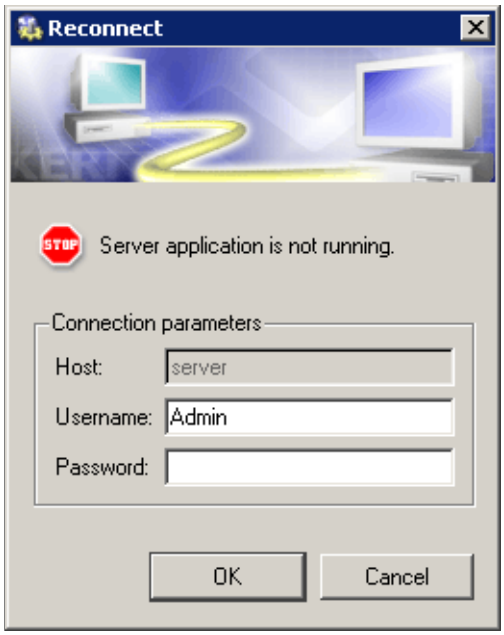
- The left column provides a structured list (tree) of sections of the administration window. Individual items (sections) and their subsections can be viewed and hidden by simple clicks. Whenever the *Kerio Administration Console* is closed, the current status of the tree is saved and kept for the following connection to the window.
- The right section of the main window shows contents of a section selected in the left column (or a list of subsections included in the selected item).

Main menu of the administration window

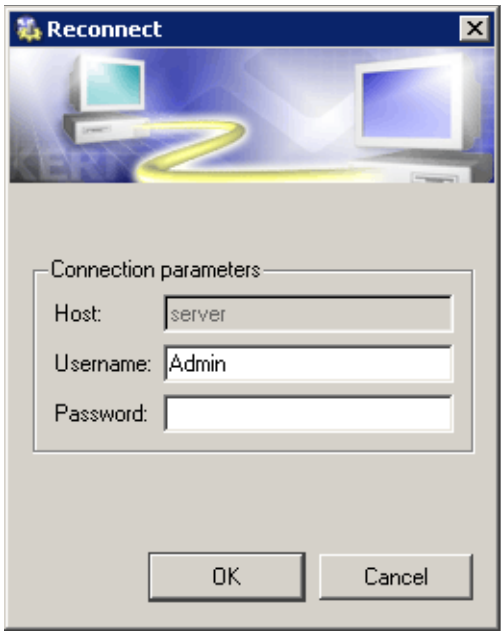
The main menu provides the following options:

File

- *Reconnect* — reconnection to the *WinRoute Firewall Engine* after a connection drop-out (caused for example by a restart of the *Engine* or by a network error). For security reasons, a username and password authentication is required for re-connections.



Note: The *Administration Console* automatically detects connection drop-outs. Drop-outs are usually detected during data download/upload from or to the server (i.e. when the *Apply* button is pressed or when a user switches between individual sections of the *Administration Console*). In such cases, the authentication dialog along with an error report is opened automatically.



- *New Connection* — this option can be useful when the console will be used for administration of multiple server applications (e.g. *WinRoute* at multiple servers). The *New Connection* option opens the main dialog window of the *Kerio Administration Console*. This window provides a menu of possible connections as well as an option through which a new connection can be set (for details refer to the *Kerio Administration Console — Help* document).

  The *New Connection* option opens the same dialog as running the *Kerio Administration Console* from the *Start* menu.

- *Quit* — terminates the session (this option logs the user out of the server and closes the administration window). The session may also be closed by clicking on the cross in the right corner at the top of the window or by pressing *Alt+F4*.

Help

- *Administrator's Guide* — this option opens the administrator's guide in the *HTML Help* format. For details concerning help files refer to the *Kerio Administration Console — Help* document.

- *About* — this page provides information about current version of the application (*WinRoute's* administration module in this case), a link to our company's website, etc.

Status bar

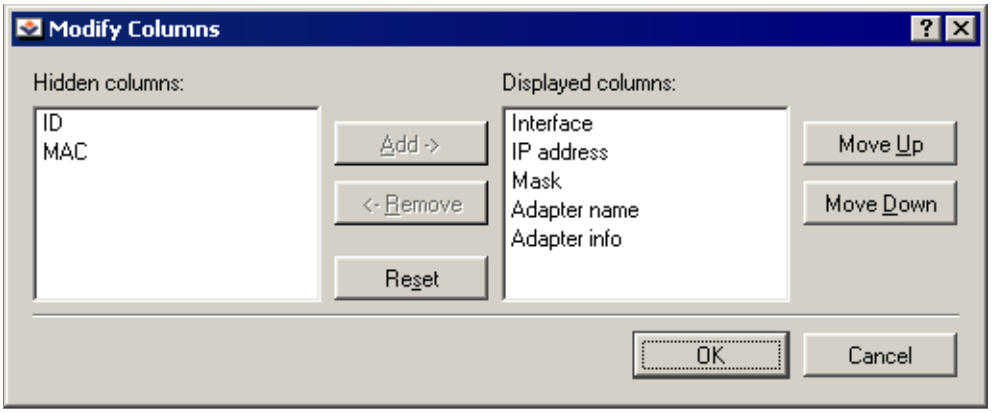
At the bottom of the administration window, there is a status bar which provides the following information:

- Opened section of the administration (selected in the left column). This information is for reference and it can be helpful for example when any part of the tree is not visible (when a lower screen resolution is used, for example).
- Name or IP address of the server and port of the server application (*WinRoute* uses port 44333).
- Name of the currently connected user.
- Current status of the *Kerio Administration Console*: *Ready* (waits for user's response), *Loading* (loading data from the server) or *Saving* (saving changes to the server).

3.2 Views Setup

Many sections of the *Kerio Administration Console* are in table form where each line represents one record (e.g. detailed information about user, information about interface, etc.) and the columns consist of individual entries for these records (e.g. name of server, MAC address, IP address, etc.).

WinRoute administrators can define — according to their liking — the way how the information in individual sections will be displayed. By clicking the right mouse button in any of the sections listed, a context menu will be displayed. It includes the *Modify Columns* entry. This entry opens a dialog window where users can select which columns will or will not be displayed.



The *Hidden Columns* field contains columns that are intended to be hidden whereas *Displayed Columns* contains columns that are to be displayed. By clicking on the *Add* button, selected columns will be moved from the "hidden" to the "displayed" group. Clicking on the *Remove* button will do the opposite. Clicking on the *Refresh* button will restore default settings.

The *Move Up* and *Move Down* buttons move the selected column up and down within the group. This allows the administrator to define the order the columns will be displayed.

The order of the columns can also be adjusted in the window view. Left-click on the column name, hold down the mouse button and move the column to the desired location.

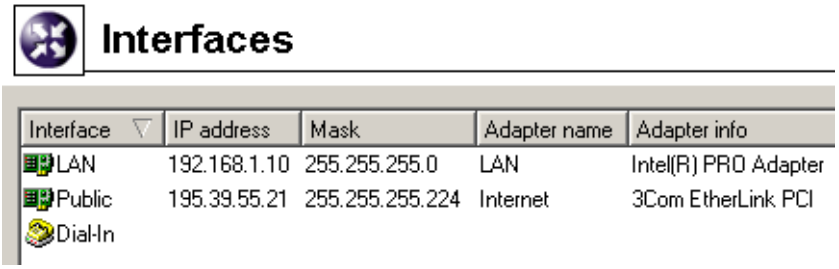
The width of individual columns can be adjusted by moving the dividing line between the column headers.

Chapter 4

Settings for Interfaces and Network Services

4.1 Interfaces

WinRoute functions as a router for all *WinRoute's* network interfaces installed within the system. The interfaces are listed in the *Configuration / Interface* section of the *WinRoute Administration Console*.



Interface The name used for interface identification within *WinRoute*. It should be unique for easy reference, e.g. *Internet* for the interface connected to the Internet connection. We recommend you not to use duplicate interface names as they could cause problems during traffic policy definitions or routing table modifications.

The name can be edited later (see below) with no affect on *WinRoute's* functionality.

The icon to the left of the name represents the interface type (network adapter, dial-up connection, satellite connection, VPN server, VPN tunnel).

Note: Unless the name is edited manually, this item displays the name of the adapter as assigned by the operating system (see the *Adapter name* entry).

IP Address and Mask IP address and the mask of this interface's subnet.

Adapter name The name of the adapter (e.g. "LAN connection 2"). The name is for reference only.

Adapter info Adapter identification string returned by the device driver.

ID A unique identifier of the adapter in the operating system (see also chapter 17.1).

MAC Hardware (MAC) address of a corresponding network adapter.

Use the buttons at the bottom of the interface list to remove or edit properties of the chosen interface. If no interface is chosen or the selected interface does not support a certain function, appropriate buttons will be inactive.

Add Adds a new dial-up interface or a VPN channel (see below).

If a new network adapter is added it must be installed and configured in the operating system first in order to for *WinRoute* to detect it automatically.

Edit Displays detailed information and enables editing of the interface's parameters.

Remove Removes the selected interface from *WinRoute*. This can be done under the following conditions:

- the dial-up is hung-up
- the network adapter is not active or it is not physically present

WinRoute does not allow removing an active network or dial-up adapter.

Dial or Hang Up You can use *WinRoute's* Web interface (see chapter 8) to dial or hang up lines. If a network adapter is selected, these buttons are inactive.

Refresh Use this button to refresh the list of interfaces.

Special interfaces

In addition to network adapters, the following two interfaces are provided in the *Interfaces* section:

Dial-In This interface represents the server of the *RAS* service (dial-up connection to the network) on the *WinRoute* host. This interface can be used for definition of traffic rules (see chapter 5) for RAS clients which are connecting to this server.

The *Dial-In* interface cannot be configured or removed.

Notes:

1. If both the RAS server and *WinRoute* are used, the RAS server must be configured so that it assigns clients IP addresses from a subnet which is not used in any segment of the local network, otherwise standard IP routing will not be performed properly.

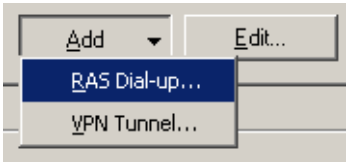
2. *WinRoute* DHCP server can be used for assigning IP addresses to RAS clients (see chapter 4.4).

VPN server This interface represents a server which provides a connection for the proprietary VPN client of Kerio Technologies. Double-click on this interface or click on *Edit* to edit settings and parameters of the VPN server. The *VPN server* interface cannot be removed.

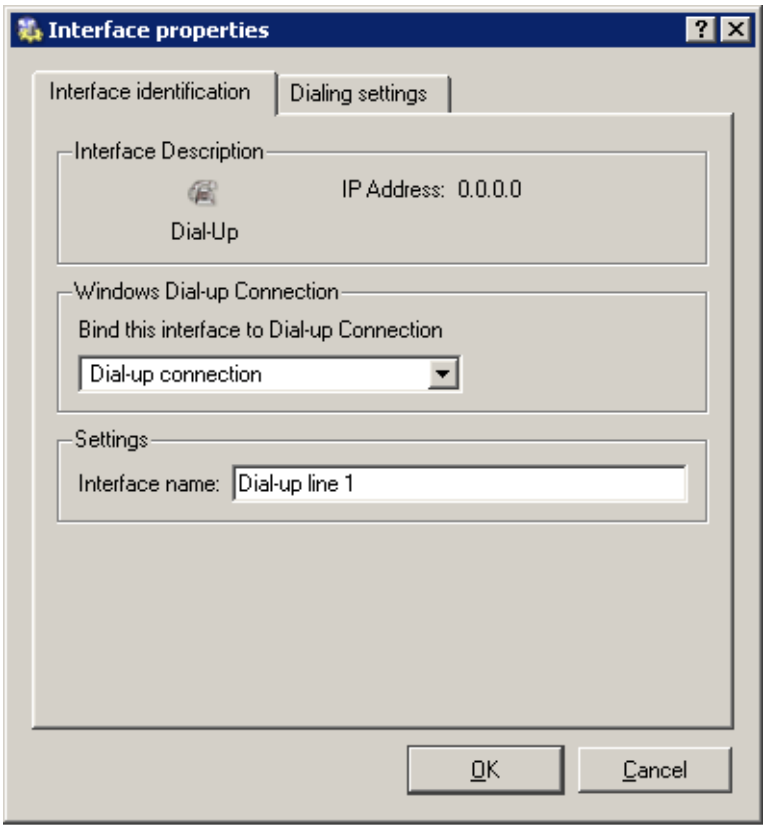
For detailed information on the proprietary VPN solution in *WinRoute* refer to chapter 12.

Adding Interfaces

Click on the *Add* button to add a new interface, either a dial-up or a VPN tunnel (i.e. *server-to-server* VPN connection).



The following text describes only new dial-up connections. Description on how to add a VPN tunnel is provided in chapter 12.4.



Bind this interface... Select the Windows RAS connection that you use to connect to your ISP.

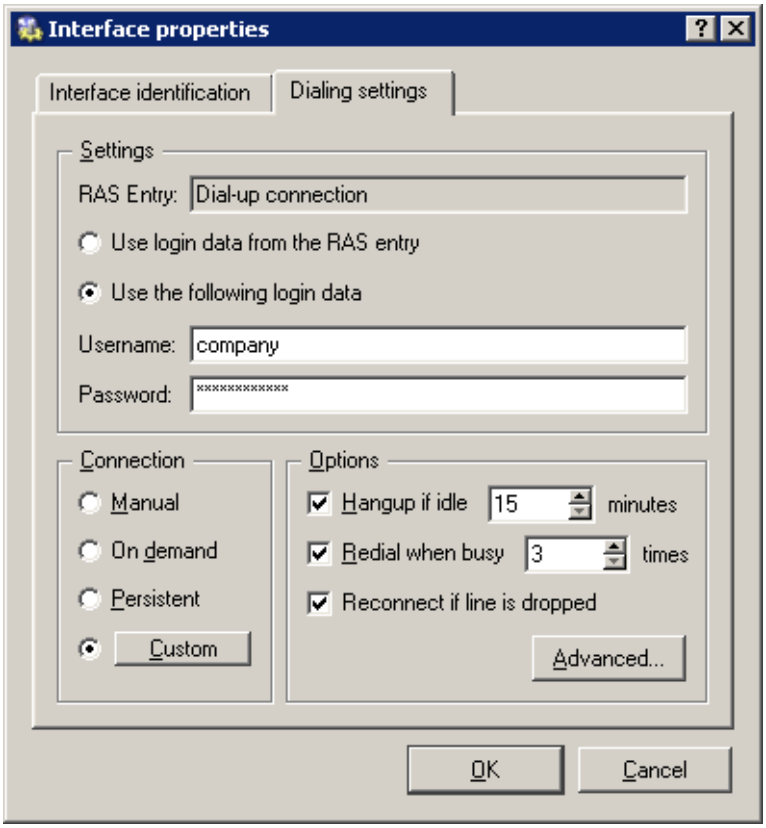
Notes:

1. *WinRoute* searches for connections only in the system "phonebook". When creating a new connection for *WinRoute* it is necessary to set that dial-up connections are available to all users, otherwise the operating system saves a corresponding dial-up connection in the profile of the user who created it and *WinRoute* will not be able to find the connection).

2. We recommend you to test any dial-up connection you create before *WinRoute* is installed.

Interface name Unique name that will identify the line within *WinRoute*.

In the *Dialing Settings* tab you can specify the details of when and how the line will be dialed. Manual dialing is set as default.



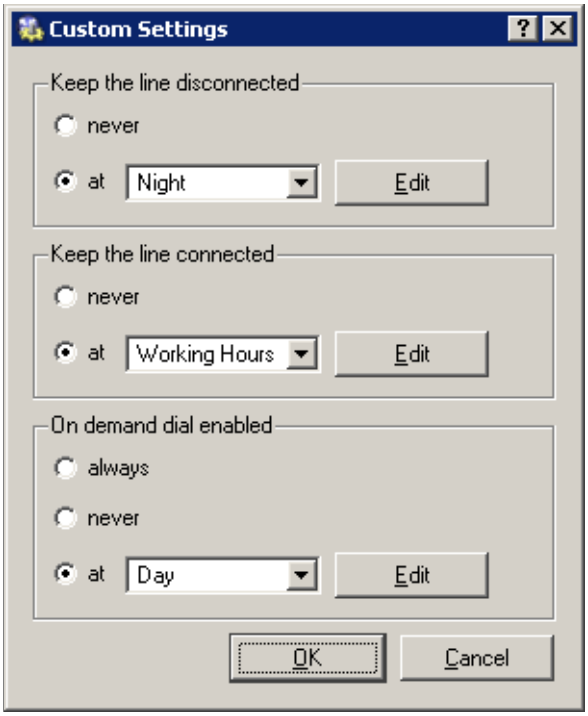
RAS Entry The Windows *Dial-up Connection* entry that has been selected in the *Interface identification* tab. The name RAS item is displayed for informational purposes.

Use login data from the RAS entry Enable this option to use login data saved in a corresponding *RAS Entry* configuration for authentication at the remote server.

Use the following login data Use the *Username* and *Password* entries to enter login data which will be used for authentication at the remote server. This option can be useful for example when for any reason it is not desirable to save the login data in the operating system, when the data is supposed to be edited remotely (via the *Administration Console*) or in case of problem solving.

Connection Connection type that can be used for dialing:

- *Manual* — the line can only be dialed manually, either from the *Kerio Administration Console* or from *WinRoute's* Web interface (see chapter 8).
- *On Demand* — the line will be dialed whenever a host on the LAN tries to access the Internet (incoming packet). To see details about the *WinRoute* and system on-demand dial configuration refer to chapter 11.3.
- *Persistent* — the line will be dialed immediately after the *WinRoute Firewall Engine* service is started and it will be kept active (and will be reconnected if the line is dropped for some reason).
- *Custom* — here you can set with great detail and complexity when the line should be dialed persistently or on demand or not dialed at all.



In sections of the dialog window you can select time ranges for each dialing type. Click on the *Edit* button to open a dialog where time ranges can be created or edited. For more information about time ranges refer to chapter 9.2.

This is how the user defined dialing works:

- The *Keep the line disconnected* option is processed prior to all other options. The line is kept disconnected during this period (or it is hung-up automatically).
- The time range for the *Keep the line connected* option is processed as seconds. During this period the line will be kept connected.
- The *On demand dial enabled* option is processed with the lowest priority. If the *always* option is selected, on-demand dial will be allowed anytime when it is not conflicting with the time range of the *never* option.

Options Advanced parameters for the *Manual*, *On Demand* and *Custom* dial types. In case of persistent connection these options are irrelevant (*WinRoute* keeps the line connected).

Hangup if idle Defines time range during which no data will be allowed to pass through this interface. Outside this period, the line will be disconnected automatically. With each incoming or outgoing packet, the timer of inactivity is set to zero.

There is no such thing as optimum length of the timeout period. If it is too short, the line is dialed too frequently, if too long, the line is kept connected too long. Both increase the Internet connection costs.

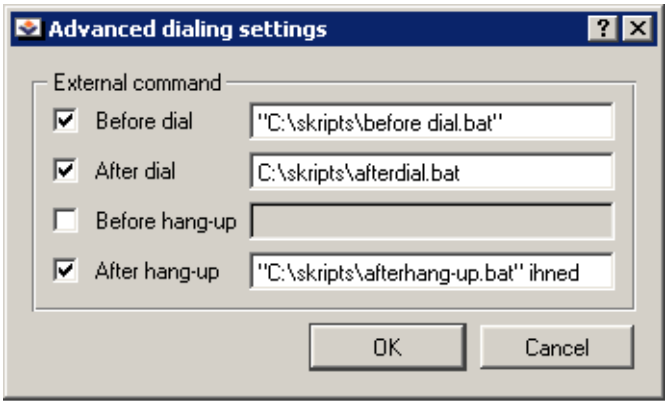
Redial when busy If line is busy when dialed, *WinRoute* will redial unless either connected successfully or the maximal user defined number of attempts is completed. If the connection attempt fails, the demand on dial will be ignored. According to this fact, connection attempts will not be repeated later automatically.

Reconnect if line is dropped If line drop-out is detected, *WinRoute* will try to reconnect automatically.

Advanced dialing settings *WinRoute* allows launching an application or a command in the following situations: *Before dial*, *After dial*, *Before hang-up* or/and *After hang-up*.

Path to the executable file must be complete. If the path includes spaces it must be closed into quotes, otherwise the part after a space will be considered as a parameter(s) of a batch file. If the path to the file is quoted, the text which follows the closing quote mark is also considered as batch file parameter(s).

Warning: If *WinRoute* is running as a service in the operating system, the application will be executed in the background.



Note: In case of the *Before dial* and *Before hang-up* options, the system does not wait for its completion after startup of the program.

Edit Interface parameters

Click *Edit* to modify parameters of a selected interface. The *Interface properties* dialog, identical with the dialog for adding of a new RAS dial-up, is opened in case of RAS dial-ups. Only the *Interface name* entry can be edited in case of network adapters.

For *VPN server* and VPN tunnels, a dialog for setting of the *VPN server* (see chapter 12.1) or a VPN tunnel (for details, refer to chapter 12.4) will be opened.

4.2 Connection Failover

WinRoute allows for definition of connection failover (secondary connection). This alternate connection is enabled automatically whenever a dropout of the primary Internet connection is detected. Functionality of the primary connection is tested by sending of *ICMP Echo Requests* (*PING*) to selected computers. When *WinRoute* finds out that the primary connection is recovered again, the alternate connection is disabled and the primary one is established automatically.

Any network interface or dial connection defined in *WinRoute* can be used as an alternate connection (see chapter 4.1). Traffic rules permitting or denying relevant communication through the alternate connection must be defined. This means that a network connected to the alternate interface must be added to the *Destination* section of all rules defining traffic going out to the Internet through the primary connection.

For details on traffic rules refer to chapter 5.2.

Example: Primary connection used for traffic going out to the Internet is performed by a network adapter (labeled as *Internet* in *WinRoute*). A *Dial-up Connection* interface will

be used for the alternate connection. We want to deny the *Telnet* service in direction from the local network to the Internet.

To meet these requirements, the following rules are set. Two destination items are specified for each rule: network connected to the *Internet* interface (primary connection) and network connected to the *Dial-up Connection* interface (alternate connection).

- *Forbid Telnet* — connection to *Telnet* in direction from the local network to the Internet will be forbidden.
- *NAT* — translation of source IP addresses will be performed for connections from the local network to the Internet (shared Internet connection).
- *Firewall → Internet* — the *WinRoute* host will be allowed to connect to the Internet (NAT is not necessary since this host has its proper IP address).

Name	Source	Destination	Service	Action	Translation
Forbid Telnet	LAN	Internet Dial-up connection	Telnet	✗	
NAT	LAN	Internet Dial-up connection	Any	✓	NAT (Default outgoing interface)
Firewall -> Internet	Firewall	Internet Dial-up connection	Any	✓	
IPSec clients -> serv	IPSec client	IPSec servers	IPSec IKE	✓	NAT (Default outgoing interface)

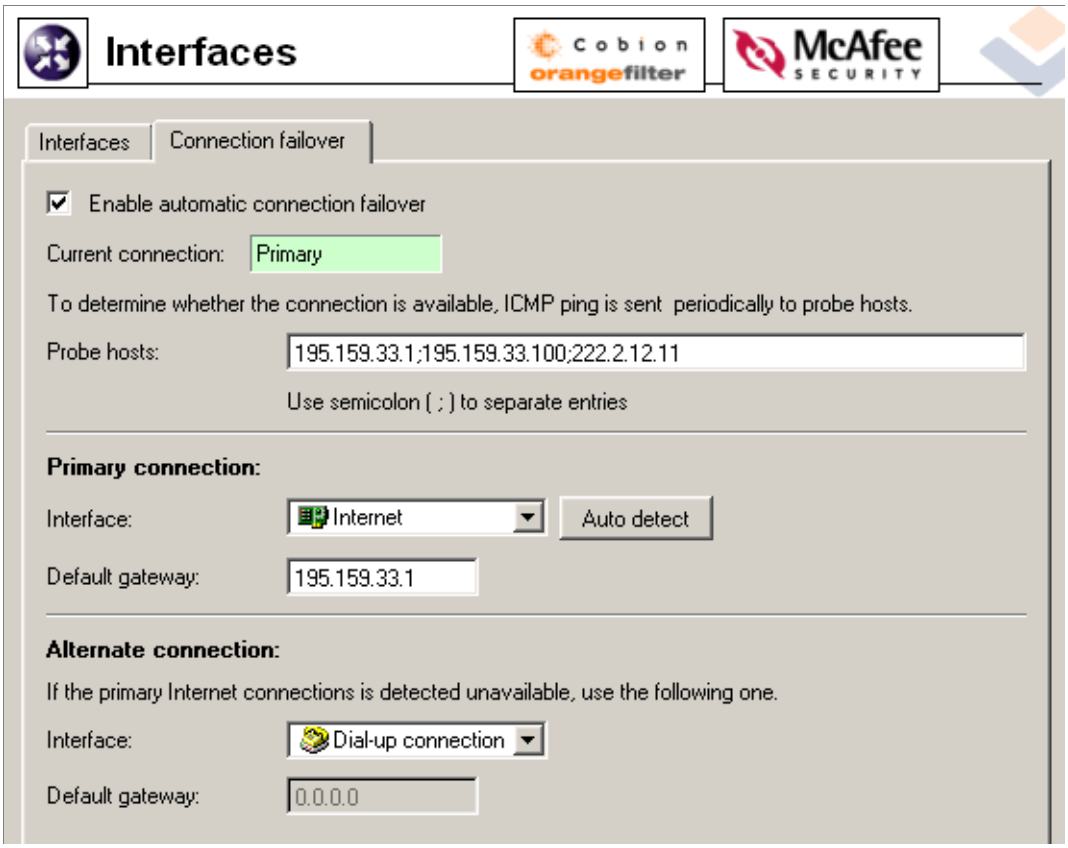
Notes:

1. Traffic rules must be defined by the moment when *Connection Failover Setup* (see below) is enabled, otherwise the connection will not function properly.
2. Use the *Default outgoing interface* option in the *NAT* rule to ensure that the source IP address in packets going from the local network to the Internet is always resolved to the appropriate IP address (i.e. to the IP address of either the primary or alternate interface — accordingly to which one is used at that moment).

   To specify an IP address for NAT, two independent rules must be defined — one for the primary and the other for an alternate connection.

Connection Failover Setup

Use the *Connection failover* tab in *Configuration / Interfaces* to define a secondary connection.



Enable automatic connection failover Use this option to enable/disable connection failover.

Current connection This item informs users on which connection is currently active:

- *Primary* — primary connection (in a green field)
- *Secondary* — alternate (secondary) connection (in a purple field)

Note: Current connections can be switched any time. To view the current status click on the *Refresh* button (at the bottom of the *Connection failover* tab).

Probe hosts Use this entry to specify IP address(es) of at least one computer (or a router, etc.). *WinRoute* will test availability of specified IP address(es) in regular intervals. If at least one of the tested devices is available, the primary connection is considered as functioning.

Note:

1. Connection failover is enabled only if at least one probe host is specified (*WinRoute* is not able to detect fails of the primary connection unless at least one probe host is defined).

2. Probe hosts must be represented by computers or network devices which are permanently running (servers, routers, etc.). Workstations which are running only a few hours per day are irrelevant as probe hosts.

3. Probe hosts must not block *ICMP Echo Requests* (*PING*) since such requests are used to test availability of these hosts — otherwise the hosts will be always considered as unavailable.

Primary connection Parameters of the primary Internet connection. The connection can be defined as follows:

- network interface with a default gateway
- dial-up connection

Only interfaces and dial-up connections defined through the *Interfaces* tab are available in the *Interface* entry (see chapter 4.1).

Default settings (default gateway and a corresponding interface) are detected in the operating system after *WinRoute* installation, or when the *Enable automatic connection failover* option is enabled the first time. This can be also be achieved by clicking on the *Detect* button.

If no default gateway is defined in the operating system (i.e. when the primary connection is performed by a dial-up which is currently hung-up), a connection cannot be detected automatically — the primary connection must be set by hand.

Alternate connection Use this section to set parameters for an alternate internet connection which will be established in case that a primary connection dropout is detected. The alternate connection can be defined as a network interface with a default gateway or as a dial-up connection (like for the primary connection).

Note: The same adapter as for the primary connection can be used, however, the default gateway must be different. This way we can be sure that a different router in the same network (subnet) will be used when the primary connection is dropped out.

Dial-up Use

The following issues must be taken into consideration if a dial-up is used for the primary and/or the alternate connection:

1. Connection failover is relevant only if performed by a permanent connection (using a network adapter or a permanently connected dial-up). If an on-demand dial-up (or a dial-up connection dialed by hand) was used for the primary connection, the alternate connection would be established automatically after each hang-up of the primary connection.

2. If a dial-up is used for alternate connection, it is not important whether this line is dialed on demand — *WinRoute* will dial and hang up the line whenever needed.

   However, problems can be caused by the *Hang-up if idle* option — whenever the alternate line is disconnected automatically, *WinRoute* will not dial it again (unless the primary connection is recovered and then fails again).

For these reasons we recommend you to set dial-up parameters as follows:

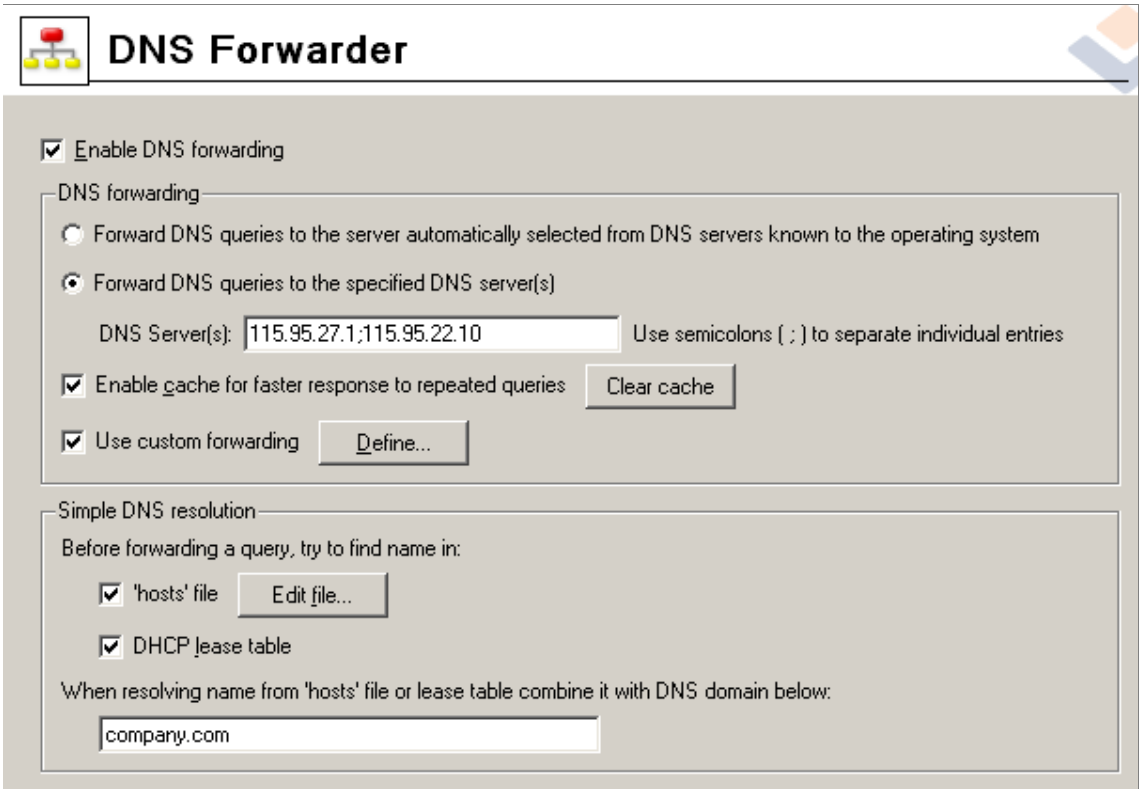
- for the primary connection — *persistent connection*
- for the alternate connection — *manual dialing*

4.3 DNS Forwarder

In *WinRoute*, the *DNS Forwarder* plug-in can be used to enable easier configuration for DNS hosts within local networks or to speed up responses to repeated DNS queries. At local hosts, DNS can be defined by taking the following actions:

- use IP address of the primary or the back-up DNS server. This solution has the risk of slow DNS responses.
- use the DNS server within the local network (if available). The DNS server must be allowed to access the Internet in order to be able to respond even to queries sent from outside of the local domain.
- use *DNS Forwarder* in *WinRoute*. *DNS Forwarder* can be also used as a basic DNS server for the local domain (see below) or as a forwarder for the existing server.

In *WinRoute* default settings the *DNS Forwarder* is switched on and set up so that all DNS queries are forwarded by one of the DNS servers defined in the operating system (usually it is a DNS server provided by your ISP). The configuration can be fine-tuned in *Configurations / DNS Forwarder*.



Enable DNS forwarding This option switches between the on/off modes of the *DNS Forwarder* (the service is running on the port 53 and UDP protocol is used by this service). If *DNS Forwarder* is not used for your network configuration, it can be switched off. If you want to run another DNS server on the same host, *DNS Forwarder must* be switched off, or there will be a collision on the port.

DNS forwarding *DNS Forwarder* must know at least one DNS server to forward queries to. This option defines how *DNS Forwarder* will identify the IP address of the server:

- *Forward DNS queries to the server automatically...* — functional Internet connection is required. At least one DNS server must be defined within TCP/IP configuration (in Windows, DNS servers are defined at a particular adapter, however, these settings will be used within the entire operating system).

  DNS Forwarder can read these settings and use the same DNS servers. This provides the following benefit — the hosts within the local network and the *WinRoute* host will use the same DNS server.

- *Forward DNS queries to the specified DNS server(s)* — DNS queries will be forwarded to the specified DNS server/servers (if more than one server specified, they are considered primary, secondary, etc.). This option should be used when there is the

need to monitor where DNS queries are forwarded to or to create a more complex configuration.

Enable cache for faster response of repeated queries If this option is on, all responses will be stored in local *DNS Forwarder* cache. Responses to repeated queries will be much faster (the same query sent by various clients is also considered as a repeated query).

Physically, the DNS cache is kept in RAM. However, all DNS records are also saved in the `DnsCache.cfg` file (see chapter 17.1). This means that records in DNS cache are kept even after *WinRoute Firewall Engine* is stopped or *WinRoute* is disconnected.

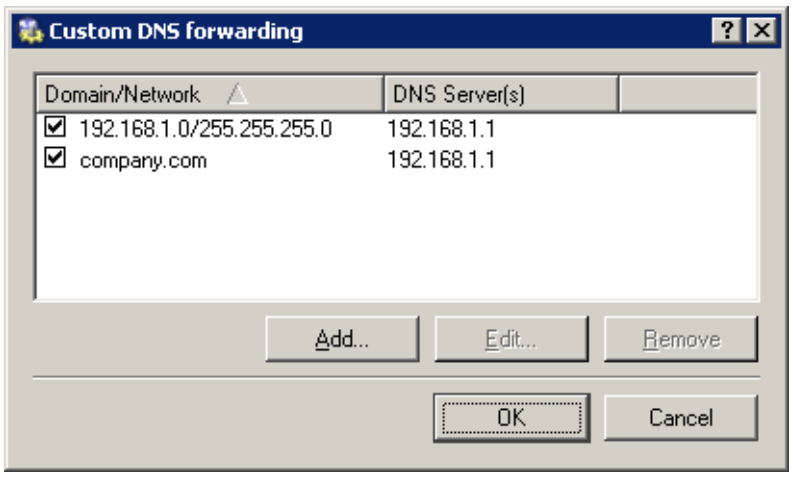
Notes:

1. Time period for keeping DNS logs in the cache is specified individually in each log (usually 24 hours).

2. Use of DNS also speeds up activity of the built-in proxy server (see chapter 4.5).

Use custom forwarding Use this option to define custom settings for forwarding certain DNS queries to other DNS servers. This can be helpful for example when we intend to use a local DNS server for the local domain (the other DNS queries will be forwarded to the Internet directly — this will speed up the response).

Use the *Define* button to open the dialog for definition of custom rules.

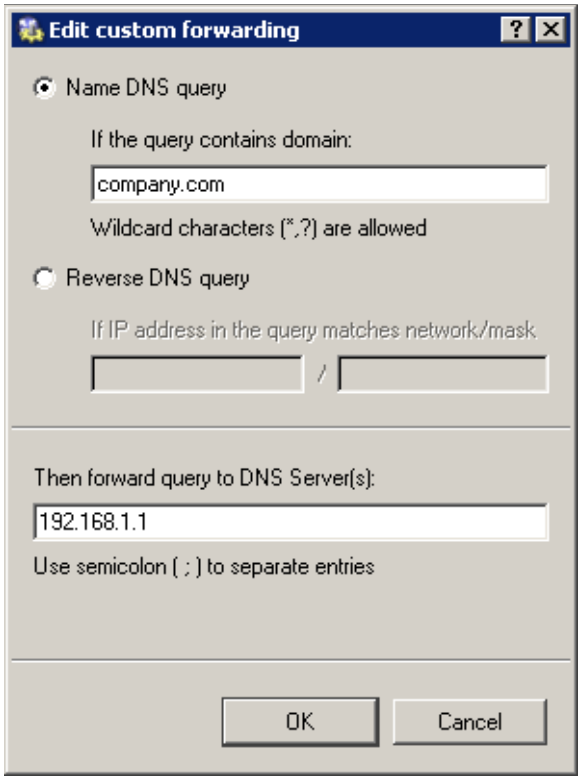


DNS server can be specified for:

- a domain — queries requiring names of computers included in the particular domain will be forwarded to this DNS server (so called `A` queries)

- a subnet — queries requiring IP addresses of the particular domain will be forwarded to the DNS server (reverse domain — `PTR` queries)

Click on the *Add* or the *Edit* button to open a dialog where custom DNS forwarding rules can be defined.



- Use the *Name DNS query* alternative to specify rule for DNS queries on names of computers included in the particular domain (or multiple domains). Use the *If the query contains domain* entry to specify name of the particular domain.

  Specification of a domain name may contain * (asterisk — substitutes any number of characters) and/or ? (question mark — substitutes a single character). The rule will be applied to all domains matching with the string.

  Example: Domain name will be represented by the string `?erio.c*`. The rule will be applied for example to domains `kerio.com`, `cerio.cz`, `aerio.c`, etc.

- Use the *Reverse DNS query* alternative to specify rule for DNS queries on IP addresses in a particular subnet. Subnet is specified by a network address and a corresponding mask (i.e. `192.168.1.0 / 255.255.255.0`).

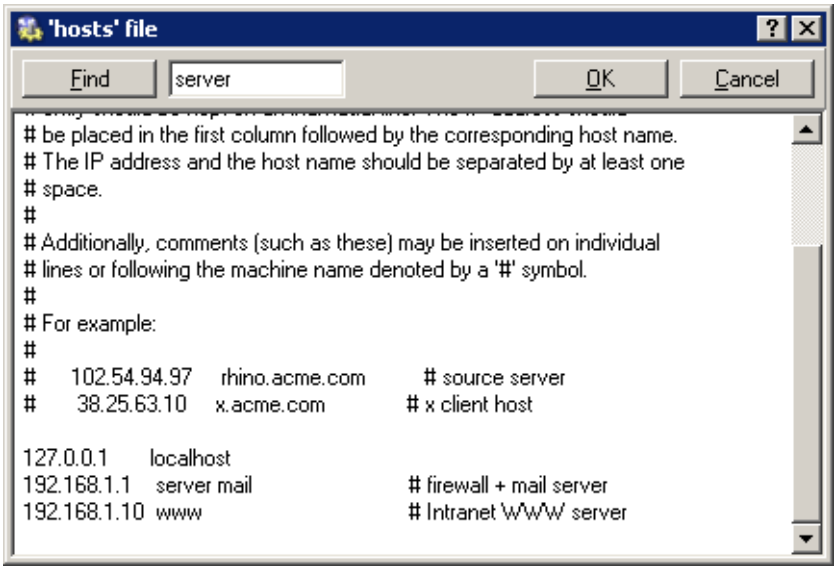
- Use the *Then forward query to DNS Server(s)* field to specify IP address(es) of one or more DNS server(s) to which queries will be forwarded. Use semicolons to separate individual entries.

If multiple DNS servers are entered, they are considered as primary, secondary, etc. If no server is specified, then DNS queries meeting the rule will not be forwarded to any DNS server — *WinRoute* will only scan the local `hosts` file or tables of DHCP server (see below).

Simple DNS Resolution *DNS Forwarder* can be used as a simple DNS server for one of your local domains as well. This can be performed due to the following functions:

- *'host' file* — this file can be found in any operating system supporting TCP/IP. Each row of this file includes host IP addresses and a list of appropriate DNS names. When any DNS query is received, this file will be checked first to find out whether the desired name or IP address is included. If not, the query is forwarded to a DNS server.

  If this function is on, *DNS Forwarder* follows the same rule. Use the *Edit* button to open a special editor where the `HOSTS` file can be edited via *Kerio Administration Console* even if this console is connected to *WinRoute* remotely.



- *DHCP lease table*— if the hosts within local network are configured by the DHCP server in *WinRoute* (see chapter 4.4), the DHCP server knows what IP address was defined for each host. After starting the system, the host sends a request for IP address definition including the name of the host.

  DNS Forwarder can access DHCP lease tables and find out which IP address has been assigned to the host name. If asked to inform about the local name of the host, *DNS Forwarder* will always respond with the current IP address.

... combine the name ... with DNS domain Insert the name of the local DNS domain in this text field.

If a host sends a query to obtain an IP address, it uses the name only (it has not found out the domain yet). *DNS Forwarder* needs to know the name of the local domain to answer queries on fully qualified local DNS names (names including the domain).

The problem can be better understood through the following example:

The local domain's name is `company.com`. The host called `john` is configured so as to obtain an IP address from the DHCP server. After the operating system is started the host sends to the DHCP server a query with the information about its name (`john`). The DHCP server will respond with the IP address `192.168.1.56` and it will keep information about assigning the IP address from the table to the `john` host.

Another host that wants to start communication with the host will send a query on the `john.company.com` name (the `john` host in the `company.com` domain). If the local domain name would not have been known by *DNS Forwarder*, the forwarder would send the query to the DNS server as it would not recognize that it is a name from the local domain. However, as *DNS Forwarder* knows the local domain name, the `company.com` name will be separated and the `john` host with the appropriate IP address will be easily looked up in the DHCP table.

Note: If the local domain is specified in *DNS Forwarder*, local names with or without the domain can be recorded in the `HOSTS` file.

4.4 DHCP server

The DHCP protocol (*Dynamic Host Configuration Protocol*) is used for easy TCP/IP configuration of hosts within the network. The DHCP server selects appropriate configuration parameters (IP address with appropriate subnet mask and other optional parameters, such as IP address of the default gateway, addresses of DNS servers, domain name, etc.) for the client stations.

The DHCP server assigns clients IP addresses within a predefined scope for a certain period (*lease time*). If an IP address is to be kept, the client must request an extension on the period of time before the lease expires. If the client has not required an extension on the lease time, the IP address is considered free and can be assigned to another client.

So called reservations can be also defined on the DHCP server — certain clients will have their own IP addresses reserved. Addresses can be reserved for a hardware address (MAC) or a host name. These clients will have fixed IP address. These addresses are configured automatically.

Using DHCP brings two main benefits. First, the administration is much easier than with the other protocols as all settings may be done at the server (it is not necessary to

configure individual workstations). Second, many network conflicts are eliminated (i.e. one IP address cannot be assigned to more than one workstation, etc.).

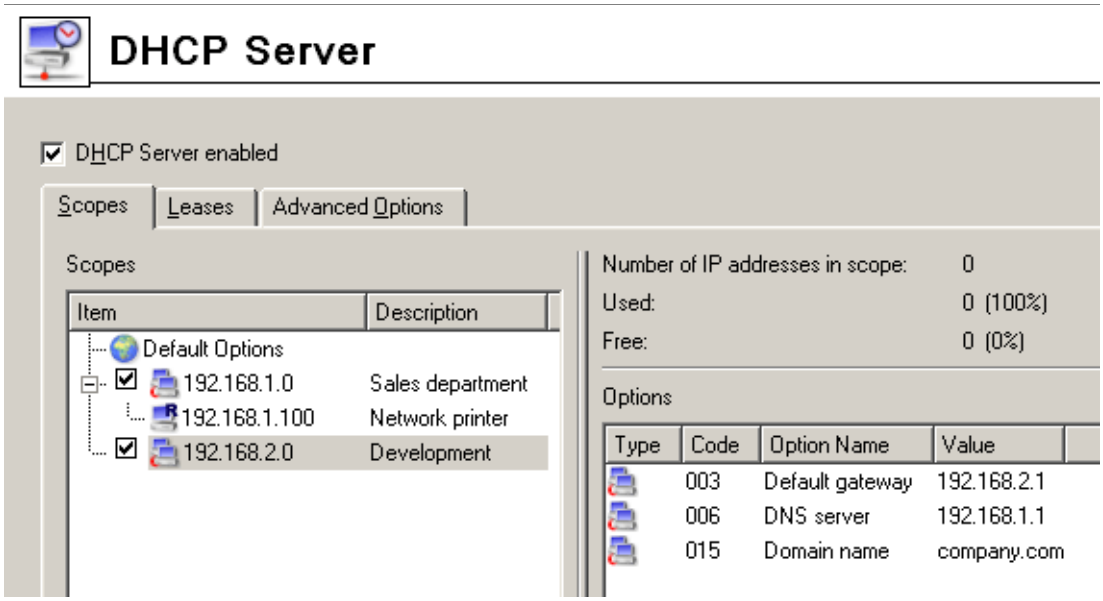
DHCP Server Configuration

To configure the DHCP server in *WinRoute* go to *Configuration / DHCP Server*. Here you can define IP scopes, reservations or optional parameters, and view information about occupied IP addresses or statistics of the DHCP server.

The DHCP server can be enabled/disabled using the *DHCP Server enabled* option (at the top). Configuration can be modified even when the DHCP server is disabled.

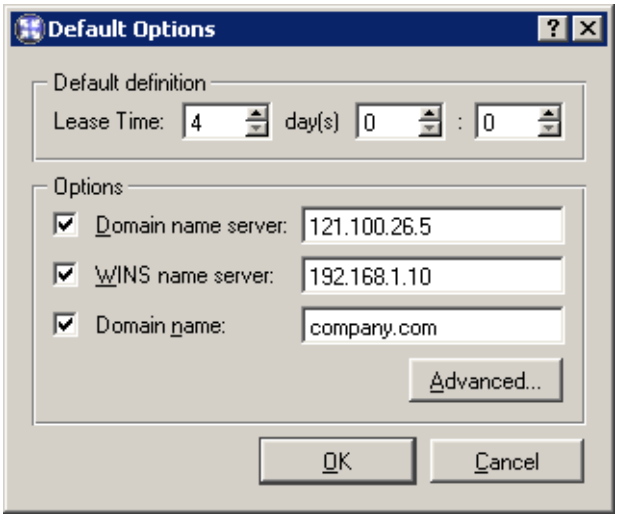
Definition of Scopes and Reservations

To define scopes including optional parameters and to reserve IP addresses for selected clients go to the *Scopes* dialog. The tab includes two parts — in one address scopes and in the other reservations are defined:



In the *Item* column, you can find subnets where scopes of IP addresses are defined. The IP subnet can be either ticked to activate the scope or unticked to make the scope inactive (scopes can be temporarily switched off without deleting and adding again). Each subnet includes also a list of reservations of IP addresses that are defined in it.

In the *Default options* item (the first item in the table) you can set default parameters for DHCP server.



Lease time Time for which an IP address is assigned to clients. This IP address will be automatically considered free by expiration of this time (it can be assigned to another client) unless the client requests lease time extension or the address release.

DNS server Any DNS server (or multiple DNS servers separated by semicolons) can be defined. We recommend you to use *DNS Forwarder* in *WinRoute* as the primary server (first in the list) — IP address of the *WinRoute* host. *DNS Forwarder* can cooperate with DHCP server (see chapter 4.3) so that it will always use correct IP addresses to response to requests on local host names.

WINS server IP address of the WINS server.

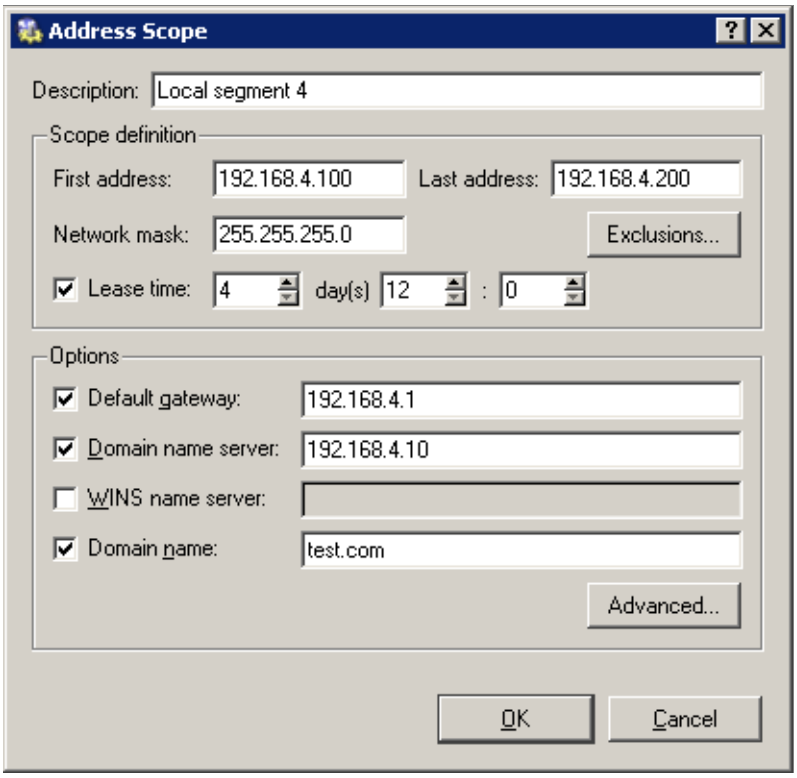
Domain Local Internet domain. Do not specify this parameter if there is no local domain.

Advanced Click on the *Advanced* button to open the dialog which includes list of all optional parameters supported by DHCP protocol (including the ones described above).You can add any parameter supported by DHCP protocol and set its value.

Default parameters are automatically matched with address scopes unless configuration of a particular scope is defined (the *Address Scope/Options* dialog). The same rule is applied on scopes and reservations (parameters defined for a certain address scope are used for the other reservations unless parameters are defined for a specific reservation). Weight of individual parameters corresponds with their position in the tree hierarchy.

Select the *Add / Scope* option to view the dialog for address scope definition.

Note: Only one scope can be defined for each subnet.



Description Comment on the new address scope (just as information for *WinRoute* administrator).

First address, Last address First and last address of the new scope.

Note: If possible, we recommend you to define the scope larger than it would be defined for the real number of users within the subnet.

Mask Mask of the appropriate subnet. It is assigned to clients together with the IP address.

Note: The *Kerio Administration Console* application monitors whether first and last address belong to the subnet defined by the mask. If this requirement is not met, an error will be reported after the confirmation with the *OK* button.

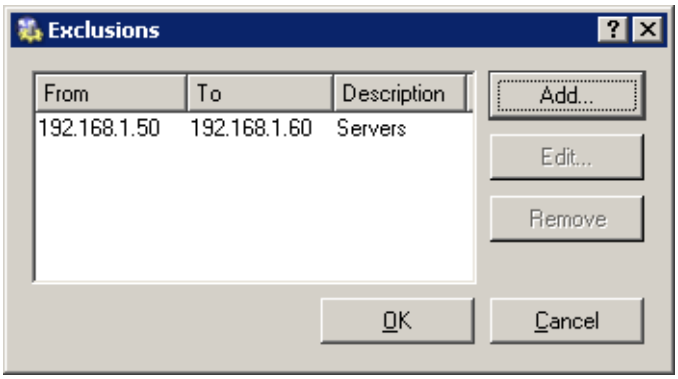
Lease time Time period during which the client can use the IP address. Unless the client has requested extension of the lease time during this period, the IP address is considered free and can be assigned to another client.

Exclusions *WinRoute* enables the administrator to define only one scope in within each subnet. To create more individual scopes, follow these instructions:

- create address scope covering all desired scopes
- define so called exclusions that will not be assigned

Example: In `192.168.1.0` subnet you intend to create two scopes: from `192.168.1.10` to `192.168.1.49` and from `192.168.1.61` to `192.168.1.100`. Addresses from `192.168.1.50` to `192.168.1.60` will be left free and can be used for other purposes.

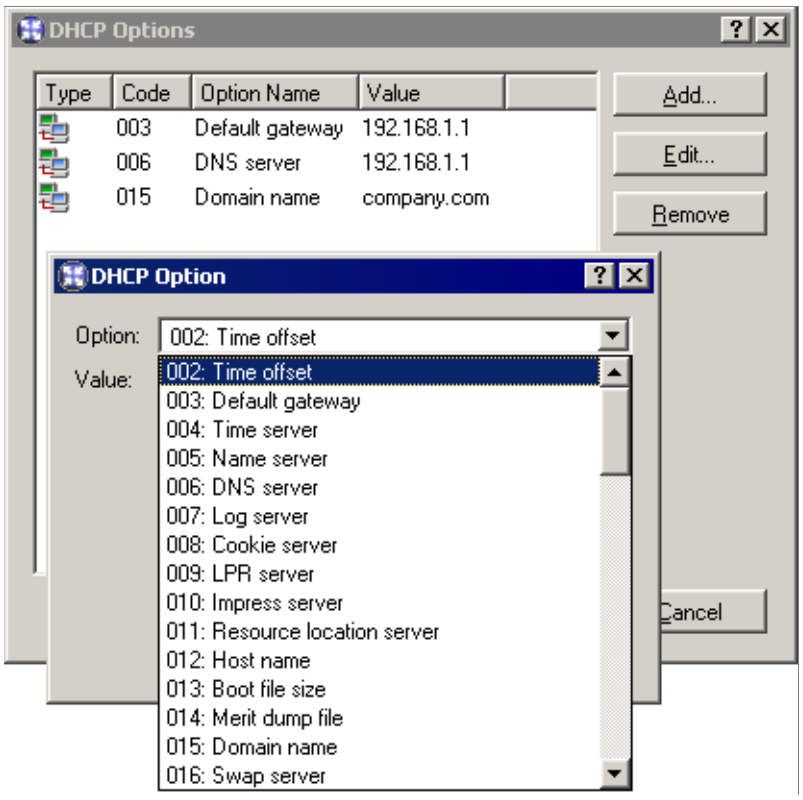
Create the scope from `192.168.1.10` to `192.168.1.100` and click on the *Exclusions* button to define the scope from `192.168.1.50` to `192.168.1.60`. These addresses will not be assigned by the DHCP server.



Parameters In the *Address Scope* dialog, basic DHCP parameters of the addresses assigned to clients can be defined:

- *Default Gateway* — IP address of the router that will be used as the default gateway for the subnet from which IP addresses are assigned.
- *DNS server* — any DNS server (or more DNS servers separated with semicolons). We recommend you to use the *DNS Forwarder* in *WinRoute* as the primary DNS server (IP address of the *WinRoute* host). The reason is that the *DNS Forwarder* can cooperate with the DHCP server (see chapter 4.3) and it will always respond to requests on local host names with the correct IP address.
- *WINS name server* — In networks with multiple routers it is sometimes necessary to use a *WINS* (Windows Internet Naming Service) to resolve local *NetBIOS* computer names.
- *Domain* — local internet domain. Do not define this parameter unless there is a local domain.

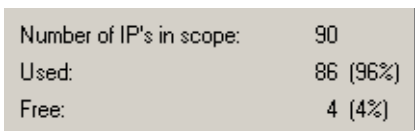
Advanced Click on this button to open a dialog with a complete list of advanced parameters supported by DHCP (including the four mentioned above). Any parameter supported by DHCP can be added and its value can be set within this dialog. This dialog is also a part of the *Address Scopes* tab.



To view configured DHCP parameters and their values within appropriate IP scopes see the right column in the *Address Scope* tab.

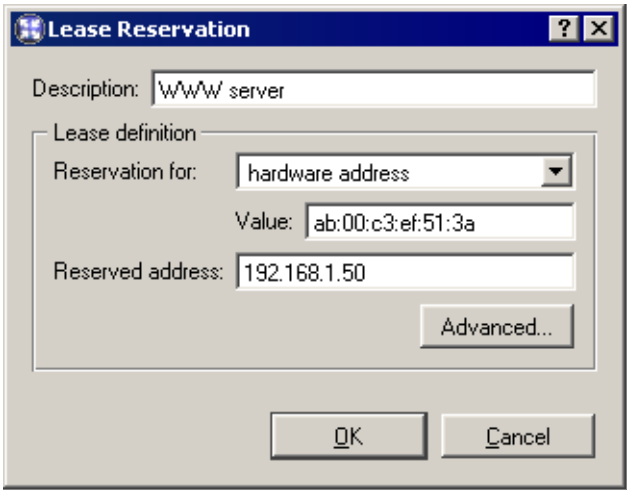
Note: Simple DHCP server statistics are displayed at the right top of the *Address Scope* tab. Each scope is described with the following items:

- total number of addresses within this scope
- number and percentage proportion of leases
- number and percentage proportion of free addresses



Lease Reservations

DHCP server enables the administrator to book an IP address for any host. To make the reservation click on the *Add / Reservations* button in the *Scopes* folder.



Any IP address included in a defined subnet can be reserved.

IP addresses can be reserved for:

- hardware (MAC) address of the host — it is defined by hexadecimal numbers separated by colons, i.e.

  `00:bc:a5:f2:1e:50`

  or by dashes— for example:

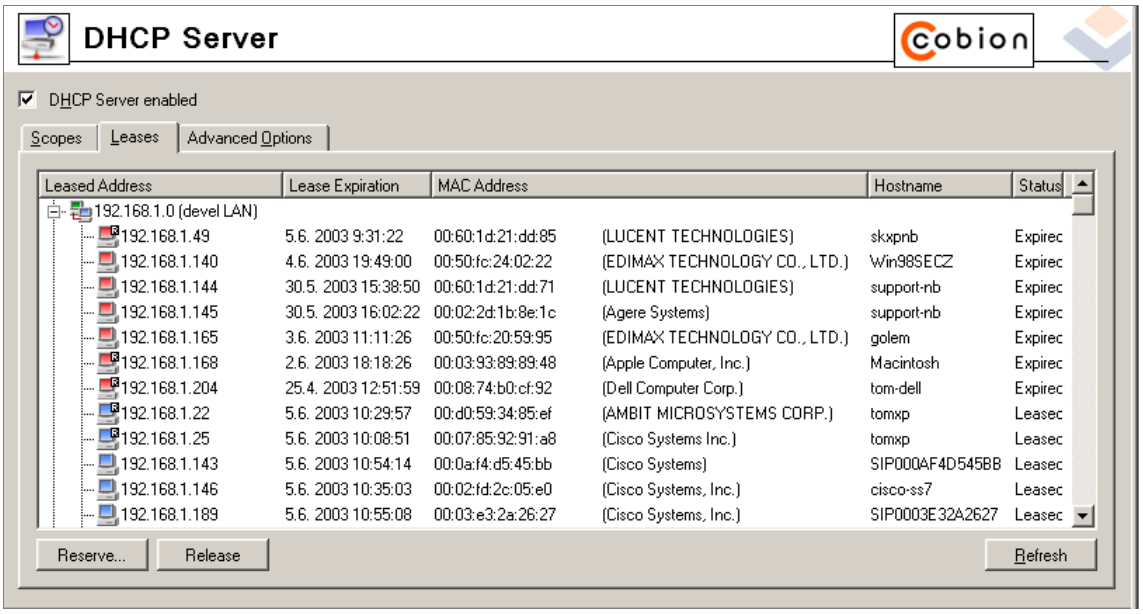
  `00-bc-a5-f2-1e-50`

  The MAC address of a network adapter can be detected with operating system tools (i.e. with the `ipconfig` command) or with a special application provided by the network adapter manufacturer.

- host name — DHCP requests of most DHCP clients include host names (i.e. all Windows operating systems), or the client can be set to send a host name (i.e. Linux operating system).

Leases

IP scopes can be viewed in the *Leases* tab. These scopes are displayed in the form of trees. All current leases within the appropriate subnet are displayed in these trees.



Note: Icon color represents address status (see below). Icons marked with R represent reserved addresses.

Columns in this section contain the following information:

- *Leased Address* — leased IP address
- *Lease Expiration* — date and time specifying expiration of the appropriate lease
- *MAC Address* — hardware address of the host that the IP address is assigned to (including name of the network adapter manufacturer).
- *Hostname* — name of the host that the IP address is assigned to (only if the DHCP client at this host sends it to the DHCP server)
- *Status* — status of the appropriate IP address; *Leased* (leased addresses), *Expired* (addresses with expired lease — the client has not asked for the lease to be extended yet), *Declined* (the lease was declined by the client) or *Released* (the address has been released by the client).

  Notes:

  1. Data about expired and released addresses are kept by the DHCP server and can be used later if the same client demands a lease.
  2. Declined addresses are handled according to the settings in the

     Options tab (see below).

The following columns are hidden by default:

- *Last Request Time* — date and time when the recent request for a lease or lease extension was sent by a client
- *Lease Remaining Time* — time remaining until the appropriate *Lease Expiration*

Use the *Release* button to release a selected IP address immediately (independently of its status). Released addresses are considered free and can be assigned to other clients immediately.

Click on the *Reserve* button to reserve a selected (dynamically assigned) IP address based on the MAC address or name of the host that the address is currently assigned to. The *Scopes* tab with a dialog where the appropriate address can be leased will be opened automatically. All entries except for the *Description* item will be already defined with appropriate data. Define the *Description* entry and click on the *OK* button to assign a persistent lease for the IP address of the host to which it has been assigned dynamically.

Note: The MAC address of the host for which the IP is leased will be inserted to the lease reservation dialog automatically. To reserve an IP address for a hostname, change settings of the *Reservation For* and *Value* items.

Options

Other DHCP server parameters can be set in the *Options* tab.

