

Kerio WinRoute Firewall 6

Administrator's Guide

© Kerio Technologies s.r.o. All rights reserved.

This guide provides detailed description on configuration and administration of *Kerio WinRoute Firewall*, version 6.7.1. All additional modifications and updates reserved. User interfaces *Kerio StaR* and *Kerio Clientless SSL-VPN* are focused in a standalone document, *Kerio WinRoute Firewall — User's Guide*. The *Kerio VPN Client* application is described in a stand-alone document *Kerio VPN Client — User's Guide*.

For current version of the product, go to <http://www.kerio.com/firewall/download>. For other documents addressing the product, see <http://www.kerio.com/firewall/manual>.

Information regarding registered trademarks and trademarks are provided in appendix [A](#).

Products *Kerio WinRoute Firewall* and *Kerio VPN Client* include open source software. To view the list of open source items included, refer to attachment [B](#).

Contents

- 1 Quick Checklist 7**

- 2 Introduction 9**
 - 2.1 What’s new in 6.7.1 9
 - 2.2 Conflicting software 10
 - 2.3 System requirements 11
 - 2.4 Installation - Windows 12
 - 2.5 Initial configuration wizard (Windows) 17
 - 2.6 Upgrade and Uninstallation - Windows 19
 - 2.7 Installation - Software Appliance and VMware Virtual Appliance 20
 - 2.8 Upgrade - Software Appliance / VMware Virtual Appliance 23
 - 2.9 WinRoute Components 23
 - 2.10 WinRoute Engine Monitor (Windows) 24
 - 2.11 The firewall’s console (Software Appliance / VMware Virtual Appliance) 25

- 3 WinRoute Administration 27**
 - 3.1 Administration Console - the main window 28
 - 3.2 Administration Console - view preferences 31

- 4 Product Registration and Licensing 32**
 - 4.1 License types and number of users 32
 - 4.2 License information 33
 - 4.3 Registration of the product in the Administration Console 35
 - 4.4 Product registration at the website 43
 - 4.5 Subscription / Update Expiration 43
 - 4.6 User counter 45

- 5 Network interfaces 47**

- 6 Internet Connection 53**
 - 6.1 Persistent connection with a single link 54
 - 6.2 Connection with a single leased link - dial on demand 57
 - 6.3 Connection Failover 62
 - 6.4 Network Load Balancing 66

- 7 Traffic Policy 71**
 - 7.1 Network Rules Wizard 71
 - 7.2 How traffic rules work 78
 - 7.3 Definition of Custom Traffic Rules 78
 - 7.4 Basic Traffic Rule Types 90

7.5	Policy routing	95
7.6	User accounts and groups in traffic rules	98
7.7	Partial Retirement of Protocol Inspector	99
7.8	Use of Full cone NAT	101
7.9	Media hairpinning	102
8	Configuration of network services	104
8.1	DNS module	104
8.2	DHCP server	110
8.3	Dynamic DNS for public IP address of the firewall	119
8.4	Proxy server	121
8.5	HTTP cache	124
9	Bandwidth Limiter	130
9.1	How the bandwidth limiter works and how to use it	130
9.2	Bandwidth Limiter configuration	130
9.3	Detection of connections with large data volume transferred	135
10	User Authentication	137
10.1	Firewall User Authentication	137
11	Web Interface	141
11.1	Web interface preferences	141
11.2	User authentication at the web interface	146
12	HTTP and FTP filtering	147
12.1	Conditions for HTTP and FTP filtering	147
12.2	URL Rules	148
12.3	Content Rating System (Kerio Web Filter)	154
12.4	Web content filtering by word occurrence	158
12.5	FTP Policy	162
13	Antivirus control	167
13.1	Conditions and limitations of antivirus scan	167
13.2	How to choose and setup antiviruses	168
13.3	HTTP and FTP scanning	172
13.4	Email scanning	176
13.5	Scanning of files transferred via Clientless SSL-VPN (Windows)	178
14	Definitions	180
14.1	IP Address Groups	180
14.2	Time Ranges	181
14.3	Services	183
14.4	URL Groups	187

15	User Accounts and Groups	190
15.1	Viewing and definitions of user accounts	191
15.2	Local user accounts	193
15.3	Local user database: external authentication and import of accounts	203
15.4	User accounts in Active Directory — domain mapping	204
15.5	User groups	210
16	Administrative settings	214
16.1	System configuration (Software Appliance / VMware Virtual Appliance)	214
16.2	Setting Remote Administration	215
16.3	Update Checking	216
17	Advanced security features	218
17.1	P2P Eliminator	218
17.2	Special Security Settings	221
18	Other settings	224
18.1	Routing table	224
18.2	Universal Plug-and-Play (UPnP)	227
18.3	Relay SMTP server	229
19	Status Information	231
19.1	Active hosts and connected users	231
19.2	Network connections overview	238
19.3	List of connected VPN clients	242
19.4	Alerts	243
20	Basic statistics	248
20.1	Volume of transferred data and quota usage	248
20.2	Interface statistics	250
21	Kerio StaR - statistics and reporting	254
21.1	Monitoring and storage of statistic data	254
21.2	Settings for statistics and quota	256
21.3	Connection to StaR and viewing statistics	259
22	Logs	262
22.1	Log settings	262
22.2	Logs Context Menu	265
22.3	Alert Log	269
22.4	Config Log	269
22.5	Connection Log	270
22.6	Debug Log	271
22.7	Dial Log	273
22.8	Error Log	275

22.9	Filter Log	276
22.10	Http log	277
22.11	Security Log	278
22.12	Sslvpn Log	280
22.13	Warning Log	280
22.14	Web Log	281
23	Kerio VPN	283
23.1	VPN Server Configuration	284
23.2	Configuration of VPN clients	289
23.3	Interconnection of two private networks via the Internet (VPN tunnel) ...	291
23.4	Exchange of routing information	296
23.5	Example of Kerio VPN configuration: company with a filial office	297
23.6	Example of a more complex Kerio VPN configuration	310
24	Kerio Clientless SSL-VPN (Windows)	335
24.1	Configuration of WinRoute's SSL-VPN	335
24.2	Usage of the SSL-VPN interface	337
25	Specific settings and troubleshooting	338
25.1	Configuration Backup and Transfer	338
25.2	Configuration files	339
25.3	Automatic user authentication using NTLM	340
25.4	FTP on WinRoute's proxy server	343
25.5	Internet links dialed on demand	346
26	Technical support	351
26.1	Essential Information	351
26.2	Tested in Beta version	352
A	Legal Notices	353
B	Used open source items	354
	Glossary of terms	357
	Index	364

Chapter 1

Quick Checklist

In this chapter you can find a brief guide for a quick setup of *Kerio WinRoute Firewall* (referred to as “*WinRoute*” within this document). 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 (e.g. *Ethernet* or *WiFi* network adapter), another must be connected to the Internet (e.g. *Ethernet* or *WiFi* network adapter, USB ADSL modem, analog modem or an ISDN adapter).

On *Windows*, 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 and in the wizard provide required basic parameters (for details, see chapter [2.4](#) or [2.7](#)).
3. Set interface groups and basic traffic rules using the *Network Rules Wizard* (see chapter [7.1](#)).
4. Run the *DHCP server* and set required IP ranges including their parameters (subnet mask, default gateway, DNS server address/domain name). For details, see chapter [8.2](#).
5. Check *DNS* module settings. Define the local DNS domain if you intend to scan the hosts file and/or the DHCP server table. For details, see chapter [8.1](#).
6. Set user mapping from the *Active Directory* domain or create/import local user accounts and groups. Set user access rights. For details see chapter [15](#).
7. Define IP groups (chapter [14.1](#)), time ranges (chapter [14.2](#)) and URL groups (chapter [14.4](#)), that will be used during rules definition (refer to chapter [14.2](#)).
8. Create URL rules (chapter [12.2](#)) and set the *Kerio Web Filter* module (chapter [12.3](#)). Set HTTP cache and automatic configuration of browsers (chapter [8.5](#)). Define FTP rules (chapter [12.5](#)).

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.

External antivirus must be installed before it is set in *WinRoute*, otherwise it is not available in the combo box.

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:

- *Automatic configuration* — activate the *Automatically detect settings* option (*Internet Explorer*) or specify URL for automatic configuration (other types of browsers). For details, refer to chapter [8.5](#).
- *Manual configuration* — select type of connection via the local network or define IP address and appropriate proxy server port (see chapter [8.4](#)).

Introduction

2.1 What's new in 6.7.1

In version 6.7.1, *WinRoute* brings the following new features:

Kerio WinRoute Firewall Software Appliance / VMware Virtual Appliance

Kerio WinRoute Firewall is now available as a so called software appliance (*Software Appliance / VMware Virtual Appliance*). This appliance is distributed as a full installation package with the firewall and operating system and can be installed on a physical or virtual computer without an operating system. *Software Appliance* cannot be installed on a computer with another operating system. Installation package of the standalone product for installation on *Linux* is not available.

Kerio WinRoute Firewall in the *Software Appliance / VMware Virtual Appliance* edition provides the same features as the *Windows* version, only with the *Kerio Clientless SSL-VPN* interface missing.

Web administration interface (Web Administration)

The new *Web Administration* interface allows both remote and local administration of the firewall without the need to install the *Kerio Administration Console*. This interface allows configuration of crucial *WinRoute* parameters — the interface, traffic policy, *HTTP* and *FTP* filtering rules, user accounts and groups, etc. However, the *Kerio Administration Console* is still available and allow setting of all configuration options.

The *Web Administration* interface is available at `https://server:4081/admin` (server stands for the firewall name or IP address and 4081 for the default port of its web interface).

Refer to chapter [3](#) for more information.

Exporting and Importing Configuration

WinRoute now includes also a special backup-and-recovery tool which allows to back up and recover full configuration including local user accounts and SSL certificates. These functions allow easy and quick recovery of the firewall for cases of hardware failure, transfer to another computer and cloning of an identical configuration for multiple firewalls. To export or import configuration, go to the *Web Administration* interface.

More details can be found in chapter [25.1](#).

Kerio Web Filter, the new web page rating module

Kerio Web Filter is a special module, used for rating of web pages in accordance to their content categories. In *WinRoute*, it replaces the *ISS OrangeWeb Filter* module. The way of how filtering rules are created is the same as before.

More details can be found in chapter [12.3](#).

Support for Windows 7

Kerio WinRoute Firewall now includes full support for the new operating system *Microsoft Windows 7*.

2.2 Conflicting software

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.

The computer where *WinRoute* is installed (the host) can be also used as a workstation. However, it is not recommended — user interaction may affect performance of the operating system which affects *WinRoute* performance badly.

Collision of low-level drivers

WinRoute collides with system services and applications the low-level drivers of whose use a similar or an identical technology. The security log contains the following types of services and applications:

- The *Internet Connection Firewall / Internet Connection Sharing* system service. *WinRoute* can detect and automatically disable this service.
- The system service *Routing and Remote Access Service (RRAS)* in *Windows Server* operating systems. This service allows also sharing of Internet connection ([NAT](#)). *WinRoute* can detect if NAT is active in the *RRAS* service; if it is, a warning is displayed. In reaction to the alert message, the server administrator should disable NAT in the *RRAS* configuration.
If NAT is not active, collisions should be avoided and *WinRoute* can be used hand in hand with the *RRAS* service.
- Network firewalls — e.g. *Microsoft ISA Server*.
- Personal firewalls, such as *Sunbelt Personal Firewall*, *Zone Alarm*, *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 applications of this type 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 23](#)). Otherwise, we recommend you to test a particular VPN server or VPN client with *WinRoute* trial version or to contact our technical support (see [chapter 26](#)).

Note: VPN implementation included in *Windows* operating system (based on the 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* module,
- 67/UDP — *DHCP* server,
- 1900/UDP — the *SSDP Discovery* service,
- 2869/TCP — the *UPnP Host* service.

The *SSDP Discovery* and *UPnP Host* services are included in the UPnP support (refer to chapter [18.2](#)).

- 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.

- 443/TCP — server of the *SSL-VPN* interface (only in *WinRoute* on *Windows* — see chapter [24](#)),
- 3128/TCP — HTTP proxy server (see chapter [8.4](#)),
- 4080/TCP — web interface of the firewall (refer to chapter [11](#)),
- 4081/TCP — secured (SSL-encrypted) version of the firewall's web interface (see chapter [11](#)),
- 4090/TCP+UDP — proprietary VPN server (for details refer to chapter [23](#)).

Antivirus applications

Most of the modern desktop antivirus programs (antivirus applications designed to protect desktop workstations) scans also network traffic — typically *HTTP*, *FTP* and email protocols. *WinRoute* also provides with this feature which may cause collisions. Therefore it is recommended to install a server version of your antivirus program on the *WinRoute* host. The server version of the antivirus can also be used to scan *WinRoute's* network traffic or as an additional check to the integrated antivirus *McAfee* (for details, see chapter [13](#)).

If the antivirus program includes so called realtime file protection (automatic scan of all read and written files), it is necessary to exclude directories cache (HTTP cache in *WinRoute* see chapter [8.5](#)) and `tmp` (used for antivirus check). If *WinRoute* uses an antivirus to check objects downloaded via HTTP or FTP protocols (see chapter [13.3](#)), the cache directory can be excluded with no risk — files in this directory have already been checked by the antivirus.

The *McAfee* integrated antivirus plug-in does not interact with antivirus application installed on the *WinRoute* host (provided that all the conditions described above are met).

2.3 System requirements

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

- CPU 1 GHz,
- 1 GB RAM,
- Two network interfaces (including dial-ups).

For *Windows*:

- 50 MB free disk space for installation of *Kerio WinRoute Firewall*.
- Disk space for statistics (see chapter [21](#)) and logs (in accordance with traffic flow and logging level — see chapter [22](#)).
- to keep the installed product (especially its configuration files) as secure as possible, it is recommended to use the *NTFS* file system.

For *Kerio WinRoute Firewall Software Appliance*:

- Minimum 3 GB hard disk.
- No operating system is required to be installed on the computer. Any existing operating system will be removed from the computer.

For *Kerio WinRoute Firewall VMware Virtual Appliance*:

- *VMware Player*, *VMware Workstation* or *VMware Server*.
- 3 GB free disk space.

The following browsers can be used to access the *WinRoute (Kerio StaR* — see chapter [21](#) and *Kerio SSL-VPN* — see chapter [24](#)) web services:

- *Internet Explorer 7 or higher*,
- *Firefox 2 or higher*,
- *Safari 3 or higher*.

2.4 Installation - Windows

Installation packages

Kerio WinRoute Firewall is distributed in two editions: one is for 32-bit systems and the other for 64-bit systems (see the product's download page: <http://www.kerio.com/firewall/download>).

The 32-bit edition (the “*win32*” installation package) supports the following operating systems:

- *Windows 2000*,
- *Windows XP* (32 bit),
- *Windows Server 2003* (32 bit),
- *Windows Vista* (32 bit),
- *Windows Server 2008* (32 bit).

The 64-bit edition (the “*win64*” installation package) supports the following operating systems:

- *Windows XP* (64 bit),
- *Windows Server 2003* (64 bit),
- *Windows Vista* (64 bit),
- *Windows Server 2008* (64 bit).

Older versions of *Windows* operating systems are not supported.

Note:

1. *WinRoute* installation packages include the *Kerio Administration Console*. The separate *Kerio Administration Console* installation package (file `kerio-kwf-admin*.exe`) is designed for full remote administration from another host. This package is identical both for 32-bit and 64-bit *Windows* systems. For details on *WinRoute* administration, see chapter [3](#).
2. For correct functionality of the *Kerio StaR* interface (see chapter [21](#)), it is necessary that the *WinRoute* host's operating system supports all languages that would be used in the *Kerio StaR* interface. Some languages (Chinese, Japanese, etc.) may require installation of supportive files. For details, refer to documents regarding the corresponding operating system.

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, WiFi, etc.) and at least one interface connected to the Internet. You can use either a network adapter (Ethernet, WiFi, 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 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. by `kerio-kwf-6.6.0-5700-win32.exe`), it is possible to select a language for the installation wizard. Language selection affects only the installation, language of the user interface can then be set separately for individual *WinRoute* components.

In the installation wizard, you can choose either *Full* or *Custom* installation. Custom mode will let you select optional components of the program:

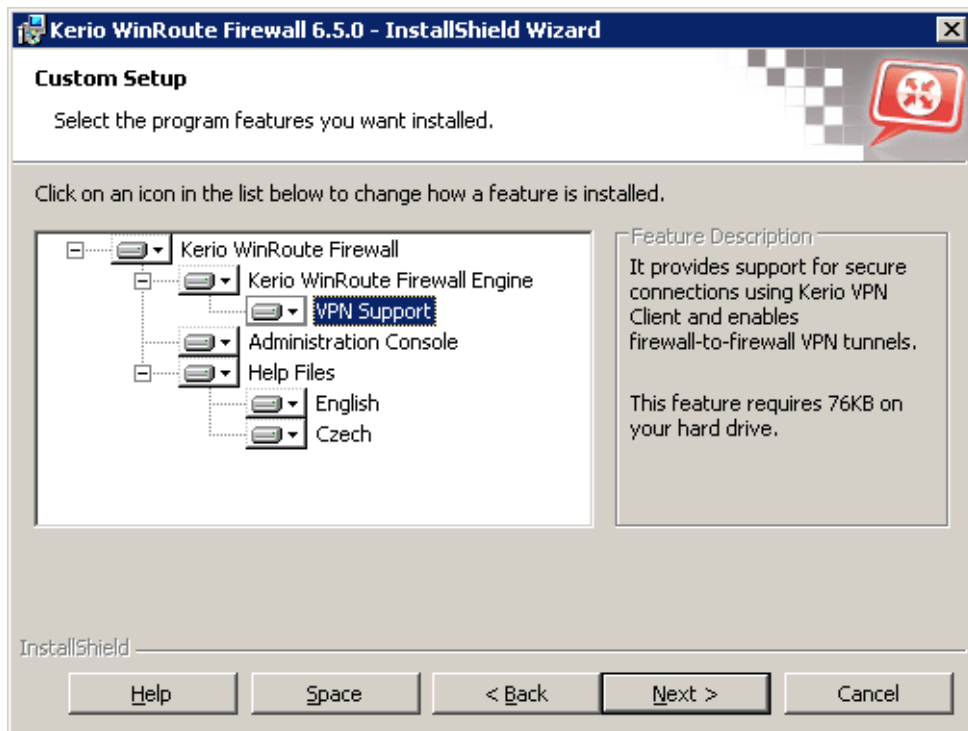


Figure 2.1 Installation — customization by selecting optional components

- *Kerio WinRoute Firewall Engine* — core of the application.
- *VPN Support* — proprietary VPN solution developed by *Kerio Technologies (Kerio VPN)*.
- *Administration Console* — the *Kerio Administration Console* application (universal console for all server applications of *Kerio Technologies*) including *WinRoute* administration tools.
- *Help files* — this manual in the *HTML Help* format. For help files details, see *Kerio Administration Console — Help* (available at <http://www.kerio.com/firewall/manual>).

Go to chapter 2.9 for a detailed description of all *WinRoute* components. For detailed description on the proprietary VPN solution, refer to chapter 23.

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 2.5).

Under usual circumstances, a reboot of the computer is not required after the installation (a restart may be required if the installation program rewrites shared files which are currently in use). This will install the *WinRoute* low-level driver into the system kernel. *WinRoute Engine* will be automatically launched when the installation is complete. The engine runs as a service.

Note:

1. 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.

2. The installation program does not allow to install the *Administration Console* separately. Installation of the *Administration Console* for the full remote administration requires a separate installation package (file `kerio-kwf-admin*.exe`).

Protection of the installed product

To provide the firewall with the highest security possible, it is necessary to ensure that undesirable (unauthorized) persons has no access to the critical files of the application, especially to configuration files. If the *NTFS* system is used, *WinRoute* refreshes settings related to access rights to the directory (including all subdirectories) where the firewall is installed upon each startup. Only members of the *Administrators* group and local system account (*SYSTEM*) are assigned the full access (read/write rights), other users are not allowed access the directory.

Warning

If the *FAT32* file system is used, it is not possible to protect *WinRoute* in the way suggested above. For this reason, it is recommended to install *WinRoute* only on computers which use the *NTFS* file system.

Conflicting Applications and System Services

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

1. *Windows Firewall's* system components¹ and *Internet Connection Sharing*.

These components provide the same low-level functions as *WinRoute*. If they are running concurrently with *WinRoute*, the network communication would not be functioning correctly and *WinRoute* might be unstable. Both components are run by the *Windows Firewall / Internet Connection Sharing* system service.²

Warning

To provide proper functionality of *WinRoute*, it is *necessary* that the *Internet Connection Firewall / Internet Connection Sharing detection* is stopped and forbidden!

¹ In *Windows XP Service Pack 1* and older versions, the integrated firewall is called *Internet Connection Firewall*.

² In the older *Windows* versions listed above, the service is called *Internet Connection Firewall / Internet Connection Sharing*.

2. Universal Plug and Play Device Host and SSDP Discovery Service

The services support *UPnP* (Universal Plug and Play) in the *Windows XP*, *Windows Server 2003*, *Windows Vista* and *Windows Server 2008* operating systems. However, these services collide with the *UPnP* support in *WinRoute* (refer to chapter [18.2](#)).

The *WinRoute* installation includes a dialog where it is possible to disable colliding system services.



Figure 2.2 Disabling colliding system services during installation

By default, the *WinRoute* installation disables all the colliding services listed. Under usual circumstances, it is not necessary to change these settings. Generally, the following rules are applied:

- The *Windows Firewall / Internet Connection Sharing (ICS)* service should be disabled. Otherwise, *WinRoute* will not work correctly. The option is a certain kind of warning which informs users that the service is running and that it should be disabled.
- To enable support for the *UPnP* protocol in *WinRoute* (see chapter [18.2](#)), it is necessary to disable also services *Universal Plug and Play Device Host* and *SSDP Discovery Service*.
- If you do not plan to use support for *UPnP* in *WinRoute*, it is not necessary to disable the *Universal Plug and Play Device Host* and *SSDP Discovery Services*.

Note:

1. Upon each startup, *WinRoute* detects automatically whether the *Windows Firewall / Internet Connection Sharing* is running. If it is, *WinRoute* stops it and makes a record in the

warning log. This helps assure that the service will be enabled/started immediately after the *WinRoute* installation.

2. On *Windows XP Service Pack 2*, *Windows Server 2003*, *Windows Vista* and *Windows Server 2008*, *WinRoute* registers in the *Security Center* automatically. This implies that the *Security Center* always indicates firewall status correctly and it does not display warnings informing that the system is not protected.

2.5 Initial configuration wizard (Windows)

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

Setting of administration username and 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.

Kerio WinRoute Firewall 6.5.0 - InstallShield Wizard

Administrative Account

Configure an account with full access to the administration.

Please provide username and password for an account which will have full access to the administration.

Once the installation is complete this user can use the administration console to login and configure the product.

Username: Admin

Password: ●●●●●●●●●●

Confirm Password: ●●●●●●●●●●

Do not leave the password blank and make it at least six characters long.

InstallShield

< Back Next > Cancel

Figure 2.3 Initial configuration — Setting of administration username and password

Password and its confirmation must be entered in the dialog for account settings. Name Admin can be changed in the *Username* edit box.

Note: If the installation is running as an upgrade, this step is skipped since the administrator account already exists.

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 7). 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 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. Allowing full access from a point might endanger security.

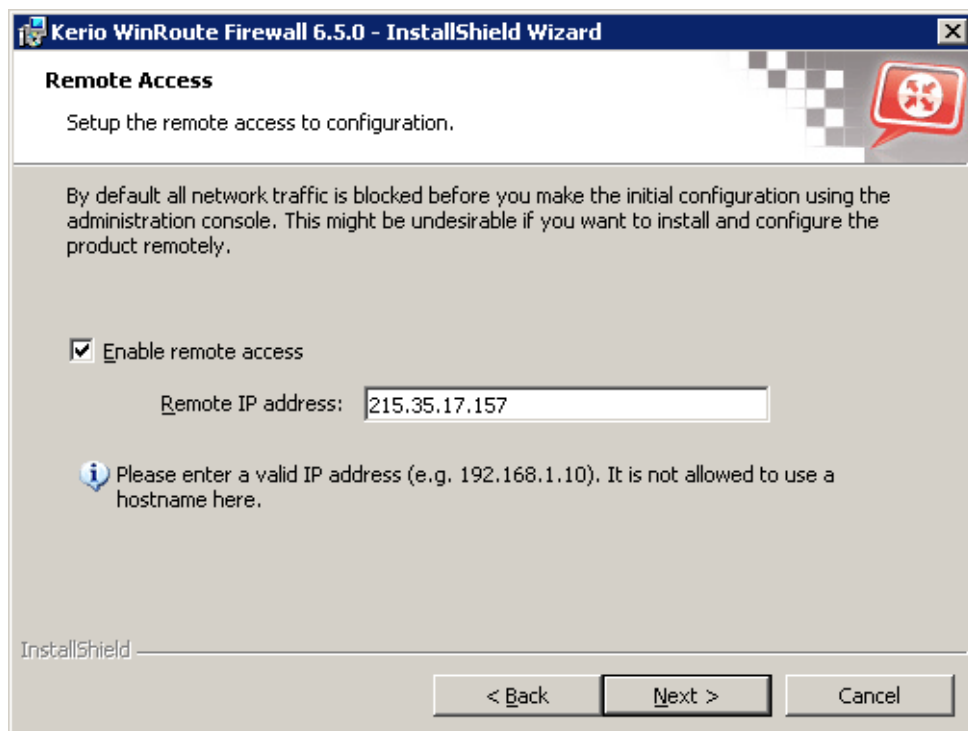


Figure 2.4 Initial configuration — Allowing remote administration

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.

Warning

The remote access rule is disabled automatically when *WinRoute* is configured using the network policy wizard (see chapter [7.1](#)).

2.6 Upgrade and Uninstallation - Windows

Upgrade

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. License, all logs and user defined settings are kept safely.

Note: This procedure applies to upgrades between versions of the same series (e.g. from 6.6.0 to 6.6.1) or from a version of the previous series to a version of the subsequent series (e.g. from 6.5.2 to 6.6.0). For case of upgrades from an older series version (e.g. 6.3.1), full compatibility of the configuration cannot be guaranteed and it is recommended to upgrade “step by step” (e.g. 6.3.1 → 6.4.0 → 6.5.0 → 6.6.0) or to uninstall the old version along with all files and then install the new version “from scratch”.

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 details, refer to chapter [16.3](#).

Uninstallation

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* directory can be optionally deleted.

(the typical path is C:\Program Files\Kerio\WinRoute Firewall)

— configuration files, SSL certificates, license key, logs, etc.

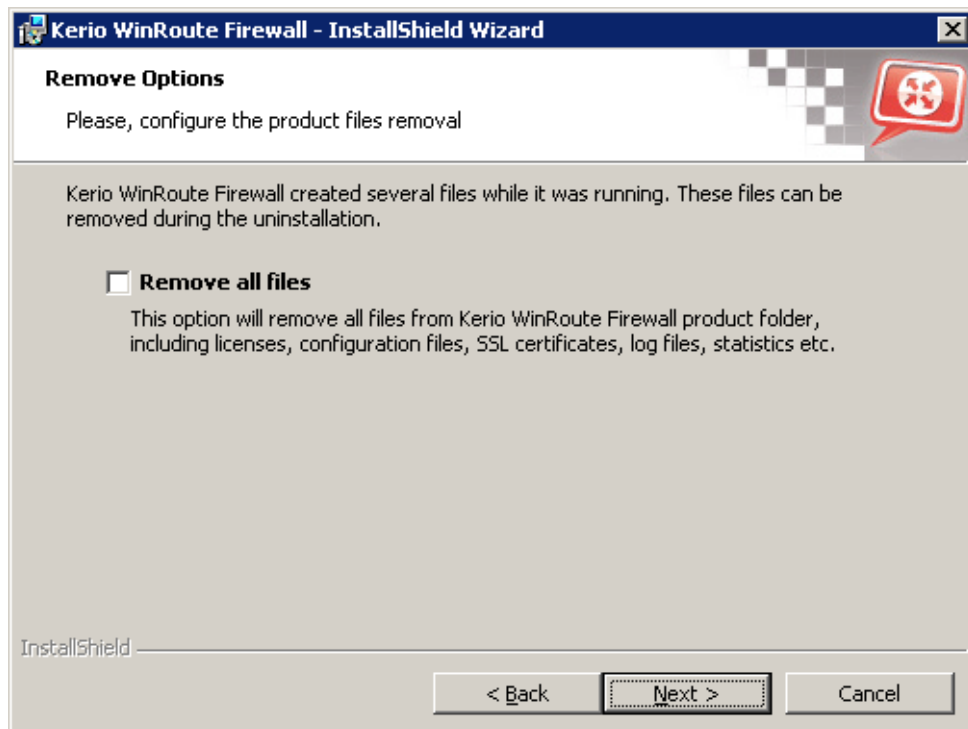


Figure 2.5 Uninstallation — asking user whether files created in WinRoute should be deleted

Keeping these files may be helpful for copying of the configuration to another host or if it is not sure whether the SSL certificates were issued by a trustworthy certification authority.

During uninstallation, the *WinRoute* installation program automatically refreshes the original status of the *Windows Firewall / Internet Connection Sharing, Universal Plug and Play Device Host* and *SSDP Discovery Service* system services.

2.7 Installation - Software Appliance and VMware Virtual Appliance

WinRoute in the software appliance edition is distributed:

- as an ISO of the installation CD which is used to install the system and then install the firewall either on a physical or virtual computer (*Software Appliance*),
- as a virtual appliance for *VMware* (*VMware Virtual Appliance*).

Standalone *WinRoute* installation package for installation on previously installed *Linux* is not available.

Software Appliance / VMware Virtual Appliance installation process consists of the following simple steps:

Start of the installation

Software Appliance

ISO image of the installation CD can be burned on a physical CD and then the CD can be used for installation of the system on the target computer (either physical or virtual). In case of virtual computers, the ISO image can be also connected as a virtual CD ROM, without the need to burn the installation ISO file on a CD.

Note: Kerio WinRoute Firewall Software Appliance cannot be installed on a computer with another operating system. Existing operating system on the target disk will be removed within the installation.

VMware Virtual Appliance

Unzip the distribution package (*Zip*) and open the *.vmx* file in your *VMware* application (*VMware Player*, *VMware Workstation*, *VMware Server*). This runs the *Kerio WinRoute Firewall* installer.

The following steps are identical both for *Software Appliance* and *Virtual Appliance*.

Language selection

The selected language will be used both for *WinRoute* installation and for the firewall's console (see chapter [2.11](#)).

Selection of target hard disk

If the installation program detects more hard disks in the computer, then it is necessary to select a disk for *WinRoute* installation. Content of the selected disk will be completely removed before *WinRoute* installation, while other disk are not affected by the installation.

If there is an only hard disk detected on the computer, the installer continues with the following step automatically. If no hard disk is found, the installation is closed. Such error is often caused by an unsupported hard disk type or hardware defect.

Selection of network interface for the local network and access to administration

The installer lists all detected network interfaces of the firewall. Select an interface which is connected to the local (trustworthy) network which the firewall will be remotely administered from.

In the field, a computer may have multiple interfaces of the same type and it is therefore not easy to recognize which interface is connected to the local network and which to the Internet. To a certain extent, hardware addresses of the adapters can be a clue or you can experiment — select an interface, complete the installation and try to connect to the administration. If the connection fails, use option *Network Configuration* in the main menu of the firewall's console to change the settings (see chapter [2.11](#)).

There can also arise another issue — that the program does not detect some or any network adapters. In such case, it is recommended to use another type of the physical or virtual (if the

virtual computer allows this) adapter or install *WinRoute Software Appliance* on another type of virtual machine. If such issue arises, it is highly recommended to consult the problem with the *Kerio Technologies* technical support (see chapter [26](#)).

provided that no network adapter can be detected, it is not possible to continue installing *WinRoute*.

Setting of the local interface's IP address

It is now necessary to define IP address and subnet mask for the selected local network interface. These parameters can be defined automatically by using information from a DHCP server or manually.

For the following reasons, it is recommended to set local interface parameters manually:

- Automatically assigned IP address can change which may cause problems with connection to the firewall administration (although the IP address can be reserved on the DHCP server, this may bring other problems).
- In most cases *WinRoute* will be probably used itself as a DHCP server for local hosts (workstations).

Admin password

The installation requires specification of the password for the account *Admin* (the account of the main administrator of the firewall). Username *Admin* with this password are then used for access:

- to the firewall's console (see chapter [2.11](#)),
- to the remote administration of the firewall via the web administration interface (see chapter [3](#)),
- to the remote administration of the firewall via the *Kerio Administration Console* (see chapter [3](#)).

Remember this password or save it in a secured location and keep it from anyone else!

Time zone, date and time settings

Many *WinRoute* features (user authentication, logs, statistics, etc.) require correct setting of date, time and time zone on the firewall. Select your time zone and in the next page check (and change, if necessary) date and time settings.

Completing the installation

Once all these parameters are set, the *WinRoute Firewall Engine* service (daemon) is started. While the firewall is running, the firewall's console will display information about remote administration options and change of some basic configuration parameters — see chapter [2.11](#).

2.8 Upgrade - Software Appliance / VMware Virtual Appliance

WinRoute can be upgraded by the following two methods:

- by starting the system from the installation CD (or a mounted ISO) of the new version. The installation process is identical with the process of a new installation with an the only exception that at the start the installer asks you whether to execute an upgrade (any existing data will be kept) or a new installation (all configuration files, statistics, logs, etc will be removed). For details, see chapter [2.7](#).
- by the *Kerio Administration Console* update checker. For details, refer to chapter [16.3](#)

2.9 WinRoute Components

WinRoute consists of the following modules:

WinRoute Firewall Engine

WinRoute Firewall Engine is the core of the program that provides 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 (Windows only)

Allows viewing and modification of the *Engine's* status (stopped / running) and setting of start-up preferences (i.e. whether *Engine* and *Monitor* should be run automatically at system start-up). It also provides easy access to the *Administration Console*. For details, refer to chapter [2.10](#).

Note: *WinRoute Firewall Engine* is independent on the *WinRoute Engine Monitor*. The *Engine* can be running even if there is no icon in the system tray.

Kerio Administration Console (Windows only)

It is a versatile console for full local or remote administration of *Kerio Technologies* server products. For successful connection to an application you need a plug-in with an appropriate interface.

Kerio Administration Console is installed on *Windows* hand-in-hand with the appropriate module during the installation of *Kerio WinRoute*. The separate installation package *Kerio Administration Console* for *WinRoute* is available for remote administration from another host. The *Kerio Administration Console* is available for *Windows* only, but it can be used for administration of both *WinRoute* installed on *Windows* and *Kerio WinRoute Firewall Software Appliance / VMware Virtual Appliance*.

Detailed guidance for *Kerio Administration Console* is provided in *Kerio Administration Console — Help* (<http://www.kerio.com/firewall/manual>).

The firewall's console (Software Appliance / VMware Virtual Appliance only)

The firewall's console is a simple interface permanently running on the *WinRoute* host. It allows to set basic parameters of the operating system and the firewall for cases when it is not possible to administer it remotely via the *Web Administration* interface or the *Kerio Administration Console*.

2.10 WinRoute Engine Monitor (Windows)

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

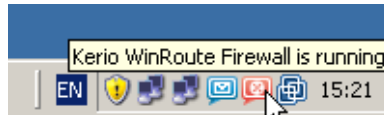


Figure 2.6 WinRoute Engine Monitor icon in the Notification Area

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. For this time the icon gets grey and is inactive.

On Windows, left double-clicking on this icon runs the *Kerio Administration Console* (described later). Use the right mouse button to open the following menu:

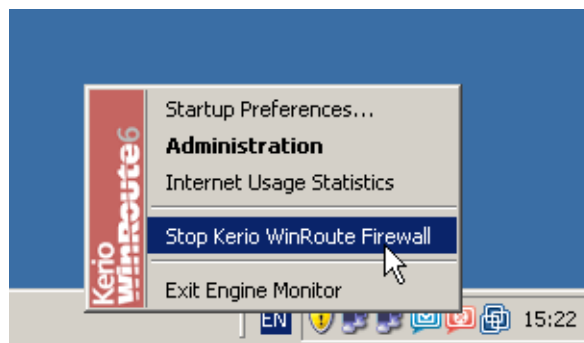


Figure 2.7 WinRoute Engine Monitor menu

Start-up 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. Both options are enabled by default.

Administration

Runs *Kerio Administration Console* (equal to double-clicking on the *WinRoute Engine Monitor* icon).

Internet Usage Statistics

Opens *Internet Usage Statistics* in the default browser. For details, see chapter [21](#).

Start / Stop WinRoute Firewall

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).

Note:

1. If a limited version of *WinRoute* is used (e.g. a trial version), a notification is displayed 7 days before its expiration. This information is displayed until the expiration.
2. *WinRoute Engine Monitor* is available in English only.

2.11 The firewall's console (Software Appliance / VMware Virtual Appliance)

On the console of the computer where *Kerio WinRoute Firewall Software Appliance / VMware Virtual Appliance* is running, information about the firewall remote administration options is displayed. Upon authenticating by the administration password (see above), this console allows to change some basic settings, restore default settings after installation and shut down or restart the computer.

By default, the console shows only information about URL or IP address which can be used for firewall administration via the firewall's web administration interface or the *Kerio Administration Console*. To access configuration options, authentication with the *Admin* password is required (*Admin* is the main firewall administrator's account). If idle for some time, the user gets logged out automatically and the welcome page of the console showing details on the firewall's remote administration is displayed again.

The firewall's console provides the following configuration options:

Network Interface Configurations

This option allows to show or/and edit parameters of individual network interfaces of the firewall. Each interface allows definition of automatic configuration via *DHCP* or manual configuration of IP address, subnet mask and default gateway.

Note: No default gateway should be set on interfaces connected to the local network, otherwise this firewall cannot be used as a gateway for the Internet access.

Remote administration policy settings

When you change the firewall's traffic policy (see chapter [7](#)) via the web administration interface or the *Kerio Administration Console*, you may happen to block access to the remote administration accidentally.

If you are sure that the firewall's network interfaces are configured correctly and despite of that it is not possible to access the remote administration, you can use the *Remote Administration* option to change the traffic policy so that the rules do not block remote administration on any interface.

Upon saving changes in traffic rules, the *Kerio WinRoute Firewall Engine* service will be restarted automatically.

In the field, unblocking of the remote administration means that a rule will be added to the top of the traffic policy table that would allow access *KWF Admin* (connection with the *Kerio Administration Console*), *KWF WebAdmin* (unsecured web interface) and *KWF WebAdmin-SSL* (secured web interface) services from any computer .

Shutting down / restarting the firewall

If you need to shut your computer down or reboot it, these options provide secure closure of the *Kerio WinRoute Firewall Engine* and shutdown of the firewall's operating system.

Restoring default configuration

This option restores the default firewall settings as installed from the installation CD or upon the first startup of the *VMware* virtual host. All configuration files and data (logs, statistics, etc.) will be removed and it will then be necessary to execute the initial configuration of the firewall again as if a new installation (see chapter [2.7](#)).

Restoring the default configuration can be helpful if the firewall's configuration is accidentally damaged that much that it cannot be corrected by any other means.

Chapter 3

WinRoute Administration

For *WinRoute* configuration, two tools are available:

The Web Administration interface

The *Web Administration* interface allows both remote and local administration of the firewall via a common web browser. In the current version of *WinRoute*, the *Web Administration* allows configuration of all crucial *WinRoute* parameters:

- network interfaces,
- traffic rules,
- *HTTP* and *FTP* filtering rules,
- user accounts, groups and domains,
- IP groups, URL groups, time ranges and network services.

The *Web Administration* interface is available at `https://server:4081/admin` (`server` stands for the firewall name or IP address and 4081 for the default port of its web interface). *HTTPS* traffic between the client and the *WinRoute Firewall Engine* is encrypted. This protects the communication from tapping and misuse. It is recommended to use the unsecured version of the *Web Administration* (the *HTTP* protocol) only for local administration of *WinRoute* (i.e. administration from the computer where it is installed).

Kerio Administration Console

Kerio Administration Console (referred to as the *Administration Console* in this document) is an application used for administration of all Kerio Technologies' server products. All *WinRoute* parameters can be configured here.

Using this program you can access the firewall either locally (from the *WinRoute* host) or remotely (from another host). Traffic between *Administration Console* and *WinRoute Firewall Engine* is encrypted. This protects you from tapping and misuse.

Kerio Administration Console is installed on *Windows* hand-in-hand with it during the installation of *Kerio WinRoute*.

The separate installation package *Kerio Administration Console* for *WinRoute* is available for remote administration from another host. The *Kerio Administration Console* is available for *Windows* only, but it can be used for administration of both *WinRoute* installed on *Windows* and *Kerio WinRoute Firewall Software Appliance / VMware Virtual Appliance*. Detailed guidelines for the *Administration Console* are provided under *Kerio Administration Console — Help* (to view these guidelines, use option *Help* → *Contents* in the main *Administration Console* window, or you can download it from <http://www.kerio.com/firewall/manual>).

The following chapters of this document address individual sections of the *Administration Console*, the module which allows full configuration. The *Web Administration* interface is almost identical as the *Administration Console* and its sections.

Note:

1. The *Web Administration* interface and the *Administration Console* for *WinRoute* are available in 16 localization versions. The *Web Administration* interface allows language selection by simple switching of the flag located in the top right corner of the window or by following the browser language preferences. The *Administration Console* allows language settings in the *Tools* menu of the login dialog box.
2. Upon the first login to the *Administration Console* after a successful *WinRoute* 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 [7.17.1](#).

3.1 Administration Console - the main window

The *WinRoute* administration dialog window (“administration window”) will be opened upon a successful login to the *WinRoute Firewall Engine* through the *Administration Console*. This window is divided into two parts:

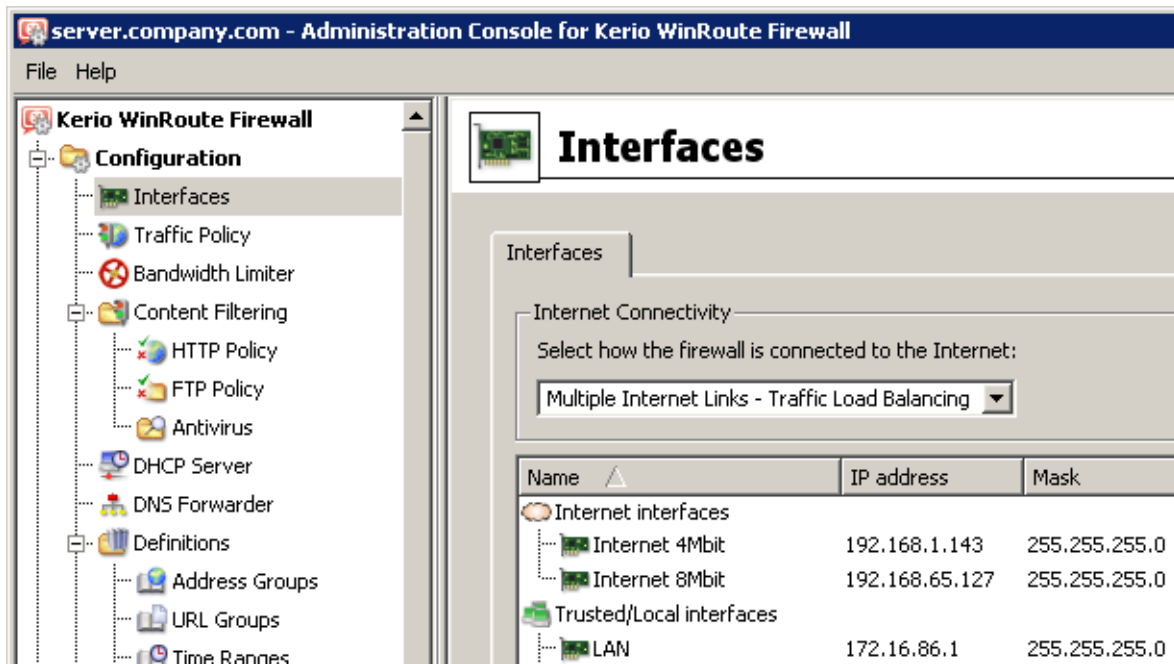


Figure 3.1 The main window of Administration Console for WinRoute

- The left column contains the tree view of sections. The individual sections of the tree can be expanded and collapsed for easier navigation. *Administration Console* remembers the current tree settings and uses them upon the next login.
- In the right part of the window, the contents of the section selected in the left column is displayed (or a list of sections in the selected group).

Administration Window — Main menu

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).
- *New connection* — opens the main window of the *Administration Console*. Use a bookmark or the login dialog to connect to a server.
This option can be useful when the console will be used for administration of multiple server applications (e.g. *WinRoute* at multiple servers). For details, refer to the *Help* section in the *Administration Console* manual.
Note: The *New Connection* option opens the same dialog as running the *Administration Console* from the *Start* menu.
- *Quit* — this option terminates the session (users are logged out of the server and the administration window is closed). The same effect can be obtained by clicking the little cross in the upper right corner of the window or pressing *Alt+F4* or *Ctrl+Q*.

The Edit menu (on the welcome page only)

Options under *Edit* are related to product registration and licensing. The options available in the menu depend on the registration status (for example, if the product is registered as a trial version, it is possible to use options of registration of a purchased license or a change of registration data).

- *Copy license number to clipboard* — copies the license number (the *ID licence* item) to the clipboard. This may be helpful e.g. when ordering an upgrade or subscription, where the number of the base license is required, or when sending an issue to the *Kerio Technologies* technical support.
- *Register trial version* — registration of the product's trial version.
- *Register product* — registration of a product with a purchased license number.
- *Install license* — use this option to import your license key file (for details, see chapter [4.4](#)).

Help menu

- *Show Server's Identity* — this option provides information about the firewall which the *Administration Console* is currently connected to (name or IP address of the server, port and SSL-certificate fingerprint). This information can be used

for authentication of the firewall when connecting to the administration from another host (see *Kerio Administration Console — Help*).

- *Administrator's guide* — this option displays the administrator's guide in *HTML Help* format. For details about help files, see *Kerio Administration Console — Help* manual.
- *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

The status bar at the bottom of the administration window displays the following information (from left to right):



Figure 3.2 Administration Console status bar

- The section of the administration window currently selected in the left column. This information facilitates navigation in the administration window when any part of the section tree is not visible (e.g. when a lower screen resolution is selected).
- Name or IP address of the server and port of the server application (*WinRoute* uses port 44333).
- Name of the user logged in as administrator.
- Current state of the *Administration Console*: *Ready* (waiting for user's response), *Loading* (retrieving data from the server) or *Saving* (saving changes to the server).

Detection of WinRoute Firewall Engine connection drop-out

Administration Console is able to detect the connection failure automatically. The failure is usually detected upon an attempt to read/write the data from/to the server (i.e. when the *Apply* button is pressed or when a user switches to a different section of *Administration Console*). In such case, a connection failure dialog box appears where the connection can be restored.

After you remove the cause of the connection failure, the connection can be restored. *Administration Console* provides the following options:

- *Apply & Reconnect* — connection to the server will be recovered and all changes done in the current section of the *Administration Console* before the disconnection will be saved,
- *Reconnect* — connection to the server will be recovered without saving any changes performed in the particular section of the console before the disconnection.

If the reconnection attempt fails, only the error message is shown. You can then try to reconnect using the *File → Restore connection* option from the main menu, or close the window and restore the connection using the standard procedure.