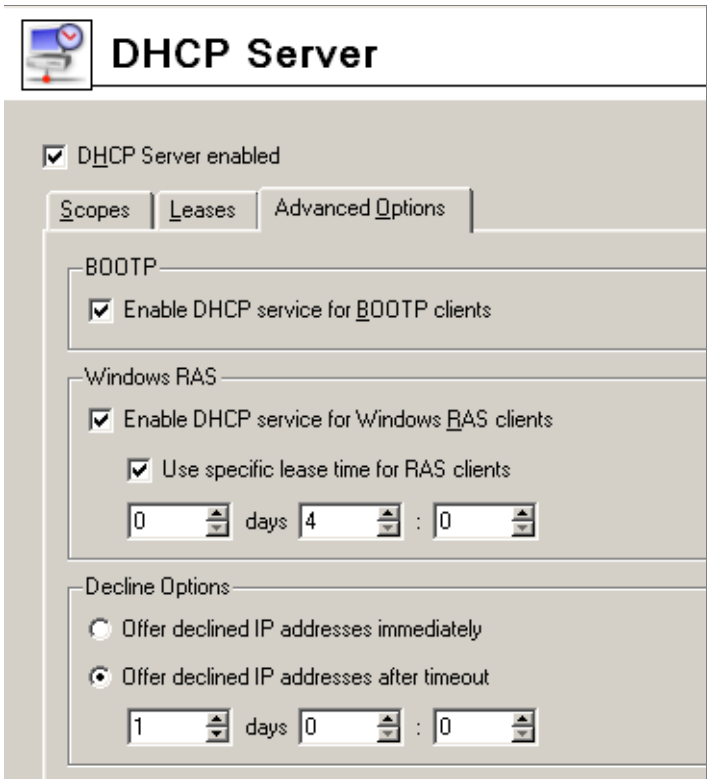
BOOTP If this option is enabled, the DHCP server will assign IP addresses (including optional parameters) also to clients of BOOTP protocol (protocol used formerly to DHCP— it assigns configurations statically only, according to MAC addresses).

Windows RAS Through this option you can enable DHCP service for RAS clients (Remote Access Service). You can also specify time when the service will be available to RAS clients (an IP address will be assigned) if the default value is not convenient.

Declined options These options define how declined IP addresses (*DHCPDECLINE* report) will be handled. These addresses can be either considered released and assigned to other users if needed (the *Offer immediately* option) or blocked during a certain time for former clients to be able to use them (the *Declined addresses can be offered after timeout* option).

4.5 Proxy server

Even though the NAT technology used in *WinRoute* enables direct access to the Internet from all local hosts, it contains a standard HTTP proxy server. Under certain conditions



the direct access cannot be used or it is inconvenient . The following list describes the most common situations:

1. To connect from the *WinRoute* host it is necessary to use the proxy server of your ISP. In this case the Internet cannot be accessed directly.

   Proxy server included in *WinRoute* can forward all queries to so called *parent proxy server*).

2. Internet connection is performed via a dial-up and access to certain Web pages is blocked (refer to chapter 6.1). If a direct connection is used, the line will be dialed before the HTTP query could be detected (line is dialed upon a DNS query or upon a client's request demanding connection to a Web server). If a user connects to a forbidden Web page, *WinRoute* dials the line and blocks access to the page — the line is dialed but the page is not opened.

   Proxy server can receive and process clients' queries locally. The line will not be dialed if access to the requested page is forbidden.

3. *WinRoute* is deployed within a network with many hosts where proxy server has been used. It would be too complex and time-consuming to re-configure all the hosts.

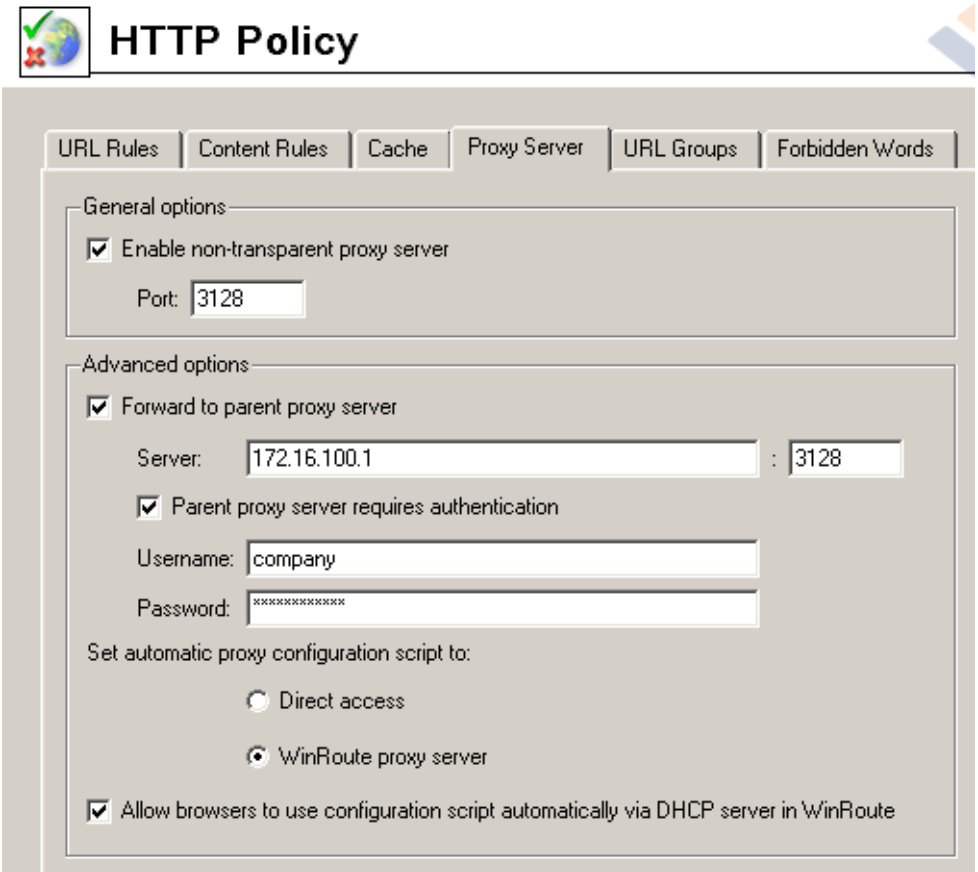
The Internet connection functionary is kept if proxy server is used — it is not necessary to edit configuration of individual hosts (or only some hosts should be reconfigured).

4. *WinRoute* may be filtering sites and objects transparently, however it is possible for a Web page to contain a redirect to a non-standard TCP port. In this case the transparent HTTP proxy will not be used. If the browser is configured to use a proxy server then the redirect will continue to be processed through the proxy server.

Note: The Proxy server in *WinRoute* supports only the HTTP and HTTPS protocols. Use direct access to support traffic using the FTP protocol unless you use a parent proxy server that supports FTP protocol (*WinRoute's* proxy server can "get" FTP to the parent proxy server).

Proxy Server Configuration

To configure proxy server parameters open the *Proxy server* tab in *Configuration / Content Filtering / HTTP Policy*.



Enable non-transparent proxy server This option enables the HTTP proxy server in *WinRoute* on the port inserted in the *Port* entry (3128 port is set by the default).

Warning : If you use a port number that is already used by another service or application, *WinRoute* will accept this port, however, the proxy server will not be able to run and the following report will be logged into the *Error* log (refer to chapter 16.8):

```
failed to bind to port 3128:  another application is using this
port
```

If you are not sure that the port you intend to use is free, click on the *Apply* button and check the *Error* log (check whether the report has or has not been logged) immediately.

Forward to parent proxy server Tick this option for *WinRoute* to forward all queries to the parent proxy server which will be specified by the following data:

- *Server* — DNS name or IP address of parent proxy server and the port on which the server is running (3128 port is used by the default).
- *Parent proxy server requires authentication* — if parent proxy server requires authentication through username and password, enable this option and specify the *Username* and *Password* entries.

  Note: The name and password for authentication to the parent proxy server is sent with each HTTP request. Only *Basic* authentication is supported.

  The *Forward to parent proxy server* option specifies how *WinRoute* will connect to the Internet (for update checks, downloads of *McAfee* updates and for connecting to the online *Cobion* databases).

Set automatic proxy configuration script to If a proxy server is used, Web browsers on client hosts must be configured correctly. Most common Web browsers (e.g. *Microsoft Internet Explorer*, *Netscape/Mozilla*, *Opera*, etc.) enable automatic configuration of corresponding parameters by using a script downloaded from a corresponding Website specified with URL.

In the case of *WinRoute's* proxy server, the configuration script is saved at

```
http://192.168.1.1:3128/pac/proxy.pac,
```

where 192.168.1.1 is the IP address of the *WinRoute* host and number 3128 represents the port of the proxy server (see above).

The *Allow browsers to use configuration script automatically...* option adjusts the configuration script in accord with the current *WinRoute* configuration and the settings of the local network:

- *Direct access* — no proxy server will be used by browsers
- *WinRoute proxy server* — IP address of the *WinRoute* host and the port on which the proxy server is running will be used by the browser (see above).

Note: The configuration script requires that the proxy server is always available (even if the *Direct access* option is used).

Allow browsers to use configuration script automatically... The *Microsoft Internet Explorer* browser can be configured automatically if the DHCP server is used and if the *Automatically detect settings* option is enabled in the browser settings.

WinRoute's DHCP server must be running (see chapter 4.4), otherwise the function will not work. TCP/IP parameters at the host can be static — *Microsoft Internet Explorer* sends a special DHCP query when started.

TIP: This way *Microsoft Internet Explorer* browsers at all local hosts can be configured by a single click.

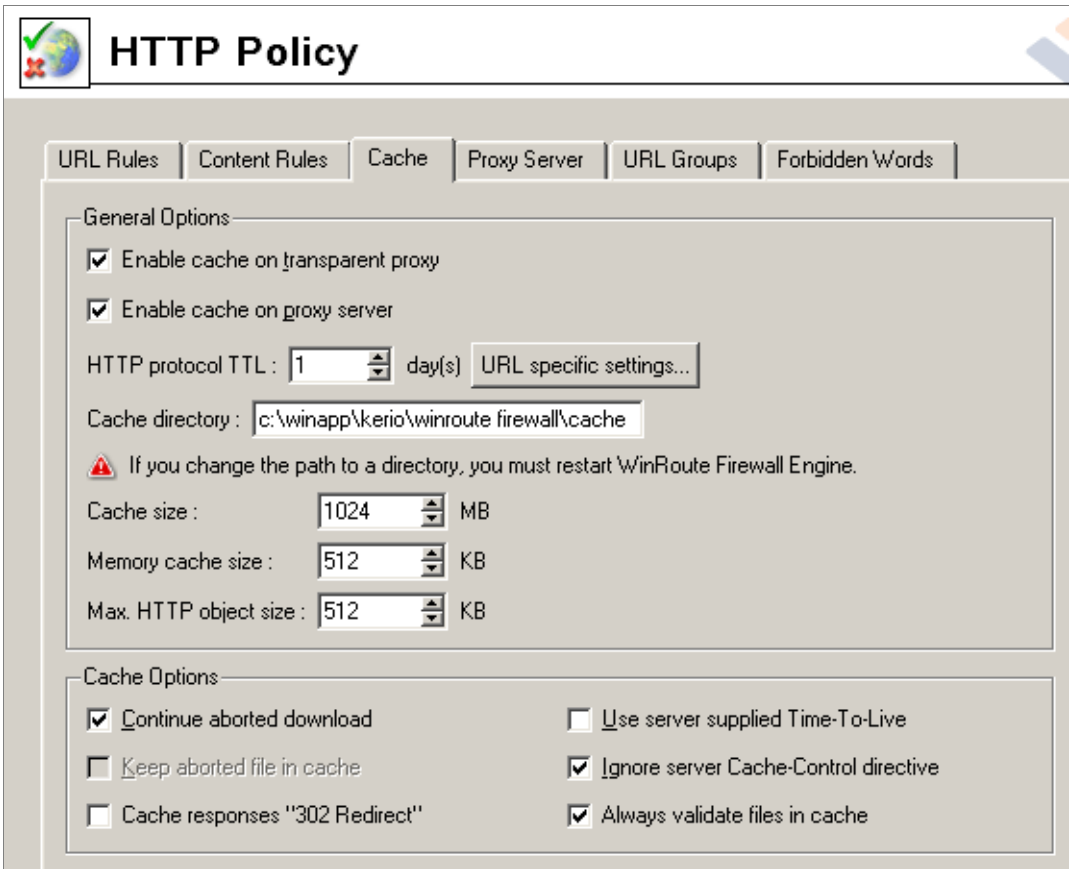
4.6 HTTP cache

Using cache to access Web pages that are opened repeatedly reduces Internet traffic. Downloaded files are saved to the harddisk of the *WinRoute* host so that it is not necessary to download them from the Web server again later.

All objects are stored in cache for a certain time only (*Time To Live — TTL*). This time defines whether checks for the most recent versions of the particular objects will be performed upon a new request of the page. The required object will be found in cache unless the *TTL* timeout has expired. If it has expired, a check for a new update of the object will be performed. This ensures continuous update of objects that are stored in the cache.

The cache can be used either for direct access or for access via the proxy server. If you use direct access, the HTTP protocol inspector must be applied to permitted TCP port 80 and 443 traffic. (refer to chapters 5.2 and 9.3).

To set HTTP cache parameters go to the *Cache* tab in *Configuration / Content Filtering / HTTP Policy*.



Enable cache on transparent proxy This option enables cache for HTTP traffic that uses the HTTP protocol inspector (direct access to the Internet)

Enable cache on proxy server Enables the cache for objects downloaded via *WinRoute's* proxy server (see chapter 4.5)

HTTP protocol TTL Default time of object validity within the cache. This time is used when:

- TTL of a particular object is not defined (to define TTL use the *URL specific settings* button —see below)
- TTL defined by the Web server is not accepted (the *Use server supplied Time-To-Live* entry)

Cache directory Directory that will be used to store downloaded objects. The `cache` file under the directory where *WinRoute* is installed is used by default.

Warning: Changes in this entry will not be accepted unless the *WinRoute Firewall Engine* is restarted.

Cache size Size of the cache file on the disc. Maximal size of this file is determined by file system: *FAT16* — 2GB, 4GB are allowed for other file systems.

Note: If 98 percent of the cache is full, a so called cleaning will be run — this function will remove all objects with expired TTL. If no objects are deleted successfully, no other objects can be stored into the cache unless there is more free space on the disc (made by further cleaning or by manual removal).

Memory cache size Maximal memory cache size in the main storage. This cache is used especially to accelerate records to the cache on the disc.

If the value is too high the host's performance can be affected negatively (cache size should not exceed 10 per cent of the computing memory).

Max HTTP object size maximal size of the object that can be stored in cache.

With respect to statistics, the highest number of requests are for small objects (i.e. HTML pages, images, etc.). Big sized objects, such as archives (that are usually downloaded at once), would require too much memory in the cache.

Cache Options Advanced options where cache behavior can be defined.

- *Continue aborted download* — tick this option to enable automatic download of objects that have been aborted by the user (using the *Stop* button in a browser). Users often abort downloads for slow pages. If any user attempts to open the same page again, the page will be available in the cache and downloads will be much faster.

- *Keep aborted files in cache* — if this option is enabled, the server will save even incomplete objects into the cache (object downloads which have not been finished). Downloads will be faster when the page is opened again.

  The *Keep aborted files in cache* option will be ignored if the *Continue aborted download* option is enabled.

- *Cache redirect responses* — HTTP responses that contain redirections will be cached.

- *Use server supplied Time-To-Live* — objects will be cached for time specified by the Web server from which they are downloaded. If TTL is not specified by the server, the default TTL will be used (see the *HTTP protocol TTL* item).

- *Ignore server Cache-Control directive* — *WinRoute* will ignore directives for cache control of Web pages.

Pages often include a directive that the page will not be saved into the cache. Enable the *Ignore server Cache-Control directive* option to make *WinRoute* accept only *no-store* and *private* directives.

Note: WinRoute examines HTTP header directives of responses, not Web pages.

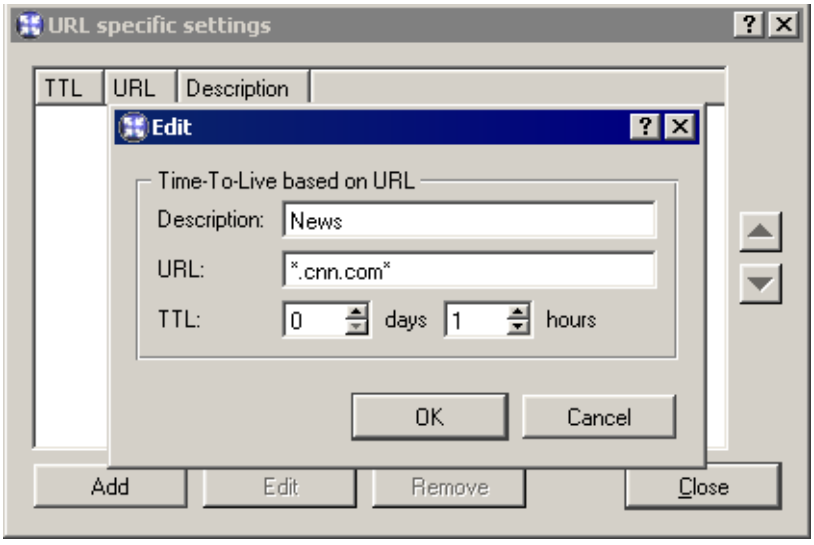
- *Always validate file in cache* — with each query *WinRoute* will check the server for updates of objects stored in the cache (regardless of whether the client demands this).

Note: Clients can always require a check for updates from the Web server (regardless of the cache settings). Use a combination of the *Ctrl-F5* keys to do this using either the *Microsoft Internet Explorer* or the *Netscape/Mozilla* browser. You can set browsers so that they will check for updates automatically whenever a certain page is opened (then you will only refresh the particular page).

URL Specific Settings

The default cache TTL of an object is not necessarily convenient for each page. You may require not to cache an object or shorten its TTL (i.e. for pages that are accessed daily).

Use the *URL specific settings* button to open a dialog where TTL for a particular URL can be defined.



Rules within this dialog are ordered in a list where the rules are read one by one from the top downwards (use the arrow buttons on the right side of the window to reorder the rules).

Description Text comment on the entry (informational purpose only)

URL URL for which cache TTL will be specified. URLs can have the following forms:

- complete URL (i.e. `www.kerio.com/cz/index.html`)
- substring using wildcard matching (i.e. `*news.com*`)
- server name (i.e. `www.kerio.com`) — represents any URL included at the server (the string will be substituted for `www.kerio.com/*` automatically.

TTL TTL of objects matching with the particular URL.

The *0 days, 0 hours* option means that objects will not be cached.

Chapter 5

Traffic Policy

Traffic Policy belongs to of the basic *WinRoute* configuration. All the following settings are displayed and can be edited within the table:

- security (protection of the local network including the *WinRoute* host from Internet intrusions
- IP address translation (or NAT, Network Address Translation — technology which enables transparent access of the entire local network to the Internet with one public IP address only)
- access to the servers (services) running within the local network from the Internet (port mapping)
- controlled access to the Internet for local users

Traffic policy rules can be defined in *Configurations / Traffic Policy*. The rules can be defined either manually (advanced administrators) or using the wizard (recommended).

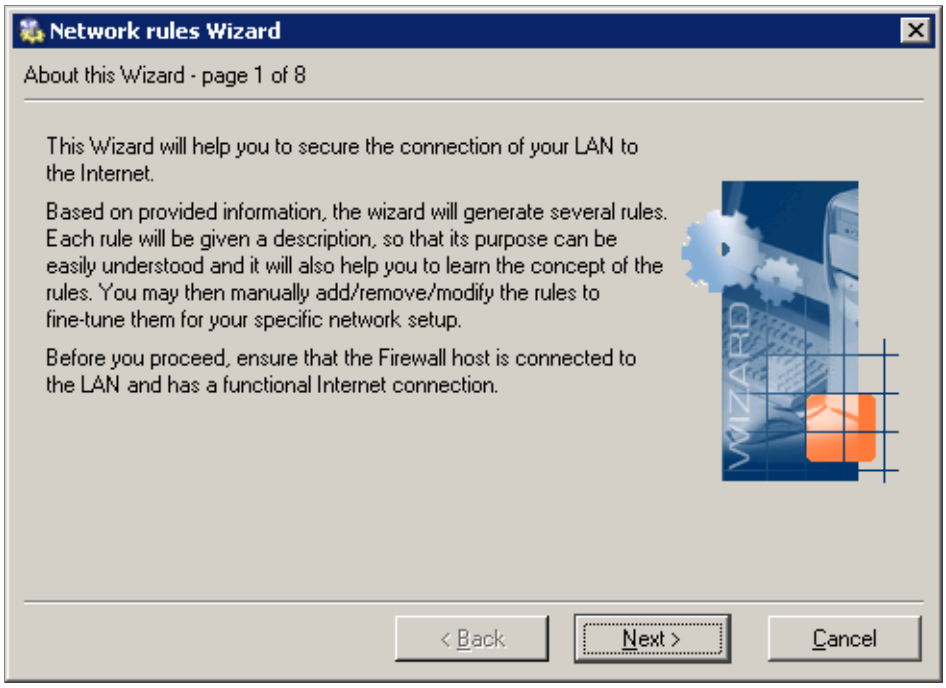
5.1 Network Rules Wizard

The network rules wizard demands only the data that is essential for creating a basic set of traffic rules. The rules defined in this wizard will enable access to selected services to the Internet from the local network, and ensure full protection of the local network (including the *WinRoute* host) from intrusion attempts from the Internet. To guarantee reliable *WinRoute* functionality after the wizard is used, all existing rules are removed and substituted by rules created automatically upon the new data.

Click on the *Wizard* button to run the network rules wizard.

Note: The existing traffic policy is substituted by new rules after completing the entire process after confirmation of the last step. This means that during the process the wizard can be stopped and canceled without losing existing rules.

Step 1 — information

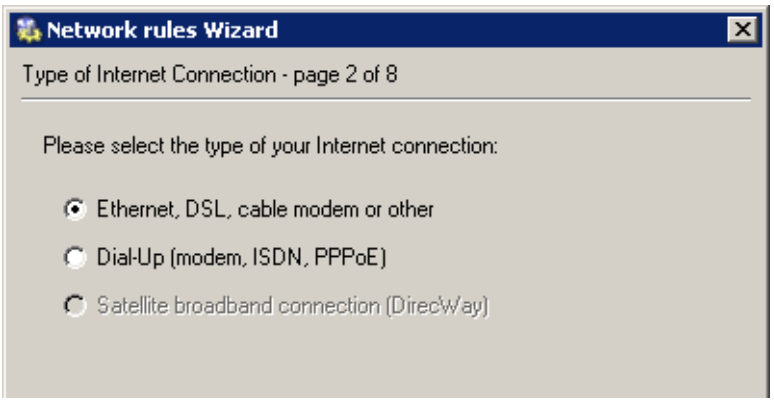


To run successfully, the wizard requires the following parameters on the *WinRoute* host:

- at least one active adapter connected to the local network
- at least either one active adapter connected to the Internet or one dial-up defined. The dial-up needn't be active to run the wizard.

Step 2 — selection of Internet connection type

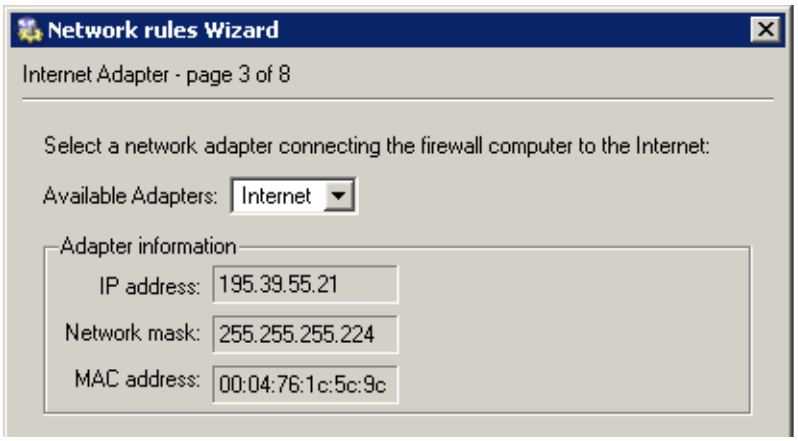
Select the appropriate type of Internet connection that is used — either a network adapter (Ethernet, WaveLAN, DSL, etc.), a dialed line (analog modem, ISDN, etc.) or the *DirecWay* satellite system. *DirecWay* is available only if a corresponding device driver is detected in the operating system.



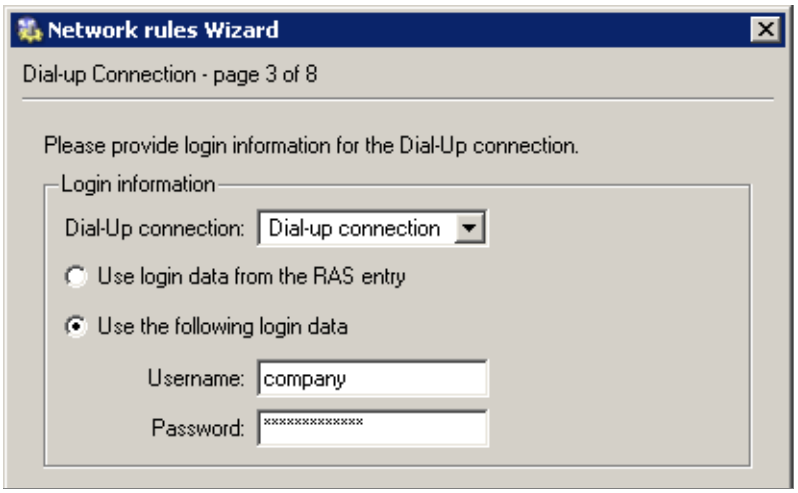
Step 3 — network adapter or dial-up selection

If the network adapter is used to connect the host to the Internet, it can be selected in the menu. To follow the wizard instructions easily, IP address, network mask and MAC address of the selected adapter are displayed as well.

Note: The Web interface with the default gateway is listed first. Therefore, in most cases the appropriate adapter is already set within this step.



In case of a dial line, the appropriate type of connection (defined in the operating system) must be selected and login data must be specified.



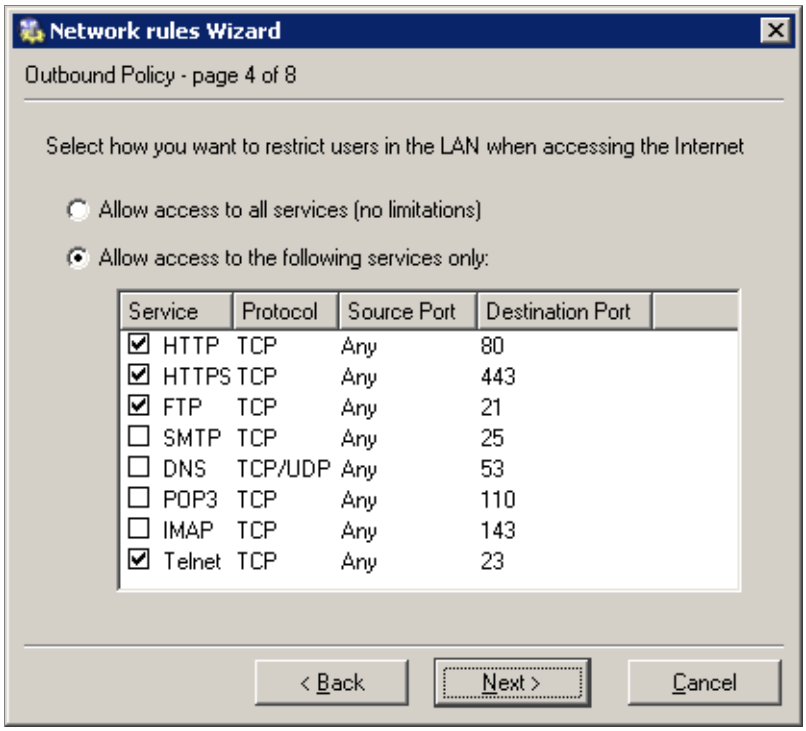
- *Use login data from the RAS entry* — username and password for authentication at the remote server will be copied from a corresponding *Windows* RAS entry. The

RAS connection must be saved in the system "phonebook" (the connection must be available to any user).

- *Use the following login data* — specify *Username* and *Password* that will be used for authentication at the remote server. This option can be helpful for example when it is not desirable to save the login data in the operating system or if later it would be edited.

Step 4 — Internet access limitations

Select which Internet services will be available for LAN users:



Allow access to all services Internet access from the local network will not be limited. Users can access any Internet service.

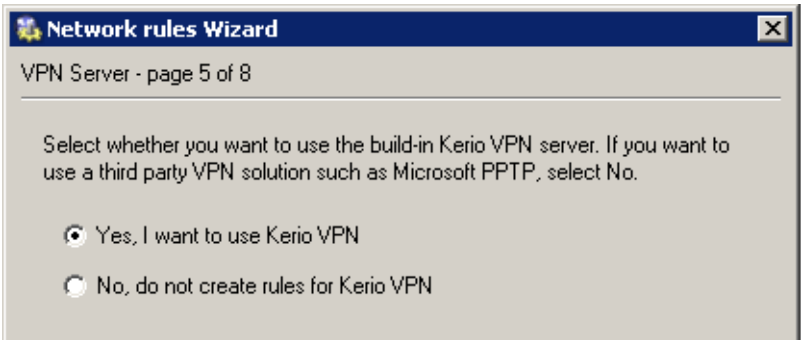
Allow access to the following services only Only selected services will be available from the local network.

Note: In this dialog, only basic services are listed (it does not depend on what services were defined in *WinRoute* — see chapter 9.3). Other services can be allowed by definition of separate traffic policy rules— see below.

Step 5 — enabling Kerio VPN traffic

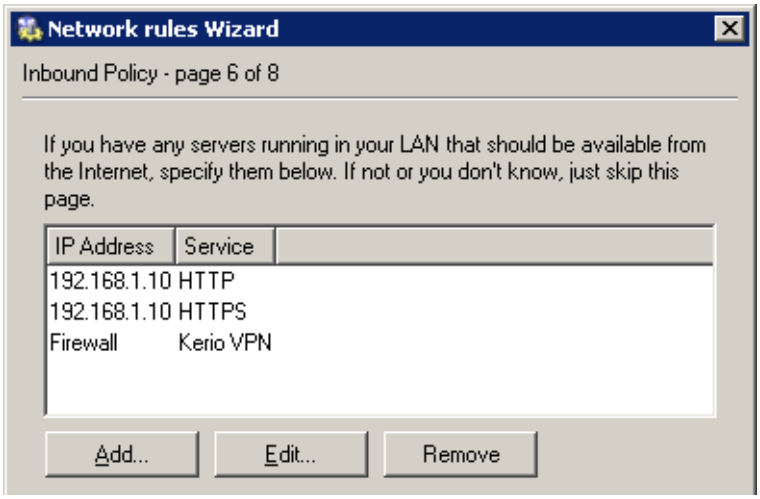
To use *WinRoute's* proprietary VPN solution in order to connect remote clients or to create tunnels between remote networks, select *Yes, I want to use Kerio VPN*. Specific services and address groups for VPN will be added. For detailed information on the proprietary VPN solution integrated in *WinRoute*, refer to chapter 12.

If you intend not to use the solution or to use a third-party solution (e.g. *Microsoft PPTP*, *Nortel IPSec*, etc.), choose the *No, do not create rules for Kerio VPN* option.

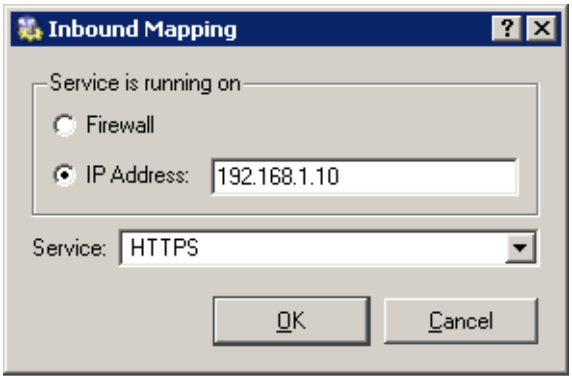


Step 6 — specification of servers that will be available within the local network

If any service (e.g. WWW server, FTP server, etc. which is intended be available from the Internet) is running on the *WinRoute* host or another host within the local network, define it in this dialog.



The dialog window that will open a new service can be activated with the *Add* button.



Service is running on Definition of the host where the service is running:

- *Firewall* — the host where *WinRoute* is installed
- *IP Address* — address of a server within the local network (the host that the service is running on)

  Note: access to the Internet through *WinRoute* must be defined in the default gateway of the host, otherwise the service will not be available.

Service Selection of a service to be enabled. The service must be defined in *Configurations / Definitions / Services* formerly (see chapter 9.3).

Note: Majority of common services is predefined in *WinRoute*.

Step 6 — NAT

If you only use one public IP address to connect your private local network to the Internet, run the *NAT* function (IP address translation). Do not trigger this function if *WinRoute* is used for routing between two public networks or two local segments (neutral router).

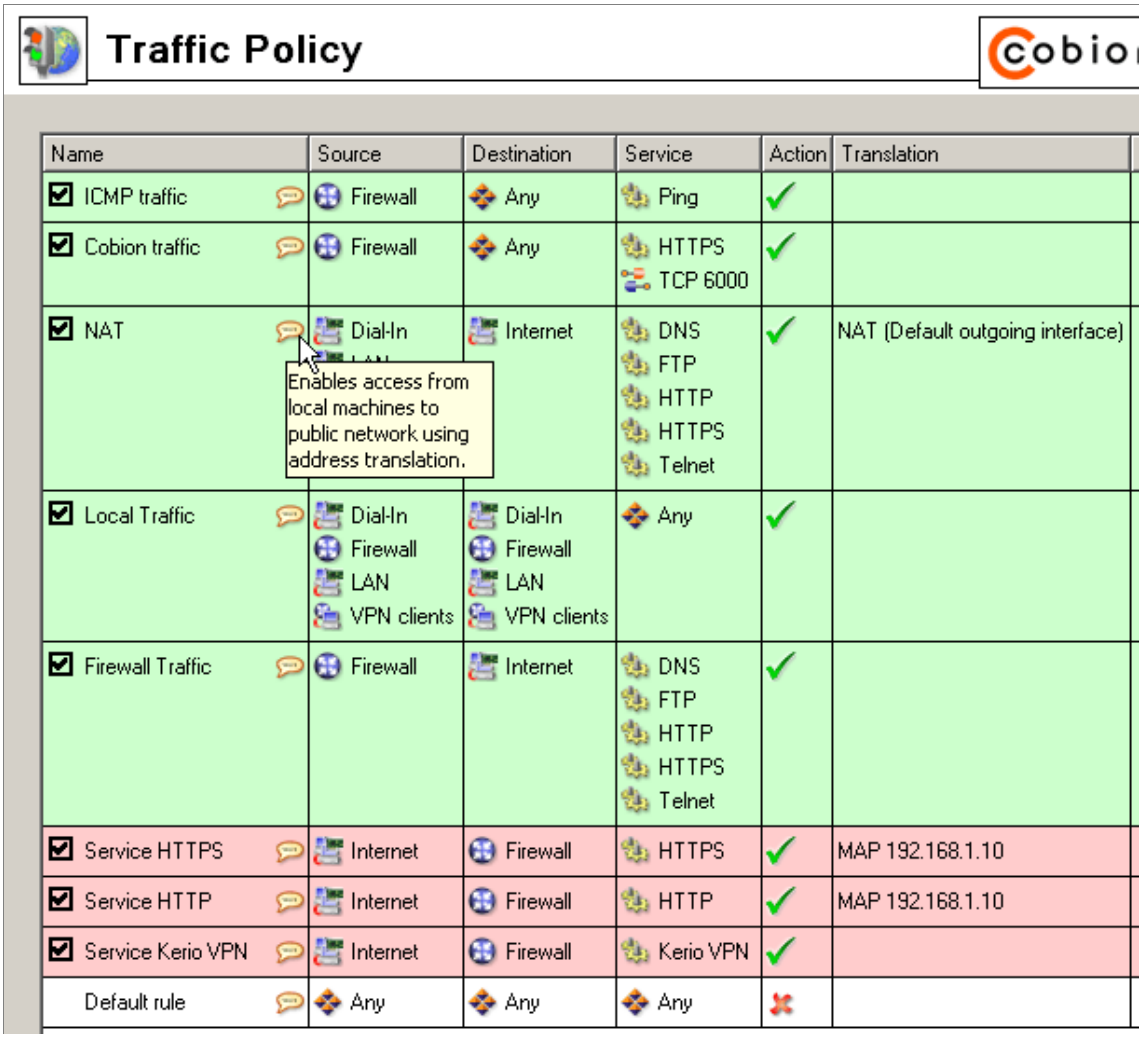
Step 7 — generating the rules

In the last step an information window warns users that the traffic policy will be built upon the inserted data and all the existing data will be deleted and removed with the new rules.

Warning: This is the last chance to cancel the process and keep the existing traffic policy. Click on the *Finish* button to delete the existing rules and replace them with the new ones.

Rules Created by the Wizard

The traffic policy is better understood through the traffic rules created by the Wizard in the previous example.

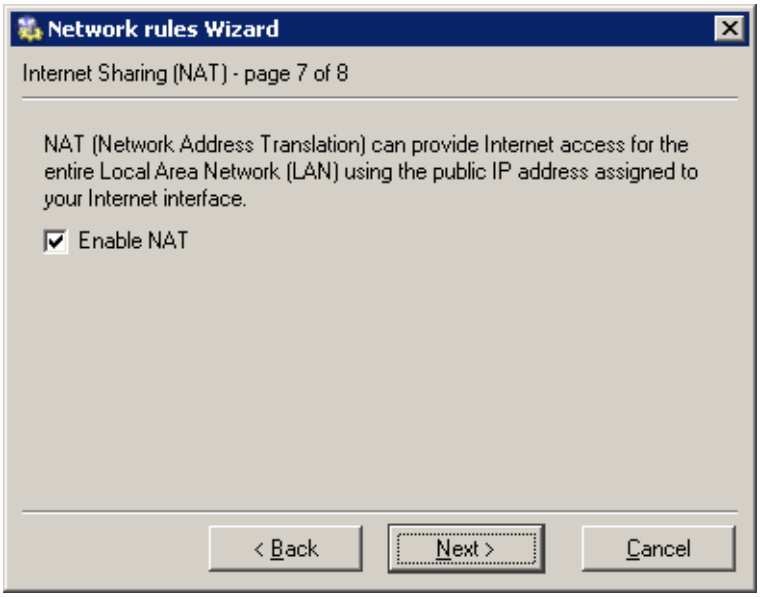


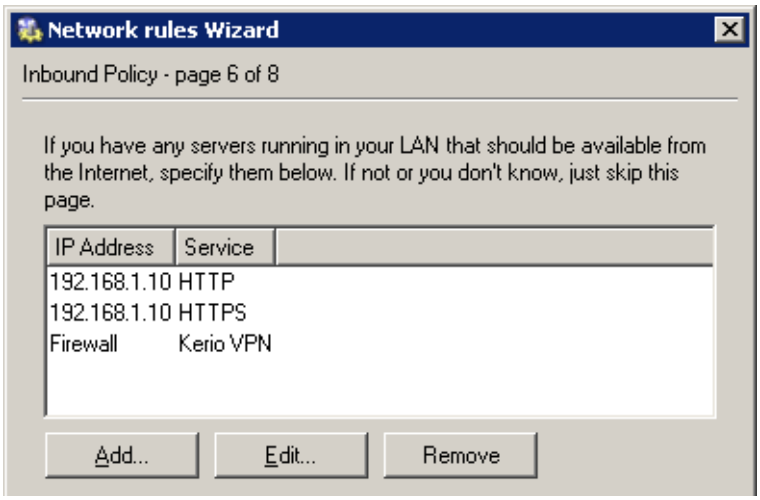
ICMP traffic This rule can be added whenever needed with no respect to settings within individual steps. You can use the *PING* command to send a request on a response from the *WinRoute* host. Important issues can be debugged using this command (i.e.Internet connection functionality can be verified).

Note: The *ICMP traffic* rule does not allow clients to use the *PING* command from the local network to the Internet. If you intend to use the command anyway, you must add the *Ping* feature to the *NAT* rules (for details see chapter 5.2).

Cobion Traffic If *Cobion* is used (the *Cobion OrangeFilter* module for classification of Websites), this rule is used to allow communication with corresponding databases. Do not disable this traffic, otherwise *Cobion* might not function well.

NAT If this rule is added, the source (private) addresses in all packets directed from the local network to the Internet will be substituted with addresses of the interface





connected to the Internet (see the Wizard, steps 3 and 6). However, only services selected within step 4 can be accessed.

The *Dial-In* interface is also included in the *Source* column. This means that all *RAS* clients connected to the server can use the *NAT* technology to access the Internet.

Local Traffic This rule enables all traffic between local hosts with the *WinRoute* host. The *Source* and *Destination* items within this rule include all *WinRoute* host's interfaces except the interface connected to the Internet (this interface has been chosen in step 3).

In this rule, the *Source* and *Destination* items cover also the *Dial-In* and the *VPN clients* interfaces. This means that the *Local Traffic* rule also allows traffic between local hosts and *RAS* clients/VPN clients connected to the server.

Note: Access to the *WinRoute* host is not limited as the Wizard supposes that this host belongs to the local network. Limitations can be done by modification of an appropriate rule or by creating a new one. An inconvenient rule limiting access to the *WinRoute* host might block remote administration or it might cause some Internet services to be unavailable (all traffic directed to the Internet passes through this host).

Firewall Traffic This rule enables access to certain services from the *WinRoute* host. It is similar to the *NAT* rule except from the fact that this rule does not perform IP translation (this host connects to the Internet directly).

HTTP and HTTPS These rules map all *HTTP* and *HTTPS* services running at the host with the `192.168.1.10` IP address (step 6). These services will be available on IP addresses of the external interface (step 3).

Default rule This rule denies all communication that is not allowed by other rules. The default rule is always listed at the end of the rule list and it cannot be removed.

The default rule allows the administrator to select what action will be taken with undesirable traffic attempts (*Deny* or *Drop*) and to decide whether packets or/and connections will be logged.

Note: To see detailed descriptions of traffic rules refer to chapter 5.2.

5.2 Definition of Custom Traffic Rules

To fine-tune the *WinRoute* settings, you can define your own rules or edit the rules generated by the wizard. Advanced administrators can create all the rules according to their specific needs without using the wizard.

Note: If you would like to control user connections to WWW or FTP servers, use the special tools available in *WinRoute* (see chapter 6) rather than traffic rules.

How traffic rules work

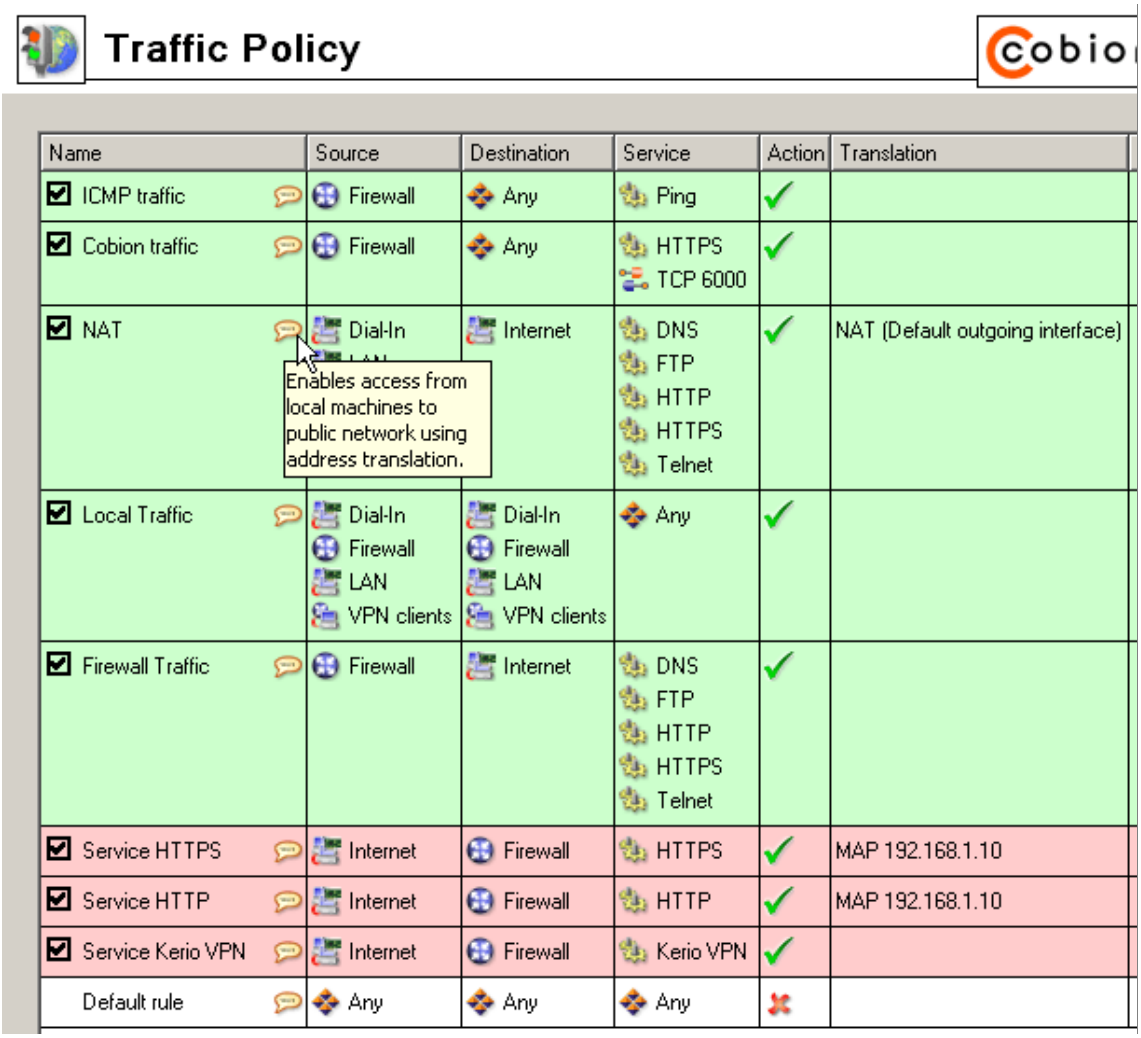
The traffic policy consists of rules ordered by their priority. When the rules are applied they are processed from the top downwards and the first suitable rule found is applied. The order of the rules can be changed with the two arrow buttons on the right side of the window.

An explicit rule denying all traffic is shown at the end of the list. This rule cannot be edited or removed. If there is no rule to allow particular network traffic, then the "catch all" deny rule will discard the packet.

Rule definitions

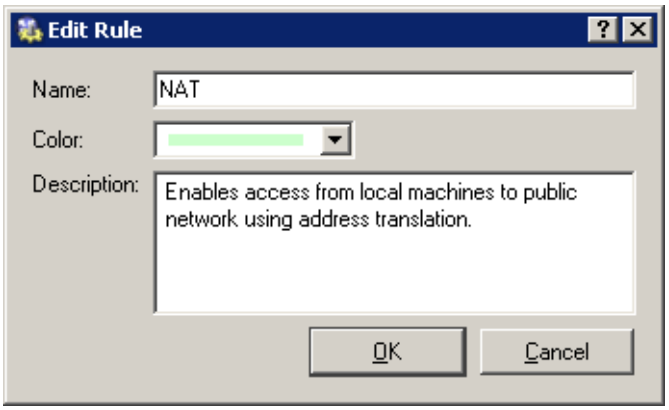
The traffic rules are displayed in the form of a table, where each rule is represented by a row and rule properties (name, conditions, actions — for details see below) are described in the columns. Left-click in a selected field of the table (or right-click a rule and choose the *Edit...* option in the context menu) to open a dialog where the selected item can be edited.

To define new rules press the *Add* button. Move the new rule within the list using the arrow buttons.



Name Name of the rule. It should be brief and unique. More detailed information can be included in the *Description* entry.

Matching fields next to names can be either ticked to activate or unticked to disable. If a particular field is empty, *WinRoute* will ignore the rule. This means that you need not remove and later redefine these rules when troubleshooting a rule.



The background color of each row can be defined as well. To set the color of the list background right click in a cell belonging to the desired row in the *Name* column and select *Edit name and color*.

Any text describing the particular rule may be used to specify the *Description* entry (up to 1024 characters). Specification of this entry is optional.

If the description is specified, the "bubble" symbol is displayed in the *Name* column next to the rule name. Place the mouse pointer over the bubble to view the rule description.

It is recommended to describe all created rules for better reference (automatic descriptions are provided for rules created by the wizard).

Note: Descriptions and colors do not affect rule functionality.

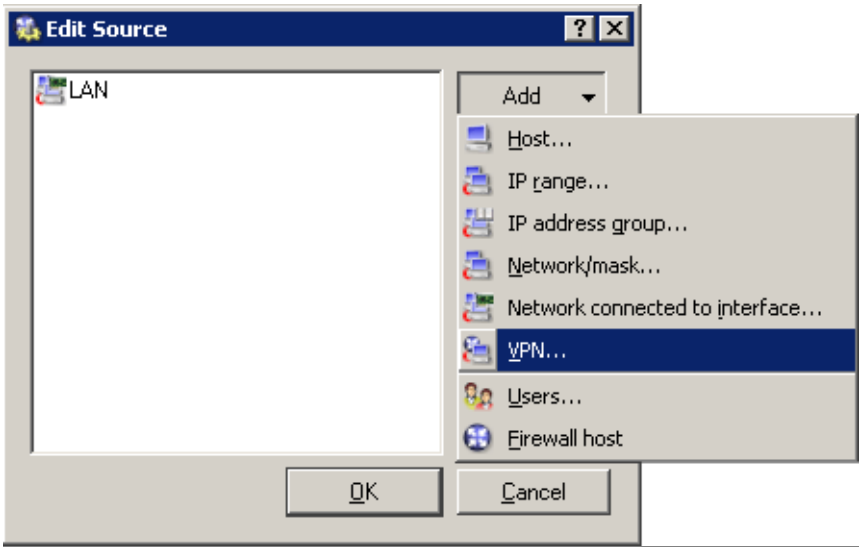
Source and Destination Definition of the source or destination of the traffic defined by the rule.

A new source or destination item can be defined after clicking the *Add* button:

- *Host* — the host IP address or name (e.g. `192.168.1.1` or `www.company.com`)

  Warning: If either the source or the destination computer is specified by DNS name, *WinRoute* tries to identify its IP address while processing a corresponding traffic rule.

  If no corresponding record is found in the cache, the *DNS forwarder* forwards the query to the Internet. If the connection is realized by a dial-up which is currently



hung-up, the query will be sent after the line is dialed. The corresponding rule is disabled unless IP address is resolved from the DNS name. Under certain circumstances denied traffic can be let through while the denial rule is disabled (such connection will be closed immediately when the rule is enabled again).

For the reasons mentioned above we recommend you to specify source and destination computer only through IP addresses in case that you are connected to the Internet through a dial-up!

- *Network* — subnet defined with network address and mask

  (e.g. `192.168.1.0/255.255.255.0`)

- *IP range* — e.g. `192.168.1.10—192.168.1.20`

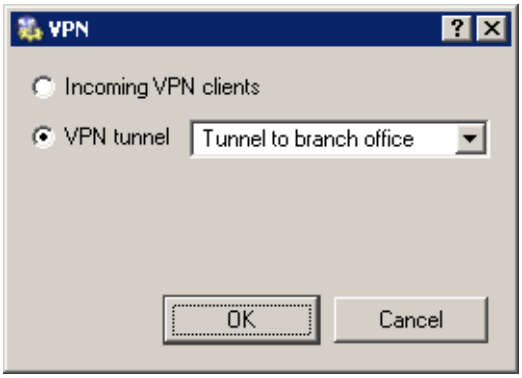
- *Subnet with mask* — subnet defined by network address and mask (e.g. `192.168.1.0/255.255.255.0`)

- *Network connected to interface* — This represents all IP addresses which reside behind the particular interface.

- *VPN* — virtual private network (created with the *WinRoute* VPN solution). This option can be used to add the following items:

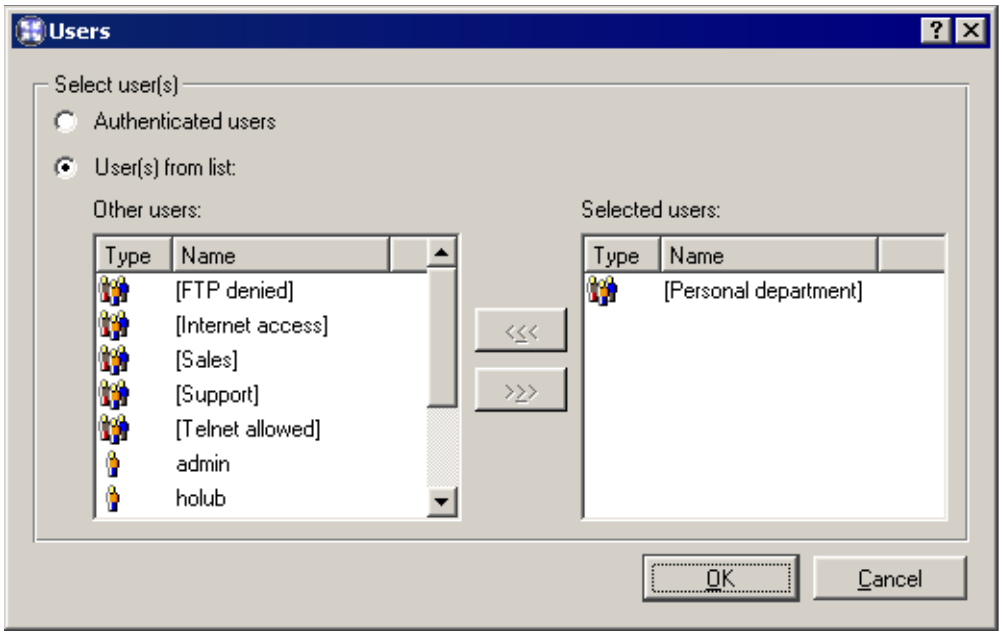
  - *Incoming VPN connections (VPN clients)* — all VPN clients connected to the *WinRoute* VPN server via the *Kerio VPN Client*

  - *Incoming VPN connections (VPN tunnel)* — network connected to this server from a remote server via the VPN tunnel



For detailed description on the *WinRoute* VPN solution refer to chapter 12.

- *Users* — users or groups that can be chosen in a special dialog



The *Authenticated users* option makes the rule valid for all users authenticated to the firewall (see chapter 8.2).

In the traffic policy, each user/group or host is represented by IP address from which it/he/she is connected (for more details about user authentication see chapter 8.2).

Notes:

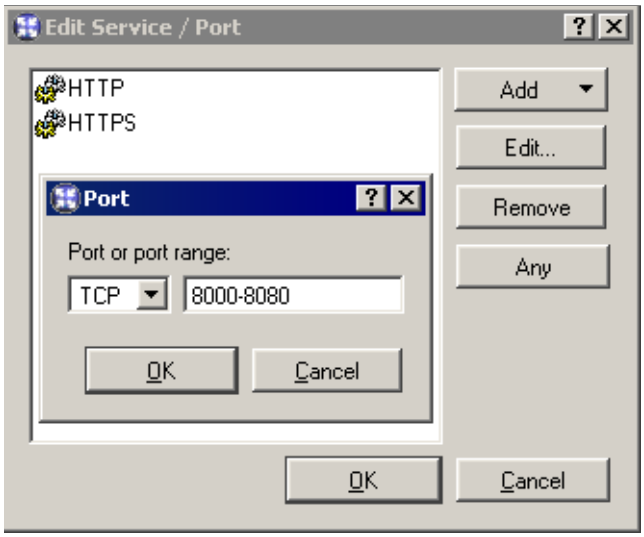
1. If you require authentication for any rule, it is necessary to ensure that a rule exists to allow users to connect to the firewall authentication page. This service uses TCP port 4080 for HTTP and 4081 for HTTPS.

2. If you use HTTP, *WinRoute* can automate user re-direction to the authentication page (for details see chapter 6.1). Other services do not allow this feature. Users should be informed that they are required to pass through the authentication page prior to accessing demanded services (see chapters 8 and 8.2).

- *Firewall* — a special address group including all interfaces of the host where the firewall is running. This option can be used for example to permit traffic between the local network and the *WinRoute* host.

Note: Use the *Any* button to replace all defined items with the *Any* item (this item is also used by default for all new rules). This item will be removed automatically when at least one new item is added.

Service Definition of service(s) on which the traffic rule will be applied. Any number of services defined either in *Configurations / Definitions / Services* or using protocol and port number (or by port range — a dash is used to specify the range) can be included in the list.



Notes:

1. If the protocol inspector of the particular protocol is used for the service definition, this module will be applied on the traffic meeting this rule. If the rule can be applied on all services (the *Any* button), all necessary protocol inspectors will be applied automatically.

If desired, you can define a rule without using protocol inspectors (for details see chapter 9.3) in order to bypass the protocol inspector for particular IP hosts.

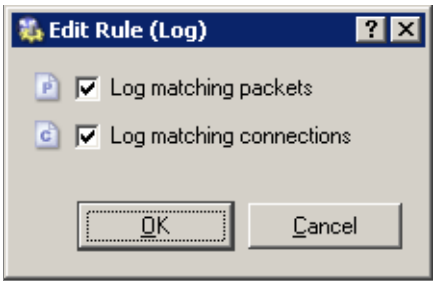
2. To substitute all defined items by the *Any* item use the *Any* button (this is also the default value for creating a new rule). If at least one new service is added, the *Any* item will be removed automatically.

Action Action that will be taken by *WinRoute* when a given packet has passed all the conditions for the rule (the conditions are defined by the *Source*, *Destination* and *Service* items). The following actions can be taken:

- *Permit* — traffic will be allowed by the firewall
- *Deny* — client will be informed that access to the address or port is denied. The client will be warned promptly, however, it is informed that the traffic is blocked by firewall.
- *Drop* — all packets that fit this rule will be dropped by firewall. The client will not be sent any notification and will consider the action as a network outage. The action is not repeated immediately by the client (it expects a response and tries to connect later, etc.).

Note: It is recommended to use the *Deny* option to limit the Internet access for local users and the *Drop* option to block access from the Internet.

Log The following actions can be taken to log traffic:



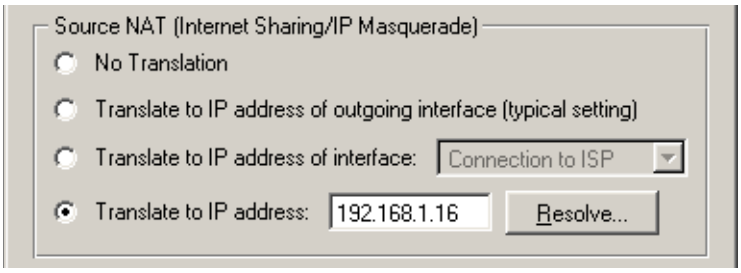
- *Log matching packets* — all packets matching with rule (permitted, denied or dropped, according to the rule definition) will be logged in the *Filter* log.
- *Log matching connections* — all connections matching this rule will be logged in the *Connection* log (only for permit rules).Individual packets included in these connections will not be logged.

  Note: Connections cannot be logged for deny nor drop rules.

Translation Source or/and destination IP address translation.

The source IP address translation can be also called IP masquerading or Internet connection sharing. The source (private) IP address is substituted by the IP address of the interface connected to the Internet in packets routed from the local network to the Internet. Therefore, the entire local network can access the Internet transparently, but it is externally considered as one host.

IP translation is defined as follows:

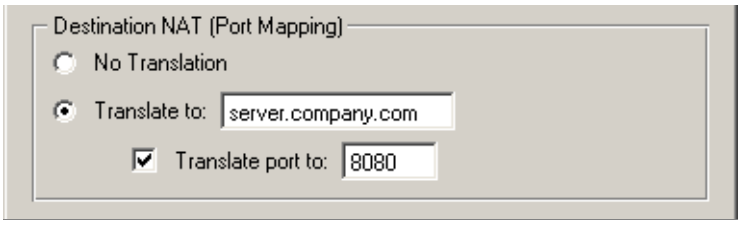


- *No Translation* — source address is not modified. This option is set by default and it is not displayed within traffic rules.
- *Translate to IP address of outgoing interface* — *WinRoute* will translate the source address of an outgoing packet to the IP address of the network interface from where the packet will be forwarded.
- *Translate to IP address of interface* — selection of an interface. IP address of the appropriate packet will be translated to the primary address of this interface. This option is relevant if the return path should be different than the upstream path.
- *Translate to IP address* — an IP address to which the source address will be translated. (i.e. secondary IP address of an interface connected to the Internet). If you only know DNS name of your host, use the *Resolve* button to translate the DNS name to IP address.

  Note: The IP address must be assigned to an interface (bound by TCP/IP stack) of the *WinRoute* host!

Destination address translation (also called port mapping) is used to allow access to services hosted behind the firewall. All incoming packets that meet defined rules are re-directed to a defined host (destination address is changed). From the client's point of view, the service is running on the IP address of the Firewall.

Options for destination NAT (port mapping):



- *No Translation* — destination address will not be modified.
- *Translate to* — IP address that will substitute the packet's destination address. This address also represents the IP address of the host on which the service is actually running.

  The *Translate to* entry can be also specified by DNS name of the destination computer. In such cases *WinRoute* finds a corresponding IP address using a DNS query.

  Warning: We recommend you not to use names of computers which are not recorded in the local DNS since rule is not applied until a corresponding IP address is found. This might cause temporary malfunction of the mapped service.
- *Translate port to* — during the process of IP translation you can also substitute the port of the appropriate service. This means that the service can run at a port that is different from the port from which it is mapped.

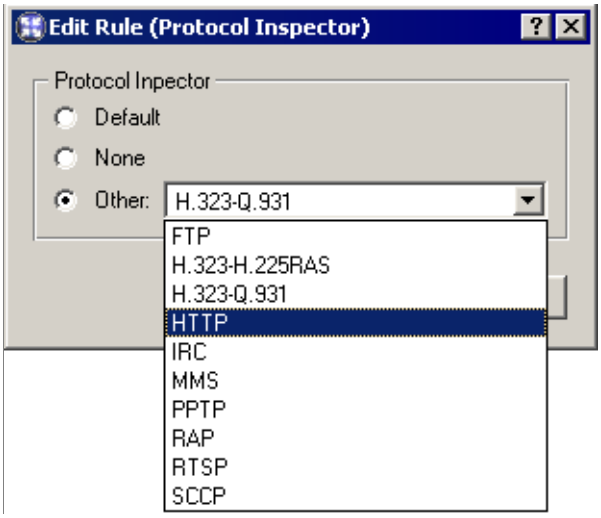
  Note: This option cannot be used unless only one service is defined in the *Service* entry within the appropriate traffic rule and this service uses only one port or port range.

The following columns are hidden by the default settings of the *Traffic Policy* dialog:

Valid on Time interval within which the rule will be valid. Apart from this interval *WinRoute* ignores the rule.

The special *always* option can be used to disable the time limitation (it is not displayed in the *Traffic Policy* dialog).

Protocol Inspector Selection of a protocol inspector that will be applied on all traffic meeting the rule. You can choose from the following options:



- *Default* — all necessary protocol inspectors (or inspectors of the services listed in the *Service* entry) will be applied on traffic meeting this rule.
- *None* — no inspector will be applied (regardless of how services used in the *Service* item are defined).
- *Other* — selection of a particular inspector which will be used on traffic meeting this rule (all *WinRoute's* protocol inspectors are available).