Note: After a connection failure, the *Web Administration* interface is redirected and opened at the login page automatically. Any unsaved changes will get lost.

3.2 Administration Console - view preferences

Many sections of the *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. When you right-click each of the above sections, a pop-up menu with *Modify columns* option is displayed. This entry opens a dialog window where users can select which columns will be displayed/hidden.

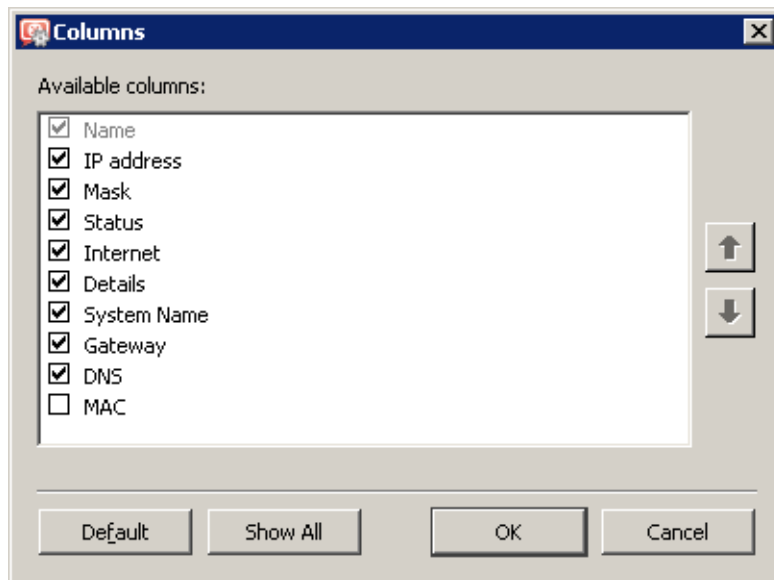


Figure 3.3 Column customization in Interfaces

This dialog offers a list of all columns available for a corresponding view. Use checking boxes on the left to enable/disable displaying of a corresponding column. You can also click the *Show all* button to display all columns. Clicking on the *Default* button will restore default settings (for better reference, only columns providing the most important information are displayed by default).

The arrow buttons move the selected column up and down within the list. 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.

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

Chapter 4

Product Registration and Licensing

When purchased, *Kerio WinRoute Firewall* must be registered. Upon registration of the product, so called license key is generated (the `license.key` file — see chapter [25.1](#)). 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 means that the trial version differs from the full *WinRoute* version only in the aspect whether the license has been registered or not. This gives each customer an opportunity to test and try the product in the particular environment during the 30-day period. Then, once the product is purchased, the customer can simply register the installed version by the purchased license number (see chapter [4.3](#)). This means that it is not necessary to uninstall the trial version and reinstall the product.

Once the 30-day trial period expires, *WinRoute* cuts the speed of all network traffic of the computer where it is installed to 4 KB/s. Also, the routing is blocked (which implies that the *WinRoute's* host cannot be used as a gateway for the Internet). Upon registration with a valid license number (received as a response to purchase of the product), *WinRoute* is available with full functionality.

Note: If your license key gets lost for any reason (e.g. after the hard drive breakdown or by an accidental removal, etc.), you can simply use the basic product's purchase number to recover the license. The same method can be used also for change of the firewall's operating system (*Windows / Software Appliance / VMware Virtual Appliance*) — the license keys cannot be used across different operating systems. If the license number gets lost, contact the *Kerio Technologies* sales department.

4.1 License types and number of users

License types (optional components)

WinRoute can optionally include the following components: *McAfee* antivirus (refer to chapter [13](#)) or/and the *Kerio Web Filter* module for web pages rating (see chapter [12.3](#)). 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.
- plug-in expiration date— specifies the date by which the *McAfee* antivirus stops functioning and cannot be used anymore.

Warning

Owing to persistent incidence of new virus infections we recommend you to use always the most recent antivirus versions.

Kerio Web Filter subscriptions

Kerio Web Filter module is provided as a service. License is defined only by an expiration date which specifies when this module 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.

Deciding on a number of users (licenses)

WinRoute's license key includes information about maximal number of users allowed to use the product. In accordance with the licensing policy, number of users is number of hosts protected by *WinRoute*, i.e. sum of the following items:

- All hosts in the local network (workstations and servers),
- all possible VPN clients connecting from the Internet to the local network.

The host where *WinRoute* is installed is not included in the total number of users.

Warning

If the maximal number of licensed users is exceeded, *WinRoute* may block traffic of some hosts!

4.2 License information

The license information can be displayed by selecting *Kerio WinRoute Firewall* (the first item in the tree in the left part of the *Administration Console* dialog window — this section is displayed automatically whenever the *WinRoute* administration is entered).



Figure 4.1 Administration Console welcome page providing license information

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.

Operational system

Name of the operating system on which the *WinRoute Firewall Engine* service is running. This is an informative item only — the purchased license can be used for any supported operating system.

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 (for details, refer to chapter [4.6](#)).

Company

Name of the company (or a person) to which the product is registered.

Depending on the current license, links are displayed at the bottom of the image:

1. For unregistered versions:

- *Become a registered trial user* — registration of the trial version. This type of registration is tentative and it is not obligatory. The registration provides users free technical support for the entire trial period.
- *Register product with a purchased license number* — registration of a purchased product.

Once purchased, the product must be registered. Otherwise, it will keep behaving as a trial version!

2. For registered versions:

- *Update registration info* — this link can be used to update information about the person/company to which the product is registered and/or to add subscription license numbers or add-on licenses (add users).

In any case, the registration wizard will be started where basic data are required and additional data can also be defined. For detailed information on the wizard, refer to chapter [4.3](#).

If the update checker is enabled (refer to chapter [16.3](#)), the *A new version is available, click here for details...* notice is displayed whenever a new version is available. Click on the link to open the dialog where the new version can be downloaded and the installation can be started (for details, see chapter [16.3](#)).

Note: Right-clicking in the main page of the *Administration Console* opens a context pop-up menu with the same options as are provided in the *Edit* menu in the main toolbar of the administration window (see chapter [3.1](#)).

4.3 Registration of the product in the Administration Console

WinRoute registration, change of registration details, adding of add-on licenses and subscription updates can be done in the *Administration Console* by clicking on a corresponding link on the welcome page (see chapter [4.2](#)) or by using a corresponding option in the *Edit* menu in the main menu for the administration window (see chapter [3.1](#)).

Registration of the trial version

By registering the trial version, users get free email and telephonic technical support for the entire trial period. In return, *Kerio Technologies* gets valuable feedback from these users. Registration of the trial version is not obligatory. However, it is recommended since it provides certain benefits. Such a registration *does not oblige* users to purchase the product.

Clicking on *Become a registered trial user* launches the registration wizard.

1. On the first page of the wizard, read the security code displayed in the picture and type it to the text field (this protects the registration server from misuse). The security code is not case-sensitive.

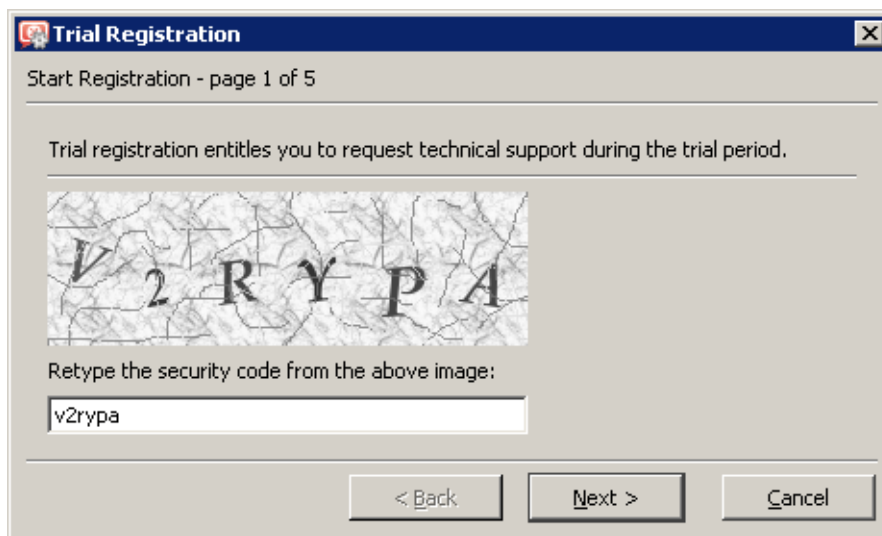


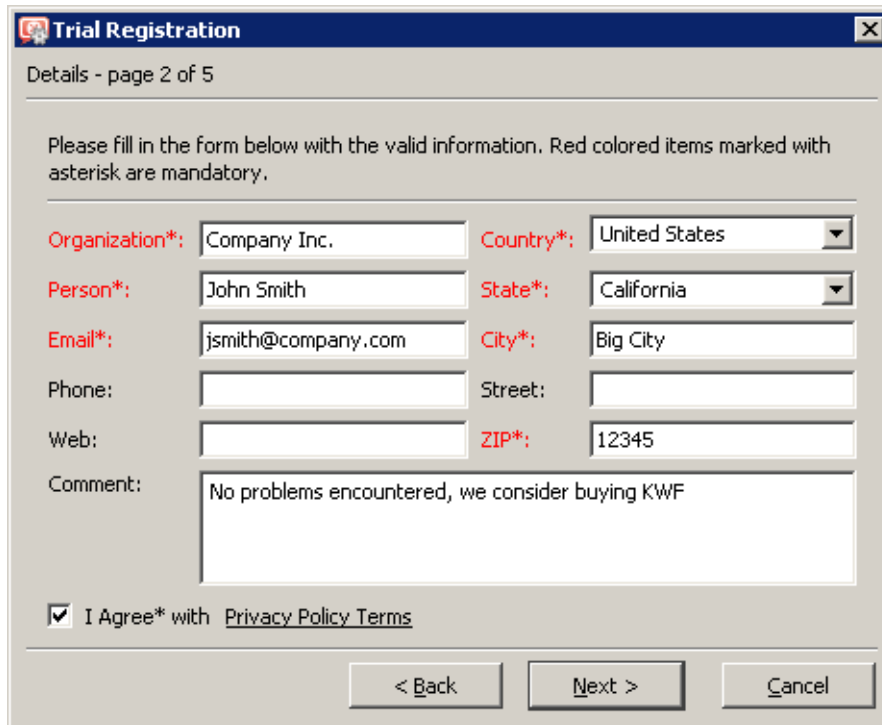
Figure 4.2 Trial version registration — security code

2. On the second page, enter information about the trial version user (person, company). It is also necessary that the user accepts the *Privacy Policy Terms*. Otherwise, the information cannot be stored in the *Kerio Technologies* database.

Use the *E-mail address* textfield to enter a valid email address. It is recommended to use the address of the user who is performing the registration. At this address, confirmation of the registration will be demanded when the registration is completed.

3. Page three includes optional information. It is not obligatory to answer these questions, however, the answers help *Kerio Technologies* accommodate demands of as many customers as possible.

4.3 Registration of the product in the Administration Console



Trial Registration [X]

Details - page 2 of 5

Please fill in the form below with the valid information. Red colored items marked with asterisk are mandatory.

Organization*: Company Inc. Country*: United States

Person*: John Smith State*: California

Email*: jsmith@company.com City*: Big City

Phone: Street:

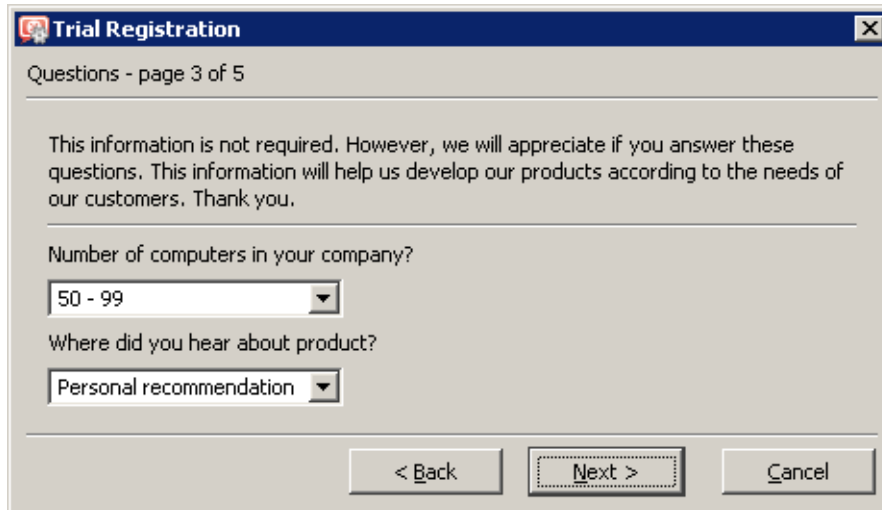
Web: ZIP*: 12345

Comment:
No problems encountered, we consider buying KWF

I Agree* with [Privacy Policy Terms](#)

< Back Next > Cancel

Figure 4.3 Trial version registration — user information



Trial Registration [X]

Questions - page 3 of 5

This information is not required. However, we will appreciate if you answer these questions. This information will help us develop our products according to the needs of our customers. Thank you.

Number of computers in your company?
50 - 99

Where did you hear about product?
Personal recommendation

< Back Next > Cancel

Figure 4.4 Trial version registration — other information

4. The fourth page provides the information summary. If any information is incorrect, use the *Back* button to browse to a corresponding page and correct the data.
5. The last page of the wizard provides user's *Trial ID*. This ID is a unique code used for identification of the registered user when asking help at our technical support.

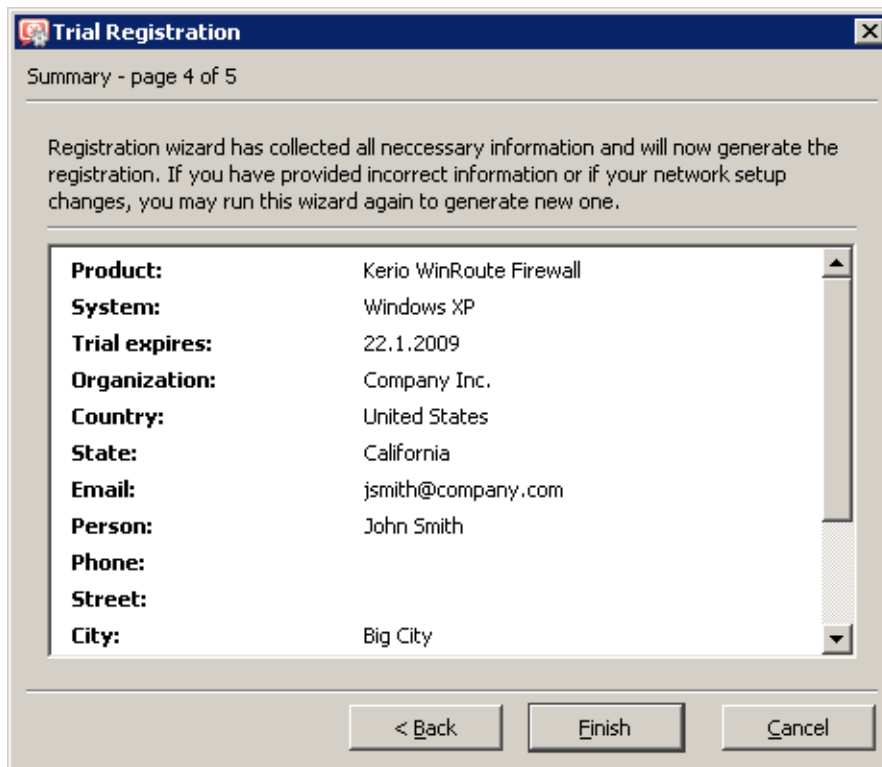


Figure 4.5 Registration of the trial version — summary

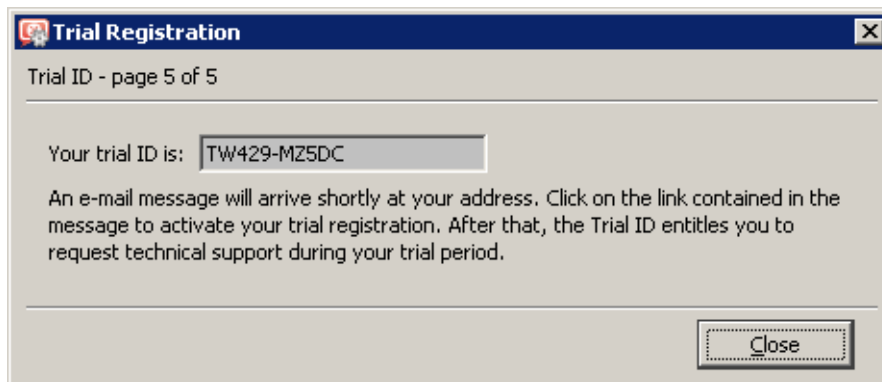


Figure 4.6 Trial version registration — Trial ID

At this point, an email message (in the language set in the *Administration Console*) where confirmation of the registration is demanded is sent to the email address specified on the page two of the wizard. Click on the link in the email message to complete the registration and to make the *Trial ID* valid. The main purpose of the confirmation process is to check that the email address is valid and that the user really wants to be registered.

Registration of the purchased product

Follow the *Register product with a purchased license number* link to run the registration wizard.

1. On the first page of the wizard, it is necessary to enter the license number of the basic product delivered upon its purchase and retype the security code displayed at the picture in the text field (this protects the server from misuse). The security code and the license number are not case-sensitive.

Figure 4.7 Product registration — license number of the basic product and the security code

2. On the second page, it is possible to specify license numbers of add-ons (added users), optional components and subscriptions. The page also includes any license numbers associated with the basic product that have already been registered.

Click on *Add* to add purchased license numbers. Each number is checked immediately. Only valid license numbers are accepted.

The license numbers added recently can be edited or removed. Registered license numbers (recorded in previous registrations) cannot be removed.

3. On the third page, enter information about the user (person, company). It is also necessary that the user accepts the *Privacy Policy Terms*. Otherwise, the information cannot be stored in the *Kerio Technologies* database.

Use the *E-mail address* textfield to enter a valid email address. It is recommended to use the address of the user who is performing the registration. At this address, confirmation of the registration will be demanded when the registration is completed.

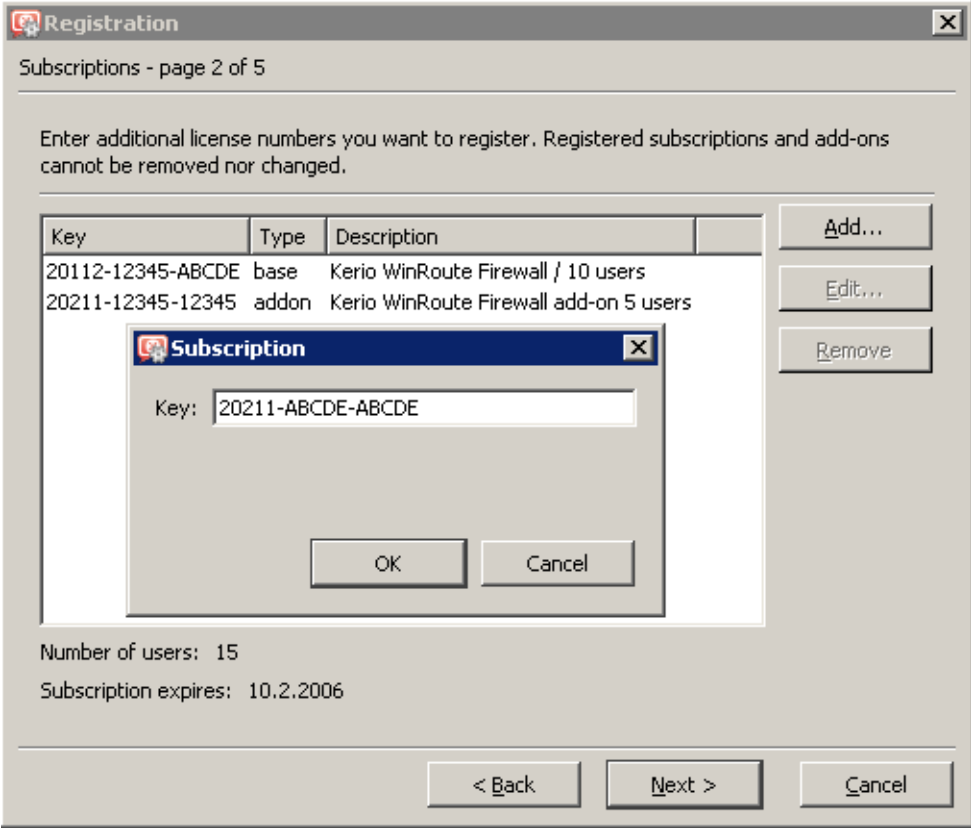


Figure 4.8 Product registration — license numbers of additional components, add-ons and subscription

Registration

Details - page 3 of 5

Please fill in the form below with the valid information. Red colored items marked with asterisk are mandatory.

Organization*: Company Inc. Country*: United States

Person*: John Smith State*: California

Email*: jsmith@company.com City*: Big City

Phone: Street:

Web: www.company.com ZIP*: 12345

Comment: No comment

I Agree* with [Privacy Policy Terms](#)

< Back Next > Cancel

Figure 4.9 Product registration — user information

4. Page four includes optional information. It is not obligatory to answer these questions, however, the answers help *Kerio Technologies* accommodate demands of as many customers as possible.

These questions are asked only during the primary (original) registration. If these questions have already been answered, the page is skipped and the registration process consists of four steps only.

5. The last page provides the information summary. If any information is incorrect, use the *Back* button to browse to a corresponding page and correct the data.

Click on *Finish* to use the information to generate a unique license key. The new license is applied immediately (restart is not required).

Notes:

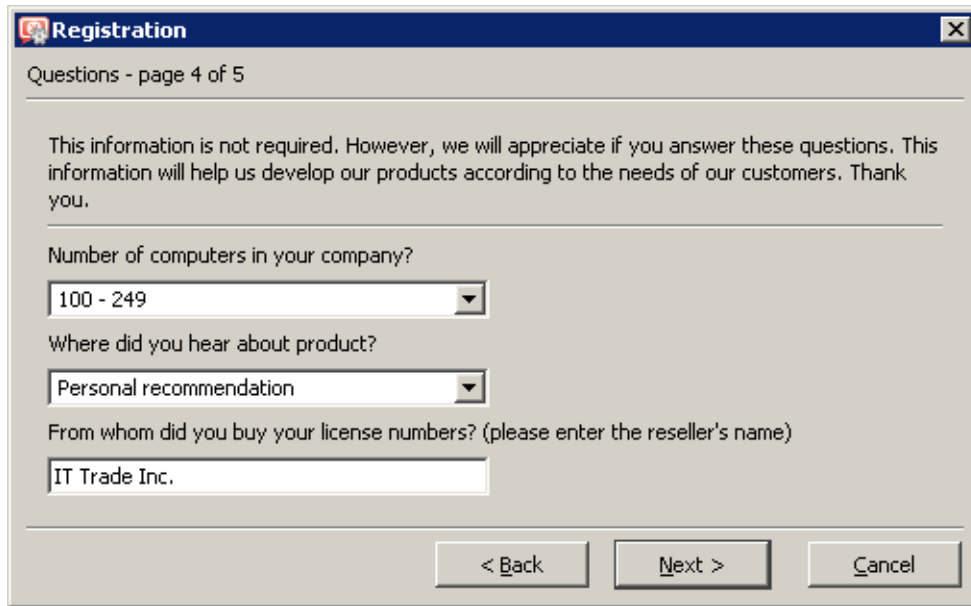


Figure 4.10 Product registration — other information

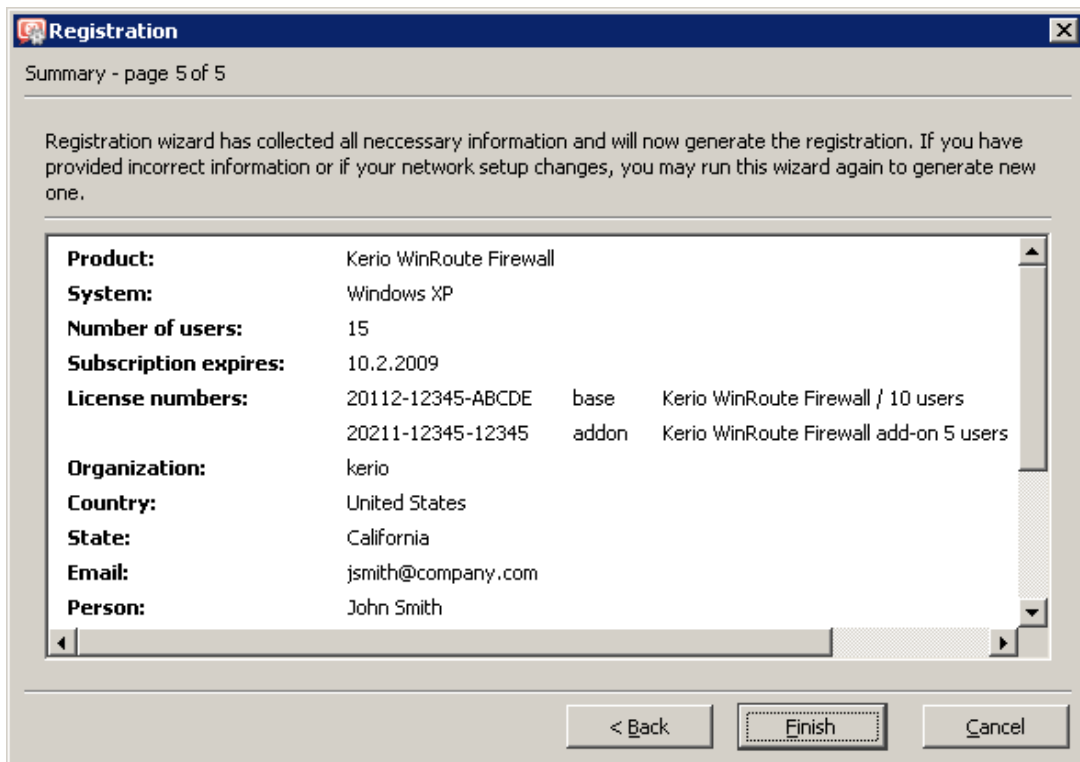


Figure 4.11 Product registration — summary

1. The license key is generated only for the operating system on which *WinRoute* was installed during the registration (*Windows / Linux*). The license can be used for any platform but the license key is always generated for the particular platform only.
2. If an error is reported upon finishing of the registration process (e.g. failure of net-

work connection, etc.), simply restart the wizard and repeat the registration.

4.4 Product registration at the website

If, by any reason, registration of *WinRoute* cannot be performed from the *Administration Console*, it is still possible to register the product at *Kerio Technologies* website. To open the registration form, use the *Support* → *Register License* option in the main menu.

The form is similar to the registration wizard described in chapter 4.3. The corresponding license key file is based on the registration form and it is automatically generated upon its completion and confirmation.

In the registration, specify correctly the operating system you will use the license on (*Windows* or *Linux*). The license can be used for any platform but the license key is always generated for the particular platform only.

License key installation

Two methods can be used to install the license key:

- By using the *Install license* in the *Edit* menu available in the main toolbar of the administration window (see chapter 3.1). Click this link to open the standard system dialog for opening of a file.

If the installation of the license key is completed successfully, the license is activated immediately. Information about the new license is displayed on the *Administration Console* welcome page.

This method can also be used for remote installation of the license key (the license key file must be saved on the disk of the host from which the remote installation is performed).

- By copying the license key file to a corresponding directory.

The license key must be saved in the `license` folder in the *WinRoute*'s installation directory.

(the typical path is `C:\Program Files\Kerio\WinRoute Firewall\license`)

It is necessary that the file name (`license.key`) is not changed!

To activate the license, it is necessary to restart (stop and run again) the *WinRoute Firewall Engine*.

Note: If possible, it is recommended to register *WinRoute* from the *Administration Console* (it is not necessary to restart the *WinRoute Firewall Engine*).

4.5 Subscription / Update Expiration

WinRoute automatically alerts the administrator in case the *WinRoute* license's expiration date, the expiration of the *McAfee* antivirus or of *Kerio Web Filter* and/or expiration of the update rights (so called subscription) for *WinRoute* or the *McAfee* antivirus is coming soon. These alert only inform the administrator that they should prolong the subscription of *WinRoute* or renew the corresponding license.

Administrators are informed in two ways:

- By a pop-up bubble tip (this function is featured by the *WinRoute Engine Monitor* module),
- by an pop-up window upon a login to the *Administration Console* (only in case of expiration of subscription).

Note: *WinRoute* administrators can also set posting of license or subscription expiration alerts by email or SMS (see chapter [19.4](#)).

Bubble alerts (Windows)

Seven days before the date, *WinRoute Engine Monitor* starts to display the information about number of days remaining to the subscription/license expiration several times a day (in regular intervals).

This information is displayed until *WinRoute* or any of its components stops functioning or *WinRoute* or *McAfee* subscription expires. The information is also stopped being displayed immediately after the registration of the subscription or a license of a particular component (for details, see chapter [4.3](#)).

Notices in the Administration Console

Starting 30 days ago a subscription expiration, a warning informing about number of the days left to the expiration or informing that the subscription has already expired is displayed upon each login. The warning also contains a link to the *Kerio Technologies* website where you can find detailed subscription information as well as order subscription for an upcoming period.

The warning stops being displayed when a license number of a new subscription is registered (refer to chapter [4.3](#)).

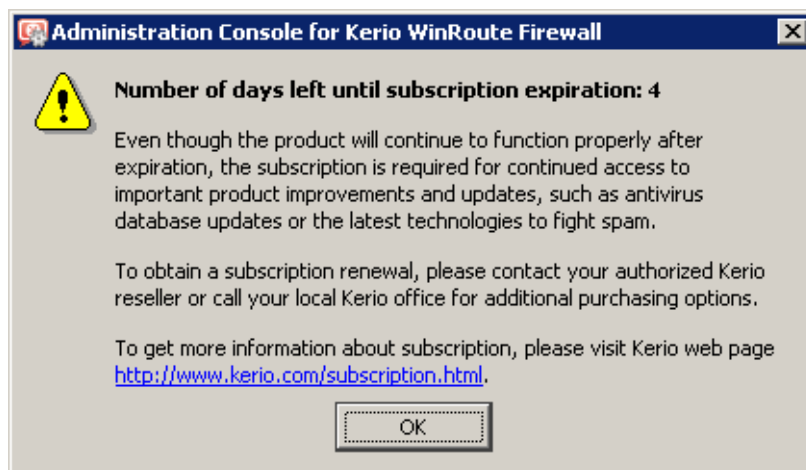


Figure 4.12 The notice informing about upcoming subscription expiration

4.6 User counter

This chapter provides a detailed description on how *WinRoute* checks whether number of licensed users has not been exceeded.

The *WinRoute* license does not limit number of user accounts. Number of user accounts does not affect number of licensed users.

Warning

The following description is only a technical hint that may be used for troubleshooting. License policy must be borne in mind when deciding for a license purchase — see chapter [4.1!](#)

The license counter works as follows:

Start WinRoute

Upon *WinRoute* is started, the table of clients include the firewall only. Number of used licenses is zero.

Note: Table of clients is displayed in the *Active Hosts* section in the *Administration Console* — see chapter [19.1](#).

License counter

Whenever a communication of any *WinRoute's* client is detected, the IP address is used to identify whether a record does already exist in the table of clients. If not, a new record including the IP address is added to the table and the number of licenses is raised by 1.

The following items are considered as clients:

1. All hosts from which users are connected to the firewall
2. All clients of the *WinRoute's* proxy server (see chapter [8.4](#))
3. All local hosts communication of which is routed between Internet interfaces and *WinRoute's* local interfaces. The following items belong to this group:
 - Each host which is connected to the Internet while no user is authenticated from the host,
 - All local servers mapped from the Internet,
 - All VPN clients connected to the local network from the Internet.

Licenses are not limited by:

- DNS requests handled by the *DNS* module (*Warning:* If clients use a DNS server located outside the local network, such communication is considered as communication with the Internet),
- DHCP traffic (using either the *WinRoute's DHCP server* or another DHCP server installed on the *WinRoute* host),
- Local communication between the firewall (e.g. access to shared disks) and hosts from which no user is connected to the firewall.

License release

Idleness time (i.e. time for which no [packet](#) with a corresponding IP address meeting all conditions is detected) is monitored for each record in the table of clients. If the idleness time of a client reaches 15 minutes, the corresponding record is removed from the table and the number of licenses is decreased by 1. Released license can be used by another host.

Chapter 5

Network interfaces

WinRoute is a network firewall. This implies that it represents a gateway between two or more networks (typically between the local network and the Internet) and controls traffic passing through network adapters (*Ethernet*, *WiFi*, dial-ups, etc.) which are connected to these networks.

WinRoute functions as an IP router for all *WinRoute*'s network interfaces installed within the system.³ The linchpin of the firewall's configuration therefore is correct configuration of network interfaces.

Network interfaces of the firewall can be displayed and configured in the *Administration Console* or in the *Web Administration's Configuration* → *Interfaces* section.

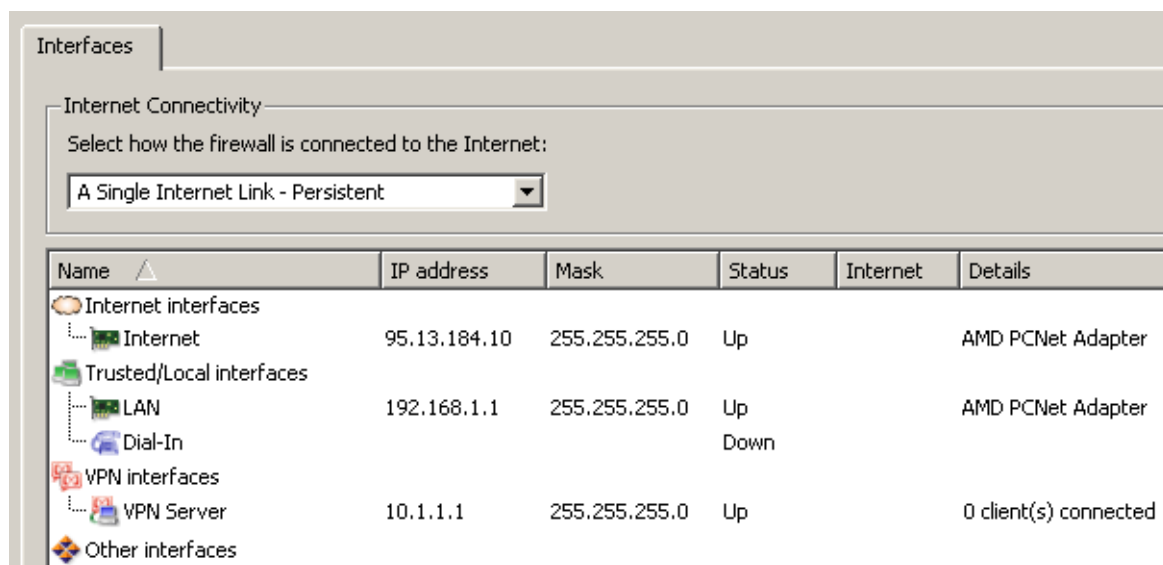


Figure 5.1 Network interfaces

Groups of interfaces

To simplify the firewall's configuration and make it as comfortable as possible, network interfaces are sorted in groups in *WinRoute*. In the firewall's traffic rules, these groups as well as individual interfaces can be used in *Source* and *Target* (refer to chapter 7.3). The main benefit of groups of interfaces is that in case of change of internet connection, addition of a new line,

³ If you want to disable *WinRoute* for any of these interfaces, go to the adapter's properties and disable *Kerio WinRoute Firewall* (the *WinRoute*'s low level driver). However, for security reasons and to guarantee full control over the network traffic, it is strongly unrecommended to disable *WinRoute*'s low level driver on any network adapter!

change of a network adapter etc., there is no need to edit traffic rules — simple adding of the new interface in the correct group will do.

In *WinRoute*, the following groups of interfaces are defined:

- *Internet interfaces* — interfaces which can be used for Internet connection (network cards, wireless adapters, dial-ups, etc.),
- *Trusted / Local interfaces* — interfaces connected to local private networks protected by the firewall (typically *Ethernet* or *WiFi* cards),
- *VPN interfaces* — virtual network interfaces used by the *Kerio VPN* proprietary solution (VPN server and created VPN tunnels — for details, refer to chapter [23](#)),
- *Other interfaces* — interfaces which do not belong to any of the groups listed above (i.e. a network card for [DMZ](#), idle dial-up, etc.).

Groups of interfaces cannot be removed and it is not possible to create new ones (it would not be of any help).

During the initial firewall configuration by *Traffic rules wizard* (see chapter [7.1](#)), interfaces will be sorted in correct groups automatically. This classification can be later changed (with certain limits — e.g. VPN server and VPN tunnels cannot be moved from the *VPN interfaces* group).

To move an interface to another group, drag it by mouse to the desired destination group or select the group in properties of the particular interface — see below.

Note: If the initial configuration is not performed by the wizard, all interfaces (except VPN interfaces) are set as *Other interfaces*. Before you start creating traffic rules, it is recommended to define correctly interfaces for Internet connection as well as interfaces for the local network — this simplifies definitions of the rules significantly.

Special interfaces

Interfaces include also the following special items:

VPN server

This interface is used as a server for connection of the proprietary VPN client (*Kerio VPN Client* — this solution can be downloaded for free from <http://www.kerio.com/firewall/download>). VPN servers are always sorted in the *VPN interfaces* group.

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 solution *Kerio VPN*, refer to chapter [23](#).

Dial-In (on Windows only)

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 [7](#)) for *RAS* clients which are connecting to this server.

Dial-In interfaces are considered as trustworthy (clients connected via this interface use it to access the local network). This interface cannot be either configured or removed. If

you do not consider RAS clients as parts of trustworthy networks for any reason, you can move the *Dial-In* interface to *Other interfaces*.

Note:

1. If both RAS server and *WinRoute* are used, the RAS server must be configured to assign clients IP addresses of a subnet which is not used by any segment of the local network. *WinRoute* performs standard IP routing which might not function unless this condition is met.
2. For assigning of IP addresses to RAS clients connecting directly to the *WinRoute* host, *it is not possible* to use the *WinRoute's* DHCP server. For details, see chapter [8.2](#).

Viewing and editing interfaces

In the list of interfaces, *WinRoute* shows parameters related to firewall's configuration and operations:

Name

The unique name used for interface identification within *WinRoute*. It should be clear for easy reference, e.g. *Internet* for the interface connected to the Internet connection.

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, 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.

If the more IP addresses are set for the interface, the primary IP address will be displayed.

On *Windows*, the address assigned to the interface as first is considered as primary.

Status

Current status of the interface (up/down).

Internet

This information indicates the method the interface uses for Internet connection (primary/secondary connection, bandwidth used).

Details

Adapter identification string returned by the device driver.

System Name

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

Gateway

IP address of the default gateway set for the particular interface.

DNS

IP address of the primary DNS server set on the interface.

MAC

Hardware (MAC) address of a corresponding network adapter. This entry is empty for dial-ups as its use would be meaningless there.

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 VPN Tunnel

Use this option to create a new server-to-server VPN tunnel. Details on the proprietary *Kerio VPN* solution are provided in chapter 23.

Note: In *Software Appliance / VMware Virtual Appliance*, it is also possible to add new interfaces (dial-up, PPTP or PPPoE connections) — see section *Adding new interface*. If *WinRoute* is installed on *Windows*, it is necessary to define new connections by standard methods right in the operating system.

Modify

Click on *Edit* to view and/or modify parameters of the selected interface.

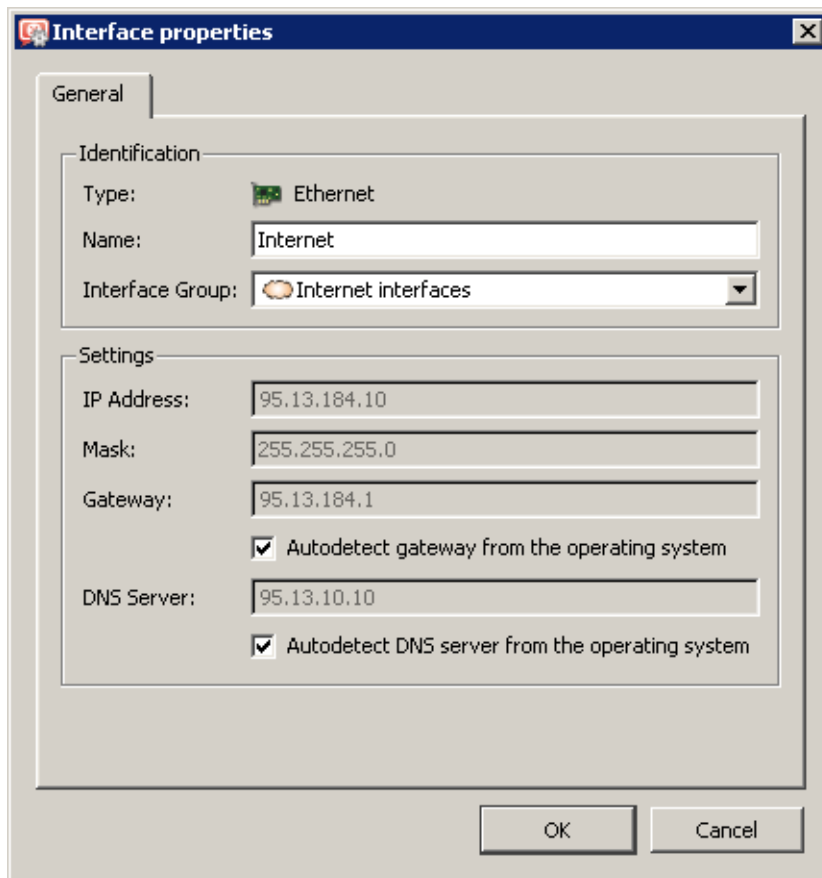


Figure 5.2 Editing interfaces

In *WinRoute*, it is specify to specify a special name for each interface (names taken from the operating system can be confusing and the new name may make it clear). It is also possible to change the group of the interface (Internet, secure local network, another network — e.g. [DMZ](#)).

It is also possible to change the default gateway and edit parameters of DNS servers. In the *Software Appliance / VMware Virtual Appliance* edition, all parameters of the network interface can be set in this dialog.

For dial-ups it is also possible to set login data and dialing options (see chapter [6.2](#)).

For *VPN server* and VPN tunnels, a dialog for setting of the *VPN server* (see chapter [23.1](#)) or a VPN tunnel (refer to chapter [23.3](#)) will be opened.

Remove

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

- the interface is an inactive (disabled) VPN tunnel,
- the network adapter is not active or it is not physically present,
- the interface is a dial-up which no longer exists in the system.

Network cards and dial-ups defined in the operating system as well as established VPN tunnels cannot be removed in *WinRoute*.

Note:

1. Records related to network cards or dial-ups that do not exist any longer (those that have been removed) do not affect *WinRoute*'s functionality — such interfaces are considered as inactive (as in case of a hung-up dial-up).
2. When an adapter is removed, the *Nothing* value is automatically used for corresponding items of all traffic rules where the interface was used. These rules will be disabled. This ensures that the traffic policy is not endangered (for details, refer to chapter [7.3](#)).

Dial or Hang Up / Enable, Disable

Function of these buttons depend on the interface selected:

- For dial-up, PPTP and PPPoE connections, the *Dial* and *Hang-up* buttons are available and they are used to handle the line by hand.

Note: Users with appropriate rights can also control dial-ups in the user web interface (see chapter [15.2](#) and the *Kerio WinRoute Firewall — User's Guide*).

- For VPN tunnels, the *Enable* and *Disable* buttons are available that can be used to enable /disable the VPN tunnel selected for details, see chapter [23.3](#)).

In the *Software Appliance / VMware Virtual Appliance* edition, it is also possible to block individual network adapters.

- If the *Dial-in* interface or a VPN server is selected, these buttons are inactive.

Adding new interface (Software Appliance / VMware Virtual Appliance)

In the *Software Appliance / VMware Virtual Appliance* edition, *WinRoute* allows to add new network interfaces (dial-up, PPPoE and PPTP connections) right in the administration console.

Click on *Add* to open a menu and select type of the new interface (dial-up can be added only if an analog or ISDN modem is installed on the firewall host).

The new interface needs an easily identifiable name that will be showed in *WinRoute* and it needs to be added to a group of interfaces (this item can be changed as desired any time later).

Other parameters of the interface depend on the selected interface type. Most types require username and password for access verification.

Optionally, you can specify IP address of a specific DNS server which will then be used as the primary DNS server for Internet connections via this interface.

The *Dialing settings* can be used to set time intervals in which the connection should be established persistently and when it should be disconnected. Out of these intervals, the connection will be established on demand (i.e. it will be established automatically any time *WinRoute* needs to send a [packet](#) to the corresponding network). For details about on-demand dialing, see chapters [6.2](#) and [25.5](#).

Internet Connection

The basic function of *WinRoute* is connection of the local network to the Internet via one or more Internet connections (Internet links). Depending on number and types of Internet links, *WinRoute* provides various options of Internet connection:

A Single Internet Link — Persistent

The most common connection of local networks to the Internet. In this case, only one Internet connection is available and it is used persistently (typically *Ethernet*, *WiFi*, *ADSL* or cable modems). It is also possible to use dial-like links which can be connected persistently, such as *PPPoE* connections or *CDMA* modems.

A Single Internet Link — Dial On Demand

This type of connection is fit for links which are charged by connection time — typically modems for analog or *ISDN* links. The link is down by default and *WinRoute* dials it in response to a query demanding access from the local network to the Internet. If no data are transferred via the link for some time, *WinRoute* hangs it up to reduce connection costs.

Multiple Internet Links — Failover

Where reliability (availability of the Internet connection) is an issue and two Internet links are available, the connection failover feature can help. If the primary link fails, *WinRoute* switches to the secondary link automatically. Users may therefore notice just a very short disconnection of the Internet connection. When the connection on the primary link is recovered, *WinRoute* automatically switches back to it. For most part of users, this operation takes so short to be even noticeable.

Multiple Internet Links Traffic Load Balancing

If throughput (connection speed) is an issue, *WinRoute* can use multiple links concurrently and spread data transferred between the LAN and the Internet among these links. In standard conditions and settings, this also works as connection failover — if any of the links fails, transferred data are spread among the other (working) links.

In all cases, *WinRoute* works in the mode of shared Internet connection. Sharing uses the [NAT](#) (IP address translation) technology, hiding the entire local network behind a public IP address of the firewall (or multiple addresses — depending on the type of Internet connection applied). *WinRoute* can also be used as a neutral [router](#) (router without NAT). However, this mode is not the best connection of the LAN to the Internet — it requires expert configuration and advanced security.

This involves selection of the Internet connection type in the *Configuration* → *Interfaces* section of the *WinRoute* configuration, setting corresponding interfaces for connection to the Internet and definition of corresponding traffic rules (see chapter [7.3](#)).

Hint

All necessary settings can be done semi-automatically with use of *Traffic Policy Wizard* — see chapter [7.1](#). Following chapters provide with guidelines for setting of individual Internet connection types as well as with description on configuration of the corresponding interface and traffic rules in the wizard. The information available there can be used for customization of settings (e.g. for setting of a new local subnetwork or for change of Internet connection).

6.1 Persistent connection with a single link

Requirements

The *WinRoute* hosting computer must be connected to the Internet by a leased line (typically *Ethernet* or *WiFi* card). Parameters of this interface will be set with use of information supplied by the ISP provider or they can be configured automatically with the DHCP protocol.

It is also possible to use a dial-like link which can be connected persistently, such as *PPPoE* connections or *CDMA* modems. *WinRoute* will keep this type of link connected persistently (in case of connection failure, the connection is automatically recovered immediately).

This connection type also requires one or more network cards for connection of individual segments of the LAN. Default gateway must *NOT* be set on any of these cards!

If possible, it is also recommended functionality of the Internet connection before installing *WinRoute*.

Configuration with the wizard

On the second page of the *Traffic Policy Wizard* (see chapter [7.1](#)), select *A Single Internet Link* — *Persistent*.

On the third page of the wizard, select a network interface (Internet link). As a preselection, the interface where *WinRoute* detected the default gateway is used. Therefore, in most cases the appropriate adapter is already set within this step.

If you select a link which is defined as a dial-up (see above), valid username and password are required. If this information is saved in the operating system, *WinRoute* can enter it automatically.

In the *Software Appliance* / *VMware Virtual Appliance* edition, the wizard allows:

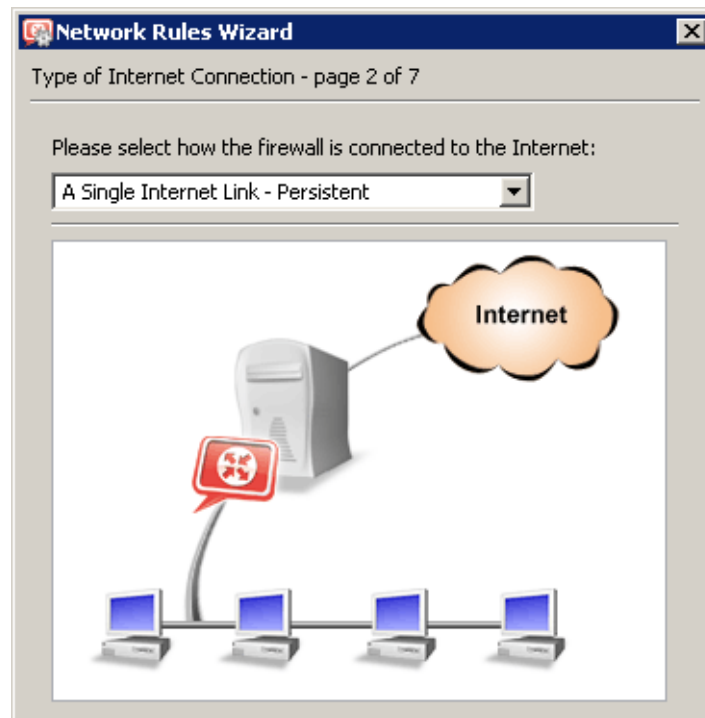


Figure 6.1 Traffic Policy Wizard — persistent connection with a single link

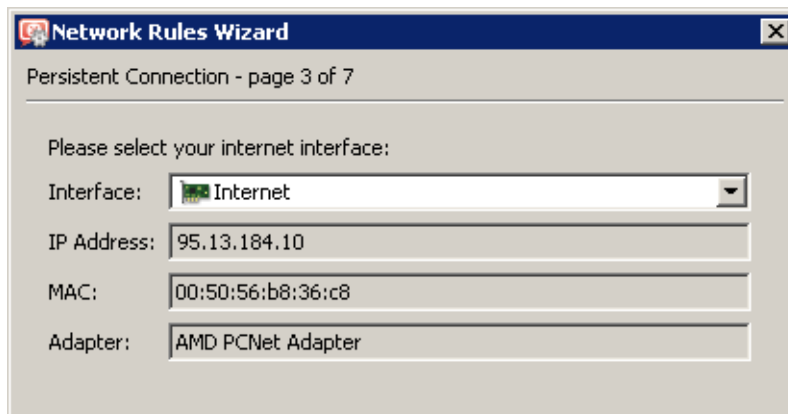


Figure 6.2 Network Policy Wizard — selection of an interface for the Internet connection

- to configure parameters of the selected interface,
- to create a new interface (*PPPoE*, *PPTP* or dial-up).

For details on network interfaces, see chapter [5](#).

Notes:

1. On the top of the list, the Internet interface where the default gateway is set is offered. Therefore, in most cases the appropriate adapter is already set within this step.
2. If the more IP addresses are set for the interface, the primary IP address will be displayed. On *Windows*, the address assigned to the interface as first is considered as primary.
3. The other pages of the *Traffic Policy Wizard* do not concern Internet connection type. They are focused in detail in chapter [7.1](#)

Resulting interface configuration

When you finish set-up in *Traffic Policy Wizard*, the resulting configuration can be viewed under *Configuration* → *Interfaces* and edited if desirable.

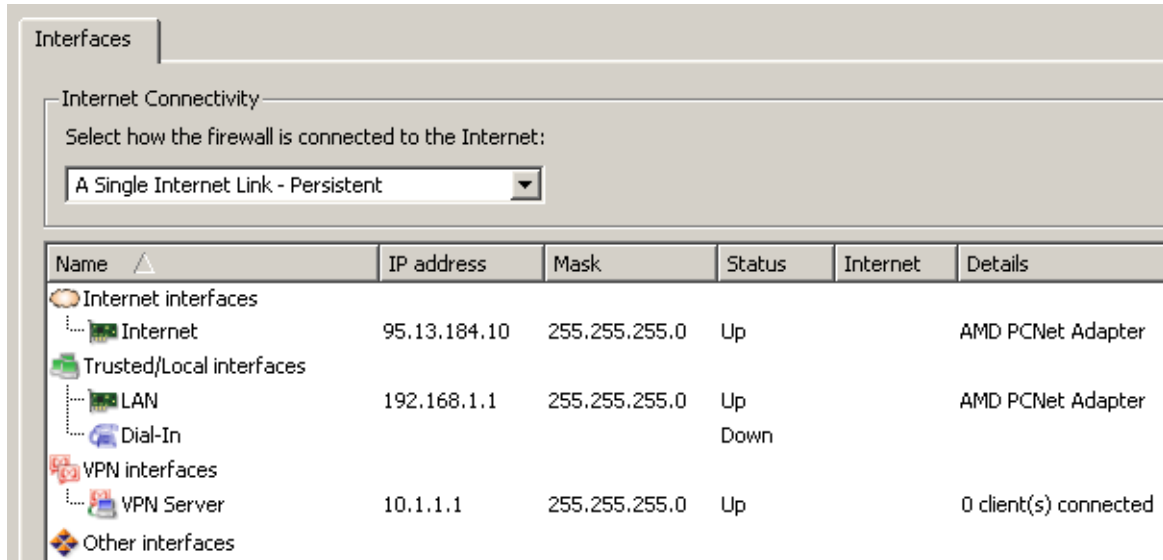


Figure 6.3 Configuration of interfaces — connection by a single leased link

The *Internet Interfaces* groups includes only card *Internet* selected in the third page of the wizard. Other interfaces (including *Dial-In*) are considered as segments of the LAN and put in *Trusted / Local interfaces*.

If the setting does not mirror the real configuration of the network correctly (for instance there is an interface planned for [DMZ](#)), you can move the particular interface to *Other Interfaces*. For these interfaces, it will be necessary to define corresponding traffic rules manually (see chapter [7.3](#)).

It is also possible to add new interfaces to the *Internet Interfaces* group. [Packets](#) will then be routed to corresponding target networks in accordance with the system routing table (see also chapter [18.1](#)) and IP address translation will be applied ([NAT](#)). However, such configuration is not significantly helpful in place.

Warning

It is necessary that in the *Single internet Link* mode the default gateway is set only at the “main” Internet interface! If *WinRoute* detects more default gateways, error is announced. Solve this problem immediately, otherwise traffic from the firewall and the LAN to the Internet will not work correctly.

6.2 Connection with a single leased link - dial on demand

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 Dial on demand. For further description see below.
- Line is disconnected automatically if idle for a certain period (no data is transmitted in both directions).
- Maintenance of persistent connection or disconnection of the link within defined time ranges.

Requirements

The corresponding device must be installed on the *WinRoute* (usually an analog or an ISDN modem) and the corresponding dial-up connection must be created in the operating system. It is not necessary to define and save login data in the dial-up settings, this information can be defined directly in *WinRoute*. This connection type also requires one or more network cards for connection of individual segments of the LAN. Default gateway must *NOT* be set on any of these cards!

We recommend you to create and test a dial-up connection before installing *WinRoute*.

Warning

Before configuring the LAN and the firewall for a Internet link dialed on demand, please pay special attention to the information provided in chapter [25.5](#). Correct configuration of the network with respect to specific qualities and behavior of on demand dial helps to avoid subsequent problems.

Configuration with the wizard

On the second page of the *Traffic Policy Wizard* (see chapter [7.1](#)), select *A Single Internet Link — Dial on Demand*.

On the third page of the wizard, select a corresponding dial-up connection (Internet link). If authentication data are not saved in the operating system, username and password are required.

In the *Software Appliance / VMware Virtual Appliance* edition, the wizard allows:

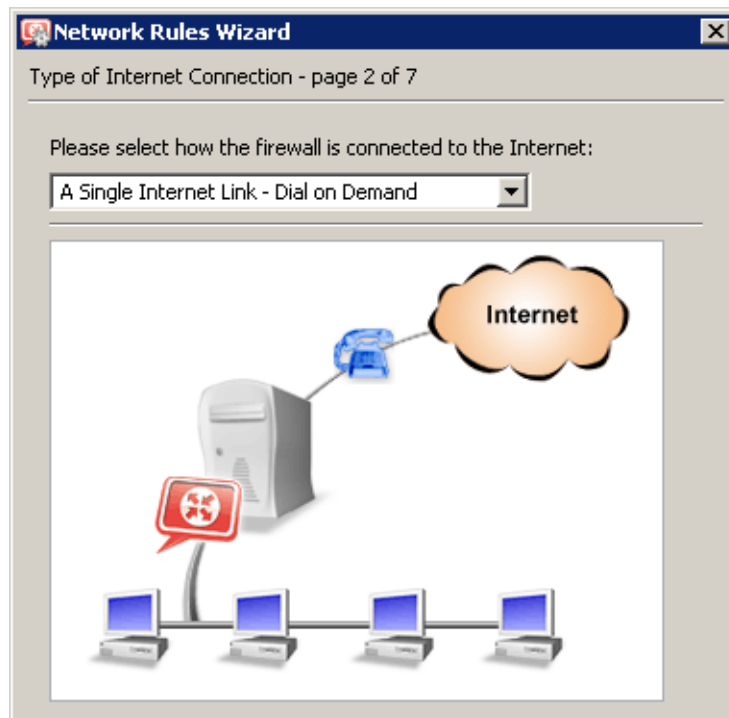


Figure 6.4 Traffic Policy Wizard — dial on demand

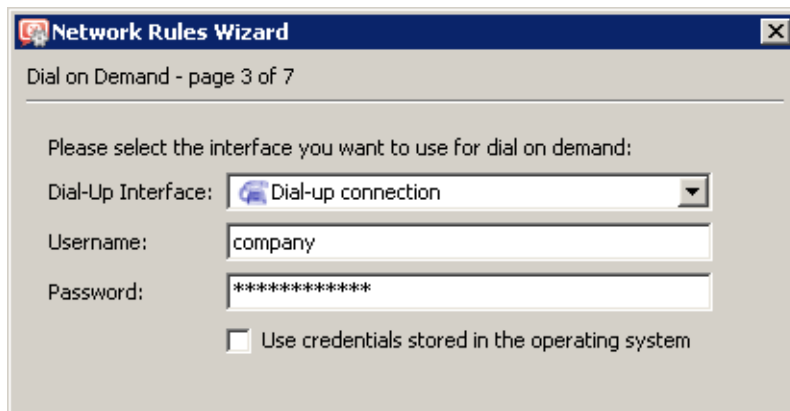


Figure 6.5 Network Policy Wizard — selection of an interface for the Internet connection

- to configure parameters of the selected interface,
- to create a new interface (*PPPoE*, *PPTP* or dial-up).

For details on network interfaces, see chapter 5.

Resulting interface configuration

When you finish set-up in *Traffic Policy Wizard*, the resulting configuration can be viewed under *Configuration* → *Interfaces* and edited if desirable.

The *Internet Interfaces* group includes only the *Dial-up connection* link selected in the third page of the wizard. This connection is set up as a dial-on-demand link (see information in the column labeled as *Internet*). Other interfaces (including *Dial-In*) are considered as segments of the LAN and put in *Trusted / Local interfaces*.

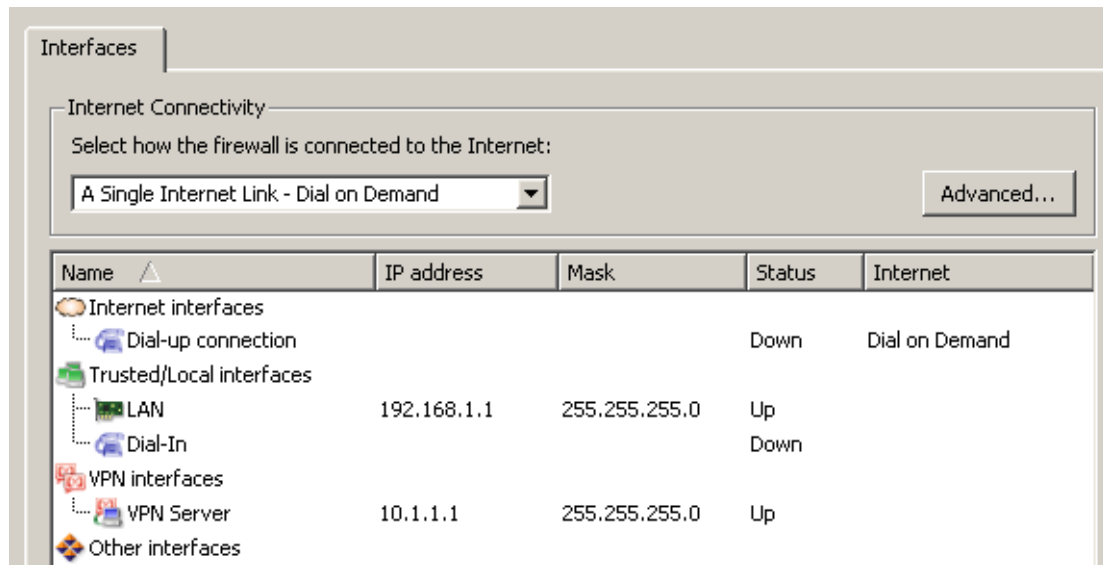


Figure 6.6 Configuration of interfaces — an on-demand dial link

The *Internet interfaces* group can include multiple dial-ups. However, only one of these links can be set for on-demand dialing. If another link is dialed manually, *WinRoute* will route packets to the corresponding destination network in accordance with the system routing table (see also chapter 18.1) and perform IP address translation (NAT). However, such configuration would be of any use. It is therefore recommended to keep only a single on-demand-dial link in the *Internet interfaces* group.

To change the dial-on-demand link, use the corresponding option in the interface edit dialog (see chapter 5) or use the context menu (by right-clicking on the link).

Warning

In the *Dial on Demand* mode, default gateway must NOT be set on any network interface of the firewall! On-demand dialing is based on absence of the default gateway (if no route exist in the [routing table](#) where a [packet](#) would be directed, *WinRoute* create a default gateway by dialing an Internet link).

Dialing options

For dial-ups, the interface settings dialog (see chapter 5) includes also the *Dialing settings* tab where specific parameters for dial-up connections can be set:

Login information

If login data for the particular dial-up connection change, it can be updated here or it is also possible to use the data saved in the operating system (if saved there).

Time intervals for persistent connection and persistent hang-up

Under certain circumstances it may be needed that dial on demand works only within a certain time period (typically in working hours) and that the link is hung-up outside this range. With respect to cost rates of individual providers, it can sometimes be most

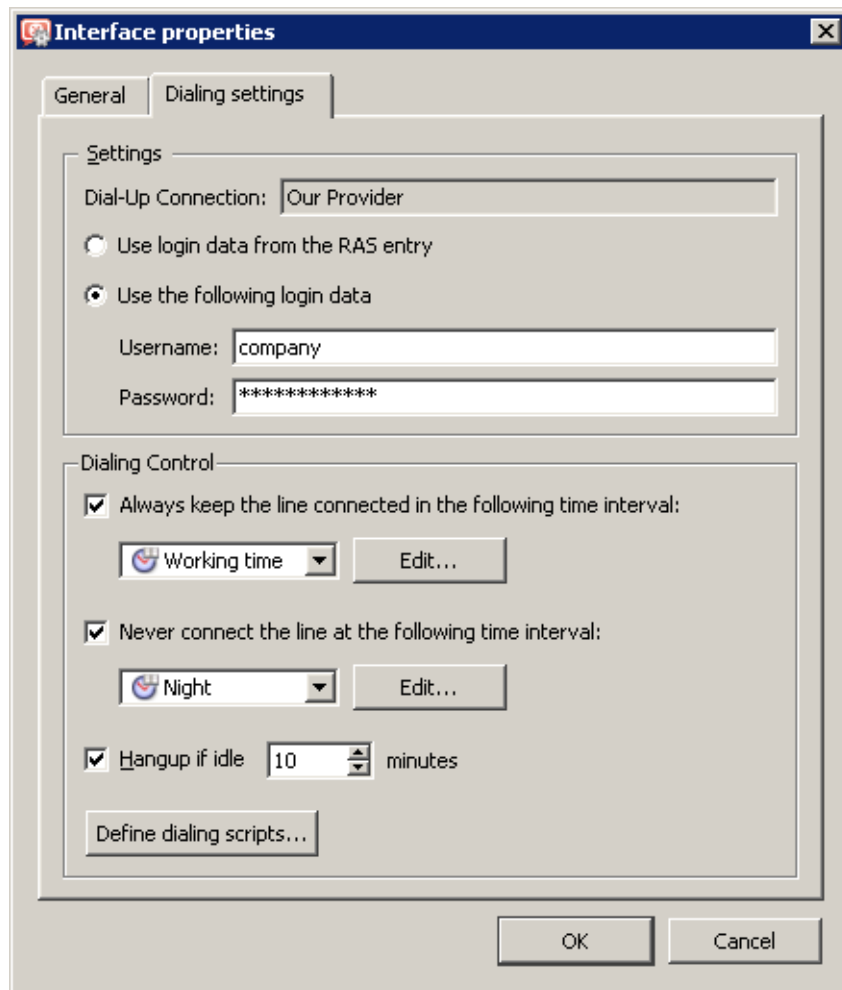


Figure 6.7 Interface properties — dialing settings

efficient to keep the link up persistently even in times with dense network communication.

For these purposes, it is possible to set time intervals for persistent connection and/or hang-up.

If the time intervals overlap, the interval in which the link is hung-up rules over the other. In times outside the defined ranges, the link is dialed on demand.

Note:

1. If a static route over a dial-up is defined in *WinRoute*'s routing table, this link will be dialed whenever a packet is routed through there. Settings for the interval within which the link should be hung-up persistently will be ignored in this case. For details, see chapter [18.1](#).
2. The dialing settings do not include an explicit option of connection recovery upon failures. In case of connection outage, connection will or will not be recovered in dependence on the current mode of the link:
 - If the link should be connected persistently at the moment of the failure, the

connection is recovered automatically.

- If the connection is set to be hung-up at the moment of the outage, the connection will not be recovered.
- In mode of on-demand dial (i.e. outside the intervals defined), connection will be recovered in response to the first request (i.e. packet sent from the local network to the Internet).

Automatic hangup when idle

Dial-ups are usually charged by connection time. When no data are transferred via the connection, there is no reason to keep the link up. Therefore, it is possible to set also idleness time after which the link will be hung-up automatically.

For optimal idleness timeout length, it is necessary to know how the Internet connection is charged in the particular case. If the idleness timeout is too short, it may result in too frequent hanging up and dialing of the link which might be very uncomfortable and in certain cases even increase connection costs.

Note: In the time interval where persistent connection of the link is set (see above), the idleness timeout is ignored.

Dialing scripts

In some cases there is a special need of running a program or a script (execute a batch command) along with dialing or hanging up a link. This can be helpful for example if a special type of modem is used that must be controlled by a special program provided by its developers.

WinRoute allows launching any program or a command in the following situations: *Before dial*, *After dial*, *Before hang-up* or/and *After hang-up*.

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.

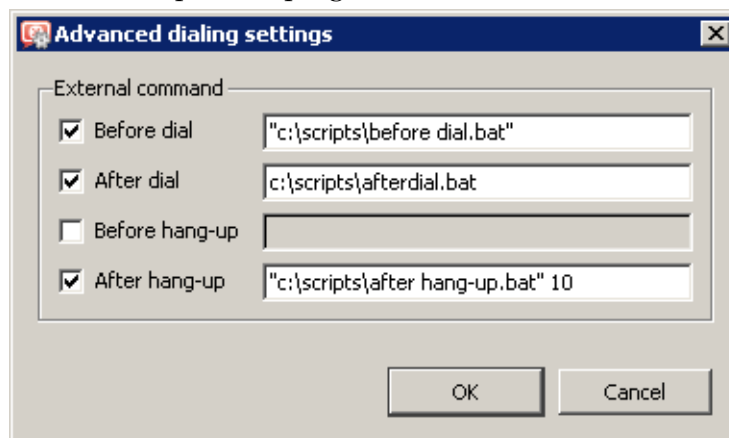


Figure 6.8 Dial-up — external commands

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

WinRoute is running in the operating system as a service. Therefore, external applications and operating system's commands will run in the background only (in the *SYSTEM* account). The same rules are applied for all external commands and external programs called by scripts. Therefore, it is not highly unrecommended to use interactive applications (i.e. applications with user interaction) for the actions described above. Otherwise, interactive applications are running as "invisible" until the next reboot or until the particular process is ended by the *Windows Task Manager*. Under specific circumstances, such application might also block other dials or hang-ups.

6.3 Connection Failover

WinRoute allows guarantee Internet connection by an alternative (back-up) connection (so called connection failover). This connection failover is launched automatically whenever failure of the primary connection is detected. When *WinRoute* finds out that the primary connection is recovered again, the secondary connection is disabled and the primary one is re-established automatically.

Requirements

The computer hosting *WinRoute* must have two network interfaces for Internet connection: a leased line (*Ethernet*, *WiFi*) or a dial-up with persistent connection (*CDMA*, *PPPoE*) for primary connection and a leased line or a dial-up for secondary (failover) connection.

This connection type also requires one or more network cards for connection of individual segments of the LAN. Default gateway must *NOT* be set on any of these cards (cards for the LAN)!

In case of dial-ups, it is also necessary to define corresponding telephone connection in the operating system. It is not necessary that login data for telephone connections are saved in the system, this information can be specified directly in *WinRoute*.

Both the primary and the secondary link may be configured automatically by the DHCP protocol. In that case, *WinRoute* looks all required parameters up in the operating system.

It is recommended to check functionality of both the primary and the secondary link out before installing *WinRoute*:

- If these links are two dial-ups, dial one after the other and check the Internet connection.
- If the primary link is leased and the secondary a dial-up, test the primary link connection first and the secondary connection second. Dialing of the link opens (creates) a new default route via this link which allows us to test Internet connection on the secondary link.
- In case of two leased links, the simplest way is to disable one of the connections in the operating system and test the other (enabled) link. And, as implied, test the other in the same way when the first link is checked.

Warning

Connection failover is relevant only if performed by a persistent connection (i.e. the primary connection uses a network card or a persistently connected dial-up). Failing that, the secondary connection would be activated upon each hang-up of the primary link automatically.

Configuration with the wizard

On the second page of the *Traffic Policy Wizard* (see chapter [7.1](#)), select *Multiple Internet Links — Failover*.

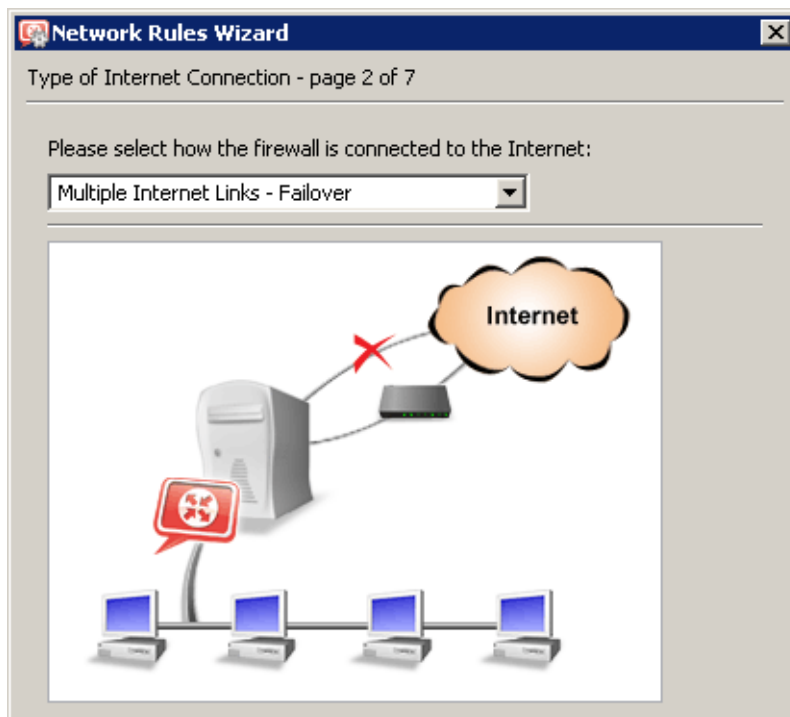


Figure 6.9 Traffic Policy Wizard — Internet connection failover

In the third step of the wizard, select a network interface for the primary connection (leased or persistent dial-up link) and for the secondary connection (leased or dial-up link). If login data for the selected telephone connections are not saved in the operating system, enter the valid username and password.

In the *Software Appliance / VMware Virtual Appliance* edition, the wizard allows:

- to configure parameters of the selected primary and secondary interface,
- to create a new primary or/and secondary interface (*PPPoE*, *PPTP* or dial-up).

For details on network interfaces, see chapter [5](#).

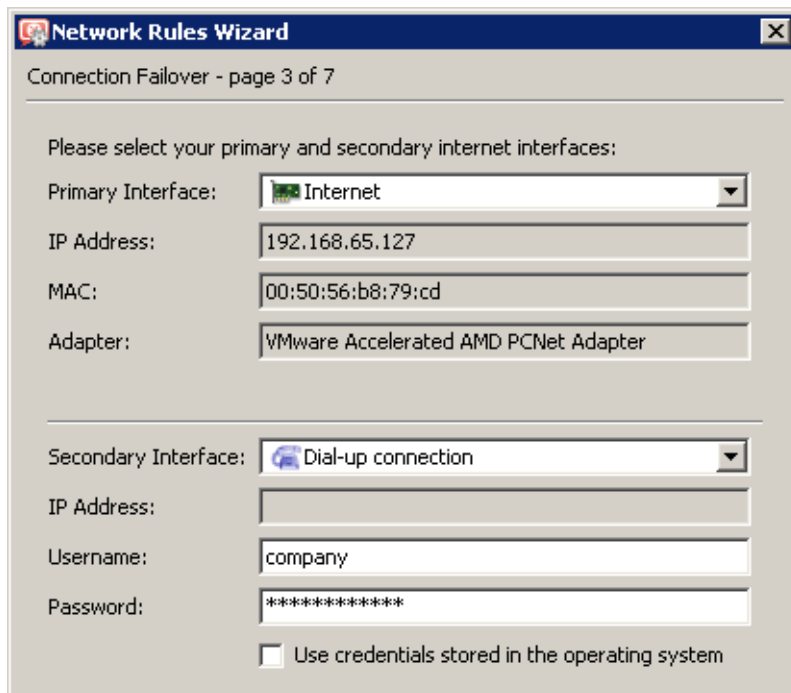


Figure 6.10 Traffic Policy Wizard — failover of a leased link by a dial-up

Resulting interface configuration

When you finish set-up in *Traffic Policy Wizard*, the resulting configuration can be viewed under *Configuration* → *Interfaces* and edited if desirable.

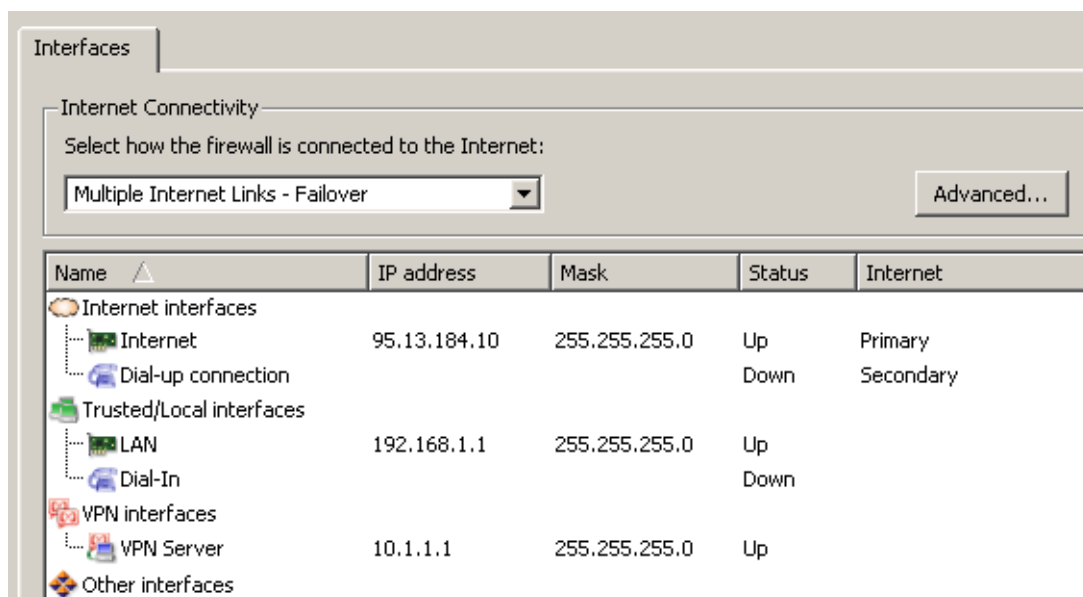


Figure 6.11 Configuration of interfaces — Internet connection failover

The *Internet interfaces* group includes the *Internet* and the *Dial-up* link selected as primary and secondary (failover) on the third page of the wizard. The information provided in the *Internet* column states which link is used for primary and which one for secondary connection. The *Status* column informs of the link status (up/down) as well as of the fact whether the link is active (just being used as Internet connection at the moment) or not.

Other interfaces (including *Dial-In*) are considered as segments of the LAN and put in *Trusted / Local interfaces*.

The *Internet interfaces* group can include also other links. If these links are connected, standard [routing](#) with IP address translation (NAT) will be applied. Obviously, these links will not be backed up by any failover. Such configuration is not of any particular help, anyway. It is recommended to use the *Internet interfaces* for primary and secondary connection links only.

To change settings of primary and secondary connection, use corresponding options in the interface edit dialog (see chapter 5) or use the context menu called up by right-clicking on the corresponding link. However, under any circumstances, always a single link can be set as primary connection and a single one as secondary.

Probe hosts

Functionality of primary Internet connection is regularly tested by sending an *ICMP* request for a response (*PING*) to certain hosts or network interfaces. By default, the default gateway of the primary connection is used as the probe host. If the default gateway is not available, the Internet connection is not working (correctly).

If the primary default gateway cannot be used as the testing computer by any reason, it is possible to specify IP addresses of other (one or more) testing computers upon clicking on *Advanced*. If at least one of the tested devices is available, the primary connection is considered as functioning.

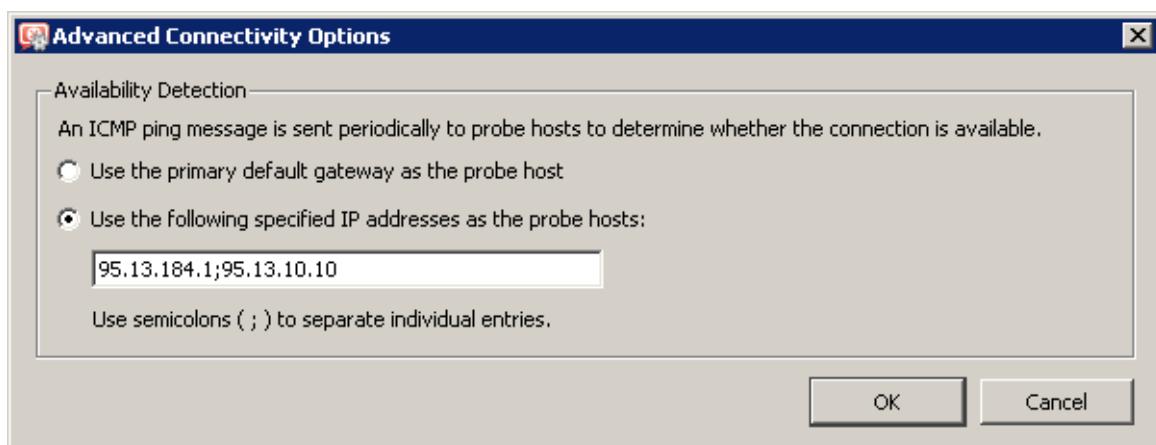


Figure 6.12 Internet connection failover — setting probe hosts

Note:

1. 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. This is one of the cases where the primary default gateway cannot be used as the testing computer.
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. *ICMP* queries sent to probe hosts cannot be blocked by the firewall's traffic rules.

6.4 Network Load Balancing

If at least two Internet links are available, *WinRoute* can divide traffic in parts sent by either of them. The benefits of such solution are evident — Internet connection throughput gets better (i.e. speed of data transmission between the LAN and the Internet increases) and response time gets shorter for connections to servers in the Internet. If special traffic policy is not defined (so called *policy routing* — see chapter 7.5), then individual links are also backed-up mutually (see also chapter 6.3) — in case of failure of one of the lines, the traffic is routed via another.

Note:

1. Network load balancing is applied only to outbound traffic via the default route. If the [routing table](#) (see chapter 18.1) defines a route to a destination network, traffic to the network will always be routed through the particular interface.
2. Network load balancing does not apply to the traffic of the firewall itself. This traffic is processed directly by the operating system and, therefore, the standard [routing](#) is applied here (the default route with the lowest metric value will always be used).

Requirements

The computer hosting *WinRoute* must have two network interfaces for connection to the Internet, i.e. leased (*Ethernet*, *WiFi*) or persistently connected dial-up links (*CDMA*, *PPPoE*). Usual dial-ups (analog modem, *ISDN*) are not suitable, because it is not possible to dial on demand in the network load balancing mode.

This connection type also requires one or more network cards for connection of individual segments of the LAN. Default gateway must *NOT* be set on any of these cards (cards for the LAN)!

In case of dial-ups (*CDMA*, *PPPoE*), it is also necessary to define corresponding telephone connection in the operating system. It is not necessary that login data for telephone connections are saved in the system, this information can be specified directly in *WinRoute*.

Both the primary and the secondary link may be configured automatically by the DHCP protocol. In that case, *WinRoute* looks all required parameters up in the operating system.

It is recommended to check functionality of individual Internet links out before installing *WinRoute*. The following testing methods can be applied (to both links):

- If these links are two dial-ups, connect one after the other and check access to the Internet.
- If one link is leased and the other a dial-up, test the leased link connection first and then dial the other one. Dialing of the link opens (creates) a new default route via this link which allows us to test Internet connection on the secondary link.
- In case of two leased links, the simplest way is to disable one of the connections in the operating system and test the other (enabled) link. And, as implied, test the other in the same way when the first link is checked.

This method can be applied to any number of Internet lines.

Configuration with the wizard

On the second page of the *Traffic Policy Wizard* (see chapter [7.1](#)), select *Multiple Internet Links* — *Traffic Load Balancing*.

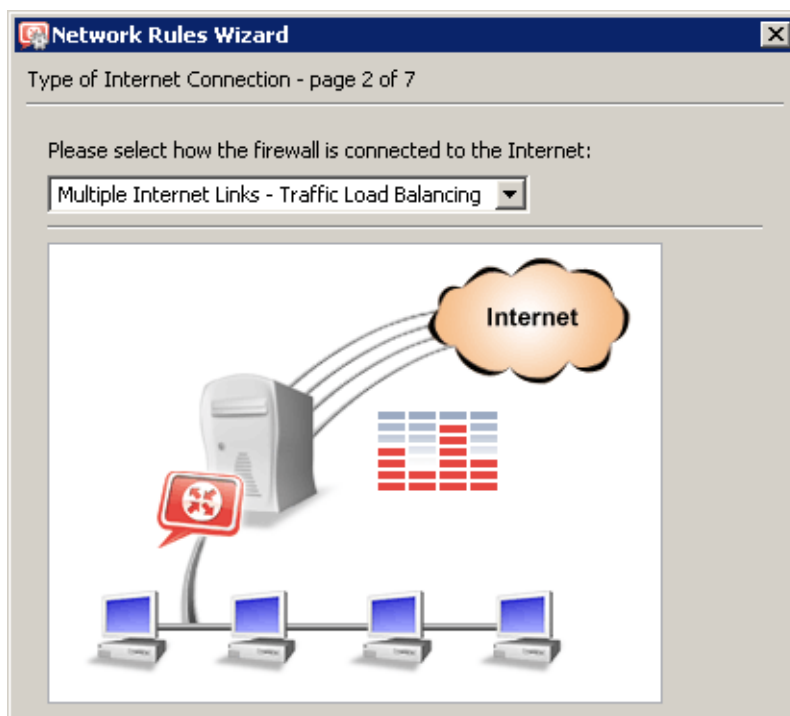


Figure 6.13 Network Policy Wizard — network load balancing

On the third page of the wizard, add all links (one by one) which you intend to use for traffic load balancing.

In the *Software Appliance / VMware Virtual Appliance* edition, the wizard allows:

- to configure parameters of the selected interface,
- to create a new interface (*PPPoE*, *PPTP* or dial-up).

For details on network interfaces, see chapter 5.

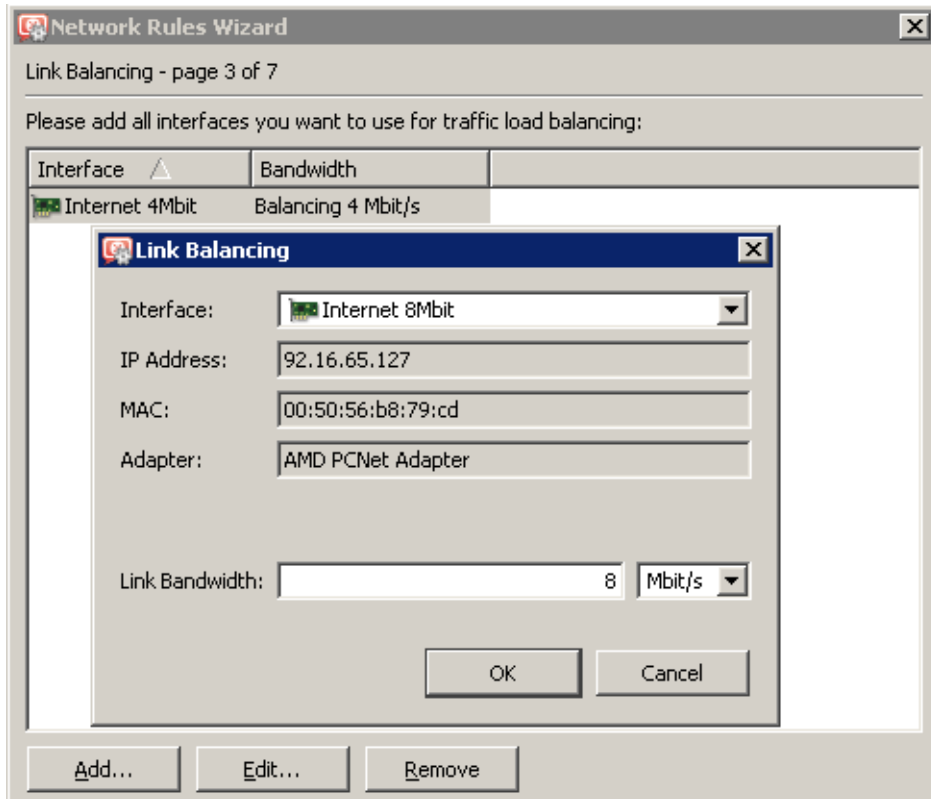


Figure 6.14 Traffic Policy Wizard — failover of a leased link by a dial-up

For each link, specification of bandwidth is required (i.e. traffic speed). The absolute value of the link speed is not important (however, just for reference reasons, it should correspond with the link speed suggested by the ISP). The important aspect is the ratio of speed between individual links — it determines how Internet traffic will be divided among these links.

If login data for the selected telephone connections are not saved in the operating system, valid username and password are required.

Example

Let us suppose there are two Internet links available. You set their bandwidth values to *4 Mbit/s* and *8 Mbit/s*. Total (proposed) speed of the Internet connection is therefore *12 Mbit/s*, while one link provides one third of this capacity and the other link provides two thirds. Simply said, one third of overall Internet traffic will be routed through one link and the resting two thirds through the other one.

Resulting interface configuration

When you finish set-up in *Traffic Policy Wizard*, the resulting configuration can be viewed under *Configuration* → *Interfaces* and edited if desirable.

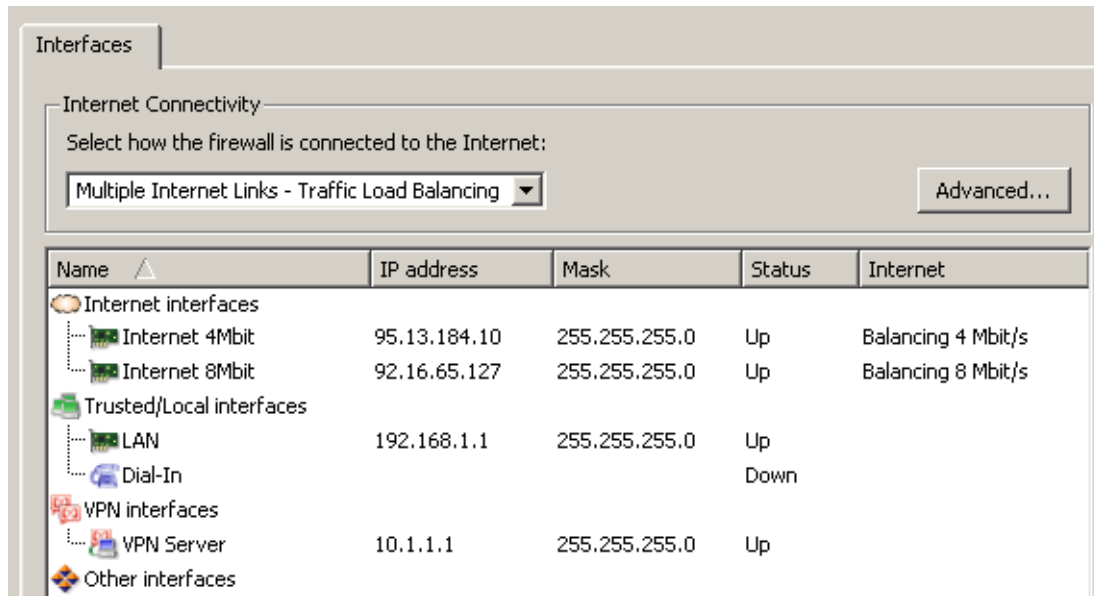


Figure 6.15 Configuration of interfaces — network traffic load balancing

The *Internet interfaces* group includes the *Internet 4Mbit* and the *Internet 8Mbit* link selected as an interface for Internet traffic load balancing on the third page of the wizard.

The *Internet* column shows proposed speed of individual links (see above). The *Status* column informs of the current status of the link (up/down) as well as of the fact whether the link is active, i.e. whether connection on this Internet link is working and part of Internet traffic can be routed through it.

Other interfaces (including *Dial-In*) are considered as segments of the LAN and put in *Trusted / Local interfaces*.

For any new link added to the *Internet interfaces* group, the default speed of *1 Mbit/s* will be set. Then it is possible and also recommended to edit the proposed link speed in the interface settings (see chapter 5) with respect to its real speed, which makes the balancing efficient and working smoothly.

Hint

Speed of one or more links can be set even for *0 Mbit/s*. Such links will then not be used for network traffic load balancing, but for traffic routing in accordance with specific traffic rules (see chapter 7.5). However, availability of these links will still be tested and the links will serve as alternative for case that all the other links fail.

Advanced settings (optimization, dedicated links, etc.)

In basic configuration, network load balancing is applied automatically with respect to their proposed speeds (see above). It is possible to use traffic rules to modify this algorithm (e.g. by dedicating one link for a particular traffic). This issue is described in detail in chapter [7.5](#).

Probe hosts

Functionality of individual Internet links is regularly tested by sending an *ICMP* request for a response (*PING*) to certain hosts or network interfaces. By default, the default gateway of the particular link is used as the probe host. If the default gateway is not available, the tested link is not working (correctly).

If the primary default gateway (i.e. the default gateway set for the tested link) cannot be used as the testing computer by any reason, it is possible to specify IP addresses of other (one or more) testing computers upon clicking on *Advanced*. If at least one of the tested devices is available, the Internet connection in question is considered as functioning.

The specified probe hosts will be used for testing of availability of *all* Internet links. Therefore, the group of testing computers should include a few hosts belonging to various subnets of the Internet.

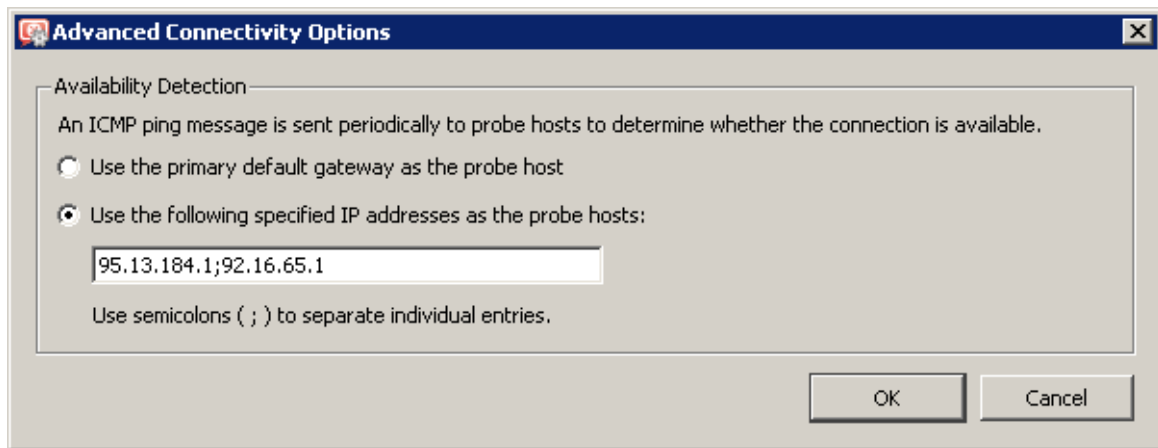


Figure 6.16 Network load balancing — setting probe hosts

Note:

1. 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. This is one of the cases where the default gateway cannot be used as the testing computer.
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. *ICMP* queries sent to probe hosts cannot be blocked by the firewall's traffic rules.

Chapter 7

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).

It is recommended to create basic traffic rules and later customize them as desired. Advanced administrators can create all the rules according to their specific needs without using the wizard.

7.1 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. This connection is not required to be dialed at the moment of the wizard's startup.