  Warning: Do not use this option unless the appropriate traffic rule defines a protocol belonging to the inspector. Functionality of the service might be affected by using an inappropriate inspector.

Note: Use the *Default* option for the *Protocol Inspector* item if a particular service (see the *Service* item) is used in the rule definition (the protocol inspector is included in the service definition).

5.3 Basic Traffic Rule Types

In this chapter you will find some rules used to manage standard configurations. Using these examples you can easily create a set of rules for your network configuration.

IP Translation

IP translation (NAT) is a term used for the exchange of a private IP address in a packet going out from the local network to the Internet with the IP address of the Internet interface of the *WinRoute* host. The following example shows an appropriate traffic rule:

Source Interface connected to the private local network.

If the network includes more than one segment and each segment is connected to an individual interface, specify all the interfaces in the *Source* entry.

If the local network includes other routers, it is not necessary to specify all interfaces (the interface which connects the network with the *WinRoute* host will be satisfactory).

Destination Interface connected to the Internet.

Service This entry can be used to define global limitations for Internet access. If particular services are defined for IP translations, only these services will be used for the IP translations and other Internet services will not be available from the local network.

Action To validate a rule one of the following three actions must be defined: Permit, Drop, Deny.

Translation In the *Source NAT* section select the *Translate to IP address of outgoing interface* option (the primary IP address of the interface via which packets go out from the *WinRoute* host will be used for NAT).

To use another IP address for the IP translation, use the *Translate to IP address* option and specify the address. The address should belong to the addresses used for the Internet interface, otherwise IP translations will not function correctly.

Warning: Only in very specific and unique situations is it necessary to define both source and destination NAT. For example, you are hosting a service on the LAN that requires a port mapping, however the local server cannot have a default gateway, or it uses a gateway other than *WinRoute*. In this case it is possible to perform source NAT on traffic passed to the internal server so that it will reply back to the *WinRoute* firewall.

Note: The previously defined rule allows outgoing traffic initiated from the local network to the Internet. It is also necessary to define a rule to allow traffic initiated from the *WinRoute* host (defined by source *Firewall*). Because the *WinRoute* host is directly connected to the Internet, it is not necessary to enable translation. The default "catch all" rule at the bottom of the filter list will enforce stateful packet inspection of the *WinRoute* host.

Name	Source	Destination	Service	Action	Translation
Firewall traffic	Firewall	Any	Any	✓	

Port mapping

Port mapping allows services hosted on the local network (typically in private networks) to become available over the Internet. The locally hosted server would behave as if it existed directly on the Internet. The traffic rule therefore must be defined as in the following example:

Name	Source	Destination	Service	Action	Translation
Web server	Internet	192.168.1.10	HTTP HTTPS	✓	MAP 192.168.1.10

Source Interface connected to the Internet (requests from the Internet will arrive on this interface).

Destination The *WinRoute* host labelled as *Firewall*, which represents all IP addresses bound to the firewall host.

Service You can select one of the predefined services (see chapter 9.3) or define an appropriate service with protocol and port number.

Any service that is intended to be mapped to one host can be defined in this entry. To map services for other hosts you will need to create a new traffic rule.

Action Select the *Allow* option, otherwise all traffic will be blocked and the function of port mapping will be irrelevant.

Translation In the *Destination NAT (Port Mapping)* section select the *Translate to IP address* option and specify the IP address of the host within the local network where the service is running.

Using the *Translate port to* option you can map a service to a different port. This allows services to be available on non-standard ports without the necessity of modifying the port used by the server application.

Warning: In the *Source NAT* section should be set to the *No Translation* option. Combining source and destination IP address translation is relevant under special conditions only .

Note: For proper functionality of port mapping, the locally hosted server must point to the *WinRoute* firewall as the default gateway. Otherwise, it will be necessary to enable *Source NAT* in addition to *Destination NAT* .

Multihoming

Multihoming is a term used for situations when one network interface connected to the Internet uses multiple public IP addresses. Typically, multiple services are available through individual IP addresses (this implies that the services are mutually independent).

Example: In the local network a web server `web1` with IP address `192.168.1.100` and a web server `web2` with IP address `192.168.1.200` are running in the local network. The interface connected to the Internet uses two public IP addresses — `63.157.211.10` and `63.157.211.11`. We want the server `web1` to be available from the Internet at the IP address `63.157.211.10`, the server `web2` at the IP address `63.157.211.11`.

The two following traffic rules must be defined in *WinRoute* to enable this configuration:

Name	Source	Destination	Service	Action	Translation
Web server #1 mapping	Internet	63.157.211.10	HTTP	✓	MAP 192.168.1.100
Web server #2 mapping	Internet	63.157.211.11	HTTP	✓	MAP 192.168.1.200

Source Interface which is connected to the Internet (incoming requests from Internet clients will be accepted through this interface).

Destination An appropriate IP address of the interface connected to the Internet (use the *Host* option for insertion of an IP address).

Service Service which will be available through this interface (the *HTTP* service in case of a Web server).

Action Use the *Permit* option, otherwise the traffic will be blocked.

Translation Go to the *Destination NAT (Port Mapping)* section, select the *Translate to IP address* option and specify IP address of a corresponding Web server (`web1` or `web2`).

Limiting Internet Access

Access to Internet services can be limited in several ways. In the following examples, the limitation rules use IP translation. There is no need to define other rules as all traffic that would not meet these requirements will be blocked by the default "catch all" rule.

Other methods of Internet access limitations can be found in the *Exceptions* section (see below).

Note: Rules mentioned in these examples can be also used if *WinRoute* is intended as a neutral router (no address translation) — in the *Translation* entry there will be no translations defined.

1. Allow access to selected services only. In the translation rule in the *Service* entry specify only those services that are intended to be allowed.

Name	Source	Destination	Service	Action	Translation
NAT	LAN	Internet	DNS FTP HTTP HTTPS Telnet	✓	NAT (Default outgoing interface)

2. Limitations sorted by IP addresses. Access to particular services (or access to any Internet service) will be allowed only from selected hosts. In the *Source* entry define the group of IP addresses from which the Internet will be available. This group must be formerly defined in *Configuration / Definitions / Address Groups* (see chapter 10.4).

Name	Source	Destination	Service	Action	Translation
NAT for allowed hosts	Internet access	Internet	Any	✓	NAT (Default outgoing interface)

   Note: This type of rule should be used only if each user has his/her own host and the hosts have static IP addresses.

3. Limitations sorted by users. Firewall monitors if the connection is from an authenticated host. In this case you must define user accounts in *WinRoute* and users must authenticate using the firewall authentication page before access is granted to the specified service.

Name	Source	Destination	Service	Action	Translation
NAT for a group of users	[Internet access]	Internet	Any	✓	NAT (Default outgoing interface)

   Alternatively you can define the rule to allow only authenticated users to access specific services. Any user that has a user account in *WinRoute* will be allowed to access the Internet after authenticating to the firewall. Firewall administrators can easily monitor which services and which pages are opened by each user (it is not possible to connect anonymously).

Name	Source	Destination	Service	Action	Translation
NAT for a group of users	Authenticated users	Internet	Any	✓	NAT (Default outgoing interface)

Note: Detailed information about user connections to the firewall can be found in chapter 8.2.

The rules mentioned above can be combined in various ways (i.e. a user group can be allowed to access certain Internet services only).

Exceptions

You may need to allow access to the Internet only for a certain user/address group, whereas all other users should not be allowed to access this service.

This will be better understood through the following example (how to allow a user group to use the *Telnet* service for access to servers in the Internet). Use the two following rules to meet these requirements:

- First rule will deny selected users (or a group of users/IP addresses, etc.) to access the Internet.
- Second rule will deny the other users to access this service.

Name	Source	Destination	Service	Action
Allow Telnet for a group of users	[Telnet allowed]	Internet	Telnet	✓
Forbid Telnet	Any	Internet	Telnet	✗

Chapter 6

Content Filtering

WinRoute provides a wide range of features to filter traffic using HTTP and FTP protocols.

Here are the main purposes of HTTP and FTP content filtering:

- to block access to undesirable Web sites (i.e. pages that do not relate to employees' work)
- to block certain types of files (i.e. illegal content)
- to block or to limit viruses, worms and Trojan horses

HTTP protocol — Web pages filtering:

- access limitations according to URL (substrings contained in URL addresses)
- blocking of certain HTML items (i.e. scripts, ActiveX objects, etc.)
- filtering based on classification by the *Cobion Orange Filter* plug-in (worldwide Website classification database)
- limitations based on occurrence of denied words (strings)
- antivirus control of downloaded objects

FTP protocol — control of access to FTP servers:

- access to certain FTP servers is denied
- limitations based on or file names
- transfer of files is limited to one direction only (i.e. download only)
- certain FTP commands are blocked
- antivirus control of transferred files

Content filtering requirements

The following conditions must be met to ensure smooth functionality of content filtering:

1. Traffic must be controlled by an appropriate protocol inspector.

   Note: An appropriate protocol inspector is activated automatically unless its use is denied by traffic rules. For details see chapter 5.2.

2. Connections must not be encrypted. SSL encrypted traffic (HTTPS and FTPS protocols) cannot be monitored. In this case you can block access to certain servers using traffic rules (see chapter 5.2).

   Note: If the proxy server is used (see chapter 4.5), It is also possible to filter HTTPS servers (e.g. `https://www.kerio.com/`). However, it is not possible to filter individual objects at these servers.

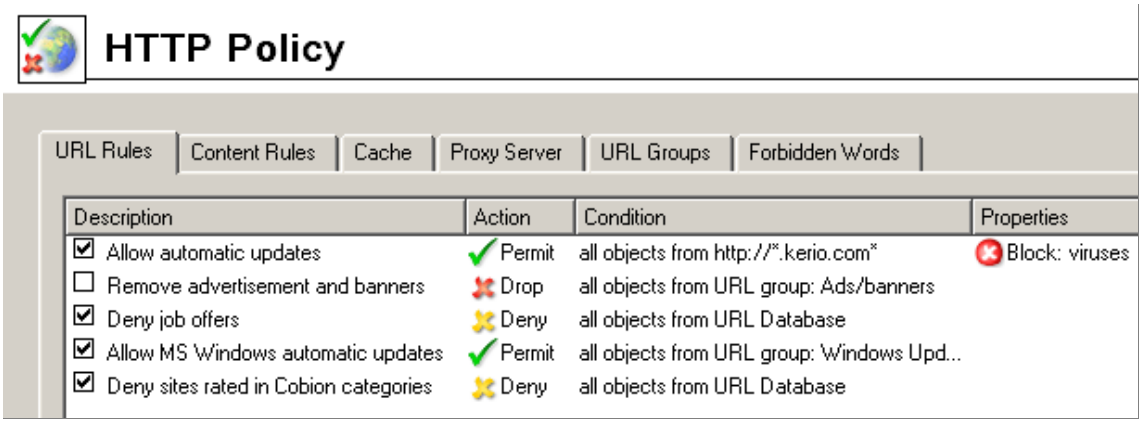
3. FTP protocols cannot be filtered if the secured authentication (*SASO*) is used.

Note: WinRoute provides only tools for filtering and access limitations. Decisions on which Web sites and file types will be blocked must be made by the administrator (or another qualified person).

6.1 URL Rules

These rules allow the administrator to limit access to Web pages with URLs that meet certain criteria. URL rules can also enforce user authentication by re-directing browsers to the authentication page (see chapter 8.2). This means that the authentication page is not opened manually by the user when accessing a page that requires authentication.

To define URL rules go to the *URL Rules* tab in *Configuration / Content Filtering / HTTP Policy*.



Rules are read starting from the top. The list can be re-ordered using the arrow buttons at the right side of the dialog window. If a requested URL passes through all rules without any match, access to the site is allowed. All URLs are allowed by default (unless denied by a URL rule).

The following items (columns) can be available in the *URL Rules* tab:

- *Description* — description of a particular rule (for reference only). You can use the checking box next to the description to enable/disable the rule (for example, for a certain time).
- *Action* — action which will be performed if all conditions of the rule are met (*Permit* — access to the page will be allowed, *Deny* — connection to the page will be denied and denial information will be displayed, *Drop* — access will be denied and a blank page will be opened, *Redirect* — user will be redirected to the page specified in the rule).
- *Condition* — condition which must be met to apply the rule (e.g. URL matches certain criteria, page is included in a particular category of the Cobion database, etc.).
- *Properties* — advanced options for the rule (e.g. anti-virus check, content filtering, etc.).
- *IP Groups* — IP group to which the rule is applied. The IP groups include addresses of clients (workstations of users who connect to the Internet through *WinRoute*).
- *Valid Time* — time interval during which the rule is applied.
- *Users List* — list of users and user groups to which the rule applies.

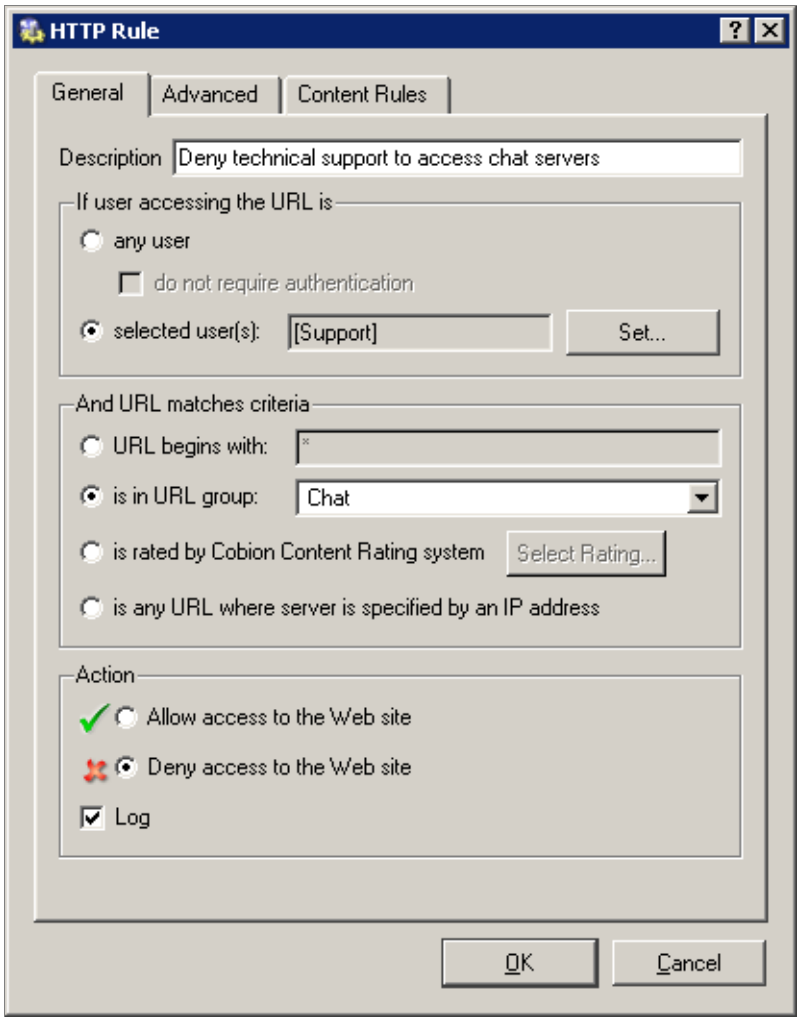
Note: The default *WinRoute* installation includes several predefined URL rules. The rules are inactive by default. *WinRoute* administrators can enable or edit them if desirable.

URL Rules Definition

To create a new rule, select a rule after which the new rule will be added, and click *Add*. You can later use the arrow buttons to reorder the rule list.

Note: URLs which do not match with any URL rule are available for any authenticated user (any traffic permitted by default). To allow accessing only a specific web page group and block access to other web pages, a rule denying access to any URL must be placed at the end of the rule list.

Use the *Add* button to open a dialog for creating a new rule.



Open the *General* tab to set general rules and actions to be taken.

Description Comment on the appropriate rule function (information for *WinRoute's* administrator).

If user accessing the URL is This option specifies on which users the rule will be applied:

- *any user* — for all users (no authentication required).
- *selected user(s)* — for selected users or/and user groups who have authenticated to the firewall.

  Use the *Set* button to open a dialog where users and groups can be selected (hold the *Ctrl* and *Shift* keys to select more users/groups at once).

Note: Specification of users/groups is irrelevant unless combined with a rule that requires user authentication.

And URL matches criteria Specification of URL (or URL group) on which this rule will be applied:

- *URL begins with* — this item can include either entire URL

  (i.e. `www.kerio.com/index.html`) or only a substring of a URL using an asterisk (wildcard matching) to substitute any number of characters (i.e. `*.kerio.com*`)

- *is in URL group* — selection of a URL group which the URL will belong to (see chapter 9.4)

- *is rated by Cobion Content Rating system* — the rule will be applied on all pages matched with a selected category by the *Cobion Orange Filter* plug-in (see chapter 6.3).

  Click on the *Select Rating...* button to select from *Cobion Orange Filter* categories. Read more in chapter 6.3.

- *is any URL where server is given as IP address* — by enabling this option users will not be able to bypass URL based filters by connecting to Web sites by IP address rather than domain name.

Action Selection of an action that will be taken whenever a user accesses a URL meeting a rule:

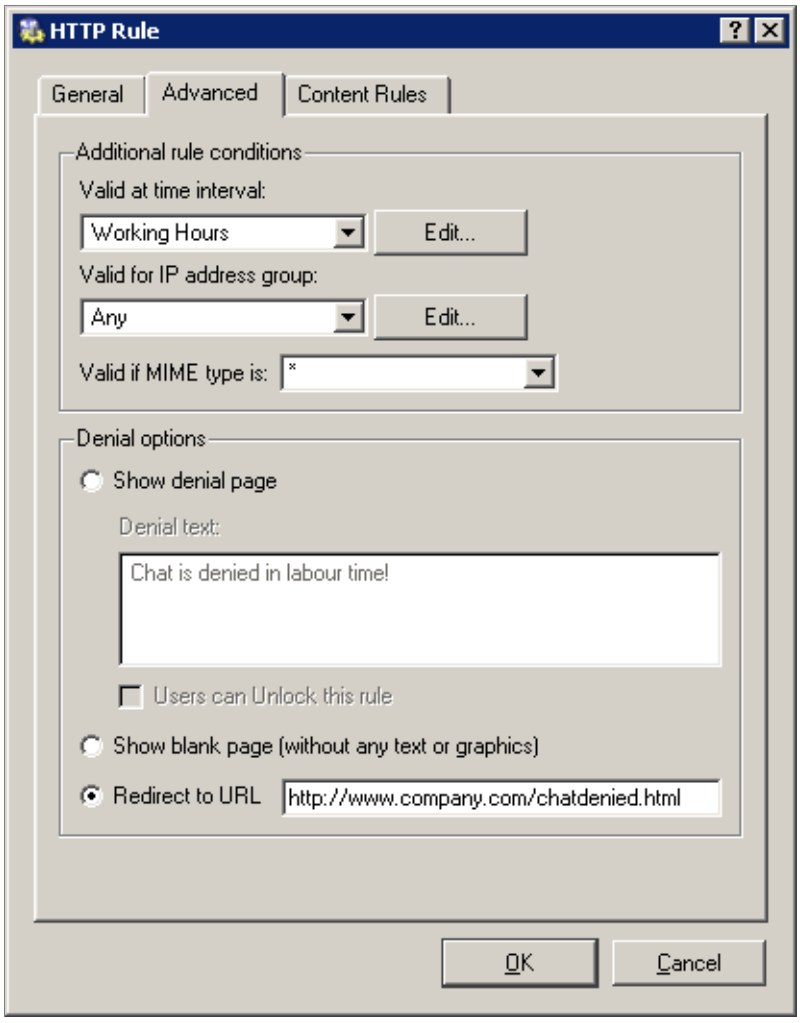
- *Allow access to the Web site*

- *Deny access to the Web site* — requested page will be blocked. The user will be informed that the access is denied or a blank page will be displayed (according to settings in the *Advanced* tab — see below).

Tick the *Log* option to log all pages meeting this rule in the *Filter* log (see chapter 16.9).

Go to the *Advanced* tab to define more conditions for the rule or/and to set options for denied pages.

Valid at time interval Selection of a time interval within which the rule will be valid (out of this interval the rule will be inactive). Use the *Edit* button to open a dialog where time ranges can be modified (for details see chapter 9.2).



Valid for IP address group Selection of IP address group on which the rule will be applied (client addresses). Use the *Any* option if you intend to make the rule independent of client addresses.

Click on the *Edit* button to open a dialog where IP addresses can be edited (for details see chapter 9.1).

Valid if MIME type is The rule will be valid for a certain MIME type only (for example, `text/html` — HTML documents, `image/jpeg` — images in the JPEG format, etc.).

You can either select one of the predefined MIME types or define a new one. An asterisk substitutes any subtype (i.e. `image/*`). An asterisk stands for any MIME type — the rule will be independent of the MIME type.

Denial options Advanced options for denied pages. Whenever a user attempts to open a page that is denied by the rule, *WinRoute* will display:

- a page informing the user that access to the required page is denied as it is blocked by the firewall. This page can also include an explanation of the denial (the *Denial text* item).

  The *Unlock* button will be displayed in the page informing about the denial if the *Users can Unlock this rule* is ticked. Using this button users can force *WinRoute* to open the required page even though this site is denied by a URL rule. The page will be opened for 10 minutes. Each user can unlock a limited number of denied pages (up to 10 pages at once). All unlocked pages are logged in the *Filter* log (see chapter 16.9).

  Notes:

  1. Only subscribed users are allowed to unlock rules.

  2. If any modifications are done within URL rules, all unlock rules are removed immediately.

- a blank page — user will not be informed why access to the required page was denied. It will be as if the server is unavailable and a connection could not be established.

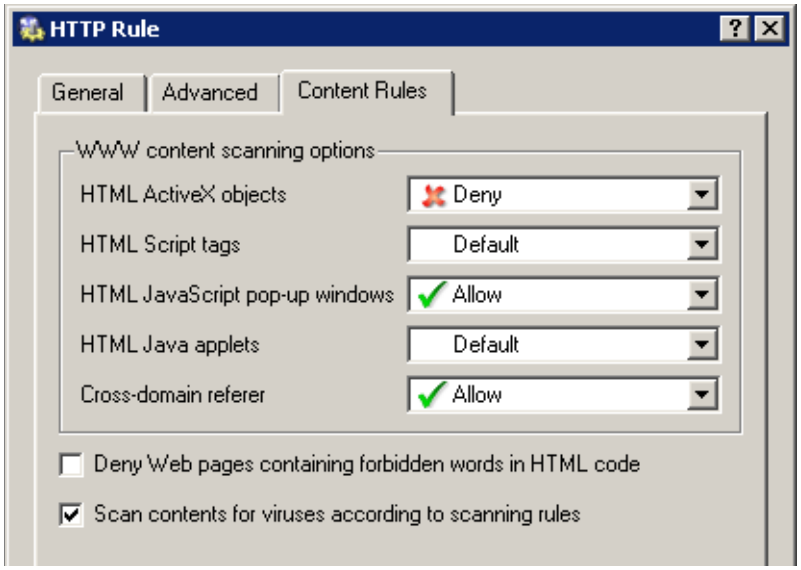
- another page — user's browser will be redirected to the specified URL. This option can be helpful for example to define a custom page with a warning that access to the particular page is denied.

New rules will be added below the rule that had been marked before the *Add* button was used. Use the arrow buttons on the right side of the dialog window to locate the new rule in the list.

You can use the checkboxes next to rules to temporarily disable them without needing to delete and reconfigure the rule if it should be needed at a later time.

Note: Access to URLs which do not meet any rules are implicitly allowed. If you intend to allow access to a limited URL group while denying everything else, you must define a rule that will deny access to any URL (using '*') at the end of the list.

Open the *Content Rules* tab (in the *HTTP Rules* section) to specify details for content filter rules.



WWW content scanning options In this section you can define advanced parameters for filtering of objects contained in Web pages which meet the particular rule (for details refer to chapter 6.2). These parameters will be applied only to users which will not be allowed to "override Content filter rules". Users allowed to override these rules use their custom settings.

One of the following alternatives can be set for each object type:

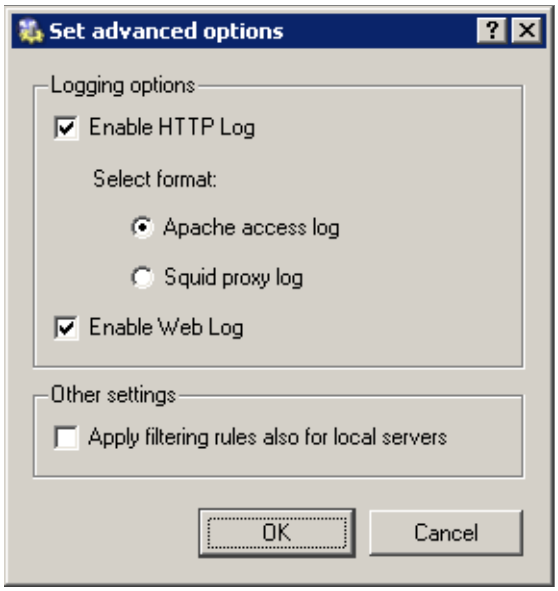
- *Allow* — these objects will be displayed.
- *Deny* — these objects will be filtered out of the page
- *Default* — global rules or custom rules of a particular user will be applied to such objects (this implies that this rule will not affect filtering of such objects)

Deny Web pages containing ... Use this option to deny users to access Web pages containing words/strings defined in the *Configuration/HTTP Policy* section (for details refer to chapter 6.4).

Scan content for viruses according to scanning rules Antivirus check according to settings in the *Configuration / Content Filtering / Antivirus* section will be performed (see chapter 7) if this option is enabled.

HTTP Inspection Advanced Options

Click on the *Advanced* button in the *HTTP Policy* tab to open a dialog where parameters for the HTTP inspection module can be set.



Use the *Enable HTTP Log* and *Enable Web Log* options to enable/disable logging of HTTP queries (opened web pages) to the *HTTP* log (see chapter 16.10) and to the *Web* log (refer to chapter 16.13).

You can also select the format of the log for the *Enable HTTP Log* item (*Apache* access log or *Squid* proxy log). This may be important especially when the log would be processed by a specific analysis tool.

Both *HTTP* and *Web* logs are enabled and the *Apache* option is selected by default.

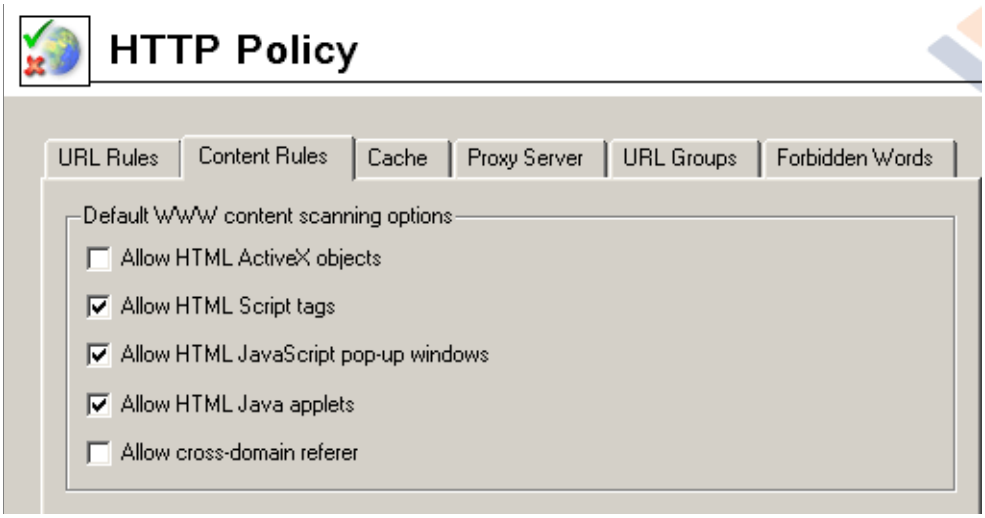
Use the *Apply filtering rules also for local server* to specify whether content filtering rules will be applied to local WWW servers which are available from the Internet (see chapter 5). This option is disabled by default — the protocol inspector only scans HTTP protocol syntax and performs logging of queries (WWW pages) according to the settings.

6.2 Content Rules

In *WinRoute* you can also block certain features contained in HTML pages.

To define content global filtering rules go to the *Content Rules* tab in the *Configuration / Content Filtering/ HTTP Policy* section. Special settings for individual pages can be defined in URL Rules section (refer to chapter 6.1).

These parameters also apply to HTTP traffic of computers from which no user is connected. Special settings are used for users connected through the firewall (see chapter 10.1).



Allow HTML ActiveX objects Microsoft ActiveX features (security defects during implementation of this technology allow the execution of applications on client hosts, apart from other features).

Allow HTML Script tags HTML `<script>` tags — commands of programming languages, such as JavaScript, VBScript, etc.

Allow HTML JavaScript pop-up windows New browser windows are opened automatically — usually commercial pop-up windows.

The `window.open()` method will be blocked in all scripts by *WinRoute* unless this option is active.

Allow HTML Java applets HTML `<applet>` tags (*Java Applet*)

Allow inter-domain referrer A `Referrer` item included in an HTTP header.

This item includes the URL of the page opened prior to the currently opened page. If the *Allow inter-domain referrer* is off, `Referrer` items that include a server name different from the current HTTP request will be blocked.

The *Cross-domain referrer* function protects users' privacy (the `Referrer` item can be monitored to see which pages are opened by each user).

Note: Settings in the *Content Rules* tab are applied on unauthenticated users. Each authenticated user can customize filtering rules at the user preferences page (see chapter 8.3). However, users that are not allowed to *override WWW content rules* (refer to chapter 10.1) cannot permit HTML features that are denied globally.

6.3 Content Rating System (Cobion Orange Filter)

Cobion Orange Filter plug-in is a part of the *Cobion* system which is integrated in *WinRoute*. It enables *WinRoute* to rate Web page content. Each page is sorted into predefined categories. Access to the page will be either permitted or denied according to this classification.

Cobion Orange Filter uses a dynamic worldwide database which includes URLs and classification of Web pages. This database is maintained by special servers that perform page ratings. Whenever a user attempts to access a Web page, *WinRoute* sends a request on the page rating. According to the classification of the page the user will be either allowed or denied to access the page. To speed up URL rating the data that have been once acquired can be stored in the cache and kept for a certain period.

Note: The *Cobion Orange Filter* plug-in was designed and tested especially on pages in English. Efficiency of its appliance on non-English pages is lower (about 70 % of the full efficiency).

Cobion Orange Filter Deployment

To enable classification of Websites by the *Cobion OrangeFilter* plug-in, *Cobion* must be running and all corresponding parameters must be set. For detailed guidance, see chapter 11.4.

Whenever *WinRoute* processes a URL rule that requires classification of pages, the *Cobion Orange Filter* content rating plug-in is activated. The usage will be better understood through the following example that describes a rule denying all users to access pages containing job offers.

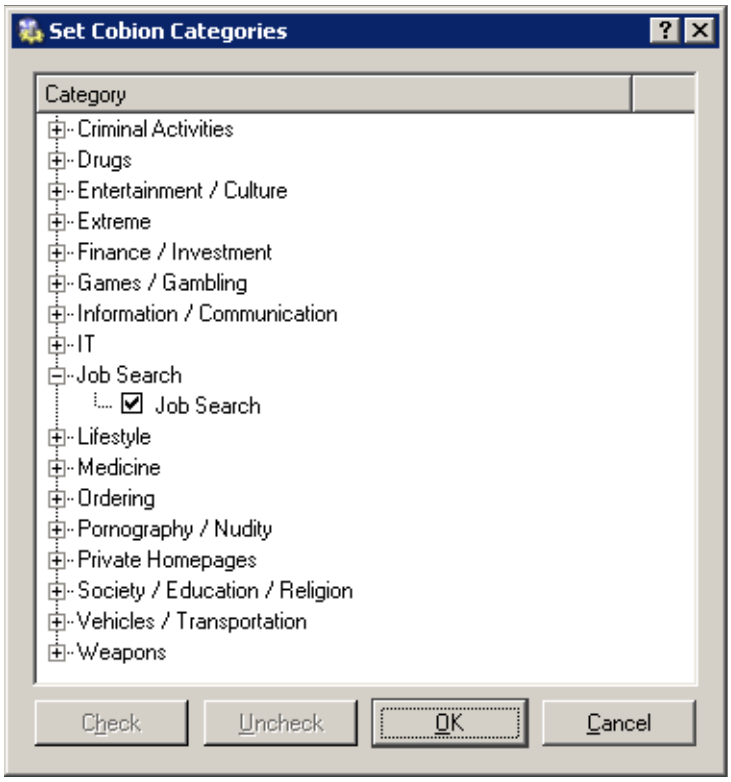
the following rule has been defined in the *URL Rules* tab in *Configuration / Content Filtering / HTTP Rules*:

The *is rated by Cobion Content Rating system* is considered the key parameter. The URL of each opened page will be rated by the *Cobion Orange Filter* plug-in. Access to each page matching with a rating category included in the database will be denied.

Use the *Select Rating* button to open a dialog where *Cobion Orange Filter* rating categories can be chosen. Select the *Job Search* rating category (pages including job offers).

Notes:

1. Use the *Check* button to check all items included in the selected category. You can uncheck all items in the category by clicking *Uncheck*.
2. We recommend you to enable Unlock for rules that use the *Cobion Orange Filter* rating system (the *Users can Unlock this rule* option in the *Advanced* tab). This option will allow users to unlock pages blocked for incorrect classification.



6.4 Filtering by Words

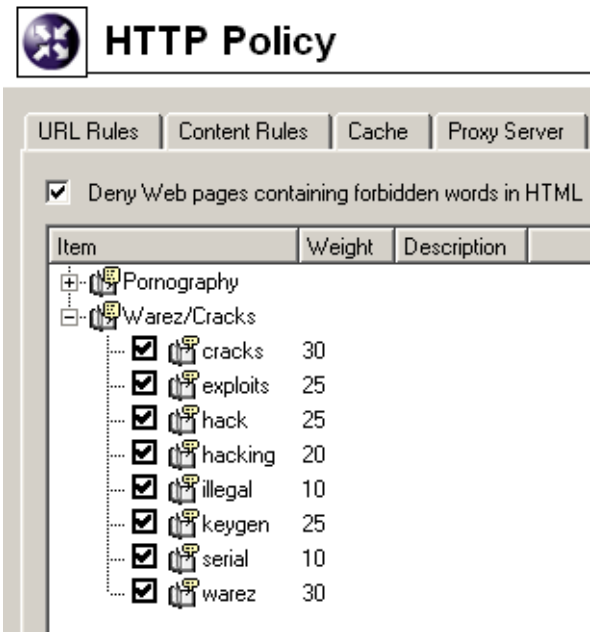
WinRoute can also filter Web pages that include undesirable words. This filtering is applied globally on all HTTP traffic . The filtering is applied after URL rules (only if access to the demanded page is permitted).

This is the filtering principle: Denied words are matched with values, called weight (represented by a whole positive integer). Weights of these words contained in a required page are summed (weight of each word is counted only once regardless of how many times the word is included in the page). If the total weight exceeds the defined limit, the page is blocked.

Words are sorted into groups. This feature only makes *WinRoute* easier to follow. All groups have the same priority and all of them are always tested.

To define word groups go to the *Word Groups* tab in *Configuration / Content Filtering / HTTP Rules*.

Individual groups and words included in them are displayed in form of trees. To switch individual words use matching fields located next to them. Ticked rules will be ignored. Due to this function it is not necessary to remove rules and define them again later .



Note: The following word groups are predefined in the default *WinRoute* installation:

- *Pornography* — Words that typically appear on pages with erotic themes.
- *Warez / Cracks* — . Words that typically appear on pages offering downloads of illegal software, license key generators etc.

All key words in predefined groups are disabled by default. A *WinRoute* administrator can modify the weight for each word.

Deny pages with weight over Upper bound of total page weight (sum of all forbidden words detected at the page). If the total weight of the tested page exceeds this limit, access to the page will be denied (each word is counted only once, regardless of the count of individual words).

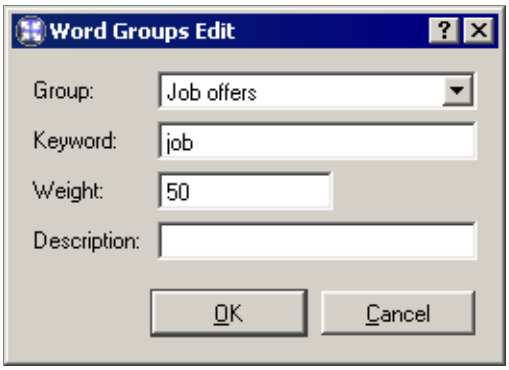
Use the *Add* button to add a new word into a group or to create a new group.

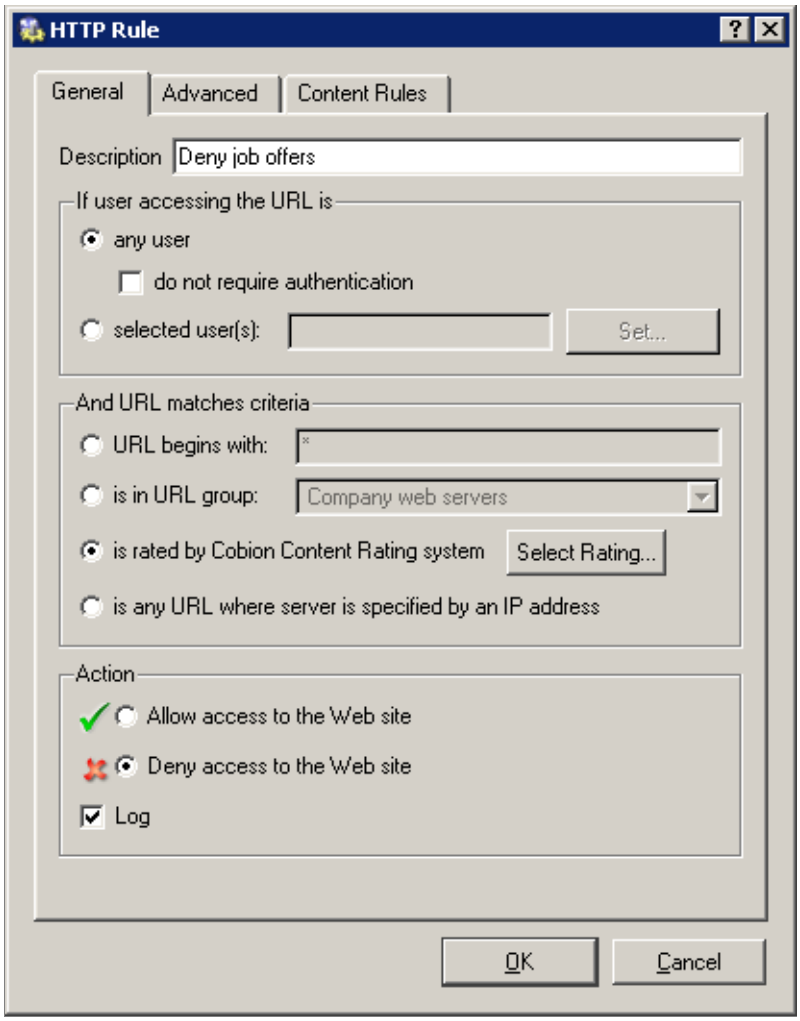
Group Selection of a group to which the word will be included. You can also add a new name to create a new group.

Keyword Forbidden word that is to be scanned for

Weight Word weight (affects decision about the page denial)

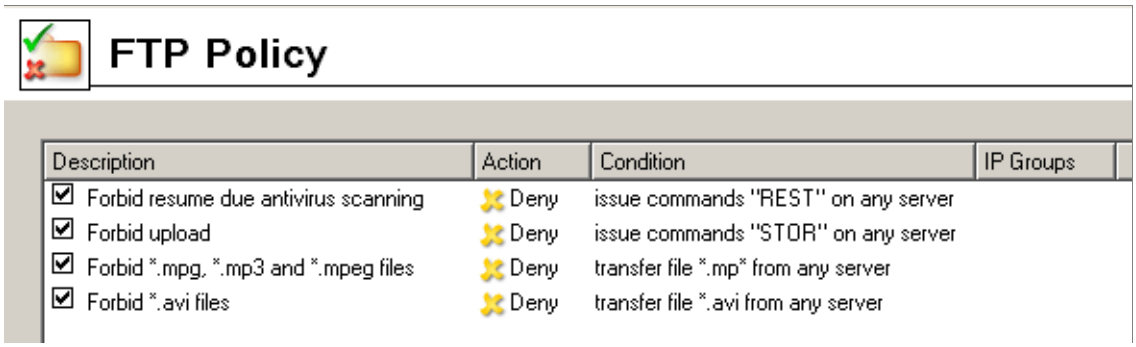
Description A comment on the word or group.





6.5 FTP Policy

To define rules for access to FTP servers go to *Configuration / Content Filtering / FTP Rules*.



Rules in this section are tested from the top of the list downwards (you can order the list entries using the arrow buttons at the right side of the dialog window). Testing is stopped when the first convenient rule is met. If the query does not match any rule, access to the FTP server is implicitly allowed.

Notes:

1. The default *WinRoute* configuration includes a set of predefined rules for FTP traffic. These rules are disabled by default. These rules are available to the *WinRoute* administrators.

2. A rule which blocks completion of interrupted download processes (so called *resume* function executed by the REST FTP command). This function is essential for proper functionality of the antivirus control: for reliable scanning, entire files must be scanned.

   If undesirable, this rule can be disabled. This is not recommended as it might jeopardize scanning reliability. However, there is a more secure way to limit this behavior: create a rule which will allow unlimited connections to a particular FTP server. The rule will take effect only if it is placed before the *Resume* rule.

Use the *Add* button to define a new FTP rule.

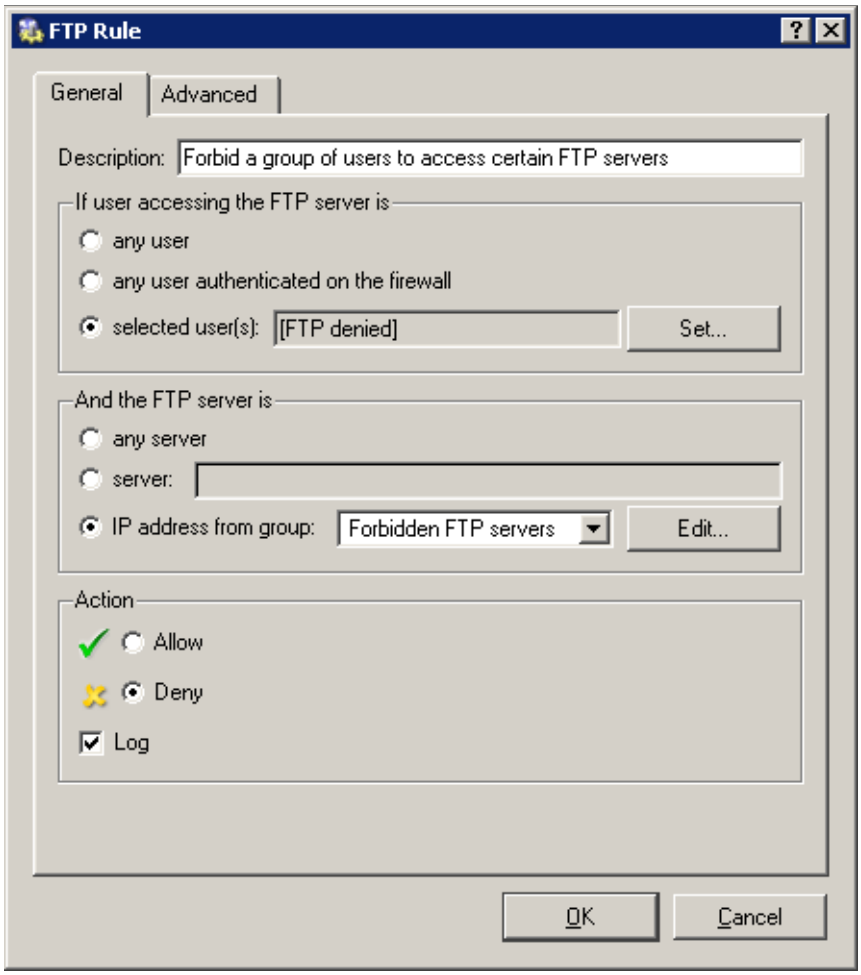
General conditions and actions that are to be taken can be defined in the

General tab.

Description Description of the rule (information for the administrator).

If user accessing the FTP server is Select which users this rule will be applied on:

- *any user* — the rule will be applied on all users (regardless whether authenticated on the firewall or not).
- *any user authenticated on the firewall* — applied on all authenticated users.
- *selected user(s)* — applied on selected users or/and user groups.



Click on the *Set* button to select users or groups (hold the *Ctrl* and the *Shift* keys to select more that one user/group at once).

Note: Rules designed for selected users (or all authenticated users) are irrelevant unless combined with a rule that denies access of non-authenticated users.

And the FTP server is Specify FTP servers on which this rule will be applied:

- *any server* —any FTP server
- *server* — IP address of DNS name of a particular FTP server.

  If an FTP server is defined through a DNS name, *WinRoute* will automatically perform IP address resolution from DNS. The IP address will be resolved immediately when settings are confirmed by the *OK* button (for all rules where the FTP server was defined by a DNS name).

Warning: Rules are disabled unless a corresponding IP address is found!

- *IP address from group* — selection of IP addresses of FTP servers that will be either denied or allowed.

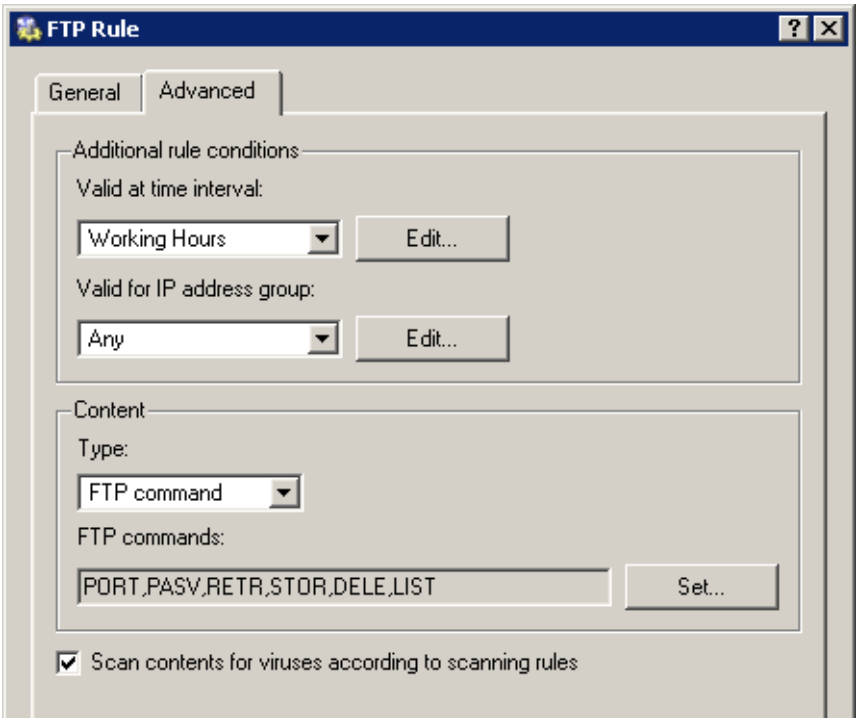
  Click on the *Edit* button to edit IP groups (for details see chapter 9.1).

Action Select an action that will be taken when requirements for users and the FTP server are met:

- *Allow* — *WinRoute* allows connection to selected FTP servers under conditions set in the *Advanced* tab— see below).
- *Deny* — *WinRoute* will block certain FTP commands or FTP connections (according to the settings within the *Advanced* tab).

Use the *Log* option to log all FTP access attempts that have met this rule into the *Filter* log (see chapter 16.9).

Go to the *Advanced* tab to define other conditions that must be met for the rule to be applied and to set advanced options for FTP communication.



Valid at time interval Selection of the time interval during which the rule will be valid (apart from this interval the rule will be ignored). Use the *Edit* button to edit time intervals (for details see chapter 9.2).

Valid for IP address group Selection of IP address group on which the rule will be applied. Client (source) addresses are considered). Use the *Any* option to make the rule independent of clients.

Use the *Edit* button to edit IP groups (for details see chapter 9.1).

Content Advanced options for FTP traffic content.

Use the *Type* option to set a filtering method:

- *Download*, *Upload*, *Download / Upload* — transport of files in one or both directions.

  If any of these options is chosen, you can specify names of files on which the rule will be applied using the *File name* entry. Wildcard matching can be used to specify a file name (i.e. `*.exe` for executables).

- *FTP command* — selection of commands for the FTP server on which the rule will be applied

- *Any* — denies all traffic (any connection or command use)

Scan content for viruses according to scanning rules Use this option to enable/disable scanning for viruses for FTP traffic which meet this rule.

This option is available only for allowing rules — it is meaningless to apply antivirus check to denied traffic.

New rules will be added below the rule marked before using the *Add* button. Use the arrow buttons at the right side of the dialog window to move the rule within the list.

Use matching fields next to appropriate rules to switch rules off. Ticked rules will be ignored. Due to this function it is not necessary to remove rules and define them again later.

Note: Access to FTP servers that do not meet any rules are implicitly allowed. To allow access to a limited number of FTP servers and block other pages, add a new rule (using the wildcard "*") that will deny access to any URL to the end of the list.

Chapter 7

Antivirus Check

WinRoute provides antivirus check of objects (files) transmitted by HTTP, FTP, SMTP and POP3 protocols. In case of HTTP and FTP protocols, the *WinRoute* administrator can specify which types of objects will be scanned.

Any supported antivirus program can be used to perform antivirus checks.

WinRoute supports several antivirus programs developed by various companies, such as Eset Software, Grisoft, F-Secure, etc.). Antivirus licenses must meet the license policy of a corresponding company (usually, the license is limited by the same or higher number of users as *WinRoute* is licensed for, or a server license).

However, supported antiviruses as well as versions and license policy of individual programs may change. For up-to-date information please refer to (`http://www.kerio.com/kwf`).

WinRoute is also distributed in a special version which includes integrated *McAfee* antivirus. External *McAfee Anti-Virus* programs are not supported by *WinRoute*.

7.1 How to choose and setup an antivirus

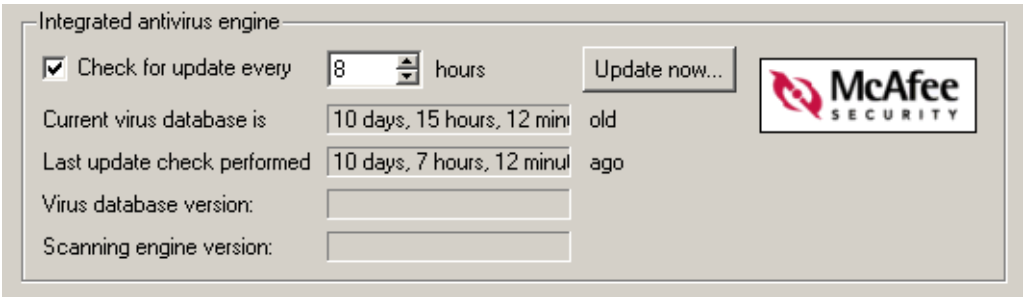
To select an antivirus, open the *Antivirus* tab in *Configuration / Content Filtering / Antivirus*.

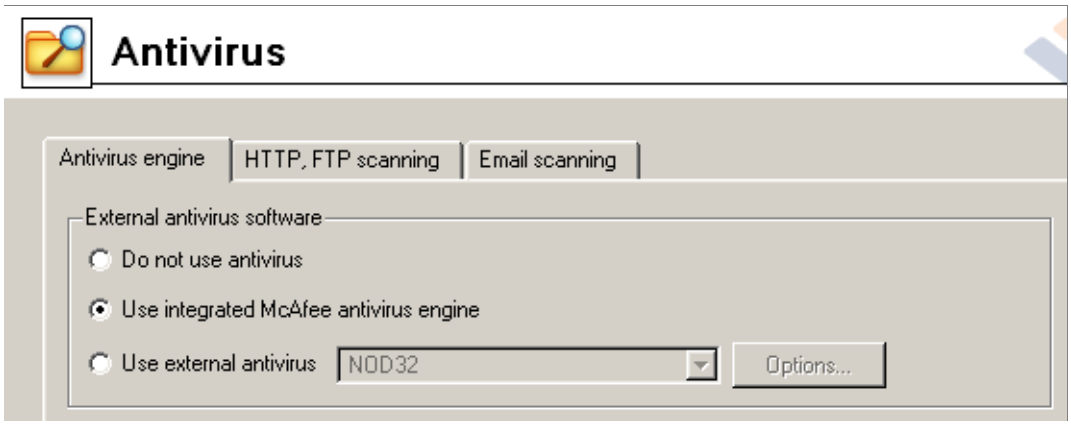
Integrated McAfee

To enable the integrated *McAfee* antivirus, select *Use integrated McAfee antivirus engine* in the *Antivirus* tab. This option is not available unless the license key for *WinRoute* includes a license for the *McAfee* antivirus or in trial versions. For detailed information on the license policy, refer to chapter 13.

Use the *Integrated antivirus engine* section in the *Antivirus* tab to set update parameters for *McAfee*.

Check for update every ... hours Time interval of checks for new updates of the virus database and the antivirus engine (in hours). If any new update is available, it will be downloaded automatically by *WinRoute*.





If the update attempt fails (i.e. the server is not available), detailed information about the attempt will be logged into the *Error* log (refer to chapter 16.8).

Each download (update) attempt sets the *Last update check performed* value to zero.

Current virus database is ... old Information regarding the age of the current database.

Note: If the value is too high, this may indicate that updates of the database have failed several times. In such cases, we recommend you to perform a manual update check by the *Update now* button and view the *Error* log.

Last update check performed Time that has passed since the last update check.

Virus database version Database version that is currently used.

Scanning engine version *McAfee* scanning engine version used by *WinRoute*.

Update now Use this button for immediate update of the virus database and of the scanning engine.

After you run the update check using the *Update now...* button, an informational window displaying the update process will be opened. You can use the *OK* button to close it — it is not necessary to wait until the update is finished.

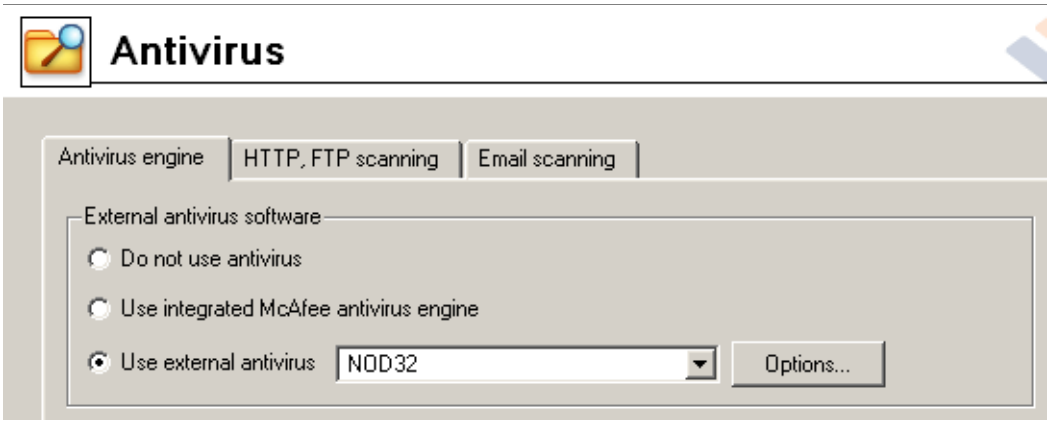
If updated successfully, the version number of the new virus database or/and the new antivirus version(s), as well as information regarding the age of the current virus database will be displayed. If the update check fails (i.e. the server is not available), an error will be reported and detailed information about the update attempt will be logged into the *Error* log.

The *Last update check performed* entry is set to zero whenever a new update check is performed.

External antivirus

For external antivirus, check the *Use external antivirus* option in the *Antivirus* tab and select an antivirus to be deployed from the combo box. This box provides all external antivirus programs supported in *WinRoute* by special *plugins*.

Warning: External antivirus must be installed before it is set, otherwise it is not available in the combo box. It is recommended to stop the *WinRoute Firewall Engine* service before an antivirus installation.



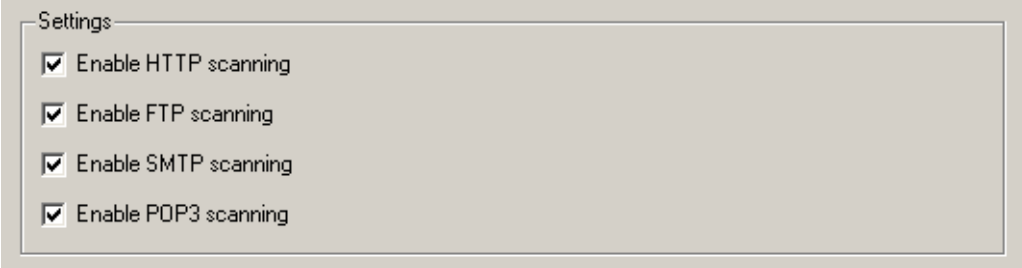
Use the *Options* button to set advanced parameters for the selected antivirus. Dialogs for individual antiviruses differ (some antivirus programs may not require any additional settings). For detailed information about installation and configuration of individual antivirus programs, refer to

`http://www.kerio.com/kwf`.

Click *Apply* to test the selected antivirus. If the test is passed successfully, the antivirus will be used from the moment on. If not, an error is reported and no antivirus will be set. Detailed information about the failure will be reported in the *Error* log (see chapter 16.8).

Antivirus settings

Check items in the *Settings* section of the *Antivirus* tab to enable antivirus check for individual protocols.



Parameters for HTTP and FTP scanning can be set in the *HTTP and FTP scanning* (refer to chapter 7.2), while SMTP and POP3 scanning can be configured in the *Email scanning* tab (see chapter 7.3).

7.2 HTTP and FTP scanning

As for HTTP and FTP traffic, objects (files) of selected types are scanned. Transmitted data are cached and checked by the antivirus program. If a virus is found, *WinRoute* does not send the last cached part of the file to the client and drops it. This means that the client receives an incomplete (damaged) file which cannot be executed so that the virus cannot be activated.

Warning:

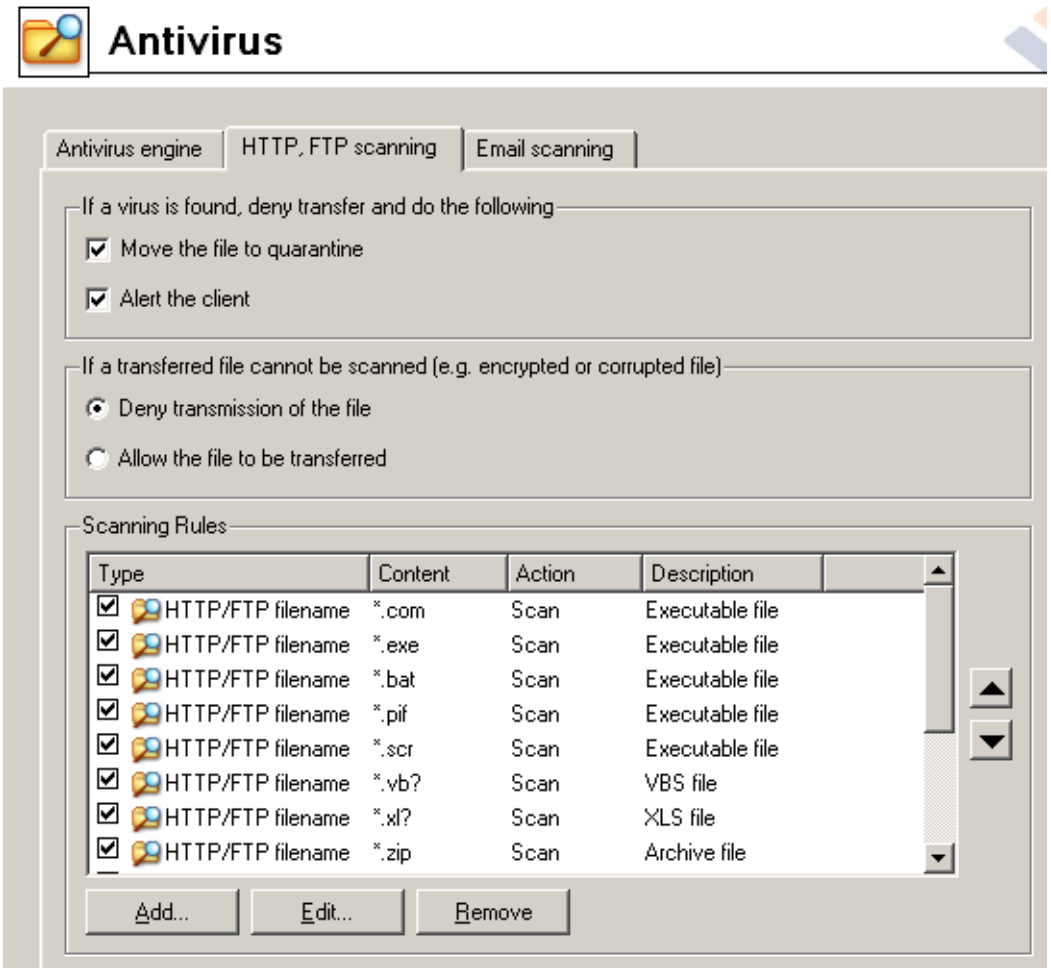
1. The purpose of the antivirus check is only to detect infected files, it is not possible to heal them!

2. If the antivirus check is disabled in HTTP and FTP filtering rules, objects and files matching corresponding rules are not checked. For details, refer to chapters 6.1 and 6.5).

To set parameters of HTTP and FTP antivirus check, open the *HTTP, FTP scanning* tab in *Configuration / Content Filtering / Antivirus*.

Use the *If a virus is found...* entry to specify actions to be taken whenever a virus is detected in a transmitted file:

- *Move the file to quarantine* — the file will be saved in a special directory on the *WinRoute* host. *WinRoute* administrators can later try to heal the file using an antivirus program and if the file is recovered successfully, the administrator can provide it to the user who attempted to download it.



The `quarantine` subdirectory under the *WinRoute* directory is used for the quarantine

(`C:\Program Files\Kerio\WinRoute Firewall\quarantine` by default). Infected files (files which are suspected of being infected) are saved into this directory with names which are generated automatically. Name of each file includes information about protocol, date, time and connection number used for the transmission.

Warning: When handling files in the `quarantine` directory, please consider carefully each action you take, otherwise a virus might be activated and the *WinRoute* host could be attacked by the virus!