Figure 7.1 Traffic Policy Wizard — introduction

Steps 2 and 3— internet connection settings

On the second page of the wizard, select how the LAN will be connected to the Internet with *WinRoute* (leased link, dial-up, leased link with connection failover or multiple links with network traffic load balancing).

On the third page, you can set parameters for the selected type of Internet connection.

Individual options of Internet connection are addressed thoroughly in chapter 6.

Note:

1. Selection of Internet connection type does not affect resulting traffic rules, but only configuration of interfaces and their classification in groups (see chapters 5 and 6).
2. The *Traffic Policy Wizard* no longer includes the option to enable /disable IP address translation (NAT) which was available in older versions of *WinRoute*. In all created traffic rules, NAT is enabled automatically. The reason for this is that modes of network load balancing, connection failover and on-demand dialing cannot actually be used without NAT.

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.

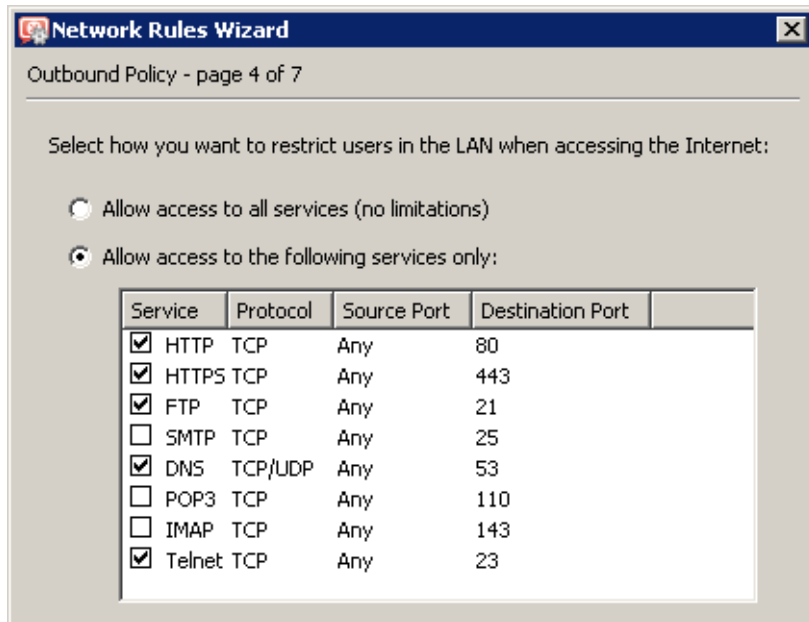


Figure 7.2 Network Policy Wizard — enabling access to Internet services

Allow access to the following services only

Only selected services will be available from the local network.

Note:

1. Defined restrictions will be applied also to the firewall itself.
2. In this dialog, only basic services are listed (it does not depend on what services were defined in *WinRoute* — see chapter 14.3). Other services can be allowed by modification of *NAT* traffic rules (for LAN hosts) or *Firewall* traffic rules (for the firewall) or by adding custom rules. For details, see chapter 7.3.

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, keep the *Create rules for Kerio VPN server* selected. Specific services and address groups for *Kerio VPN* will be added. For detailed information on the proprietary VPN solution, refer to chapter 23.

If you intend not to use the solution or to use a third-party solution (e.g. *Microsoft PPTP*, *Nortel IPSec*, etc.), disable the *Create rules for Kerio VPN* option.

To enable remote access to shared items in the local network via a web browser, keep the *Create rules for Kerio Clientless SSL-VPN* option enabled. This interface is independent from *Kerio VPN* and it can be used along with a third-party VPN solution. For detailed information, see chapter 24.

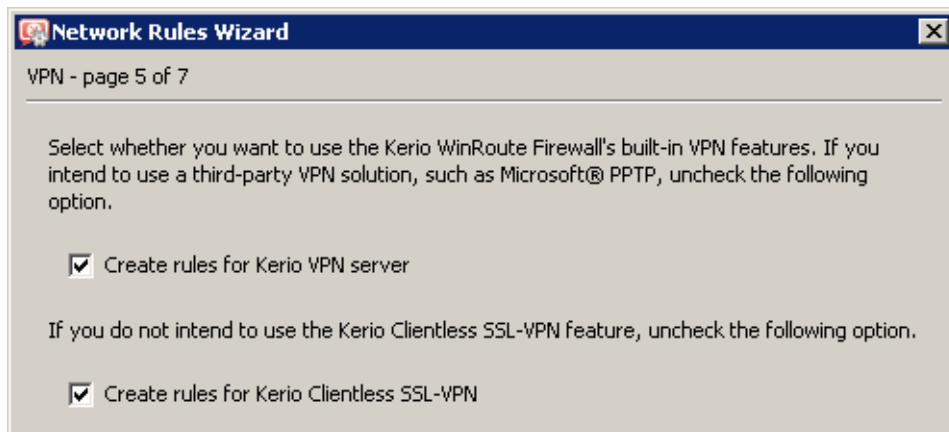


Figure 7.3 Network Policy Wizard — Kerio VPN

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.

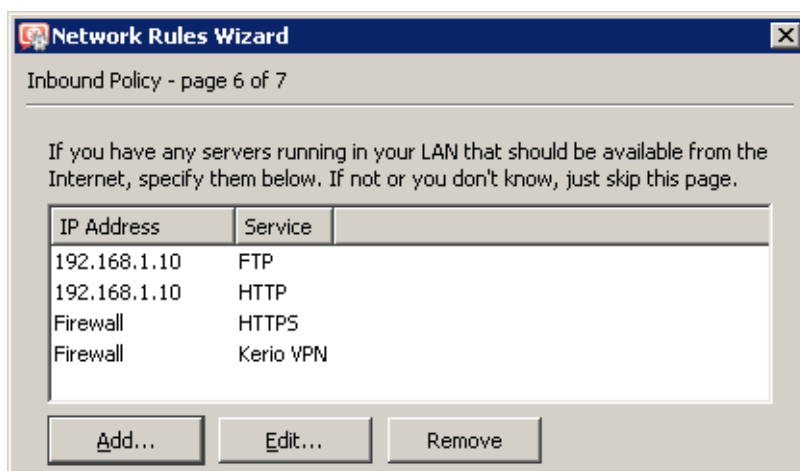


Figure 7.4 Network Policy Wizard — enabling local services

Note: If creating of rules for Kerio VPN was required in the previous step, the *Kerio VPN* and *HTTPS* firewall services will be automatically added to the list of local servers. If these services are removed or their parameters are modified, VPN services will not be available via the Internet!

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

Service is running on

Select a computer where the corresponding service is running (i.e. the host to which traffic coming in from the Internet will be redirected):

- *Firewall* — the host where *WinRoute* is installed
- *Local host with IP address* — another host in the local network (local server)

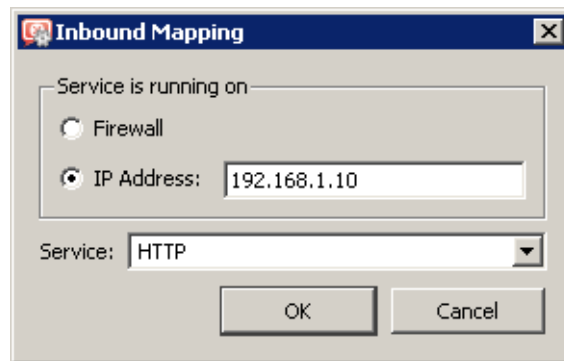


Figure 7.5 Network Policy Wizard — mapping of the local service

Note: Access to the Internet through *WinRoute* must be defined at 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 [14.3](#)). Majority of common services is predefined in *WinRoute*.

Step 7 — generating the rules

In the last step, traffic rules are generated in accordance with data specified. All existing rules will be removed and replaced by the new rules.

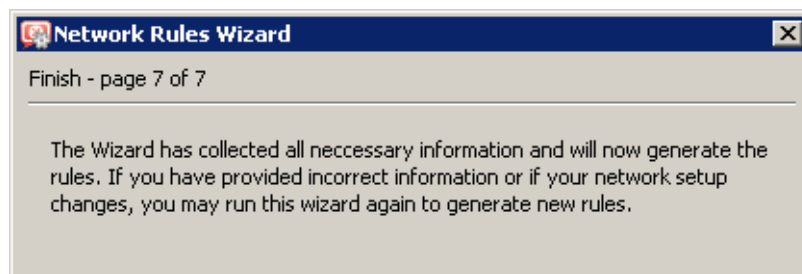


Figure 7.6 Network Rules Wizard — the last step

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.

These rules are not affected by the selected type of Internet connection (the wizard, pages 2 and 3).

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service FTP	Any	Firewall	FTP	✓	MAP 192.168.1.10
<input checked="" type="checkbox"/> Service HTTP	Any	Firewall	HTTP	✓	MAP 192.168.1.10
<input checked="" type="checkbox"/> Service HTTPS	Any	Firewall	HTTPS	✓	
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> NAT	Trusted/Local	Internet	DNS FTP HTTP HTTPS	✓	NAT
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients All VPN tunnels Trusted/Local	Firewall All VPN clients All VPN tunnels Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	DNS FTP HTTP HTTPS	✓	
Default rule	Any	Any	Any	✗	

Figure 7.7 Traffic Policy generated by the wizard

FTP Service and HTTP Service

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 at IP addresses of the “outbound” interface of the firewall (i.e. the interface connected to the Internet — page 3).

Note: Since *WinRoute 6.4.0*, mapped services can be accessed also from local networks — it is therefore not necessary to use another (private) IP address for connections from local clients. Therefore, the *Source* value is set to *Any*. For details, see chapter 7.3.

Kerio VPN Service and HTTPS Service

The *Kerio VPN service* rule enables connection to the *WinRoute*’s VPN server (establishment of control connection between a VPN client and the server or creation of a VPN tunnel — for details, see chapter 23).

The *HTTPS Service* rule allows connection via the *Clientless SSL-VPN* interface (access to shared network items via a web browser — for details, see chapter 24).

These rules are not created unless the option allowing access to a particular service is enabled in step 5.

Note: In these rules, value for *Source* is also set to *Any*. The main reason for this is to keep consistent with rules for mapped services (all these rules are defined in page 6 of the wizard). Access to firewall services from the local network is, under normal conditions, allowed by the *Firewall traffic* rule but this is not always true.

NAT

This rule sets that in all [packets routed](#) from the local network to the Internet, the source (private) IP address will be replaced by the address of the Internet interface through which the [packet](#) is sent from the firewall. Only specified services can be accessed by the Internet connection (the wizard, page 4).

The *Source* item of this rule includes the *Trusted / Local interfaces* group and the *Destination* item includes group *Internet interfaces*. This makes the rule applicable to any network configuration. It is not necessary to change this rule whenever a new segment of the LAN is connected or Internet connection is changed.

By default, the *Trusted / Local interfaces* group includes also a *Dial-In* interface, i.e. all RAS clients connecting to this server can access the Internet with the NAT technology.

Local Traffic

This rule allows all traffic between local hosts and the firewall (i.e. the computer where *WinRoute* is installed). In this rule, items *Source* and *Destination* include the *Trusted / Local interfaces* group (see chapter 5) and the special group *Firewall*.

By default, the *Trusted / Local interfaces* group includes also a *Dial-In* interface. This means that the *Local Traffic* rule also allows traffic between local hosts and RAS clients/VPN clients connected to the server.

If creating of rules for *Kerio VPN* was set in the wizard (the wizard, page 5), the *Local Traffic* rule includes also special address groups *All VPN tunnels* and *All VPN clients*. This implies that, by default, the rule allows traffic between the local network (firewall), remote networks connected via VPN tunnels and VPN clients connecting to the *WinRoute's* VPN 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 between the LAN and 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).

Default rule

This rule drops 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 [7.3](#).

7.2 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 rule is applied that meets [connection](#) or [packet](#) parameters — i.e. order of the rules in the list is key. 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.

Note:

1. Unless any other traffic rules are defined (by hand or using the wizard), all traffic is blocked by a special rule which is set as default.
2. To control user connections to WWW or FTP servers and filter contents, use the special tools available in *WinRoute* for these purposes (see chapter [12](#)) rather than traffic rules.

7.3 Definition of Custom Traffic Rules

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.

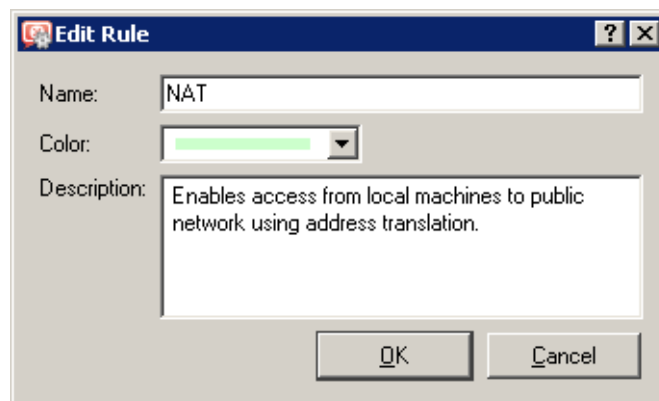


Figure 7.8 Traffic rule — name, color and rule description

The background color of each row with this rule can be defined as well. Use the *Transparent* option to make the background transparent (background color of the whole list will be used, white is usually set). Colors allow highlighting of rules or distinguishing of groups of rules (e.g. rules for incoming and outgoing traffic).

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

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). This is helpful for later reference (at the first glance, it is clear what the rule is used for). *WinRoute* administrators will appreciate this when fine-tuning or trouble-shooting.

Note: Descriptions and background colors of the rules are used for better reference and greater comfort — they do not influence the firewall’s functionality.

Source, Destination

Definition of the source or destination of the traffic defined by the rule.

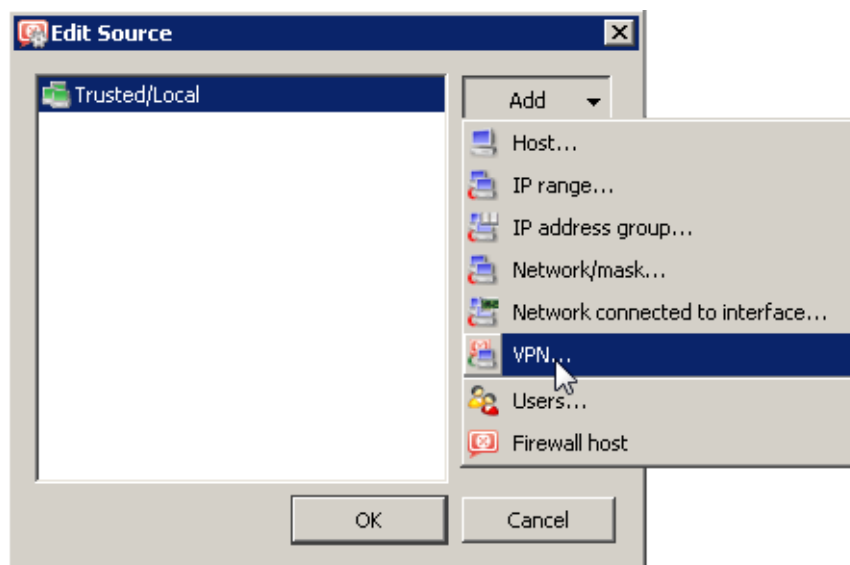


Figure 7.9 Traffic rule — source address definition

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 computers only through IP addresses in case that you are connected to the Internet through a dial-up!

- *IP range* — e.g. 192.168.1.10—192.168.1.20
- *IP address group* — a group of addresses defined in *WinRoute* (refer to chapter 14.1)
- *Subnet with mask* — subnet defined by network address and mask (e.g. 192.168.1.0/255.255.255.0)
- *Network connected to interface* — selection of the interface or a group of interfaces from which the packet comes in (*Source*) or via which they are sent out (*Destination*).

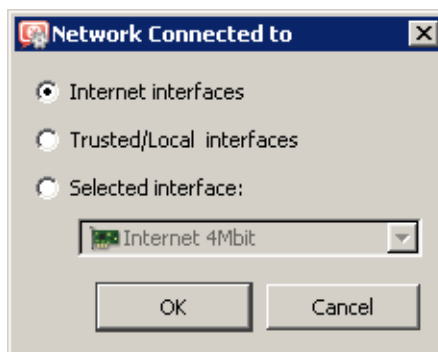


Figure 7.10 Traffic rule — selecting an interface of a group of interfaces

Groups of interfaces allow creation of more general rules independent from any particular network configuration (e.g. it is not necessary to change such rules when Internet connection is changed or when a new LAN segment is added). It is recommended to define traffic rules associated with groups of interfaces wherever possible. For details on network interfaces and groups of interfaces, see chapter 5.

Note: Only the *Internet interfaces* and the *Trusted / Local interfaces* group can be used in traffic rules. Another method is used to add interfaces for *Kerio VPN*(see below). The *Other interfaces* group includes interfaces of various types that were not filed in another group. For this reason, traffic rules for such group would not be of much use.

- *VPN* — virtual private network (created with *Kerio VPN*). This option can be used to add the following items:
 1. *Incoming VPN connections (VPN clients)* — all VPN clients connected to the *WinRoute* VPN server via the *Kerio VPN Client*
 2. *VPN tunnel* — network connected to this server from a remote server via the VPN

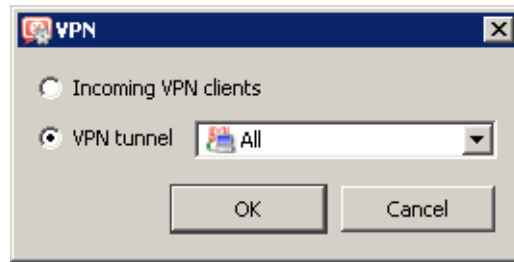


Figure 7.11 Traffic rule — VPN clients / VPN tunnel in the source/destination address definition

tunnel The *All* option covers all networks connected by all VPN tunnels defined which are active at the particular moment.

For detailed information on the proprietary VPN solution integrated in *WinRoute*, refer to chapter [23](#).

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

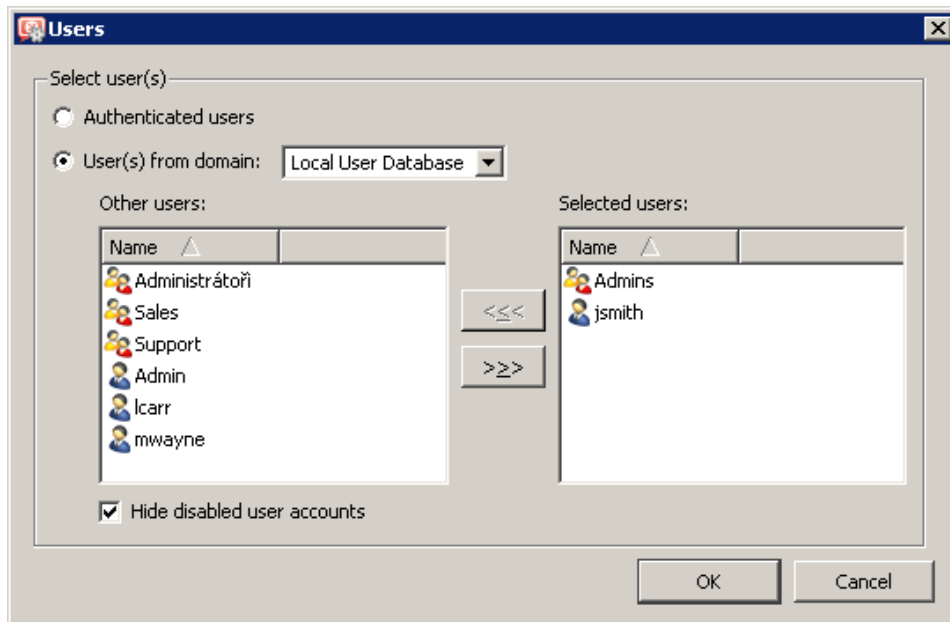


Figure 7.12 Traffic rule — users and groups in the source/destination address definition

The *Authenticated users* option makes the rule valid for all users authenticated to the firewall (see chapter [10.1](#)). Use the *User(s) from domain* option to add users/groups from mapped *Active Directory* domains or from the local user database (for details, refer to chapter [15](#)).

Hint

Users/groups from various domains can be added to a rule at a moment. Select a domain, add users/groups, choose another domain and repeat this process until all demanded users/groups are added.

In traffic rules, user are represented by IP address of the host they are connected (authenticated) from. For detailed description on user authentication, refer to chapter [10.1](#).

Note:

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. If users use each various hosts to connect from, IP addresses of all these hosts must be considered.
2. If user accounts or groups are used as a source in the Internet access rule, automatic redirection to the authentication page nor NTLM authentication will work. Redirection requires successful establishment of connection to the destination server.

If traffic policy is set like this, users must be told to open the authentication page (see chapters [11](#) and [10.1](#)) in their browser and login before they are let into the Internet.

This issue is described in detail in chapter [7.6](#).

- *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.

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.

Use the *Remove* button to remove all items defined (the *Nothing* value will be displayed in the item list). This is helpful when rules are changed — it is not necessary to remove items one by one. Whenever at least one item is added, the *Nothing* value will be removed automatically. If the *Nothing* value is kept for the *Source* or/and *Destination* item, a corresponding rule is disabled.

The *Nothing* value takes effect when network interfaces (see chapter [5](#)) and users or groups (see chapter [15](#)) are removed. The *Nothing* value is automatically used for all *Source*, *Destination* or/and *Service* items of rules where a removed interface (or a user account, a group or a service) has been used. Thus, all these rules are disabled.

Definition of rules with the *Nothing* value in any column is not of any use — it is more useful to use the checkbox in the *Name* column instead to disable a rule.

Note: Removed interfaces cannot be replaced by the *Any* value, otherwise the traffic policy might be changed fundamentally (e.g. an undesirable traffic might be allowed).

Service

Definition of service(s) on which the traffic rule will be applied. Any number of services defined either in *Configurations* → *Definitions* → *Services* (see chapter [14.3](#)) or using protocol and port number (or by port range — a dash is used to specify the range) can be included in the list.

Use the *Any* button to replace all defined items with the *Any* item (this item is also used by default for all new rules). Whenever at least one new service is added, the *Any* value removed automatically.

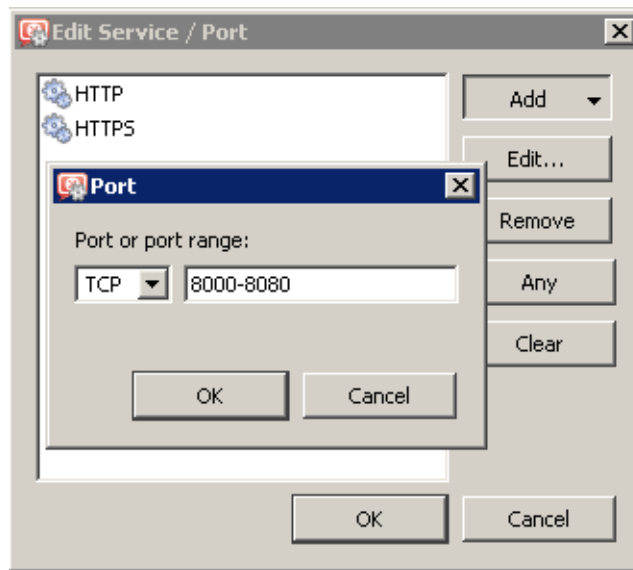


Figure 7.13 Traffic rule — setting a service

Use the *Remove* button to remove all items defined (the *Nothing* value will be displayed in the item list). Whenever at least one service is added, the *Nothing* value will be removed automatically. If the *Nothing* value is kept in the *Service* column, the rule is disabled.

The *Nothing* value is important for removal of services (see chapter 14.3). The *Nothing* value is automatically used for the *Service* item of rules where a removed service has been used. Thus, all these rules are disabled. Inserting the *Nothing* value manually is not meaningful—a checking box in the *Name* column can be used instead.

Note: If there is a protocol inspector for a certain service in *WinRoute*, it is applied to all corresponding traffic automatically. If desired to bypass the protocol inspector for certain traffic, it is necessary to define this exception in the particular traffic rule. For detailed information, see chapter 7.7.

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 (the client 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.

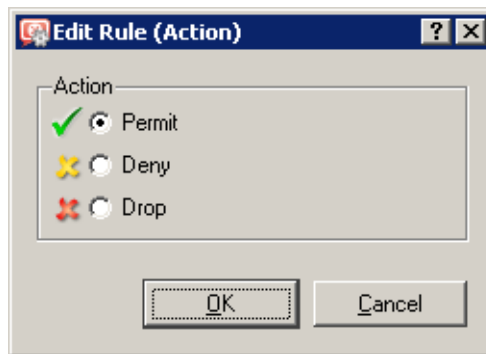


Figure 7.14 Traffic rule — selecting an action

Translation

Source or/and destination IP address translation.

Source IP address translation (NAT — Internet connection sharing)

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 outgoing 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.

Source address translation is used in traffic rules applied to traffic from the local private network to the Internet. In other rules (traffic between the local network and the firewall, between the firewall and the Internet, etc.), NAT is meaningless. For detailed information and examples of rules, refer to chapter [7.4](#).

For source address translation, *WinRoute* offers these options:

Automatic IP address selection

By default, in packets sent from the LAN to the Internet the source IP address will be replaced by IP address of the Internet interface of the firewall through which the packet is sent. This IP address translation method is useful in the general rule for access from the LAN to the Internet (see chapter [7.4](#)), because it works correctly in any Internet connection configuration and for any status of individual links (for details, see chapter [6](#)).

If *WinRoute* works in the mode of network traffic load balancing (see chapter [6.4](#)), you can select a method which will be used for spreading the traffic between the LAN and the Internet over individual Internet links:

- *Load balancing per host* — all traffic from the specific host (client) in the LAN will always be routed via the same Internet link. All connections from the client will be established from the same source IP address (the public address of the particular interface of the firewall). This method is set as default, because it guarantees the same behavior as in case of clients connected directly to the Internet. However,

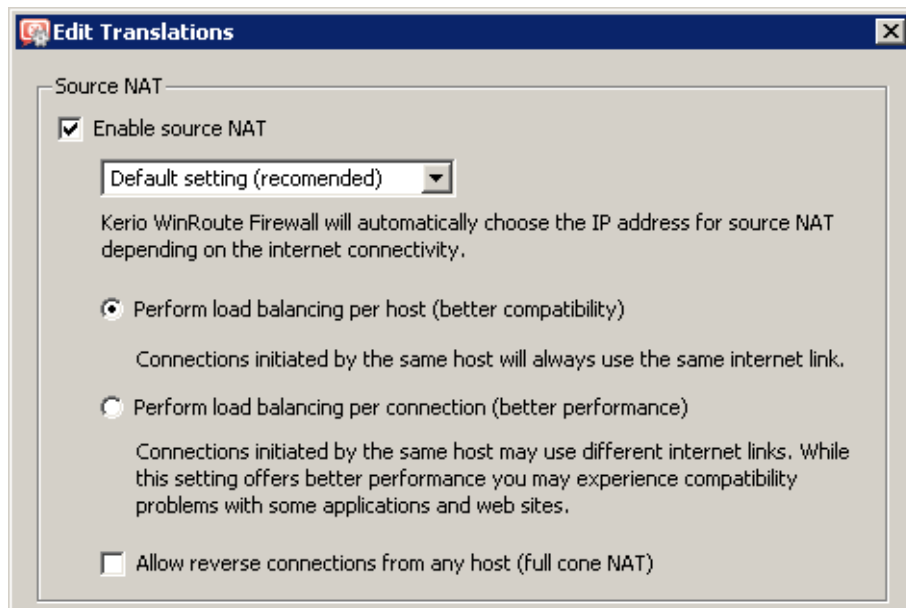


Figure 7.15 Traffic rule — NAT — automatic IP address selection

load balancing dividing the traffic among individual links may be not optimal in this case.

- *Load balancing per connection* — for each [connection](#) established from the LAN to the Internet will be selected an Internet link to spread the load optimally. This method guarantees the most efficient use of the Internet connection's capacity. However, it might also introduce problems and collisions with certain services. The problem is that individual connections are established from various IP addresses (depending on the firewall's interface from which the packet is sent) which may be considered as an attack at the destination server which might result in closing of the session, blocking of the traffic, etc.

If another type of Internet connection is used (a single leased link, on demand dialing or connection failover), these options have no effect on *WinRoute's* functionality.

— **Hint** —

For maximal efficiency of the connection's capacity, it is possible to combine both load balancing methods. In the general rule for access from the LAN to the Internet, use load balancing per connection and add a rule for specific services (servers, clients, etc.) which will employ the load balancing per host method. For details, see also chapter [7.4](#).

NAT to IP address of a specific interface

It is possible to select a specific interface which will be used for the source NAT in outgoing packets. This also determines that packets will be sent to the Internet via this specific link. This allows definition of rules for sending of a specific traffic through a selected — so called *policy routing* — see chapter [7.5](#).

If the selected Internet link fails, Internet will be unavailable for all traffic meeting criteria (specific services, clients, etc.) specified by this rule. To prevent from such situations, it is possible to allow use of an alternative (back-up) interface (link) for cases of the link's

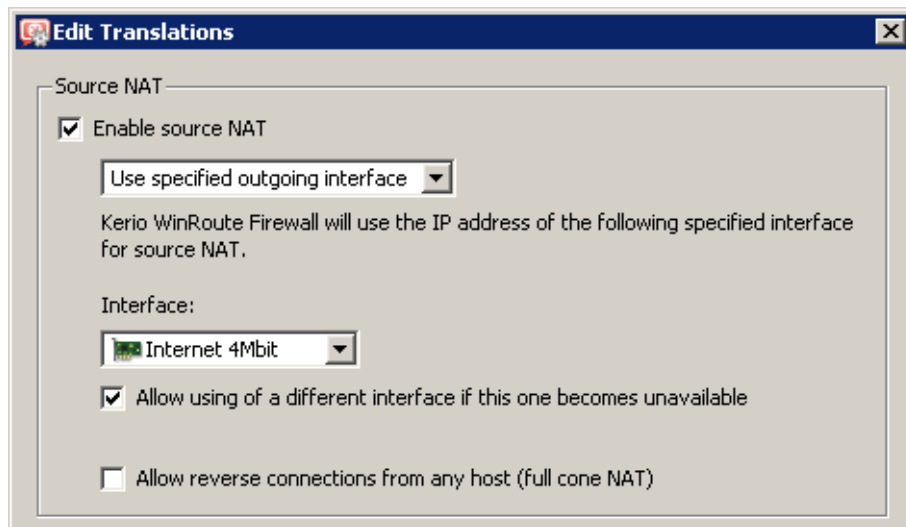


Figure 7.16 Traffic rule — NAT — NAT with specific interface (its IP address)

failure. If set as suggested, *WinRoute* will behave like in mode of automatic interface selection (see above) if the such failure occurs.

NAT with a specified IP address

It is also possible to specify an IP address for NAT which will be used as the source IP address for all packets sent from the LAN to the Internet. This option is available above all to keep the environment compatible with older *WinRoute* versions. However, use of a fixed IP address has many limitations:

- It is necessary to use an IP address of one of the firewall's Internet interfaces. If any other address is used (including even local private addresses). NAT will not work correctly and packets sent to the Internet will be dropped.
- For obvious reasons, specific IP address cannot be used for NAT in the Internet connection failover and the network traffic load balancing modes.

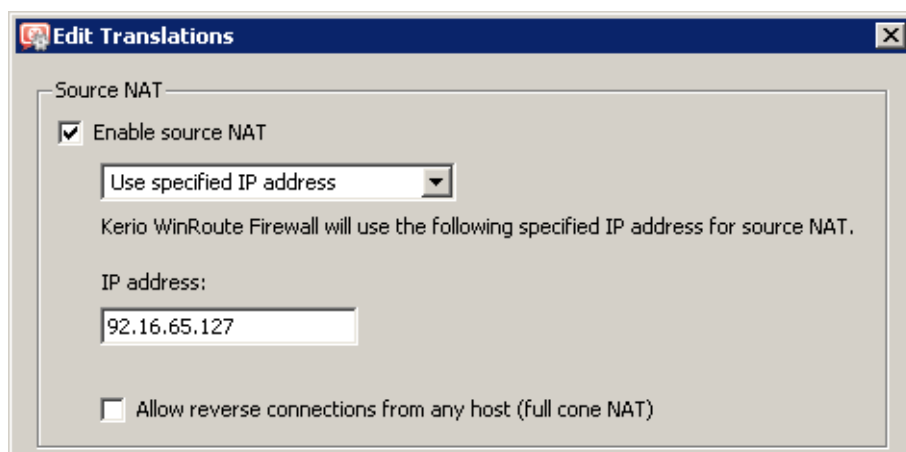


Figure 7.17 Traffic rule — NAT — NAT with specific IP address

Full cone NAT

For all NAT methods it is possible to set mode of allowing of incoming packets coming from any address — so called *Full cone NAT*.

If this option is off, *WinRoute* performs so called *Port restricted cone NAT*. In outgoing packets transferred from the local network to the Internet, *WinRoute* replaces the source IP address of the particular interface by public address of the firewall (see above). If possible, the original source port is kept; otherwise, another free source port is assigned. As to incoming traffic, only packets sent from the same IP address and port from which the outgoing packet was sent are let in. This translation method guarantees high security — the firewall will not let in any packet which is not a response to the sent request.

However, many applications (especially applications working with multimedia, Voice over IP technologies, etc.) use another traffic method where other clients can (with direct connection established) connect to a port “opened” by an outgoing packet. Therefore, *WinRoute* supports also the *Full cone NAT* mode where the described restrictions are not applied for incoming packets. The port then lets in incoming packets with any source IP address and port. This translation method allows running of applications in the private network that would either work only partially or they would not work at all.

For example of using of *Full cone NAT* for VoIP applications, refer to chapter [7.8](#).

— **Warning** —

Use of *Full cone NAT* brings certain security threats — the port opened by outgoing connection can be accessed without any restrictions being applied. For this reason, it is recommended to enable *Full cone NAT* only for a specific service (i.e. to create a special rule for this purpose).

By any means do not allow Full cone NAT in the general rule for traffic from the local network to the Internet⁴! Such rule would significantly decrease security of the local network.

Note:

1. Older versions of *WinRoute* (to version 6.3.1 incl.) used so called *Symmetric NAT* where each outgoing connection on the firewall was assigned a new source port from the reserved range. For this reason, since 6.4.0 *WinRoute* includes significantly improved support for VoIP and multimedia applications than the previous versions even without using special traffic rules. Both methods have the same security level — they differ only in method of assigning source ports on the firewall.
2. The method of IP address translation having been used since version 6.4.0 (i.e. *Port restricted cone NAT*) allows also using of the *IPSec* protocol. Special support for *IPSec* included in older versions of *WinRoute* is not needed any longer.

⁴ Typically the *NAT* rule created by the *Traffic policy wizard* — see chapter [7.1](#).

Destination NAT (port mapping):

Destination address translation (also called port mapping) is used to allow access to services hosted in private local networks behind the firewall. All incoming packets that meet defined rules are re-directed to a defined host (destination address is changed). This actually “moves” to the Internet interface of the *WinRoute* host (i.e. IP address it is mapped from). From the client’s point of view, the service is running on the IP address from which it is mapped (usually on the firewall’s IP address).

Options for destination NAT (port mapping):

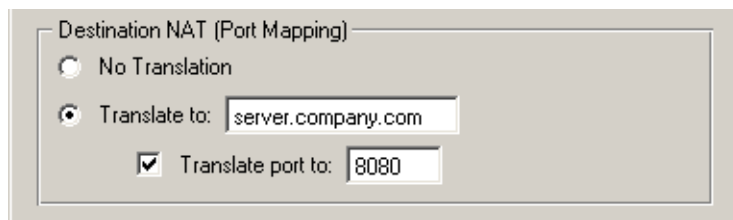


Figure 7.18 Traffic rule — destination address translation

- *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 where it is available from the Internet.

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.

For examples of traffic rules for port mapping and their settings, refer to chapter [7.4](#).

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.

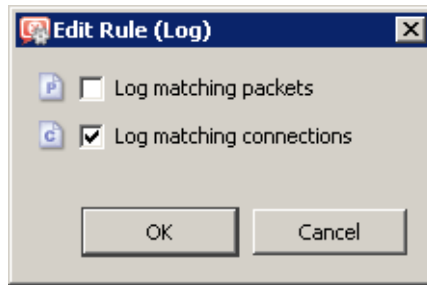


Figure 7.19 Traffic rule — packet/connection logging

Note: Connection cannot be logged for blocking and dropping rules (connection is not even established).

The following columns are hidden in the default settings of the *Traffic Policy* window (for details on showing and hiding columns, see chapter 3.2):

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).

When a denying rule is applied and/or when an allowing rule's appliance terminates, all active network connections matching the particular rule are closed immediately.

Protocol inspector

Selection of a protocol inspector that will be applied on all traffic meeting the rule. The menu provides the following options to select from:

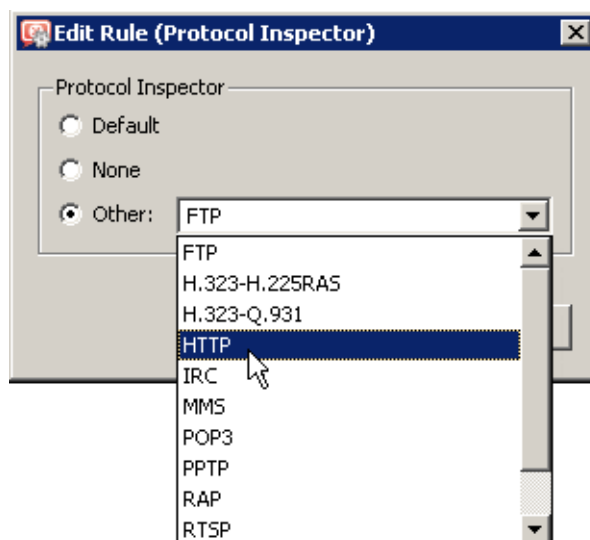


Figure 7.20 Traffic rule — protocol inspector selection

- *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 applied to traffic meeting this rule (all *WinRoute's* protocol inspectors are available). No other protocol inspector will be applied to the traffic, regardless of settings of services in the *Service* section. 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.
For more information, refer to chapter [7.7](#).

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).

7.4 Basic Traffic Rule Types

WinRoute traffic policy provides a range of network traffic filtering options. 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 (NAT)

IP translation (as well as Internet connection sharing) 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. This technology is used to connect local private networks to the Internet by a single public IP address.

The following example shows an appropriate traffic rule:

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT	 Trusted/Local	 Internet	 Any		NAT

Figure 7.21 A typical traffic rule for NAT (Internet connection sharing)

Source

The *Trusted / Local interfaces* group. This group includes all segments of the LAN connected directly to the firewall. If access to the Internet from some segments is supposed to be blocked, the most suitable group to file the interface into is *Other interfaces*. If the local network consists of cascaded segments (i.e. it includes other [routers](#)), it is not necessary to customize the rule in accordance with this fact — it is just necessary to set [routing](#) correctly (see chapter [18.1](#)).

Destination

The *Internet interfaces* group. With this group, the rule is usable for any type of Internet connection (see chapter 6) and it is not necessary to modify it even if Internet connection is changed.

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 *Default settings* option (the primary IP address of the interface via which packets go out from the *WinRoute* host will be used for NAT). This also guarantees versatility of this rule — IP address translation will always be working correctly, regardless the Internet connection type and the particular link type via which the [packet](#) will be sent to the Internet.

Warning

The *No translation* option should be set in the *Destination address translation* section, otherwise the rule might not function. Combining source and destination IP address translation is relevant under special conditions only .

Placing the rule

The rule for destination address translation must be preceded by all rules which deny access to the Internet from the local network.

Note: Such a rule allows access to the Internet from any host in the local network, not from the firewall itself (i.e. from the *WinRoute* host)!

Traffic between the firewall and the Internet must be enabled by a special rule. Since *WinRoute* host can access the Internet directly, it is not necessary to use NAT.






Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Firewall traffic 	 Firewall	 Any	 Any		

Figure 7.22 Rule for traffic between the firewall and hosts in the Internet

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 (public address of the *WinRoute* host).

Since 6.4.0, *WinRoute* allows to access mapped services also from the local network. This avoids problems with different DNS records for the Internet and the local network.

Traffic rule for port mapping can be defined as follows:

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Web server	Any	Firewall	HTTP HTTPS	✓	MAP 192.168.1.10

Figure 7.23 Traffic rule that makes the local web server available from the Internet

Source

Mapped services can be accessed by clients both from the Internet and from the local network. For this reason, it is possible to keep the *Any* value in the *Source* entry (or it is possible to list all relevant interface groups or individual groups — e.g. *Internet* and *LAN*).

Destination

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

This service will be available at all addresses of the interface connected to the Internet. To make the service available at a particular IP address, use the *Host* option and specify the IP address (see the multihoming example).

Service

Services to be available. You can select one of the predefined services (see chapter 14.3) 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 port which is different from the one where the service is available from the Internet.

— **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. Port mapping will not function well unless this condition is met.

Placing the rule

As already mentioned, mapped services can be accessed also from the local network. During access from the local network, connection is established from the local (private) IP address to an IP address in the Internet (the firewall’s public IP address). If the rule for mapped service is preceded by a rule allowing access from the local network to the Internet, according to this rule the packet would be directed to the Internet and then

dropped. Therefore, it is recommended to put all rules for mapped services *at the top* of the table of traffic rules.

Note: If there are separate rules limiting access to mapped services, these rules must precede mapping rules. It is usually possible to combine service mapping and access restriction in a single rule.

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).

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 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
<input checked="" type="checkbox"/> Web server #1 mapping	 Any	 63.157.211.10	 HTTP		MAP 192.168.1.100
<input checked="" type="checkbox"/> Web server #2 mapping	 Any	 63.157.211.11	 HTTP		MAP 192.168.1.200

Figure 7.24 Multihoming — web servers mapping

Source

Any (see the previous example referring to mapping of single service).

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

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

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

Sometimes, it is helpful to limit users access to the Internet services from the local network. 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
<input checked="" type="checkbox"/> NAT	 Trusted/Local	 Internet	 DNS  FTP  FTP  HTTP  HTTPS  Telnet		NAT

Figure 7.25 Internet connection sharing — only selected services are available

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 15.5).

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT for allowed hosts	 Internet access	 Internet	 Any		NAT

Figure 7.26 Only selected IP address group(s) is/are allowed to connect to the Internet

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 accordance with this fact, the traffic is permitted or denied.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT for a group of users	 Internet access	 Internet	 Any		NAT

Figure 7.27 Only selected user group(s) is/are allowed to connect to the Internet

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
<input checked="" type="checkbox"/> NAT for a group of users	 Authenticated users	 Internet	 Any		NAT

Figure 7.28 Only authenticated users are allowed to connect to the Internet

For detailed description on user authentication, refer to chapter [10.1](#).

Note:

1. The rules mentioned above can be combined in various ways (i.e. a user group can be allowed to access certain Internet services only).
2. Usage of user accounts and groups in traffic policy follows specific rules. For detailed description on this topic, refer to chapter [7.6](#).

Exclusions

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	Translation
<input checked="" type="checkbox"/> Allow Telnet for a group of users	 Telnet allowed	 Internet	 Telnet		NAT
<input checked="" type="checkbox"/> Deny Telnet	 Any	 Internet	 Telnet		

Figure 7.29 Exception — Telnet is available only for selected user group(s)

7.5 Policy routing

If the LAN is connected to the Internet by multiple links with load balancing (see chapter [6.4](#)), it may be needed that one link is reserved for a certain traffic, leaving the rest of the load for the other links. Such a measure is useful if it is necessary to keep important traffic swinging (email traffic, the informational system, etc.), i.e. not slowed down by secondary or even

marginal traffic (web browsing, online radio channels, etc.). To meet this crucial requirement of an enterprise data traffic, it is necessary to consider and employ, besides the destination IP address, additional information when [routing packets](#) from the LAN to the Internet, such as source IP address, protocol, etc. This approach is called *policy routing*.

In *WinRoute*, policy routing can be defined by conditions in traffic rules for Internet access with IP address translation (NAT). This approach brings wide range of options helping to meet all requirements for routing and network load balancing.

Note: Policy routing traffic rules are of higher priority than routes defined in the [routing table](#) (see chapter [18.1](#)).

Example: A link reserved for email traffic

Let us suppose that the firewall is connected to the Internet by two links with load balancing with speed values of 4 Mbit/s and 8 Mbit/s. One of the links is connected to the provider where the mailservers are also hosted. Therefore, it is desirable that all email traffic (*SMTP, IMAP, POP3* protocols and their secured versions) is routed through this link.

Define the following traffic rules to meet these requirements:

- First rule defines that NAT is applied to email services and the *Internet 4 Mbit* interface is used.
- The other rule is a general NAT rule with automatic interface selection (see chapter [7.4](#)).

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT - Dedicated link for email	Trusted/Local	Internet	IMAP IMAPS POP3 POP3S SMTP SMTPS	✓	NAT (Internet 4Mbit)
<input checked="" type="checkbox"/> NAT	Trusted/Local	Internet	Any	✓	NAT

Figure 7.30 Policy routing — a link reserved for email traffic

Setting of NAT in the rule for email services is shown in figure [7.31](#). It is recommended to allow use of a back-up link for case that the reserved link fails. Otherwise, email services will be unavailable when the connection fails.

Let us suppose that the mailservers provide also *Webmail* and *CalDAV* services which use *HTTP(s)* protocol. Adding these protocols in the first rule would make all web traffic routed through the reserved link. To reach the desired goal, the rule can be modified by reserving the link for traffic with a specific server — see figure [7.32](#).

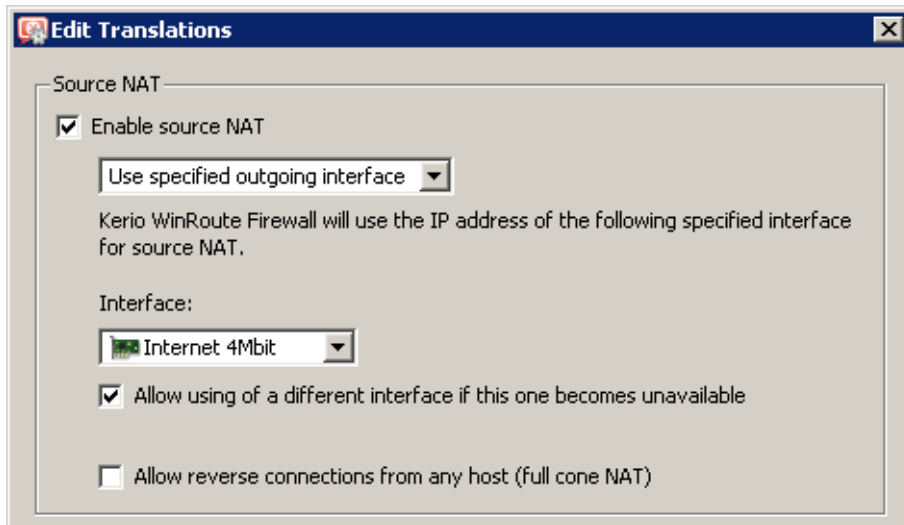


Figure 7.31 Policy routing — setting NAT for a reserved link

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT - Dedicated link for email	Trusted/Local	mail.server.com	Any	✓	NAT (Internet 4Mbit)
<input checked="" type="checkbox"/> NAT	Trusted/Local	Internet	Any	✓	NAT

Figure 7.32 Policy routing — a link reserved for a specific server

Note: In the second rule, automatic interface selection is used. This means that the *Internet 4Mbit* link is also used for network traffic load balancing. Email traffic is certainly still respected and has higher priority on the link reserved by the first rule. This means that total load will be efficiently balanced between both links all the time.

If you need to reserve a link *only* for a specific traffic (i.e. route other traffic through other links), go to *Configuration* → *Interfaces* and set the speed of the link to *0 Mbit/s*. In this case the link will not be used for load balancing. Only traffic specified in corresponding traffic rules will be routed through it.

Example: Optimization of network traffic load balancing

WinRoute provides two options of network traffic load balancing: per host (clients) or per connection (for details, refer to chapter 7.3). With respect to variability of applications on individual hosts and of user behavior, the best solution (more efficient use of individual links) proves to be the option of load balancing per connection. However, this mode may encounter problems with access to services where multiple connections get established at one moment (web pages and other web related services). The server can consider source addresses in individual connections as connection recovery after failure (this may lead for instance to expiration of the session) or as an attack attempt (in that case the service can get unavailable).

This problem can be bridged over by policy routing. In case of “problematic” services (e.g. *HTTP* and *HTTPS*) the load will be balanced per host, i.e. all connections from one client will be routed through a particular Internet link so that their IP address will be identical (a single

IP address will be used). To any other services, load balancing per connection will be applied — thus maximally efficient use of the capacity of available links will be reached.

Meeting of the requirements will be guaranteed by using two NAT traffic rules — see figure 7.33. In the first rule, specify corresponding services and set the *per host* NAT mode. In the second rule, which will be applied for any other services, set the *per connection* NAT mode.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT - Balancing per host	Trusted/Local	Internet	HTTP HTTPS	✓	NAT
<input checked="" type="checkbox"/> NAT - Balancing per connection	Trusted/Local	Internet	Any	✓	NAT Balancing Per Connection

Figure 7.33 Policy routing — load balancing optimization

7.6 User accounts and groups in traffic rules

In traffic rules, source/destination can be specified also by user accounts or/and user groups. In traffic policy, each user account represents IP address of the host from which user is connected. This means that the rule is applied to users authenticated at the firewall only (when the user logs out, the rule is not effective any longer). This chapter is focused on various issues relating to use of user accounts in traffic rules as well as hints for their solution.

Note: For detailed information on traffic rules definition, refer to chapter 7.3.

How to enable certain users to access the Internet

How to enable access to the Internet for specific users only? Assuming that this problem applies to a private local network and Internet connection is performed through NAT, simply specify these users in the *Source* item in the NAT rule.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> NAT	jsmith lcarr mwayne	Internet	Any	✓	NAT

Figure 7.34 This traffic rule allows only selected users to connect to the Internet

Such a rule enables the specified users to connect to the Internet (if authenticated). However, these users must open the *WinRoute* interface’s login page manually and authenticate (for details, see chapter 10.1).

However, with such a rule defined, all methods of automatic authentication will be ineffective (i.e. redirecting to the login page, NTLM authentication as well as automatic authentication from defined hosts). The reason is that the automatic authentication (or redirection to the login page) is not invoked unless connection to the Internet is being established (for license

counting reasons — see chapter 4.6). However, this NAT rule blocks any connection unless the user is authenticated.

Enabling automatic authentication

The automatic user authentication issue can be solved easily as follows:

- Add a rule allowing an unlimited access to the *HTTP* service before the NAT rule.

Name	Source	Destination	Service	Action	Translation	Protocol Inspector
<input checked="" type="checkbox"/> WWW without authentication	Trusted/Local	Internet	HTTP	✓	NAT	Default
<input checked="" type="checkbox"/> NAT	jsmith lcarr mwayne	Internet	Any	✓	NAT	Default

Figure 7.35 These traffic rules enable automatic redirection to the login page

- In URL rules (see chapter 12.2), allow specific users to access any Web site and deny any access to other users.

HTTP Policy

URL Rules | Content Rules | Cache | Proxy Server | Forbidden Words | ISS OrangeWeb Filter

Description	Action	Users List	Condition
<input checked="" type="checkbox"/> Allow access to selected users	✓ Permit	jsmith,lcarr,mwayne	all objects
<input checked="" type="checkbox"/> Deny access to all users	✗ Deny		all objects

Figure 7.36 These URL rules enable specified users to access any Web site

User not authenticated yet who attempts to open a Web site will be automatically redirected to the authentication page (or authenticated by NTLM, or logged in from the corresponding host). After a successful authentication, users specified in the *NAT* rule (see figure 7.35) will be allowed to access also other Internet services. As well as users not specified in the rules, unauthenticated users will be disallowed to access any Web site or/and other Internet services.

Note: In this example, it is assumed that client hosts use the *WinRoute DNS Forwarder* or local DNS server (traffic must be allowed for the DNS server). If client stations used a DNS server in the Internet (this configuration is not recommended!), it would be necessary to include the *DNS* service in the rule which allows unlimited Internet access.

7.7 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, define corresponding source and destination IP addresses and a traffic rule for this service that will define explicitly that no protocol inspector will be used.

Example

A banking application (client) communicates with the bank’s server through its proper protocol which uses TCP protocol at the port 2000. 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.

This port is used by the Cisco SCCP protocol. The protocol inspector of the SCCP would be applied to the traffic of the banking client under normal circumstances. However, this might affect functionality of the application or endanger its security.

A special traffic rule, as follows, will be defined for all traffic of the banking application:

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 used by this communication.

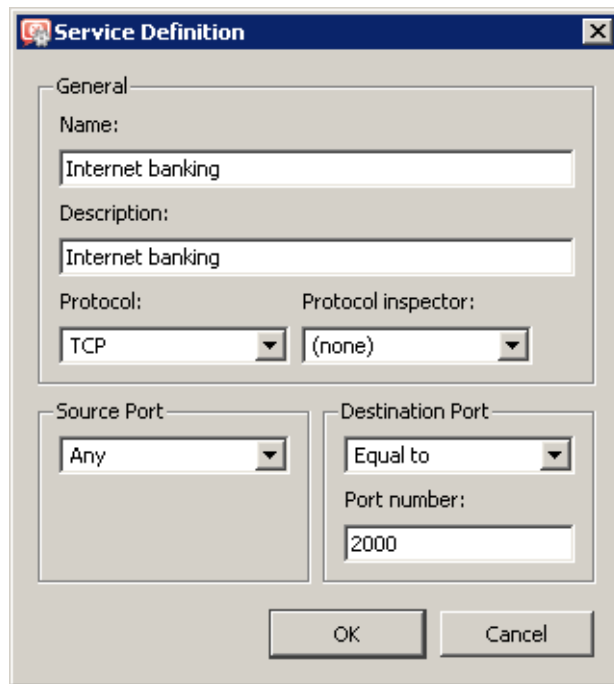


Figure 7.37 Service definition without inspector protocol

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. Specify that no protocol inspector will be applied.

Name	Source	Destination	Service	Action	Protocol Inspector
<input checked="" type="checkbox"/> Internet banking	192.168.1.15	server.bank.com	Internet banking	✓	None

Figure 7.38 This traffic rule allows accessing service without protocol inspection

Note: In the default configuration of the *Traffic rules* section, the *Protocol inspector* column is hidden. To show it, modify settings through the *Modify columns* dialog (see chapter [3.2](#)).

— **Warning** —

To disable a protocol inspector, it is not sufficient to define a service that would not use the inspector! Protocol inspectors are applied to all traffic performed by corresponding protocols by default. To disable a protocol inspector, special traffic rules must be defined.

7.8 Use of Full cone NAT

However, many applications (especially applications working with multimedia, Voice over IP technologies, etc.) use another traffic method where other clients can (with direct connection established) connect to a port “opened” by an outgoing packet. For these cases, *WinRoute* includes a special mode of address translation, known as *Full cone NAT*. In this mode, opened port can be accessed from any IP address and the traffic is always redirected to a corresponding client in the local network.

Use of *Full cone NAT* may bring certain security risk. Each connection established in this mode opens a possible passage from the Internet to the local network. To keep the security as high as possible, it is therefore necessary to enable *Full cone NAT* for particular clients and services only. The following example refers to an IP telephone with the SIP protocol.

Note: For details on traffic rules definition, refer to chapter [7.3](#).

Example: SIP telephone in local network

In the local network, there is an IP telephone registered to an SIP server in the Internet. The parameters may be as follows:

- IP address of the phone: 192.168.1.100
- Public IP address of the firewall: 195.192.33.1
- SIP server: sip.server.com

Since the firewall performs IP address translation, the telephone is registered on the SIP server with the firewall’s public address (195.192.33.1). If there is a call from another telephone to this telephone, the connection will go through the firewall’s address (195.192.33.1) and the corresponding port. Under normal conditions, such connection can be established only directly from the SIP server (to which the original outgoing connection for the registration was established). However, use of *Full cone NAT* allows such connection for any client calling to the SIP telephone in the local network.

Full cone NAT will be enabled by an extremely restrictive traffic rule (to keep the security level as high as possible):

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Full Cone NAT	192.168.1.100	sip.server.com	SIP	✓	Full Cone NAT

Figure 7.39 Definition of a Full cone NAT traffic rule

- *Source* — IP address of an SIP telephone in the local network.
- *Destination* — name or IP address of an SIP server in the Internet. *Full cone NAT* will apply only to connection with this server.
- *Service* — *SIP* service (for an SIP telephone). *Full cone NAT* will not apply to any other services.
- *Action* — traffic must be allowed.
- *Translation* — select a source NAT method (see chapter 7.3) and enable the *Allow returning packets from any host (Full cone NAT)* option.

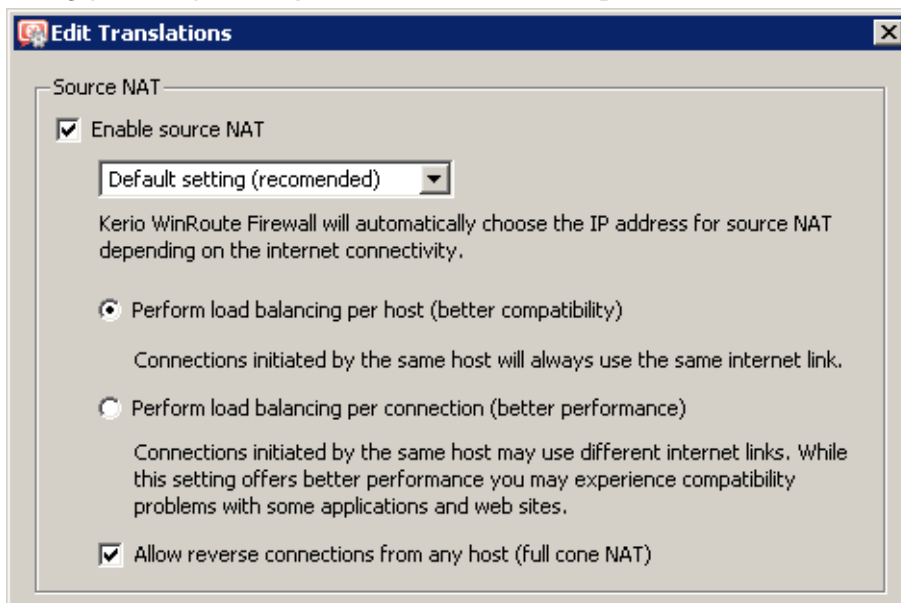


Figure 7.40 Enabling Full cone NAT in the traffic rule

Rule for *Full cone NAT* must precede the general rule with NAT allowing traffic from the local network to the Internet.

7.9 Media hairpinning

WinRoute allows to “arrange” traffic between two clients in the LAN which “know each other” only from behind the firewall’s public IP address. This feature of the firewall is called *hairpinning* (with the *hairpin* root suggesting the packet’s “U-turn” back to the local network). Used especially for transmission of voice or visual data, it is also known as *media hairpinning*.

Example: Two SIP telephones in the LAN

Let us suppose two SIP telephones are located in the LAN. These telephones authenticate at a SIP server in the Internet. The parameters may be as follows:

- IP addresses of the phones: 192.168.1.100 and 192.168.1.101
- Public IP address of the firewall: 195.192.33.1
- SIP server: sip.server.com

For the telephones, define corresponding traffic rules — see chapter 7.8 (as apparent from figure 7.39, simply specify *Source* of the *Full cone NAT* traffic rule by IP address of the other telephone).

Both telephones will be registered on SIP server under the firewall's public IP address (195.192.33.1). If these telephones establish mutual connection, data packets (for voice transmission) from both telephones will be sent to the firewall's public IP address (and to the port of the other telephone). Under normal conditions, such packets would be dropped. However, *WinRoute* is capable of using a corresponding record in the NAT table to recognize that a packet is addressed to a client in the local network. Then it translates the destination IP address and sends the packet back to the local network (as well as in case of port mapping). This ensures that traffic between the two phones will work correctly.

Note:

1. Hairpinning requires traffic between the local network and the Internet being allowed (before processed by the firewall, packets use a local source address and an Internet destination address — i.e. this is an outgoing traffic from the local network to the Internet). In default traffic rules created by the wizard (see chapter 7.1), this condition is met by the *NAT* rule.
2. In principle, hairpinning does not require that *Full cone NAT* is allowed (see chapter 7.8). However, in our example, *Full cone NAT* is required for correct functioning of the *SIP* protocol.

Chapter 8

Configuration of network services

This chapter provides guidelines for setting of basic services in *WinRoute* helpful for easy configuration and smooth access to the Internet:

- *DNS* module — this service is used as a simple DNS server for the LAN,
- *DHCP server* — provides fully automated configuration of LAN hosts,
- *DDNS* client — provides automatic update of firewall logs in public dynamic DNS,
- *Proxy server* — enables access to the Internet for clients which cannot or do not want to use the option of direct access,
- *HTTP cache* — this service accelerates access to repeatedly visited web pages (for direct connections with proxy server).

8.1 DNS module

In *WinRoute*, the *DNS Forwarder* module 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. All requests from each computer in the local network will be sent to the Internet.
- 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 the *DNS* module in *WinRoute*. It can be also used as a basic DNS server for the local domain or/and as a forwarder for the existing server.

If possible, it is recommended to use the *DNS* module as a primary DNS server for LAN hosts (the last option). The *DNS* module provides fast processing of DNS requests and their correct routing in more complex network configurations. The *DNS* module can answer directly to repeated requests and to requests for local DNS names, without the need of contacting DNS servers in the Internet.

If the *DNS* module cannot answer any DNS request on its own, it can forward it to a DNS server set for the Internet link through which the request is sent. For details addressing configuration of the firewall's network interfaces, see chapter [5](#), more information on Internet connection options, refer to chapter [6](#).

The DNS module configuration

By default, DNS server (the *DNS forwarder* service), cache (for faster responses to repeated requests) and simple DNS names resolver are enabled in *WinRoute*.

The configuration can be fine-tuned in *Configuration* → *DNS*.

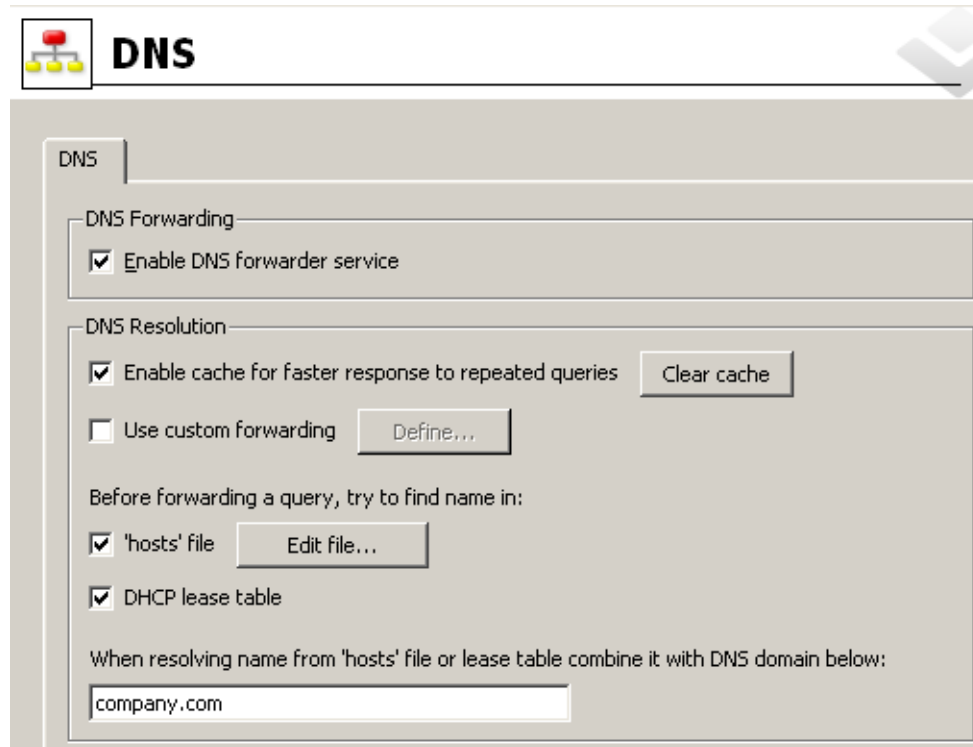


Figure 8.1 DNS settings

Enable DNS forwarder

This option enables DNS server in *WinRoute*. Without other configuration, any DNS requests are forwarded to DNS servers on the corresponding Internet interface.

If the *DNS forwarder* service is disabled, the *DNS* module is used only as a *WinRoute*'s DNS resolver.

Warning

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 disabled, otherwise collision might occur at the DNS service's port (53/UDP).

Enable cache for faster response of repeated queries

If this option is on, all responses will be stored in local *DNS* 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 25.2). This means that records in DNS cache are kept even after *WinRoute Firewall Engine* is stopped or the firewall is closed.

Note:

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 *WinRoute*'s non-transparent proxy server (see chapter [8.4](#)).

Clear cache

Clear-out of all records from the *DNS* cache (regardless of their lifetime). This feature can be helpful e.g. for configuration changes, dial-up testing, error detection, etc.

Use custom forwarding

Use this option to enable settings for forwarding certain DNS queries to other DNS servers (see below).

Simple DNS resolution

The *DNS* module can answer some DNS requests on its own, typically requests regarding local host names. In local network, no other DNS server is required, neither it is necessary to save information about local hosts in the public DNS. For hosts configured automatically by the DHCP protocol (see chapter [8.2](#)), the response will always include the current IP address.

Before forwarding a query...

These options allow setting of where the *DNS* module would search for the name or IP address before the query is forwarded to another DNS server.

- *'hosts' 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, the *DNS* module follows the same rule. Use the *Edit* button to open a special editor where the *hosts* file can be edited within the *Administration Console* even if this console is connected to *WinRoute* remotely (from another host).

- *DHCP lease table*— if the hosts within local network are configured by the DHCP server in *WinRoute* (see chapter [8.2](#)), 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.

The *DNS* module 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. Actually, this is a method of dynamical DNS update.

Note: If both options are disabled, the *DNS* module forwards all queries to other DNS servers.

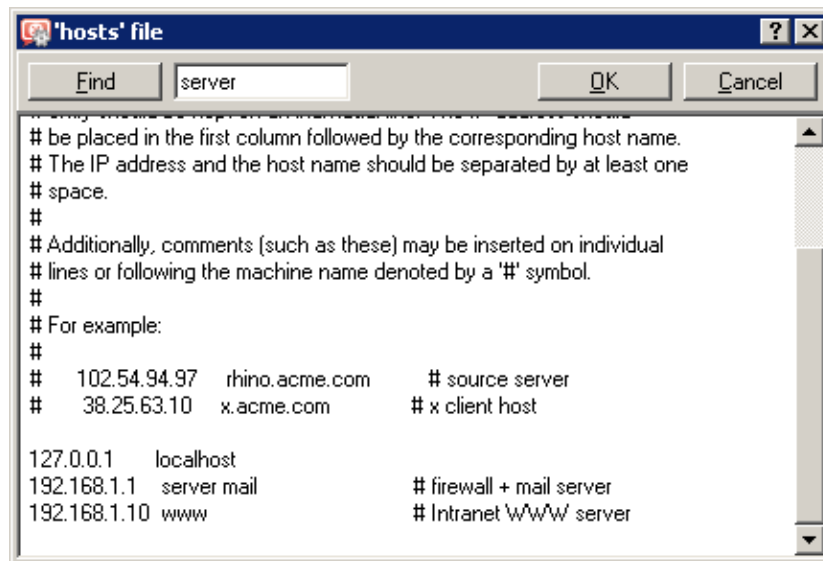


Figure 8.2 Editor of the Hosts system file

Local DNS domain

In the *When resolving name from the 'hosts' file or lease table combine it with DNS domain below* entry, specify name of the local DNS domain.

If a host or a network device sends a request for an IP address, it uses the name only (it has not found out the domain yet). Therefore, only host names without domain are saved in the table of addresses leased by DHCP server. The DNS module needs to know the name of the local domain to answer queries on fully qualified local DNS names (names including the domain).

Note: If the local domain is specified in the DNS module, local names with or without the domain can be recorded in the hosts system file.

The problem can be better understood through the following example.

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 assigns the host IP address `192.168.1.56`. The DHCP server then keeps the information that the IP address is assigned 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 the DNS module, the forwarder would pass the query to another DNS server as it would not recognize that it is a local host. 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.

Enable DNS forwarding

The *DNS* module allows forwarding of certain DNS requests to specific DNS servers. This feature 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). DNS forwarder's settings also play role in configuration of private networks where it is necessary to provide correct forwarding of requests for names in domains of remote subnets (for details, check chapter 23).

Request forwarding is defined by rules for DNS names or subnets. Rules are ordered in a list which is processed from the top. If a DNS name or a subnet in a request matches a rule, the request is forwarded to the corresponding DNS server. Queries which do not match any rule are forwarded to the “default” DNS servers (see above).

Note: If *Simple DNS resolution* is enabled (see below), the forwarding rules are applied only if the *DNS* module is not able to respond by using the information in the *hosts* system file and/or by the DHCP lease table.

Clicking on the *Define* button in the *DNS* module configuration (see figure 8.1) opens a dialog for setting of rules concerning forwarding of DNS queries.

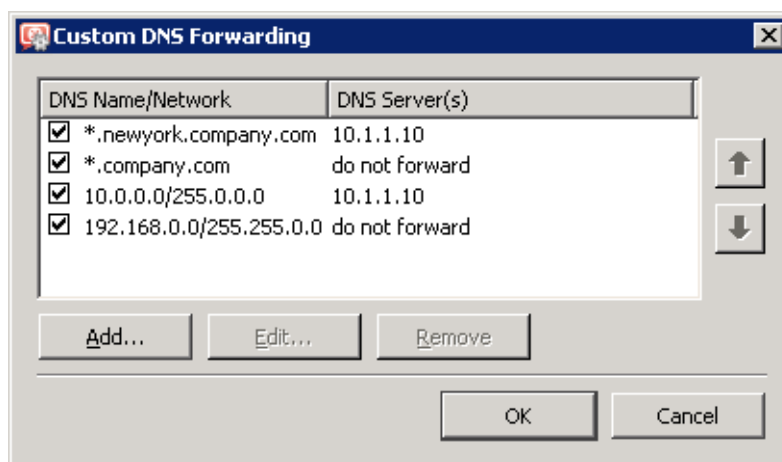


Figure 8.3 Specific settings of DNS forwarding

The rule can be defined for:

- DNS name — queries requiring names of computers 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)

Rules can be reordered by arrow buttons. This enables creating of more complex combinations of rules — e.g. exceptions for certain workstations or subdomains. As the rule list is processed from the top downwards, rules should be ordered starting by the most specific one (e.g. name of a particular computer) and with the most general one at the bottom (e.g. the main domain of the company). Similarly to this, rules for reversed DNS queries should be ordered by subnet mask length (e.g. with 255.255.255.0 at the top and 255.0.0.0 at the bottom). Rules for

queries concerning names and reversed queries are independent from each other. For better reference, it is recommended to start with all rules concerning queries for names and continue with all rules for reversed queries, or vice versa.

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

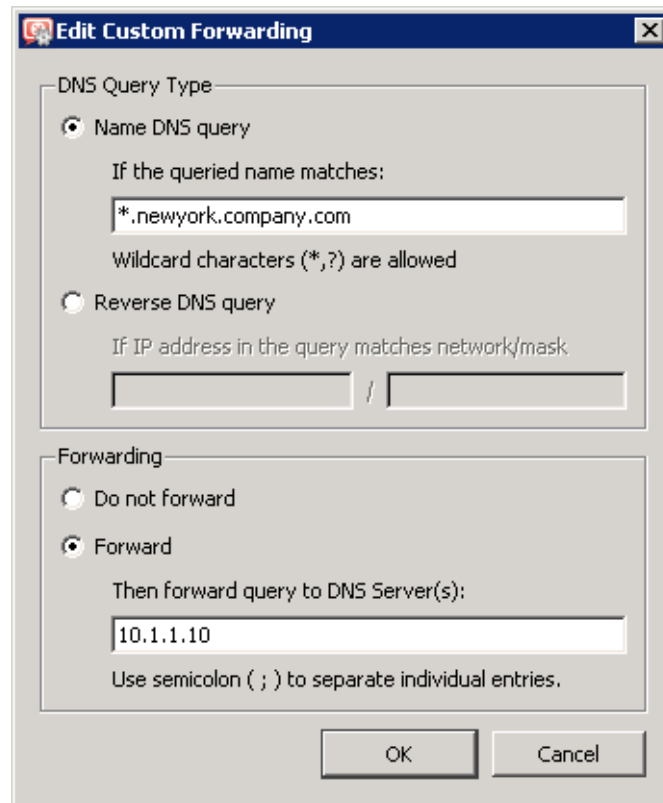


Figure 8.4 DNS forwarding — a new rule

- The *Name DNS query* option allows specification of a rule for name queries. Use the *If the queried name matches* entry to specify a corresponding DNS name (name of a host in the domain).

It is usually desirable to forward queries to entire domains rather than to specific names. Specification of a domain name may therefore contain * wildcard symbol (asterisk — substitutes any number of characters) and/or ? (question mark — substitutes a single character). The rule will be applied to all names matching with the string (hosts, domains, etc.).

— **Example:** —

DNS name will be represented by the string `?erio.c*`. The rule will be applied to all names in domains `kerio.com`, `cerio.com`, `aerio.c` etc., such as on `www.kerio.com`, `secure.kerio.com`, `www.aerio.c`, etc.

Warning

In rules for DNS requests, it is necessary to enter an expression matching the full DNS name! If, for example, the `kerio.c*` expression is introduced, only names `kerio.cz`, `kerio.com` etc. would match the rule and host names included in these domains (such as `www.kerio.cz` and `secure.kerio.com`) would not!

- 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.

If multiple DNS servers are specified, they are considered as primary, secondary, etc. If the *Do not forward* option is checked, DNS queries will not be forwarded to any other DNS server — *WinRoute* will search only in the hosts local file or in DHCP tables (see below). If requested name or IP address is not found, non-existence of the name/address is reported to the client.

8.2 DHCP server

The DHCP protocol (*Dynamic Host Configuration Protocol*) is used for easy TCP/IP configuration of hosts within the network. Upon an operation system start-up, the client host sends a configuration request that is detected by the DHCP server. 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. All client parameters can be set at the server only — at individual hosts, enable the option that TCP/IP parameters are configured automatically from the DHCP server. For most operating systems (e.g. *Windows*, *Linux*, etc.), this option is set by default — it is not necessary to perform any additional settings at client hosts.

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. This is performed automatically and transparently.

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:

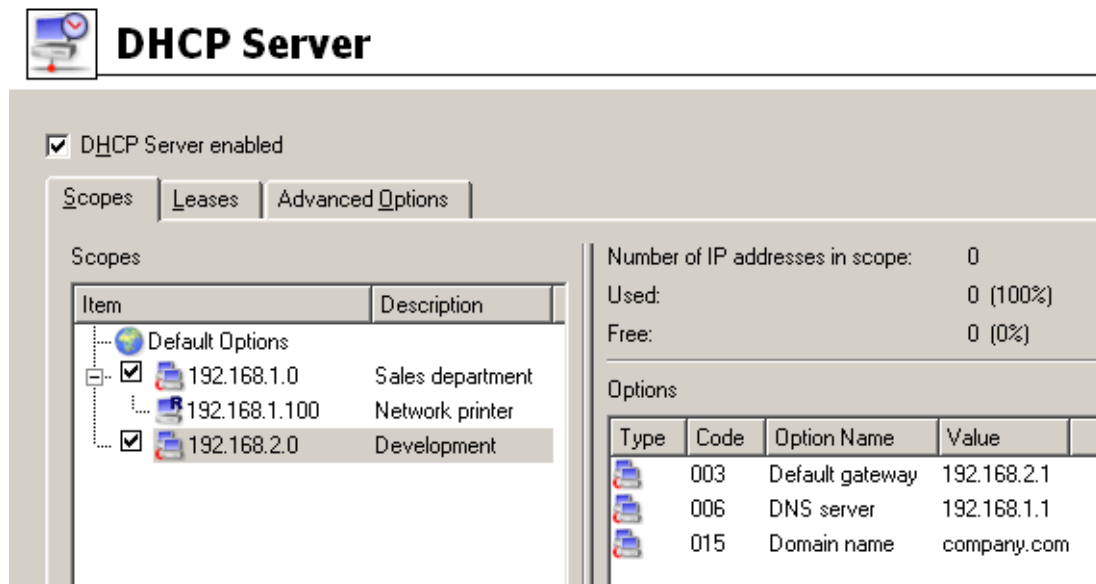


Figure 8.5 DHCP server — IP scopes

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.

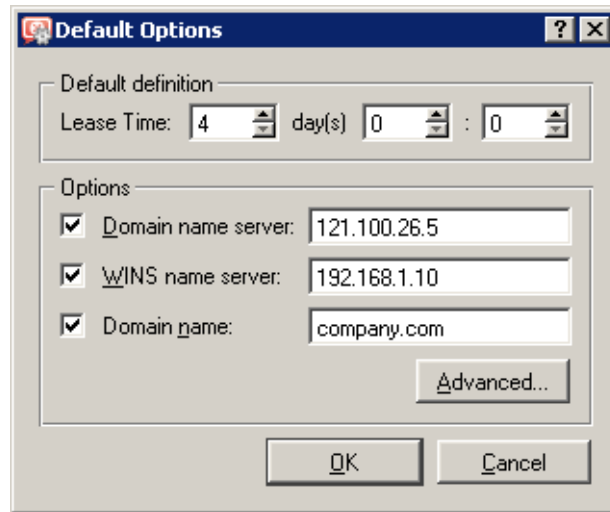


Figure 8.6 DHCP server — default DHCP parameters

DNS server

Any DNS server (or multiple DNS servers separated by semicolons) can be defined. We recommend you to use the *WinRoute's DNS* module as the primary server (first in the list) — IP address of the *WinRoute* host. The *DNS* module can cooperate with DHCP server (see chapter 8.1) 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 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.

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).

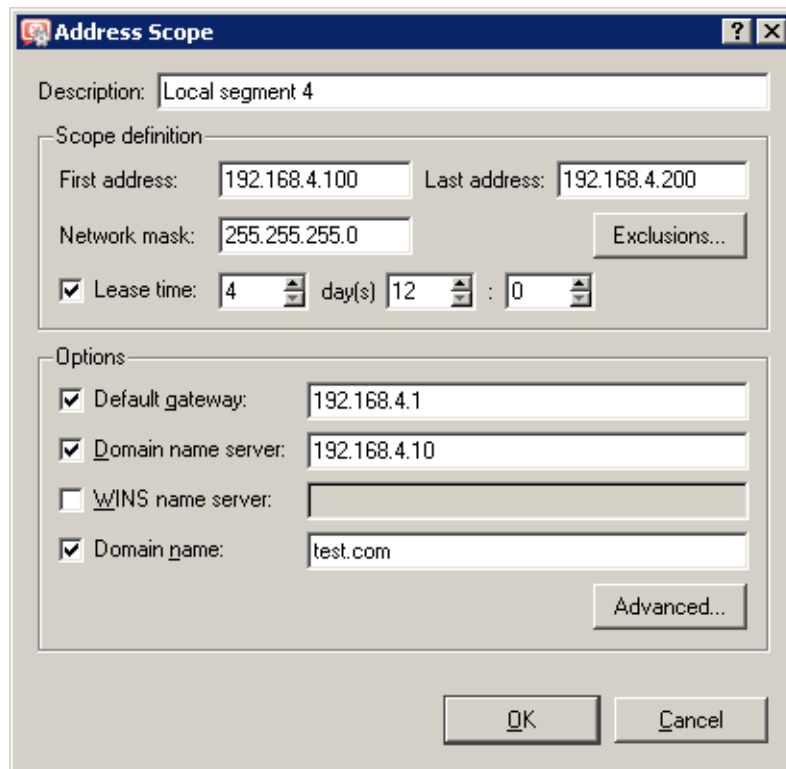


Figure 8.7 DHCP server — IP scopes definition

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.

Subnet mask

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

Note: The *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 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.

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.

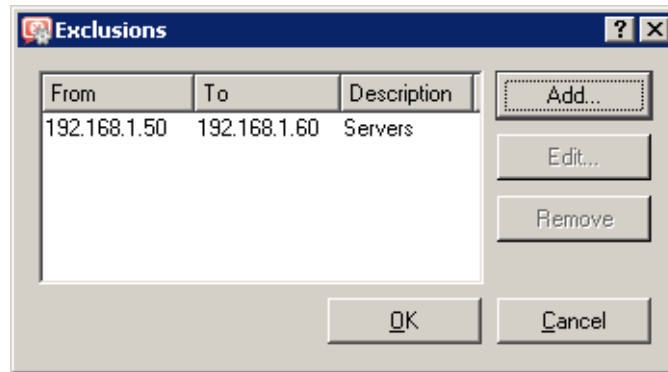


Figure 8.8 DHCP server — IP scopes exceptions

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. IP address of the interface the network is connected to. Default gateway of another network would be useless (not available to clients).
- *DNS server* — any DNS server (or more DNS servers separated with semicolons). We recommend you to use the *WinRoute's DNS* module as the primary server (first in the list) — IP address of the *WinRoute* host. The *DNS* module can cooperate with DHCP server (see chapter 8.1) so that it will always use correct IP addresses to response to requests on local host names.
- *WINS server*
- *Domain* — local Internet domain. Do not specify this parameter if there is no local domain.

— **Warning** —

This parameter is not used for specification of the name of *Windows NT* 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.

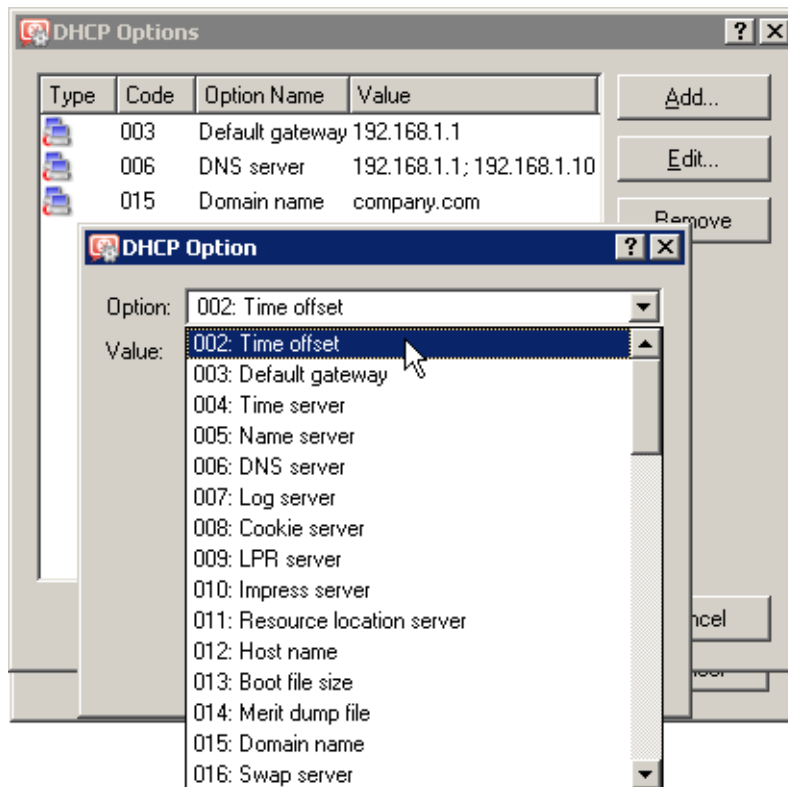


Figure 8.9 DHCP server — DHCP settings

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

Number of IP's in scope:	90
Used:	86 (96%)
Free:	4 (4%)

Figure 8.10 DHCP server — statistics (leased and free IP addresses within the scope)

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. This address can but does not have to belong to the scope of addresses dynamically leased, and it can also belong to any scope used for exceptions.

IP addresses can be reserved for:

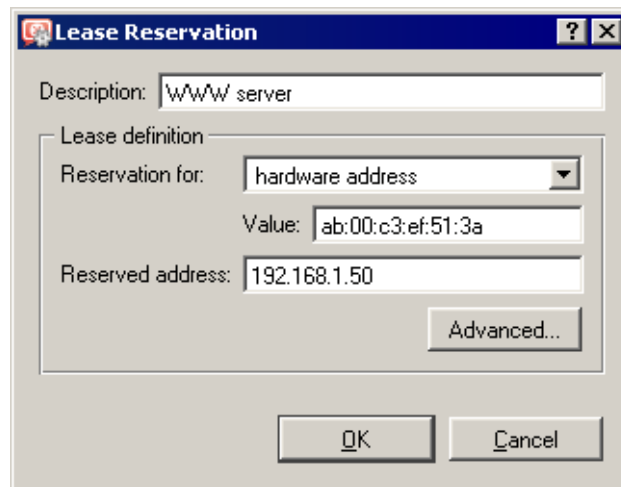


Figure 8.11 DHCP server — reserving an IP address

- 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 (e.g. the *Linux* operating system).

Click *Advanced* to set DHCP parameters which will accompany the address when leased. If the IP address is already included to a scope, DHCP parameters belonging to the scope are used automatically. In the *Lease Reservation* dialog window, additional parameters can be specified or/and new values can be entered for parameters yet existing.

Note: Another way to reserve an IP address is to go to the *Leases* tab, find the IP address leased dynamically to the host and reserve it (for details, see below).

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

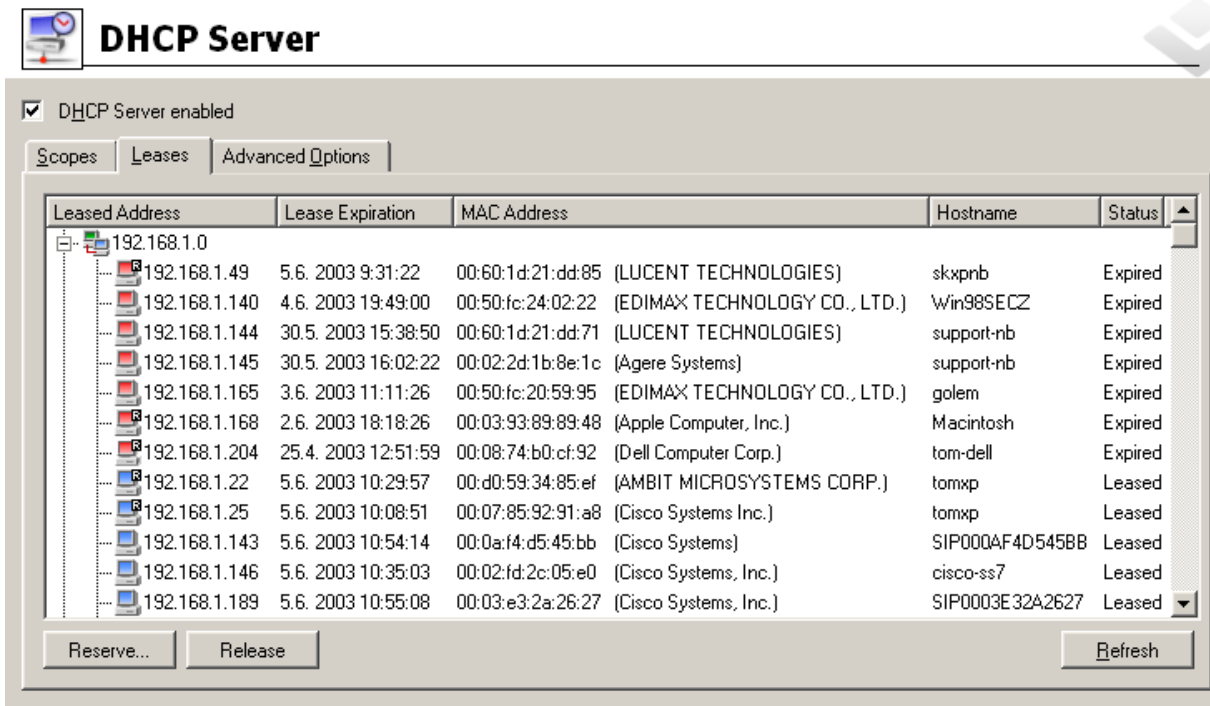


Figure 8.12 DHCP server — list of leased and reserved IP addresses

- *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).

Note:

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. If free IP addresses are lacked, these addresses can be leased to other clients.
2. Declined addresses are handled according to the settings in the *Options* tab (see below).

The following columns are hidden by default (for details on showing and hiding columns, see chapter 3.2):

- *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.

DHCP server — advanced options

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

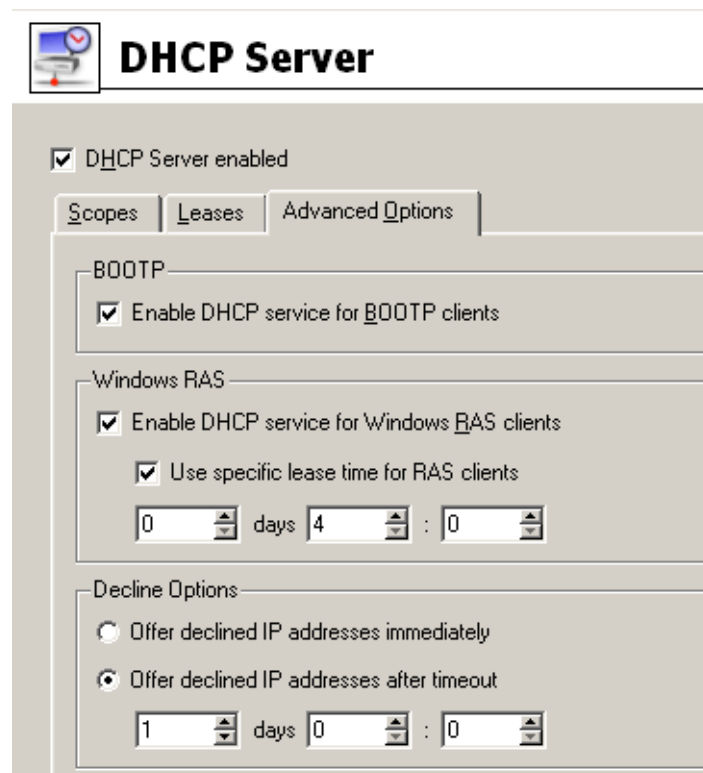


Figure 8.13 DHCP server — advanced options

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.

Warning

1. DHCP server cannot assign addresses to RAS clients connecting to the RAS server directly at the *WinRoute* host (for technical reasons, it is not possible to receive DHCP queries from the local RAS server). For such cases, it is necessary to set assigning of IP addresses in the RAS server configuration.
 2. The RAS service in *Windows* leases a new IP address for each connection (even if requested by the same client). *WinRoute* includes RAS clients in total number of clients when checking whether number of licensed users has been exceeded (see chapter 4.6). This implies that repeated connection of RAS clients may cause exceeding of the number of licensed users (if the IP scope for the RAS service is too large or/and an address is leased to RAS clients for too long time). Remote clients will be then allowed to connect and communicate with hosts in the local network, while they will not be allowed to connect to the Internet via *WinRoute*.
-

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).

8.3 Dynamic DNS for public IP address of the firewall

Kerio WinRoute Firewall provides (among others) services for remote access from the Internet to the local network (*VPN server* — see chapter 23 and the *Clientless SSL-VPN* interface — see chapter 24). Also other services can be accessible from the Internet — e.g. the *Kerio StaR* interface (see chapter 21), remote administration of *WinRoute* by the *Administration Console* (see chapter 16.2) or any other service (e.g. web server in local network — see chapter 7.4). These services are available at the firewall's public IP address. If this IP address is static and there exists a corresponding DNS record for it, a corresponding name can be used for access to a given service (e.g. `server.company.com`). If there is no corresponding DNS record, it is necessary to remember the firewall's IP address and use it for access to all services. If the public IP address is dynamic (i.e. it changes), it is extremely difficult or even impossible to connect to these services from the Internet.

This problem is solved by *WinRoute's* support for dynamic DNS. Dynamic DNS provides DNS record for a specific name of a server which will always keep the current IP address. This method thus allows making mapped services always available under the same server name, regardless of the fact if IP address changes and how often.

How cooperation with dynamic DNS works

Dynamic DNS (*DDNS*) is a service providing automatic update of IP address in DNS record for the particular host name. Typically, two versions of DDNS are available:

- free — user can choose from several second level domains (e.g. `no-ip.org`, `ddns.info`, etc.) and select a free host name for the domain (e.g. `company.ddns.info`).
- paid service — user registers their own domain (e.g. `company.com`) and the service provider then provides DNS server for this domain with the option of automatic update of records.

User of the service gets an account which is used for access authentication (this will guarantee that only authorized users can update DNS records. Update is performed via secured connection (typically HTTPS) to make sure that the traffic cannot be tapped. Dynamic DNS records can be updated either manually by the user or (mostly) by a specialized software — *WinRoute* in this case.

If *WinRoute* enables cooperation with dynamic DNS, a request for update of the IP address in dynamic DNS is sent upon any change of the Internet interface's IP address (including switching between primary and secondary Internet connection — see chapter 6.3). This keeps DNS record for the particular IP address up-to-date and mapped services may be accessed by the corresponding host name.

Note:

1. Usage of DDNS follows conditions of the particular provider.
2. Dynamic DNS records use very short time-to-live (TTL) and, therefore, they are kept in cache of other DNS servers or forwarders for a very short time. Probability that the client receives DNS response with an invalid (old) IP address is, therefore, very low.
3. Some DDNS servers also allow concurrent update of more records. Wildcards are used for this purpose.

Example: In DDNS there exist two host names, both linked to the public IP address of the firewall: `fw.company.com` and `server.company.com`. If the IP address is changed, it is therefore possible to send a single request for update of DNS records with name `*.company.com`. This requests starts update of DNS records of both names.

DDNS configuration in WinRoute

To set cooperation with the dynamic DNS server, go to the *Dynamic DNS* folder in *Configuration* → *Advanced Options*.

As already mentioned, the first step is to make an account (i.e. required dynamic DNS record with appropriate access rights) at a DDNS provider. *WinRoute* now supports these DDNS providers:

- *ChangeIP* (<http://www.changeip.com/>),
- *DynDNS* (<http://www.dyndns.org/>),
- *No-IP* (<http://www.no-ip.com/>).