- *Alert the client* — *WinRoute* alerts the user who attempted to download the file by an email message including information that a virus was detected and download was stopped for security reasons.

Alert messages can be sent under the following circumstances: the user is authenticated and connected to the firewall, a valid email address is set in a corresponding user account (see chapter 10.1) and the SMTP server used for mail sending is configured correctly (refer to chapter 11.9).

Note: Regardless of the fact whether the *Alert the client* option is used, alerts can be sent to specified addresses (e.g. addresses of network administrators) whenever a virus is detected. For details, refer to chapter 14.3.

In the *If the transferred file cannot be scanned* section, actions to be taken when the antivirus check cannot be applied to a file (e.g. the file is compressed and password-protected, damaged, etc.):

- *Deny transmission of the file* — *WinRoute* will consider these files as infected and deny their transmission.

  TIP: It is recommended to combine this option with the *Move the file to quarantine* function — the *WinRoute* administrator can extract the file and perform manual antivirus check if a user asks him/her

- *Allow the file to be transferred* — *WinRoute* will treat compressed password-protected files and damaged files as trustful (not infected).

  Generally, use of this option is not secure. However, it can be helpful for example when users attempt to transmit big volume of compressed password-protected files and the antivirus is installed on the workstations.

HTTP and FTP scanning rules

These rules specify when antivirus check will be applied. By default (if no rule is defined), all objects transmitted by HTTP and FTP are scanned.

Note: WinRoute contains a set of predefined rules for HTTP and FTP scanning. By default, all executable files as well as all *Microsoft Office* files are scanned. The *WinRoute* administrator can change the default configuration.

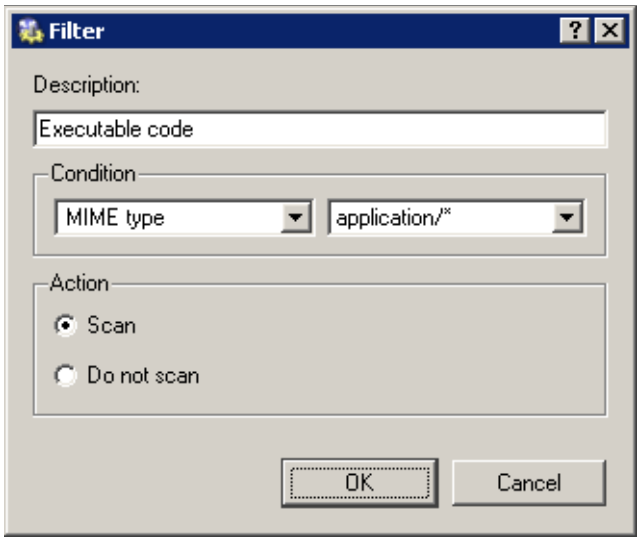
Scanning rules are ordered in a list and processed from the top. Arrow buttons on the right can be used to change the order. When a rule which matches the object is found, the appropriate action is taken and rule processing is stopped.

New rules can be created in the dialog box which is opened after clicking the *Add* button.

Description Description of the rule (for reference of the *WinRoute* administrator only)

Condition Condition of the rule:

- *HTTP/FTP filename* — this option filters out certain filenames (not entire URLs) transmitted by FTP or HTTP (e.g. `*.exe`, `*.zip`, etc.).



If only an asterisk is used for the specification, the rule will apply to any file transmitted by HTTP or FTP.

The other two conditions can be applied only to HTTP:

- *MIME type* — MIME types can be specified either by complete expressions (e.g. `image/jpeg`) or using a wildcard matching (e.g. `application/*`).
- *URL* — URL of the object (e.g. `www.kerio.com/img/logo.gif`), a string specified by a wildcard matching (e.g. `*.exe`) or a server name (e.g. `www.kerio.com`). Server names represent any URL at a corresponding server (`www.kerio.com/*`).

Note: If a MIME type or a URL is specified only by an asterisk, the rule will apply to any HTTP object.

Action Settings in this section define whether or not the object will be scanned.

If the *Do not scan* alternative is selected, antivirus control will not apply to transmission of this object.

The new rule will be added after the rule which had been selected before *Add* was clicked. You can use the arrow buttons on the right to move the rule within the list.

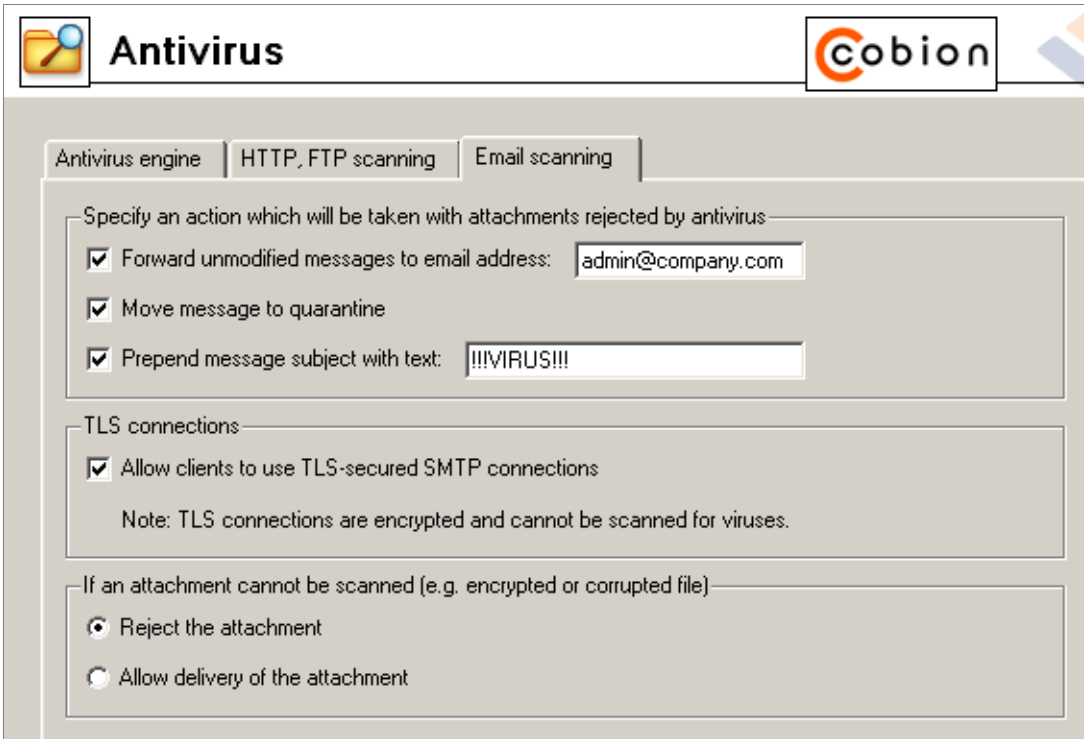
Checking the box next to the rule can be used to disable the rule. Rules can be disabled temporarily so that it is not necessary to remove rules and create identical ones later.

Note: If the object does not match with any rule, it will be scanned automatically. If only selected object types are to be scanned, a rule disabling scanning of any URL or MIME type must be added to the end of the list (the *Skip all other files* rule is predefined for this purpose).

7.3 Email scanning

SMTP and POP3 protocols scanning settings are defined through this tab. If scanning is enabled for at least one of these protocols, all attachments of transmitted messages are scanned. Through the *Email scanning* tab, actions to be taken when a virus is detected as well as advanced parameters can be set.

Warning: Antivirus control within *WinRoute* can detect and block infected attachments. Attached files cannot be healed by this control!



In the *Specify an action which will be taken with attachments...* section, the following actions can be set for messages considered by the antivirus as infected:

- *Forward unmodified messages to email address* — untrustworthy messages will be, unchanged, forwarded to a specified email address (usually to the network administrator). The recipient can then try to heal infected files and later send them to their original addresses.

  Note: For email sending, SMTP Relay Server must be set in *WinRoute* — see chapter 11.9.

- *Move message to quarantine* — untrustworthy messages will be moved to a special directory on the *WinRoute* host. The *WinRoute* administrator can try to heal infected files and later send them to their original addressees.

For the quarantine, the special `quarantine` subdirectory under the *WinRoute* directory is used

(`C:\Program Files\Kerio\WinRoute Firewall\quarantine` by default). Messages with untrustworthy attachments are saved to this directory under names which are generated automatically by *WinRoute*. Each filename includes information about protocol, date, time and the connection number used for transmission of the message.

Note: Regardless of what action is set to be taken, the attachment is always removed and a warning message is attached instead.

Use the *TLS connections* section to set firewall behavior for cases where both mail client and the server support TLS-secured SMTP traffic.

In case that TLS protocol is used, encrypted connection is established first. Then, client and server agree on switching to the secure mode (encrypted connection). If the client or the server does not support TLS, encrypted connection is not used and the traffic is performed in a non-secured way.

If the connection is encrypted, firewall can analyze it and perform antivirus check for transmitted messages. *WinRoute* administrator can select one of the following alternatives:

- Enable TLS. This alternative is suitable for such cases where protection from wiretapping is prior to antivirus check of email.

  TIP: In such cases, it is recommended to install an antivirus engine at individual hosts that would perform local antivirus check.

- Disable TLS. Secure mode will not be available. Clients will automatically assume that the server does not support TLS and messages will be transmitted through an unencrypted connection. Firewall will perform antivirus check for all transmitted mail.

The *If an attachment cannot be scanned* section defines actions to be taken if one or multiple files attached to a message cannot be scanned for any reason (e.g. password-protected archives, damaged files, etc.):

- *Reject the attachment* — *WinRoute* reacts in the same way as when a virus was detected (including all the actions described above).

- *Allow delivery of the attachment* — *WinRoute* behaves as if password-protected or damaged files were not infected.

Note: Generally, this option is not secure. However, it can be helpful for example when users attempt to transmit big volume of compressed password-protected files and the antivirus is installed on the workstations.

Chapter 8

Web Interface and User Authentication

WinRoute contains a special Web server that can be used for several purposes, such as an interface for user connections, dial-up control or cache management. This Web server is available over SSL or using standard HTTP with no encryption (both versions include identical pages).

Refer to the list below for URLs of individual pages (`'server'` refers to the name or IP of the *WinRoute* host, `4080` represents a standard HTTP interface port).

- the main page (*Index*) — includes only links to the pages listed below

  `https://server:4080/`

- user authentication at the firewall (login and logout page)

  `http://server:4080/fw/login`

  `http://server:4080/fw/logout`

- modifications of user configuration (password, global limitations for accessing WWW pages, etc.)

  `http://server:4080/fw/pref`

- viewing user statistics (i.e. IP address, login time, size of the data transmitted, number of filtered objects, etc.)

  `http://server:4080/fw/stat`

- dialing and disconnecting dial-ups

  `http://server:4080/fw/dial`

- viewing statistics of HTTP cache with functions for deleting and searching for saved objects

  `http://server:4080/fw/cache`

- viewing HTTP rules (see chapter 6.1) not related to the user or the host that is used to connect to the Web interface

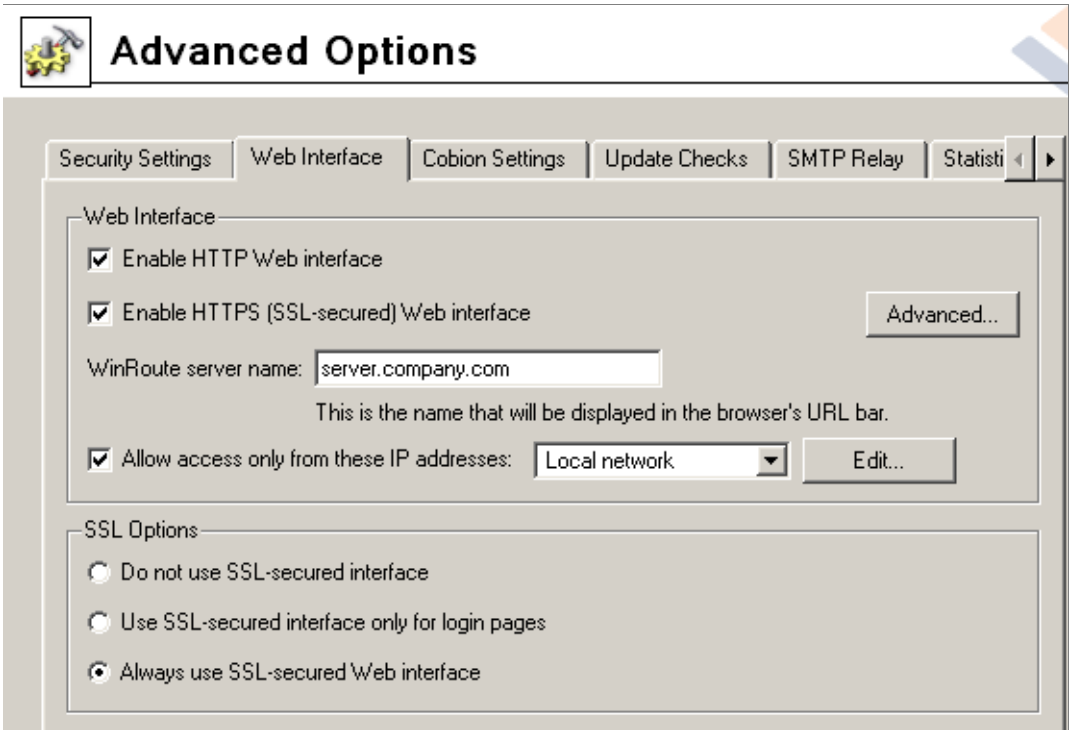
  `http://server:4080/fw/http_restr`

To use the encrypted version specify the HTTPS protocol and number of the port that the encrypted Web interface is running on (default is 4081) — e.g.

```
https://server:4081/fw/login
```

8.1 Web Interface Parameters Configuration

To define basic *WinRoute* Web interface parameters go to the *Web Interface* folder in *Configuration / Advanced Options*.



Enable Web Interface (HTTP) This option enables unencrypted (HTTP) version of the Web interface (the port 4080 is used by default for this interface).

Enable Web Interface over SSL (HTTPS) This option enables encrypted (HTTPS) version of the Web interface (the port 4081 is used by default for this interface).

WinRoute server name Server DNS name that will be used for purposes of the Web interface (e.g. `server.company.com`). The name need not be necessarily identical with the host name, however, there must exist an appropriate entry in DNS for proper name resolution.

Note: If all clients accessing the Web Interface use the *DNS Forwarder* in *WinRoute* as a DNS server, there is no need to add the server name to DNS. The name is already known and combined with the name of the local domain — see chapter 4.3).

Allow access only from these IP addresses Select IP addresses which will always be allowed to connect to the Web interface (usually hosts in the local network). You can also click the *Edit* button to edit a selected group of IP addresses or to create a new IP group (details in chapter 9.1).

Note: Access restrictions are applied to both unencrypted and encrypted versions of the Web interface.

In *SSL Options* you can set pages to which users will be redirected if the firewall requires user authentication (see chapter 10.1).

- *Do not use SSL-secured interface* — users will be redirected to the unencrypted authentication page.

  Warning: This option is not very secure (i.e. user passwords can be tapped). However, it can be quite safely used in a local network behind a firewall. It is also necessary to use this option if a valid SSL certificate is not available, or in case that any other technical problems arise.

- *Use SSL-secured interface only for login pages* — users will be automatically redirected to the secured authentication page. Other pages of the Web interface (e.g. denial information, error alerts, etc.) will not be encrypted.

- *Always use SSL-secured Web interface* — encrypted version will be used for all pages of the Web interface.

Web Interface: Advanced options

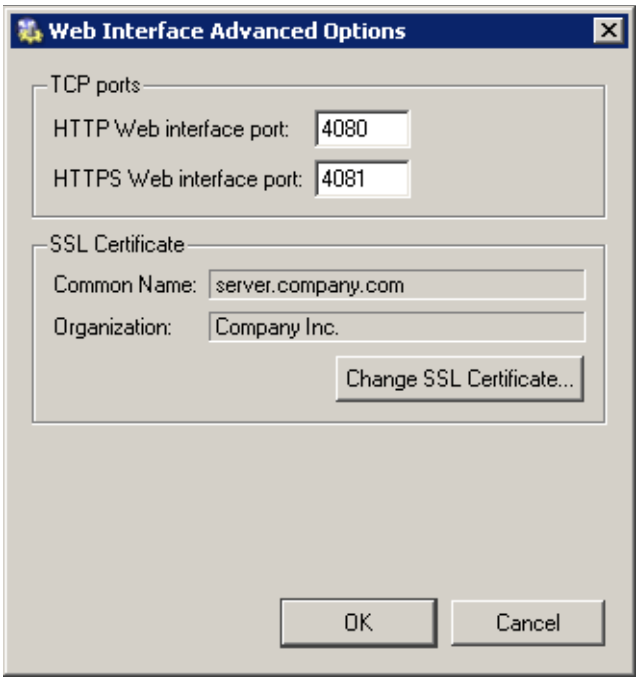
Advanced parameters for the Web interface can be set upon clicking on the *Advanced* button.

TCP ports Use this section to set ports for unencrypted and encrypted versions of the Web interface (default ports are `4080` for the unencrypted and `4081` for the encrypted version of the Web interface).

TIP: If no WWW server is running on the *WinRoute* host, standard ports (i.e. `80` for HTTP and `443` for HTTPS) can be used for the Web interface. In such cases, the port number is not necessarily required in URLs for pages of the Web interfaces.

Warning: If any of the entries are specified by a port which is already used by another service or application, and the

Apply button (in *Configuration / Advanced Options*) is clicked, *WinRoute* will accept this port, however, the Web interface will not run at the port and an error in the following format will be reported in the *Error* log (see chapter 16.8):




```
Socket error:  Unable to bind socket for service to port 80.
(5002) Failed to start service "WebAdmin"
bound to address 192.168.1.10.
```

If you are not sure that specified ports are free, check the *Error* log immediately after clicking *Apply* to find out whether the corresponding error has been logged.

SSL certificate Basic information (server name, name of the organization by which the certificate was issued) about currently used SSL certificate are provided in this section. Click the *Change SSL certificate* button to create a new certificate or to import a certificate issued by a public certification authority.

Server SSL certificate

The principle of an encrypted *WinRoute* Web interface is based on the fact that all communication between the client and server is encrypted to protect it from wiretapping and misuse of the transmitted data. The SSL protocol uses an asymmetric encryption first to facilitate exchange of the symmetric encryption key which will be later used to encrypt the transmitted data.

Two keys are used for the asymmetric encryption — public to encrypt and private to decipher. The public (encrypting) key is available to all users that intend to connect to the server, whereas the private (deciphering) key is available for the server only and it

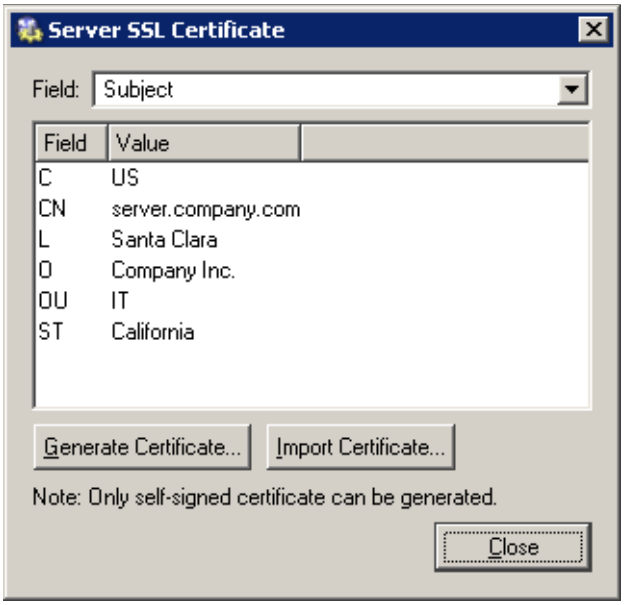
must be kept close. The client also needs to verify the server's identity. For this purpose there is a so called certificate. The certificate contains the public key of the server, server name, information about validity and other data. To ensure authenticity of the certificate, it must be verified and subscribed by the third party, or certificate authority.

The communication between the client and server is as follows: the client generates a symmetrical key and encrypts it with the public key of the server (gained from the server certificate). The server deciphers it with the unique private key. Therefore, only these two parties know the symmetrical key.

Generate or Import Certificate

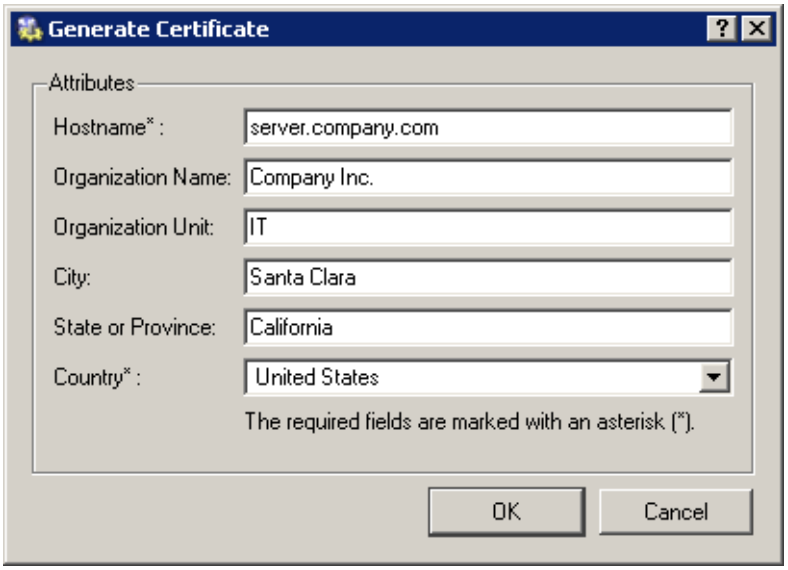
WinRoute provides a sample certificate for testing. You will find it in the `server.crt` file under the `sslcert` subdirectory where *WinRoute* is installed. The other file (`server.key`) includes the private key of the server. This certificate is identical in each *WinRoute* application. This means that only encrypted services will function, but practically no security is ensured (everyone knows the private key — thus any user is allowed to decipher public communication).

Click on the *Change SSL certificate* (in the dialog for advanced settings for the Web interface) to view the dialog with the current server certificate. By selecting the *Field* (certificate entry) option you can view information either about the certificate issuer (*Issuer*) or about the subject (*Subject*) represented by your server.



To get your own unique certificate that you will use to authenticate identity of your server, use one of the two methods described below.

To create your own (self-signed) certificate click on the *Generate certificate* button in the dialog that displays the current server's certificate. Insert required data about the server and your company into the dialog entries. Only entries marked with an asterisk (*) are required.



Click on the *OK* button to view the *Server SSL certificate* dialog. The certificate will be started automatically (you will not need to restart your operating system).

A new (self-signed) certificate is unique. It is created by your company, addressed to your company and based on the name of your server. Unlike the testing version of the certificate, this certificate ensures your clients security, as only you know the private key and the identity of your server is guaranteed by the certificate. In their browsers, clients will be informed that the certificate authority is not reliable; however, they will install it into the browser as they trust the owner of this certificate. This ensures secure communication and there will be no more warnings displayed as the certificate has all the necessary features.

The other option is to get a signed certificate from a public certificate authority (e.g. Verisign, Thawte, SecureSign, SecureNet, Microsoft Authenticode, etc.). The certification process is quite complex and requires special technical knowledge. For detailed instructions contact Kerio technical support.

Web Interface Language Preferences

WinRoute's Web Interface is available in various languages. The language is set automatically according to each users' preferences defined in the Web browser (this function is available in most browsers). English will be used if no preferred language is available .

Individual language versions are saved in definition files in the `weblang` subdirectory under the directory where *WinRoute* is installed. Each language is represented by the two following files: `xx.def` and `xx.res`. The `xx` string stands for a standard language abbreviation that consists of two characters (i.e. `en` stands for English, etc.). The first rows of `xx.def` include appropriate language abbreviations (it is equal to the abbreviation contained in the file name). The second row contains coding used for the appropriate language (i.e. `ISO-8859-1` is used for English). This coding must be used for both language files.

WinRoute administrators can easily modify texts of the Web Interface pages or create new language versions.

Note: Changes in the `xx.def` file will be applied after restarting the *WinRoute Firewall Engine*.

8.2 Firewall User Authentication

WinRoute allows administrators to monitor connections (packet, connection, Web pages or FTP objects and command filtering) related to each user. The username in each filtering rule represents the IP address of the host(s) from which the user is connected.

In addition to authentication based access limitations, user login can be used to effectively monitor activity using logs (see chapter 16), and status (see chapter 14.2) and hosts and users (see chapter 14.1). If there is no user connected from a certain host, only the IP address of the host will be displayed in the logs and statistics.

Users can connect:

- manually — in the browser user will open page

  ```
  http://server:4080/fw/login
  ```

  (the name of the server and the port number are examples only — see chapter 8)

- redirection — by accessing any Web site (unless access to this page is explicitly allowed to unauthenticated users — see chapter 6.1)

- using NTLM— if *Microsoft Internet Explorer* is used and the user is authenticated in a Windows NT domain or Active Directory, the user can be authenticated automatically (the login page will not be displayed). For details see the *User Authentication Options* section.

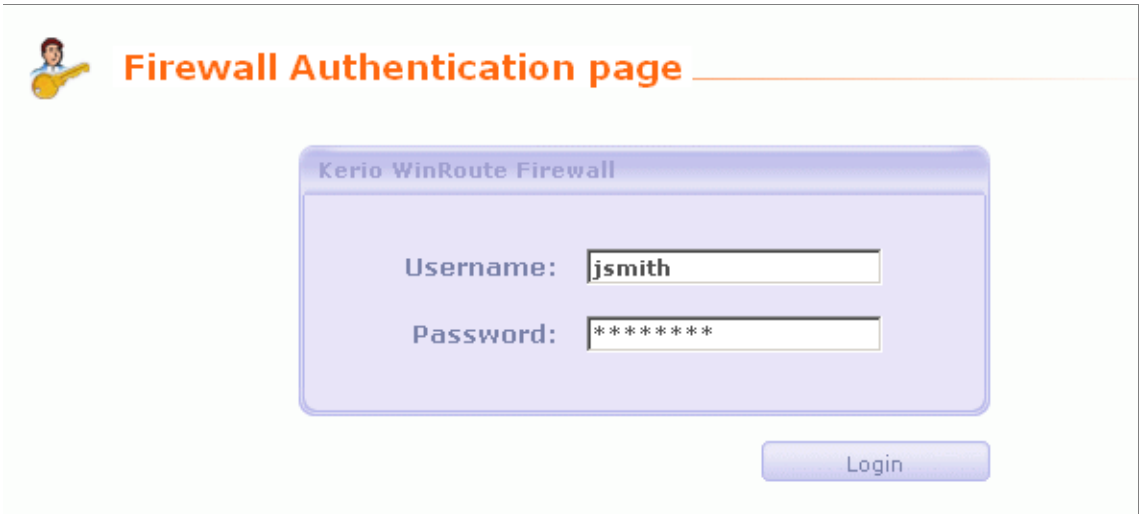
Login by re-direction is performed in the following way: user enters URL pages that he/she intends to open in the browser. *WinRoute* detects whether the user has already authenticated. If not, *WinRoute* will re-direct the user to the login page automatically.

After a successful login, the user is automatically re-directed to the requested page or to the page including the information where the access was denied.

Note: If the *Do not use SSL-secured interface* option is enabled in the parameters for the Web interface (see chapter 8.1), users are re-directed to the encrypted login page automatically. If not, users are re-directed to the unencrypted login page.

Login page

Authentication page through which users login to the firewall against username and password.



If the user is re-directed to the page automatically (after inserting the URL of a page for which the firewall authentication is required), he/she will be re-directed to the formerly requested site after successful login attempt. Otherwise, a reference page will be opened from which users can open other pages of the Web interface (e.g. user preferences, dial-up control, cache management, etc.). For detailed information, refer to the following chapters.

8.3 User Preferences

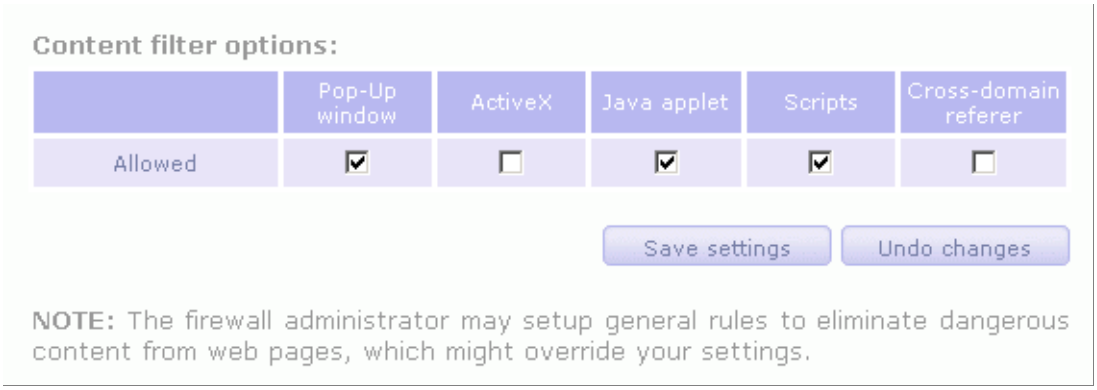
If a user has opened the user menu (by ticking the option at the login page), the user is automatically re-directed to the user menu page. This page provides links to (apart of others):

- formerly requested *URL* page — if the login page has not been displayed automatically, this item will be empty

- user preferences page (*User Preferences*)
- user statistics page (*Statistics*)

User Preferences

The first part of the page enables the administrator to permit or deny certain features of WWW pages.



Content filter options If the checkbox under a filter is enabled, this feature will be available (it will not be blocked by the firewall).

If a certain feature is disabled in the parameters of a user account (see chapter 10.1), a corresponding item within this page is inactive (user cannot change settings of the item). Users are only allowed to make the settings more restrictive. In other words, users cannot enable an HTML item denied by the administrators for themselves.

- *Pop-Up Window* — automatic opening of new windows in the browser (usually advertisements)

  This option will block the `window.open()` method in scripts

- *ActiveX* — Microsoft ActiveX features (this technology enables, for example, execution of applications at client hosts)

  This option blocks `<object>` and `<embed>` HTML tags

- *Java applet* `<applet>` HTML tag blocking
- *Scripts* — `<script>` HTML tag blocking (commands of JavaScript, VBScript, etc.)
- *Cross-domain referrer* — blocking of the `Referrer` items in HTTP headers. This item includes pages that have been viewed prior to the current page. The *Cross-domain referrer* option blocks the `Referrer` item in case this item does not match the required server name.

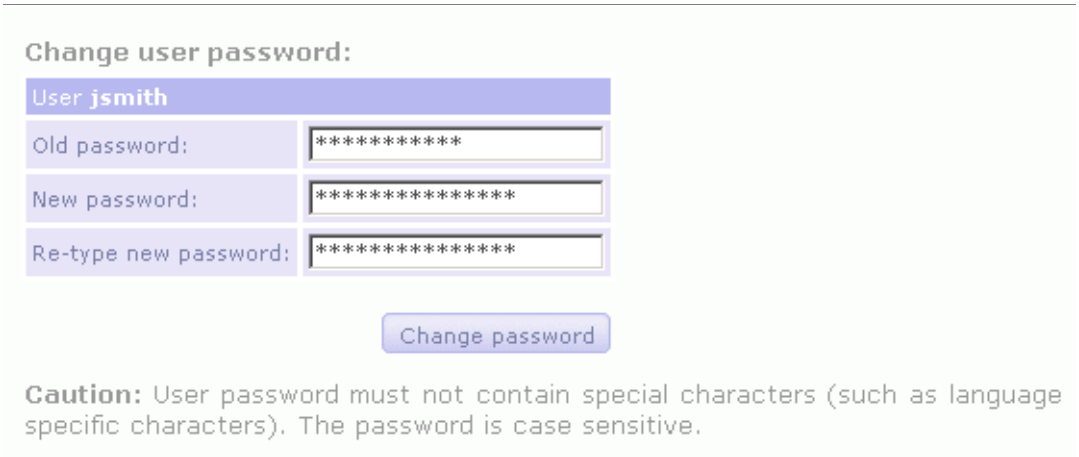
Cross-domain referrer blocking protects users' privacy (the `Referrer` item can be monitored to determine which pages are opened by a user).

Save settings To save and activate settings, click on this button.

Undo changes With this button you can restore your former settings.

Note: Changes in configuration of content filtering in a user account will take effect upon a next login of the user.

User password can be modified at the bottom part of the page:



To change a password, enter the current user password, new password, and the new password confirmation into the appropriate text fields. Save the new password with the *Change password* button.

Warning: Passwords can be changed only if the user is configured in the *WinRoute* internal database (see chapter 10.1). If another authentication method used, the *WinRoute Firewall Engine* will not be allowed to change the password. Then, the *Change user password* section is not even displayed in the page of user preferences.

8.4 User statistics

The following data will be displayed at the *User statistics* page:

- *Login information* — username, IP address that the user is connected from, login duration and method of login (*SSL* — encrypted login page (*SSL* — encrypted login

page; *Plaintext* — unencrypted login page; *NTLM* — secure authentication in Windows NT or Windows 2000, *Proxy* — authentication at *WinRoute's* proxy server);

- *Traffic Statistics* — size of outgoing and incoming data (in bytes) and number of sent HTTP requests;
- *Content filter statistics* — number of filtered objects of all individual types (see above).

All data are recorded and measured after the first login of a user. All statistics are deleted after the user logs out or if the *WinRoute Firewall Engine* is restarted.

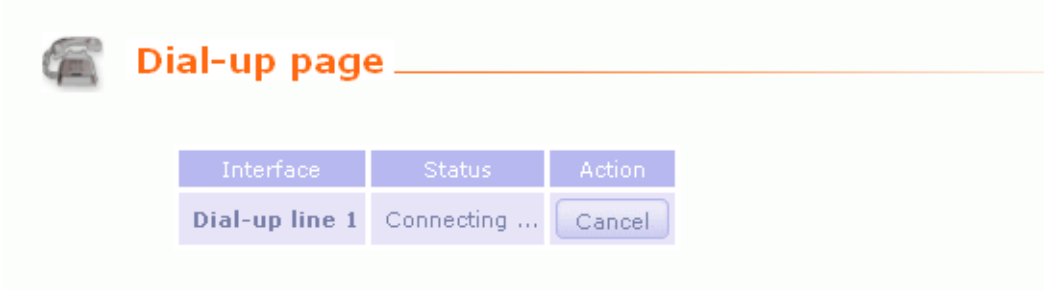
8.5 Web Policy Viewing

Click on the *Web policy* link at any page of the *WinRoute* Web Interface to view current rules and limitations of access to Web pages. The policy is related to the appropriate user and host. If no user is connected, limitation settings for the IP address of the host that is used to connect to the Web Interface will be displayed.

To learn more details about rules for accessing Web pages refer to chapter 6.1.

8.6 Dial-up

All RAS lines defined in *WinRoute* are listed at the *Dial-up* page (see chapter 4.1). Each dial-up provides the following information:



- Dial-up Status — *Disconnected*, *Connecting* (the line is being dialed), *Connected*, *Disconnecting* (the line is being disconnected).
- command — *Dial* or *Hang-up* (line status).

Note: The *Dial-up* page is automatically refreshed in regular time intervals. This ensures that only the current dial-up status will be displayed.

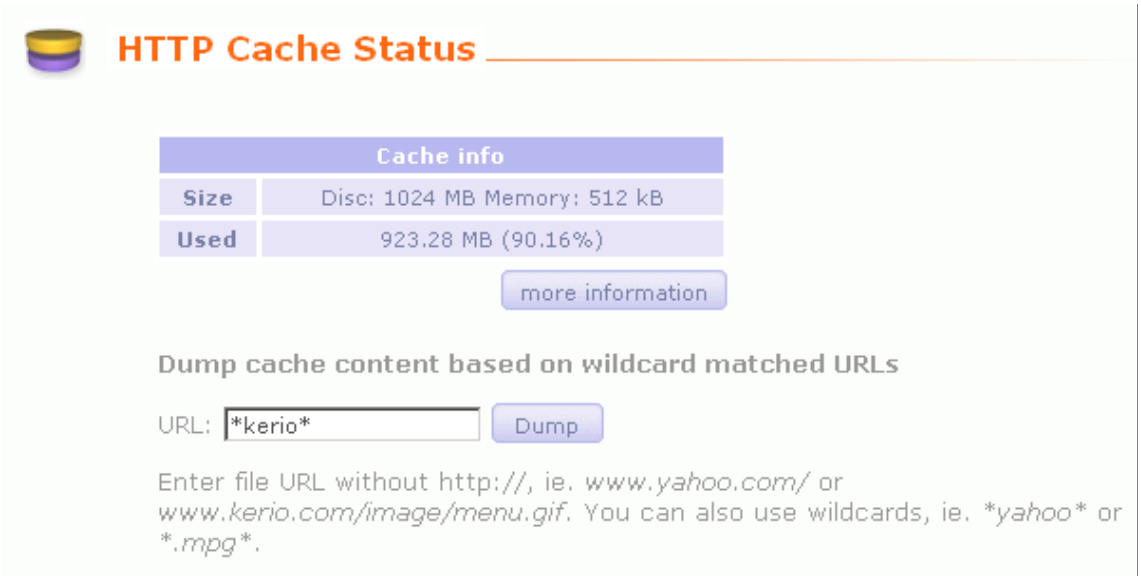
This page can be viewed by any user (login is not required); however, if the *Dial* or *Hang up* buttons are clicked, authentication is verified. Users that intend to control dial-up

lines need special rights (the *User can dial* option in the user account configurations section — see chapter 10.1).

8.7 HTTP Cache Administration

To view and/or remove objects contained in the HTTP cache go to the *Cache* tab. Open the *Cache content* page of the *WinRoute* Web Interface to view and/or delete objects in the HTTP cache. Only users that have rights to read the *WinRoute* configuration can open this page (either by inserting the URL directly or using the *Cache* link at the bottom of any Web interface page) (if the user is not authenticated yet, automatic redirection to the authentication page will be performed). To remove objects from the cache, full administration rights are required. To read detailed information about user access rights see chapter 10.1.

Note: For information on defining HTTP parameters see chapter 4.6.



Cache parameters

Click on the *more information* link to view tables including the following features:

- Number of saved files, total size of all files and average file size
- File size distribution table (by 1 KB)

- Number of objects found or not found in the cache
- Information on cache maintenance (number of upkeeps, time since the last upkeep and its duration)

Searching in cache

Use the *URL:* text field with the *Dump* button to search for objects matching the appropriate URL. Located objects are displayed in a table (up to 100 entries). Each entry contains an object's size, time-to-live (TTL) in hours and the *Delete* button to remove the object from the cache if needed.

All objects matching the appropriate URL can be removed from the cache using the *Delete all* button (not only the entries displayed in the table, if more than 100 entries match the specified URL).

TIP: All entries can be removed from the cache by inserting only an asterisk (*) into the *URL:* text field and using the *Delete all* button.

Chapter 9

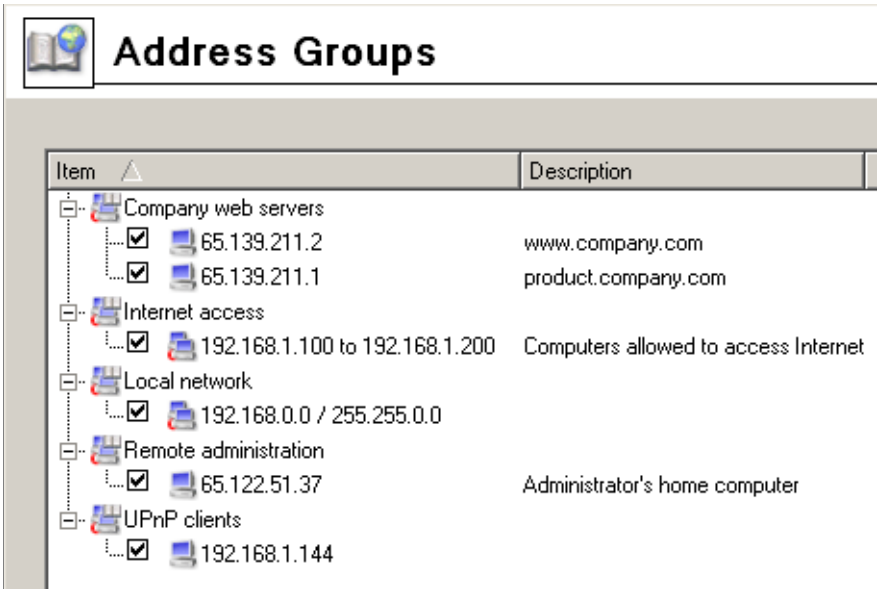
Definitions

9.1 Address Groups

Address groups allow the administrator to easily define restricted access to certain services, such as remote administration. Each group will be given a name during configuration. Groups can include combinations of IP addresses, IP ranges, IP subnets or even other groups.

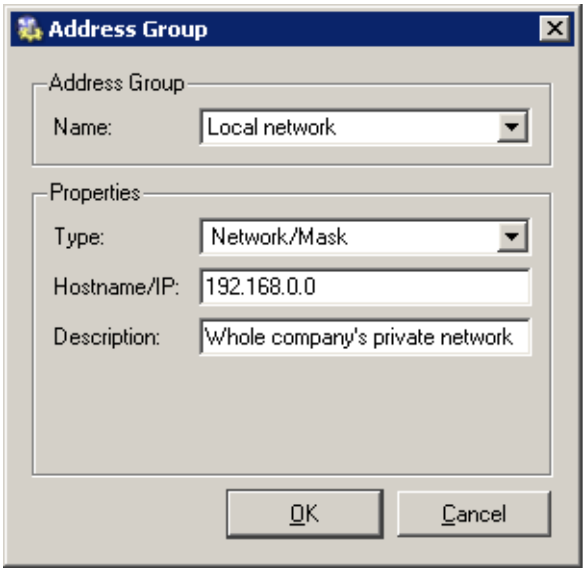
Adding or Editing Address Groups

You can define the Address groups in *Configuration / Definitions / Address Groups*.



By clicking on the *Add* button you can add a new group or an item to a group. The *Edit* button opens a dialog for editing and the *Remove* button removes the group or the item selected.

By clicking on the *Add* button a dialog for adding a new address group is displayed.



Name Name of the group. Add a new name to create a new group. Insert the group name to add a new item to an existent group.

Type Type of the new item:

- *Host* (IP address or DNS name of a particular host)
- *Network / Mask* (subnet with a corresponding mask)
- *Network / Range* (IP range)
- *Address group* (another group of IP addresses — groups can be cascaded)

IP Address and Mask Parameters of the new item (related to the selected type).

Description Description of the address group. Comments for the administrator.

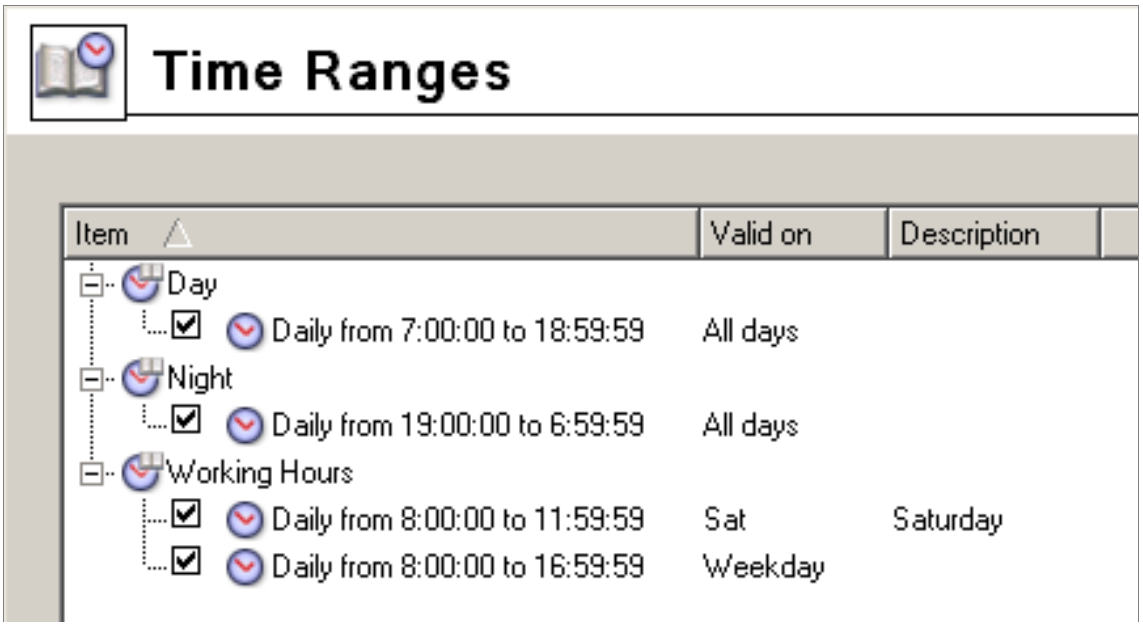
Note: Each IP group must include at least one item. Groups with no item will be removed automatically.

9.2 Time Ranges

Time ranges in *WinRoute* are closely related to traffic policy rules (see chapter 5). *WinRoute* allows the administrator to set a time period where each rule will be applied. These time ranges are actually groups that can consist of any number of various intervals and single actions.

Using time ranges you can also set dial-up parameters — see chapter 4.1.

To define time ranges go to *Configuration / Definitions / Time Ranges*.



Time range types

Three types of time intervals can be used to define time ranges:

Absolute The time interval is defined with the initial and expiration date and it is not repeated

Weekly This interval is repeated weekly (according to the day schedule)

Daily It is repeated daily (according to the hour schedule)

Time Ranges Definition

You can create, edit or remove time ranges in *Configurations / Definitions / Time Ranges*.

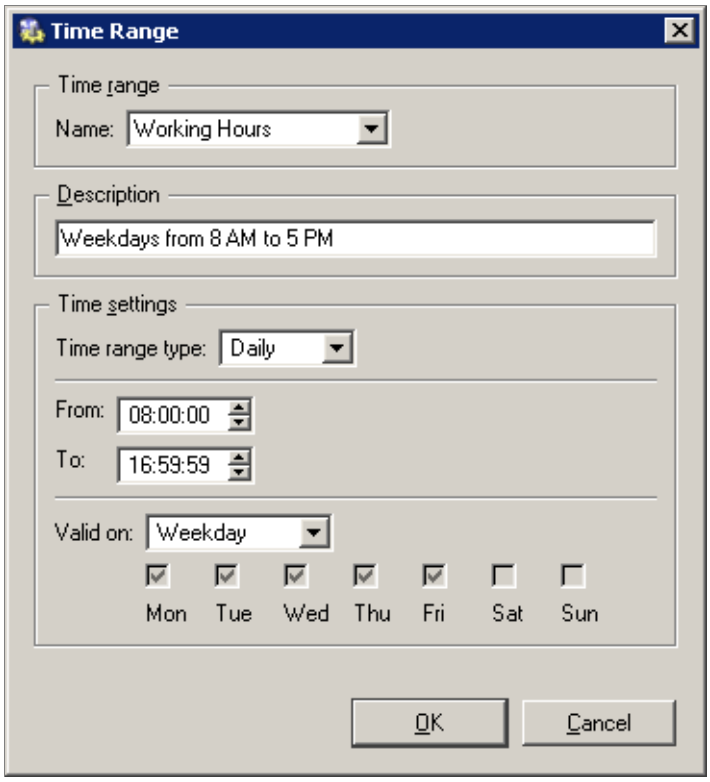
Click on the *Add* button to open a dialog for time ranges definition:

Name Unique name (identification) of a time range. Insert a new name to create a new time range. Insert the name of an existent time range to add a new item to this range.

Description Time ranges description, for the administrator only

Time range type Time range type: *Daily*, *Weekly* or *Absolute*. The last type refers to the user defined initial and terminal date.

From / To This function helps to define the beginning and end of a time interval. Beginning and end hours, days or dates can be defined according to the selected time range type



Valid at days Defines days when the interval will be valid. You can either select particular weekdays (*Selected days*) or use one of the predefined options (*All Days*, *Weekday* — from Monday to Friday, *Weekend* — Saturday and Sunday).

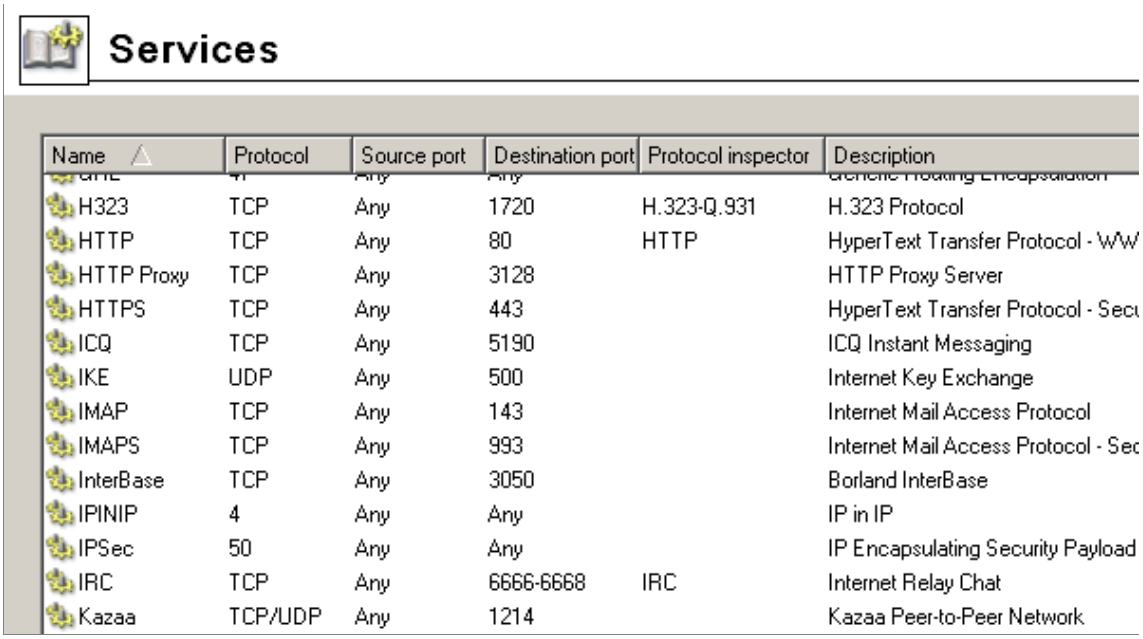
Notes:

1. each time range must contain at least one item. Time ranges with no item will be removed automatically.
2. It is not possible to include one time range into another.

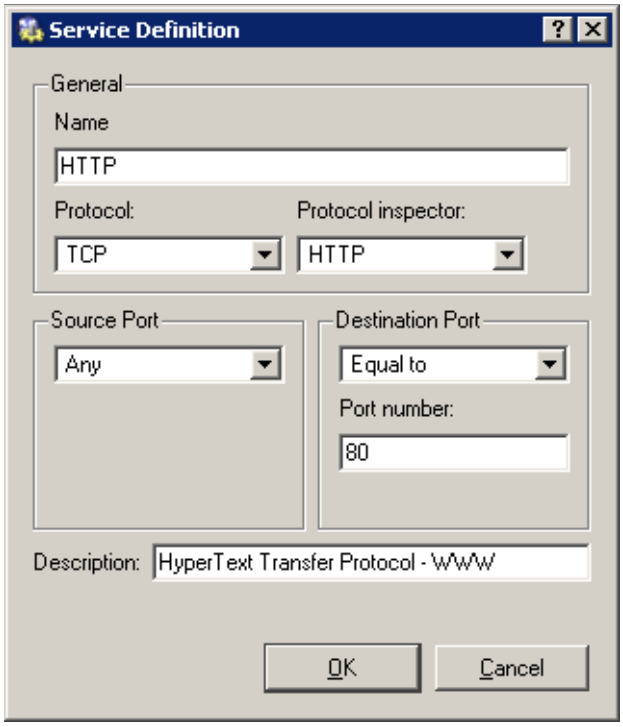
9.3 Services

WinRoute services enable the administrator to define communication rules easily (by permitting or denying access to the Internet from the local network or by allowing access to the local network from the Internet). Services are defined by a communication protocol and by a port number (e.g. the *HTTP* service uses the TCP protocol with the port number 80). You can also match so-called protocol inspector with certain service types (for details see below).

Services can be defined in *Configurations / Definitions / Services*. Some standard services, such as HTTP, FTP, DNS etc., are already predefined in the default *WinRoute* installation.



Clicking on the *Add* or the *Edit* button will open a dialog for service definition.

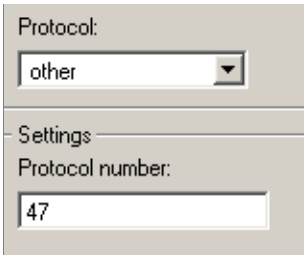


Name Service identification within *WinRoute*. It is strongly recommended to use a concise name to keep the program easy to follow.

Protocol The communication protocol used by the service.

Most standard services uses the *TCP* or the *UDP* protocol, or both when they can be defined as one service with the *TCP/UDP* option.

The option *other* enables the administrator to specify a protocol using the number contained in its IP packet header. Any protocol carried in IP (e.g. GRE — protocol number is 47) can be defined this way.



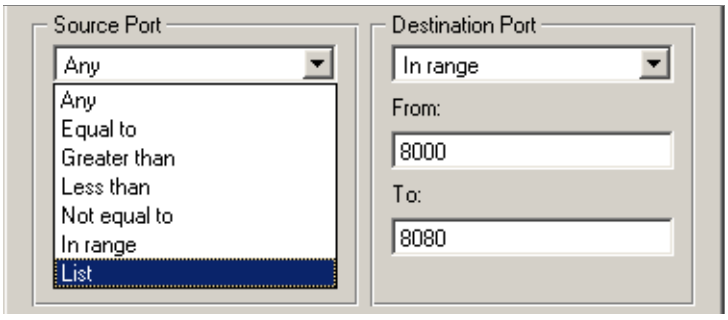
Protocol Inspector *WinRoute* protocol inspector (see below) that will be used for this service.

Warning: Each inspector should be used for the appropriate service only.

Source Port and Destination Port If the TCP or UDP communication protocol is used, the service is defined with its port number. In case of standard client-server types, a server is listening for connections on a particular port (the number relates to the service), whereas clients do not know their port in advance (port are assigned to clients during connection attempts). This means that source ports are usually not specified, while destination ports are usually known in case of standard services.

Note: Specification of the source port may be important, for example during the definition of communication filter rules. For more information go to chapter 5.2.

Source and destination ports can be specified as:



- *Any* — all the ports available (1-65535)
- *Equal to* —a particular port (e.g.80)

- *Greater than*, *Less than* — all ports with a number that is either greater or less than the number defined
- *Not equal to* — all ports that are not equal to the one defined
- *In range* — all ports that fit to the range defined (including the initial and the terminal ones)
- *List* — list of the ports divided by comas (e.g. `80,8000,8080`)

Description Comments for the service defined. It is strongly recommended describing each definition, especially with non-standard services so that there will be minimum confusion when referring to the service at a later time.

Protocol Inspectors

WinRoute includes special plug-ins that monitor all traffic using application protocols, such as HTTP, FTP or others. The modules can be used to modify (filter) the communication or adapt the firewall's behavior according to the protocol type. Benefits of protocol inspectors can be better understood through the two following examples:

1. *HTTP protocol inspector* monitors traffic between clients (browsers) and Web servers. It can be used to block connections to particular pages or downloads of particular objects (i.e. images, pop-ups, etc.).

2. With active FTP, the server opens a data connection to the client. Under certain conditions this connection type cannot be made through firewalls, therefore FTP can only be used in passive mode. The *FTP protocol inspector* distinguishes that the FTP is active, opens the appropriate port and redirects the connection to the appropriate client in the local network. Due to this fact, users in the local network are not limited by the firewall and they can use both FTP modes (active/passive).

A protocol inspector is active if it is included in a service that is used in a traffic rule. If a rule for any service is defined, all *WinRoute's* protocol inspectors that meet this rule will be activated automatically.

Notes:

1. Protocol inspectors recognize application protocols through transport layer protocols (TCP or UDP) and the number of the port that is used by the appropriate service. If a service is running at a non-standard port (i.e. *HTTP* on port number `8080`), the

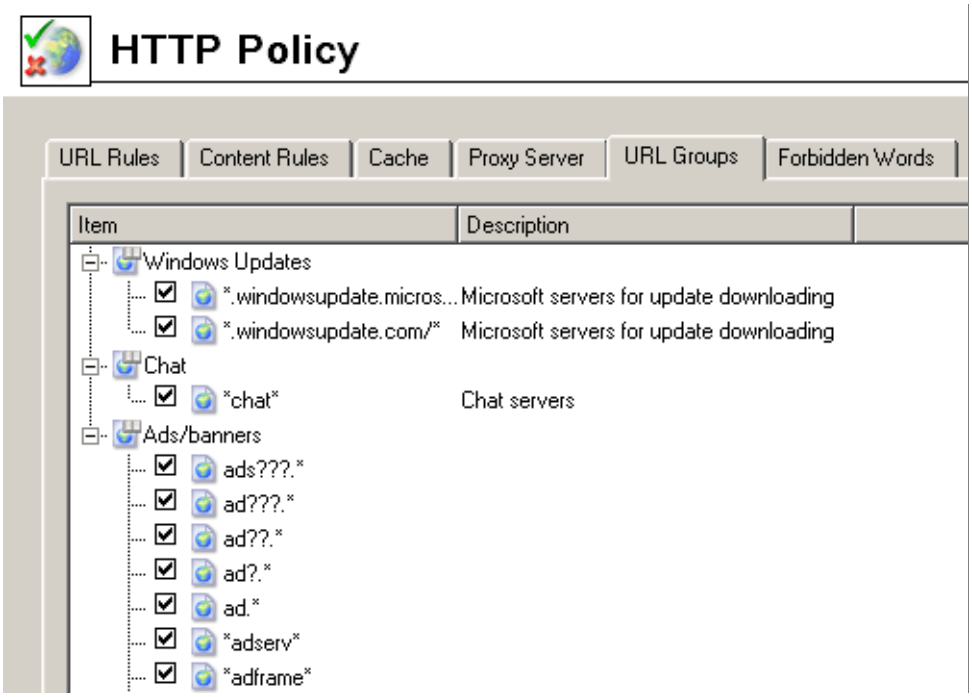
protocol inspector will not be used. In this case you could create a custom service for port 8080 which uses the *HTTP* protocol inspector.

2. Under certain circumstances, appliance of a protocol inspector is not desirable. Therefore, it is possible to disable a corresponding inspector temporarily. For details, refer to chapter 17.2.

9.4 URL Groups

URL Groups enable the administrator to define HTTP rules easily (see chapter 6.1). For example, to disable access to a group of Web pages, you can simply define a URL group and assign permissions to the URL group, rather than defining permissions to each individual URL rule.

URL groups can be defined in the *Configuration / Definitions / URL Groups* section.

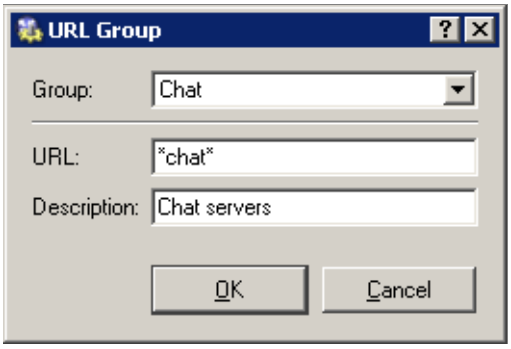


Tick or untick matching fields next to each URL to enable or disable the appropriate URL. This way you can deactivate URLs with no need to remove them and to define them again.

Note: The default *WinRoute* installation already includes a predefined URL group:

- *Ads/Banners* common URLs of pages that contain advertisements, banners, etc.

Click on the *Add* button to display a dialog where a new group can be created or a new URL can be added to existing groups.



Group Name of the group to which the URL will be added. This option enables the administrator to:

- select a group to which the URL will be added
- add a name to create a new group to which the URL will be included.

URL The URL that will be added to the group.

- full address of a server, a document or a web page without protocol specification (`http://`)
- use substrings with the special `*` and `?` characters. An asterisk stands for any number of characters, a question-mark represents one character.

Examples:

- `www.kerio.cz/index.html` — a particular page
- `www.*` — all URL addresses starting with `www.` `www.*`
- `www.kerio.com` — all URLs at the `www.kerio.com` server (this string is equal to the `www.kerio.com/*` string)
- `*sex*` — all URL addresses containing the `sex` string
- `*sex??.cz*` — all URL addresses containing such strings as `sexxx.cz`, `sex99.cz`, etc.

Description The URL description (comments and notes for the administrator).

Chapter 10

User Accounts and Groups

10.1 User Accounts

User accounts in *WinRoute* improve control of user access to the Internet from the local network. User accounts can be also used to access the *WinRoute* administration using the *Kerio Administration Console*. A basic administrator account is created during the *WinRoute* installation process. This account has full rights for *WinRoute* administration. It can be removed if there is at least one other account with full administration rights.

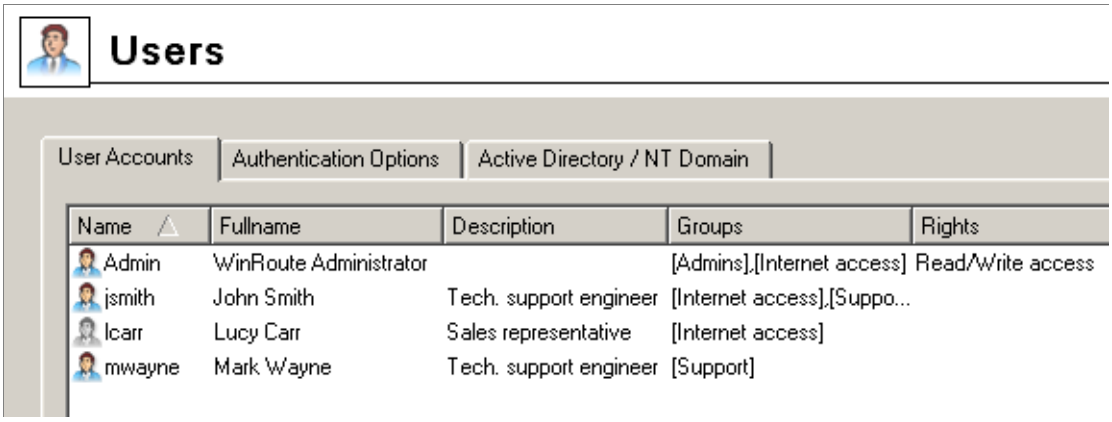
Note: If you have lost access to the *WinRoute* administration contact Kerio technical support.

1.

2.

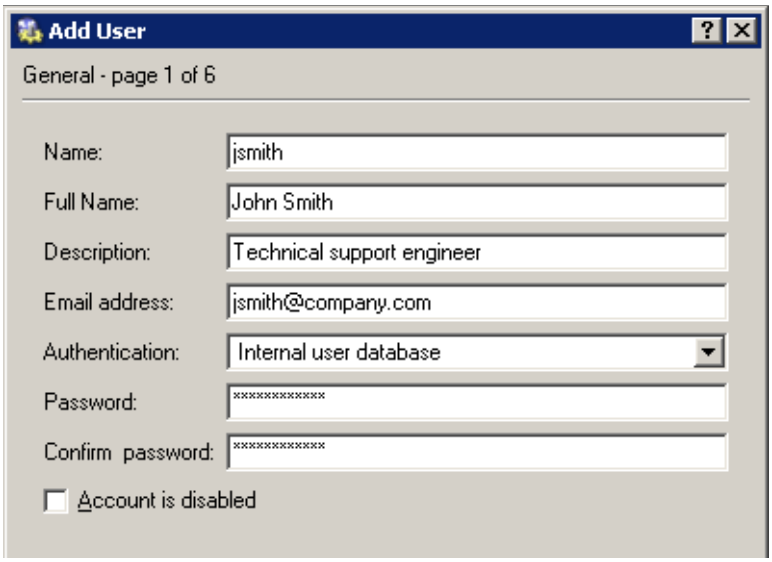
Creating a New User Account

New user accounts can be defined in the *User Accounts* tab under *Users and Groups / Users*.



Use the *Add* button to open a dialog where new user accounts can be defined.

Step 1 — basic information:



Name Username used to log into the program. Usernames are not case-sensitive.

Warning: We recommend not to use special characters (non-English languages) which might cause problems when authenticating via the Web interface.

Full name Full name of the user (usually first name and surname of the user)

Description More information about the user (e.g. grade, position within the company, etc.)

The *Full Name* and the *Description* items have informative values only. Any type of information can be included or the field can be left empty.

Email address Email address of the user that alerts (see chapter 14.3) and other information (e.g. warning if a limit for data transmission is exceeded, etc.) will be sent to. A valid email address should be set for each user, otherwise some of the *WinRoute* features may not be used efficiently.

Note: A relay server must be set in *WinRoute* for each user, otherwise sending of alert messages to users will not function. For details, see chapter 11.9.

Authentication User authentication (see below)

Account is disabled Suspension of a user account without removing it.

Note: For example, this option can be used to create a user account for a user that will not be used immediately (e.g. an account for a new employee who has not taken up yet).

Authentication options:

Internal User Database User account information is stored locally to *WinRoute*. Passwords can be later edited using the Web interface — see chapter 8). NTLM authentication cannot be used for this authentication method.

Warning: Passwords can include printable characters only (letters, digits, punctuation) and are case-sensitive. We recommend not to use special characters (non-English languages) which might cause problems when authenticating to the Web interface.

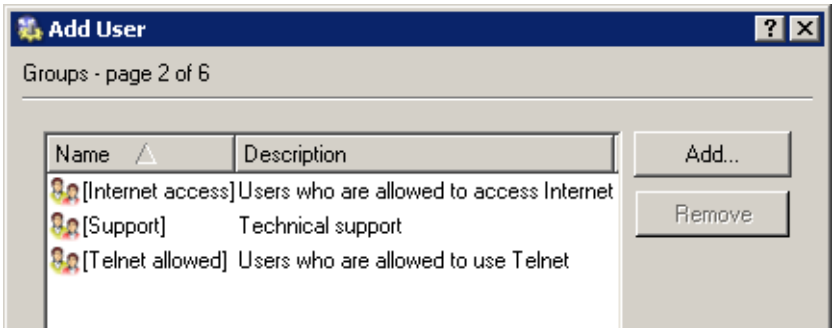
Windows NT Domain Users are authenticated in Windows NT Domain.

This method of authentication cannot be used unless *WinRoute* is running on Windows NT 4.0 / 2000 / XP operating systems.

NT domain / Kerberos 5 Users are authenticated through the Windows NT domain (Windows NT 4.0) or through the Active Directory (Windows 2000/2003).

Go to the *Active Directory / NT domain* tab to set parameters for user authentication through the NT domain or through the Active Directory.

Step 2 — groups:



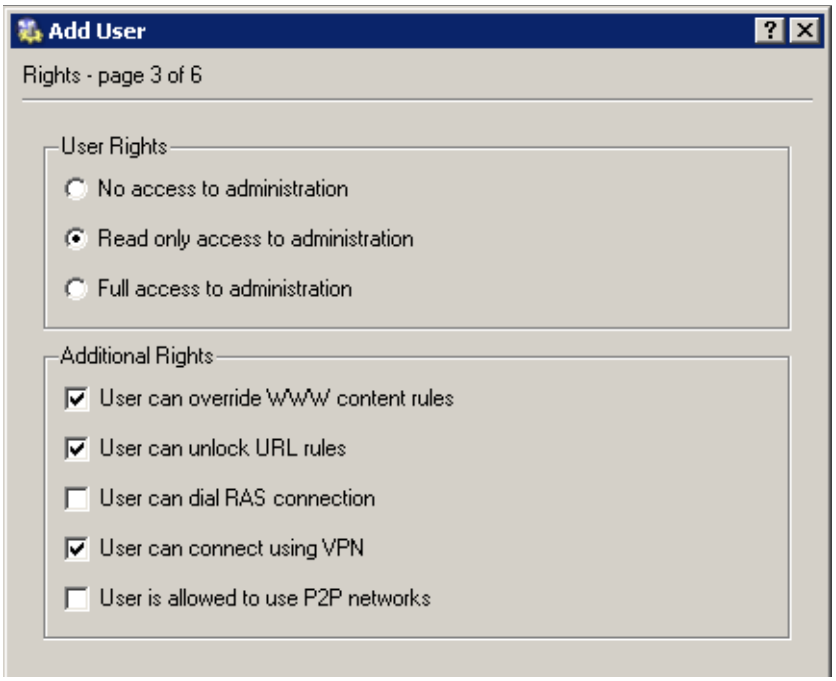
Groups into which the user will be included can be added or removed with the *Add* or the *Remove* button within this dialog (to create new groups go to *User and Groups / Groups* — see chapter 10.4). Follow the same guidelines to add users to groups during group definition. It is not important whether groups or users are defined first.

TIP: While adding new groups you can mark more than one group by holding either the *Ctrl* or the*Shift* key.

Step 3 — access rights:

Each user must have one of the three types of access rights.

No access to administration The user has no rights to access the *WinRoute* administration. This setting is commonly used for the majority of users.



Read only access to administration The user can access *WinRoute*. He or she can read settings and logs but cannot edit them.

Full access to administration The user can read or edit all the records and settings and his or her rights are equal to the administrator rights (`Admin`). If there is at least one user with the full access to the administration, the default `Admin` account can be removed.

Advanced options:

User can override WWW content rules User can customize personal Web content filtering settings independently of the global configuration (for details, refer to *Step 4*).

User can unlock URL rules This option allows the user to unlock Web pages with a forbidden content (the *Unlock* button will be available to the user in the denial page — see details in chapter 6.1).

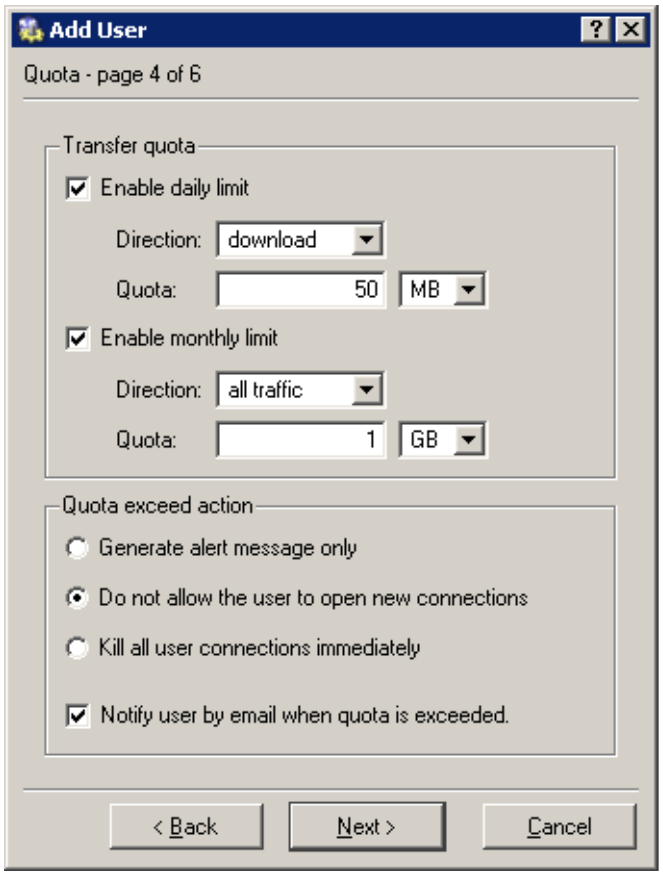
User can dial RAS connection The user is allowed to dial RAS connection in the Web interface (see chapter 8.6) or in the *Administration Console* (in case that the user also possesses at least read rights — for more information, see chapter 4.1).

Note: If the user does not possess this right, he/she will not be allowed to control RAS lines.

User can connect using VPN The user is allowed to connect through *WinRoute's* VPN server (using *Kerio VPN Client*). For detailed information, refer to chapter 12.

User is allowed to use P2P networks Traffic of this user will not be blocked if *P2P* (*Peer-to-peer*) networks are detected. Refer to chapter 11.10.

Step 4 — quota for data transmission



Daily and monthly limit for volume of data transferred by a user, as well as actions to be taken when the quota is exceeded, can be set in this section.

Transfer quota Limit settings

- *Enable daily limit* — daily limit parameters.

  Use the *Direction* combo box to select which transfer direction will be controlled (*download* — incoming data, *upload* — outgoing data, *all traffic* — both incoming and outgoing data).

  The limit can be set in the *Quota* entry using megabytes or gigabytes.

- *Enable monthly limit* — monthly limit parameters. To set this quota, follow the same instructions as for the daily limit.

Quota exceed action Set actions which will be taken whenever a quota is exceeded:

- *Generate alert message only* — no limits will be applied to the user. This option can also be combined with the *Notify user by email when quota is exceeded* option (the user will only be warned about exceeding the quota).
- *Do not allow the user to open new connections* — the user will be allowed to continue using the opened connections, however, will not be allowed to establish new connections (i.e. to connect to another server, download a file through FTP, etc.)
- *Kill all the user's connections immediately* — all traffic of this user will be blocked without hesitation.

Note: If a quota is exceeded, and a blocking action is taken, the restrictions will continue being applied until the end of the quota period (day or month). To cancel these restrictions before the end of a corresponding period, the following actions can be taken:

- disable temporarily a corresponding limit, raise its value or switch to the *Do not block anything* mode
- reset statistics of a corresponding user (see chapter 15.3).

Check the *Notify user by email when quota is exceeded* option to enable sending of warning messages to the user in case that a quota is exceeded. A valid email address must be specified for the user (see *Step 1*). SMTP Relay must be set in *WinRoute* (see chapter 11.9).

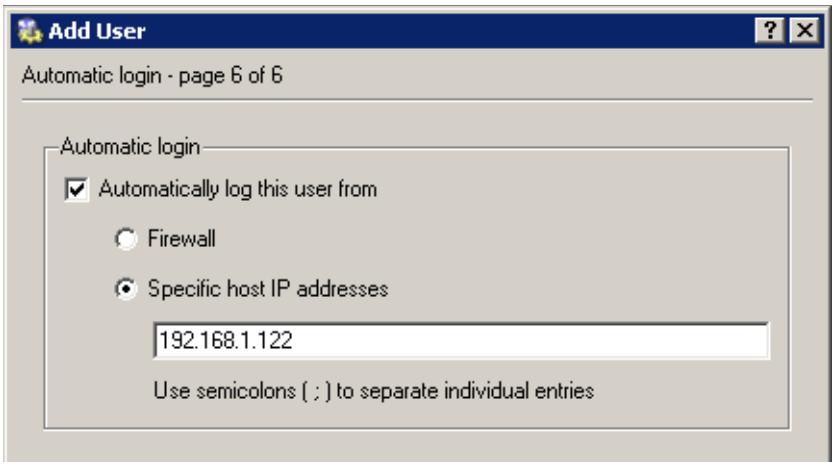
TIP: If you wish that your *WinRoute* administrator is also notified when a quota is almost exceeded, set the notification parameters in *Configuration / Logs and Alerts*. For details, refer to chapter 14.3.

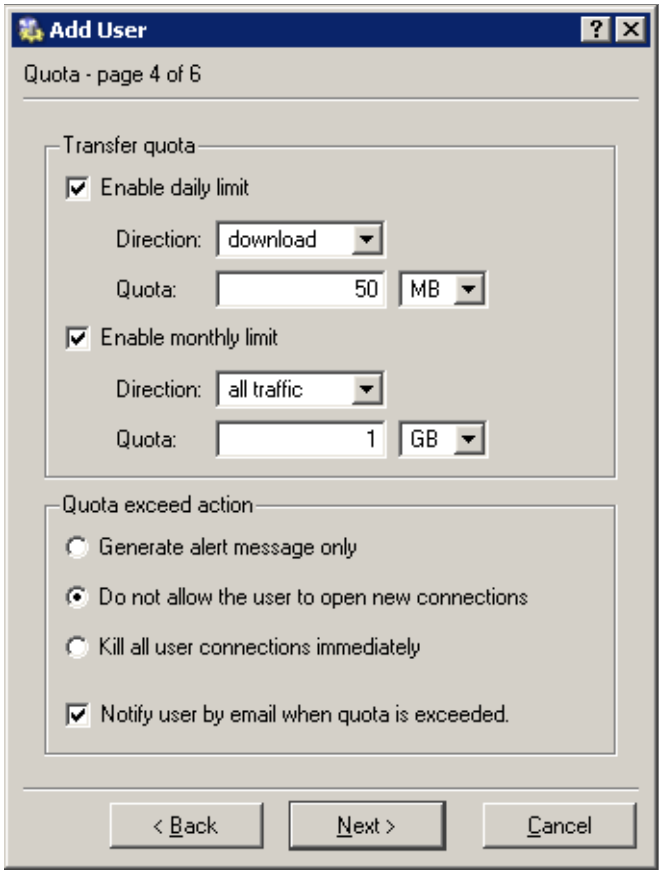
Step 5 — content rules

Within this step special content filter rules settings for individual users can be defined. Global rules (defined in the *Content Rules* tab in the *Configuration / Content Filtering / HTTP Policy* section) are used as default (when a new user account is defined). For details see chapter 6.2).

Note: These settings are available to the user and can be changed in the corresponding page of *WinRoute's* Web interface (see chapter 8.3).Users who are allowed to "override content rules" can customize their settings. Users who are not allowed to override rules can enable or/and disable only features which are available for them (set in their personal configuration).

Step 6 — automatic login





If a user works at a reserved workstation (i.e. this computer is not by any other user) with a fixed IP address (static or reserved at the DHCP server), the user can use automatic login from the particular IP address. This implies that whenever a connection attempt from this IP address is detected, *WinRoute* assumes that the connection is performed by the particular user and it does not require authentication. The user is logged-in

automatically and all functions are available as if connected against the username and password.

This implies that only one user can be automatically authenticated from each IP address. When a user account is being created, *WinRoute* automatically detects whether the specified IP address is used for automatic login or not.

Automatic login can be set for the firewall (i.e. for the *WinRoute* host) or/and for any other host(s) (i.e. when the user connects also from an additional workstation, such as notebooks, etc.).

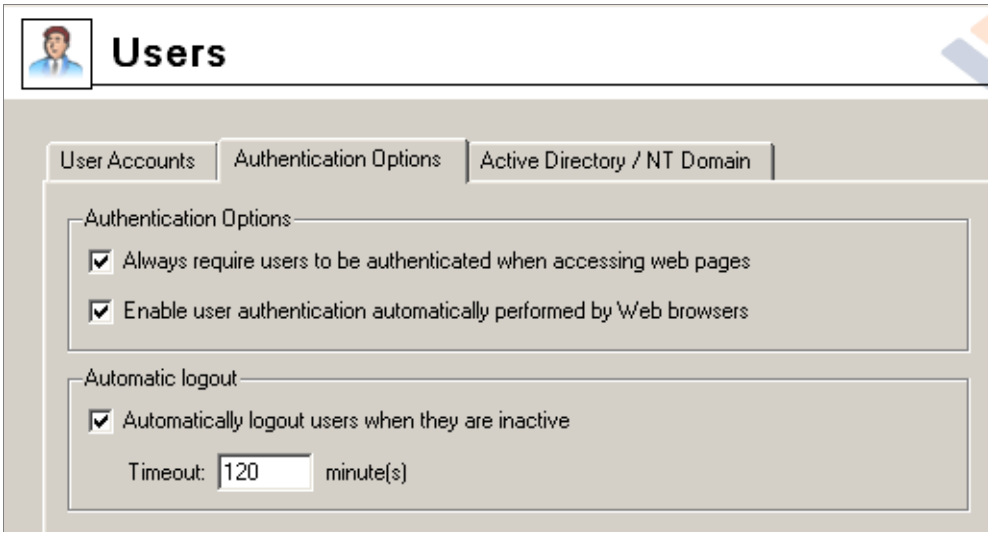
Warning: Automatic login decreases user's security. If an unauthorized user works on the host for which automatic login is enabled, he/she automatically uses the identity of the host's user. Therefore, automatic login should be accompanied by another security feature, such as by user login to the operating system.

User Account Editing and Displaying Statistics

The *Edit* button opens a dialog for editing user account parameters. This dialog has all the properties of the Add User dialog window described above. All the setting options are included in one window only.

10.2 User Authentication settings

Go to the *Authentication Options* tab to set parameters for authentication to the firewall.



Always require users to be authenticated ... Enable this option to require user authentication any time an unauthenticated user attempts to open a Web page. This

implies that the user will be automatically redirected to the authentication page if not authenticated yet (see chapter 8.2) and the demanded Web page will be opened after a successful login.

If the option is disabled, user authentication will be required only for Web pages which are not available (are denied by URL rules) to unauthenticated users (refer to chapter 6.1).

Note: User authentication is used both for accessing a Web page (or/and other services) and for monitoring of activities of individual users (the Internet is not anonymous).

Enable user authentication automatically ... If the *Microsoft Internet Explorer* (version 5.01 or later) is used, the user can use automatic login (using NTLM). The following conditions are applied to this authentication method:

1. The server (i.e. the *WinRoute* host) belongs to a corresponding Windows NT or Kerberos 5 (Windows 2000/2003) domain.
2. Client host belongs to the domain.
3. User at the client host is required to authenticate to this domain (i.e. local user accounts cannot be used for this purpose).
4. *WinRoute Firewall Engine* is running as a service or it is running under a user account with administrator rights to the *WinRoute* host.
5. NTLM cannot be used for authentication in the internal database.

Automatically logout users when they are inactive Time interval (in minutes) of allowed user inactivity. After this period expires, the user will be automatically logged out from the firewall. The default timeout value is 120 minutes (2 hours).

This situation often comes up when a user forgets to logout from the firewall. Therefore, it is not recommended to disable this option (or set the value to 0), otherwise login data of a user who forgot to logout might be misused by an unauthorized user.

10.3 External authentication and import of user accounts

WinRoute supports the following methods of saving of user accounts and of user authentication:

- *Internal user database* — user accounts and their passwords are saved in *WinRoute* (see above). During authentication, usernames are compared to the data in the internal database.

This method of saving accounts and user authentication is particularly adequate for networks without a proper domain, as well as for special administrator accounts (user can authenticate locally even if the network communication fails).

On the other hand, in case of networks with proper domains (Windows NT or Active Directory), local accounts in *WinRoute* may cause increased demands on administration since accounts and passwords must be maintained twice (at the domain and in *WinRoute*).

- *Internal user database with authentication at the domain* — although user accounts are saved in the *WinRoute* database, users are authenticated through the domain (i.e. passwords are not saved in a corresponding user account under *WinRoute*). Obviously, usernames in *WinRoute* must match with the usernames in the domain.

  This method is not so demanding as far as the administration is concerned. When, for example, a user wants to change the password, it can be simply done at the domain and the change will be automatically applied to the account in *WinRoute*. In addition to this, it is not necessary to create user accounts in *WinRoute* by hand, as they can be imported from a corresponding domain.

- *Active Directory accounts (automatic import)* — if Active Directory (Windows 2000 Server /Server 2003) is used, automatic import of user accounts can be set. It is not necessary to define accounts in *WinRoute*, nor import them, since it is possible to to configure templates by which specific parameters (such as access rights, content rules, transfer quotas, etc.) will be set for new *WinRoute* users. A corresponding user account will be imported upon the first login of the user to *WinRoute*.

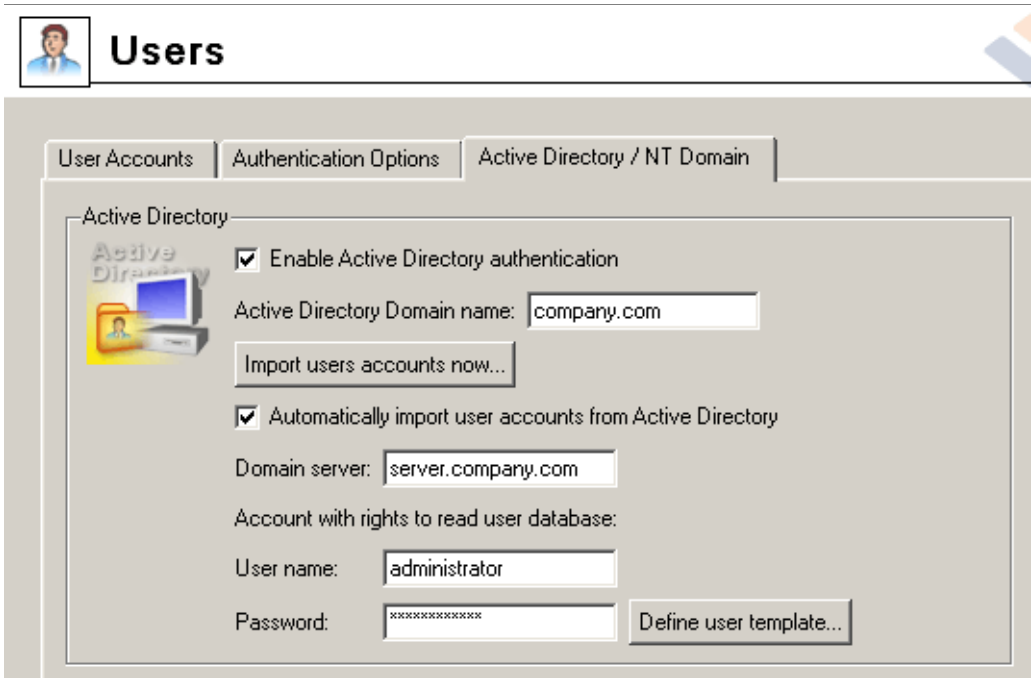
  This method is less demanding on the administration (all user accounts are administered through Active Directory).

Note: In cases when users are authenticated at the domain (the last two descriptions), it is recommended to create at least one local account with full rights to administration in *WinRoute*, so that it is possible to connect to the *WinRoute* administration even if the network or the domain fails.

Active Directory

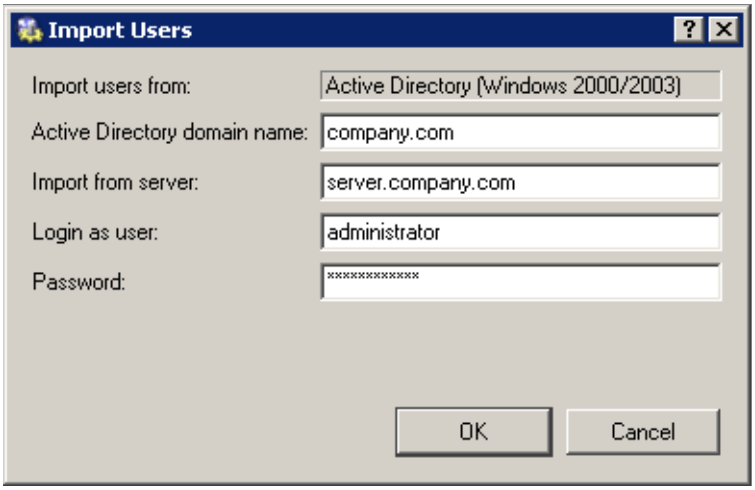
Parameters for user authentication at Active Directory (or/and automatic import of user accounts) can be configured in the *Active Directory / NT domain* tab.

Enable Active Directory authentication This option enables/disables Active Directory. If it is disabled, all accounts which use Active Directory authentication are unavailable (these users cannot connect to their accounts).



Active Directory domain name The domain (*Kerberos realm*) at which users will be authenticated. Only complete domain names are accepted (e.g. `company.com`, not just `company`).

Import user accounts now Use this button to open a dialog for immediate import (download) of user accounts from Active Directory.



The following information is required for import of accounts:

- *Active Directory domain name* — name of the domain from which user accounts will be imported (e.g `company.com`).
- *Import from server* — DNS name or IP address of the *Active Directory* domain server (e.g. `server.company.com` or `192.168.1.1`).
- *Login as user*, *Password* — username and password of a user who belongs to the domain (i.e. has an account in this domain). No special user rights are required.

If no problem arises (i.e. the inserted data is correct, the server is available, etc.), a list of accounts will be displayed upon clicking *OK* that are ready to be imported to *WinRoute*.

Note: The *NT domain / Kerberos 5* authentication method will be set for all imported accounts.

Automatically import user accounts from Active Directory This option enables automatic import of accounts from Active Directory. Like in case of manual import of accounts, name or IP address of the domain server as well as username and password are required for the authentication (for details, see above).

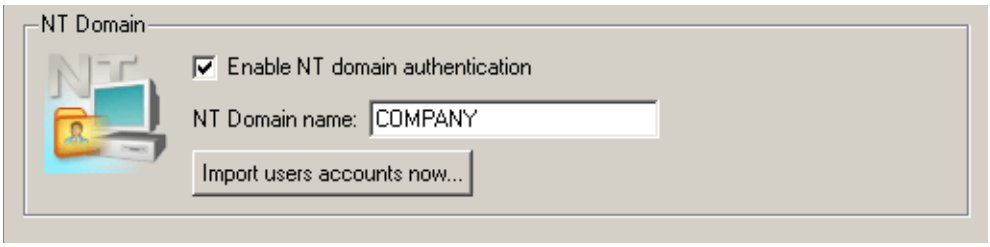
Define user template This button opens a dialog where a configuration template for imported user accounts (specific parameters for *WinRoute*) can be set.

The dialog is similar to the dialog for user account modification, however, it includes only the *Groups*, *Rights*, *Quota* and *Content rules* tabs (since the other parameters cannot be set at once for multiple users). For details, refer to chapter 10.1

NT domain

Parameters for user authentication at the NT domain can be set in the *Active Directory / NT domain* tab.

Warning: Do not use this method if the domain server runs on OS Windows 2000 Server /Server 2003! Use the *Active Directory* in this case.

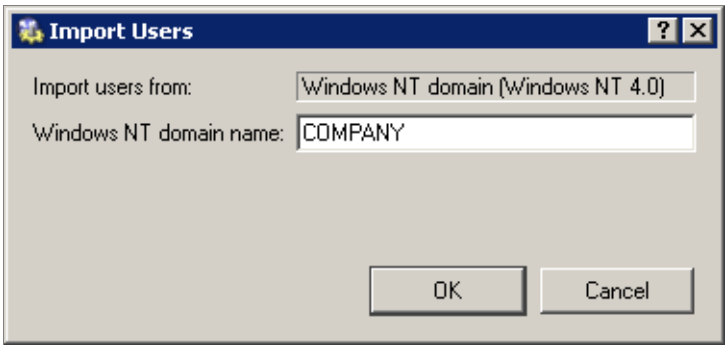


Enable NT domain authentication This option enables/disables NT domain authentication. If it is disabled, all accounts which use NT domain authentication are unavailable (these users cannot connect to their accounts).

NT domain name Name of the domain (e.g. `COMPANY`) where users will be authenticated.

Note: The host where *WinRoute* is installed must belong to this domain.

Import user accounts now Use this option to open a dialog where NT domain user accounts can be imported. Specify the *Windows NT domain name* entry.



If no problem arises (i.e. the domain name is correct, the server is available, etc.), a list of accounts will be displayed upon clicking *OK* that are ready to be imported to *WinRoute*.

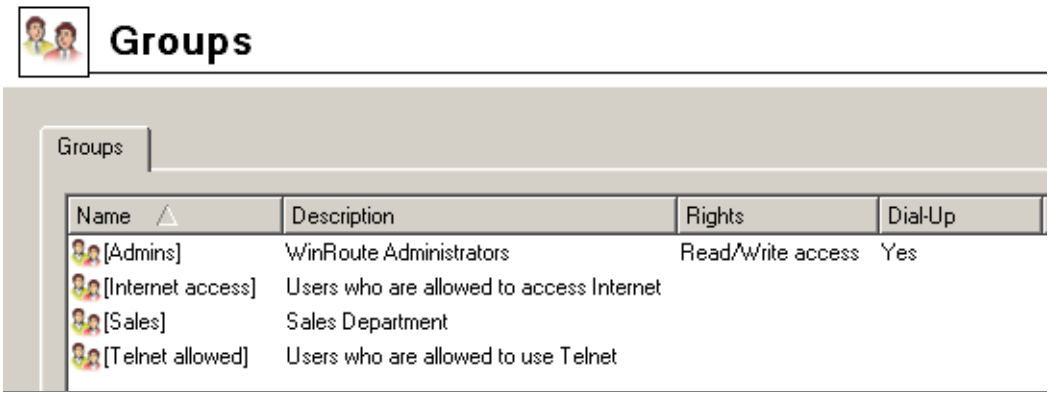
Note: The *NT domain / Kerberos 5* authentication method will be set for all imported accounts.

10.4 User Groups

User accounts can be sorted into groups . Creating user groups provides the following benefits:

- Specific access rights can be defined for a user group. These rights complement rights of each user.
- Each group can be used when traffic and access rules are defined. This simplifies the definition process so that you will not need to define the same rule for each user.

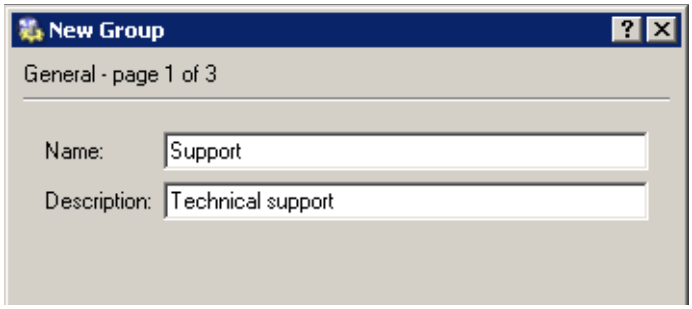
User groups can be defined in *User and Groups / Groups*.



Creating a New User Group

Clicking on the *Add* button will open a dialog where new user groups can be created.

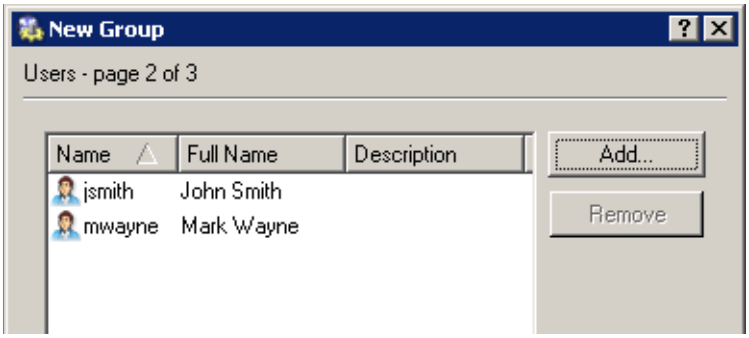
Step 1 — group name and description:



Name Group name (group identification).

Description Group description. It has an informative purpose only and may contain any information or the field can be left empty.

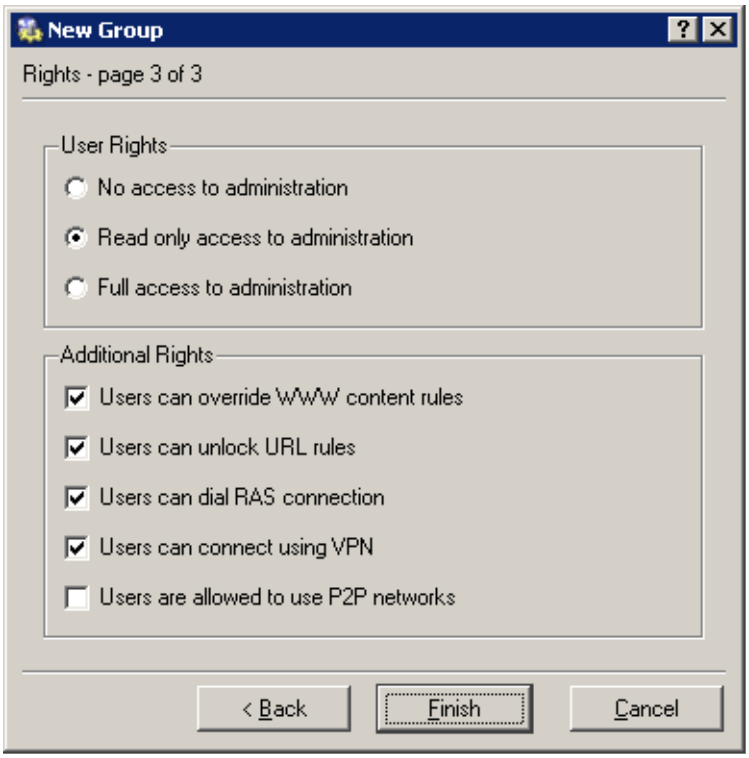
Step 2 — group members



Users can be added or removed from the group with the *Add/Remove* buttons. If user accounts have not been created yet, the group can be left empty and users can be added during the account definition (see chapter 10.1).

Tip: To select more than one user hold the *Ctrl* or the *Shift* key.

Step 3 — group access rights



Each group must have one of the following three types of access rights:

No access to administration Users included in this group cannot access the *WinRoute* administration.

Read only access to administration Users included in this group can access the *WinRoute* administration. However, they can only read the records and settings and they are not allowed to edit them.

Full access to administration Users included in this group have full access rights for the administration.

Advanced options:

Users can override WWW content rules User belonging to the group can customize personal Web content filtering settings independently of the global configuration (for details see chapters 6.2 a 8.3).

User can unlock URL rules User will be allowed to unlock Web pages with forbidden content.

Users can dial RAS connection Users included in this group will be allowed to connect and hang up the dialup lines defined in *WinRoute* (with *Kerio Administration Console* or with WWW administration interface, see chapter 8).

Users can connect using VPN Members of this group can connect through *WinRoute's* VPN server (for details, refer to chapter 12.1).

Users are allowed to use P2P networks The *P2P Eliminator* module (detection and blocking of *Peer-to-peer* networks — see chapter 11.10) will not be applied to members of this group.

Group access rights are combined with user access rights. This means that current user rights are defined by actual rights of the user and by rights of all groups in which the user is included.

Chapter 11

Advanced Settings

11.1 Remote Administration Settings

Remote administration can be either permitted or denied by definition of the appropriate traffic rule. Traffic between *WinRoute* and *Kerio Administration Console* is performed by TCP and UDP protocols over port 44333. The definition can be done with the predefined service *KWF Admin*.

How to allow remote administration from the Internet

In the following example we will demonstrate how to allow *WinRoute* remote administration from some Internet IP addresses.

- *Source* — group of IP addresses from which remote administration will be allowed.

  For security reasons it is not recommended to allow remote administration from an arbitrary host within the Internet (this means: do not set *Source* as the Web interface).

- *Destination* — *Firewall* (host where *WinRoute* is running)

- *Service* — *KWF Admin* (predefined service— *WinRoute* administration)

- *Action* — *Permit*

- *Translation* — Because the engine is running on the firewall there is no need for translation.

Note: Be very careful while defining traffic rules, otherwise you could block remote administration from the host you are currently working on. If this happens, the connection between *Kerio Administration Console* and *WinRoute Firewall Engine* is interrupted (upon clicking on the *Apply* button in *Configuration / Traffic Policy*). Local connections (from the *WinRoute Firewall Engine's* host) cannot be blocked by any rule.

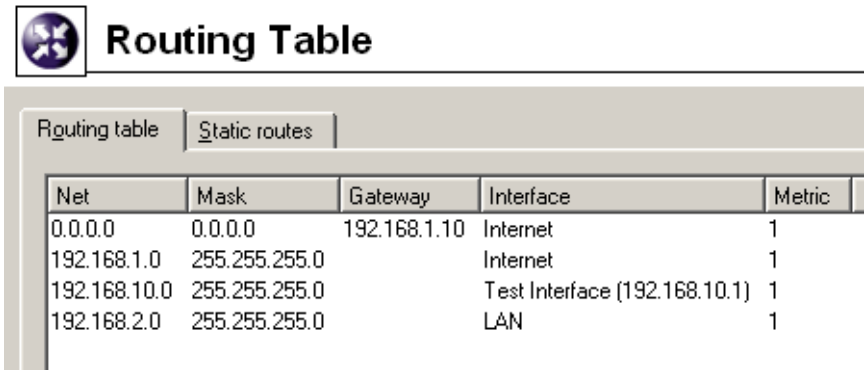
TIP: The same method can be used to enable or disable remote administration of *Kerio MailServer* through *WinRoute* (the *KMS Admin* service can be used for this purpose).

11.2 Routing Table

Using *Kerio Administration Console* you can view or edit the system routing table of the *WinRoute* host. This can be useful especially to resolve routing problems remotely (it is not necessary to use applications for terminal access, remote desktop, etc.).

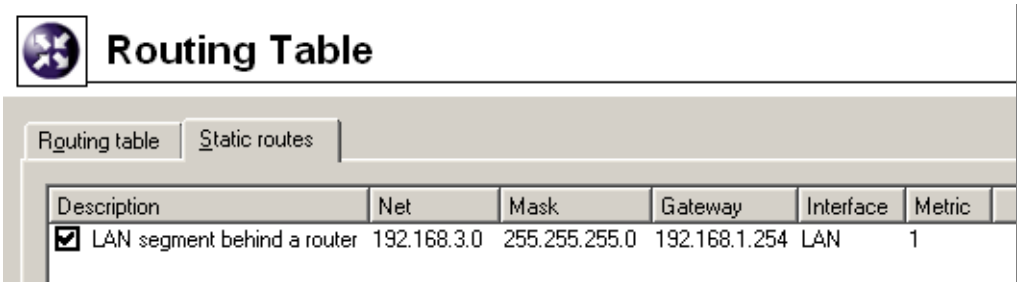
To view or modify the routing table go to *Configuration / Routing Table*. This section is divided into two tabs:

- *Routing Table* — current routing table of the operating system (including so called persistent routes under Windows 2000 and Windows XP operating systems).



  You can also add or remove dynamic routes. New dynamic routes are valid only until the operating system restart or unless removed using the `route` system command.

- *Static Routes* — persistent routes refreshed by *WinRoute* even after restart of the operating system.



  WinRoute contains a special mechanism that is used to generate and maintain static routes in the routing table. All routes defined in the *Static Routes* folder are stored

in the configuration file and inserted into the system routing table after each startup of *WinRoute Firewall Engine*. These routes are monitored while *WinRoute* is running — if any of the routes are removed with the `route` command, it will be automatically reinserted by *WinRoute*.

Notes:

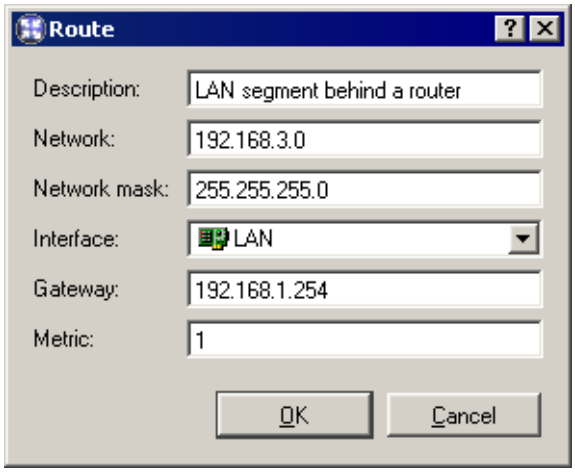
1. Persistent routes are not used for implementation of static routes as this function is not available on all operating systems.

2. If you use a dial-up connection, then packets routed via this route dial a line (for more information refer to chapter 11.3).

Warning: Changes in the routing table might interrupt the connection between the *WinRoute Firewall Engine* and the *Kerio Administration Console*. Therefore, only experienced users with knowledge of IP routing should use this feature.

Definitions of Dynamic and Static Rules

Click on the *Add* (or *Edit* when a particular route is selected) button to display a dialog for route definition.



Description Comment about the route. This item is available in the *Static Routes* folder only.

Network, Network Mask IP address and mask of the destination network.

Interface Selection of an interface through which the specific packet should be forwarded.

Gateway IP address of the gateway (router) which can route to the destination network. The IP address of the gateway must be in the same IP subnet as the selected interface.

Metric "Distance" of the destination network. The number stands for the number of routers that a packet must pass through to reach the destination network.

Metric is used to find the best route to the desired network. The lower the metric value, the "shorter" the route is.

Note: Metric in the routing table may differ from the real network topology. It may be modified according to the priority of each line, etc.

Removing routes from the Routing Table

The following rules are used for route removal:

- Routes in the *Static Routes* folder are managed by *WinRoute*. Removal of any of the routes within this folder would remove the route from the system routing table immediately and permanently (after clicking on the *Apply* button).
- Manually defined dynamic routes will be removed regardless of how they were added, whether in *Kerio Administration Console* or using the `route` command.
- Persistent routes will be removed from the routing table only after restart of the operating system. It will be automatically refreshed upon reboot. There are many methods that can be used to create persistent routes (the methods vary according to operating system — in some systems, the `route -p` command can be used, etc.).

11.3 Demand Dial

If the *WinRoute* host is connected to the Internet via dial-up, *WinRoute* can automatically dial the connection when users attempt to access the Internet. *WinRoute* provides the following options of dialing/hanging control:

- Line is dialed when a request from the local network is received. This function is called Demand dial. For further description see below.
- Line is disconnected automatically if idle for a certain period (no data is transmitted in both directions). For a description of the automatic disconnection, refer to chapter 4.1.

How demand dial works

First, the function of demand dial must be activated within the appropriate line (either permanently or during a defined time period). This may be defined in *Configuration / Interfaces* (for details see chapter 4.1).

Second, there must be no default gateway in the operating system (no default gateway must be defined for any network adapter).

If *WinRoute* receives a packet from the local network, it will compare it with the system routing table. If no default route is available, *WinRoute* holds the packet in the cache and dials the appropriate line if the demand dial function is enabled. This creates an outgoing route in the routing table via which the packet will be sent.

The line may be either disconnected manually or automatically if idle for a certain time period.

Notes:

1. To ensure correct functionality of demand dialing there must be no default gateway set at network adapters. If there is a default gateway at any interface, packets to the Internet would be routed via this interface (no matter where it is actually connected to) and *WinRoute* would not dial the line.

2. If multiple demand dial RAS lines are defined in *WinRoute*, the one that was defined first will be used. *WinRoute* does not enable automatic selection of a line to be dialed.

3. Lines can be also dialed if this is defined by a static route in the routing table (refer to chapter 11.2). If a static route via the dial-up is defined, the packet matching this route will dial the line. This line will not be used as the default route — the *Use default gateway on remote network* option in the dial-up definition will be ignored.

4. According to the factors that affect total time since receiving the request until the line is dialed (i.e. line speed, time needed to dial the line, etc.) the client might consider the destination server unavailable (if the timeout expires) before a successful connection attempt. However, *WinRoute* always finishes dial attempts. In such cases, simply repeat the request, i.e. with the *Refresh* button in your browser.

Technical Peculiarities and Limitations

Demand dialing has its peculiarities and limitations. The limitations should be considered especially within designing and configuration of the network that will use *WinRoute* for connection and of the dial-up connected to the Internet.

1. Demand dial cannot be performed directly from the host where *WinRoute* is installed because it is initiated by *WinRoute* low-lever driver. This driver holds packets and

decides whether the line should be dialed or not. If the line is disconnected and a packet is sent from the local host to the Internet, the packet will be dropped by the operating system before the *WinRoute* driver is able to capture it.

2. Typically the server is represented by the DNS name within traffic between clients and an Internet server. Therefore, the first packet sent by a client is represented by the DNS query that is intended to resolve a host name to an IP address.

   In this example, the DNS server is the *WinRoute* host (this is very common) and the line to the Internet is disconnected. A client's request on this DNS server is traffic within the local network and, therefore, it will not result in dialing the line. If the DNS server does not have the appropriate entry in the cache , it must forward the request to another server on the Internet. The packet is forwarded to the Internet by the local DNS client that is run at the *WinRoute* host. This packet cannot be held and it will not cause dialing of the line. Therefore, the DNS request cannot be answered and the the traffic cannot continue.

   For these reasons, *WinRoute DNS Forwarder* enables automatic dialing (if the DNS server cannot respond to the request itself). This function is dependent on demand dial — if the demand dial function is disabled, the *DNS Forwarder* will not dial the line.

   Note: If the DNS server is located on another host within the local network or clients within the local network use an Internet DNS server, then the limitation is irrelevant and the dialing will be available. If clients' DNS server is located on the Internet, the line will be dialed upon a client's DNS query. If a local DNS server is used, the line will be dialed upon a query sent by this server to the Internet (the default gateway of the host where the DNS server is running must be set to the IP address of the *WinRoute* host).

3. It can be easily understood through the last point that if the DNS server is to be running at the *WinRoute* host, it must be represented by *DNS Forwarder* because it can dial the line if necessary.

   If there is a domain that is based on Active Directory in the Windows 2000 local network, Microsoft DNS server must be used as communication with Active Directory is performed according to special types of DNS requests. Microsoft DNS server does not support automatic dialing. Moreover, it cannot be used at the same host as *DNS Forwarder* as it would cause collision of ports.

   As understood from the facts above, if the Internet connection is to be available via dial-up, *WinRoute cannot* be used at the same host where Windows 2000 server Active Directory and Microsoft DNS are running.

4. If *DNS Forwarder* is used, *WinRoute* can dial as a response to a client's request if the following conditions are met:

   - Destination server must be defined by DNS name so that the application can create a DNS query.
   - In the operating system, set the primary DNS server to the IP address of the firewall). In Windows operating system, go to TCP/IP properties and set the IP address of this interface as the primary DNS.
   - *DNS Forwarder* must be configured to forward requests to one of the defined DNS servers (the *Forward queries to the specified DNS server(s)* option). Automatic detection of DNS servers are not available. For details refer to chapter 4.3.

5. The *Proxy server* in *WinRoute* (see chapter 4.5) also provides direct dial-up connections. A special page providing information on the connection process is opened (the page is refreshed in short periods). Upon a successful connection, the browser is redirected to the specified Website.

Setting Rules for Demand Dial

Demand dial functions may cause unintentional dialing. It's usually caused by DNS queries that are handled by the *DNS Forwarder* The following causes apply:

- User host generates a DNS query in the absence of the user. This traffic attempt may be a banner from a local HTML page or automatic update of an installed application.
- *DNS Forwarder* performs dialing in response to requests of names of local hosts. Define DNS for the local domain properly (use the `hosts` system file of the *WinRoute* host — for details see chapter 4.3).

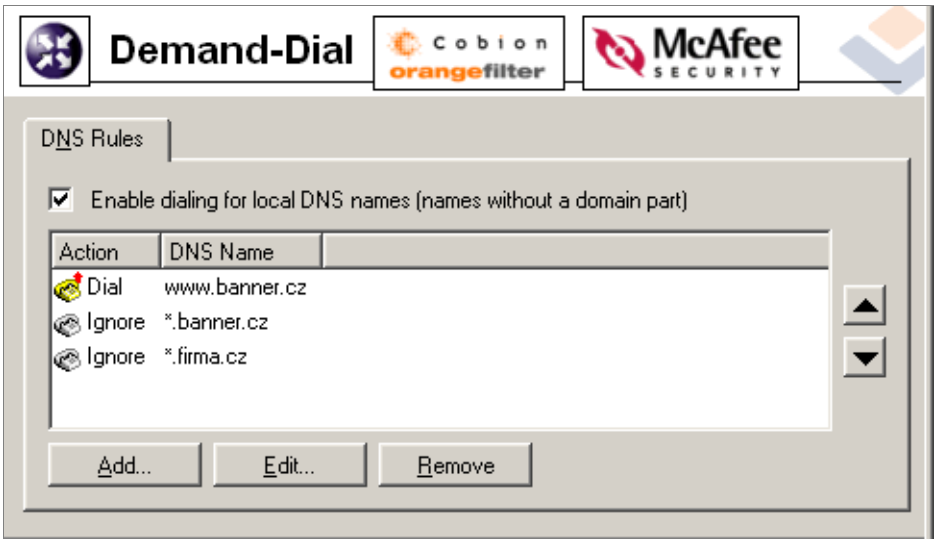
Note: In *WinRoute*, unwanted traffic may be blocked. However, for security reasons it is recommended to detect the root of the problem (i.e. use antivirus to secure the workstation, etc.).

In *Configuration / Demand Dial* within *Kerio Administration Console*, detailed rules for dialing certain DNS names may be defined.

In this section you can create a rule list of DNS names.

Either whole *DNS name* or only its end or beginning completed by an asterisk (*) may be entered. An asterisk may stand for any number of characters.

In *Actions* you can select from the *Dial* or *Ignore* options. Use the second option to block dialing of the line in response to a query on the DNS name.



Rule lists are searched downwards (rule order can be modified with the arrows at the right side of the window). When the system detects the first rule that meets all requirements, the desired action is executed and the search is stopped. All DNS names missing a suitable rule will be dialed automatically by *DNS Forwarder* when demanded.

The *Dial* function can be used for creating advanced and more complex rules. For example, dial can be permitted for one name within the domain and denied for the others (see the figure).

Dial of local DNS names Local DNS names are names of hosts within the domain (names that do not include a domain).

Example: The local domain is called `company.com`. The host is called `pc1`. The full name of the host is `pc1.company.com` whereas local name in this domain is `pc1`.

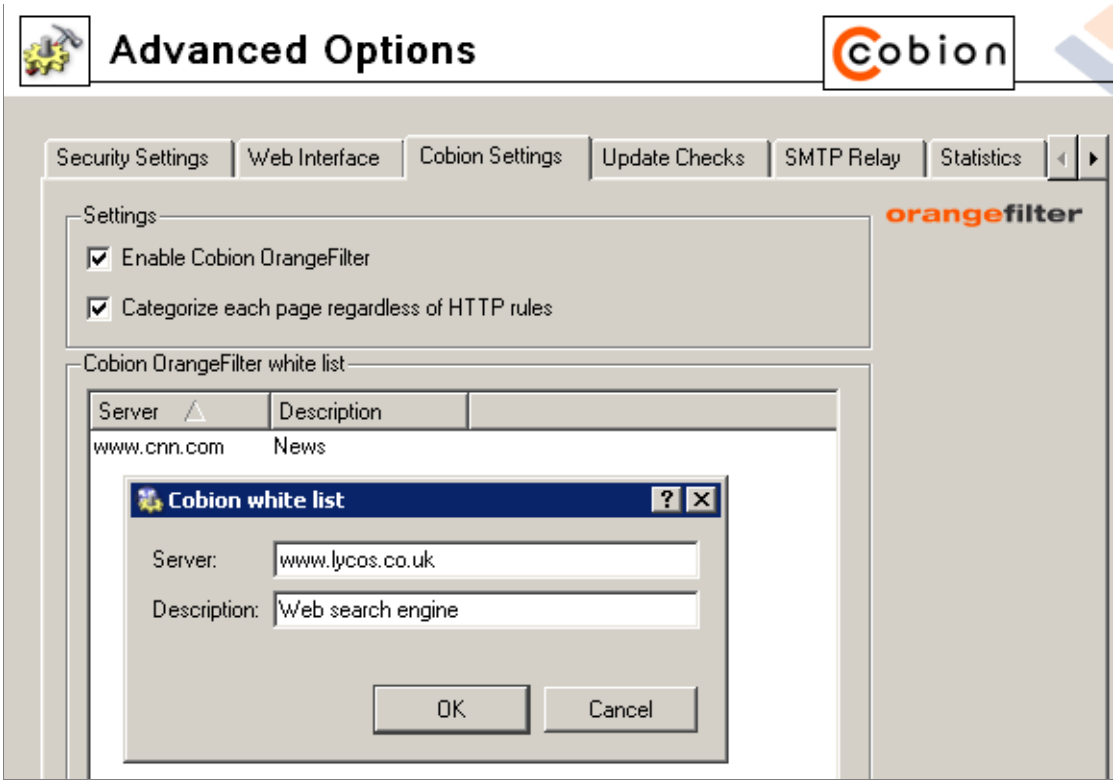
Local names are usually stored in the database of the local DNS server (in this example, the names are stored in the `hosts` file at the *WinRoute* host that uses *DNS Forwarder*). Set by default, *DNS Forwarder* does not dial these names as names are considered non-existent unless they can be found in the local DNS database.

If the primary server of the local domain is located outside of the local network, it is necessary that the *DNS Forwarder* also dials the line if requests come from these names. Activate the *Enable dialing for local DNS names* option in the *Other settings* tab to enable this (at the top of the *Demand Dial* dialog window).

11.4 Cobion Settings

Cobion can be set and configured through the *Cobion Settings* tab in *Configuration / Advanced Options*.

Note: A special license is associated with *Cobion*. Unless *WinRoute* includes a *Cobion* license, then the *Cobion* system behaves as a trial version only (this means that it is automatically disabled after 30 days from the *WinRoute* installation and options in the *Cobion Settings* tab will not be available). For detailed information about the licensing policy, read chapter 13.



Cobion white list Servers specified through this section will not be tested by the *Cobion Orange Filter* system. Click on the *Add* button to add a new item (server).

Server There are several methods to define a server. You can define either an exact URL (i.e. `www.google.com/index.html`), URL substring using wildcard matching (i.e. `*.g?ogle.com*`) or a server name (i.e. `www.google.com`). A server name is represented by any URL on a given server (`www.google.com/*`).

Description For reference only.

Enable Cobion OrangeFilter module use this option to enable/disable *Cobion* — i.e. the *Cobion OrangeFilter* module for classification of websites.

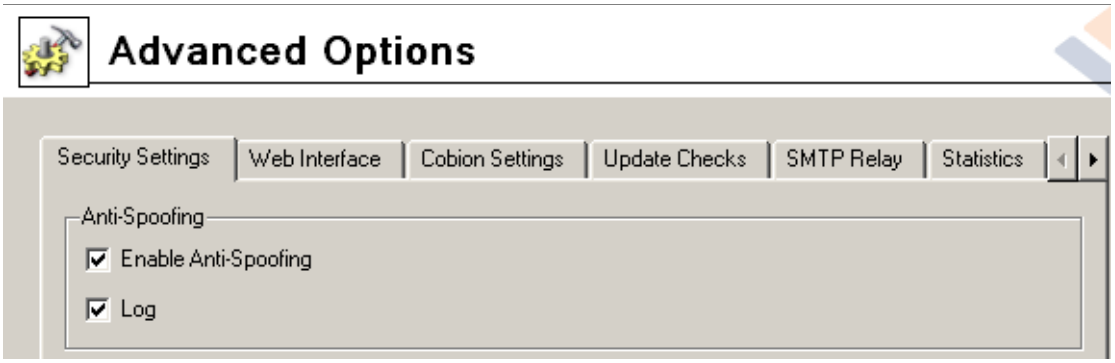
If *Cobion* is disabled:

- the other options in the *Cobion Settings* tab are not available,
- all URL rules which use the *Cobion OrangeFilter* classification are disabled (for details, refer to chapter 6.3).

Categorize each page regardless of HTTP rules Enable this option to let *Cobion OrangeFilter* categorize all Web pages (included denied ones). This can be useful especially for statistic monitoring (see chapter 15.3).

11.5 Security Settings

WinRoute provides several security options which cannot be defined by traffic rules. These options can be set in the *Security settings* tab of the *Configuration / Advanced Options* section.



Anti-Spoofing

Spoofing is a process of translating the IP address of a given packet so that a firewall will believe the request came from a trusted source. Although the packet cannot be routed back to the initial source, there is potential for unnecessary network congestion and possible denial of service. *WinRoute* is capable of monitoring traffic to verify that packets arriving on an interface do not have a source address which is associated with a network of an opposing interface. In other words, such traffic (although possible) is never justified and should therefore be discarded.

The *Anti-Spoofing* function can be configured in the *Anti-Spoofing* folder in *Configuration / Advanced Options*.

Enable Anti-Spoofing This option activates *Anti-Spoofing*.

Log If this option is on, all packets that have not passed the anti-spoofing rules will be logged in the *Security* log (for details see chapter 16.11).

Connections Count Limit

This function defines a limit for the maximum number of connections per host. This function can be enabled/disabled and set through the *Security Settings* tab in *Configuration / Advanced Options*.

This function can be helpful especially for the following cases:

- Any service (e.g. WWW server) which is available from the Internet (allowed by traffic rules —see chapter 5) is running on the local network. Connection count limits protect internal servers from flooding (*DoS* type attacks — *Denial of Service*).

  In this case, the limit is applied to the local server — sum of all connections of all connected clients must not exceed this limit.

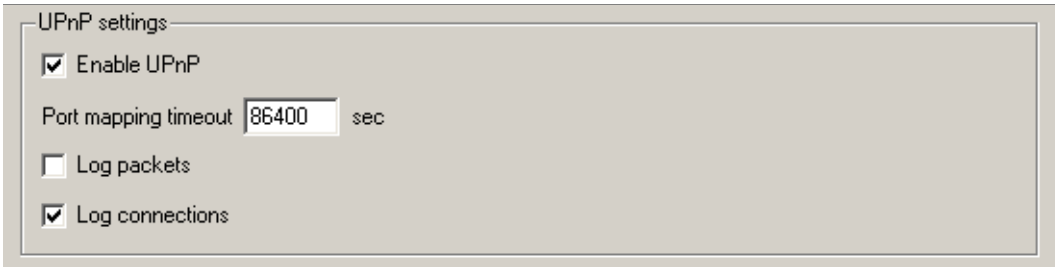
- Client computer (workstation) in the local network is attacked by a worm or a Trojan horse which is trying to establish a connection to many servers. Connection count limits protects the *WinRoute* host from flooding and it can reduce undesirable activities by worms and trojan horses.

  In this case, the limit is applied to a host (workstation) in the local network — the sum of all connections established from this computer to individual servers in the Internet must not exceed the limit.

11.6 Universal Plug-and-Play (UPnP)

WinRoute supports UPnP protocol (*Universal Plug-and-Play*). This protocol enables client applications (i.e. *Microsoft Messenger*) to detect the firewall and make a request for mapping of appropriate ports for the particular host. This mapping is temporary.

To configure UPnP go to the *Security Settings* folder in *Configuration / Advanced Options*.



Enable UPnP This option enables UPnP.

Warning: If *WinRoute* is running on the Windows XP operating system, check whether the following system services are not running before you start the *UPnP* function:

- *SSDP Discovery Service*
- *Universal Plug and Play Device Host*

If any of these services is running, close it and deny its automatic startup.

In *WinRoute* these services cannot be used together with *UPnP*.

Port mapping timeout For security reasons, ports required by applications are mapped for a certain time period only. Mapping is closed automatically on demand of the application or when the timeout (in seconds) expires.

UPnP also enables the application to open ports for a requested period. Here the *Port mapping timeout* parameter also represents a maximal time period that the port will be available to an application (even if the application demands a longer period, the period is automatically reduced to this value).

Log packets If this option is enabled, all packets passing through ports mapped with UPnP will be recorded in the *Security* log (see chapter 16.11).

Log connections If this option is enabled, all connections passing through ports mapped with UPnP will be recorded in the *Connection* log (see chapter 16.5).

Warning: Apart from the fact that UPnP is a useful feature, it may also endanger network security, especially in case of networks with many users where the firewall could be controlled by too many users. A *WinRoute* administrator should consider carefully whether to prefer security or functionality of applications that require UPnP.

Using traffic policy (see chapter 5.2) you can limit usage of UPnP and enable it to certain IP addresses or certain users only.

Example:

The first rule allows UPnP only from *UPnP Clients* IP group. The second rule denies UPnP from other hosts (IP addresses).

11.7 VPN using IPSec Protocol

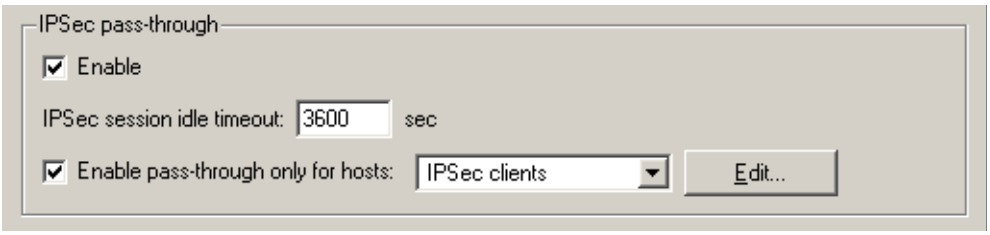
IPSec (IP Security Protocol) is an extended IP protocol. It provides encrypted security services. These services enable authentication, as well as for access and trustworthiness control. IPSec provides similar services as SSL, but it works on a network layer. Through IPSec you can create encrypted tunnels (VPN) or encrypt traffic between two hosts.

WinRoute includes so called *IPSec* pass-through. This implies that *WinRoute* does not include tools for establishing an *IPSec* connection (tunnel), however, it is able to detect *IPSec* protocol and enable it for traffic between the local network and the Internet.

Note: The IPSec Pass-Through function guarantees full functionality of existing IPSec clients and servers after deployment of *WinRoute* at the Internet gateway. If you consider designing and implementation of new virtual private networks, we recommend you to use the *WinRoute* proprietary VPN solution (see chapter 12).

IPSec preferences

IPSec preferences can be set in the *IPSec pass-through* area in the *Security Settings* tab of the *Configuration / Advanced Options* section. For detailed information on IPSec refer to chapter *WinRoute's IPSec configuration*.



Enable This option enables *IPSec* pass-through.

It is necessary to set idle timeout for *IPSec* connections (default time is 3600 seconds which is exactly 1 hour). If no data is transferred for this time and a connection is not closed properly, *WinRoute* will consider the connection closed and the pass-through is available to another computer (another IP address).

Enable pass-through only for hosts It is possible to narrow the number of hosts using *IPSec* pass-through by defining a certain scope of IP addresses (typically hosts on which IPSec clients will be run). Use the *Edit* button to edit a selected IP group or to add a new one.