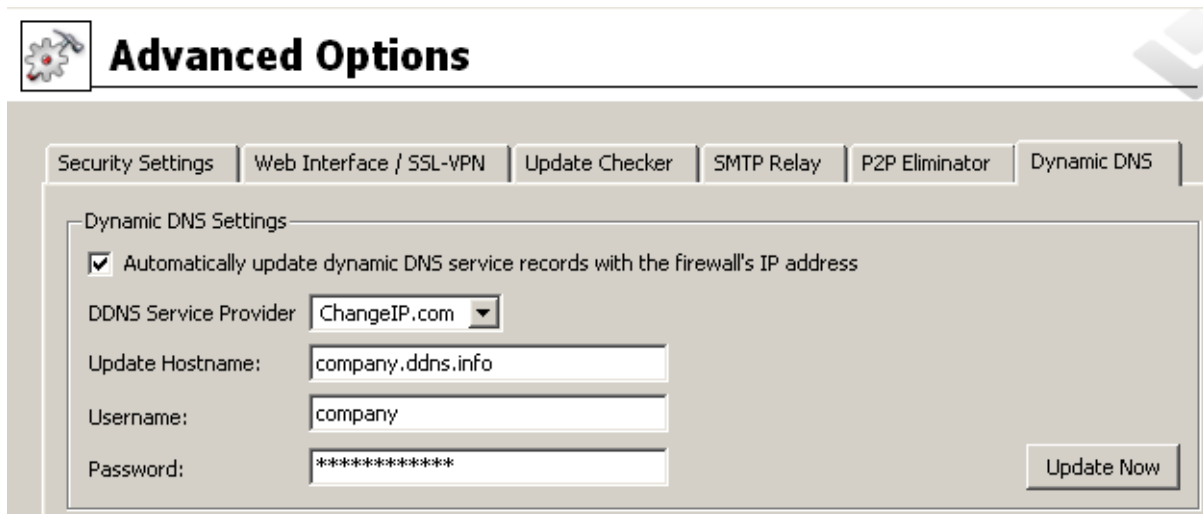


Figure 8.14 Setting cooperation with dynamic DNS server

On the *Dynamic DNS* tab, select a DDNS provider, enter DNS name for which dynamic record will be kept updated and set user name and password for access to updates of the dynamic record. If DDNS supports wildcards, they can be used in the host name.

Once this information is defined, it is recommended to test update of dynamic DNS record by clicking on *Update now*. This verifies that automatic update works well (the server is available, set data is correct, etc.) and also updates the corresponding DNS record (IP address of the firewall could have changed since the registration or the last manual update).

If an error occurs while attempting to update DNS record, an error is reported on the *Dynamic DNS* tab providing closer specification of the error (e.g. DDNS server is not available, user authentication failed, etc.). This report is also recorded in the *error* log.

8.4 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.
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 [12.2](#)). 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 functionality is kept if proxy server is used — it is not necessary to edit configuration of individual hosts (or only some hosts should be re-configured).

The *WinRoute's* proxy server can be used for HTTP, HTTPS and FTP protocols. Proxy server does not support the SOCKS protocol (a special protocol used for communication between the client and the proxy server).

Note: For detailed information on using FTP on the *WinRoute's* proxy server, refer to chapter [25.4](#).

Proxy Server Configuration

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

HTTP Policy

URL Rules | Content Rules | Cache | **Proxy Server** | Forbidden Words | ISS OrangeWeb Filter

General options

Enable non-transparent proxy server

Port:

Advanced options

Allow tunelled connections to all TCP ports

Required for HTTPS connections on non-standard ports.

Forward to parent proxy server

Server: :

Parent proxy server requires authentication

Username:

Password:

Set automatic proxy configuration script to:

Direct access

WinRoute proxy server

Allow browsers to use configuration script automatically via DHCP server in WinRoute

Figure 8.15 HTTP proxy server settings

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 [22.8](#)):

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.

Enable connection to any TCP port

This security option enables to allow or block so called tunneling of other application protocols (than HTTP, HTTPS and FTP) via the proxy server.

If this option is disabled, the proxy server allows to establish connection only to the standard HTTPS port 443) — it is supposed that secured web pages are being opened. If the option is enabled, the proxy server can establish connection to any port. It can be a non-standard HTTPS port or tunneling of another application protocol.

Note: This option does not affect the non-secured traffic performed by HTTP and/or FTP. In *WinRoute*, HTTP traffic is controlled by a protocol inspectors which allows only valid HTTP and FTP queries.

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* — enable this option if authentication by username and password is required by the parent proxy server. Specify the *Username* and *Password* login data.

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 *Kerio Web Filter* 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. *Internet Explorer*, *Firefox/SeaMonkey*, *Opera*, etc.) enable automatic configuration of corresponding parameters by using a script downloaded from a corresponding website specified by 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...

It is possible to let *Internet Explorer* be configured automatically by the DHCP server. To set this, enable the *Automatically detect settings* option.

WinRoute's DHCP server must be running (see chapter 8.2), otherwise the function will not work. TCP/IP parameters at the host can be static — *Internet Explorer* sends a special DHCP query when started.

Hint

This method enables to configure all *Internet Explorer* browsers at all local hosts by a single click.

8.5 HTTP cache

Using cache to access Web pages that are opened repeatedly reduces Internet traffic (in case of line where traffic is counted, it is also remarkable that using of cache decreases total volume of transferred data). Downloaded files are saved to the hard drive 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 the traffic. In the default configuration of *WinRoute*, this condition is met for the HTTP protocol at the default port 80 (for details, see chapters 7.3 and 14.3).

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).

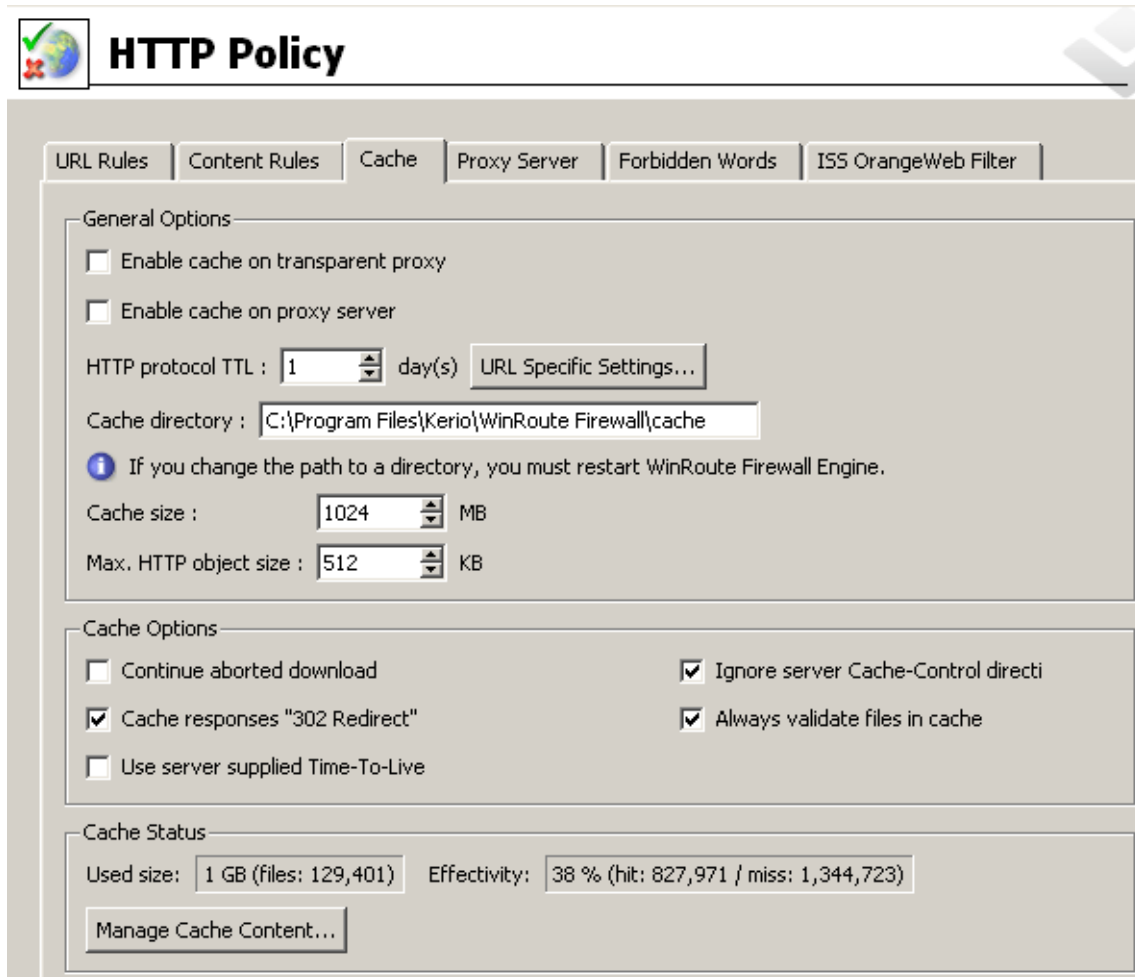


Figure 8.16 HTTP cache configuration

Enable cache on proxy server

Enables the cache for HTTP traffic via *WinRoute's* proxy server (see chapter [8.4](#)).

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. Old cache files in the original folder will be removed automatically.

Cache size

Size of the cache file on the disk. Maximal cache size allowed is *2 GB (2047 MB)*

Note:

1. If 98 per cent 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 disk (made by further cleaning or by manual removal).
2. The maximal cache size is applied in *WinRoute* since *6.2.0*. In older versions, maximal cache size allowed was *4 GB* (the threshold was cut for technical reasons). If, upon its startup, the *WinRoute Firewall Engine* detects that the cache size exceeds *2047 MB*, the size is changed to the allowed value automatically.
3. If the maximum cache size set is larger than the free space on the corresponding disk, the cache is not initialized and the following error is recorded in the *Error* log (see chapter [22.8](#)).

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.
- *Cache responses '302 Redirect'* — this option accelerates connection to redirected web pages.
Under usual circumstances, *302 Redirect* responses are not cached. HTTP protocol's return code *302* stands for temporary redirection — such redirection can be canceled any time or the target URL can change. If user applies the cached response to open a web page, the client can be redirected to an obsolete or invalid URL.
- *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).

Warning

Some web servers may attempt to bypass the cache by too short/long TTL.

- *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. This directive page may be misused for example to bypass 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 combination of the *Ctrl* and the *F5* keys to do this using either the *Internet Explorer* or the *Firefox/SeaMonkey* 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.

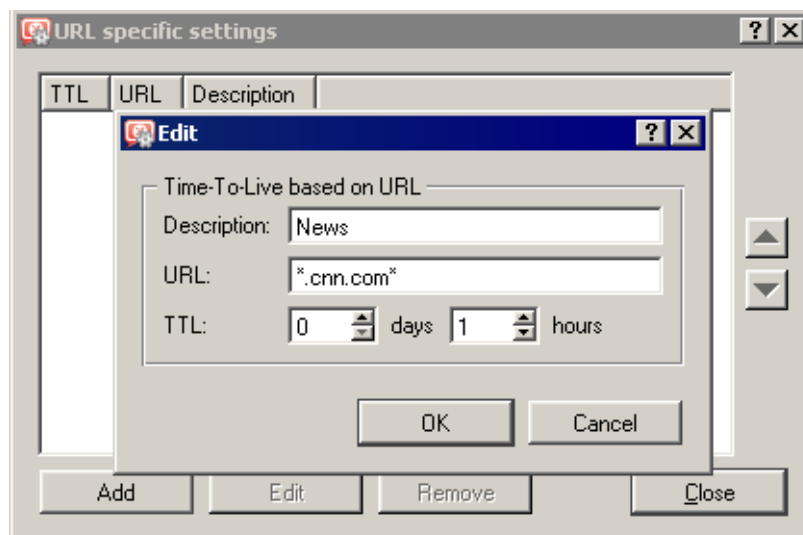


Figure 8.17 HTTP cache — specific settings for URL

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/us/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.

Cache status and administration

WinRoute allows monitoring of the HTTP cache status as well as manipulation with objects in the cache (viewing and removing).

At the bottom of the *Cache* tab, basic status information is provided such as the current cache size occupied and efficiency of the cache. The efficiency status stands for number of objects kept in the cache (it is not necessary to download these objects from the server) in proportion to the total number of queries (since the startup of the *WinRoute Firewall Engine*). The efficiency of the cache depends especially on user behavior and habits (if users visit certain webpages regularly, if any websites are accessed by multiple users, etc.) and, in a manner, it can be also affected by the configuration parameters described above. If the efficiency of the cache is permanently low (less than 5 per cent), it is recommended to change the cache configuration.

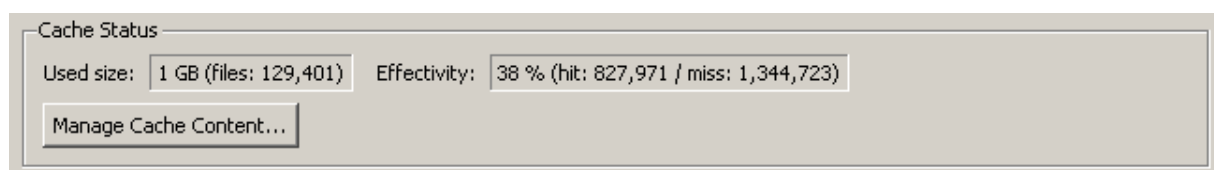


Figure 8.18 HTTP cache status information

Use the *Manage cache content...* button to open a dialog where objects kept in cache can be viewed, searched and/or removed.

To view objects in cache, specify the searched object in the *URL* entry. Objects can be specified either by an absolute URL (without protocol) — e.g. `www.kerio.com/image/menu.gif` or as a URL substring with `*` (substituting any number of any symbols and characters) and `?` (question mark substitutes a single character or symbol) wildcard symbols.

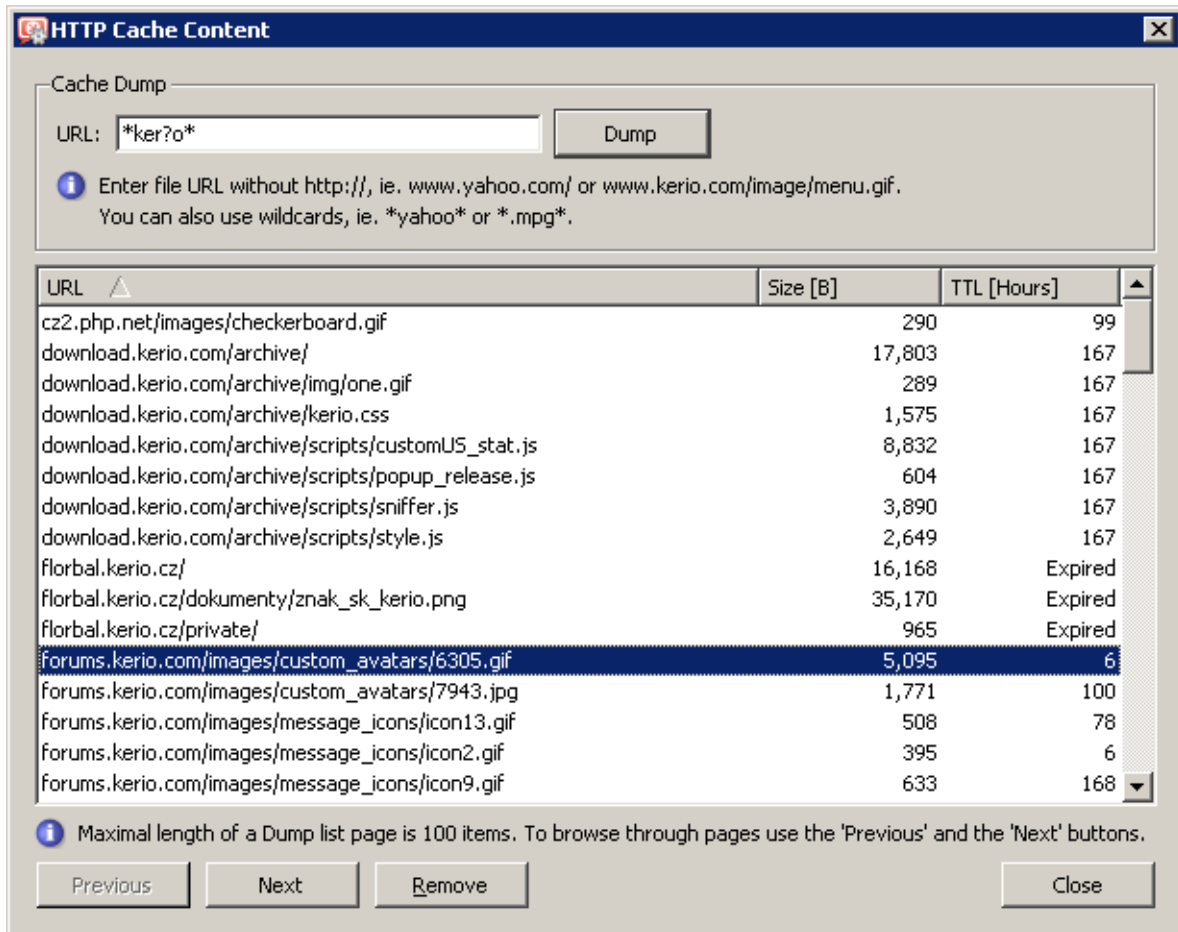


Figure 8.19 HTTP cache administration dialog

Example

Search for the `*ker?o*` string lists all objects with URL matching the specification, such as `kerio`, `kerbo`, etc.

Each line with an object includes URL of the object, its size in bytes (B) and number of hours representing *time left to the expiration*. To keep the list simple and well-organized, up to 100 items are displayed at a single page. The *Previous* and *Next* buttons can be used for browsing through the list pages.

The *Remove* button can be used to delete the selected object from the cache.

Hint

By clicking and dragging or by clicking and holding the *Ctrl* or *Shift* key, it is possible to select multiple objects.

Bandwidth Limiter

The main problem of shared Internet connection is when one or more users download or upload big volume of data and occupy great part of the line connected to the Internet (so called bandwidth). The other users are then limited by slower Internet connection or also may be affected by failures of certain services (e.g. if the maximal response time is exceeded).

The gravest problems arise when the line is overloaded so much that certain network services (such as mailserver, web server or VoIP) must be limited or blocked. This means that, by data downloads or uploads, even a single user may endanger functionality of the entire network.

The *WinRoute's Bandwidth Limiter* module introduces a solution of the most common problems associated with overloads of the Internet connection. This module is capable of recognizing connections where big data volumes are transmitted and it reserves certain part of the line's capacity for these transmissions. The remaining capacity is reserved for the other traffic (where big data volumes are not transmitted but where for example response time may play a role).

9.1 How the bandwidth limiter works and how to use it

The *Bandwidth Limiter* module provides two basic functions:

Speed limits for big data volumes transmissions

WinRoute monitors all connections established between the local network and the Internet. If a connection is considered as a transmission of big data volume, it reduces speed of such transmission to a defined value so that the other traffic is not affected. The bandwidth limiter does not apply to local traffic.

Note: Bandwidth limiting does not depend on traffic rules.

Speed limits for users with their quota exceeded

Users who have exceeded their quota for transmitted amount of data are logically considered as those who are often download or upload big data volumes. *WinRoute* enables to reduce speed of data transmission for these users so that other users and network services are not affected by their network activities. This restriction is automatically applied to users who exceed a quota (see chapter [15.1](#)).

9.2 Bandwidth Limiter configuration

The *Bandwidth Limiter* parameters can be set under *Configuration* → *Bandwidth Limiter*.

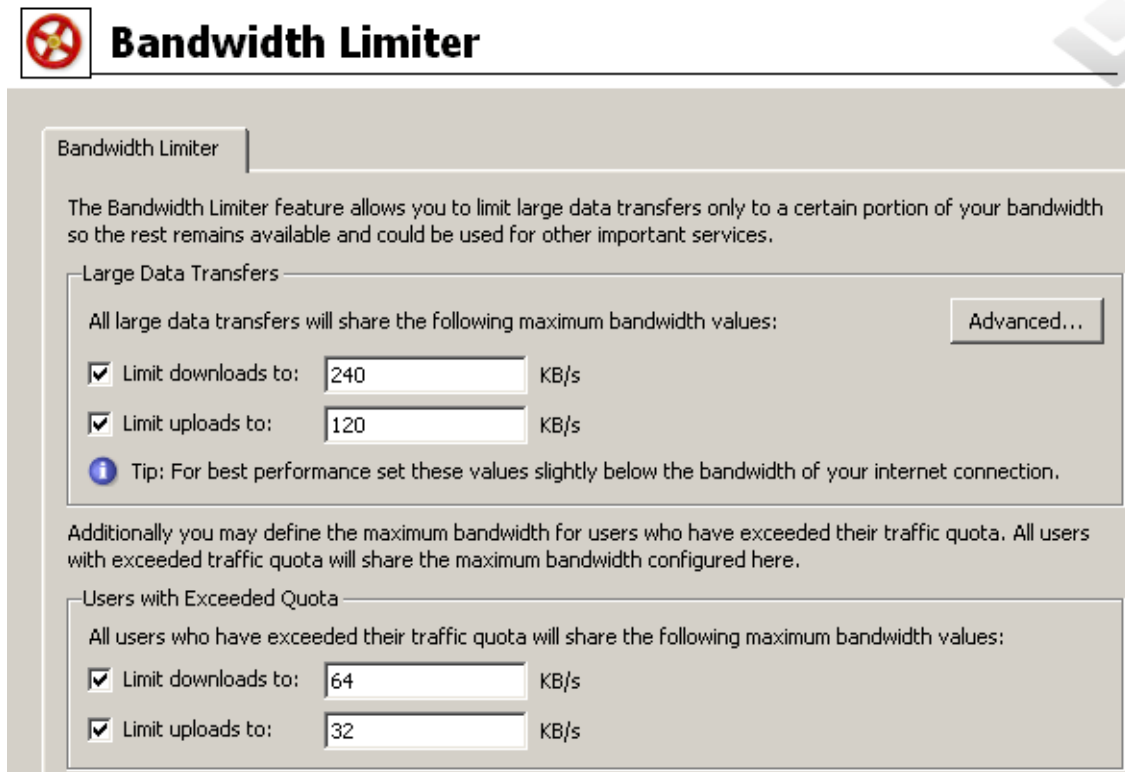


Figure 9.1 Bandwidth Limiter configuration

The *Bandwidth Limiter* module enables to define reduction of speed of incoming traffic (i.e. from the Internet to the local network) and of outgoing data (i.e. from the local network to the Internet) for transmissions of big data volumes and for users with their quota exceeded. These limits do not depend on each other. This means it is possible to use one of these functions, both or none.

Warning

In the *Bandwidth Limiter* module, speed is measured in kilobytes per second (*KB/s*), while ISPs usually use kilobits per second (*kbps*, *kbit/s* or *kb/s*), or in megabits per second (*Mbps*, *Mbit/s* or *Mb/s*). The conversion pattern is $1 \text{ KB/s} = 8 \text{ kbit/s}$.

A 256 kbit/s line's speed is 32 KB/s , a 1 Mbit/s line's speed is 128 KB/s .

Setting limit values

The top of the dialog box contains a section where limits for transfers of big data volumes can be set. These values determine bandwidth that will be reserved for these transfers. The remaining bandwidth is available for other traffic.

Tests have discovered that the optimal usage of the Internet line capacity is reached if the value is set to approximately 90 per cent of the bandwidth. If the values are higher, the bandwidth limiter is not effective (not enough speed is reserved for other connections and

services if too much big data volumes are transferred). If they are lower, full line capacity is often not employed.

Warning

For optimal configuration, it is necessary to operate with *real* capacity of the line. This value may differ from the information provided by ISP. One method of how to find out the real value of the line capacity is to monitor traffic charts (see chapter [20.2](#)) when you can be almost sure that the line is fully employed.

At the bottom of the dialog box, download and upload speed limits for users with exceeded traffic quota can be set. The bandwidth defined will be shared by all users with their quota exceeded. This implies that the total traffic volume of these users is limited by the bandwidth value set here.

No optimal values are known for these speed limits. *WinRoute* administrators decide themselves what part of the bandwidth will be reserved for these users. It is recommended to set the values so that activities of these users do not affect other users and services.

Note: It is also possible to block any traffic for a particular users who exceed their quota. The restriction described above are applied only if the *Don't block further traffic (Only limit bandwidth...)* action is set in configuration of the particular user account. For details, see chapter [15.1](#).

Advanced Options

Click on *Advanced* to define advanced *Bandwidth Limiter* parameters. These parameters apply only to large data volume transfers. They do not apply to users with exceeded quota (bandwidth values set for these users are applied without exception).

Services

Certain services may seem to perform large data volume transfers, although, in fact, they don't. Internet telephony (*Voice over IP — VoIP*) is a typical example. It is possible to define exceptions for such services so that the bandwidth limiter does not apply to them. It may also be desired to apply bandwidth limiter only to certain network services (e.g. when it is helpful to limit transfers via *FTP* and *HTTP*).

The *Services* tab enables definition of services to which bandwidth limiter will be applied:

- *Apply to all services* — the limits will be applied to all traffic between the local network and the Internet.
- *Apply to the selected services only* — the limits will apply only to the selected network services. Traffic performed by other services is not limited.
- *Apply to all except the selected services* — services specified in this section will be excluded from the bandwidth limiter restrictions, whereas the limiter will apply to any other services.

Click on *Select services* to open a dialog box where network services can be selected. Hold the *Ctrl* or the *Shift* key to select multiple services. All services defined in *Configuration* → *Definitions* → *Services* are available (for details, refer to chapter [sect-services"/>](#)).

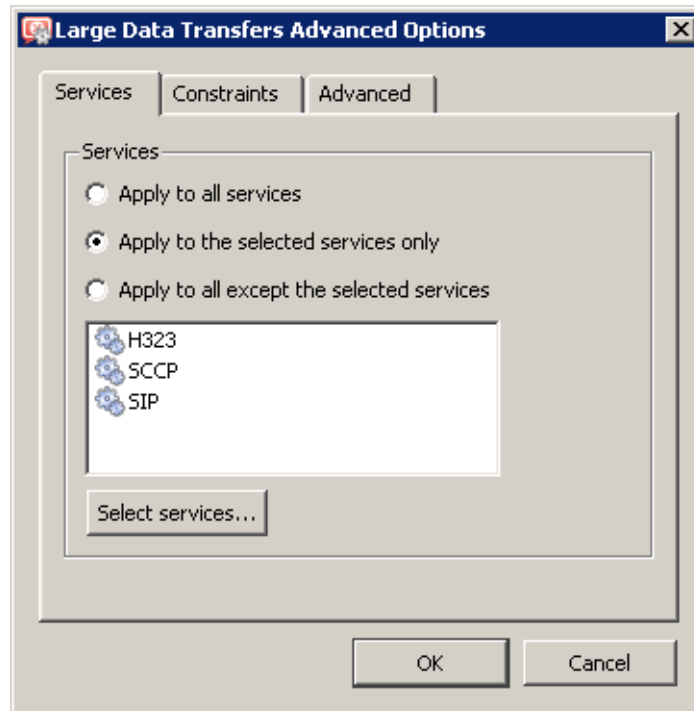


Figure 9.2 Bandwidth Limiter — network services

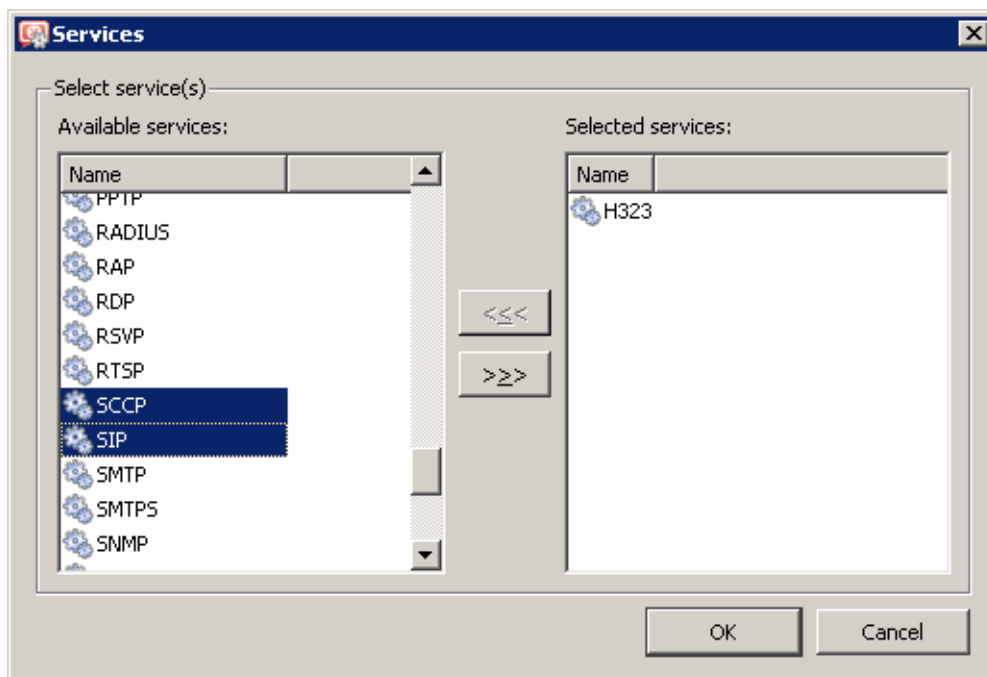


Figure 9.3 Bandwidth Limiter — selection of network services

IP Addresses and Time Interval

It may be also helpful to apply bandwidth limiter only to certain hosts (for example, it may be undesired to limit a mailserver in the local network or communication with the corporate web server located in the Internet). This exclusive IP group may contain any IP

addresses across the local network and the Internet. Where user workstations use fixed IP addresses, it is also possible to apply this function to individual users.

It is also possible to apply bandwidth limiter to a particular time interval (e.g. in work hours).

These parameters can be set on the *Constraints* tab.

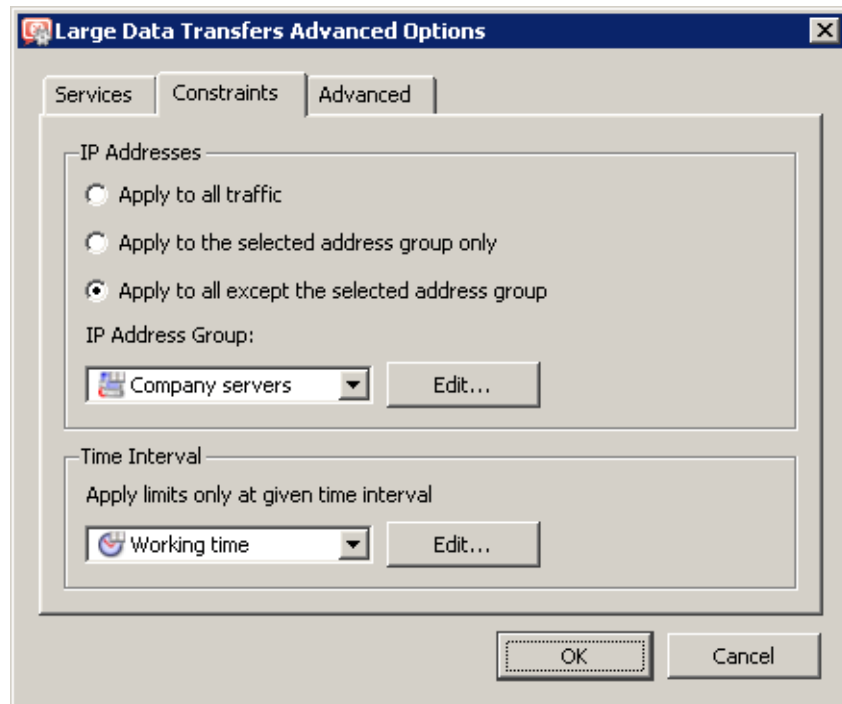


Figure 9.4 Bandwidth Limiter — IP Addresses and Time Range

At the top of the *Constraints* tab, select a method how bandwidth will be applied to IP addresses and define the IP address group:

- *Apply to all traffic* — the IP address group specification is inactive it is irrelevant.
- *Apply to the selected address group only* — the bandwidth limiter will be applied only if at least one IP address involved in a connection belongs to the address group. The other traffic will not be limited.
- *Apply to all except the selected address group* — the bandwidth limiter will not be applied if at least one IP address involved in a connection belongs to the address group. Any other traffic will be limited.

In the lower section of the *Constraints* tab, a time range within which the bandwidth would be limited can be set. Click *Edit* to edit the selected interval or to create a new one (details in chapter [14.2](#)).

Setting of parameters for detection of large data volume transfers

The *Advanced* tab enables setting of parameters that will be used for detection of transmissions of large data volume — the minimal volume of transmitted data and inactivity time interval. The default values (200 KB and 5 sec) are optimized in accordance with long-term testing in full action.

Caution! Changes of these values may reduce Bandwidth Limiter performance dramati-

cally. With exception of special conditions (testing purposes) it is highly recommended not to change the default values!

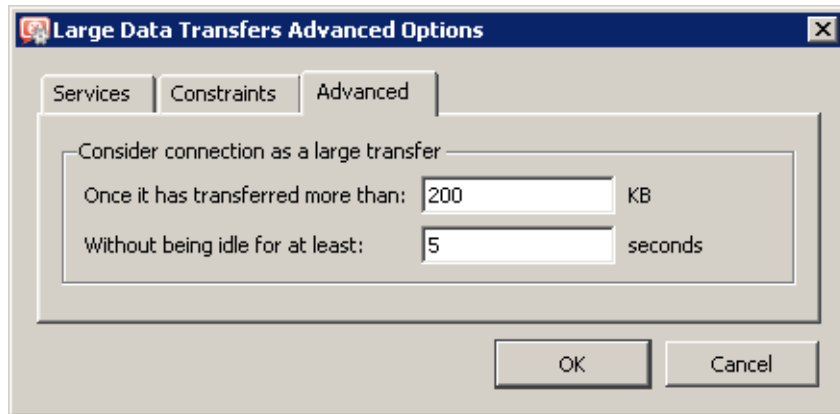


Figure 9.5 Bandwidth Limiter — setting parameters for detection of large data volume transfers
For detailed description of the detection of large data volume transmissions, refer to chapter [9.3](#).

9.3 Detection of connections with large data volume transferred

This chapter provides description of the method used by the *Bandwidth Limiter* module to detect connections where large data volumes are transmitted. This description is an extra information which is not necessary for usage of the *Bandwidth Limiter* module.

Network traffic is different for individual services. For example, web browsers usually access sites by opening one or more connections and using them to transfer certain amount of data (objects included at the page) and then closes the connections. Terminal services (e.g. *Telnet*, *SSH*, etc.) typically use an open connection to transfer small data volumes in longer intervals. Large data volume transfers typically uses the method where the data flow continuously with minimal intervals between the transfer impulses.

Two basic parameters are tested in each connection: volume of transferred data and duration of the longest idle interval. If the specified data volume is reached without the idleness interval having been thresholded, the connection is considered as a transfer of large data volume and corresponding limits are applied.

If the idle time exceeds the defined value, the transferred data counter is set to zero and the process starts anew. This implies that each connection that *once* reaches the defined values is considered as a large data volume transfer.

The value of the limit for the amount of data transmitted and the minimal idleness period are configuration parameters of the *Bandwidth Limiter* (see chapter [9.2](#)).

Examples:

The detection of connections transferring large data volumes will be better understood through the following examples. The default configuration of the detection is as follows: at least 200 KB of data must be transferred while there is no interruption for 5 sec or more.

1. The connection at figure 9.6 is considered as a transmission of large data volume after transfer of the third load of data. At this point, the connection has transferred 200 KB of data while the longest idleness interval has been only 3 sec.

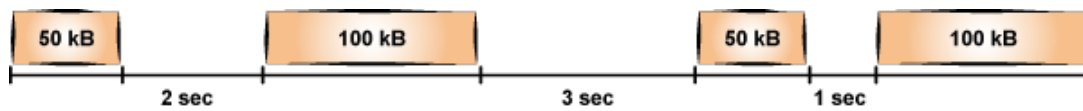


Figure 9.6 Connection example — short idleness intervals

2. Connection at figure 9.7 is not considered as a large data volume transfer, since after 150 KB of data have been transferred before an only 5 sec long idleness interval and then, only other 150 KB of data have been transmitted within the connection.



Figure 9.7 Connection example — long idleness interval

3. The connection shown at figure 9.8 transfers 100 KB of data before a 6 sec idleness interval. For this reason, the counter of transferred data is set to zero. Other three blocks of data of 100 KB are then transmitted. When the third block of data is transferred, only 200 KB of transmitted data are recorded at the counter (since the last long idleness interval). Since there is only a 3 sec idleness interval between transmission of the second and the third block of data, the connection is considered as a large data volume transfer.

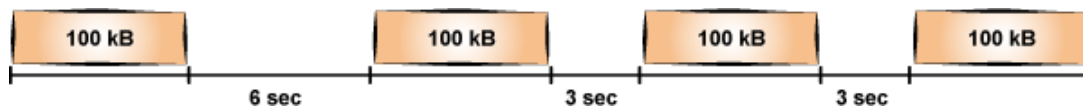


Figure 9.8 Connection example — long idleness interval at the beginning of the transfer

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 (i.e. all hosts the user is currently connected from). This implies that a user group represents all IP addresses its members are currently connected from.

Besides access restrictions, user authentication can be used also for monitoring of their activities in the *Kerio StaR* interface (see chapter [21](#)), in logs (see chapter [22](#)), in the list of opened connections (see chapter [19.2](#)) and in the overview of hosts and users (see chapter [19.1](#)). 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. In statistics, this host's traffic will be included in the group of *not logged in* users.

10.1 Firewall User Authentication

Any user with their own account in *WinRoute* can authenticate at the firewall (regardless their access rights). Users can connect:

- Manually — by opening the *WinRoute* web interface in their browser
`https://server:4081/` or `http://server:4080/`
(the name of the server and the port numbers are examples only — see chapter [11](#)).
It is also possible to authenticate for viewing of the web statistics (see chapter [21](#)) at
`https://server:4081/star` or `http://server:4080/star`
Note: Login to the *Web Administration* interface at
`https://server:4081/admin` or `http://server:4080/admin`
is not equal to user authentication at the firewall (i.e. the user does not get authenticated at the firewall by the login)!
- Automatically — IP addresses of hosts from which they will be authenticated automatically can be associated with individual users. This actually means that whenever traffic coming from the particular host is detected, *WinRoute* assumes that it is currently used by the particular user, and the user is considered being authenticated from the IP address. However, users may authenticate from other hosts (using the methods described above).
IP addresses for automatic authentication can be set during definition of user account (see chapter [15.1](#)).
This authentication method is not recommended for cases where hosts are used by multiple users (user's identity might be misused easily).

- Redirection — when accessing any website (unless access to this page is explicitly allowed to unauthenticated users — see chapter 12.2).
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: Users will be redirected to a secured or unsecured web interface according to the fact which version of web interface is allowed (see chapter 11.1). If both versions are allowed, the secured web interface will be used.
- Using NTLM — if *Internet Explorer* or *Firefox/SeaMonkey* 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 chapter 25.3.

User authentication advanced options

Login/logout parameters can be set on the *Authentication Options* tab under *Users and Groups* → *Users*.

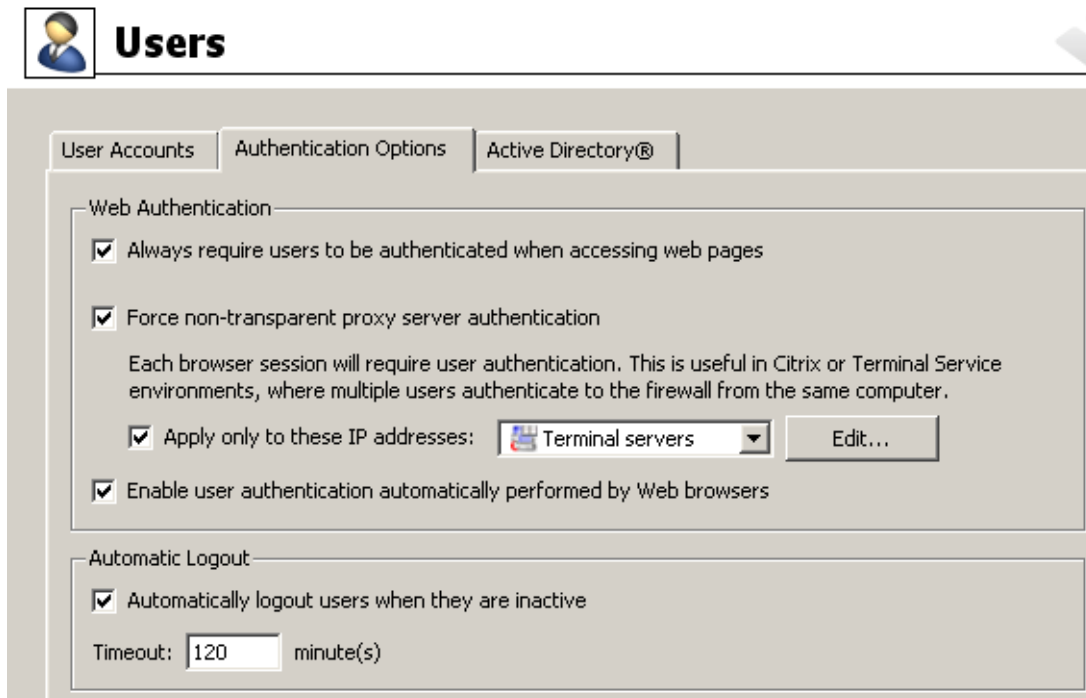


Figure 10.1 User Authentication Options

Redirection to the authentication page

If the *Always require users to be authenticated when accessing web pages* option is enabled, user authentication will be required for access to any website (unless the user is already authenticated). The method of the authentication request depends on the method used by the particular browser to connect to the Internet:

- *Direct access* — the browser will be automatically redirected to the authentication page of the *WinRoute's* web interface (see chapter [11.2](#)) and, if the authentication is successful, to the solicited web page.
- *WinRoute proxy server* — the browser displays the authentication dialog and then, if the authentication is successful, it opens the solicited web page.

If the *Always require users to be authenticated when accessing web pages* 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 [12.2](#)).

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).

Force non-transparent proxy server authentication

Under usual circumstances, a user connected to the firewall from a particular computer is considered as authenticated by the IP address of the host until the moment when they log out manually or are logged out automatically for inactivity. However, if the client station allows multiple users connected to the computer at a moment (e.g. *Microsoft Terminal Services*, *Citrix Presentation Server* or *Fast user switching* on *Windows XP*, *Windows Server 2003*, *Windows Vista* and *Windows Server 2008*), the firewall requires authentication only from the user who starts to work on the host as the first. The other users will be authenticated as this user.

In case of *HTTP* and *HTTPS*, this technical obstruction can be passed by. In web browsers of all clients of the multi-user system, set connection to the Internet via the *WinRoute's* proxy server (for details, see chapter [8.4](#)), and enable the *Enable non-transparent proxy server* option in *WinRoute*. The proxy server will require authentication for each new session of the particular browser.⁵

Forcing user authentication on the proxy server for initiation of each session may bother users working on “single-user” hosts. Therefore, it is desirable to force such authentication only for hosts used by multiple users. For this purpose, you can use the *Apply only for these IP addresses* option.

Automatic authentication (NTLM)

If the *Enable user authentication automatically...* option is checked and *Internet Explorer* (version 5.01 or later) or *Firefox/SeaMonkey* (core version 1.3 or later) is used, it is possible to authenticate the user automatically using the NTLM method.

This means that the browser does not require username and password and simply uses the identity of the first user connected to *Windows*. However, the NTLM method is not

⁵ *Session* is every single period during which a browser is running. For example, in case of *Internet Explorer*, *Firefox* and *Opera*, a session is terminated whenever all windows and tabs of the browser are closed, while in case of *SeaMonkey*, a session is not closed unless the *Quick Launch* program is stopped (an icon is displayed in the toolbar's notification area when the program is running).

available for other operating systems.

For details, refer to chapter [25.3](#).

Automatically logout users when they are inactive

Timeout is a time interval (in minutes) of allowed user inactivity. When this period expires, the user is 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, otherwise login data of a user who forgot to logout might be misused by an unauthorized user.

Web Interface

WinRoute includes a special web server which provides an interface where statistics can be viewed (*Kerio StaR*), as well as for setting of some user account parameters and for firewall administration via web browser (*Web Administration*). This Web server is available over SSL or using standard HTTP with no encryption (both versions include identical pages).

Use the following URL ('server' refers to the name or IP of the *WinRoute* host, 4080 represents a standard HTTP interface port) to open the unsecured version of the web interface.

```
https://server:4080/
```

To use the encrypted version specify the HTTPS protocol and number of the port of the encrypted Web interface (default is 4081):

```
https://server:4081/
```

This chapter addresses setting of parameters for the web interface in the *WinRoute's* administration program. *Kerio StaR* and user web interface are addressed in detail in the *Kerio WinRoute Firewall — User's Guide*.

11.1 Web interface preferences

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

Note: In *WinRoute* for *Windows*, the *Web Interfaces* tab includes also options for *Kerio SSL-VPN*. For detailed information on this component, see chapter [24](#).

Enable Web Interface (HTTP)

Use this option to open the unsecured version (HTTP) of the Web interface. The default port for this unsecured interface is 4080.

Note: The main disadvantage of usage of the unsecured web interface is that the network traffic may be tapped and user login data might be misused. Therefore, the secured web interface should be preferred.

Enable secured Web Interface (HTTPS)

Use this option to open the secured version (HTTPS) of the Web interface. The default port for this interface is 4081.

Name of the server on which *WinRoute* is running

Server DNS name that will be used for purposes of the Web interface (e.g. `server.company.com`).

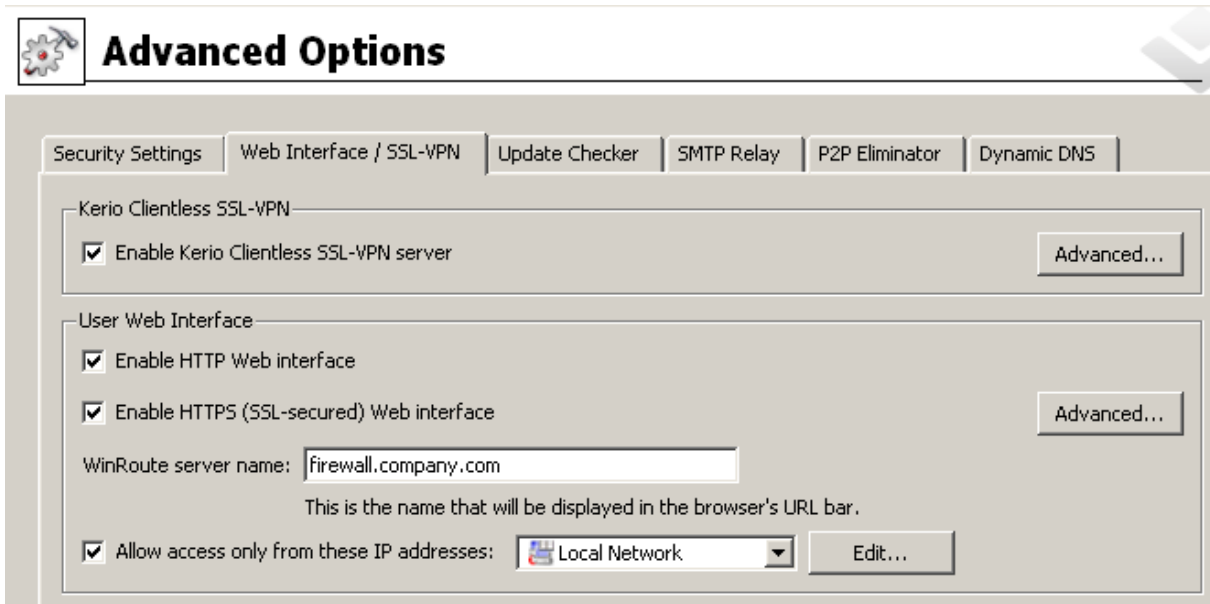


Figure 11.1 Configuration of WinRoute's Web Interface

The name need not be necessarily identical with the host name, however, there must exist an appropriate entry in DNS for proper name resolution. The SSL certificate for the secure web interface (see below) should be also issued for the server (i.e. the server name).