WinRoute's *IPSec configuration*

Generally, communication through *IPSec* must be permitted by firewall policy (for details refer to chapter 5.2). IPSec protocol uses two traffic channels:

- *IKE* (*Internet Key Exchange* — exchange of encryption keys and other information).
- encrypted data (*IP* protocol number `50` is used)

Open the *Configuration / Traffic Policy* section to define a rule which will permit communication between IPSec clients (`VPN` address group is described in the example) and IPSec server for the services (`ipsec.server.cz` server is described in the example).

Name	Source	Destination	Service	Action
IPSec traffic	IPSec clients	ipsec.server.com	IKE IPSec	✓

Note: Predefined *IPSec* and *IKE* services are provided in *WinRoute*.

IPSec client in local network

This section of the guide describes *WinRoute* configuration for cases when an IPSec client or the server is located in the local network and *WinRoute* provides translation of IP addresses (NAT — for details see chapter 5).

1. *IPSec client on WinRoute host*

   In this case IPSec traffic is not influenced by NAT (IPSec client must be set so that it uses the public IP address of the *WinRoute* host). It is only necessary to define a traffic rule permitting IPSec communication between the firewall and the IPSec server.

Name	Source	Destination	Service	Action	Translation
IPSec traffic	Firewall	ipsec.server.com	IKE IPSec	✓	

   The *Translation* column must be blank — no IP translation is performed. The pass-through setting is not important in this case (it cannot be applied).

2. *One IPSec client in the local network (one tunnel)*

   If only one IPSec tunnel from the local network to the Internet is created at one moment, then it depends on the type of IPSec client:

- If IPSec client and the IPSec server support the *NAT Traversal* function (the client and the server are able to detect that the IP address is translated on the way between them), IPSec must be *disabled* (otherwise a collision might arise).

  NAT Traversal is supported for example by *Nortel Networks'* VPN software (`http://www.nortelnetworks.com/`).

- If the IPSec client does not support *NAT Traversal*, it is necessary to *enable* IPSec pass-through in *WinRoute*.

In both cases, IPSec communication between the client and the IPSec server must be permitted by a traffic rule. NAT must be defined in the *Translation* column (in the same way as for the communication from the local network to the Internet).

3. *Multiple IPSec clients in the local network (multiple tunnels)*

   If multiple IPSec tunnels from the local network to the Internet are supposed to be created, all IPSec clients and corresponding servers must support *NAT Traversal* (see above). Support for IPSec in *WinRoute* must be *disabled* so that no collisions arise.

   Again, traffic between the local network and corresponding IPSec servers must be permitted by a traffic rule.

IPSec server in local network

An IPSec server on a host in the local network or on the *WinRoute* host must be mapped from the Internet. In this case, traffic between Internet clients and the *WinRoute* host must be permitted by a traffic rule and mapping to a corresponding host in the local network must be set.

Warning: Only a single IPSec server can be mapped from the public IP address of the firewall. For mapping of multiple IPSec servers, the firewall must use multiple public IP addresses.

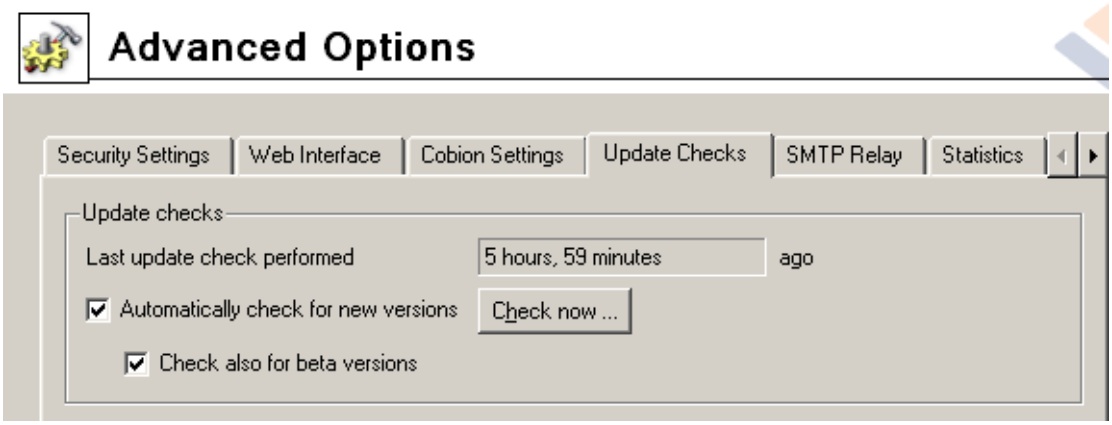
Example: We want to set that two IPSec servers will be available from the Internet — one on the *WinRoute* host and another on a host with the IP address `192.168.100.100`. The firewall interface connected to the Internet uses IP addresses `60.80.100.120` and `60.80.100.121`.

Name	Source	Destination	Service	Action	Translation
IPSec server #1	Clients of server #1	60.80.100.120	IKE IPSec	✓	
IPSec server #2	Clients of server #2	60.80.100.121	IKE IPSec	✓	MAP 192.168.1.50

11.8 Update Checking

WinRoute enables automatic check for new versions at the *Kerio Technologies* website. Whenever a new version is detected, is download and installation is offered.

Open the *Update Checking* tab in the *Configuration / Advanced Options* section to view information on a new version and to set parameters for automatic checks for new versions.



Last update check performed ... ago Information on how much time ago the last update check was performed.

If the time is too long (several days) this may indicate that the automatic update checks fail for some reason (i.e. access to the update server is blocked by a traffic rule). In such cases we recommend you to perform a check by hand (by clicking on the *Check now* button), view results in the *Debug* log (see chapter 16.6) and take appropriate actions.

Check for new versions Use this option to enable/disable automatic checks for new versions. Checks are performed:

- 2 minutes after each startup of the *WinRoute Firewall Engine*
- every 24 hours

Results of each attempted update check (succesful or not) is logged into the *Debug* log (see chapter 16.6).

Check also for beta versions Enable this option if you want *WinRoute* to perform also update checks for beta versions.

If you wish to participate in testing of *WinRoute* betaversions, enable this option. In case that you use *WinRoute* in operations in your company (i.e. at the Internet gateway of your company), we recommend you not to use this option (betaversions are not tested yet and they could endanger functionality of your networks, etc.).

Check now Click on this button to check for updates immediately. If no new version is available, user will be informed about this fact.

11.9 SMTP Relay

WinRoute provides a function which enables notification to users or/and administrators by email alerts. These alert messages can be sent upon various events, for example when a virus is detected (see chapter 7.2), when a *peer-to-peer* network is detected (refer to chapter 11.10), when an alert function is set for certain events (details in chapter 10.1) or upon reception of a notification (see chapter 14.3).

For this purpose, *WinRoute* needs an SMTP Relay Server. This server is used for forwarding of infected messages to a specified address.

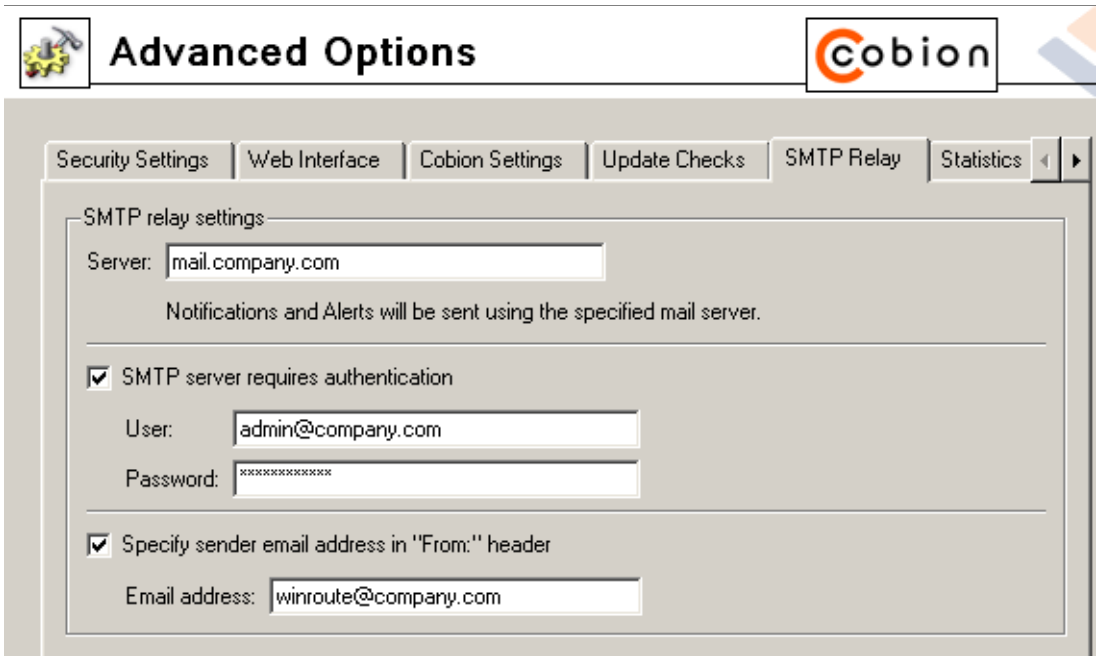
Note: WinRoute does not provided any built-in SMTP server.

To configure an SMTP server, go to the *SMTP server* tab in *Configuration / Advanced Options*.

Server Name or IP address of the server.

Note: If available, we recommend you to use an SMTP server within the local network (messages sent by *WinRoute* are often addressed to local users).

SMTP requires authentication Enable this option to require authentication through username and password at the specified SMTP server.



Specify sender email address in "From" header In this option you can specify a sender's email address (i.e. the value for the `From` header) for email sent from *WinRoute* (email or SMS notifications sent to users). Preset `From` header does not apply to messages forwarded during antivirus check (refer to chapter 7.3).

This item must be preset especially if the SMTP server strictly checks the header (messages without or with an invalid `From` header are considered as spams). The item can also be used for reference in recipient's mail client or for email classification. This is why it is always recommended to specify sender's email address in *WinRoute*.

Warnings:

1. If SMTP is specified by a DNS name, it cannot be used until *WinRoute* resolves a corresponding IP address (by a DNS query). The *IP address of specified SMTP server cannot be resolved* warning message is displayed in the *SMTP Relay* tab until the IP address is not found.

   If the warning is still displayed, this implies that an invalid (non-existent) DNS name is specified or the DNS server does not respond. Therefore, we recommend you to specify SMTP server by an IP address if possible.

2. Communication with the SMTP server must not be blocked by any rule, otherwise the *Connection to SMTP server is blocked by traffic rules* error is reported upon clicking the *Apply* button.

   For detailed information about traffic rules, refer to chapter 5.

11.10 P2P Eliminator

Peer-to-peer (*P2P*) networks are world-wide distributed systems, where each node can represent both a client and a server. These networks are used for sharing big volumes of data (mostly used for illegal data sharing). There are many similar networks, such as *DirectConnect*, *Kazaa*, etc.

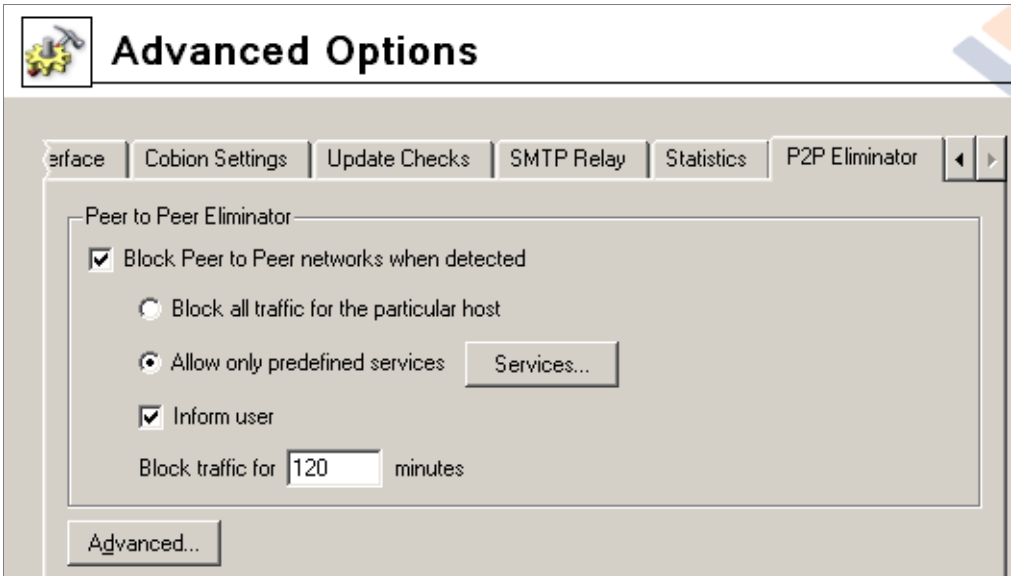
In addition to illegal data distribution, utilization of *P2P* networks overload lines via which users are connected to the Internet. Such users may limit connections of other users in the same network and may increase costs for the line (for example when volume of transmitted data is limited for the line).

WinRoute provides the *P2P Eliminator* module which detects connections to *P2P* networks and applies specific restrictions. Since there is a large variety of *P2P* networks and parameters at individual nodes (servers, number of connections, etc.) can be changed, it is hardly possible to detect all *P2P* connections. However, using various methods (such as known ports, established connections, etc.), the *P2P Eliminator* is able to detect whether a user connects to one or multiple *P2P* networks.

Note: According to thorough tests, the detection is highly reliable (probability of failure is very low).

P2P Eliminator Configuration

To configure the *P2P Eliminator* module, go to the *P2P Eliminator* tab in the *Configuration / Advanced Options* section.



The *Block P2P networks when detected* option enables *P2P Eliminator*.

As implied by the previous description, it is not possible to block connections to particular *P2P* networks. *P2P Eliminator* blocks connection to the Internet from particular hosts (*Block all traffic for the particular user*) or allow these users to connect to certain services only *Allow only predefined services*).

Use the *Services* button to open a dialog where services which will be allowed can be specified. All services defined in *Configuration / Definitions / Services* are available (for details, refer to chapter 9.3).

Use the *Block traffic for ... minutes* parameter to specify the length of time during which traffic will be blocked for the particular host. The *P2P Eliminator* module enables traffic for this user automatically when the specified time expires. The time of disconnection should be long enough to make the user consider consequences and to stop trying to connect to *peer-to-peer* networks.

Check the *Inform user* option if you wish that users at whose hosts *P2P* networks are detected will be warned and informed about actions to be taken (blocking of all traffic / time-limited restrictions for certain services and length of the period for which restrictions will be applied). This option does not apply to unauthenticated users.

Notes:

1. If a user who is allowed to use *P2P* networks (see chapter 10.1) is connected to the firewall from a certain host, no *P2P* restrictions are applied to this host. Settings in the *P2P Eliminator* tab are always applied to unauthorized users.

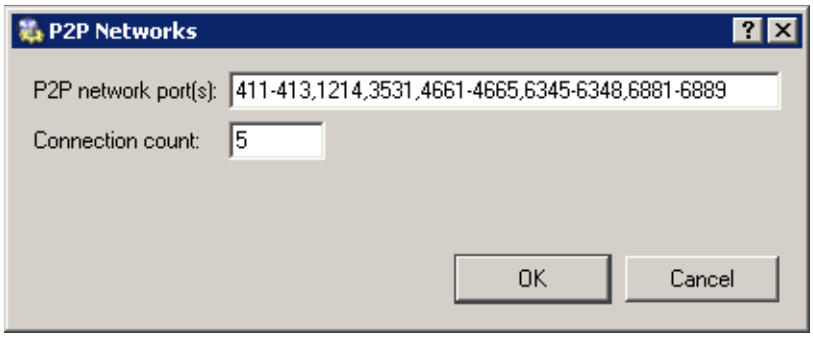
2. Information about P2P detection and blocked traffic can be viewed in the *Status / Hosts/users* section (for details, refer to chapter 14.1).

3. If you wish to notify also another person when a *P2P* network is detected (e.g. the *WinRoute* administrator), define the alert in the *Configuration / Logs &Alerts* section (for details, see chapter 14.3).

Parameters for detection of P2P networks

Click *Advanced* to set parameters for *P2P* detection:

- *P2P network port(s)* — list of ports which are exclusively used by *P2P* networks. These ports are usually ports for control connections — ports (port ranges) for data sharing can be set by users themselves.



You can use the *P2P network port(s)* entry to specify ports or port ranges. Use comas to separate individual values.

- *Connection count* — minimal number of concurrent connections which the user must reach to run *P2P* networks detection.

  Big volume of established connections is a typical feature of *P2P* networks (usually one connection for each file).

  The optimum value depends on circumstances (type of user's work, frequently used network applications, etc.) and it must be tested. If the value is too low, the system can be unreliable (users who do not use *P2P* networks might be suspected). If the value is too high, reliability of the detection is decreased (less *P2P* networks are detected).

Chapter 12

Kerio VPN

WinRoute enables secure interconnection of remote private networks using an encrypted tunnel and it provides clients secure access to their local networks via the Internet. This method of interconnection of networks (and of access of remote clients to local networks) is called virtual private network (*VPN*). *WinRoute* includes a proprietary implementation of VPN, called *Kerio VPN*.

Kerio VPN is designed so that it can be used simultaneously with the firewall and with NAT (even along with multiple translations). Creation of an encrypted tunnel between networks and setting remote access of clients at the server is very easy.

Kerio VPN enables creation of any number of encrypted *server-to-server* connections (i.e. tunnels to remote private networks). Tunnels are created between two *WinRoutes* (typically at Internet gateways of corresponding networks). Individual servers (endpoints of the tunnels) verify each other using SSL certificates — this ensures that tunnels will be created between trustworthy servers only.

Individual hosts can also connect to the VPN server in *WinRoute* (secured *client-to-server* connections). Identities of individual clients are authenticated against a username and password (transmitted also by secured connection), so that unauthorized clients cannot connect to local networks.

Remote connections of clients are performed through *Kerio VPN Client*, included in *WinRoute* (for a detailed description, view the stand-alone *Kerio VPN Client — User Guide* document).

Note: For deployment of the *Kerio VPN*, it is supposed that *WinRoute* is installed at a host which is used as an Internet gateway. If this condition is not met, *Kerio VPN* can also be used, but the configuration can be quite complicated.

Benefits of Kerio VPN

In comparison with other products providing secure interconnection of networks via the Internet, the *Kerio VPN* solution provides several benefits and additional features.

- Easy configuration (only a few basic parameters are required for creation of tunnels and for configuration of servers which clients will connect to).
- No additional software is required for creation of new tunnels (*Kerio VPN Client* must be installed at remote clients — installation file of the application is 3 MB).

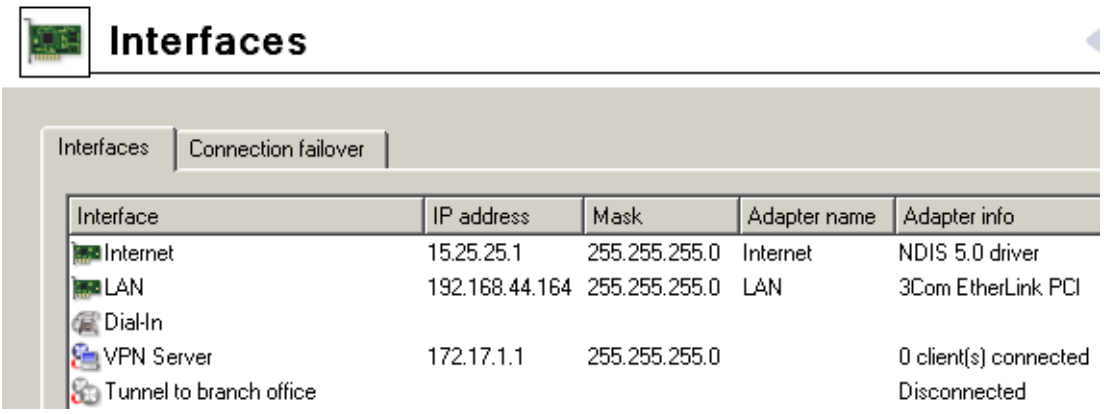
- No collisions arise while encrypted channels through the firewall are being created. It is supposed that one or multiple firewalls (with or without NAT) are used between connected networks (or between remote clients and local networks).
- No special user accounts must be created for VPN clients. User accounts in *WinRoute* (or domain accounts if the *Active Directory* is used — see chapter 8.2) are used for authentication.
- Statistics about VPN tunnels and VPN clients can be viewed in *WinRoute* (refer to chapter 15.4).

12.1 VPN server configuration

VPN server is used for connection of remote endpoints of VPN tunnels and of remote clients using *Kerio VPN Client*.

Note: Connection to the VPN server from the Internet must be first allowed by traffic rules. For detailed information, refer to chapters 12.2 and 12.4.

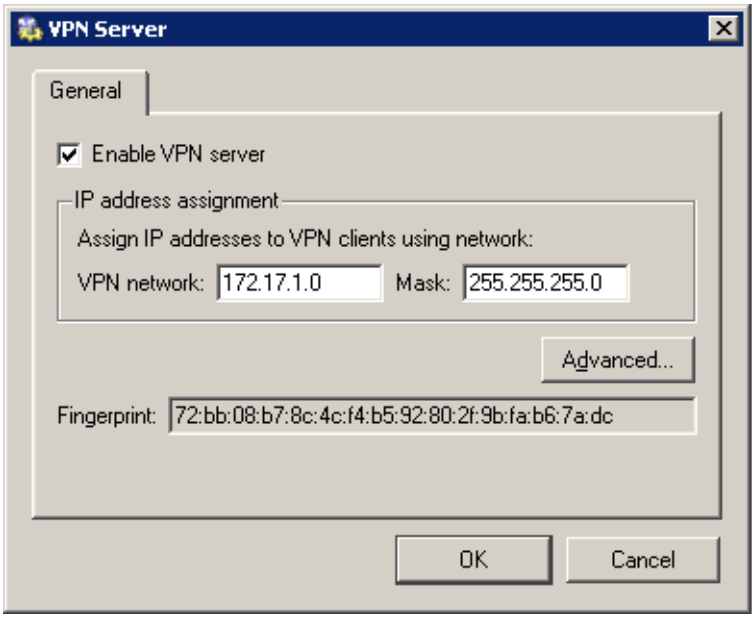
VPN server is available in the *Interfaces* tab of the *Configuration / Interfaces* section as a special interface.



Double-click on the *VPN server* interface (or select the alternative and press *Edit*, or select *Edit* from the context menu) to open a dialog where parameters of the VPN server can be set.

Enable VPN server Use this option to enable/disable VPN server. VPN server uses TCP and UDP protocols, port 4090 is used as default (the port can be changed in advanced options, however, it is usually not necessary to change it). If the VPN server is not used, it is recommended to disable it.

The action will be applied upon clicking the *Apply* button in the *Interfaces* tab.

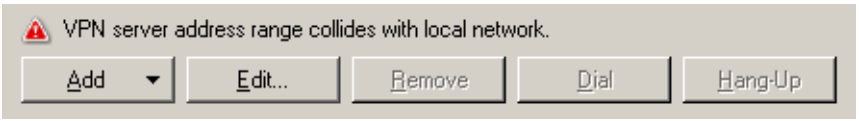


IP address assignment Specification of a subnet (i.e. IP address and a corresponding network mask) from which IP addresses will be assigned to VPN clients and to remote endpoints of VPN tunnels which connect to the server (all clients will be connected through this subnet).

By default (upon the first start-up after installation), *WinRoute* automatically selects a free subnet which will be used for VPN. Under usual circumstances, it is not necessary to change the default subnet.

Warning: Make sure that the subnet for VPN clients does not collide with any local subnet!

WinRoute can detect a collision of the VPN subnet with local subnets. The collision may arise when configuration of a local network is changed (change of IP addresses, addition of a new subnet, etc.), or when a subnet for VPN is not selected carefully. If the VPN subnet collides with a local network, a warning message is displayed upon saving of the settings (by clicking *Apply* in the *Interfaces* tab). In such cases, redefine the VPN subnet.



It is recommended to check whether IP collision is not reported after each change in configuration of the local network or/and of the VPN!

Notes:

1. Under certain circumstances, collision with the local network might also arise when a VPN subnet is set automatically (if configuration of the local network is changed later).

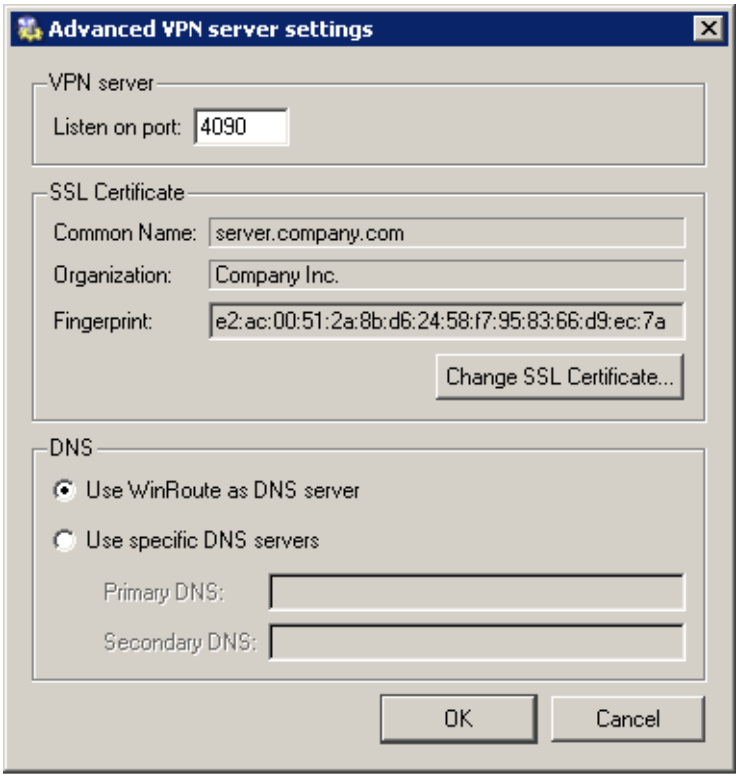
2. Regarding two VPN tunnels, it is also examined when establishing a connection whether the VPN subnet does not collide with IP ranges at the other end of the tunnel (remote endpoint).

   If a collision with an IP range is reported upon a change of configuration of the VPN server (upon clicking *Apply* in the *Interfaces* tab), the VPN subnet must be set by hand. Select a network which is not used by any of the local networks participating in the connection. VPN subnets at each end of the tunnel must not be identical (two free subnets must be selected).

Fingerprint Hexadecimal fingerprint of the current VPN server certificate. This fingerprint is required for definition of VPN tunnels (see chapter 12.4).

TIP: Certificate fingerprint can be saved to the clipboard and pasted to a text file, email message, etc.

Click on the *Advanced* button to set advanced parameters of the VPN server.



Listen on port The port on which the VPN server listens for incoming connections (both TCP and UDP protocols are used). The port 4090 is set as default (under usual circumstances it is not necessary to switch to another port).

Notes:

1. If the VPN server is already running, all VPN clients will be automatically disconnected during the port change.

2. If it is not possible to run the VPN server at the specified port (the port is used by another service), the following error will be reported in the *Error* log (see chapter 16.8) upon clicking on the *Apply* button:

```
(4103:10048) Socket error:  Unable to bind socket
  for service to port 4090.
(5002) Failed to start service "VPN"
  bound to address 192.168.1.1.
```

   To make sure that the specified port is really free, view the *Error* log to see whether an error of this type has not been reported.

SSL certificate Information about the current VPN server certificate. This certificate is used for verification of the server's identity during creation of a VPN tunnel (for details, refer to chapter 12.4). The VPN server in *WinRoute* uses the standard SSL certificate.

Click *Change SSL Certificate* to set parameters for the certificate of the VPN server. For the VPN server, you can either create a custom (self-subscribed) certificate or import a certificate created by a certification authority. Methods used for creation and import of SSL certificates are described thoroughly in chapter 8.1.

Note: If you already have a certificate created by a certification authority especially for your server (for secured Web interface), it is also possible to use it for the VPN server — it is not necessary to apply for a new certificate.

DNS Specify a DNS server which will be used for VPN clients:

- *Use WinRoute as DNS server* — IP address of a corresponding interface of *WinRoute* host will be used as a DNS server for VPN clients (VPN clients will use the *DNS forwarder*).

  If the *DNS Forwarder* is already used as a DNS server for local hosts, it is recommended to use it also for VPN clients. The *DNS forwarder* provides the fastest responses to client DNS requests and possible collision (inconsistency) of DNS records will be avoided.

Note: If the *DNS forwarder* is disabled (refer to chapter 4.3), the option is not available.

- *Use specific DNS servers* — primary and secondary DNS servers specified through this option will be set for VPN clients.

  If another DNS server than the *DNS forwarder* in *WinRoute* is used in the local network, use this option.

12.2 Configuration of VPN clients

The following conditions must be met to enable connection of remote clients to local networks via encrypted channels:

- The *Kerio VPN Client* must be installed at remote clients (for detailed description, refer to a stand-alone document, *Kerio VPN Client — User Guide*).
- Users whose accounts are used for authentication to *Kerio VPN Client* must possess rights enabling them connect to the VPN server in *WinRoute* (see chapter 10.1).
- Connection to the VPN server from the Internet as well as communication between VPN clients must be allowed by traffic rules.

Note: Remote VPN clients connecting to*WinRoute* are included toward the number of persons using the license (see chapters 13 and 13.4). Be aware of this fact when deciding what license type should be bought (or whether an upgrade to a higher number of users should be bought).

Basic configuration of traffic rules for VPN clients

Name	Source	Destination	Service	Action
VPN server connections	Internet	Firewall	Kerio VPN	✓
Local traffic	Firewall LAN VPN clients	Firewall LAN VPN clients	Any	✓

- The first rule allows connection to the VPN server in *WinRoute* from the Internet.

  To restrict the number of IP addresses from which connection to the VPN server will be allowed, edit the *Source* entry.

By default, the *Kerio VPN* service is defined for TCP and UDP protocols, port 4090. If the VPN server is running at another port, this service must be redefined.

- The second rule allows communication between the firewall, local network and VPN clients.

If the rules are set like this, all VPN clients can access local networks and vice versa (all local hosts can communicate with all VPN clients). To restrict the type of network access available to VPN clients, special rules must be defined. Examples of traffic rules are provided in chapter 12.5.

Notes:

1. If the *Network Rules Wizard* is used to create traffic rules, the described rules can be generated automatically (including matching of VPN clients with the *Source* and *Destination* items). To generate the rules automatically, select *Yes, I want to use Kerio VPN* in Step 5. For details, see chapter 5.1.

2. For access to the Internet, VPN clients use their current Internet connections. VPN clients are not allowed to connect to the Internet via *WinRoute* (configuration of default gateway of clients cannot be defined).

3. For detailed description on how to define traffic rules, refer to chapter 5.

12.3 Exchange of routing information

An automatic exchange of routing information (i.e. of data informing about routes to local subnets) is performed between endpoints of any VPN tunnel (or between the VPN server and a VPN client). This implies that routing tables at both sides are always up-to-date (even when any changes are performed in configuration at any side of the tunnel). No manual routing is necessary if both sides (networks) use *WinRoute* as their default gateways.

Note: Regarding VPN clients, routing information is sent only from the server to a corresponding client.

Exchanged routes

The following rules are applied to the exchange of routing information:

- default routes as well as routes to networks with default gateways are not exchanged (default gateway cannot be changed for remote clients and/or for remote endpoints of a tunnel),

- routes to subnets which are identical for both sides of a tunnel are not exchanged (routing of local and remote networks with identical IP ranges is not allowed).
- other routes (i.e. routes to local subnets at remote ends of VPN tunnels excluding the cases described above, all other VPN and all VPN clients) are exchanged.

Note: As implied from the description provided above, if two VPN tunnels are created, communication between these two networks is possible. The traffic rules can be configured so that connection to the local network will be disabled for both these remote networks.

Update of routing tables

Routing information is exchanged:

- when a VPN tunnel is connected or when a VPN client is connected to the server,
- when information in a routing table at any side of the tunnel (or at the VPN server) is changed,
- periodically, once per 30 secs (VPN tunnel) or once per 1 min (VPN server). The timeout starts upon each update (regardless of the update reason).

12.4 Interconnection of two private networks via the Internet (VPN tunnel)

WinRoute (version *6.0.0* or later) including support for VPN (VPN support is included in the typical installation — see chapter 2.3) must be installed in both networks to enable creation of an encrypted tunnel between a local and a remote network via the Internet ("VPN tunnel").

Note: Each installation of *WinRoute* requires its own license (see chapter 13).

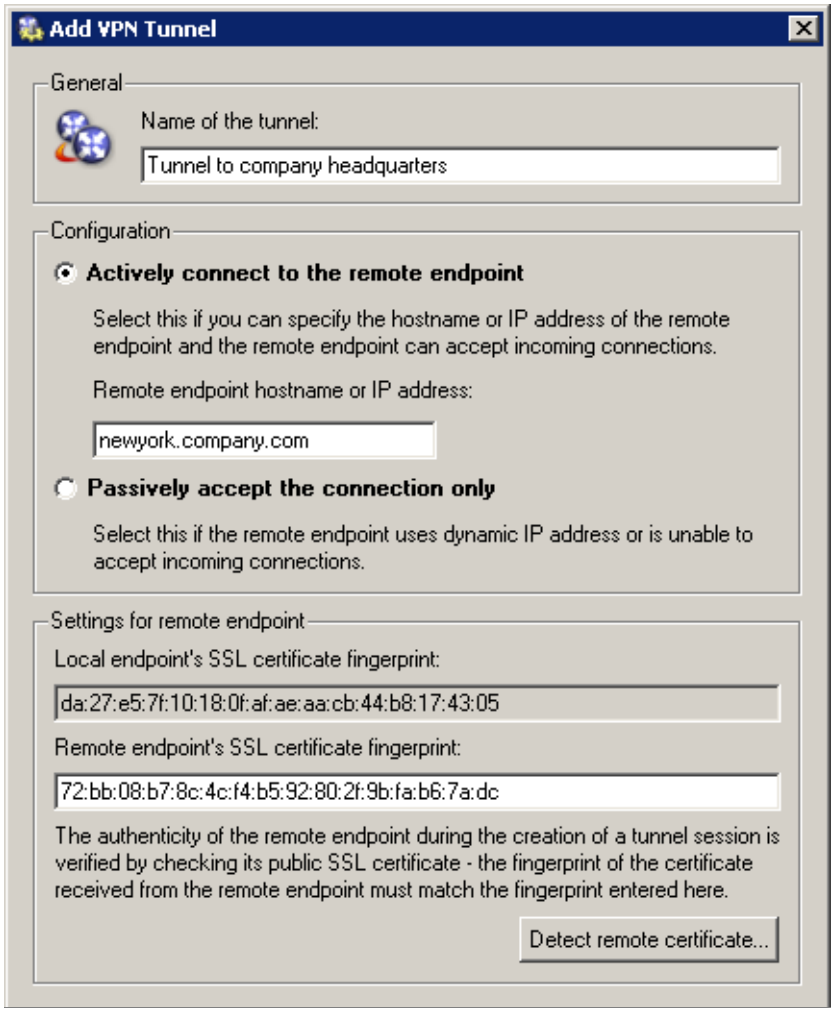
Setting up VPN servers

First, the VPN server must be allowed by the traffic policy and enabled at both ends of the tunnel. For detailed description on configuration of VPN servers, refer to chapter 12.1.

Definition of a tunnel to a remote server

VPN tunnel to the server on the other side must be defined at both ends. Use the *Add / VPN tunnel* option in the *Interfaces* section to create a new tunnel.

12.4 Interconnection of two private networks via the Internet (VPN tunnel)



Name of the tunnel Each VPN tunnel must have a unique name. This name will be used in the table of interfaces, in traffic rules (see chapter 5.2) and interface statistics (details in chapter 15.4).

Configuration Selection of a mode for the local end of the tunnel:

- *Active* — this side of the tunnel will automatically attempt to establish and maintain a connection to the remote VPN server.

  The remote VPN server specification is required through the *Remote hostname or IP address* entry. If the remote VPN server does not use the port 4090, a corresponding port number separated by a colon must be specified (e.g. `server.company.com:4100` or `10.10.100.20:9000`).

This mode is available if the IP address or DNS name of the other side of the tunnel is known and the remote endpoint is allowed to accept incoming connections (i.e. the communication is not blocked by a firewall at the remote end of the tunnel).

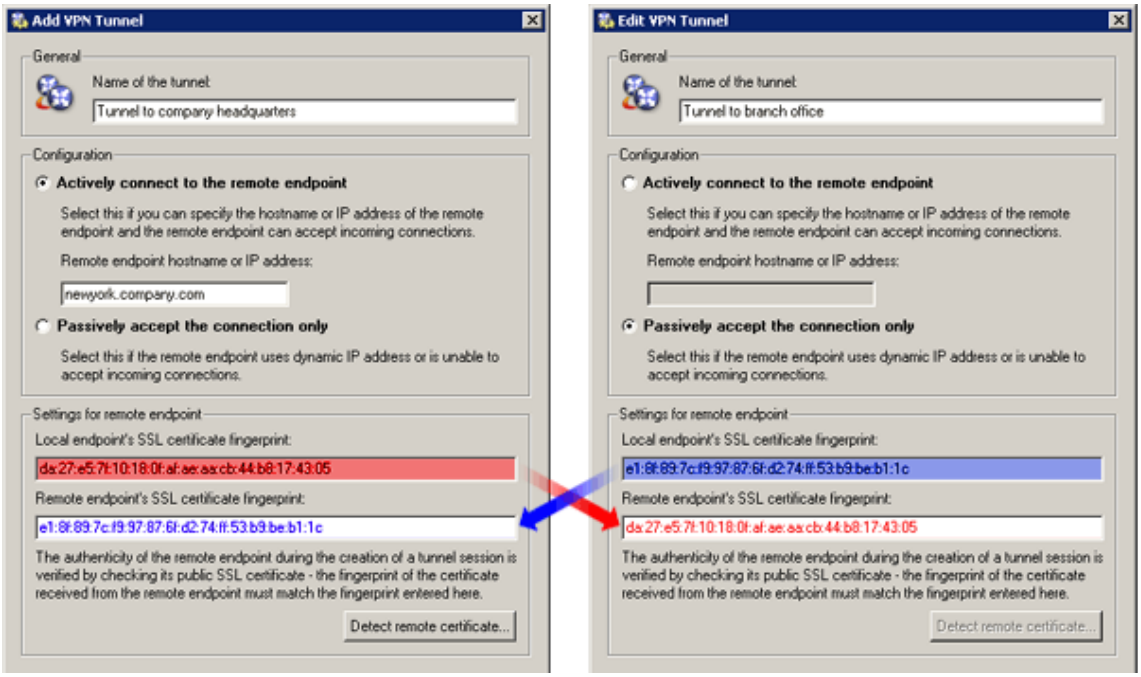
- *Passive* — this end of the tunnel will only listen for an incoming connection from the remote (active) side.

  The passive mode is only useful when the local end of the tunnel has a fixed IP address and when it is allowed to accept incoming connections.

At least one end of each VPN tunnel must be switched to the active mode (passive servers cannot initialize connection).

Configuration of a remote end of the tunnel When a VPN tunnel is being created, identity of the remote endpoint is authenticated through the fingerprint of its SSL certificate. If the fingerprint does not match with the fingerprint specified in the configuration of the tunnel, the connection will be rejected.

The fingerprint of the local certificate and the entry for specification of the remote fingerprint are provided in the *Settings for remote endpoint* section. Specify the fingerprint for the remote VPN server certificate and vice versa — specify the fingerprint of the local server in the configuration at the remote server.



If the local endpoint is set to the active mode, the certificate of the remote endpoint and its fingerprint can be downloaded by clicking *Detect remote certificate*.

If the local VPN server endpoint is configured to passively accept connections, it is not possible to automatically obtain the remote certificate fingerprint. It must be obtained manually from the remote VPN server.

DNS Settings

DNS must be set properly at both sends of the tunnel so that it is possible to connect to hosts in the remote network using their DNS names. One method is to add DNS records of the hosts (to the hosts file) at each endpoint. However, this method is quite complicated and inflexible.

If the *DNS forwarder* in *WinRoute* is used as the DNS server at both ends of the tunnel, DNS queries (for DNS rules, refer to chapter 4.3) can be forwarded to hostnames in the corresponding domain of the *DNS forwarder* at the other end of the tunnel.

Detailed guidance for the DNS configuration is provided in chapter 12.5.

Connection establishment

Active endpoints automatically attempt to recover connection whenever they detect that the corresponding tunnel has been disconnected (the first, connection establishment is attempted immediately after the tunnel is defined and upon clicking the *Apply* button in *Configuration / Interfaces*).

VPN tunnels can be disabled by the *Disable* button. Both ends of a selected tunnel should be automatically disabled (regardless of whether they are active or passive).

Note: VPN tunnels keeps their connection (by sending special packets in regular time intervals) even if no data is transmitted. This feature protects tunnels from disconnection by other firewalls or network devices between ends of tunnels.

VPN — Traffic Policy

When a VPN tunnel is created (see chapter 12.4), communication between the local network and the network connected via this tunnel must be allowed by traffic rules. If basic traffic rules are already created by the wizard (refer to chapter 12.2), simply add a corresponding VPN tunnel into the *Local Traffic* rule.

Note: Traffic rules set by this method allow full IP communication between the local network, remote network and all VPN clients. For access restrictions, define corresponding traffic rules (for local traffic, VPN clients, VPN tunnel, etc.). Examples can be found in chapter 12.5.

Name	Source	Destination	Service	Action
VPN server connections	Internet	Firewall	Kerio VPN	✓
Local traffic	Firewall LAN Tunnel to branch office VPN clients	Firewall LAN Tunnel to branch office VPN clients	Any	✓

12.5 Configuration of the VPN tunnel — Example

This chapter provides a detailed exemplary description on how to create an encrypted tunnel connecting two private networks using the *Kerio VPN*. This example can be easily customized.

Note: This example describes a more complicated pattern of VPN with access restrictions for individual local networks and VPN clients. An example of basic VPN configuration is provided in the *Kerio WinRoute Firewall — Step By Step Configuration* document.

Specification

Supposing a company has its headquarters in New York and a branch office in Chicago. We intend to interconnect local networks of the headquarters by a VPN tunnel using the *Kerio VPN*. VPN clients will be allowed to connect to the headquarters network.

The server (default gateway) of the headquarters uses the public IP address `63.55.21.12` (DNS name is `newyork.company.com`), the server of the branch office uses a dynamic IP address assigned by DHCP.

The local network of the headquarters consists of two subnets, `LAN 1` and `LAN 2`. The headquarters uses the `company.com` DNS domain.

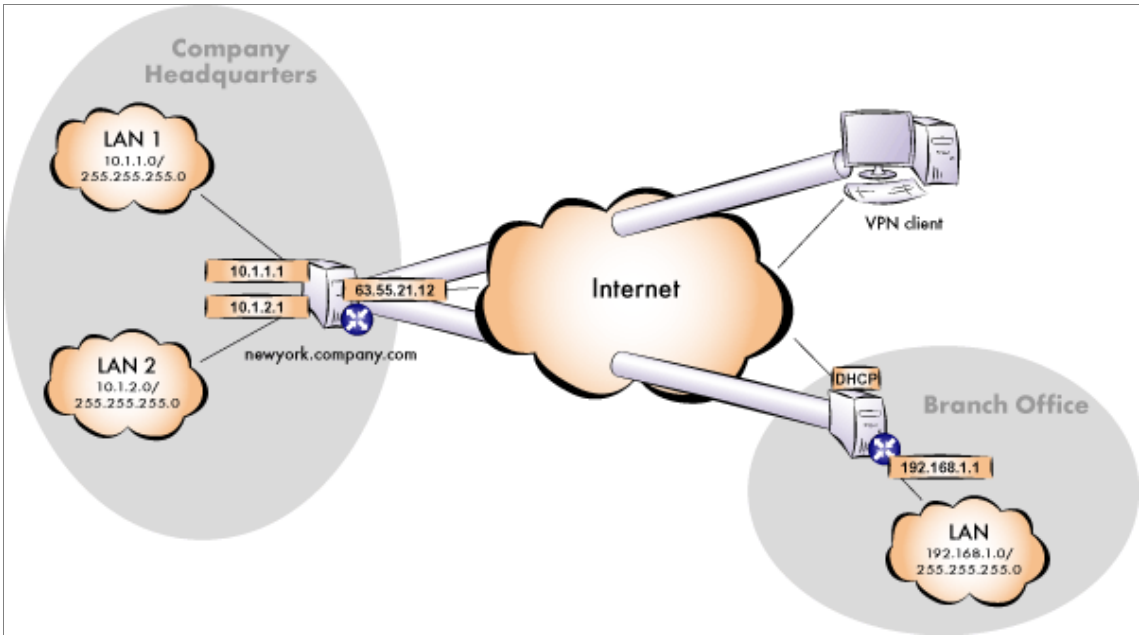
The network of the branch office consists of one subnet only (`LAN`). The branch office `filial.company.com`.

The following figure provides a scheme of the entire system, including IP addresses and the VPN tunnels that will be built.

Suppose that both networks are already deployed and set according to the figure and that the Internet connection is available.

Traffic between the network of the headquarters, the network of the branch office and VPN clients will be restricted according to the following rules:

1. VPN clients can connect to the `LAN 1` and to the network of the branch office.
2. Connection to VPN clients is disabled for all networks.



3. Only the `LAN 1` network is available from the branch office. In addition to this, only the *WWW*, *FTP* and *Microsoft SQL* services are available.

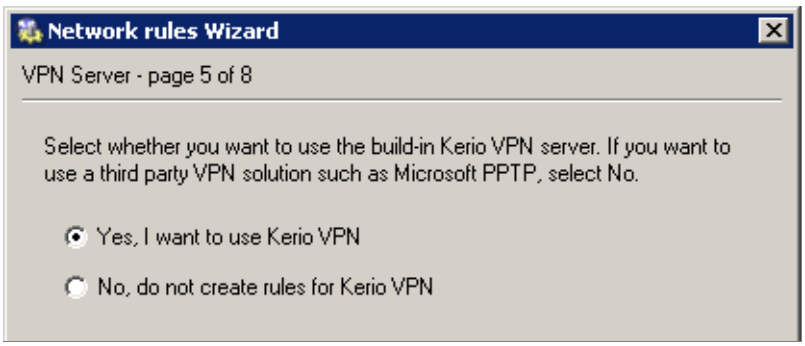
4. No restrictions are applied for connections from the headquarters to the branch office network.

5. `LAN 2` is not available to the branch office network nor to VPN clients.

Headquarters configuration

1. Install *WinRoute* (version `6.0.0` or later) at the headquarter's default gateway ("server").

2. Use *Network Rules Wizard* (see chapter 5.1) to configure the basic traffic policy in *WinRoute*. In step 5, select *Yes, I want to use Kerio VPN*.



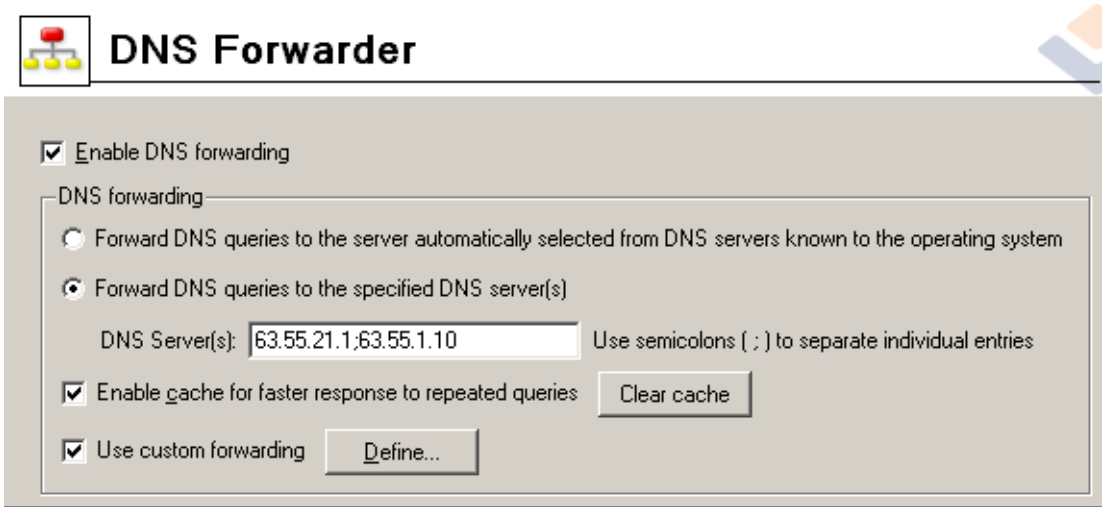
This step will create rules for connection of the VPN server as well as for communication of VPN clients with the local network (with the firewall).

Name	Source	Destination	Service	Action	Translation
Local Traffic	Dial-In Firewall LAN 1 LAN 2 VPN clients	Dial-In Firewall LAN 1 LAN 2 VPN clients	Any	✓	
Service Kerio VPN	Internet	Firewall	Kerio VPN	✓	

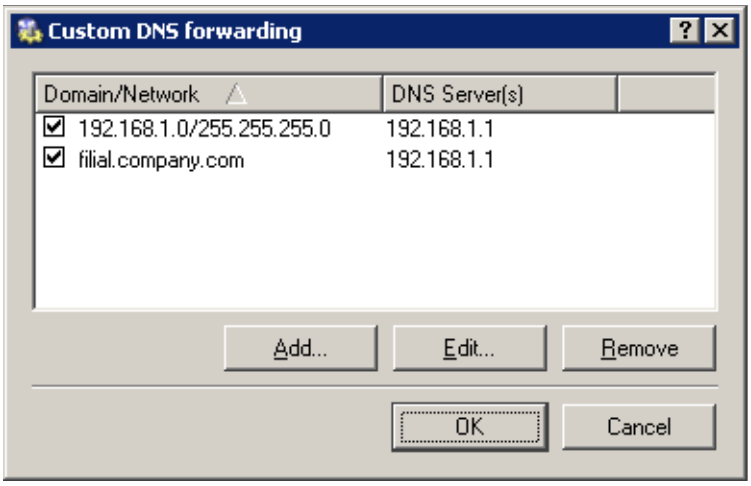
When the VPN tunnel is created, customize these rules according to the restriction requirements (Step 6).

3. Customize DNS configuration as follows:

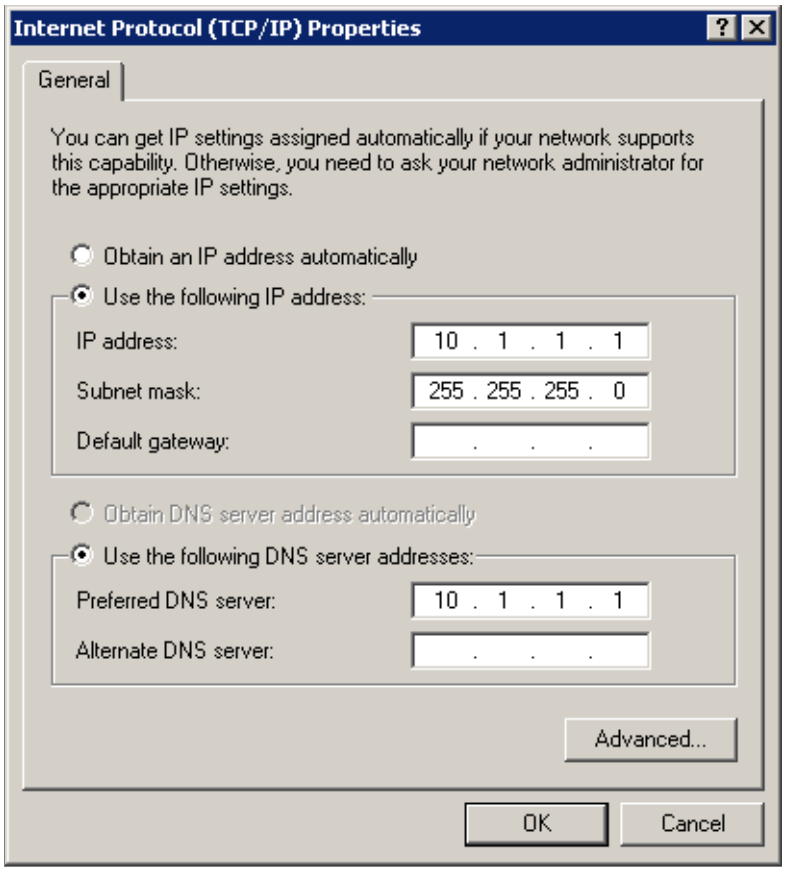
   - In configuration of the *DNS Forwarder* in *WinRoute*, specify DNS servers to which DNS queries which are not addressed to the `company.com` domain will be forwarded (primary and secondary DNS server of the Internet connection provider by default).



   - Enable the *Use custom forwarding* option and define rules for the `filial.company.com` domain. To specify the forwarding DNS server, use the IP address of the remote *WinRoute* host's interface connected to the local network.



- Set the IP address of this interface (`10.1.1.1`) as a primary DNS server for the *WinRoute* host's interface connected to the local network.

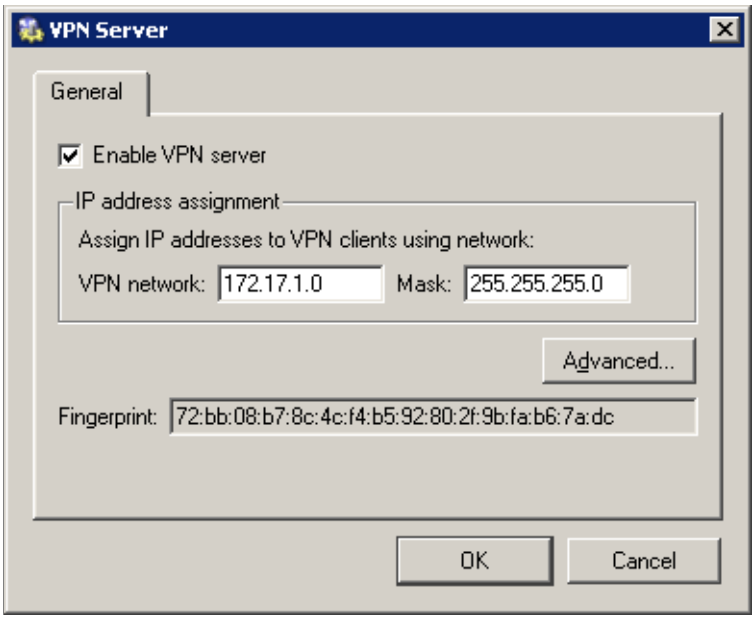


- Set the IP address `10.1.1.1` as a primary DNS server also for the other hosts.

Note: For proper functionality of DNS, the DNS database must include records for hosts in a corresponding local network. To achieve this, save DNS names and IP addresses of local hosts into the `hosts` file (if they use IP addresses) or enable co-operation of the *DNS Forwarder* with the DHCP server (in case that IP addresses are assigned dynamically to these hosts). For details, refer to chapter 4.3.

4. Enable the VPN server and configure its SSL certificate (create a self-signed certificate if no certificate provided by a certification authority is available).

   Note: A free subnet which has been selected is now specified automatically in the *VPN network* and *Mask* entries.



   For a detailed description on the VPN server configuration, refer to chapter 12.1.

5. Create a passive end of the VPN tunnel (server of the branch office uses a dynamic IP address). Use the fingerprint of the VPN server of the branch office as a specification of the fingerprint of the remote SSL certificate.

6. Customize traffic rules according to the restriction requirements.

   - In the *Local Traffic* rule, remove all items except those belonging to the local network of the company headquarters, i.e. except the firewall and `LAN 1` and `LAN 2`.

   - Define (add) the *VPN clients* rule which will allow VPN clients to connect to `LAN 1` and to the network of the branch office (via the VPN tunnel).

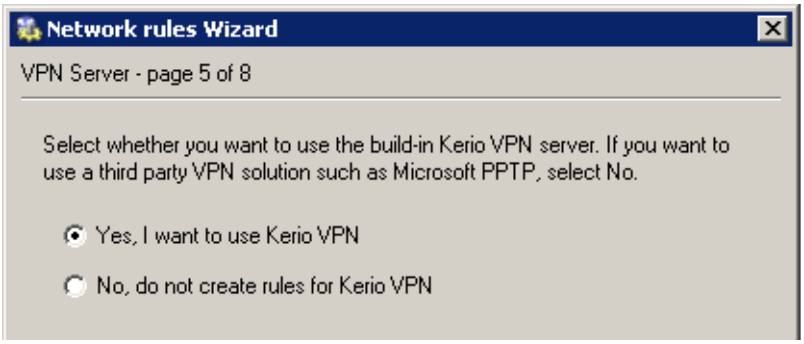
Name	Source	Destination	Service	Action
Local Traffic	Firewall LAN 1 LAN 2	Firewall LAN 1 LAN 2	Any	✓
VPN clients	VPN clients	LAN 1 Tunnel to branch office	Any	✓
Branch office	Tunnel to branch office	LAN 1	FTP HTTP MS-SQL	✓
Company headquarters	LAN 1 LAN 2	Tunnel to branch office	Any	
Service Kerio VPN	Internet	Firewall	Kerio VPN	✓

- Create the *Branch office* rule which will allow connections to services in `LAN 1`.
- Add the *Company headquarters* rule allowing connections from both headquarters subnets to the branch office network..

Rules defined this way meet all the restriction requirements. Traffic which will not match any of these rules will be blocked by the default rule (see chapter 5.2).

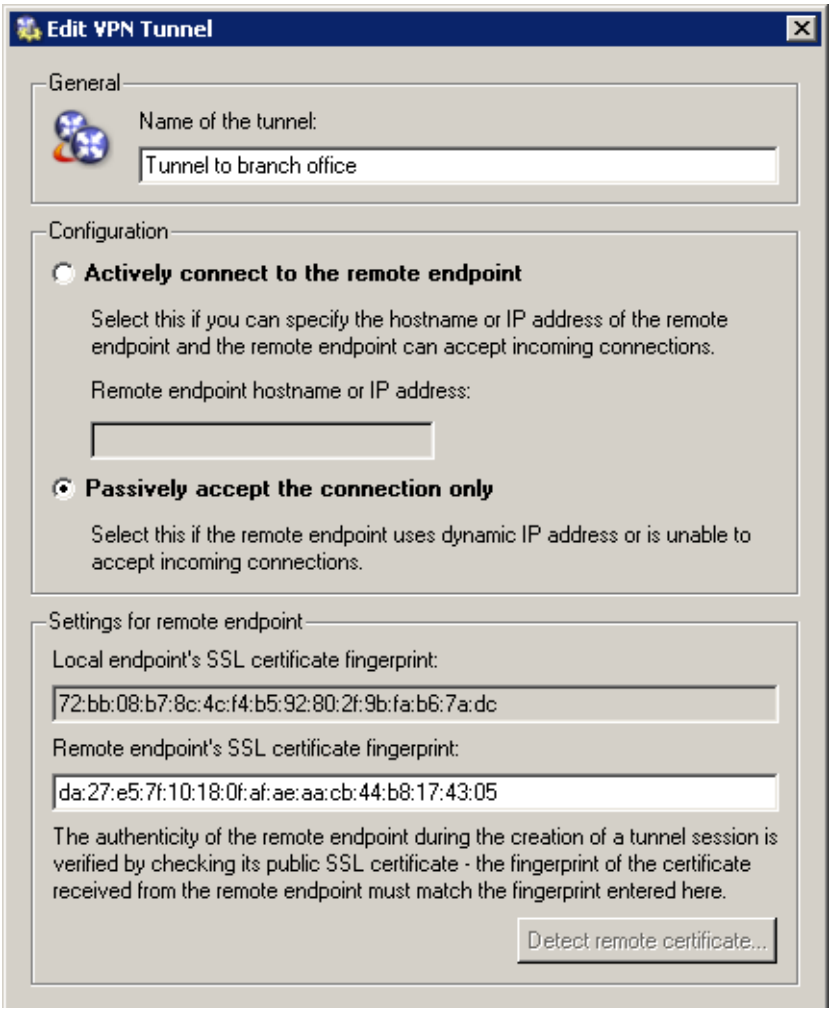
Branch office configuration

1. Install *WinRoute* (version `6.0.0` or later) at the default gateway of the branch office ("server").
2. Use *Network Rules Wizard* (see chapter 5.1) to configure a basic traffic policy in *WinRoute*. In step 5, select *Yes, I want to use Kerio VPN*.



This step will create rules for connection of the VPN server as well as for communication of VPN clients with the local network (through the firewall).

Name	Source	Destination	Service	Action	Translation
Local Traffic	Dial-In Firewall LAN VPN clients	Dial-In Firewall LAN VPN clients	Any	✓	
Service Kerio VPN	Internet	Firewall	Kerio VPN	✓	

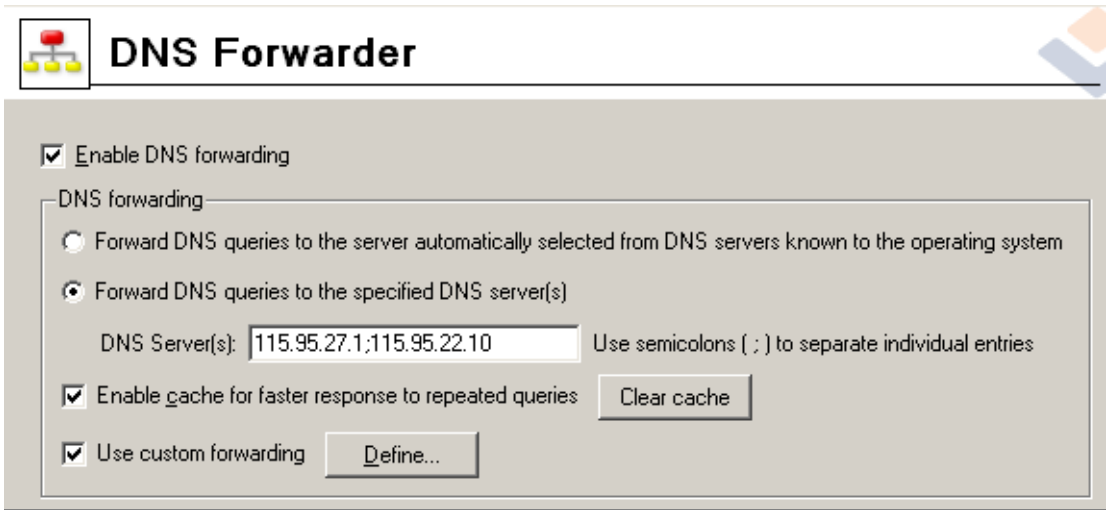


When the VPN tunnel is created, customize these rules according to the restriction requirements (Step 6).

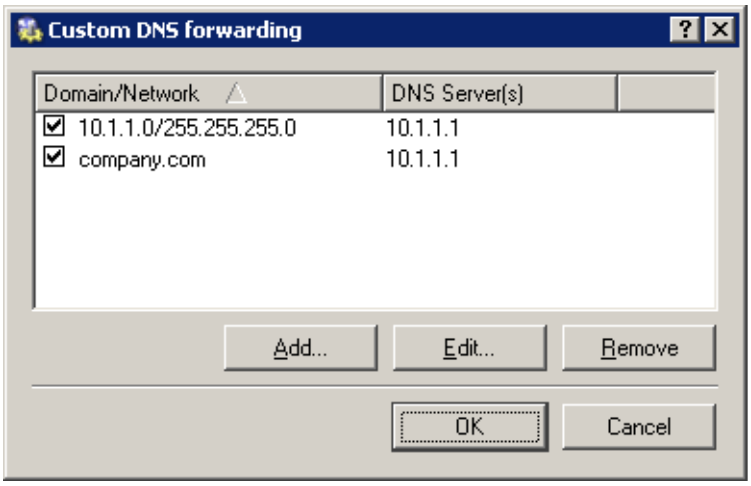
3. Customize DNS configuration as follows:

   - In configuration of the *DNS Forwarder* in *WinRoute*, specify DNS servers to which DNS queries which are not addressed to the `company.com` domain will be for-

warded (primary and secondary DNS server of the Internet connection provider by default).



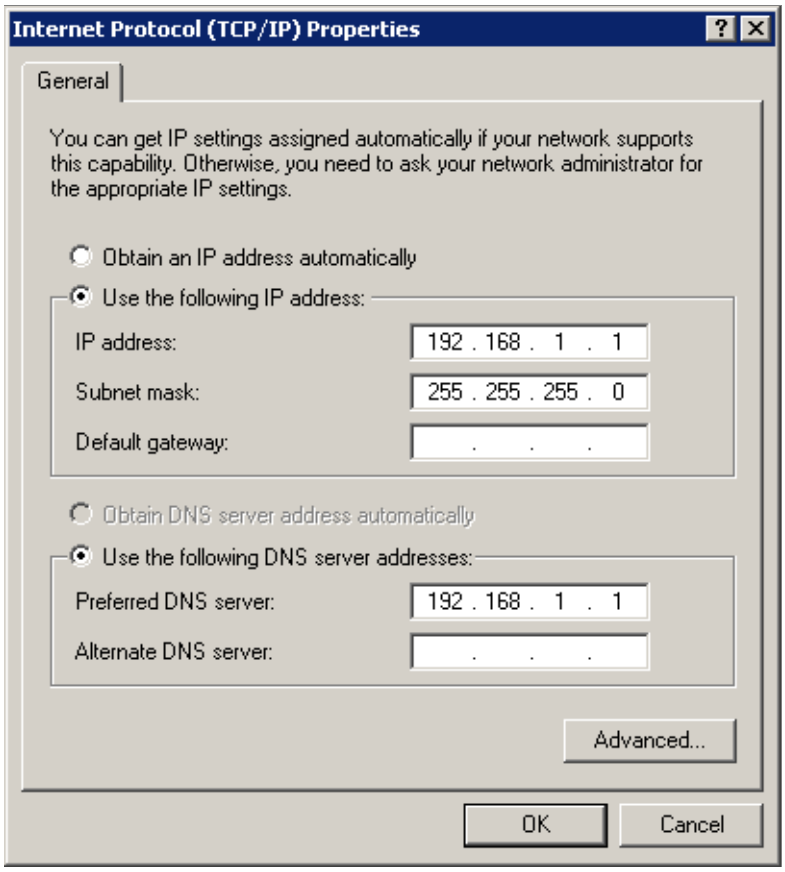
- Enable the *Use custom forwarding* option and define rules for the `company.com` domain. To specify the forwarding DNS server, use the IP address of the remote *WinRoute* host's interface connected to the local network.



- Set the IP address of this interface (`192.168.1.1`) as a primary DNS server for the *WinRoute* host's interface connected to the local network.

- Set the IP address `192.168.1.1` as a primary DNS server also for the other hosts.

Note: For proper functionality of DNS, the DNS database must include records for hosts in a corresponding local network. To achieve this, save DNS names and IP



addresses of local hosts into the `hosts` file (if they use IP addresses) or enable co-operation of the *DNS Forwarder* with the DHCP server (in case that IP addresses are assigned dynamically to these hosts). For details, refer to chapter 4.3.

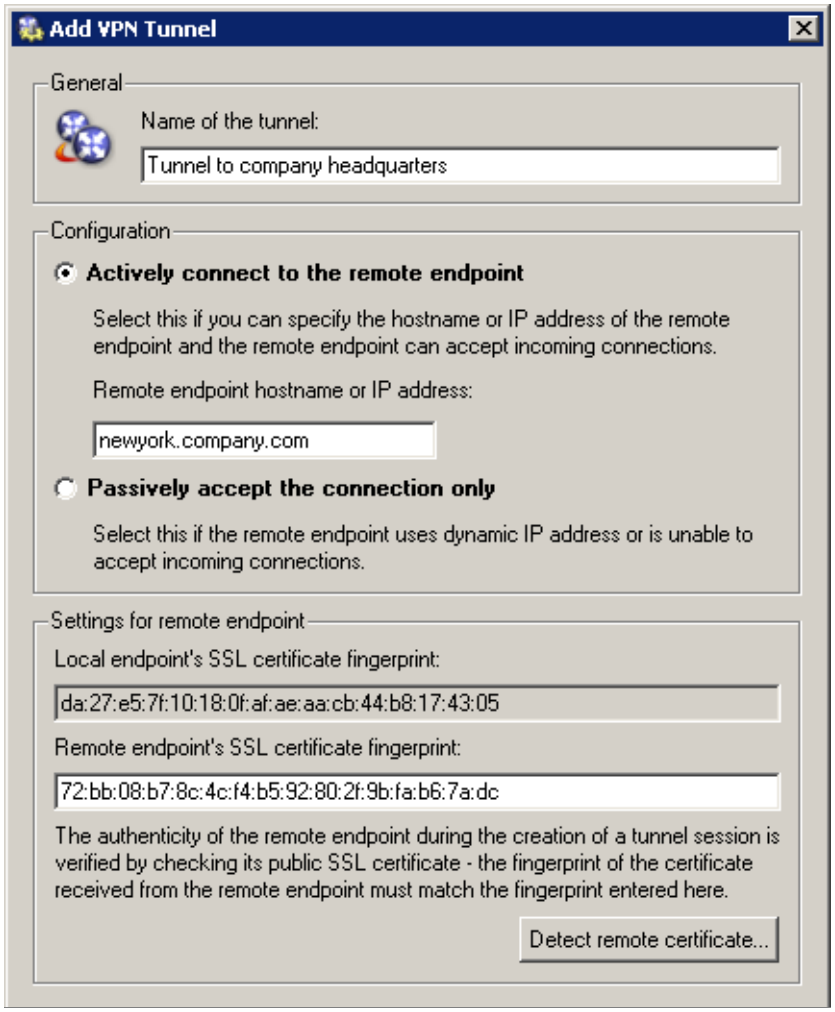
4. Enable the VPN server and configure its SSL certificate (create a self-signed certificate if no certificate provided by a certification authority is available).

   Note: A free subnet which has been selected is now specified automatically in the *VPN network* and *Mask* entries.

   For detailed description on VPN server configuration, refer to chapter 12.1.

5. Create an active endpoint of the VPN tunnel which will connect to the headquarters server (`newyork.company.com`). Use the fingerprint of the VPN server of the headquarters as a specification of the fingerprint of the remote SSL certificate.

   At this point, connection should be established (i.e. the tunnel should be created). If connected successfully, the *Connected* status will be reported in the *Adapter info* column for both ends of the tunnel. If the connection cannot be established, we recommend you to check the configuration of the traffic rules and test availability



of the remote server — in our example, the `ping newyork.company.com` command can be used at the branch office server.

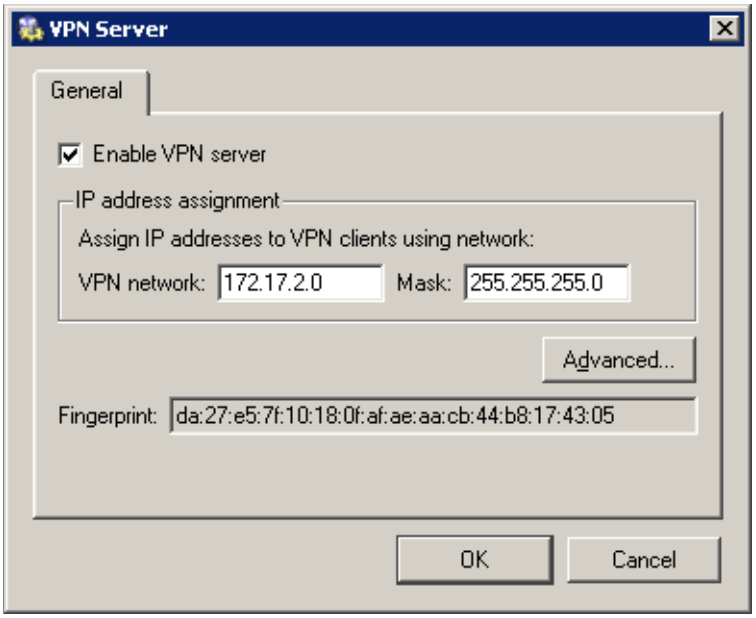
Note: If a collision of VPN network and the remote network is detected upon creation of the VPN tunnel, select an appropriate free subnet and specify its parameters at the VPN server (see Step 4).

For detailed information on how to create VPN tunnels, see chapter 12.4.

6. Add the new VPN tunnel into the *Local Traffic* rule. It is also possible to remove the *Dial-In* interface and the *VPN clients* group from this rule (VPN clients are not allowed to connect to the branch office).

   Note: It is not necessary to perform any other customization of traffic rules. The required restrictions should be already set in the traffic policy at the server of the headquarters.

Name	Source	Destination	Service	Action	Translation
Local Traffic	Firewall LAN Tunnel to company headquarters	Firewall LAN Tunnel to compa	Any	✓	
Service Kerio VPN	Internet	Firewall	Kerio VPN	✓	



VPN test

Configuration of the VPN tunnel has been completed by now. At this point, it is recommended to test availability of the remote hosts from each end of the tunnel (from both local networks).

For example, the `ping` or/and `tracert` operating system commands can be used for this testing. It is recommended to test availability of remote hosts both through IP addresses and DNS names.

If a remote host is tested through IP address and it does not respond, check configuration of the traffic rules or/and find out whether the subnets do not collide (i.e. whether the same subnet is not used at both ends of the tunnel).

If an IP address is tested successfully and an error is reported (*Unknown host*) when a corresponding DNS name is tested, then check configuration of DNS.

Chapter 13

Registration and Licensing Policy

WinRoute must be registered at Kerio Technologies website (`http://www.kerio.com/`) after the purchase. Once the product is registered, you will obtain a license key (an encrypted `license.key` file) that must be imported into the program. If the key is not imported, *WinRoute* will behave as a full-featured trial version and its license will be limited by the expiration timeout.

This also implies that the only difference between a trial version and full *WinRoute* version is whether the registration key has been imported or not. This gives each customer an opportunity to test and try the product in the particular environment during the 30-day period. It is not necessary to re-install nor reconfigure *WinRoute* to after the registration, only the license key is to be imported into the trial version.

If the trial period has expired, *WinRoute* will block all network traffic of its host. You will be allowed to access only the *Kerio Administration Console* in order to import the license key. Full functionality in *WinRoute* will be available after a valid license key is imported.

13.1 License Types

WinRoute can optionally include the following components: *McAfee* antivirus (refer to chapter 7). These components are licensed individually. License keys consist of the following information:

***WinRoute* license** Basic *WinRoute* license. Its validity is defined by the two following factors:

- update right expiration date — specifies the date by which *WinRoute* can be updated for free. When this date expires, *WinRoute* keeps functioning, however, it cannot be updated. The time for updates can be extended by purchasing a subscription.

- product expiration date — specifies the date by which *WinRoute* stops functioning and blocks all TCP/IP traffic at the host where it is installed. If this happens, a new valid license key must be imported or *WinRoute* must be uninstalled.

***McAfee* license** This license is defined by the two following dates:

- update right expiration date (independent of *WinRoute*) — when this date expires, the antivirus keeps functioning, however, neither its virus database nor the antivirus can be updated yet.

  Warning: Owing to persistent incidence of new virus infections we recommend you to use always the most recent antivirus versions.

- plug-in expiration date— specifies the date by which the antivirus stops functioning and cannot be used anymore.

Cobion license *Cobion Orange Filter* system is provided as a service. License is defined only by an expiration date which specifies when the *Cobion* system will be blocked.

Note: Refer to Kerio Technologies Website (`http://www.kerio.com/`) to get up-to-date information about individual licenses, subscription extensions, etc.

13.2 Viewing License Information and License Key Import

The license information can be displayed by selecting *Kerio WinRoute Firewall* (the first item in the tree in the left part of the *Kerio Administration Console* dialog window — this section is displayed automatically whenever the *WinRoute* administration is entered).

Product name of the product (*WinRoute*)

Copyright Copyright information.

Homepage Link to the *Kerio WinRoute Firewall* homepage (information on pricing, new versions, etc.). Click on the link to open the homepage in your default browser.

Operating system Name of the operating system on which the *WinRoute Firewall Engine* service is running.

License ID License number or a special license name.

Subscription expiration date Date until when the product can be upgraded for free.

Product expiration date Date when the product expires and stops functioning (only for trial versions or special license types).

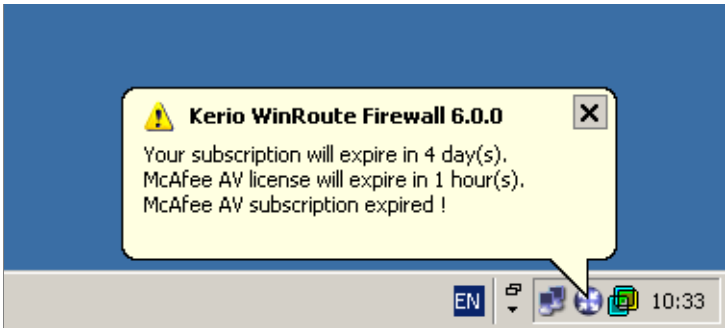
Number of users Maximal number of hosts (unique IP addresses) that can be connected to the Internet via *WinRoute* at the same time. The *WinRoute* host is not included in the count.

Company name of the company/person that has registered the product

The *Install License...* link will open the standard dialog for opening the file with the license key . The license information will be displayed if the import process has been completed successfully.

13.3 Subscription / Update Expiration

Users will be periodically informed about time that remains to the license or update expiration by the *WinRoute Engine Monitor* when *WinRoute*, the *McAfee* antivirus or the *Cobion* rating system expiration date is getting closer.



The first time this information is reported 7 days before the expiration date and then it is displayed a few times a day unless *WinRoute* or any of its components automatically stops functioning or until rights for *WinRoute's* or *McAfee's* update expires.

Note: Information about expiration will not be displayed unless the *WinRoute Engine Monitor* is running, however, the *WinRoute* administrator can specify to send notification of this event by email or SMS.

13.4 License Management

The *WinRoute* host is not counted toward the overall number of unique IP addresses.

The total number of connections includes also all connected VPN clients.

DNS queries (processed by the *DNS Forwarder* or by the DNS server at the *WinRoute* host), DHCP or other local traffic are not included in the license.

A license is considered free after 15 minutes of idleness (i.e. IP address of the corresponding host is removed from the table after this interval and the license can be used by another host).

Chapter 14

Status Information

WinRoute activities can be well monitored by the administrator (or by other users with appropriate rights). There are three types of information — status monitoring, statistics and logs.

Note: Only traffic allowed by traffic rules (see chapter 5) can be viewed. If a traffic attempt which should have been denied is detected, the rules are not well defined.

- Communication of each computer, users connected or all connections using *WinRoute* can be monitored.

  Note: Only traffic allowed by traffic rules (see chapter 5) can be viewed. If a traffic attempt which intended to be denied is detected, the rules are not well defined.

- Statistics provide information on users and network traffic for a certain time period. Statistics are viewed in the form of charts and tables. For details refer to chapter 15.

- Logs are files where information about certain activity is reported (e.g. error or warning reports, debug information etc.). Each item has one line and is marked with time specification (date and time when the action was taken). In all language versions of *WinRoute*, reports recorded are available in English only and they are generated by the *WinRoute Firewall Engine*. For details, see chapter 16.

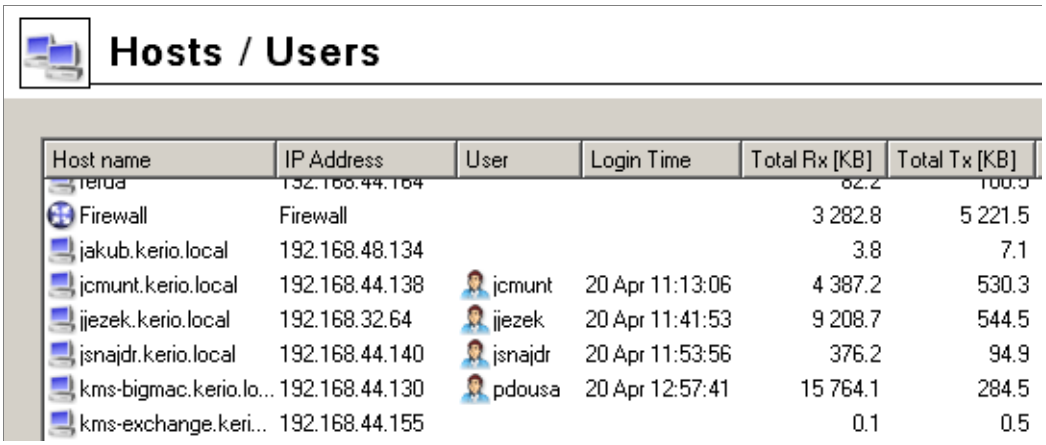
To learn what types of information and the methods for monitoring, refer to the following chapters.

14.1 Hosts and Users

In *Status / Hosts / Users*, the hosts within the local network or active users using *WinRoute* for communication with the Internet will be displayed.

Note: For more details about the firewall user's logon see chapter 8.2.

Look at the upper window to view information on individual hosts, connected users, data size/speed, etc.



The following information can be found in the *Hosts / Users* window:

Host name DNS name of a host. In case that no corresponding DNS record is found, IP address is displayed instead.

User Name of the user which is connected from a particular host. If no user is connected, the item is empty.

Currently Rx, Currently Tx Monitors current traffic speed (kilobytes per second) in both directions (from and to the host — *Rx* values represent incoming data, *Tx* values represent outgoing data)

The following columns are hidden by default. To view these columns select the *Modify columns* option in the context menu (see below).

IP Address IP address of the host from which the user is connecting from

Login time Date and time of the recent user login to the firewall

Login duration Monitors length of the connection. This information is derived from the current time status and the time when the user logged on

Inactivity time Duration of the time with zero data traffic. You can set the firewall to logout users automatically after the inactivity exceeds allowed inactivity time (for more details see chapter 8.1)

Start time Date and time when the host was first acknowledged by *WinRoute*. This information is kept in the operating system until the *WinRoute Firewall Engine* disconnected.

Total received, Total transmitted Total size of the data (in kilobytes) received and transmitted since the *Start time*

Connections Total number of connections to and from the host. Details can be displayed in the context menu (see below)

Authentication method Authentication method used for the recent user connection:

- *plaintext* — user is connected through an insecure login site *plaintext*
- *SSL* — user is connected through a login site protected by SSL security system *SSL*
- *proxy* — a *WinRoute* proxy server is used for authentication and for connection to Websites
- *NTLM* — user was authenticated with NTLM in NT domain (this is the standard type of login if Microsoft Internet Explorer 5.5 or later or Mozilla 1.4 or later is used)
- *VPN client* — user has connected to the local network using the *Kerio VPN Client* (for details, see chapter 12).

  Note: Connections are not displayed and the volume of transmitted data is not monitored for VPN clients.

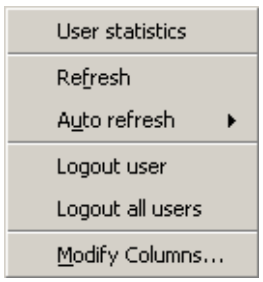
For more details about connecting and user authentication see chapter 8.2.

Information displayed in the *Hosts / Users* window can be refreshed by clicking on the *Refresh*button.

Use the *Show / Hide details* to open the bottom window providing detailed information on a user, host and open connections.

Hosts / Users Dialog Options

Clicking the right mouse button in the *Hosts / Users* window (or on the record selected) will display a context menu that provides the following options:



User statistics Use this option to switch to the *User statistics* tab in *Status / Statistics* where detailed user statistics can be viewed.

This option is available only for hosts from which a user is connected at the moment.

Refresh This option refreshes information in the *Hosts / Users* window immediately (this function is equal to the *Refresh* button displayed at the bottom of the window).

Auto refresh Settings for automatic refreshing of the information in the *Hosts / Users* window. Information can be refreshed in the interval from 5 seconds up to 1 minute or the auto refresh function can be switched off (*No refresh*).

Logout user Immediate logout of a selected user.

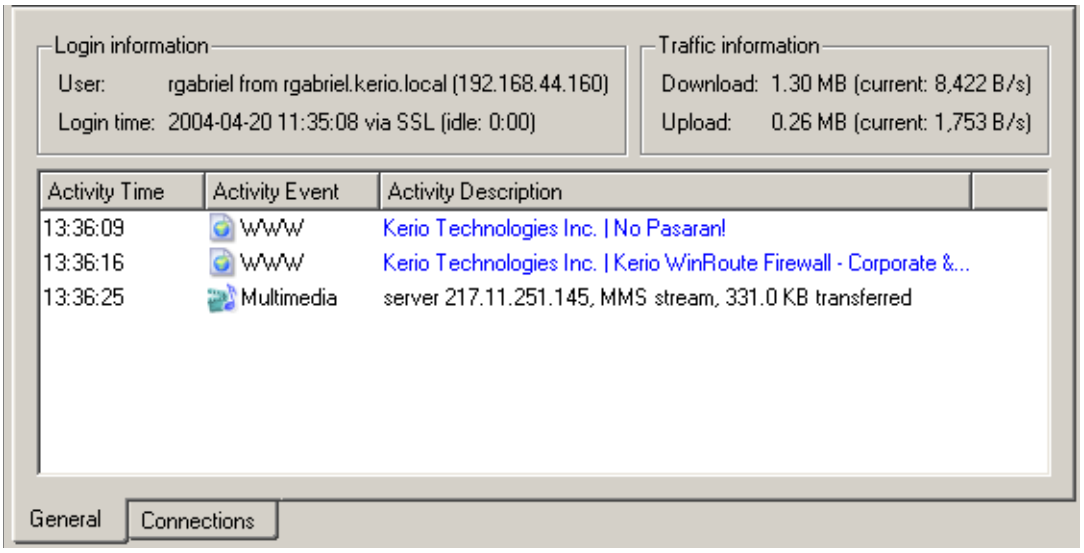
Logout all users Immediate logout of all firewall users.

Manage Columns By choosing this option you can select columns to be displayed in the *Hosts / Users* window (see chapter 3.2).

Detailed information on a selected host and user

Detailed information on a selected host and connected user are provided in the bottom window of the *Hosts / Users* section.

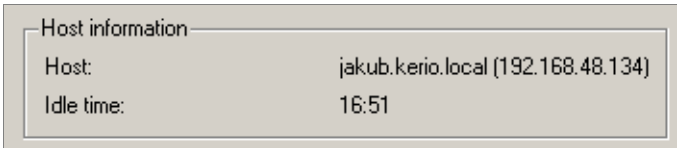
Open the *General* tab to view information on user's login, size/speed of transmitted data and information on activities of a particular user.



Login information Information on logged-in users:

- *User* — name of a user, DNS name (if available) and IP address of the host from which the user is connected
- *Login time* — date and time when a user logged-in, authentication method that was used and inactivity time (idle).

If no user is connected from a particular host, detailed information on the host are provided instead of login information.



- *Host* — DNS name (if available) and IP address of the host
- *Idle time* — time for which no network activity performed by the host has been detected

Traffic information Information on size of data received (*Download*) and sent (*Upload*) by the particular user (or host) and on current speed of traffic in both directions.

Overview of detected activities of the particular user (host) are given in the main section of this window:

Activity Time Time (in minutes and seconds) when the activity was detected.

Activity Event Type of detected activity (network communication). *WinRoute* distinguishes between the following activities: *SMTP*, *POP3*, *WWW* (HTTP traffic), *FTP*, *Streams* (real-time transmission of audio and video streams) and *P2P* (use of Peer-To-Peer networks).

Note: WinRoute is not able to recognize which type of P2P network is used. According to results of certain testing it can only "guess" that it is possible that the client is connected to such network. For details refer to chapter 11.10.

Activity Description Detailed information on a particular activity:

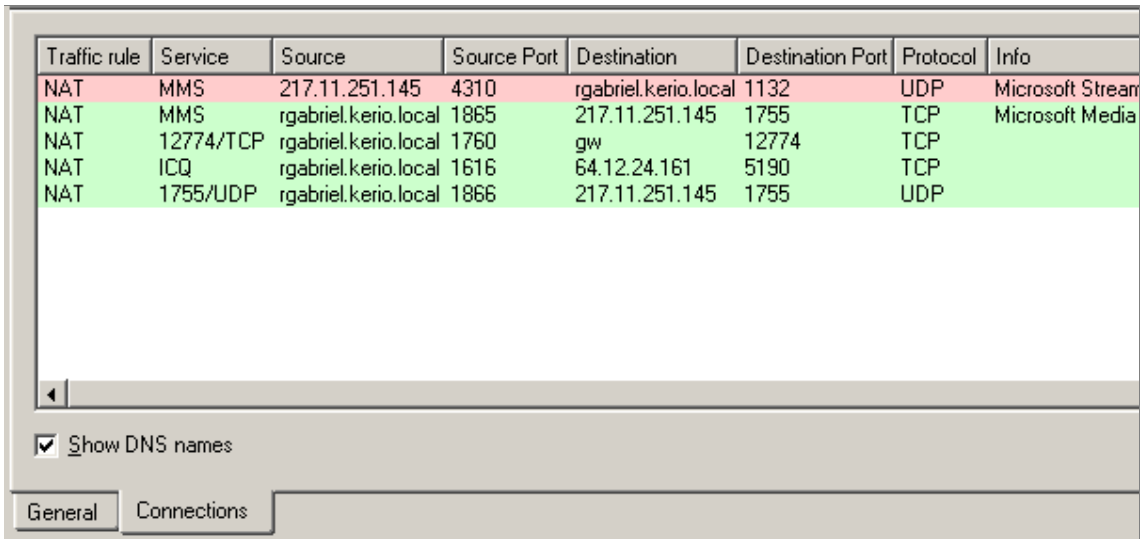
- *WWW* — title of a Web page to which the user is connected (if no title is available, URL will be displayed instead). Page title is a hypertext link — click on this link to

open a corresponding page in the browser which is set as default in the operating system.

- *SMTP, POP3* — DNS name or IP address of the server, size of downloaded/uploaded data.
- *FTP* — DNS name or IP address of the server, size of downloaded/saved data, information on currently downloaded/saved file (name of the file including the path, size of data downloaded/uploaded from/to this file).
- *Multimedia* (real time transmission of video and audio data) — DNS name or IP address of the server, type of used protocol (*MMS*, *RTSP*, *RealAudio*, etc.) and volume of downloaded data.
- *P2P* — information that the client is probably using Peer-To-Peer network.

Connections

The *Connections* tab provides detailed information on connections from and to a selected host.



Information about connections:

Traffic rule Name of the *WinRoute* traffic rule (see chapter 5) by which the connection was allowed.

Service Name of the service. For non-standard services, port numbers and protocols are displayed.

Source, Destination Source and destination IP address (or name of the host in case that the *Show DNS names* option is enabled —see below).

The following columns are hidden by default. They can be shown through the *Modify columns* dialog opened from the context menu (for details, see chapter 3.2).

Source port, Destination port Source and destination port (only for TCP and UDP protocols).

Protocol Protocol used for the transmission (TCP, UDP, etc.).

Timeout Time left before the connection will be removed from the table of *WinRoute's* connections.

Each new packet within this connection sets timeout to the initial value. If no data is transmitted via a particular connection, *WinRoute* removes the connection from the table upon the timeout expiration — the connection is closed and no other data can be transmitted through it.

Rx, Tx Volume of incoming (*Rx*) and outgoing (Tx) data transmitted through a particular connection (in KB).

Info Additional information (such as a method and URL in case of HTTP protocol).

Use the *Show DNS names* option to enable/disable showing of DNS names instead of IP addresses in the *Source* and *Destination* columns. If a DNS name for an IP address cannot be resolved, the IP address is displayed.

You can click on the *Colors* button to open a dialog where colors used in this table can be set.

Note: Upon right-clicking on a connection, the context menu extended by the *Kill connection* option is displayed. This option can be used to kill the connection immediately.

14.2 Connection Overview

In *Status / Connections*, all the network connections which can be detected by *WinRoute* include the following:

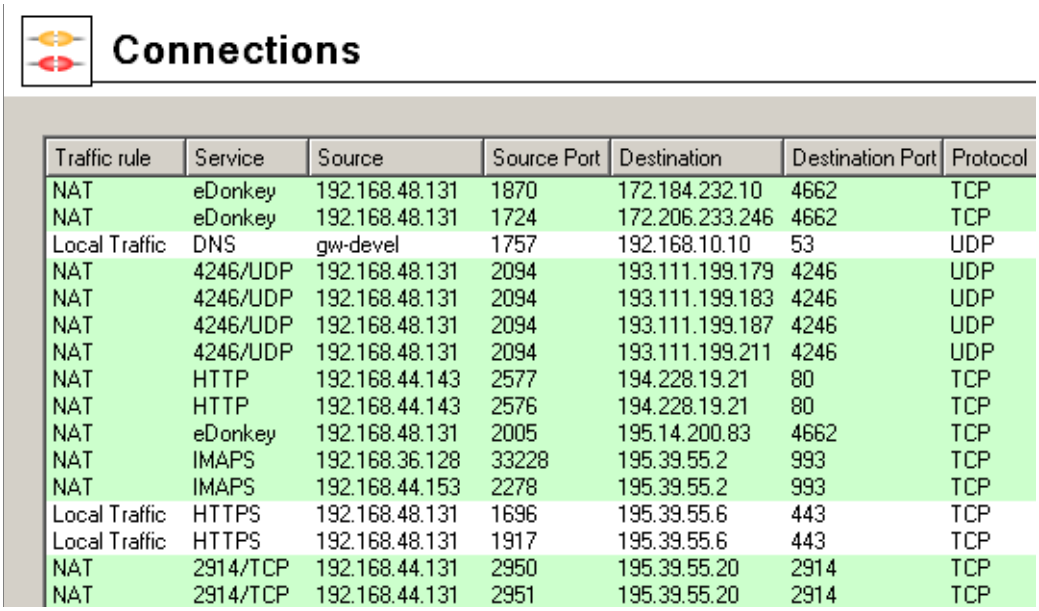
- client connections to the Internet through *WinRoute*
- connections from the host on which *WinRoute* is running
- connections from other hosts to services provided by the host with *WinRoute*
- connections performed by clients within the Internet that are mapped to services running in LAN

Notes:

1. connections among local clients will not be detected nor displayed by *WinRoute*.

2. UDP protocol is also called connectionless protocol. This protocol does not perform any connection. The communication is performed through individual messages (so-called datagrams). Periodic data exchange is monitored in this case.

WinRoute administrators are allowed to close any of the active connections.



One connection is represented by each line of this window. These are network connections, not user connections (each client program can occupy more than one connection at a given moment). The columns provide the following information:

Traffic rule Name of the *WinRoute* traffic rule (see chapter 5) by which the connection was allowed.

Service Name of transmitted service (if such service is defined in *WinRoute* — see chapter 9.3). If the service is not defined in *WinRoute*, the corresponding port number and protocol will be displayed instead (e.g. *5004/UDP*).

Source and Destination IP address of the source (the connection initiator) and of the destination. If there is an appropriate reverse record in DNS, the IP address will be substituted with the DNS name.

The following columns are hidden by default. They can be enabled through the *Modify columns* dialog opened from the context menu (for details, see chapter 3.2).

Source Port and Destination Port Ports used for the particular connection.

Protocol Communication protocol (*TCP* or *UDP*)

Timeout Time left until automatic disconnection. The countdown starts when data traffic stops. Each new data packet sets the counter to zero.

Rx and Tx Total size of data received (*Rx*) or transmitted (*Tx*) during the connection (in kilobytes). Received data means the data transferred from *Source* to *Destination*, transmitted data means the opposite.

Info An informational text describing the connection (e.g. about the protocol inspector applied to the connection).

Information in *Connections* is refreshed automatically within a user defined interval or the *Refresh* button can be used for manual refreshing.

Options of the Connections Dialog

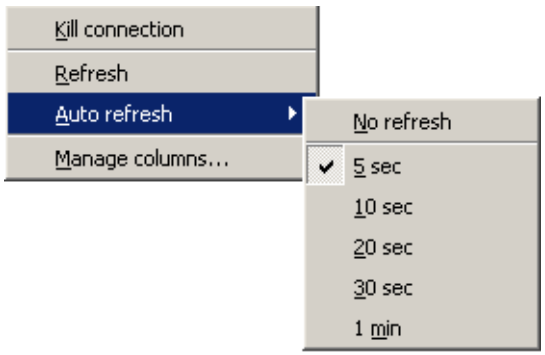
The following options are available below the list of connections:

- *Hide local connections* — connections from or/and to the *WinRoute* host will not be displayed in the *Connections* window.

  This option only makes the list better-arranged and distinguishes connections of other hosts in the local network from the *WinRoute* host's connections.

- *Show DNS names* — this option displays DNS names instead of IP addresses. If a DNS name is not resolved for a certain connection, the IP address will be displayed.

Right-click on the *Connections* window (on the connection selected) to view a context menu including the following options:



Kill connection Use this option to finish selected connection immediately (in case of UDP connections all following datagrams will be dropped).

Note: This option is active only if the context menu has been called by right-clicking on a particular connection. If called up by right-clicking in the *Connections* window (with no connection selected), the option is inactive.

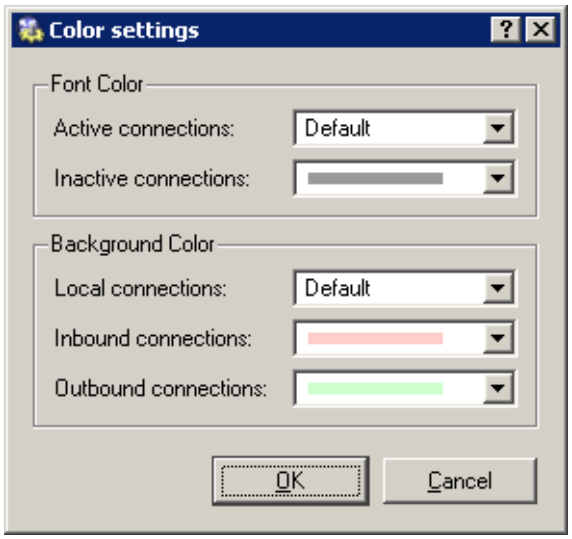
Refresh This option will refresh the information in the *Connections* window immediately. This function is equal to the function of the *Refresh* button at the bottom of the window.

Auto refresh Settings for automatic refreshing of the information in the *Connections* window. Information can be refreshed in the interval from 5 seconds up to 1 minute or the auto refresh function can be switched off (*No refresh*).

Manage columns By choosing this option you can select which columns will be displayed in the *Connections* window (see chapter 3.2).

Color Settings

Clicking on the *Colors* button displays the color settings dialog to define colors for each connection:



For each item either a color or the *Default* option can be chosen. Default colors are set in the operating system (the common setting for default colors is black font and white background).

Font Color

- *Active connections* — connections with currently active data traffic
- *Inactive connections* — TCP connections which have been closed but 2 minutes after they were killed they are still kept active — to avoid repeated packet mishandling)

Background Color

- *Local connections* — connections where an IP address of the host with *WinRoute* is either source or destination
- *Inbound connections* — connections from the Internet to the local network (allowed by firewall)
- *Outbound connections* — connections from the local network to the Internet

Note: Incoming and outgoing connections are distinguished by detection of direction of IP addresses — "out" (*SNAT*) or "in" (*DNAT*). For more details see chapter 5.

14.3 Alerts

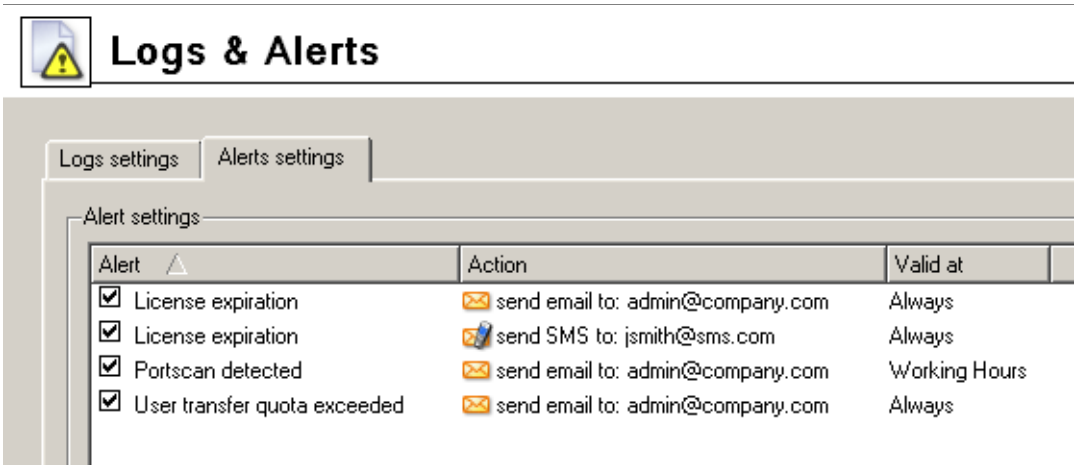
WinRoute enables automatic sending of messages informing the administrator about important events. This makes *WinRoute* administration more comfortable, since it is not necessary to connect to the firewall via the *Administration Console* too frequently to view all status information and logs (however, this does not mean that it is not worthy to do this occasionally).

WinRoute generates alert messages upon detection of any specific event for which alerts are preset. All alert messages are recorded into the *Alert* log (see chapter 16.3). The *WinRoute* administrator can specify which alerts will be sent to whom, as well as a format of the alerts. Sent alerts can be viewed in *Status / Alerts*.

Note: SMTP relay must be set in *WinRoute* (see chapter 11.9), otherwise alerting will not work.

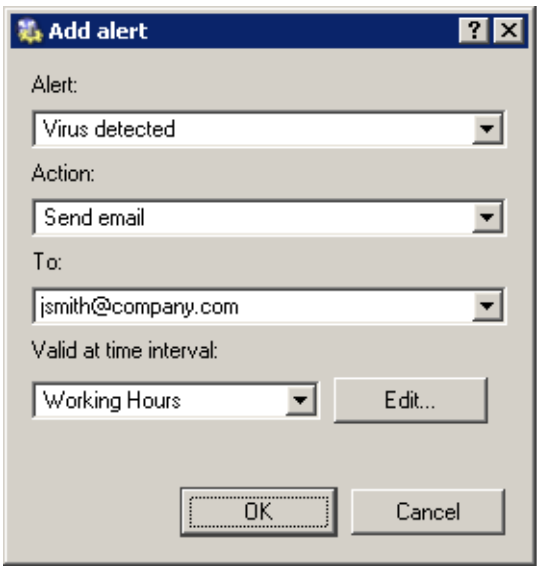
Alerts settings

Alerts settings can be configured in the *Alerts settings* tab under *Configuration / Logs & Alerts*.



This tab provides list of "rules" for alert sending. Use checking boxes to enable/disable individual rules.

Use the *Add* or the *Edit* button to (re)define an alert rule.



Alert Type of the event upon which the alert will be sent:

- *Virus detected* — antivirus engine has detected a virus in a file transmitted by HTTP, FTP, SMTP or POP3 (refer to chapter 7).
- *Portscan detected* — *WinRoute* has detected a *port scanning* attack (either an attack passing through or an attack addressed to the *WinRoute* host).

- *Host connection limit reached* — a host in the local network has reached the connection limit (see chapter 11.5). This may indicate deployment of an undesirable network application (e.g. Trojan horse or a spyware) on a corresponding host.

- *Low free disc space warning* — this alert warns the administrator that the free space of the *WinRoute* host is low (under 11 per cent of the total disc capacity).

  WinRoute needs enough disc space for saving of logs, statistics, configuration settings, temporary files (e.g. an installation archive of a new version or a file which is currently scanned by an antivirus engine) and other information. Whenever the *WinRoute* administrator receives such alert message, adequate actions should be performed immediately.

- *New version available* — a new version of *WinRoute* has been detected at the server of Kerio Technologies during an update check. The administrator can download this version from the server or from `http://www.kerio.com/` and install it using the *Administration Console* (see chapter 11.8).

- *User transfer quota exceeded* — a user has reached daily or monthly user transfer quota and *WinRoute* has responded by taking an appropriate action. For details, refer to chapter 10.1.

- *Connection failover event* — the Internet connection has failed and the system was switched to an alternate line, or vice versa (it was switched back to the primary line). For detailed information, see chapter 4.2.

- *License expiration* — expiration date for the corresponding *WinRoute* license/subscription (or license of any module integrated in *WinRoute*, such as *Cobion*, the *McAfee* antivirus, etc.) is getting closer. The *WinRoute* administrator should check the expiration dates and prolong a corresponding license or subscription (for details, refer to chapter 13).

- *Dial/Hang-up of RAS line* — *WinRoute* is dialing or hanging-up a RAS line (see chapter 4.1). The alert message provides detailed information on this event: line name, reason of the dialing, username and IP address of the host from which the request was sent.

Action Method of how the user will be informed:

- *Send email* — information will be sent by an email message,

- *Send SMS (shortened email)* — short text message will be sent to the user's cell phone.

Note: SMS messages are also sent as email. User of the corresponding cell phone must use an appropriate email address (e.g. `number@provider.com`). Sending of SMS to telephone numbers (for example via GSM gateways connected to the *WinRoute* host) is not supported.

To Email address of the recipient or of his/her cell phone (related to the *Action* settings).

Recipients can be selected from the list of users (email addresses) used for other alerts or new email addresses can be added by hand.

Valid at time interval Select a time interval in which the alert will be sent. Click *Edit* to edit the interval or to create a new one (details in chapter 9.2).

Alert templates

Formats of alert messages (email or/and SMS) are defined by templates. Individual formats can be viewed in the *Status / Alerts* section of the *Administration Console*. Templates are predefined messages which include certain information (e.g. username, IP address, number of connections, virus information, etc.) defined through specific variables. *WinRoute* substitutes variables by corresponding values automatically. The *WinRoute* administrator can customize these templates.

Templates are stored in the `templates` subdirectory of the installation directory of *WinRoute*

(`C:\Program Files\Kerio\WinRoute Firewall\templates` by default):

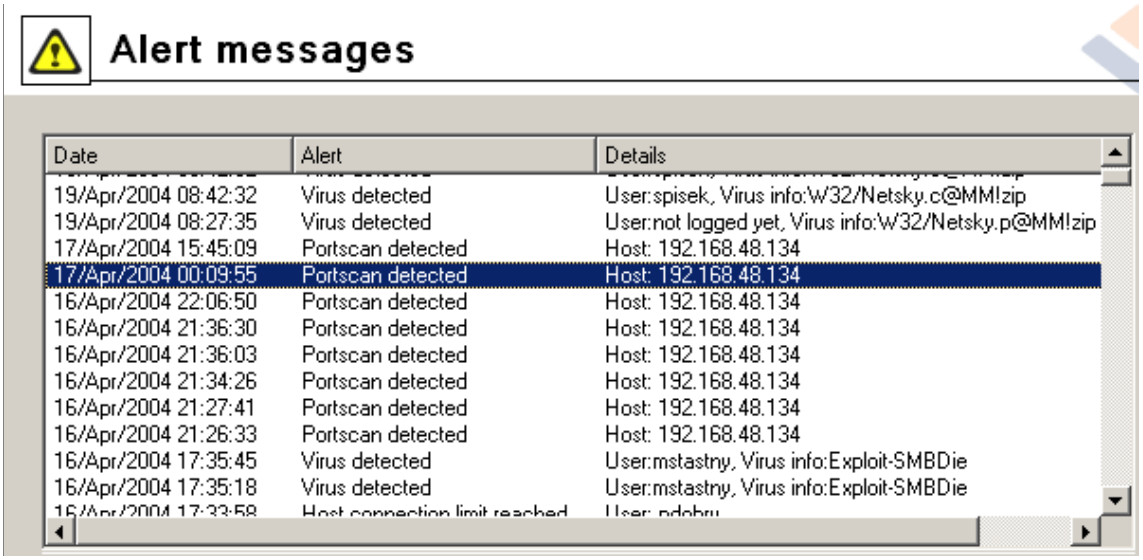
- the `console` subdirectory — messages displayed in the top section of *Status / Alerts* (overview),
- the `console\details` subdirectory — messages displayed at the bottom section of *Status / Alerts* (details),
- the `email` subdirectory — messages sent by email (each template contains a message in the plain text and HTML formats),
- the `sms` subdirectory — SMS messages sent to a cell phone.

Note: In the latest version of *WinRoute*, only English alerts are available (templates for other languages under `email` and `sms` subdirectories are ready for future versions).

Alerts overview

Overview of all sent alerts (defined in *Configuration / Logs & Alerts*) can be found under *Status / Alerts*. The language set in the *Administration Console* is used (if a template in a corresponding language is not found, the alert is displayed in English).

Overview of all sent alerts (sorted by dates and times) is provided in the top section of this window.

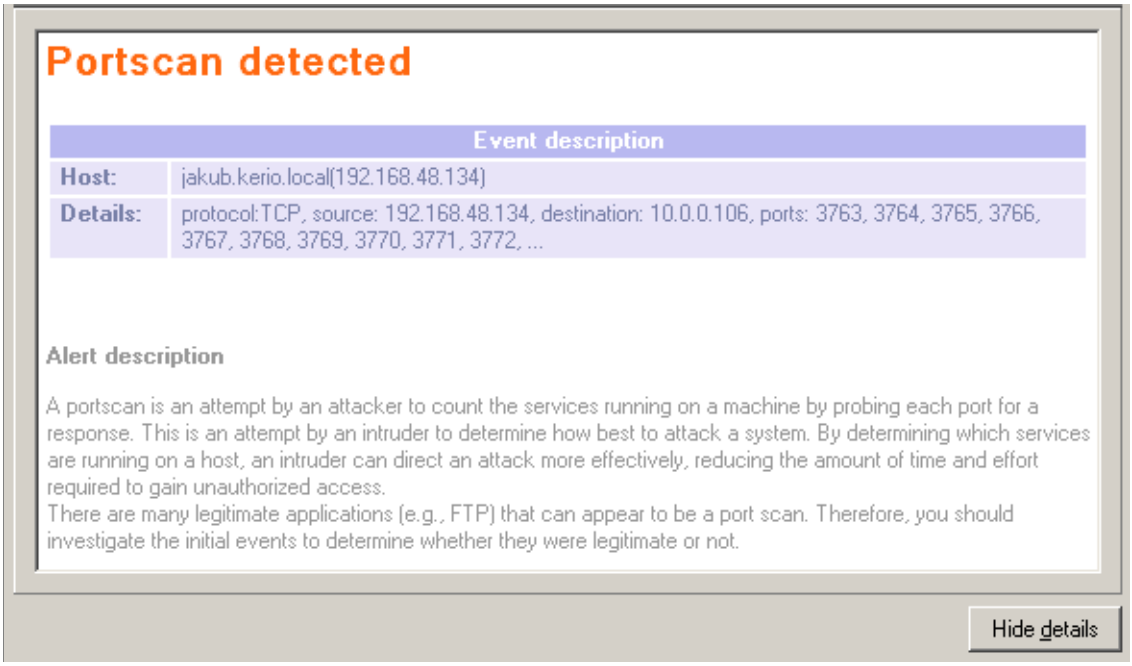


Each line provides information on one alert:

- *Date* — date and time of the event,
- *Alert* — event type,
- *Details* — basic information on events (IP address, username, virus name, etc.).

Click an event to view detailed information on the item including a text description (defined by templates under `console\details` — see above) in the bottom section of the window.

Note: Details can be optionally hidden or showed by clicking the *Hide/Show details* button (details are displayed by default).


Portscan detected
Event description
Host:
jakub.kerio.local(192.168.48.134)
Details:
protocol:TCP, source: 192.168.48.134, destination: 10.0.0.106, ports: 3763, 3764, 3765, 3766, 3767, 3768, 3769, 3770, 3771, 3772, ...
Alert description
A portscan is an attempt by an attacker to count the services running on a machine by probing each port for a response. This is an attempt by an intruder to determine how best to attack a system. By determining which services are running on a host, an intruder can direct an attack more effectively, reducing the amount of time and effort required to gain unauthorized access.
There are many legitimate applications (e.g., FTP) that can appear to be a port scan. Therefore, you should investigate the initial events to determine whether they were legitimate or not.
Hide details

Chapter 15

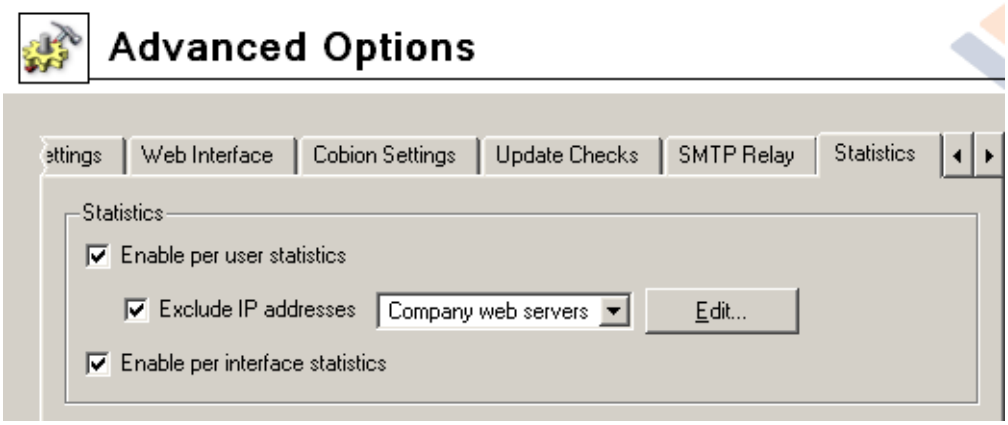
Statistics

Statistical information about users (volume of transmitted data, used services, categorization of Web pages) as well as of network interfaces of the *WinRoute* host (volume of transmitted data, load on individual lines) can be viewed in the *Status / Statistics* section of the administration console.

15.1 Preferences

Under certain circumstances (too many connected users, great volume of transmitted data, low capacity of the *WinRoute* host, etc.), viewing of statistics may slow *WinRoute* and data transmission (Internet connection) down. Be aware of this fact while opening the statistics. Statistics can also be focused only on certain destination hosts if necessary.

Statistics and their parameters can be set in the *Statistics* tab under *Configuration / Advanced Options*.



Enable per user statistics Use this option to enable/disable statistics for each local user (information provided in the *Top 20 users* and the *User statistics* tabs, see chapters 15.2 and 15.3).

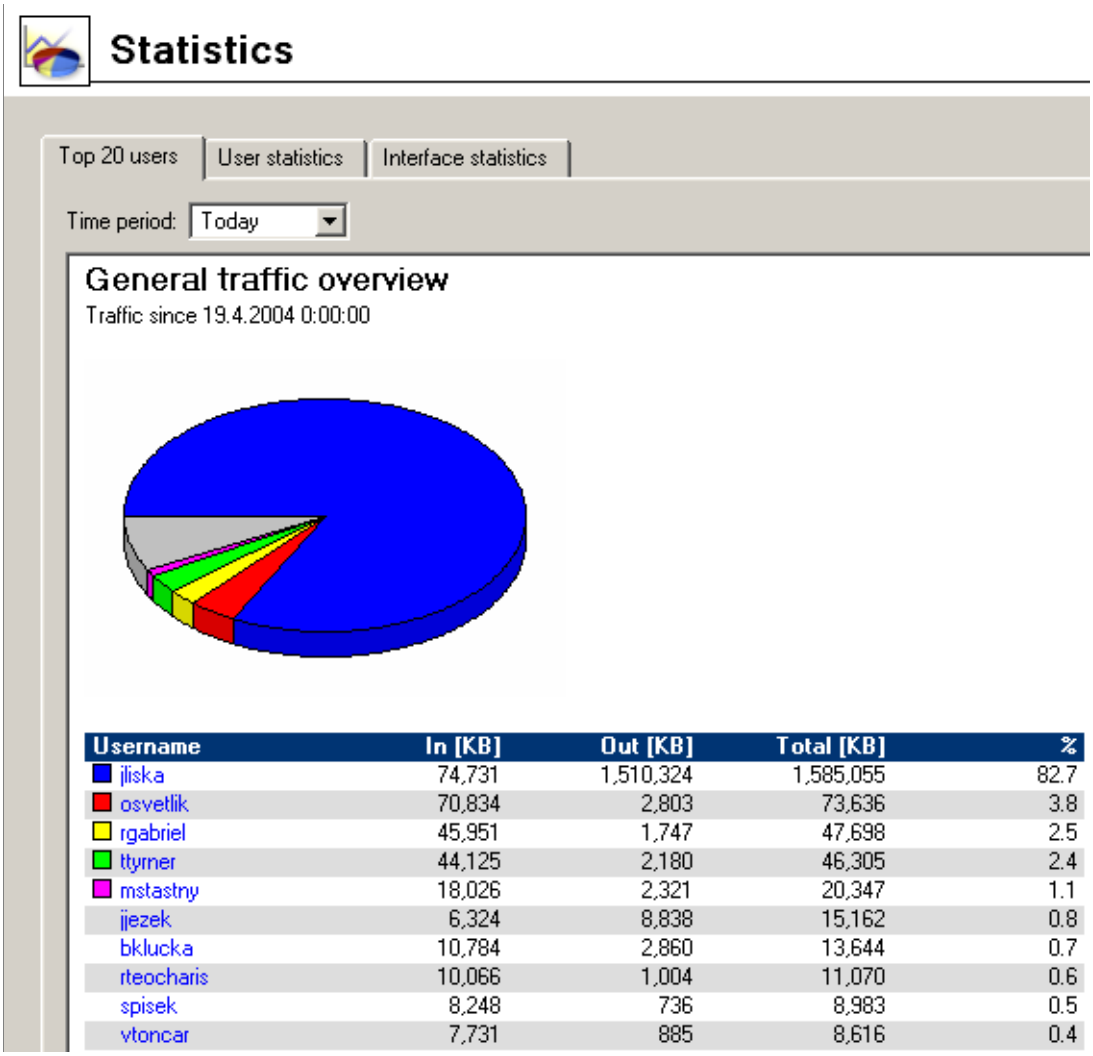
A group of IP addresses can be excluded by using the *Exclude IP addresses* (e.g. servers, testing subnet, etc.). Connections to these IP addresses will not be included in the statistics.

Note: If a user connects to the firewall both from monitored and excluded hosts, the statistics will display only information about the user's activity at hosts which are not excluded.

Enable per interface statistics This option enables/disables statistics for individual network interfaces of the *WinRoute* host, i.e. information provided in the *Interface statistics* tab (see chapter 15.4).

15.2 Top 20 users

The *Top 20 users* tab in *Status / Statistics* provides statistics on 20 users who have transmitted the greatest volume of data during a selected time period.



Username	In [KB]	Out [KB]	Total [KB]	%
jliska	74,731	1,510,324	1,585,055	82.7
osvetlik	70,834	2,803	73,636	3.8
rgabriel	45,951	1,747	47,698	2.5
ttyrner	44,125	2,180	46,305	2.4
mstastny	18,026	2,321	20,347	1.1
jjezek	6,324	8,838	15,162	0.8
bklucka	10,784	2,860	13,644	0.7
rteocharis	10,066	1,004	11,070	0.6
spisek	8,248	736	8,983	0.5
vtoncar	7,731	885	8,616	0.4

This period (*Today*, *This week*, *This month*, *Total*) can be selected in the *Time interval* box. Time periods always refer to whole time intervals (i.e. the range of the *This week* interval is from the first week day at 0:00 A.M. to the last week day at 12:00 P.M., etc. — initial and last week days depend on the configuration of a particular operating system).

The pie chart displays participation of the top five users in total volume of transmitted data for a selected time period. The grey field represents participation of the other users (including non-authenticated users).

The table below the chart provides a list of 20 users who have transmitted the greatest volume of data for the specified period. Users are listed by activity, starting with the most active user. Colors used in the chart are provided next to the names of the top five users.

The following information is provided for each user:

- volume of incoming (downloaded from the Internet) data,
- volume of outgoing (uploaded to the Internet) data,
- total volume of transmitted data (sum of downloaded and uploaded data),
- proportional participation in total volume of transmitted data in a selected time period.

Use the *Refresh* button to update data in the table and the chart.

15.3 User statistics

Detailed statistics on individual users are available in the *User statistics* tab under *Status / Statistics*.

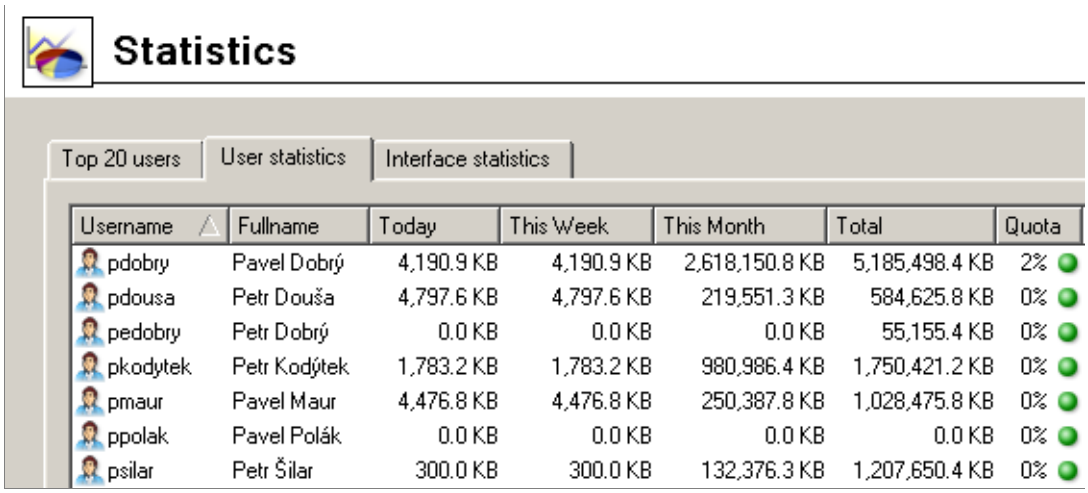
The columns of the table at the top of the window provide detailed statistics on volume of data transmitted by individual users during various time periods (today, this week, this month and total).

The *Quota* column provides usage of transfer quota by a particular user in percents (see chapter 10.1). Colors are used for better reference:

- green — 0%–74% of the quota is used
- yellow — 75%–99% of the quota is used
- red — 100% (limit reached)

Note: User quota consists of two limits: daily and monthly. The limit which is closer to reach the limit at the particular moment is showed in the *Quota* column..

The *all users* line provides total volume of data transmitted by all users in the table (even of the unrecognized ones). The *unrecognized users* item includes all users who are currently not authenticated at the firewall.

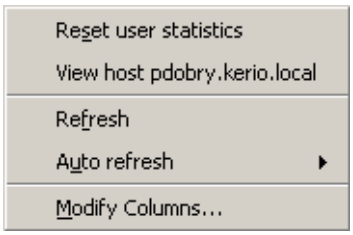


Notes:

1. Additional columns may be added through the context menu that provide volume of transmitted data, incoming (*IN*) or/and outgoing (*OUT*), for various time periods.

2. User statistics are saved in the `users.stat` file under the *WinRoute* directory. This implies that this data will be saved the next time the *WinRoute Firewall Engine* will be started.

Right-click on the table (or on an item of a selected user) to open the context menu with the following options:



- *Reset user statistics* — this option resets all values of the user's statistics. This option is available only if the mouse pointer is hovering a user at the moment when the context menu is opened.

  Note: Resetting the statistics will also unblock traffic for a corresponding user if it has been blocked after a transfer limit is reached (see chapter 10.1), since statistic values are used for quota checks.

Warning: Be aware that using this option for the

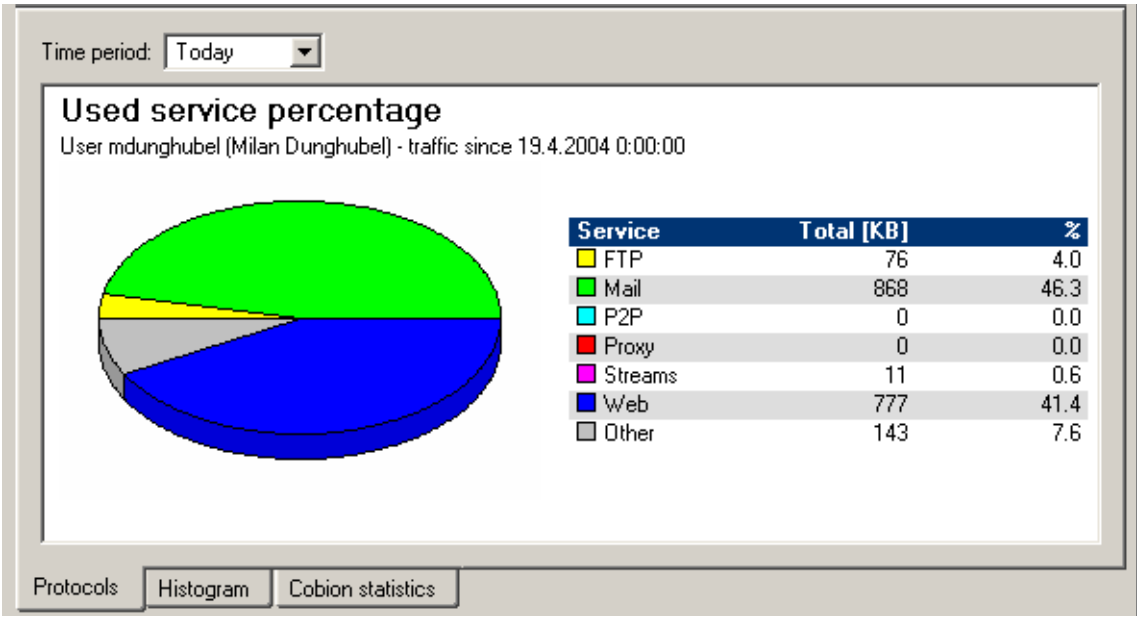
all users item resets statistics of all users, including unrecognized ones!