The server name is also used in case that *WinRoute* needs redirect the browser to the login page (for example if an unauthenticated user attempts to open a web page where authentication is required — see chapters [10.1](#) and [12.2](#)).

Notes:

1. If all clients accessing the web interface use the *DNS* module 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 [8.1](#).
2. In the *Software Appliance / VMware Virtual Appliance* edition, name of the server defined on the *System Configuration* tab is set automatically in this item — see chapter [16.1](#).

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 [14.1](#)).

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

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

Configuration of ports of the Web Interface

Use the *TCP ports* 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).

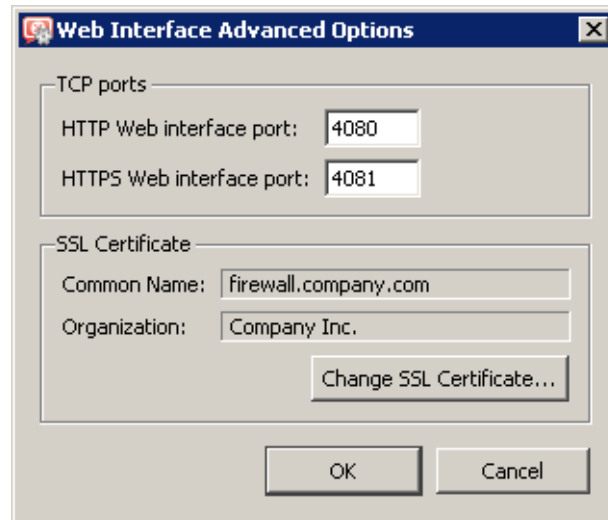


Figure 11.2 Configuration of ports in WinRoute's Web Interface

Hint: If no WWW server is running on the *WinRoute* host, the standard port of the HTTP protocol (i.e. 80) can be used for the unsecured web interface and the standard port of the HTTPS protocol (i.e. port 443) for the secured web interface. If standard ports are used, the port number is not necessarily required in URLs for pages of the web interface.

However, in *WinRoute* for *Windows*, the standard HTTPS port (443) uses the *Clientless SSL-VPN* interface (see chapter 24). Therefore, it cannot be used for secured web interface in the default configuration.