- *Refresh* — use this option to update information in the *User statistics* tab (this option is identical to the *Refresh* button at the bottom of the window).
- *Auto refresh* — settings for automatic updates of the data provided in the *User statistics* tab. The interval from 5 seconds to 1 minute can be used, or automatic refreshing of the page can be disabled by the *No refresh* option.
- *Modify columns* — use this option to select items (columns) which will be displayed in the table (see chapter 3.2).

The other section of the *User statistics* tab provides detailed statistics on a selected user (this section is divided into the following three tabs: *Protocols*, *Histogram* and *Cobion Statistics*). This section can be optionally shown / hidden by the *Show details / Hide details* button (the *Show details* mode is used by default).

Services

The *Protocols* tab provides information on services used by a selected user in a specified time period.



Use the *Time period* combo box to select a time period which will be covered by the statistics (for details, refer to chapter 15.2).

The table provides a list of the most common services. Volume of transmitted via each protocol by a selected user and their proportional participation in total traffic of the user. Proportional participation is also shown in the pie chart.

The following services are monitored:

- *Streams* — services enabling real-time transmission of sound and video files (e.g. *RTSP*, *MMS*, *RealAudio*, *MPEG Shoutcast*, etc.)
- *Mail* — email services (*SMTP*, *IMAP*, and *POP3* — both encrypted and unencrypted)
- *FTP* — unencrypted *FTP* service. Encrypted FTP (*FTPS*) cannot be monitored.
- *P2P* — *peer-to-peer* networks detected by *WinRoute*(for details, see chapter 11.10)
- *Proxy* — connections to the Internet via the *WinRoute* proxy server (see chapter 4.5)
- *WWW* — connections to Web pages (i.e. the *HTTP* and *HTTPS* protocols), except connections through the proxy server
- *Other* — other services

Note: Volumes of data transferred by individual services are measured only if these services use protocol inspectors. Other services are included in the *Other* group.

Line load

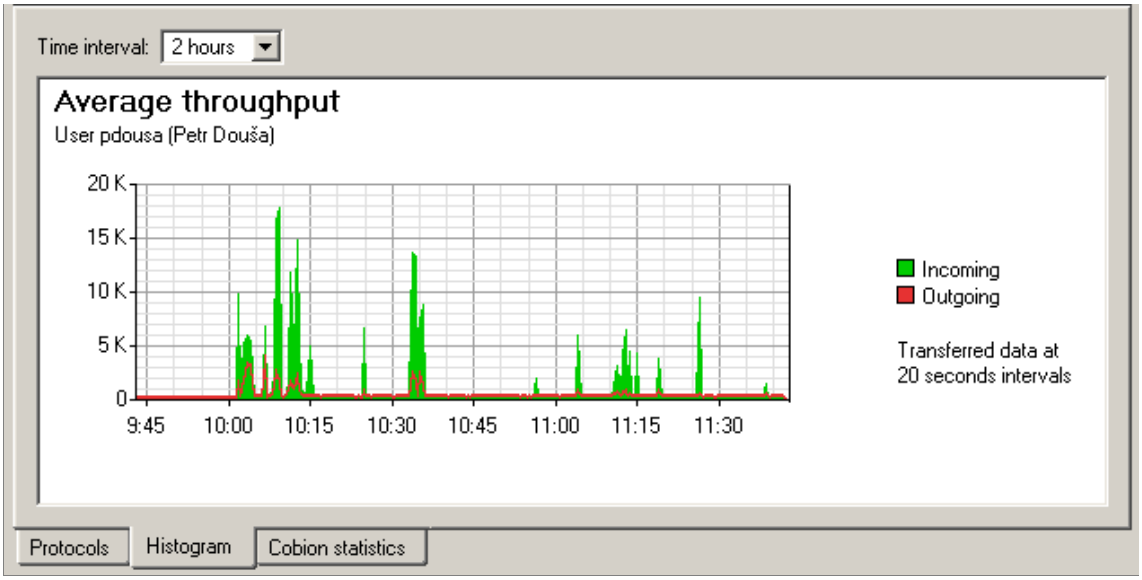
The *Histogram* tab provides a chart (timeline) informing about data transmitted by a particular user, all users or recognized users in a selected period (line load).

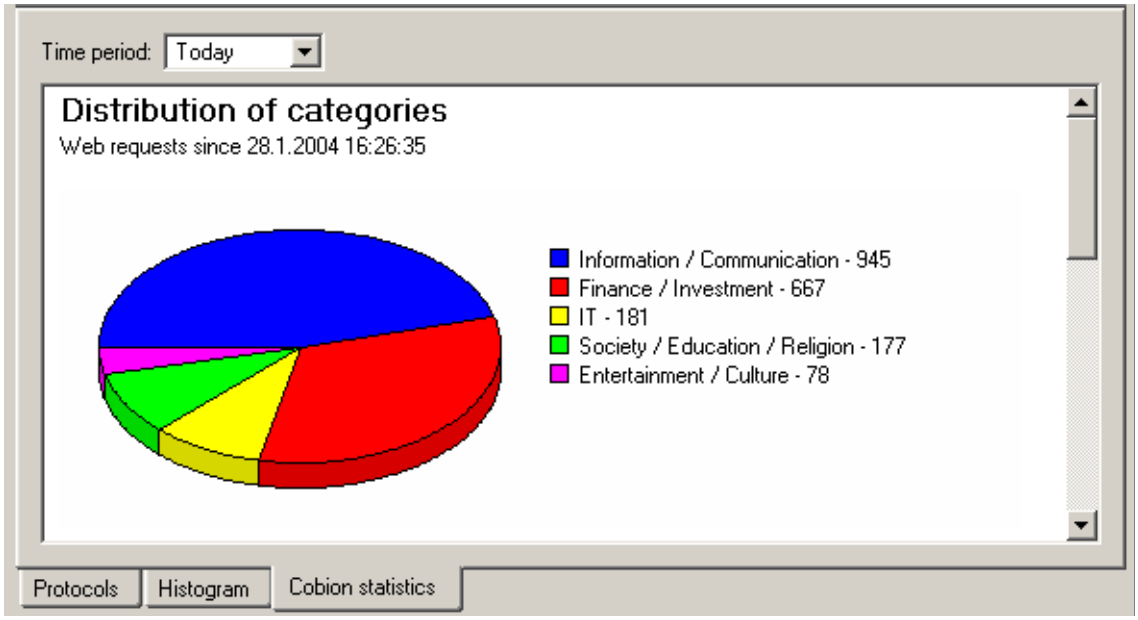
Note: In this case, the term "line" represents traffic of a user performed through *WinRoute*. Communication between hosts within the local network are not detected.

Select an item from the *Time interval* combo box to specify a time period which the chart will refer to (for details, see chapter 15.4). The x axis of the chart represents time and the y axis represents traffic speed. The x axis is measured accordingly to a selected time period, while measurement of the y axis depends on the maximal value of the time interval and is set automatically (bytes per second is the basic measure unit — *B/s*).

Classification of Web pages

The *Cobion statistics* tab provides pie chart and table statistics for Cobion categorized Web pages opened by a specific user in a selected time interval.





Select an item from the *Time interval* combo box to specify a time period which the statistics will refer to (see chapter 15.2).

The pie chart provides proportional participation of top five Web categories in the entire traffic performed in a selected period (according to number of requests). These categories are listed in the chart clue, including corresponding numbers of requests. Other categories are represented by the *Other* item.

The table below the chart provides a list of all *Cobion* categories (using [+] or [-] next to names of categories you can show or hide individual subcategories). The following information is provided for each (sub)category:

- number of requests included into this category by the *Cobion* filter,
- percentage in participation of these requests on the total number of requests.

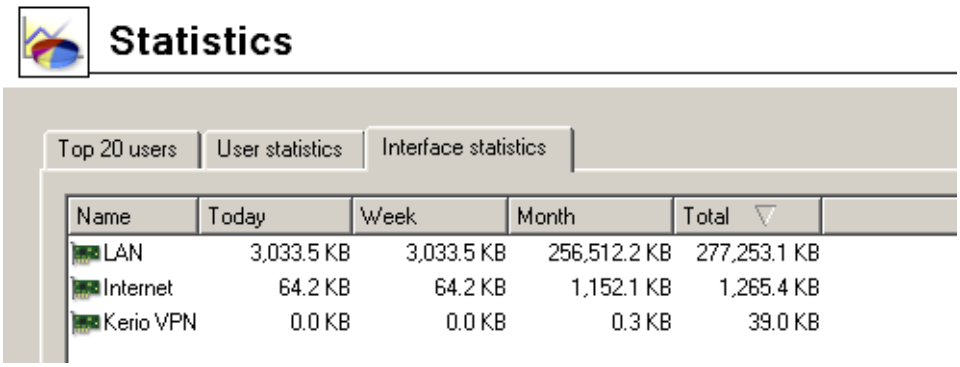
Category	Requests	%
⊞ Pornography / Nudity	0	0.0
⊞ Ordering	0	0.0
⊞ Society / Education / Religion	177	0.0
⊞ Criminal Activities	0	0.0
⊞ Extreme	0	0.0
⊞ Games / Gambling	0	0.0
⊞ Entertainment / Culture	78	0.0
⊞ Information / Communication	945	0.1
⊞ IT	181	0.0
⊞ Drugs	0	0.0
⊞ Lifestyle	0	0.0
⊞ Private Homepages	0	0.0
⊞ Job Search	0	0.0
⊞ Finance / Investment	667	0.1
⊞ Vehicles / Transportation	0	0.0
⊞ Weapons	0	0.0
⊞ Medicine	0	0.0

15.4 Interface statistics

The *Interface statistics* tab in *Status / Statistics* provides detailed information on volume of data transmitted in both directions through individual interfaces of the *WinRoute* host in selected time intervals (today, this week, this month, total).

Note: Interfaces can be represented by network adapters, dial-ups or VPN tunnels. *VPN server* is a special interface — communication of all VPN clients is represented by this item in *Interface statistics*.

Additional columns may be added through the context menu that provide volume of data transmitted through a corresponding interface, incoming (*IN*) or/and outgoing (*OUT*), for various time periods.

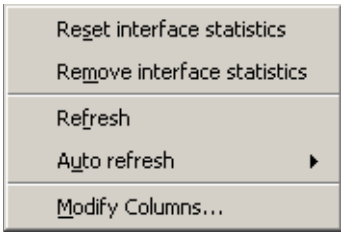


Example: The *WinRoute* host connects to the Internet through the *Public* interface and the local network is connected to the *LAN* interface. A local user downloads 10 MB of data from the Internet. This data will be counted as follows:

- *IN* at the *Public* interface is counted as an *IN* item (data from the Internet was received through this interface),
- at the *LAN* interface as *OUT* (data was sent to the local network through this interface).

Note: Interface statistics are saved into the `interfaces.stat` file in the *WinRoute* directory. This implies that they are not reset when the *WinRoute Firewall Engine* is closed.

A context menu providing the following options will be opened upon right-clicking anywhere in the table (or on a specific interface):



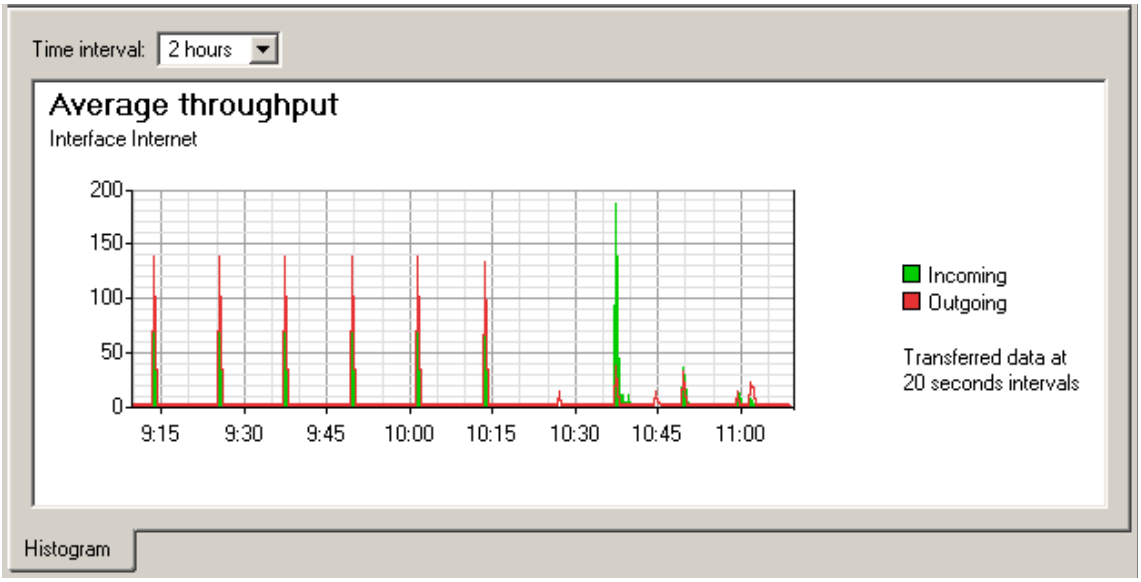
- *Reset interface statistics* — this option resets all statistics for the interface. It is available only if the mouse pointer is hovering an interface at the moment when the context menu is opened.
- *Refresh* — use this option to update information in the *Interface statistics* tab (this option is identical to the *Refresh* button at the bottom of the window).
- *Auto refresh* — settings for automatic updates of the data provided in the *Interface statistics* tab. The interval from 5 seconds to 1 minute can be used, or automatic refreshing of the page can be disabled by the *No refresh* option.

- *Modify columns* — use this option to select items (columns) which will be displayed in the table (see chapter 3.2).

- *Remove interface statistics* — removes a selected interface from the statistic overview. Only inactive interfaces (i.e. disconnected network adapters, hung-up dial-ups, disconnected VPN tunnels or VPN servers which no client is currently connected to) can be removed.

Graphical view of interface load

The traffic processes for a selected interface (transfer speed in *B/s*) and a specific time period can be viewed in the chart provided in the bottom window of the *Interface statistics* tab. Use the *Show details / Hide details* button to show or hide this chart (the show mode is set by default).



The period (*2 hours*, *1 day*, *1 week*, *1 month*) can be selected in the *Time interval* box. The current time and date is considered as the end point of each period (i.e. *2 hours* means the last two hours).

The x axis of the chart represents time and the y axis represents traffic speed. The x axis is measured accordingly to a selected time period, while measurement of the y axis depends on the maximal value of the time interval and is set automatically (bytes per second is the basic measure unit — *B/s*).

The clue on the right side of the chart provides the interval which is used for individual time impulses.

Example: Suppose the *1 day* interval is selected. Then, an impulse unit is represented by 5 minutes. This means that every 5 minutes an average traffic speed for the last 5 minutes is recorded in the chart.

Chapter 16

Logs

Logs are files where history of certain events performed through or detected by *WinRoute* are recorded and kept.

Each log is displayed in a window in the *Logs* section. Each event is represented by one record line. The lines contain time information in brackets (date and time when the event started) followed by information about the event according to the log type.

Events of individual logs can be optionally saved to files on a local drive and/or to a *Syslog* server.

Locally, the logs are saved in the files under the `logs` subdirectory where *WinRoute* is installed. The file names have this pattern:

```
file_name.log
```

(e.g. `debug.log`). Each log also includes a file with the `.idx` extension. This index file enables smoother access to logs in the *Kerio Administration Console*.

Individual logs can be rotated — after a certain time period or when a threshold of the file size is reached, log files are stored and new events are logged to a new (empty) file.

Storing logs in files enables permanent backup of logs (by copying them to another directory). Logs can be also analyzed using various analysis tools.

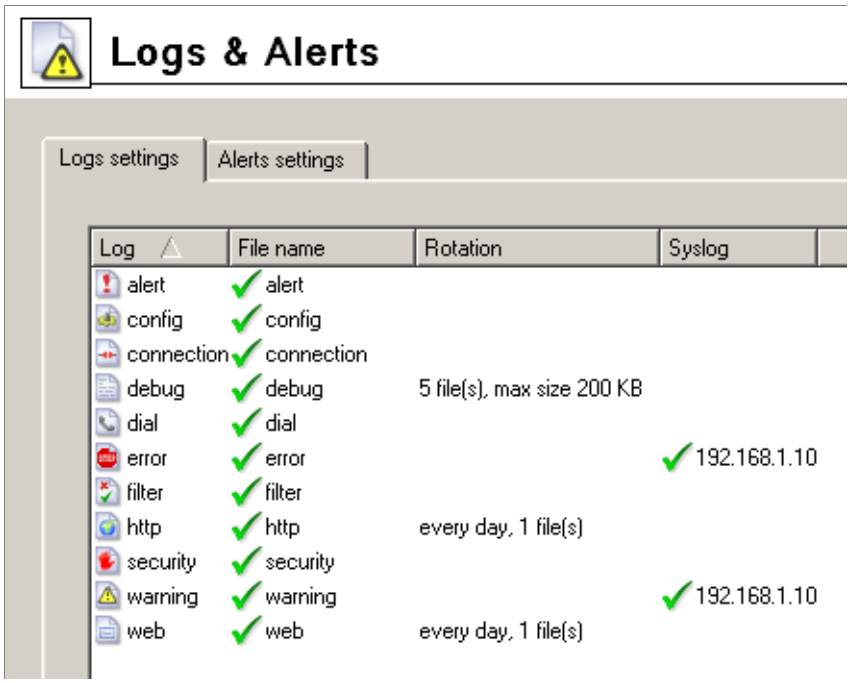
16.1 Log Settings

Log parameters (file names, rotation, sending to a *Syslog* server) can be set in the *Configuration / Log Settings* section. In this section of the guide an overview of all logs used by *WinRoute* are provided.

Double-click on a selected log (or select a log and click on the *Edit* button) to open a dialog where parameters for the log can be set.

File Logging

Use the *File Logging*tab to define file name and rotation parameters.



Enable logging to file Use this option to enable/disable logging to file according to the *File name* entry (the `.log` extension will be appended automatically).

If this option is disabled, none of the following parameters and settings will be available.

Rotate regularly Set intervals in which the log will be rotated regularly. The file will be stored and a new log file will be started in selected intervals.

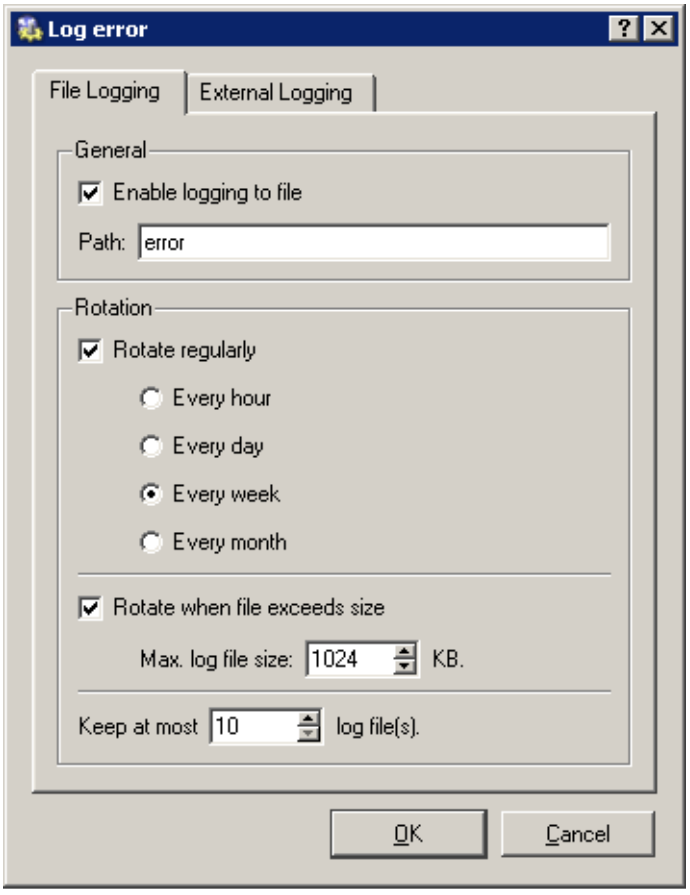
Rotate when file exceeds size Set a maximal size for each file. Whenever the threshold is reached, the file will be rotated. Maximal size is specified in kilobytes (KB).

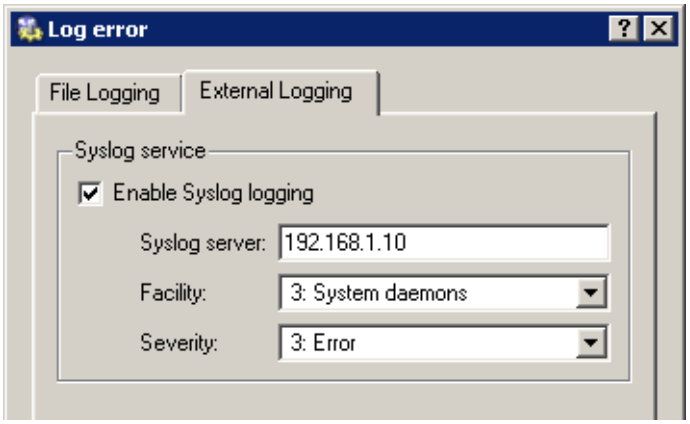
Note: If both *Rotate regularly* and the *Rotate when file exceeds size* are enabled, the particular file will be rotated whenever one of these conditions is met.

Keep at most ... log file(s) Maximal count of log files that will be stored. Whenever the threshold is reached, the oldest file will be deleted.

Syslog Logging

Parameters for logging to a *Syslog* can be defined in the *External Logging* tab.





Enable Syslog logging Enable/disable logging to a *Syslog* server.

If this option is disabled, the following entries will be unavailable.

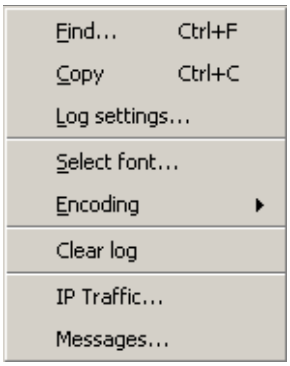
Syslog server IP address of the *Syslog* server.

Facility Facility that will be used for the particular *WinRoute* log (depends on the *Syslog* server).

Severity Severity of logged events (depends on the *Syslog* server).

16.2 Logs Context Menu

Right-click in any of the log windows to view the context menu. Here you can select from various functions or edit log parameters.



Find Use this option to search for a string in the log. Logs can be scanned either *Up* (search for older events) or *Down* (search for newer events) from the current position.

Copy This function copies selected text to the clipboard. A key shortcut from the operating system can be used (e.g. *Ctrl+C* or *Ctrl+Insert* in Windows).

Log settings A dialog where log parameters such as log file name, rotation and *Syslog* parameters can be set. These parameters can also be set in the *Log settings* tab under *Configuration / Logs and Alerts*. For detailed information, refer to chapter 16.1.

Font Within this dialog you can select a font of the log printout. All fonts installed on the host with the *Kerio Administration Console* are available.

Charset coding Coding that will be used for the log printout in *Kerio Administration Console* can be selected in this section. *UTF-8* is used by default.

TIP: Select a new encoding type if special characters are not printed correctly in non-English versions.

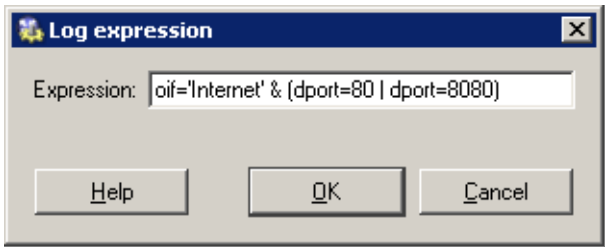
Log settings A dialog where log parameters such as log file name, rotation and *Syslog* parameters can be set. For detailed information refer to chapter 16.1.

Remove Log Removes entire log. The file will be removed (not only the information saved in the selected window).

Warning: Removed logs cannot be refreshed anymore.

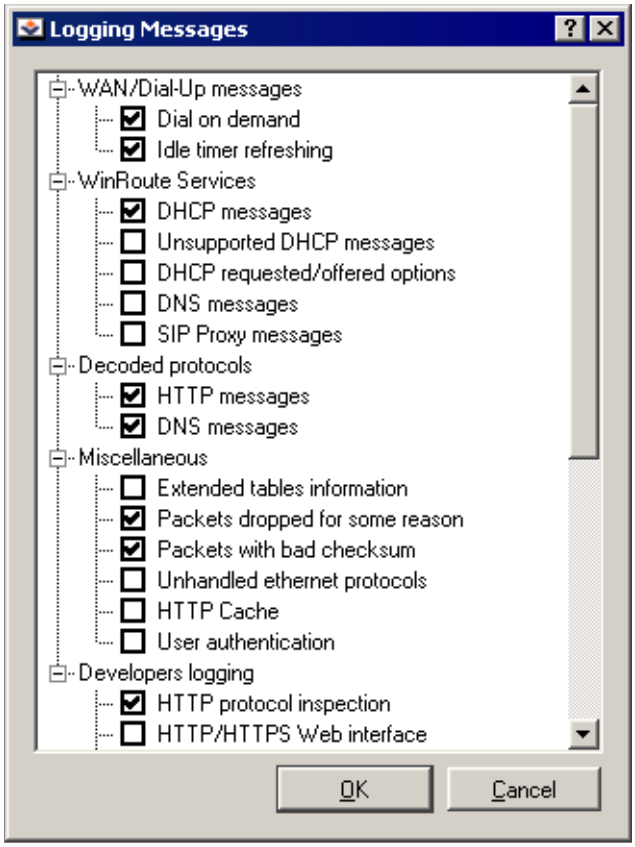
The following options are available in the *Debug* log only:

IP Traffic This function enables monitoring of packets according to the user defined log expression.



The expression must be defined with special symbols. After clicking on the *Help* button, a brief description of possible conditions and examples of their use will be displayed.

Messages This option enables the administrator to define advanced settings for information that will be monitored:



- *WAN / Dial-up messages* — information about dialed lines (request dialing, auto disconnection down-counter) and outgoing connections
- *WinRoute services* — protocols processed by services (DHCP, DNS, VPN)
- *Decoded protocols* — displays message content of all selected protocols that use *WinRoute* modules (HTTP and DNS)
- *Miscellaneous* — more information, such as information about removed packets, packets with errors, HTTP cache, user authentication, etc.
- *Developers logging* — detailed logs for debugging (can be used e.g. when resolving problems with help from technical support)

Note: If a user possessing only the read rights is connected to the *WinRoute Firewall Engine* (see chapter 10.1), the *Log settings*, *Clear log*, *IP Traffic* and *Messages* options are not available in the *Debug* context menu. Only users with full rights can access these functions.

16.3 Alert Log

The *Alert* log provides a complete history of alerts generated by *WinRoute* (e.g. alerts upon virus detection, dialing and hanging-up, reached quotas, detection of P2P networks, etc.).

Each event in the *Alert* log includes a time stamp (date and time when the event was logged) and information about an alert type (in capitals). The other items depend on an alert type.

TIP: Email and SMS alerts can be set under *Configuration / Logs & Alerts*. All sent alerts can be viewed in the *Status / Alert messages* section (for details, see chapter 14.3).

16.4 Config Log

The *Config* log stores a complete communication history between *Kerio Administration Console* and the *WinRoute Firewall Engine* — the log allows you to find out what administration actions were performed by which user, and when.

The following three types of records are written to the *Config* log:

1. *Information about user logins/logouts to/from the WinRoute's administration*

   Example:

```
[18/Apr/2003 10:25:02] standa - session opened
 for host 192.168.32.100
[18/Apr/2003 10:32:56] standa - session closed
 for host 192.168.32.100
```

- `[18/Apr/2003 10:25:02]` — date and time when the record was written to the log
- `jsmith` — the login name of the user
- `session opened for host 192.168.32.100` — information about the beginning of the communication and the IP address of the computer from which the user connected
- `session closed for host 192.168.32.100` — information about the end of the communication with the particular computer (user logout or *Kerio Administration Console* closed)

2. *Configuration database changes*

   This type of record informs about changes performed by the user in the *Kerio Administration Console*. A simplified form of the SQL language is used when communicating with the database.

   Example:

```
[18/Apr/2003 10:27:46] jsmith - insert StaticRoutes
 set Enabled='1', Description='VPN',
 Net='192.168.76.0', Mask='255.255.255.0',
 Gateway='192.168.1.16', Interface='LAN', Metric='1'
```

   - `[18/Apr/2003 10:27:46]` — date and time when the record was written
   - `jsmith` — the login name of the user
   - `insert StaticRoutes ...` — the particular command used to modify the *WinRoute's* configuration database (in this case, a static route was added to the routing table)

3. *Other configuration changes*

   A typical example of this record type is the change of traffic rules. When the user hits *Apply* in *Configuration / Traffic policy*, a complete list of current traffic rules is written to the *Config* log.

Example:

```
[18/Apr/2003 12:06:03] Admin - New traffic policy set:
[18/Apr/2003 12:06:03] Admin - 1:  name=(ICMP Traffic)
 src=(any) dst=(any) service=("Ping")
snat=(any) dnat=(any) action=(Permit),
time_range=(always) inspector=(default)
```

- `[18/Apr/2003 12:06:03]` — date and time of the change
- `Admin` — login name of the user who did the change
- `1:` — traffic rule number (rules are numbered top to bottom according to their position in the table, the numbering starts from 1)
- `name=(ICMP Traffic) ...` — traffic rule definition (name, source, destination, service etc.)

Note: The default rule (the last one in the rule table) is marked with `default:` instead of the positional number.

16.5 Connection Log

Connection logs for traffic rules which are configured to be logged using the *Log matching connections* option (refer to chapter 5).

How to read the Connection Log?

```
[18/Apr/2003 10:22:47] [ID] 613181 [Rule] NAT
 [Service] HTTP [User] james
 [Connection] TCP 192.168.1.140:1193 -> hit.top.com:80
 [Duration] 121 sec [Bytes] 1575/1290/2865 [Packets] 5/9/14
```

- `[18/Apr/2003 10:22:47]` — date and time when the event was logged (Note: Connection logs are saved immediately after a disconnection)
- `[ID] 613181` — *WinRoute* connection identification number
- `[Rule] NAT` — name of the traffic rule which has been used (a rule by which the traffic was allowed or denied).
- `[Service] HTTP` — name of a corresponding application layer service (recognized by destination port).

If the corresponding service is not defined in *WinRoute* (refer to chapter 9.3), the `[Service]` item is missing in the log.

- `[User] james` name of the user connected to the firewall from a host which participates in the traffic (If no user is connected from this host, the `<null>` value is logged)

  If no user is currently connected from the corresponding host, the `[User]` item is missing in the log.

- `[Connection] TCP 192.168.1.140:1193 -> hit.top.com:80` — protocol, source IP address and port, destination IP address and port. If an appropriate log is found in the *DNS Forwarder* cache (see chapter 4.3), the host's DNS name is displayed instead of its IP address. If the log is not found in the cache, the name is not detected (such DNS requests would slow *WinRoute* down).

- `[Duration] 121 sec` — duration of the connection (in seconds)

- `[Bytes] 1575/1290/2865` — number of bytes transferred during this connection (transmitted/accepted/total)

- `[Packets] 5/9/14` — number of packets transferred through this connection (transmitted/accepted/total)

16.6 Debug Log

Debug (debug information) is a special log which can be used to monitor certain kinds of information, especially for problem-solving. Too much information could be confusing and impractical if displayed all at the same time. Usually, only a part of the information or functions is relevant. In addition, displaying too much information slows *WinRoute's* performance. Therefore, it is strongly recommended to monitor an essential part of information and during the shortest possible period only.

16.7 Dial Log

Data about dialing and hanging up the dial-up lines, and about time spent on-line.

The following items (events) can be reported in the *Dial* log:

1. Manual connection (from the *Administration Console* — see chapter 4.1, using the Web interface — refer to chapter 8.6 or right from the operating system)

   ```
   [15/Mar/2004 15:09:27] Line "Connection" dialing,
    console 127.0.0.1 - Admin
   ```

```
[15/Mar/2004 15:09:39] Line "Connection" successfully connected
```

The first log item is reported upon initialization of dialing. The log always includes *WinRoute* name of the dialed line (see chapter 4.1). If the line is dialed from the *Administration Console* or the Web interface, the log provides this additional information:

- where the line was dialed from (`console` — *Administration Console*, `webadmin` — Web interface,
- IP address of the client (i.e. IP address of the *Administration Console* or of the Web interface),
- login name of the user who sent the dial request.

Another event is logged upon a successful connection (i.e. when the line is dialed, upon authentication on a remote server, etc.).

2. Line disconnection (manual or automatic, performed after a certain period of idleness)

```
[15/Mar/2004 15:29:18] Line "Connection" hang-up,
 console 127.0.0.1 - Admin
[15/Mar/2004 15:29:20] Line "Connection" disconnected,
 connection time 00:15:53, 1142391 bytes received,
 250404 bytes transmitted
```

The first log item is recorded upon reception of a hang-up request. The log provides information about interface name, client type, IP address and username.

The second event is logged upon a successful hang-up. The log provides information about interface name, time of connection (`connection time`), volume of incoming and outgoing data in bytes (`bytes received` and `bytes transmitted`).

3. Disconnection caused by an error (connection is dropped)

```
[15/Mar/2004 15:42:51] Line "Connection" dropped,
 connection time 00:17:07, 1519 bytes received,
 2504 bytes transmitted
```

The second event is logged upon a successful hang-up. The log provides information about interface name, time of connection (`disconnection time`), volume of incoming and outgoing data in bytes (`bytes received` and `bytes transmitted`).

4. Requested dialing (as a response to a DNS query)

```
[15/Mar/2004 15:51:27] DNS query for "www.microcom.com"
 (packet UDP 192.168.1.2:4567 -> 195.146.100.100:53)
 initiated dialing of line "Connection"
[15/Mar/2004 15:51:38] Line "Connection" successfully connected
```

   The first log item is recorded upon reception of a DNS request (the *DNS forwarder* has not found requested DNS record in its cache). The log provides:

   - DNS name from which IP address is being resolved,
   - description of the packet with the corresponding DNS query (protocol, source IP address, source port, destination IP address, destination port),
   - name of the line to be dialed.

   The second record informs on successful connection (i.e. when the line is dialed, after authentication at a remote server, etc.).

5. On-demand dialing (response to a packet sent from the local network)

```
[15/Mar/2004 15:53:42] Packet
 TCP 192.168.1.3:8580 -> 212.20.100.40:80
 initiated dialing of line "Connection"
[15/Mar/2004 15:53:53] Line "Connection" successfully connected
```

   The first record is logged when *WinRoute* finds out that the route of the packet does not exist in the routing table. The log provides the following information:

   - description of the packet (protocol, source IP address, destination port, destination IP address, destination port),
   - name of the line to be dialed.

   The second log item is recorded upon a successful connection (i.e. when the line is dialed, after authentication at a remote server, etc.).

6. Connection error (e.g. error at the modem was detected, dial-up was disconnected, etc.)

```
[15/Mar/2004 15:59:08] DNS query for "www.microsoft.com"
 (packet UDP 192.168.1.2:4579 -> 195.146.100.100:53)
 initiated dialing of line "Connection"
```

```
[15/Mar/2004 15:59:12] Line "Connection" disconnected
```

The first record represents a DNS record sent from the local network, from that the line is to be dialed (see above).

The second log item (immediately after the first one) informs that the line has been hung-up. Unlike in case of a regular disconnection, time of connection and volume of transmitted data are not provided (because the line has not been connected).

16.8 Error Log

The *Error* log displays information about serious errors that affect the functionality of the entire firewall. *WinRoute's* administrator should monitor the error log on a regular basis and solve the detected errors as quickly as possible, otherwise some (or even all) firewall services could become unavailable to the users and/or security problems can arrise.

A typical error message in the *Error* log could be: a problem when starting a service (usually a collision at a particular port number), problems when writing to the disc or when initializing anti-virus, etc.

Each record in the *Error* log contains error code and sub-code as two numbers in parentheses `(x y)`. The error code may fall into one of the following categories:

- `1-999` — system resources problem (insufficient memory, memory allocation error, etc.)
- `1000-1999` — internal errors (unable to read routing table or interface IP addresses, etc.)
- `2000-2999` — license problems (license expired, the number of users would break license limit, unable to find license file, etc.)
- `3000-3999` — configuration errors (unable to read configuration file, detected a look in the configuration of *DNS Forwarder* or the *Proxy server*, etc.)
- `4000-4999` — network (socket) errors
- `5000-5999` — errors while starting or stopping the *WinRoute Firewall Engine* (problems with low-level driver, problems when initializing system libraries, services, configuration databases, etc.)
- `6000-6999` — filesystem errors (cannot open/save/delete file)
- `7000-7999` — SSL errors (problems with keys and cerificates, etc.)

- 8000–8099 — HTTP cache errors (errors when reading/writing cache files, not enough space for cache, etc.)
- 8100–8199 — errors of the *Cobion* system
- 8200–8299 — authentication subsystem errors
- 8300–8399 — anti-virus module errors (anti-virus test not successful, problems when storing temporary files, etc.)
- 8400–8499 — dial-up error (unable to read defined dial-up connections, line configuration error, etc.)
- 8500–8599 — LDAP errors (server not found, login failed, etc.)

Note: If you are not able to correct an error (or figure out what it is caused by) which is repeatedly reported in the *Error* log, do not hesitate to contact our technical support. For detailed information, refer to chapter 18 or to http://www.kerio.com/.

16.9 Filter Log

This log contains information about web pages and objects blocked by the HTTP and FTP filters (see chapters 6.1 and 6.5) and about packets blocked by traffic rules if packet logging is enabled for the particular rule (see chapter 5 for more information). Each log line includes the following information depending on the component which generated the log:

- when an HTTP or FTP rule is applied: rule name, user, IP address of the host which sent the request, object's URL
- when a traffic rule is applied: detailed information about the packet that matches the rule (rule name, source and destination address, ports, size, etc.)

Example of a URL rule log message:

```
[18/Apr/2003 13:39:45] ALLOW URL 'McAfee update'
 192.168.64.142 standa HTTP GET
 http://update.kerio.com/nai-antivirus/datfiles/4.x/dat-4258.zip
```

- [18/Apr/2003 13:39:45] — date and time when the event was logged
- ALLOW — action that was executed (ALLOW = access allowed, DENY = access denied)

- URL — rule type (for URL or FTP)
- 'McAfee update' — rule name
- 192.168.64.142 — IP address of the client
- jsmith — name of the user authenticated on the firewall (no name is listed unless at least one user is logged in from the particular host)
- HTTP GET — HTTP method used in the request
- http:// ... — requested URL

Example of a traffic rule log message:

```
[16/Apr/2003 10:51:00] PERMIT 'Local traffic' packet to LAN,
 proto:TCP, len:47, ip/port:195.39.55.4:41272 ->
 192.168.1.11:3663, flags:  ACK PSH , seq:1099972190
 ack:3795090926, win:64036, tcplen:7
```

- [16/Apr/2003 10:51:00] — date and time when the event was logged
- PERMIT — action that was executed with the packet (PERMIT, DENY or DROP)
- Local traffic —the name of the traffic rule that was applied
- packet to — packet direction (either to or from a particular interface)
- LAN — interface name (see chapter 4.1 for details)
- proto: — transport protokol (TCP, UDP, etc.)
- len: — packet size in bytes (including the headers) in bytes
- ip/port: — source IP address, source port, destination IP address and destination port
- flags: — TCP flags
- seq: — sequence number of the packet (TCP only)
- ack: — aknowledgement sequence number (TCP only)

- `win:` — size of the receive window in bytes (it is used for data flow control — TCP only)
- `tcplen:` — TCP payload size (i.e. size of the data part of the packet) in bytes (TCP only)

16.10 HTTP Log

This log contains all HTTP requests that were processed by the HTTP inspection module (see section 9.3) or by the built-in proxy server (see section 4.5). The log has the standard format of either the *Apache* WWW server (see `http://www.apache.org/`) or of the *Squid* proxy server (see `http://www.squid-cache.org/`). The enable or disable the HTTP log, or to choose its format, go to *Configuration/ContentFiltering/HTTP Policy* (refer to section 6.1 for details).

Notes:

1. Only accesses to allowed pages are recorded in the *HTTP* log. Request that were blocked by HTTP rules are logged to the *Filter* log (see above), if the particular rule has the logging enabled (see section 6.1).
2. The *HTTP* log is intended to be processes by external analytical tools. The *Web* log (see bellow) is better suited to be viewed by the *WinRoute* administrator.

An example of HTTP log record that follows the Apache format:

```
[18/Apr/2003 15:07:17] 192.168.64.64 - rgabriel
 [18/Apr/2003:15:07:17 +0200]
 "GET http://www.kerio.com/ HTTP/1.1" 304 0 +4
```

- `[18/Apr/2003 15:07:17]` — date and time when the event was logged
- `192.168.64.64` — IP address of the client host
- `rgabriel` — name of the user authenticated through the firewall (a dash is displayed if no user is authenticated through the client)
- `[18/Apr/2003:15:07:17 +0200]` — date and time of the HTTP request. The +0200 value represents time difference from the UTC standard (+2 hours are used in this example — CET).
- `GET` — used HTTP method

- `http://www.kerio.com` — requested URL
- `HTTP/1.1` — version of the HTTP protocol
- `304` — return code of the HTTP protocol
- `0` — size of the transferred object (file) in bytes
- `+4` — count of HTTP requests tranferred through the connection

An example of HTTP log record that follows the Squid format:

```
1058444114.733 0 192.168.64.64 TCP_MISS/304 0
 GET http://www.squid-cache.org/ - DIRECT/206.168.0.9
```

- `1058444114.733` — timestamp (seconds and miliseconds since January 1st, 1970)
- `0` — download duration (not measured in *WinRoute*, always set to zero)
- `192.168.64.64` — client IP address
- `TCP_MISS` — the TCP protocol was used and the particular object was not found in the cache ("missed"). *WinRoute* always uses this value for this field.
- `304` — HTTP response code
- `0` — trasferred data amount in bytes (HTTP object size)
- `GET http://www.squid-cache.org/` — the HTTP request (HTTP method and URL of the object)
- `DIRECT` — the WWW server access method (*WinRoute* always uses direct access)
- `206.168.0.9` — IP address of the WWW server

16.11 Security Log

A log for security-related messages. Records of the following types may appear in the log:

1. *Anti-spoofing log records*

   Messages about packets that where captured by the *Anti-spoofing* module (packets with invalid source IP address — see section 11.5 for details)

   Example:

```
[17/Jul/2003 11:46:38] Anti-Spoofing:
 Packet from LAN, proto:TCP, len:48,
 ip/port:61.173.81.166:1864 -> 195.39.55.10:445,
 flags:  SYN , seq:3819654104 ack:0, win:16384, tcplen:0
```

- `packet from` — packet direction (either from `interface` or to `interface`)
- `LAN` — the name of the interface where the packet was captured (see section 4.1 for details)
- `proto:` — transport protocol (TCP, UDP, etc.)
- `len:` — packet length in bytes (including headers)
- `ip/port:` — source IP address and port and destination IP address and port
- `flags:` — TCP flags (TCP only)
- `seq:` — sequence number (TCP only)
- `ack:` — acknowledgement sequence number (TCP only)
- `win:` — size of the receive window (TCP only)
- `tcplen:` — TCP payload length in bytes

2. *FTP protocol parser log records*

   Example 1:

```
[17/Jul/2003 11:55:14] FTP: Bounce attack attempt:
 client:  1.2.3.4, server:  5.6.7.8,
 command:  PORT 10,11,12,13,14,15
```

   (attack attempt detected — a foreign IP address in the PORT command)

   Example 2:

```
[17/Jul/2003 11:56:27] FTP: Malicious server reply:
 client:  1.2.3.4, server:  5.6.7.8,
 response:  227 Entering Passive Mode (10,11,12,13,14,15)
```

   (suspicious server reply with a foreign IP address)

3. *Failed user authentication log records*

   Message format:

   ```
   Authentication: <service>: Client: <IP address>: <reason>
   ```

   - `<service>` — The *WinRoute* service to which the user attempted to authenticate (`Admin` = administration using *Kerio Administration Console*, `WebAdmin` = web administration interface, `WebAdmin SSL` = secure web administration interface, `Proxy` = proxy server user authentication)
   - `<IP address>` — IP address of the computer from which the user attempted to authenticate
   - `<reason>` — reason of the authentication failure (nonexistent user / wrong password)

   Note: For detailed information regarding user authentication see section 10.1 and 8.2.

4. *Information about the start and shutdown of the WinRoute Firewall Engine*

   a) Engine Startup:

   ```
   [17/Jul/2003 12:11:33] Engine: Startup.
   ```

   b) Engine Shutdown:

   ```
   [17/Jul/2003 12:22:43] Engine: Shutdown.
   ```

16.12 Warning Log

Warning reports are displayed in the *Warning* log. Reports included in this section represent serious errors and bugs. Warnings can display for example reports about invalid user login (invalid username or password), error in communication of the server and Web administration interface, etc.

Events recalling warning messages in this log do not seriously affect *WinRoute* functionality. However, they can point at current or possible problems. The *Warning* log can be used for example when a user has problems with functioning of some services.

Each warning message is identified by its numerical code `(code xxx:)`. The following warning categories are defined:

- `1000-1999` — system warnings (e.g. an application found that is known as conflicting)
- `2000-2999` — *WinRoute* configuration problems (e.g. HTTP rules reqire user authentication, but the WWW interface is not enabled)

- 3000-3999 — warning from individual *WinRoute* modules (e.g. DHCP server, anti-virus check, etc.)
- 4000-4999 — license warnings (subscription expiration, forthcoming expiration of *WinRoute's* license, *Cobion* license, or the *McAfee* anti-virus license)

  Note: License expiration is considered to be an error and it is logged into the *Error* log.

Examples of Warning *logs:*

```
[15/Apr/2003 15:00:51] (3004) Authentication subsystem warning:
Kerberos 5 auth:

 user james@company.com not authenticated

[15/Apr/2003 15:00:51] (3004) Authentication subsystem warning:
Invalid password for user admin

[16/Apr/2003 10:53:20] (3004) Authentication subsystem warning:
User johnblue doesn't exist
```

- The first log informs that authentication of user `jsmith` by the *Kerberos* system in the `company.com` domain failed
- The second log informs on a failed authentication attempt by user `admin` (invalid password)
- The third log informs on an authentication attempt by a user which does not exist (`johnblue`)

Note: With the above three examples, the relevant records will also appear in the `Security` log.

16.13 Web Log

This log displays HTTP requests processed either by HTTP protocol inspectors (see chapter 9.3) or by the embedded proxy server (see chapter 4.5). Unlike in the *HTTP* log, the *Web* log displays only the title of a page and the *WinRoute* user or the IP host viewing the page.

For administrators, the *Web* log is easy to read and it provides the possibility to monitor which Websites were opened by each user.

How to read the Web *Log?*

```
[24/Apr/2003 10:29:51] 192.168.44.128 james
 "Kerio Technologies | No Pasaran!" http://www.kerio.com/
```

- `[24/Apr/2003 10:29:51]` — date and time when the event was logged
- `192.168.44.128` — IP address of the client host
- `james` — name of authenticated user (if no user is authenticated through the client host, the name is substituted by a dash)
- `"Kerio Technologies | No Pasaran!"` — page title

  (content of the `<title>` HTML tag)

  Note: If the page title cannot be identified (i.e. for its content is compressed), the `"Encoded content"` will be reported
- `http://www.kerio.com/` — URL pages

Chapter 17

Troubleshooting

This chapter provides several helpful tips for solving of problems which might arise during *WinRoute* deployment.

17.1 Backup and Import of Configuration

All *WinRoute* configuration data is stored in the following files under the same directory where *WinRoute* is installed:

winroute.cfg Chief configuration file

users.cfg Information about groups and user accounts.

logs.cfg Log configurations

host.cfg Preferences for backs-up of configuration, user accounts data, DHCP server database, etc.

ids.cfg Stored for future use.

The data in these files are saved in XML format so that it can be easily modified by an advanced user or generated automatically using another application. Configuration back-up can be done by copying the files (for details see below).

Warning

Stop *WinRoute Firewall Engine* before handling configuration files as the files are retrieved during the *WinRoute Firewall Engine* startup only. Configuration files are saved after any modification is done and after the *Engine* is stopped. All modifications done during *Engine* performance will be overwritten by the configuration in the system memory when the *Engine* is stopped.

Configuration Recovery Performed Through Back-Up

To recover configuration through backed-up data (typically this need may arise when *WinRoute* is installed to a new workstation or when the operating system is being reinstalled), follow these steps:

1. Perform *WinRoute* installation on a required machine (refer to chapter 2.3).

2. Stop *WinRoute Firewall Engine*.

3. Copy backed-up configuration files `host.cfg`, `logs.cfg`, `users.cfg` and `winroute.cfg` into the *WinRoute* installation directory

   (typically `C:\Program Files\Kerio\WinRoute Firewall`).

4. Run *WinRoute Firewall Engine*.

   At this stage, *WinRoute* detects the required configuration file. Within this process, unknown network interfaces (ones which are not defined in the `winroute.cfg` configuration file) will be detected in the system. Each network interface includes a unique (randomly generated) identifier in the operating system. It is almost not possible that two identifiers were identical.

   To avoid setting up new interfaces and changing traffic rules, you can assign new identifiers to original interfaces in the `winroute.cfg` configuration file.

5. Stop *WinRoute Firewall Engine*.

6. Use a plaintext editor (e.g. *Notepad*) to open the `winroute.cfg` configuration file. Go to the following section:

   ```
   <list name="Interfaces">
   ```

   Scan this section for the original adapter. Find an identifier for a new interface in the new adapter's log and copy it to the original adapter. Remove the new interface's log.

   Example: Name of the local network interface is *LAN*. This network connection is labeled as *Local area connection* in the new operating system. Now, the following data can be found in the `Interfaces` section (only the essential parts are listed):

   ```
   <listitem>
   <variable name="Id">\DEVICE\
   {7AC918EE-3B85-5A0E-8819-CBA57D4E11C7}</variable>
   <variable name="Name">LAN</variable>
   ```

```
...
</listitem>
<listitem>
<variable name="Id">\DEVICE\
{6BF377FB-3B85-4180-95E1-EAD57D5A60A1}</variable>
<variable name="Name">Local Area Connection</variable>
...
</listitem>
```

Copy the `Local Area Connection` interface's identifier into the `LAN` interface. Remove the data for `Local Area Connection` (a relevant `listitem` section).

When all these changes are performed, the data in the configuration file relating to interface connected to the local network will be as follows:

```
<listitem>
<variable name="Id">\DEVICE\
{6BF377FB-3B85-4180-95E1-EAD57D5A60A1}</variable>
<variable name="Name">LAN</variable>
...
</listitem>
```

7. Save the `winroute.cfg` file and run *WinRoute Firewall Engine*.

   Now, the *WinRoute* configuration is identical with the original *WinRoute* configuration on the prior operating system.

17.2 Partial Retirement of Protocol Inspector

Under certain circumstances, appliance of a protocol inspector to a particular communication might be undesirable. To disable specific protocol inspection, take the following steps:

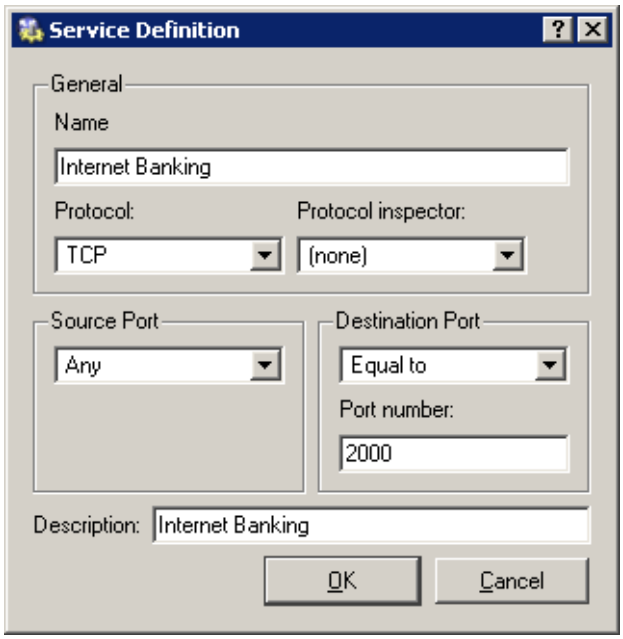
- create a service, definition of which will not include protocol inspection
- define a traffic rule for the service, including relevant source and destination addresses

Example: A banking application communicates with the bank's server through its proper protocol which uses TCP protocol at the port 2000. This port is used by the *Cisco SCCP*

protocol. If a protocol inspector is applied to an irrelevant protocol (protocol to which the inspector is not assigned), the communication will not work.

Supposing the banking application is run on a host with IP address `192.168.1.15` and it connects to the server `server.bank.com`.

1. In the *Configuration / Definitions / Services* section, define a service called *Internet Banking*: this service will use TCP protocol at the port 2000 and no protocol inspector is applied to this communication.



2. In the *Configuration / Traffic Policy* section, create a rule which will permit this service traffic between the local network and the bank's server.

Chapter 18

Technical Support

Free email and telephone technical support is provided for *Kerio WinRoute Firewall*. For contacts see the final section of this chapter. Should any issue arise *Kerio Technologies* technical staff is ready to help you.

Before you contact our technical support, please take the following steps:

- Search through this guide to find an answer. Individual chapters describe features and parameters of *WinRoute* components in detail.
- If you have not found answers here, try to find it in the *Technical Support* section of the Kerio Technologies website.

If you have not find answers to all your questions and you still intend to contact our technical support, read through the following section which will provide you with a few guidelines.

18.1 Essential Information

To be able to help you solve your problems the best and in the shortest possible time our technical support will require your configuration data and as clear information on your problem as possible. The following information should be provided in your email message:

Description

Clearly describe your problem. Provide as much information on the problem as possible (i.e. whether the issue arose after you had installed a new product version, after an upgrade, etc.).

Informational File

You can use the *Kerio Administration Console* to create a text file including your *WinRoute* configuration data. Take the following steps to generate the file:

- Run *WinRoute Firewall Engine* and connect to it through the *Kerio Administration Console*.

- If you use dial-up, connect to the Internet.
- In the *Kerio Administration Console* use the *Ctrl+S* keys.

The text file will be saved into the home directory of the particular user (i.e. `C:\Documents and Settings\Administrator`) and it will be called `kerio_support_info.txt`.

Note: The `kerio_support_info.txt` is generated by the *Kerio Administration Console*. This implies that in case you connect to the administration remotely, this file will be stored on the computer from which you connect to the *WinRoute* administration (not on the computer/server where the *WinRoute Firewall Engine* is running).

Error Log Files

The `logs` subdirectory will be created in the directory where *WinRoute* is installed (typically `C:\Program Files\Kerio\WinRoute Firewall`). This directory includes the `error.log` and `warning.log` files. Attach these two files to your email to our technical support.

18.2 Contacts

Czech Republic

Kerio Technologies s.r.o.
Anglicke nabrezi 1/2434
301 49 PLZEN
Tel.: +420 377 338 902
E-mail: `support@kerio.cz`
`http://www.kerio.cz/`

USA

United Kingdom

Kerio Technologies Inc.
2041 Mission College Blvd., Suite 100
Santa Clara, CA 95054
Tel.: +1 408 496 4500
E-mail: support@kerio.com
http://www.kerio.com/

Kerio Technologies UK Ltd.
Sheraton House
Castle Park
Cambridge, CB3 0AX
Tel.: +44 1223 370 136, +44 8707 442 205
E-mail: support@kerio.co.uk
http://www.kerio.co.uk/

Chapter 19

Glossary

DHCP DHCP (*Dynamic Host Configuration Protocol*) Serves automatic IP configuration of computers in the network. IP addresses are assigned from a scope. Parameters include a gateway or router, DNS servers, local domain etc.

DNS DNS (*Domain Name System*) A worldwide distributed database of Internet hostnames and their associated IP address. Computers use Domain Name Servers to resolve host names to IP addresses. DNS allows internet servers to be more easily recognized (i.e. `www.kerio.com` is easier to remember than `207.235.5.183`.

Firewall Software application or hardware component used to protect hosts or networks from intrusion attempts (usually from the Internet).

In this guide, the word *firewall* represents the *WinRoute* host.

Protocol inspector *WinRoute's* plug-in (partial program), which is able to monitor communication using application protocols (e.g. HTTP, FTP, MMS, etc.). Protocol inspection is used to check proper syntax of corresponding protocols (mistakes might indicate an intrusion attempt), to ensure its proper functionality while passing through the firewall (e.g. FTP in the active mode, when data connection to a client is established by a server) and to filter traffic by the corresponding protocol (e.g. limited access to Web pages classified by URLs, anti-virus check of downloaded objects, etc.).

Unless traffic rules are set to follow a different policy, each protocol inspector is automatically applied to all connections of the relevant protocol that are processed through *WinRoute*.

IP address Number consisting of 32 bits that is used to identify the host within the Internet. Each packet contains information about where it was sent from (source IP address) and to which address it is to be delivered (destination IP address).

Kerberos It is a standard protocol used for user authentication within Windows 2000. Users connect to central servers (KDC, Key Distribution Center, Windows 2000 domain controller) and the servers send them encrypted keys for connection to other servers within the network.

IPSec *IPsec* (*IP Security Protocol*) is an extended IP protocol which enables secure data transfer. It provides services similar to SSL/TLS, however, these services are provided

on a network layer. IPSec can be used for creation of encrypted tunnels between networks (VPN) — so called tunnel mode, or for encryption of traffic between two hosts— so called transport mode.

Network mask Network masks divide IP addresses into two parts (network address and address of a particular host within the network). Mask have the same form as IP addresses (i.e. `255.255.255.0`), however, its value is needed to be understood as a 32-bit number with certain number of ones on the left end and zeros as the rest. The mask cannot have an arbitrary value. The primary function of a subnet mask is to define the number of IP hosts that participate in an IP subnet. Computers in the same IP subnet should not require a router for network communication.

NAT *NAT* (*Network Address Translation*) stands for substitution of IP addresses in packets passing through the firewall:

- source address translation (*Source NAT, SNAT*) — in packets going from local networks to the Internet source (private) IP addresses are substituted with the external (public) firewall address. Each packet sent from the local network is recorded in the NAT table. If any packet incoming from the Internet matches with a record included in this table, its destination IP address will be substituted by the IP address of the appropriate host within the local network and the packet will be redirected to this host. Packets that do not match with any record in the NAT table will be dropped.

- destination address translation (*Destination NAT, DNAT*, it is also called port mapping) — is used to enable services in the local network from the Internet. If any packet incoming from the Internet meets certain requirements, its IP address will be substituted by the IP address of the local host where the service is running and the packet is sent to this host.

The *NAT* technology enables connection from local networks to the Internet using a single IP address. All hosts within the local network can access the Internet directly as if they were on a public network (certain limitations are applied). Services running on local hosts can be mapped to the public IP address.

Packet Basic data unit transmitted via computer networks. Packets consist of a header which include essential data (i.e. source and destination IP address, protocol type, etc.) and of the data body,. Data transmitted via networks is divided into small segments, or packets. If an error is detected in any packet or a packet is lost, it is not necessary to repeat the entire transmission process, only the particular packet will be re-sent.

Port 16-bit number (`1--65535`) used by TCP and UDP protocols to identify applications (services) at the host. More than one application can be run at a host simultaneously (i.e. WWW server, mail client, FTP client, etc.) Each application is identified by a port number. Ports `from 1 to 1023` are determined and they are used by standard (e.g. system) services (i.e. `80` = WWW). Ports greater than `1024` are free for use by any application (usually by clients as source ports or by nonstandard server applications).

PPTP Microsoft's proprietary protocol used for design of virtual private networks (see chapters concerning *VPN*).

Private IP addresses Local networks which do not belong to the Internet (private networks) use reserved ranges of IP addresses (private addresses). These addresses cannot be used in the Internet. This implies that IP ranges for local networks cannot collide with IP addresses used in the Internet.

The following IP ranges are reserved for private networks:

- `10.0.0.0/255.0.0.0`
- `172.16.0.0/255.240.0.0`
- `192.168.0.0/255.255.0.0`

Proxy server Common Internet connection type. Proxy servers connect clients and destination servers.

A proxy server works as an application and it is adapted for several application protocols (i.e. HTTP, FTP, Gopher, etc.). It is primarily used to facilitate Internet communication for private networks and to monitor and control Web traffic.

SPAM Unwanted, usually advertisement email.

Network adapter The equipment that connects hosts to a traffic medium. It can be represented by an Ethernet adapter, TokenRing adapter, by a modem, etc. Network adapters are used by hosts to send and receive packets. They are also referred to throughout this document as a network interface.

Routing table The information used by routers when making packet forwarding decisions. Packets are routed according to the packet's destination IP address. The routing table can be viewed in Windows operating systems using the `route print` command.

SSL SSL is a protocol used to secure and encrypt network communication. SSL was originally designed by Netscape in order to ensure secure transfer of Web pages over

HTTP protocol. Nowadays, it is used by most standard Internet protocols (SMTP, POP3, IMAP, LDAP, etc.).

Communication between the client and server operates as follows: the client generates a symmetric key and encrypts it with the public server key (obtained from the server certificate). The server decrypts it with its private key (kept solely by the server). Thus the symmetric key is known only to the server and client.

TCP *Transmission Control Protocol* is a transmission protocol which ensures reliable and sequentional data delivery. It establishes so called virtual connections and provides tools for error correction and data stream control. It is used by most of applications protocols which require reliable transmission of all data, such as *HTTP*, *FTP*, *SMTP*, *IMAP*, etc.

TCP protocol uses the folowing special control information — so called *flags*:

- *SYN* (Synchronize) — connection initiation (first packet in each connection)
- *ACK* (Acknowledgement) — acknowledgement of received data
- *RST* (Reset) — request on termination of a current connection and on initiation of a new one
- *URG* (Urgent) — urgent packet
- *PSH* (Push) — request on immediate transmission of the data to upper TCP/IP layers
- *FIN* (Finalize) — connection finalization

TCP/IP Name used for all traffic protocols used in the Internet (i.e. for IP, ICMP, TCP, UDP, etc.). *TCP/IP* does not stand for any particular protocol!

TLS Transport Layer Security. New version of SSL protocol. TLS is standardized by IETF and accepted by all significant software providers (i.e. Microsoft Corporation).

UDP *User Datagram Protokol* is a transmission protocol which transfers data through individual messages (so called datagrams). It does not establish new connections nor it provides reliable and sequentional data delivery, nor it enables error correction or data stream control. It is used for transfer of small-sized data (i.e. DNS queries) or for transmissions where speed is preferred from reliability (i.e. realtime audio and video files transmission).

VPN *Virtual Private Network, VPN* represents secure interconnection of private networks (i.e. of individual offices of an organization) via the Internet. Traffic between both networks (so called tunnel) is encrypted. This protects networks from

tapping. VPN incorporates special tunneling protocols, such as *Microsoft's IPSec* and *PPTP* (*Point-to-Point Tunnelling Protocol*).

WinRoute contains a proprietary VPN implemetation called *Kerio VPN*.

Chapter 20

Index