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 22.8):

```
Socket error: Unable to bind socket for service to port 80.
(5002) Failed to start service "WebInterface"
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 for the Web Interface

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.

The asymmetric cipher uses two keys: a public one for encrypting and a private one for decrypting. As their names suggest, the public (encrypting) key is available to anyone wishing to establish a connection with the server, whereas the private (decrypting) key is available only to the server and must remain secret. The client, however, also needs to be able to identify the server (to find out if it is truly the server and not an impostor). For this purpose there is a certificate, which contains the public server key, the server name, expiration date and other details. To ensure the authenticity of the certificate it must be certified and signed by a third party, the certification authority.

Communication between the client and server then follows this scheme: the client generates a symmetric encryption key for 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. This key is then used for encryption and decipher any other traffic.

Generate or Import Certificate

During *WinRoute* installation, a testing certificate for the SSL-secured Web interface is created automatically (it is stored in the `sslcert` subdirectory under the *WinRoute*'s installation directory, in the `server.crt` file; the private key for the certificate is saved as `server.key`). The certificate created is unique. However, it is issued against a non-existing server name and it is not issued by a trustworthy certificate authority. This certificate is intended to ensure functionality of the secured Web interface (usually for testing purposes) until a new certificate is created or a certificate issued by a public certificate authority is imported.

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 or about the subject represented by your server.

You can obtain your own certificate, which verifies your server's identity, by two means.

You can create your own self-signed certificate. Click *Generate Certificate* in the dialog where current server status is displayed. 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). When created, the certificate is saved as `server.crt` and the corresponding private key as `server.key`.

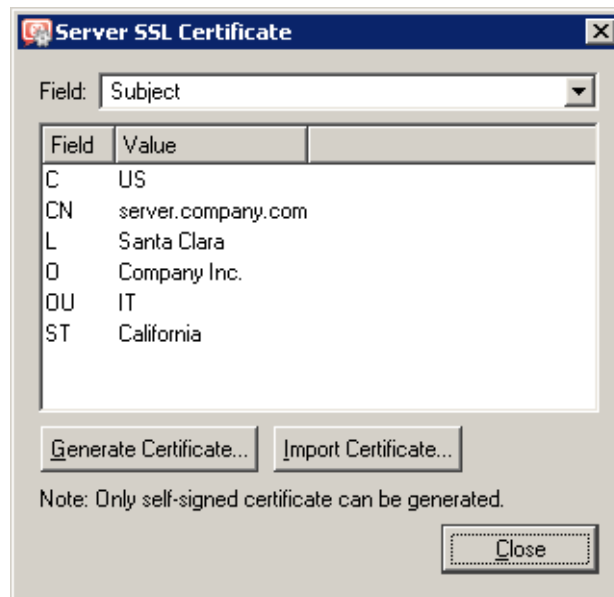


Figure 11.3 SSL certificate of WinRoute's Web interface



Figure 11.4 Creating a new "self-signed" certificate for WinRoute's Web interface

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 it is unique and the identity of your server is guaranteed by it. Clients will be warned only about the fact that the certificate was not issued by a trustworthy certification authority. However, they can install the certificate in the browser without worrying since they are aware of who and why created the certificate. Secure communication is then ensured for them and no warning will be displayed again because your certificate has all it needs.

Another option is to purchase a full certificate from a public certification authority (e.g.

Verisign, Thawte, SecureSign, SecureNet, Microsoft Authenticode, etc.).

To import a certificate, open the certificate file (*.crt) and the file including the corresponding private key (*.key). These files are stored in `sslcert` under the *WinRoute*'s installation directory.

The process of certification is quite complex and requires a certain expertise. For detailed instructions contact Kerio technical support.

11.2 User authentication at the web interface

User authentication is required for access to the *WinRoute*'s web interface. Any user with their own account in *WinRoute* can authenticate to the web interface. Depending on the right to view statistics (see chapter [15.2](#)), either *Kerio StaR* is opened or a page with status information and personal preferences is displayed upon logon.

If more than one *Active Directory* domain are used (see chapter [15.4](#)), the following rules apply to the user name:

- *Local user account* — the name must be specified without the domain (e.g. `admin`),
- *Primary domain* — missing domain is acceptable in the name specification (e.g. `jsmith`), but it is also possible to include the domain (e.g. `jsmith@company.com`),
- *Other domains* — the name specified must include the domain (e.g. `drdolittle@usoffice.company.com`).

If none or just one *Active Directory* domain is mapped, all users can authenticate by their usernames without the domain specified.

Note: Authentication at the web interface is a basic user authentication method at the firewall. Other authentication methods are described in chapter [10.1](#).

Chapter 12

HTTP and FTP filtering

WinRoute provides a wide range of features to filter traffic using HTTP and FTP protocols. These protocols are the most spread and the most used in the Internet.

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

Let's focus on filtering options featured by *WinRoute*. For their detailed description, read the following chapters.

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 *Kerio Web Filter* module (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

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

12.1 Conditions for HTTP and FTP filtering

For HTTP and FTP content filtering, the following conditions must be met:

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

An appropriate protocol inspector is activated automatically unless its use is denied by traffic rules. For details, refer to chapter 7.3.

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 7.3).
3. FTP protocols cannot be filtered if the secured authentication (SASO) is used.
4. Both HTTP and FTP rules are applied also when the *WinRoute's* proxy server is used (then, condition 1 is irrelevant). However, FTP protocol cannot be filtered if the parent proxy server is used (for details, see chapter 8.4). In such a case, FTP rules are not applied.
5. If the proxy server is used (see chapter 8.4), It is also possible to filter HTTPS servers (e.g. `https://secure.kerio.com/`). However, it is not possible to filter individual objects at these servers.

12.2 URL Rules

These rules allow the administrator to limit access to Web pages with URLs that meet certain criteria. They include other functions, such as filtering of web pages by occurrence forbidden words, blocking of specific items (scripts, active objects, etc.) and antivirus switch for certain pages.

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

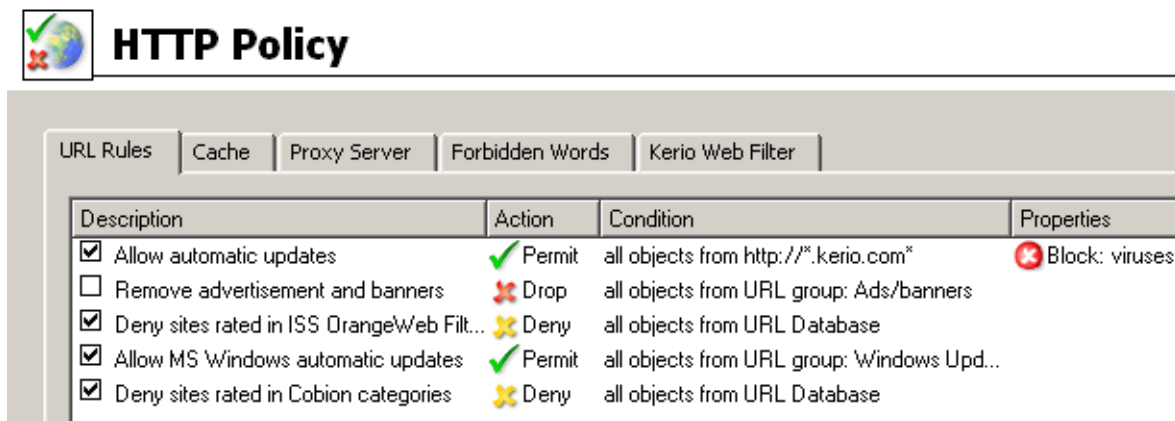


Figure 12.1 URL Rules

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). 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).

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.

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 *Kerio Web Filter* database, etc.).
- *Properties* — advanced options for the rule (e.g. anti-virus check, content filtering, etc.).

The following columns are hidden by default. To view them, use the *Modify columns* function in the context menu — for details, see chapter [3.2](#).

- *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. These rules are disabled by default. These rules are available to the *WinRoute* administrators.

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.

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

Description of the rule (information for the administrator).

If user accessing the URL is

Select which users this rule will be applied on:

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

Note:

1. It is often desired that the firewall requires user authentication before letting them open a web page. This can be set on the *Authentication Options* tab in *Users* (refer to chapter [15.1](#)). Using the *do not require authentication* option,

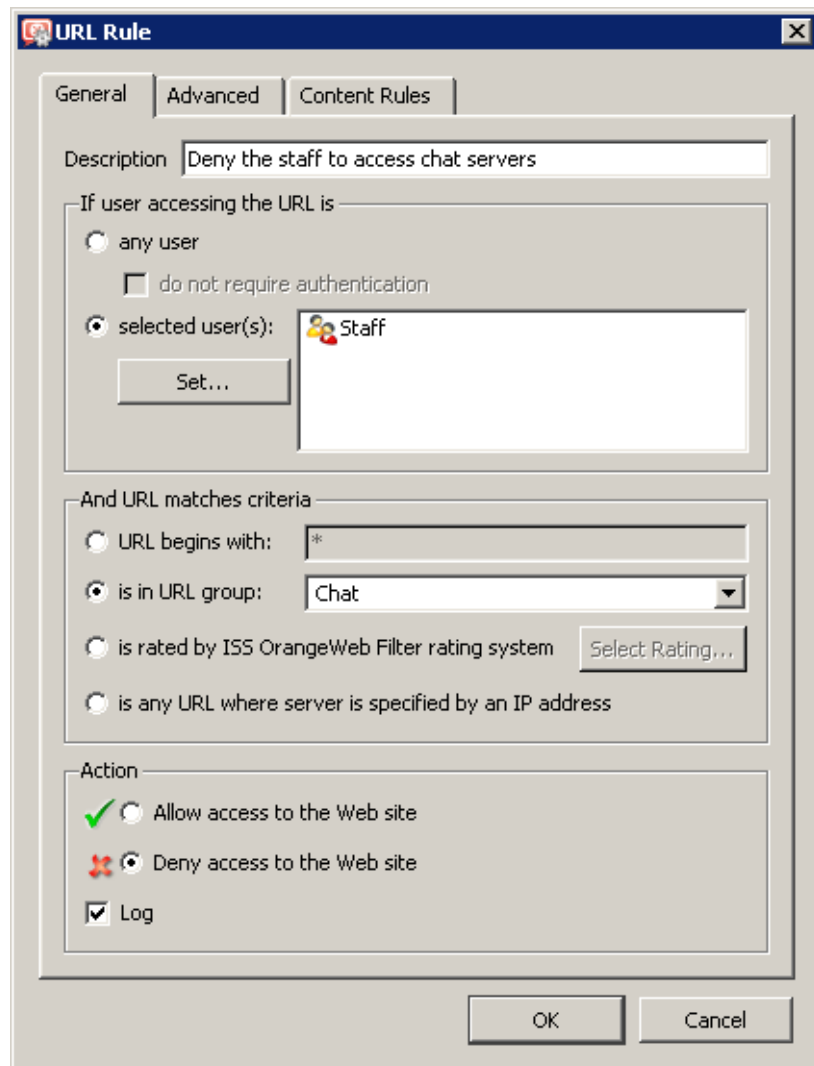


Figure 12.2 URL Rule — basic parameters

for example a rule allowing access to certain pages without authentication can be defined.

2. Unless authentication is required, the *do not require authentication* option is ineffective.
- *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 than one user /group at once).
Note: In rules, username represents IP address of the host from which the user is currently connected to the firewall (for details, see chapter [10.1](#)).

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*)
 Server names represent any URL at a corresponding server (www.kerio.com/*).

- *is in URL group* — selection of a URL group (refer to chapter [14.4](#)) which the URL should match with
- *is rated by Kerio Web Filter rating system* — the rule will be applied on all pages matched with a selected category by the *Kerio Web Filter* module.
 Click on the *Select Rating...* button to select from *Kerio Web Filter* categories. For details, refer to chapter [12.3](#).
- *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. This trick is often used by servers offering illegal downloads.

Warning

If access to servers specified by IP addresses is not denied, users can bypass URL rules where servers are specified by names.

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 [22.9](#)).

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 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 [14.2](#)).

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.

Click on the *Edit* button to edit IP groups (for details see chapter [14.1](#)).

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.

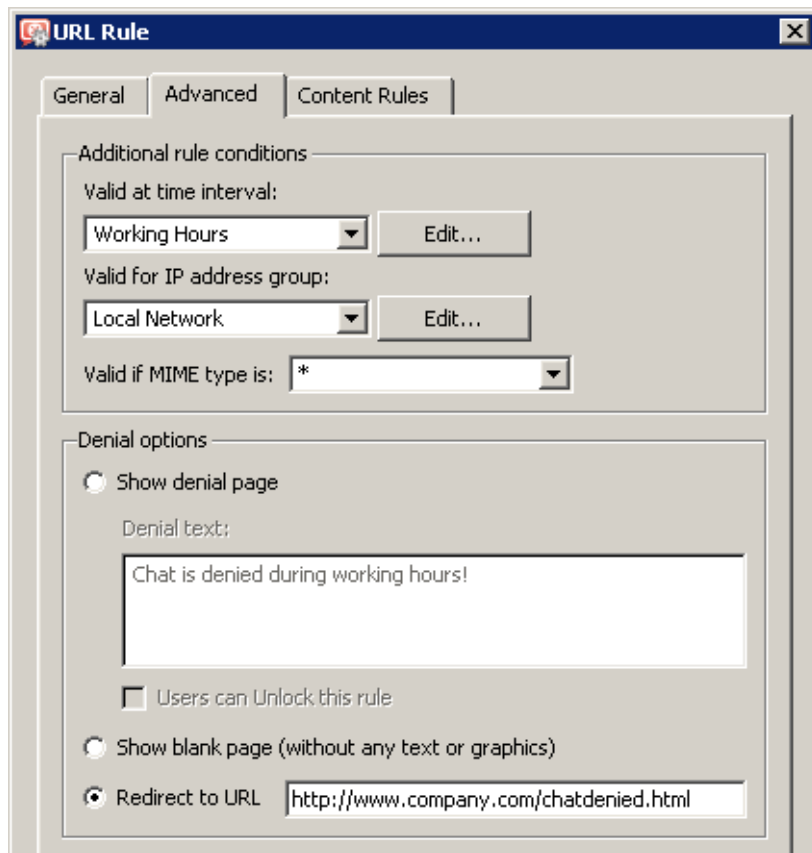


Figure 12.3 URL Rule — advanced parameters

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 enabled. Using this button users can force *WinRoute* to open the required page even though this site is denied by a URL rule. The rule will be opened for certain time (10 minutes by default). Each user can unlock a limited number of denied pages (up to 10 pages at once). All unlocked pages are logged in the *Security* log (see chapter 22.11).

Rules can be unlocked only by users with corresponding rights (see chapter 15.1). This implies that unauthenticated (anonymous) users can never unlock rules.

Note:

1. If any modifications are done within URL rules, all unlock rules are removed immediately.
2. For security reasons, no HTML tags are allowed in the restriction text. If the plaintext format is not sufficient, it is recommended to use redirection to

another page (see below).

- A blank page — user will not be informed why access to the required page was denied.
- 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.

The *Content Rules* tab allows to set rules for filtering of certain web page elements. Parameters on this tab can be set only for rules allowing access (on the *General* tab, the *Allow access to the web site* option is checked).

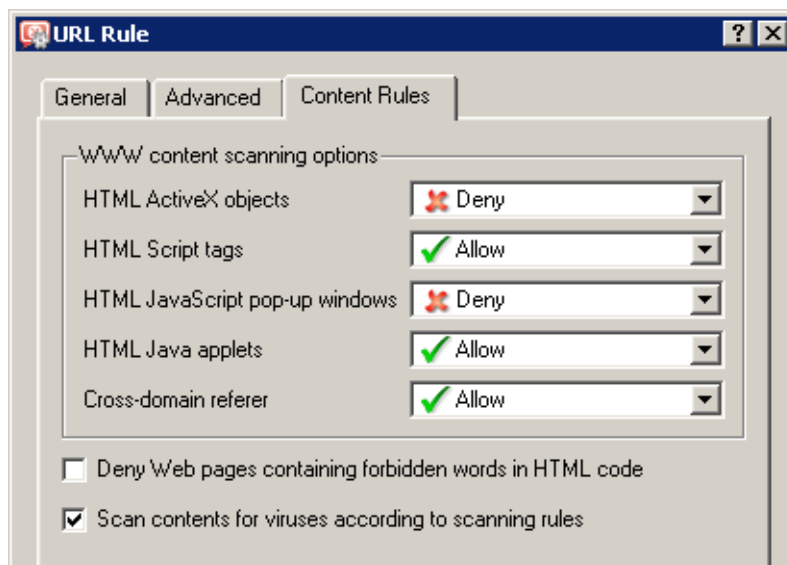


Figure 12.4 Options for Websites with content meeting a URL rule

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 [15.2](#)). Specific settings in URL rules beat user account settings.

Deny Web pages containing ...

Use this option to deny users to access Web pages containing words/strings defined on the *Forbidden Words* tab in the *Configuration/Content Filtering → HTTP Policy*.

For detailed information on forbidden words, see chapter [12.4](#).

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 [13.3](#)) 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.

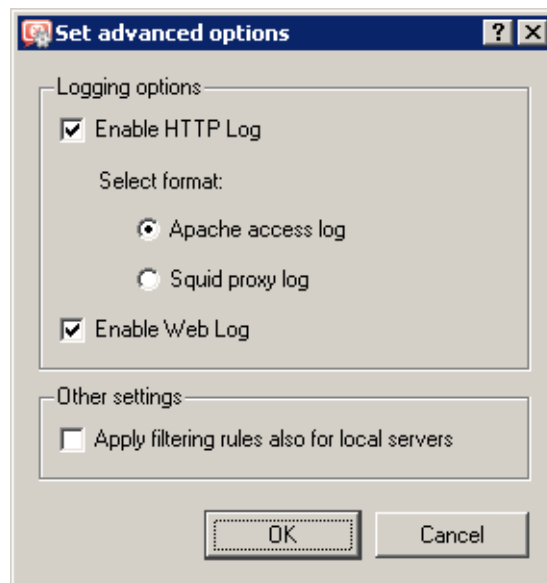


Figure 12.5 HTTP protocol inspector settings

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 22.10) and to the *Web* log (refer to chapter 22.14).

Log format can be chosen for the *Enable HTTP Log* item: *Apache* access log (<http://www.apache.org/>) or *Squid* proxy log (<http://www.squid-cache.org/>). This may be important especially when the log would be processed by a specific analysis tool.

Both *HTTP* and *Web* logs are enabled by default. The *Apache* option is selected by default for its better reference.

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 7). 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.

12.3 Content Rating System (Kerio Web Filter)

The *Kerio Web Filter* module 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.

Kerio Web 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: A special license is bound with *Kerio Web Filter* (subscription). Unless *WinRoute* includes subscription for this module, the module behaves as a trial version only (this means that it is automatically disabled after 30 days from the *WinRoute* installation and options in the *Kerio Web Filter* tab will not be available). For detailed information about the licensing policy, read chapter [44](#).

Kerio Web Filter configuration

The *Kerio Web Filter* module can be set and configured through the *Kerio Web Filter* tab in *Configuration* → *Content Filtering* → *HTTP Policy*.

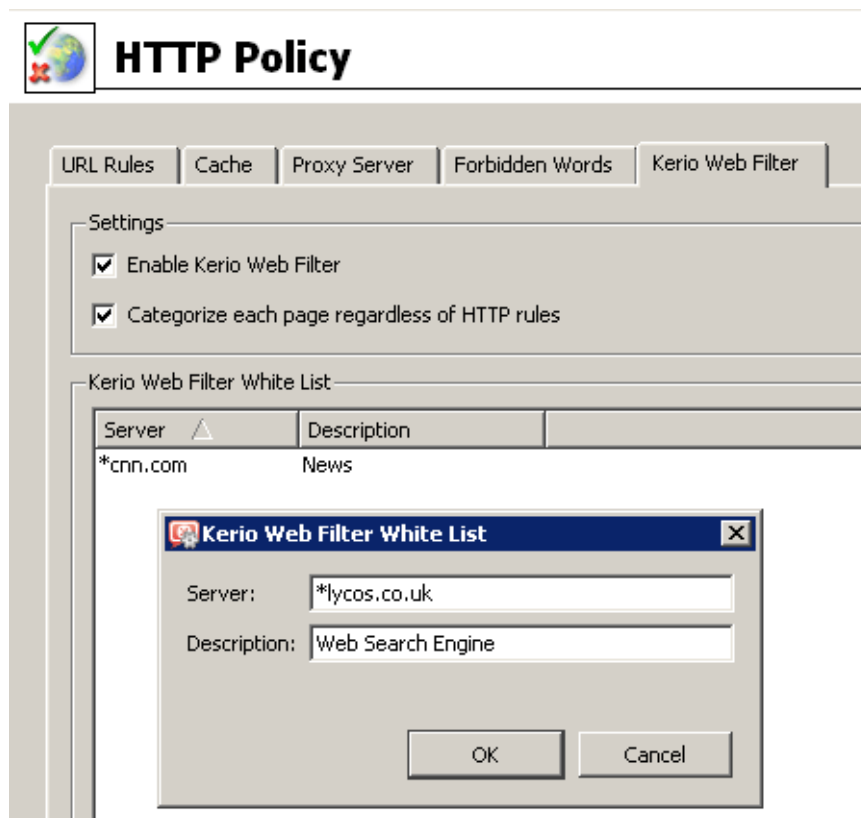


Figure 12.6 Kerio Web Filter configuration

Enable Kerio Web Filter

use this option to enable/disable the *Kerio Web Filter* module for classification of web-sites.

If *Kerio Web Filter* is disabled:

- the other options in the *Kerio Web Filter* tab are not available,
- all URL rules which use the *Kerio Web Filter* classification are disabled (for details, refer to chapter [12.3](#)).

Categorize each page regardless of HTTP rules

If this option is enabled, *Kerio Web Filter* categorization will be applied to any web pages (i.e. to all HTTP requests processed by the *HTTP* protocol inspector).

Categorization of all pages is necessary for statistics of the categories of visited web pages (see chapter [21](#)). If you do not intend to keep these statistics, it is recommended to disable this option (categorization of all web pages might be demanding and it might decrease *WinRoute* performance).

Servers (Web sites) not to be rated by the module can be specified in *Kerio Web Filter white list*. Use the *Add* button to open a dialog where a new item (server or a Web page) can be added.

Server

Use the *Server* item to specify web sites not to be classified by the *Kerio Web Filter*. The following items can be specified:

- server name (e.g. `www.kerio.com`). Server name represents any URL at a corresponding server,
- address of a particular webpage without protocol specification (`http://`) — e.g. `www.kerio.com/index.html`,
- URL using wildcard matching (e.g. `*.ker?o.*`). An asterisk stands for any number of characters (even zero), `a*.ker?o.*` question-mark represents just one symbol.

Description

Comments for the items defined. For reference only.

Kerio Web Filter use

To enable classification of Websites by the *Kerio Web Filter* module, this module must be running and all corresponding parameters must be set.

Whenever *WinRoute* processes a URL rule that requires classification of pages, the *Kerio Web Filter* module 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.

On the *URL Rules* tab in *Configuration* → *Content Filtering* → *HTTP Rules*, define a rule by using image [12.7](#) as guidance:

The *is rated by Kerio Web Filter rating system* is considered the key parameter. The URL of each opened page will be rated by the *ISS OrangeWeb Filter* module. 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 *Kerio Web Filter* rating categories can be chosen. Select the *Job Search / Job offers* rating category (pages including job offers).

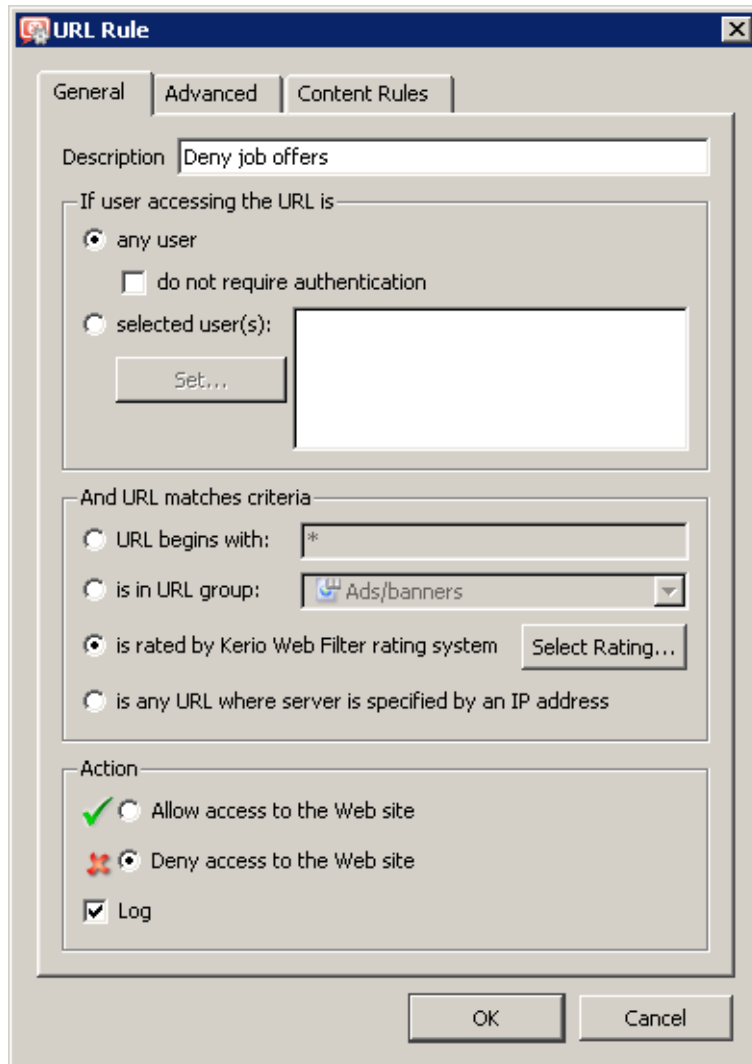


Figure 12.7 Kerio Web Filter rule

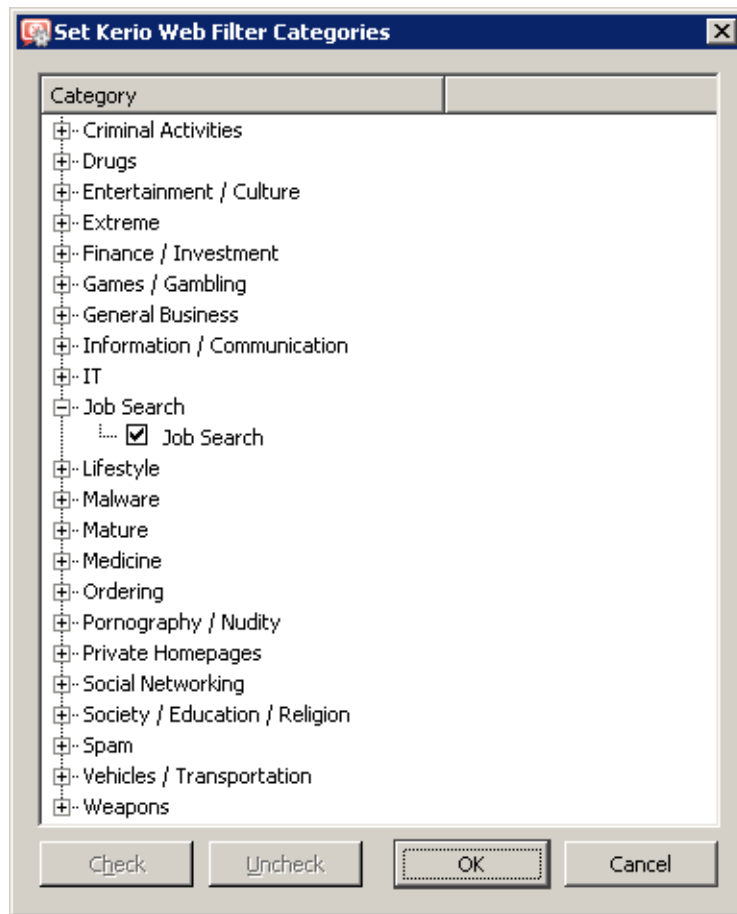


Figure 12.8 Selection of Kerio Web Filter categories

Note:

1. You can define multiple URL rules that will use the *Kerio Web Filter* rating technology. Multiple categories may be used for each rule.
2. We recommend you to unlock rules that use the *Kerio Web 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. All unlock queries are logged into the *Filter* log — here you can monitor whether unlock queries were appropriate or not.

12.4 Web content filtering by word occurrence

WinRoute can also filter Web pages that include undesirable words.

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 (so called threshold value), the page is blocked.

So called forbidden words are used to filter out web pages containing undesirable words. URL rules (see chapter [12.2](#)) define how pages including forbidden content will be handled.

— **Warning** —

Definition of forbidden words and threshold value is ineffective unless corresponding URL rules are set!

Definition of rules filtering by word occurrence

First, suppose that some forbidden words have been already defined and a threshold value has been set (for details, see below).

On the *URL Rules* tab under *Configuration* → *Content Filtering* → *HTTP Policy*, create a rule (or a set of rules) to allow access to the group of web pages which will be filtered by forbidden words. Go to the *Content Rules* tab under *HTTP Rule* to enable the web content filter.

Take a rule that will filter all web sites by occurrence of forbidden words as an example.

- On the *General* tab, allow all users to access any web site.

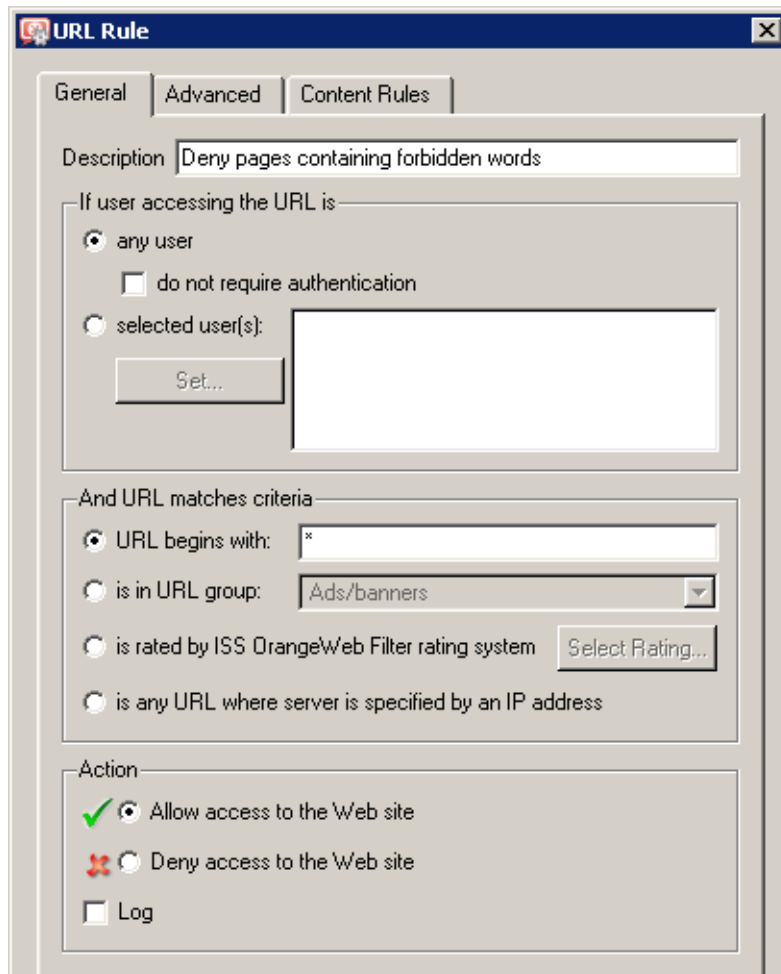


Figure 12.9 A rule filtering web pages by word occurrence (allow access)

- On the *Content Rules* tab, check the *Deny Web pages containing...* option to enable filtering by word occurrence.

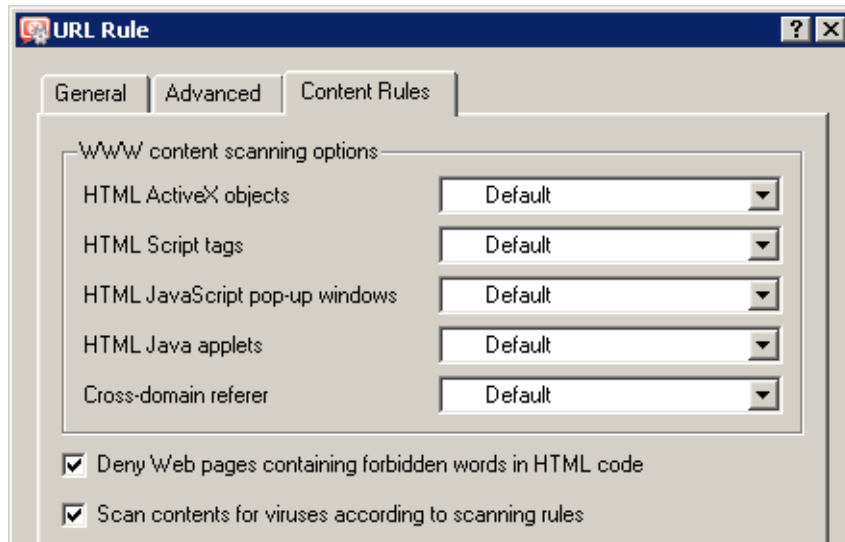


Figure 12.10 A rule filtering web pages by word occurrence (word filtering)

Word groups

To define word groups go to the *Word Groups* tab in *Configuration* → *Content Filtering* → *HTTP Policy*, the *Forbidden Words* tab. 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.

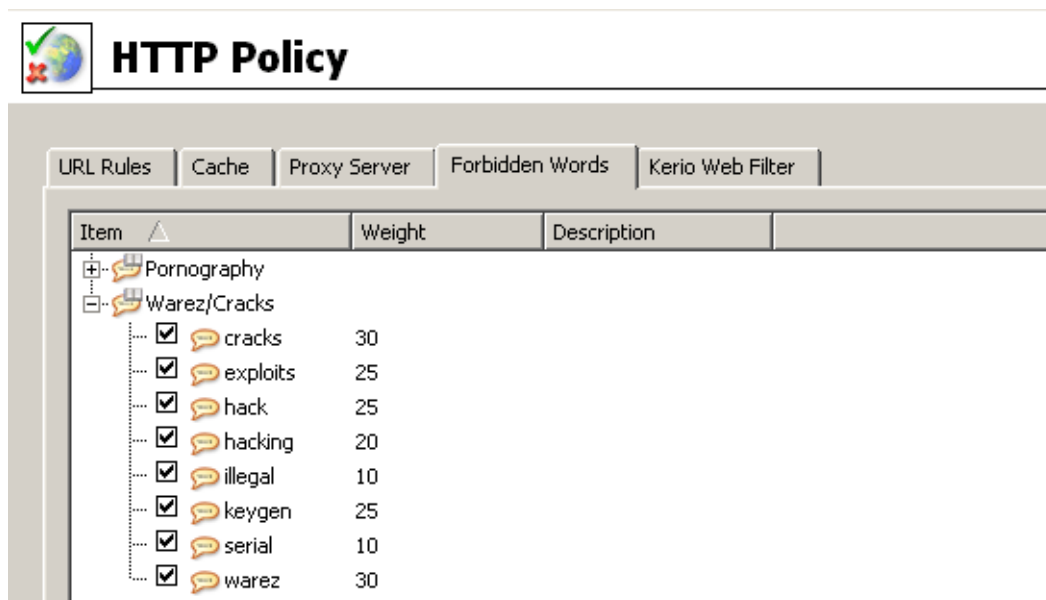


Figure 12.11 Groups of forbidden words

Individual groups and words included in them are displayed in form of trees. To enable filtering of particular words use checkboxes located next to them. Unchecked words 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 enable filtering of the particular words and modify the weight for each word.

Threshold value for Web page filtering

The value specified in *Deny pages with weight over* represents so called threshold weight value for each page (i.e. total weight of all forbidden words found 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).

Definition of forbidden words

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

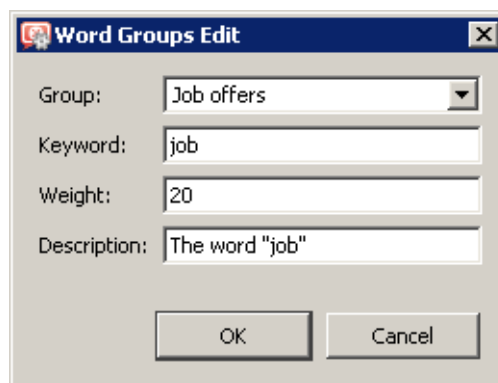


Figure 12.12 Definition of a forbidden word or/and a word group

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. This word can be in any language and it should follow the exact form in which it is used on websites (including diacritics and other special symbols and characters). If the word has various forms (declension, conjugation, etc.), it is necessary to define separate words for each word in the group. It is also possible to set various weight of words.

Weight

Word weight the level of how the word affects possible blocking or allowing of access to websites. The weight should respect frequency of the particular word in the language (the more common word, the lower weight) so that legitimate webpages are not blocked.

Description

A comment on the word or group.

12.5 FTP Policy

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

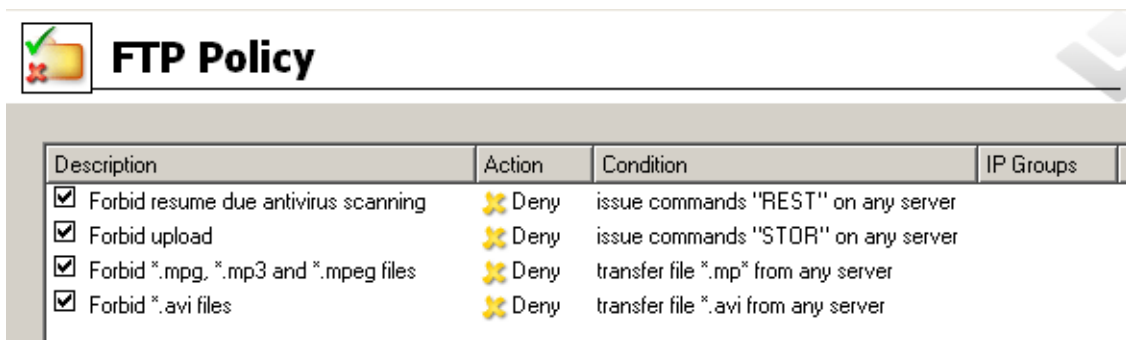


Figure 12.13 FTP Rules

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.

Note:

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.

For details on antivirus scan of FTP protocol, refer to chapter [13.3](#).

FTP 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.

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: FTP traffic which does not match any FTP rule is allowed (any traffic permitted by default). To allow accessing only a specific group of FTP servers and block access to other web pages, a rule denying access to all FTP servers must be placed at the end of the rule list.

FTP rule dialog:

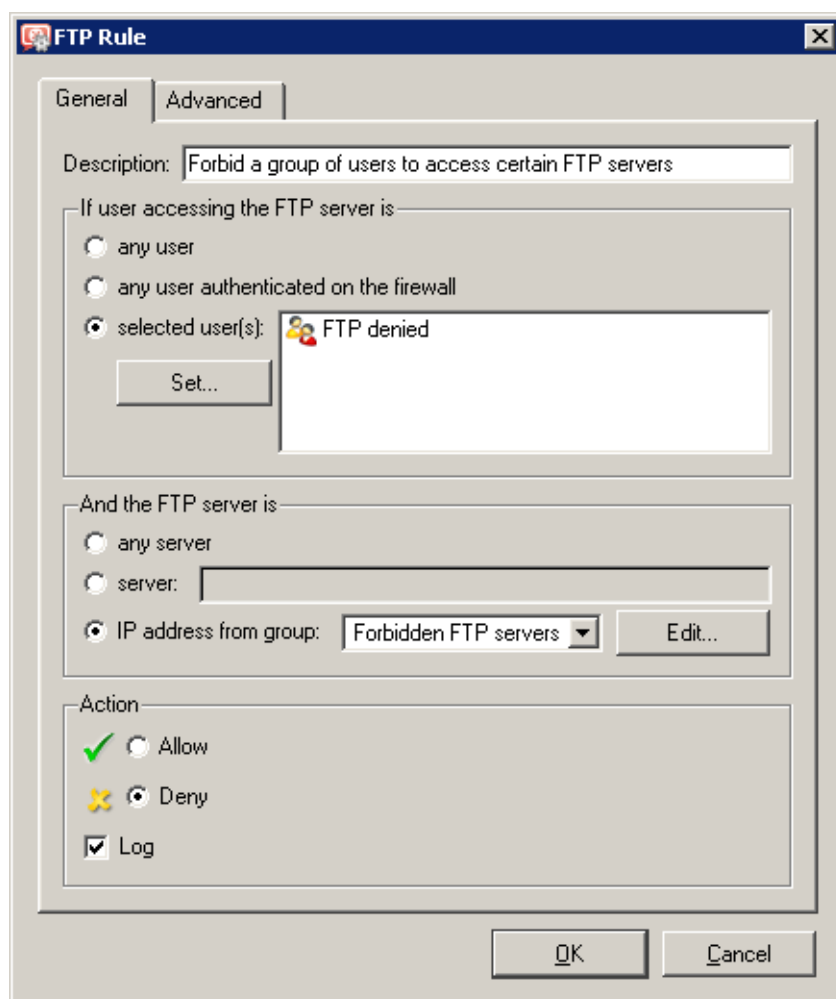


Figure 12.14 FTP Rule — basic parameters

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

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 than 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 or 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 [14.1](#)).

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).

Check the *Log* option to log all FTP connections meeting this rule in the *Filter* log (see chapter [22.9](#)).

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.

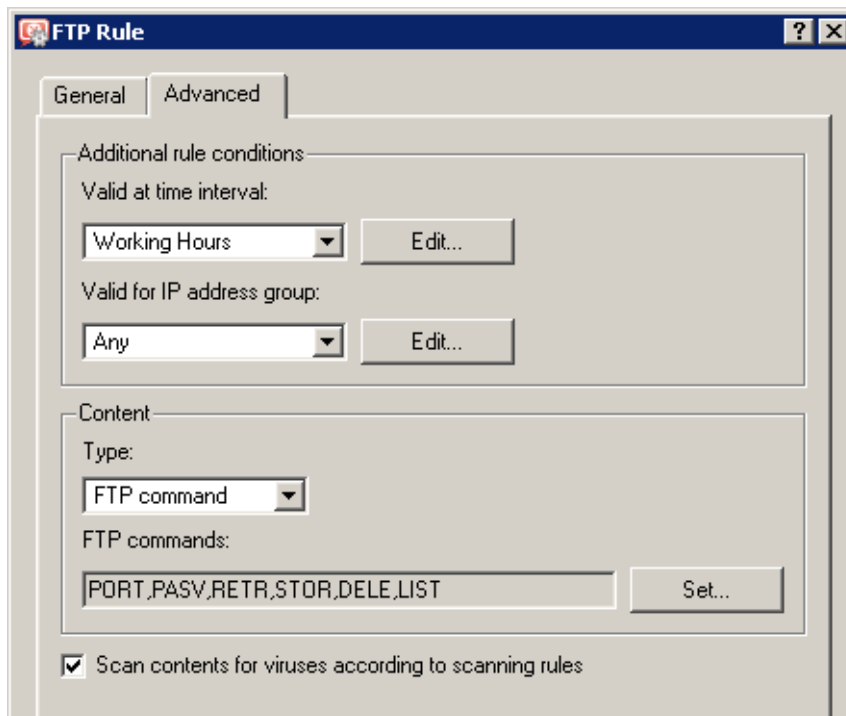


Figure 12.15 FTP Rule — advanced settings

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 [14.2](#)).

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.

Click on the *Edit* button to edit IP groups (for details see chapter [14.1](#)).

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.

Antivirus control

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.

WinRoute is also distributed in a special version which includes integrated *McAfee* antivirus. Besides the integrated module, *WinRoute* also supports many external antiviruses of third parties. 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).

WinRoute allows to use both the integrated *McAfee* antivirus and a selected external antivirus. In such a case, transferred files are checked by both antiviruses (so called dual antivirus control). This feature reduces the risk of letting in a harmful file.

However, using of two antiviruses at a time also decreases the speed of firewall's performance. It is therefore highly recommended to consider thoroughly which method of antivirus check should be used and to which protocols it should be applied and, if possible and desired, to try the configuration in the trial version of *WinRoute* before purchasing a license.

Note:

1. However, supported external antiviruses as well as versions and license policy of individual programs may change as the time flows. For up-to-date information please refer to (<http://www.kerio.com/firewall>).
2. External *McAfee Anti-Virus* programs are not supported by *WinRoute*.

13.1 Conditions and limitations of antivirus scan

Antivirus check of objects transferred by a particular protocol can be applied only to traffic where a corresponding protocol inspector which supports the antivirus is used (see chapter [14.3](#)). This implies that the antivirus check is limited by the following factors:

- Antivirus check cannot be used if the traffic is transferred by a secured channel (SSL/TLS). In such a case, it is not possible to decipher traffic and separate transferred objects.
- Within email antivirus scanning (SMTP and POP3 protocols), the firewall only removes infected attachments — it is not possible to drop entire email messages. In case of SMTP protocol, only incoming traffic is checked (i.e. traffic from the Internet to the local network — incoming email at the local SMTP server). Check of outgoing traffic causes problems with temporarily undeliverable email.

For details, see chapter [13.4](#).

- Object transferred by other than HTTP, FTP, SMTP and POP3 protocols cannot be checked by an antivirus.
- If a substandard port is used for the traffic, corresponding protocol inspector will not be applied automatically. In that case, simply define a traffic rule which will allow this traffic using a corresponding protocol inspector (for details, see chapter [7.3](#)).

Example: You want to perform antivirus checks of the HTTP protocol at port 8080.

1. Define the *HTTP 8080* service (TCP protocol, port 8080).
2. Create a traffic rule which will allow this service applying a corresponding protocol inspector.

Name	Source	Destination	Service	Action	Protocol Inspector
<input checked="" type="checkbox"/> HTTP 8080 with inspection	Trusted/Local	Internet	HTTP 8080	✓	HTTP

Figure 13.1 Traffic rule for HTTP protocol inspection at non-standard ports

Add the new rule before the rule allowing access to any service in the Internet (if such a rule exists). If the NAT (source address translation) technology is used for Internet connection, address translation must be set for this rule as well.

Note: A corresponding protocol inspector can be also specified within the service definition, or both definition methods can be used. Both methods yield the same result, however, the corresponding traffic rule is more transparent when the protocol inspector is defined in it.

13.2 How to choose and setup antiviruses

To select antiviruses and set their parameters, open the *Antivirus* tab in *Configuration* → *Content Filtering* → *Antivirus*. On this tab, you can select the integrated *McAfee* module, an external antivirus, or both.

If both antiviruses are used, each transferred object (downloaded file, an email attachment, etc.) will be first checked by the integrated *McAfee* antivirus module and then by the other antivirus (a selected external antivirus).

Integrated McAfee

To enable the integrated *McAfee* antivirus, enable *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 about the licensing policy, read chapter [44](#).

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

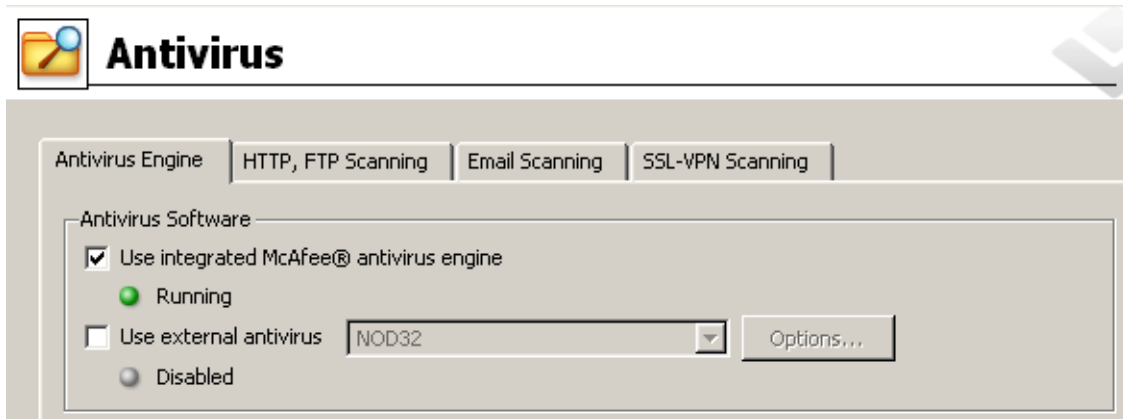


Figure 13.2 Antivirus selection (integrated antivirus)

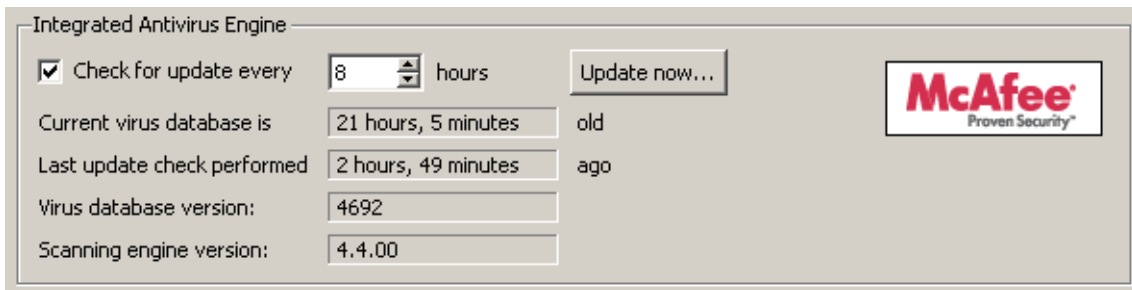


Figure 13.3 Scheduling McAfee updates

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 [22.8](#)).

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

— Warning —

To make the antivirus control as mighty as possible, it is necessary that the antivirus module is always equipped by the most recent version of the virus database. Therefore, it is recommended to keep automatic updates running and not to set too long intervals between update checks (update checks should be performed at least twice a day).

Current virus database is ...

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 ... ago

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 check 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.

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

External antivirus

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

— **Warning** —

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

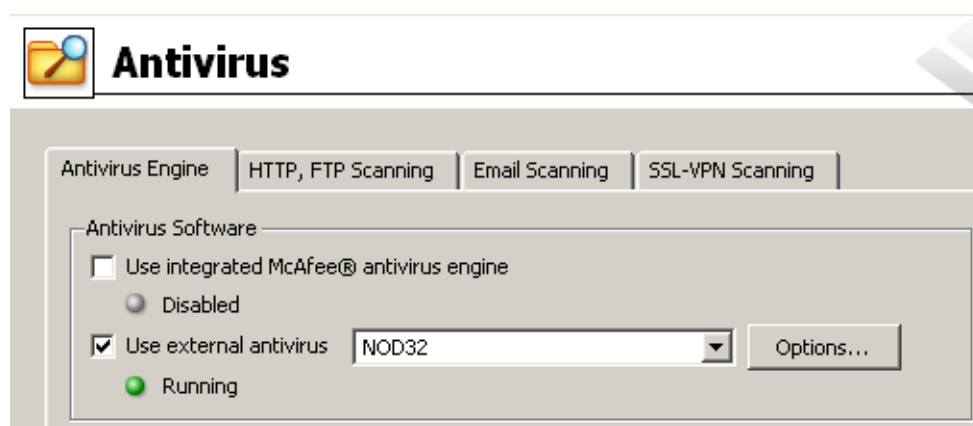


Figure 13.4 Antivirus selection (external antivirus)

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 on installation and configuration of individual antivirus programs, refer to <http://www.kerio.com/firewall/third-party>.

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 [22.8](#)).

Antivirus settings

Check items in the *Settings* section of the *Antivirus* tab to enable antivirus check for individual application protocols. By default, antivirus check is enabled for all supported modules.

In *Settings*, maximum size of files to be scanned for viruses at the firewall can be set. Scanning of large files are demanding for time, the processor and free disk space, which might affect the firewall's functionality dramatically. It might happen that the connection over which the file is transferred is interrupted when the time limit is exceeded.

The optimal value of the file size depends on particular conditions (the server's performance, load on the network, type of the data transmitted, antivirus type, etc.). *Caution! We strongly discourage administrators from changing the default value for file size limit. In any case, do not set the value to more than 4 MB.*

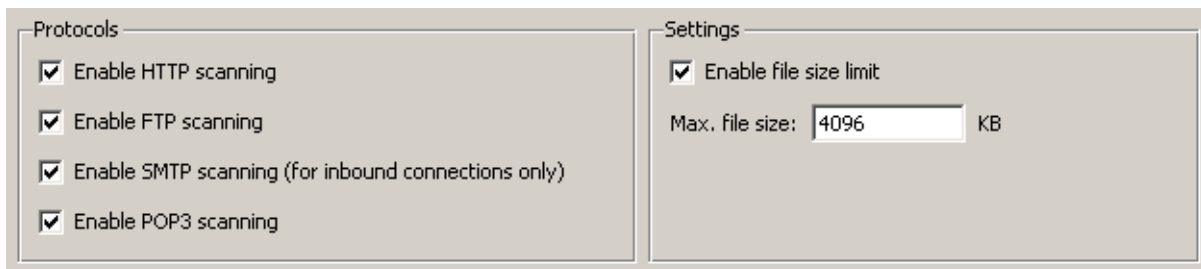


Figure 13.5 Selecting application protocols to be scanned and setting file size limits

Parameters for HTTP and FTP scanning can be set in the *HTTP and FTP scanning* (refer to chapter [13.3](#)), while SMTP and POP3 scanning can be configured in the *Email scanning* tab (see chapter [13.4](#)).

Warning

1. In case of SMTP protocol, only incoming traffic is checked (i.e. traffic from the Internet to the local network — incoming email at the local SMTP server). Checks of outgoing SMTP traffic (from the local network to the Internet) might cause problems with temporarily undeliverable email — for example in cases where the destination SMTP server uses so called *greylisting*.

To perform smooth checks of outgoing traffic, define a corresponding traffic rule using the SMTP protocol inspector. Such rule may be useful for example if clients in the local

network send their email via an SMTP server located in the Internet. Checking of outgoing SMTP traffic is not apt for local SMTP servers sending email to the Internet.

An example of a traffic rule for checking of outgoing SMTP traffic is shown at figure 13.6.


Name	Source	Destination	Service	Action	Translation	Protocol Inspector
<input checked="" type="checkbox"/> Outgoing SMTP 	 Trusted/Local	 smtp.server.com	 SMTP		NAT	SMTP

Figure 13.6 An example of a traffic rule for outgoing SMTP traffic check

2. Substandard extensions of the SMTP protocol can be used in case of communication of two *Microsoft Exchange* mailservers. Under certain conditions, email messages are transmitted in form of binary data. In such a case, *WinRoute* cannot perform antivirus check of individual attachments.

In such cases, it is recommended to use an antivirus which supports *Microsoft Exchange* and not to perform antivirus check of SMTP traffic of a particular server in *WinRoute*. To achieve this, disable antivirus check for SMTP protocol or define a corresponding traffic rule where no protocol inspector will be applied (see chapter 7.7).

13.3 HTTP and FTP scanning

As for HTTP and FTP traffic, objects (files) of selected types are scanned.

The file just transmitted is saved in a temporary file on the local disk of the firewall. *WinRoute* caches the last part of the transmitted file (segment of the data transferred) and performs an antivirus scan of the temporary file. If a virus is detected in the file, the last segment of the data is dropped. This means that the client receives an incomplete (damaged) file which cannot be executed so that the virus cannot be activated. If no virus is found, *WinRoute* sends the client the rest of the file and the transmission is completed successfully.

Optionally, a warning message informing about a virus detected can be sent to the user who tried to download the file (see the *Notify user by email* option).

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 12.2 and 12.5).
 3. Full functionality of HTTP scanning is not guaranteed if any non-standard extensions to web browsers (e.g. download managers, accelerators, etc.) are used!
-

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

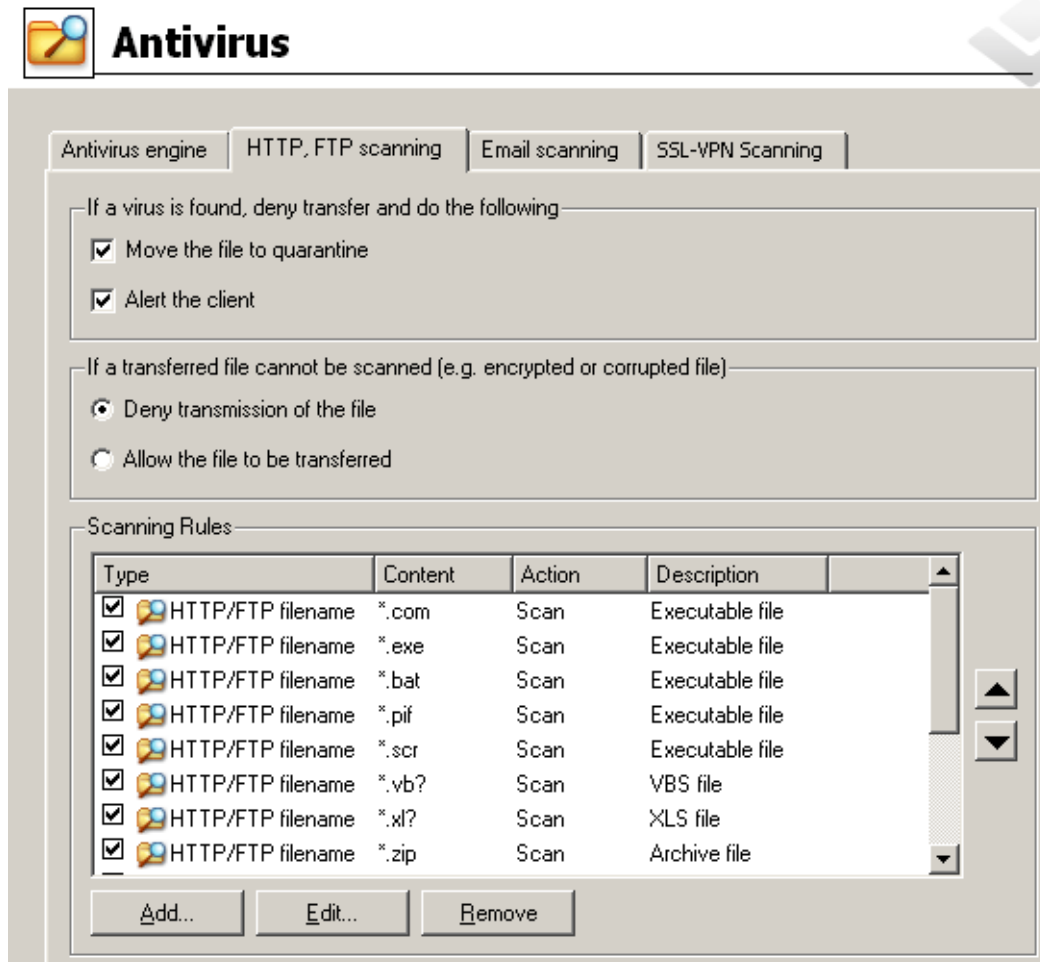


Figure 13.7 Settings for HTTP and FTP scanning

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 (the typical path is `C:\Program Files\Kerio\WinRoute Firewall\quarantine`) 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 warning that a virus was detected and download was stopped for security reasons.

WinRoute sends alert messages 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 15.1) and the SMTP server used for mail sending is configured correctly (refer to chapter 18.3).

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 19.4.

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.

Hint

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 in response to user requests.

- *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.

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.

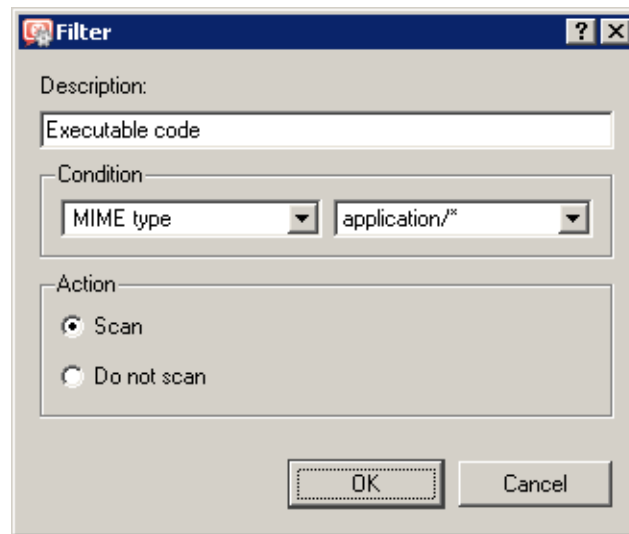


Figure 13.8 Definition of an HTTP/FTP scanning rule

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/*).

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.

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).

13.4 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.

Individual attachments of transmitted messages are saved in a temporary directory on the local disk. When downloaded completely, the files are scanned for viruses. If no virus is found, the attachment is added to the message again. If a virus is detected, the attachment is replaced by a notice informing about the virus found.

Note: Warning messages can also be sent to specified email addresses (e.g. to network administrators) when a virus is detected. For details, refer to chapter [19.4](#).

Warning

1. Antivirus control within WinRoute can only detect and block infected attachments. Attached files cannot be healed by this control!
2. Within antivirus scanning, it is possible to remove only infected attachments, entire email messages cannot be dropped. This is caused by the fact that the firewall cannot handle email messages like mailservers do. It only maintains network traffic coming through. In most cases, removal of an entire message would lead to a failure in communication with the server and the client might attempt to send/download the message once again. Thus, one infected message might block sending/reception of any other (legitimate) mail.
3. In case of SMTP protocol, only incoming traffic is checked (i.e. traffic from the Internet to the local network — incoming email at the local SMTP server). Checks of outgoing SMTP traffic (i.e. from the local network to the Internet) might cause problems with temporarily undeliverable email (for example in cases where the destination SMTP server uses so called *greylisting*).

To check also outgoing traffic (e.g. when local clients connect to an SMTP server without the local network), define a corresponding traffic rule using the SMTP protocol inspector. For details, see chapter [13.2](#).

Advanced parameters and actions that will be taken when a virus is detected can be set in the *Email scanning* tab.

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:

- *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.

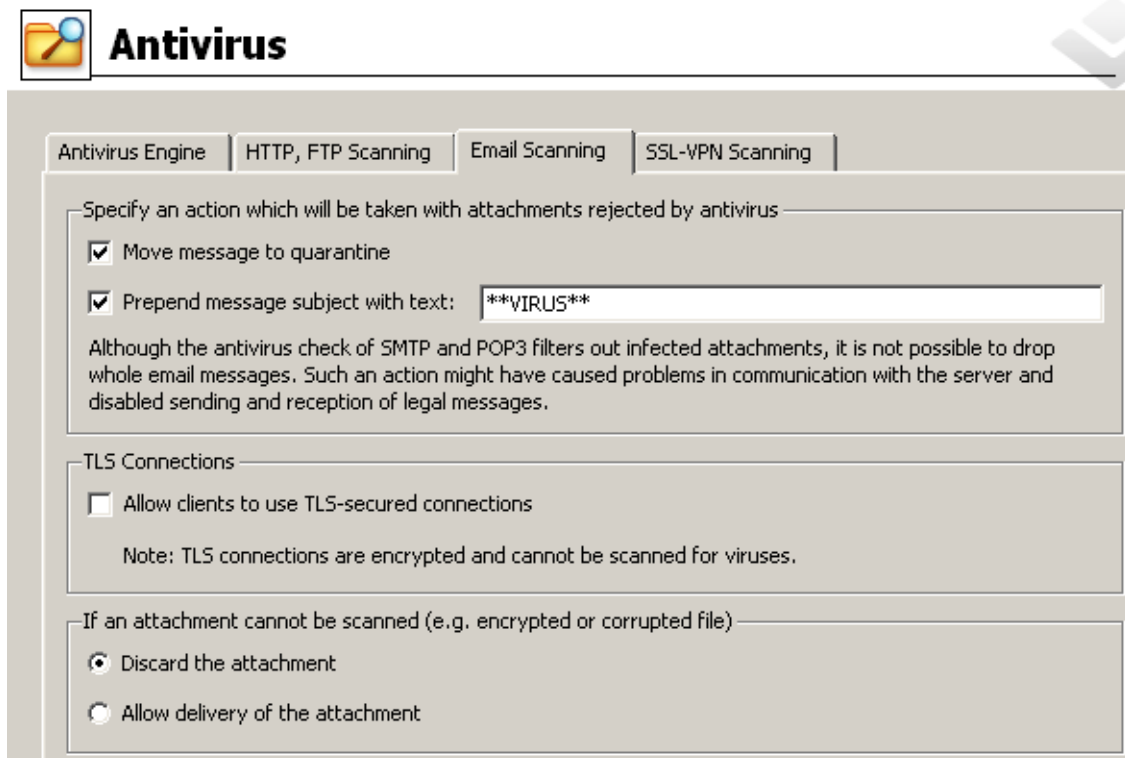


Figure 13.9 Settings for SMTP and POP3 scanning

The *quarantine* subdirectory under the *WinRoute* directory is used for the quarantine (the typical path is `C:\Program Files\Kerio\WinRoute Firewall\quarantine`) 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.

- *Prepend subject message with text* — use this option to specify a text to be attached before the subject of each email message where at least one infected attachment is found. This text informs the recipient of the message and it can be also used for automatic message filtering.

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 or POP3 traffic.

In case that TLS protocol is used, unencrypted 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 cannot 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.

Hint

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.

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 (typically password-protected archives) and the antivirus is installed on the workstations.

13.5 Scanning of files transferred via Clientless SSL-VPN (Windows)

If *WinRoute* is installed on *Windows*, the antivirus check is performed also for transfers of files between the local network and a remote client via *Clientless SSL-VPN* (see chapter 24). The *SSL-VPN Scanning* tab allows to set advanced parameters for scanning of files transferred via this interface. For the *Kerio WinRoute Firewall Software Appliance / VMware Virtual Appliance* administration, the *SSL-VPN Scanning* tab is not available.

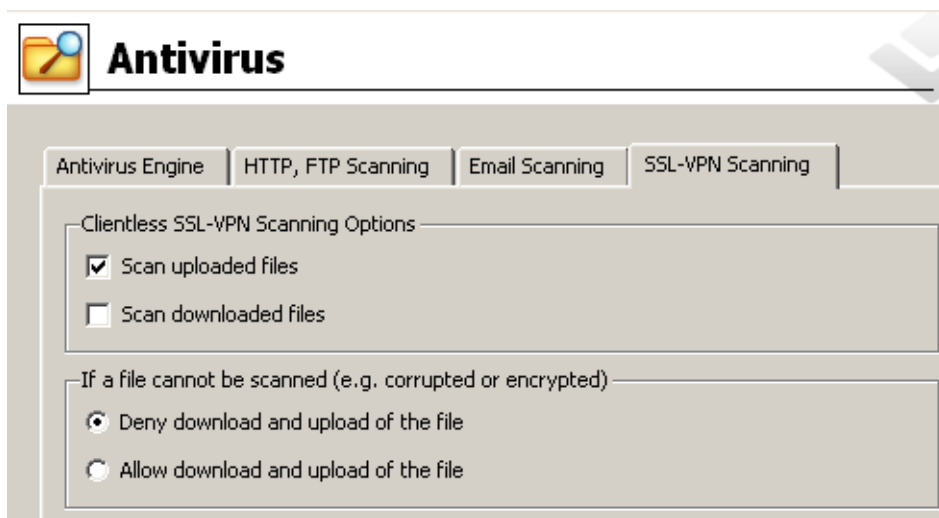


Figure 13.10 Settings for scanning of files transferred via Clientless SSL-VPN

Transfer directions

Use the top section of the *SSL-VPN Scanning* tab to set to which transfer direction the antivirus check will be applied. By default, only files downloaded from a remote client to a local host are scanned to avoid slowdown (local network is treated as trustworthy).

If the antivirus check fails

Options in the lower section of the tab specify an action which will be performed if a file cannot be scanned for any reason (encrypted or corrupted files, etc.). By default, transfer of such files is denied.

Definitions

14.1 IP Address Groups

IP groups are used for simple access to certain services (e.g. *WinRoute's* remote administration, Web server located in the local network available from the Internet, etc.). When setting access rights a group name is used. The group itself can contain any combination of computers (IP addresses), IP address ranges, subnets or other groups.

Creating and Editing IP Address Groups

You can define IP address groups in the *Configuration* → *Definitions* → *Address Groups* section.

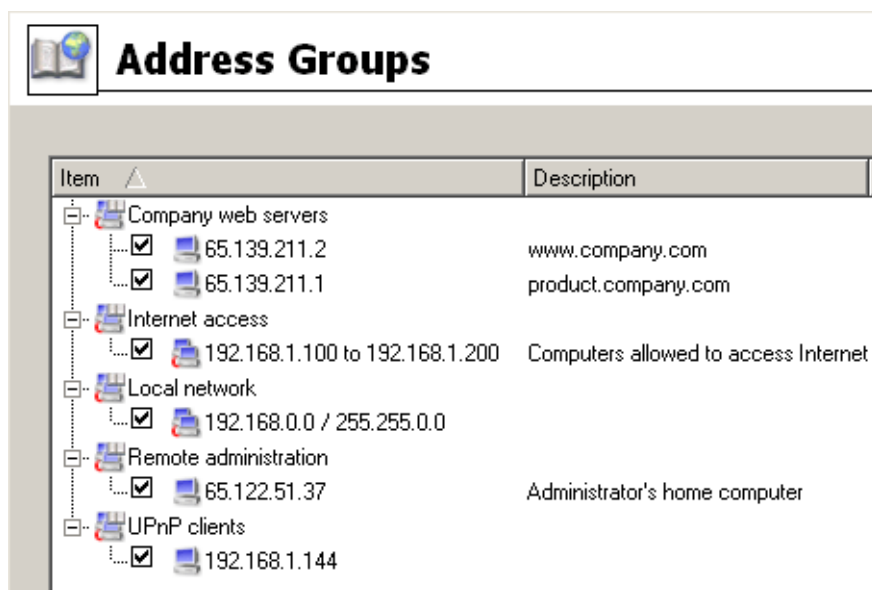


Figure 14.1 WinRoute's IP groups

Click on *Add* to add a new group (or an item to an existing group) and use *Edit* or *Delete* to edit or delete a selected group or item.

The following dialog window is displayed when you click on the *Add* button:

Name

The 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.

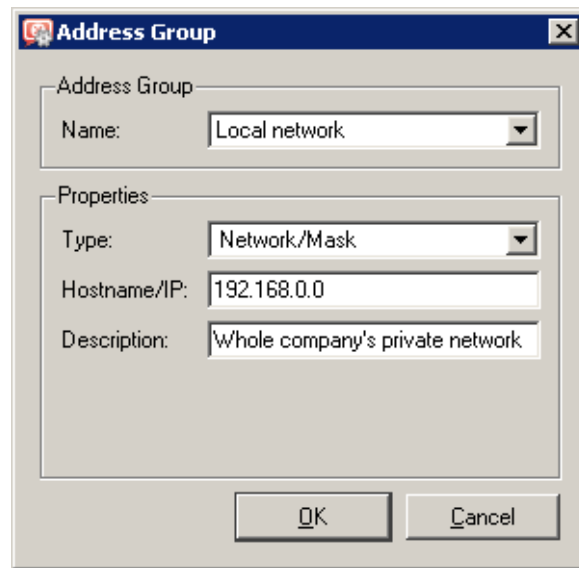


Figure 14.2 IP group definition

Type

Type of the new item:

- *Host* (IP address or DNS name of a particular host),
- *Network / Mask* (subnet with a corresponding mask),
- *IP range* (an interval of IP addresses defined by starting and end IP address including the both limit values),
- *Address group* (another group of IP addresses — groups can be cascaded),
- *Firewall* (a special group including all the firewall's IP addresses, see also chapter 7.3).

IP address, Mask...

Parameters of the new item (related to the selected type).

Description

Commentary for the IP address group. This helps guide the administrator.

Note: Each IP group must include at least one item. Groups with no item will be removed automatically.

14.2 Time Ranges

Time ranges in *WinRoute* are closely related to traffic policy rules (see chapter 7). *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 5.

To define time ranges go to *Configuration* → *Definitions* → *Time Ranges*.

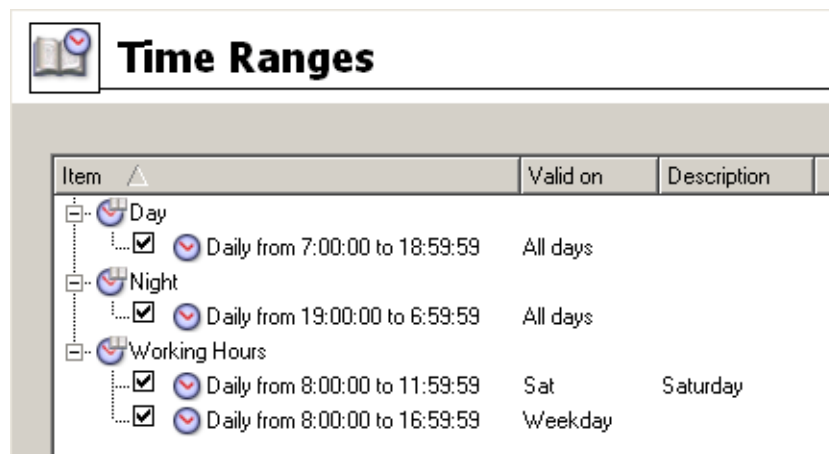


Figure 14.3 WinRoute's time intervals

Time range types

When defining a time interval three types of time ranges (subintervals) can be used:

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)

Defining Time Intervals

Time ranges can be created, edited and removed in *Configuration* → *Definitions* → *Time Ranges*.

Clicking on the *Add* button will display the following dialog window:

Name

Name (identification) of the time interval. 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

The beginning and the end of the time range. Beginning and end hours, days or dates can be defined according to the selected time range type

Figure 14.4 Time range definition

Valid on

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).


Note:

1. each time range must contain at least one item. Time ranges with no item will be removed automatically.
2. Time intervals cannot be cascaded.

14.3 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).

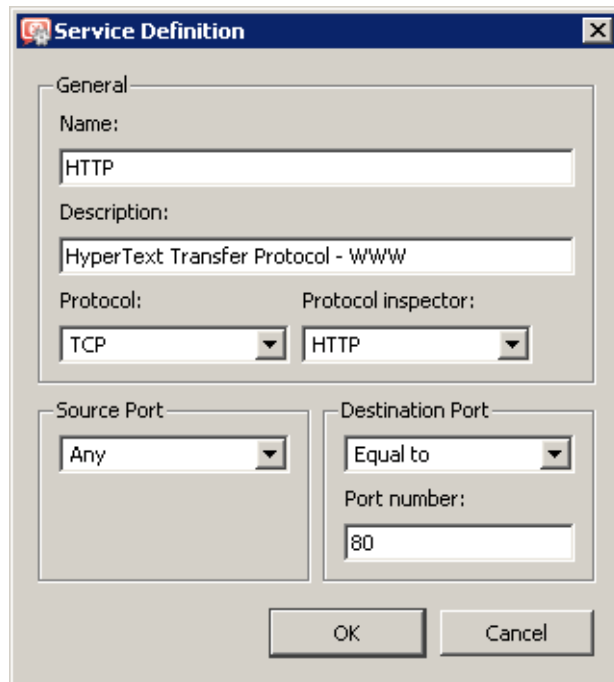
Services can be defined in *Configurations* → *Definitions* → *Services*. Some standard services, such as *HTTP*, *FTP*, *DNS* etc., are already predefined in the default *WinRoute* installation.

 **Services**

Name ▲	Protocol	Source port	Destination port	Protocol inspector	Description
H323	TCP	Any	1720		H.323 Protocol
HTTP	TCP	Any	80	HTTP	HyperText Transfer Protocol - WWW
HTTP 8080	TCP	Any	8080	HTTP	
HTTP Proxy	TCP	Any	3128		HTTP Proxy Server
HTTPS	TCP	Any	443		HyperText Transfer Protocol - Secured
ICQ	TCP	Any	5190		ICQ Instant Messaging
Ident	TCP	Any	113		Ident
IKE	UDP	Any	500		Internet Key Exchange
IMAP	TCP	Any	143		Internet Mail Access Protocol
IMAPS	TCP	Any	993		Internet Mail Access Protocol - Secured

Figure 14.5 WinRoute's network services

Clicking on the *Add* or the *Edit* button will open a dialog for service definition.



The dialog box titled "Service Definition" contains the following fields:

- General** section:
 - Name: HTTP
 - Description: HyperText Transfer Protocol - WWW
 - Protocol: TCP (dropdown)
 - Protocol inspector: HTTP (dropdown)
- Source Port** section:
 - Any (dropdown)
- Destination Port** section:
 - Equal to (dropdown)
 - Port number: 80

Buttons for OK and Cancel are located at the bottom.

Figure 14.6 Network service definition

Name

Service identification within *WinRoute*. It is strongly recommended to use a concise name to keep the program easy to follow.

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

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. Other options available are *ICMP* and *other*.

The *other* options allows protocol specification by the number in the IP packet header. Any protocol carried in IP (e.g. GRE — protocol number is 47) can be defined this way.

 A screenshot of a graphical user interface for defining a service. It features a 'Protocol:' label above a dropdown menu with 'other' selected. Below this is a 'Settings' section containing a 'Protocol number:' label and a text input field containing the number '47'.

Figure 14.7 Setting a protocol in service definition

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. Functionality of the service might be affected by using an inappropriate inspector.

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 details, refer to chapter [7.3](#).

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 commas (e.g. 80, 8000, 8080)

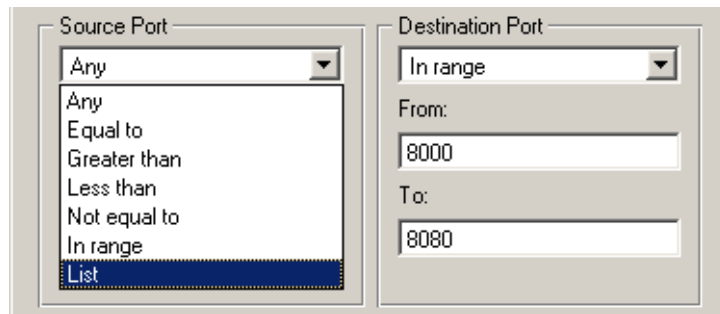


Figure 14.8 Service definition — source and destination port setting

Protocol Inspectors

WinRoute includes special subroutines 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).

The protocol inspector is enabled if it is set in the service definition and if the corresponding traffic is allowed. Each protocol inspector applies to a specific protocol and service. In the default *WinRoute* configuration, all available protocol inspectors are used in definitions of corresponding services (so they will be applied to corresponding traffic automatically), except protocol inspectors for *SIP* and *H.323* (*SIP* and *H.323* are complex protocols and protocol inspectors may work incorrectly in some configurations).

To apply a protocol inspector explicitly to another traffic, it is necessary to define a new service where this inspector will be used or to set the protocol inspector directly in the corresponding traffic rule.

Example

You want to perform inspection of the HTTP protocol at port 8080. Define a new service: TCP protocol, port 8080, HTTP protocol inspector. This ensures that *HTTP* protocol inspector will be automatically applied to any *TCP* traffic at port 8080 and passing through *WinRoute*.

Note:

1. Generally, protocol inspectors cannot be applied to secured traffic (SSL/TLS). In this case, *WinRoute* “perceives” the traffic as binary data only. This implies that such traffic cannot be deciphered.
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 [7.7](#).

14.4 URL Groups

URL Groups enable the administrator to define HTTP rules easily (see chapter [12.2](#)). 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. A URL group rule is processed significantly faster than a greater number of separate rules for individual URLs. It is also possible to cascade URL groups.

URL groups can be defined in *Configuration* → *Definitions* → *URL Groups*.

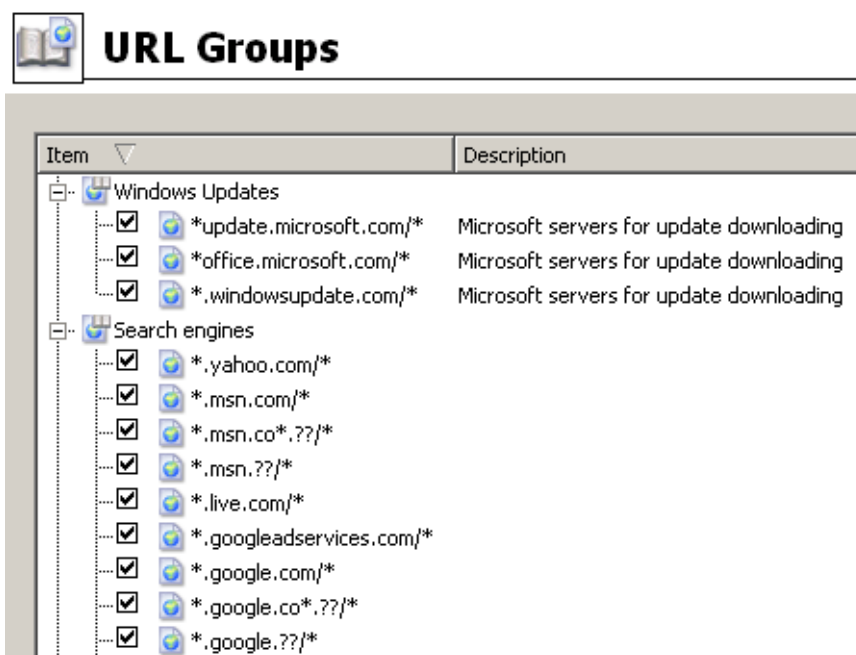


Figure 14.9 URL Groups

The default *WinRoute* installation already includes predefined URL groups:

- *Ads/Banners* — common URLs of pages that contain advertisements, banners, etc.
- *Search engines* — top Internet search engines.
- *Windows Updates* — URL of pages requested for automatic updates of Windows.

These URL groups are used in predefined URL rules (see chapter [12.2](#)). *WinRoute* administrators can use predefined groups in their custom rules or/and edit them if needed.

Matching fields next to each item of the group can be either checked to activate or unchecked to disable the item. This way you can deactivate items with no need to remove them and to define them again.

Click on the *Add* button to display a dialog where a new group can be created or a new item can be added to existing groups.

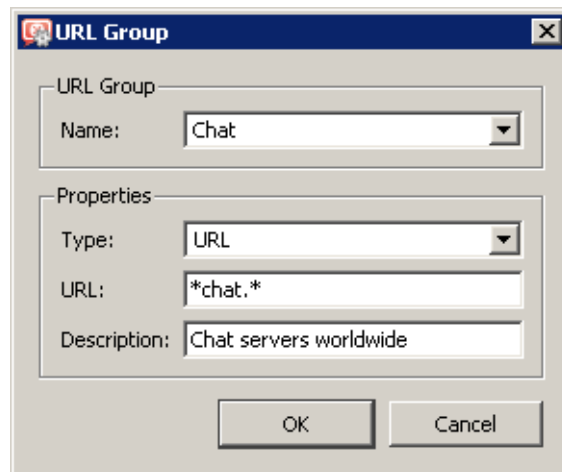


Figure 14.10 URL group definition

Name

Name of the group in which the new item will be added. Options of the *Name* entry are as follows:

- select a group to which the URL will be added,
- add a name to create a new group where the item will be included.

Type

Type of the item — URL or URL group (groups can be cascaded).

URL / URL Group

URL or URL group that will be added to the group (depending on the item type).

URL can be specified as follows:

- 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.com/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 item's description (comments and notes for the administrator).

User Accounts and Groups

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 *Administration Console* or the *Web Administration* interface.

WinRoute supports several methods of user accounts and groups saving, combining them with various types of authentication, as follows:

Internal user database

User accounts and groups and their passwords are saved in *WinRoute*. 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 within the domain

User accounts are stored in *WinRoute*. However, users are authenticated at *Windows NT* or *Active Directory* domain (i.e. password is not stored in the user account in *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.

Import of user accounts from Active Directory

If *Active Directory* (*Windows 2000 Server* or *Windows Server 2003/2008*) is used, automatic import of user accounts from it can be enabled. It is not necessary to define accounts in *WinRoute*, nor import them, since it is possible 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 automatically imported upon the first login of the user to *WinRoute*. Parameters set by using a template can be modified for individual accounts if necessary.

Note: This type of cooperation with *Active Directory* applies especially to older versions of *WinRoute* and makes these versions still compatible. In case of the first installation of *WinRoute*, it is recommended to apply transparent cooperation with *Active Directory*.

Transparent cooperation with Active Directory (Active Directory mapping)

WinRoute can use accounts and groups stored in *Active Directory* directly — no import to the local database is performed. Specific *WinRoute* parameters are added by the template of the corresponding account. These parameters can also be edited individually.

This type is the least demanding from the administrator's point of view (all user accounts and groups are managed in *Active Directory*) and it is the only one that allows using accounts from multiple *Active Directory* domains.

Note: In cases when users are authenticated at the domain (all described types excluding the first one), it is recommended to create at least one local account in *WinRoute* that has both read and write rights, or keep the original *Admin* account. This account provides connection to the *WinRoute* administration in case of the network or domain server failure.

15.1 Viewing and definitions of user accounts

To define local user accounts, import accounts to the local database or/and configure accounts mapped from the domain, go to the *User Accounts* tab in the *Users and Groups* → *Users* section.

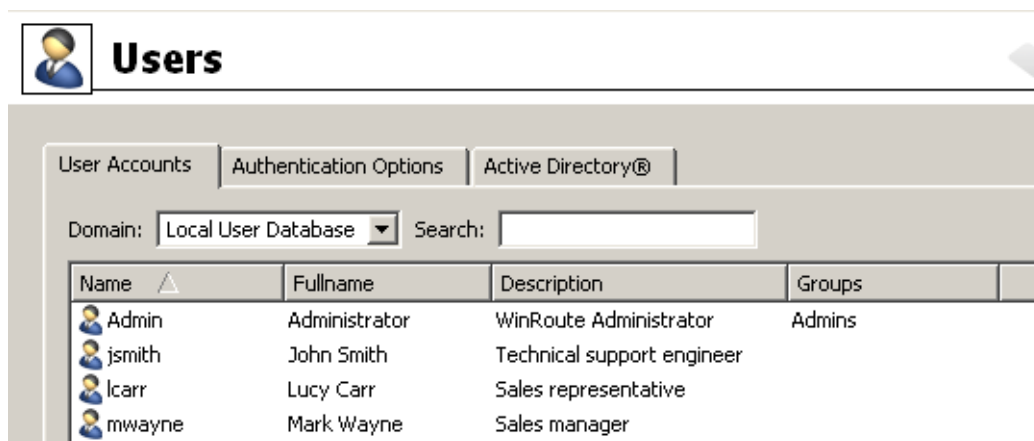


Figure 15.1 WinRoute user accounts

Domain

Use the *Domain* option to select a domain for which user accounts as well as other parameters will be defined. This item provides a list of mapped *Active Directory* domains (see chapter [15.4](#)) and the local (internal) user database.

Search

The *Search* engine can be used to filter out user accounts meeting specified criteria. The searching is interactive — each symbol typed or deleted defines the string which is evaluated immediately and all accounts including the string in either *Name*, *Full name* or *Description* are viewed. The icon next to the entry can be clicked to clear the filtering string and display all user accounts in the selected domain (if the *Search* entry is blank, the icon is hidden).

The searching is helpful especially when the domain includes too many accounts which might make it difficult to look up particular items.

Hiding / showing disabled accounts

It is possible to disable accounts in *WinRoute*. Check the *Hide disabled user accounts* to show only active (enabled) accounts.

Account template

Parameters shared by the most accounts can be defined by a template. Templates simplify administration of user accounts — shared parameters are set just once, when defining the template. It is also possible to configure some accounts (such as administrator accounts) separately, without using the template.

Templates apply to specific domains (or to the local user database). Each template includes parameters of user rights, data transfer quota and rules for content rules (for detailed description of all these parameters, refer to chapter [15.2](#)).

Local user accounts

If the *Local user database* is selected in the *Domain* item, user accounts in *WinRoute* are listed (complete information on these accounts are stored in the *WinRoute* configuration database). The following options are available for accounts in the local database:

Add, Edit, Remove

Click *Add*, *Edit* or *Remove* to create, modify or delete local user accounts (for details, see chapter [15.2](#)). It is also possible to select more than one account by using the **Ctrl** and **Shift** keys to perform mass changes of parameters for all selected accounts.

Importing accounts from a domain

Accounts can be imported to the local database from the *Windows NT* domain or from *Active Directory*. Actually, this process includes automatic copying of domain accounts (account authenticating at the particular domain) to newly created local accounts. For detailed information about import of user accounts, refer to chapter [15.3](#).

Import of accounts is recommended in case of the *Windows NT* domain. If *Active Directory* domain is used, it is recommended to use the transparent cooperation with *Active Directory* (domain mapping — see chapter [15.4](#)).

Accounts mapped from the Active Directory domain

If any of the *Active Directory* domain is selected as *Domain*, user accounts in this domain are listed.

Edit User

For mapped accounts, specific *WinRoute* parameters can be set (refer to chapter [15.2](#)). These settings are stored in the *WinRoute*'s configuration database. Information stored in *Active Directory* (username, full name, email address) and authentication method cannot be edited.

Note: It is also possible to select more than one account by using the **Ctrl** and **Shift** keys to perform mass changes of parameters for all selected accounts.

In mapped *Active Directory* domains, it is not allowed to create or/and remove user accounts. These actions must be performed in the *Active Directory* database on the relevant domain server. It is also not possible to import user accounts — such an action would take no effect in case of a mapped domain.

15.2 Local user accounts

Local accounts are accounts created in *WinRoute* or imported from a domain. These accounts are stored in the *WinRoute* configuration database (see chapter [25.2](#)). These accounts can be useful especially in domainless environments or for special purposes (typically for the firewall's administration).

Regardless on the method used for creation of the account, each user can be authenticated through the *WinRoute's* internal database, *Active Directory* or *Windows NT* domain.

The basic administrator account (*Admin*) 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.

Warning

1. All passwords should be kept safe and secret, otherwise they might be misused by an unauthorized person.
 2. If all accounts with full administration rights are removed and you logout from the *WinRoute* administration, it is not possible to connect to the *WinRoute* administration any longer. Under these conditions, a local user account (*Admin* with a blank password) will be created automatically upon the next start of the *WinRoute Firewall Engine*.
 3. Provided that you forget your administration password, contact the *Kerio Technologies* technical support (see chapter [26](#)).
-

Creating a local user account

Open the *User Accounts* tab in the *User and groups* → *Users* section. In the *Domain* combo box, select *Local User Database*.

Click on the *Add* button to open a guide to create a new user account.

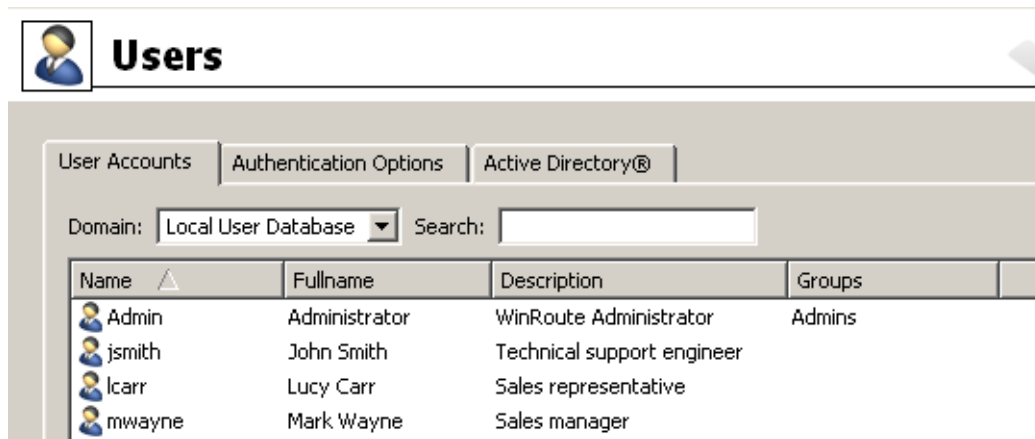


Figure 15.2 Local user accounts in WinRoute

Step 1 — basic information

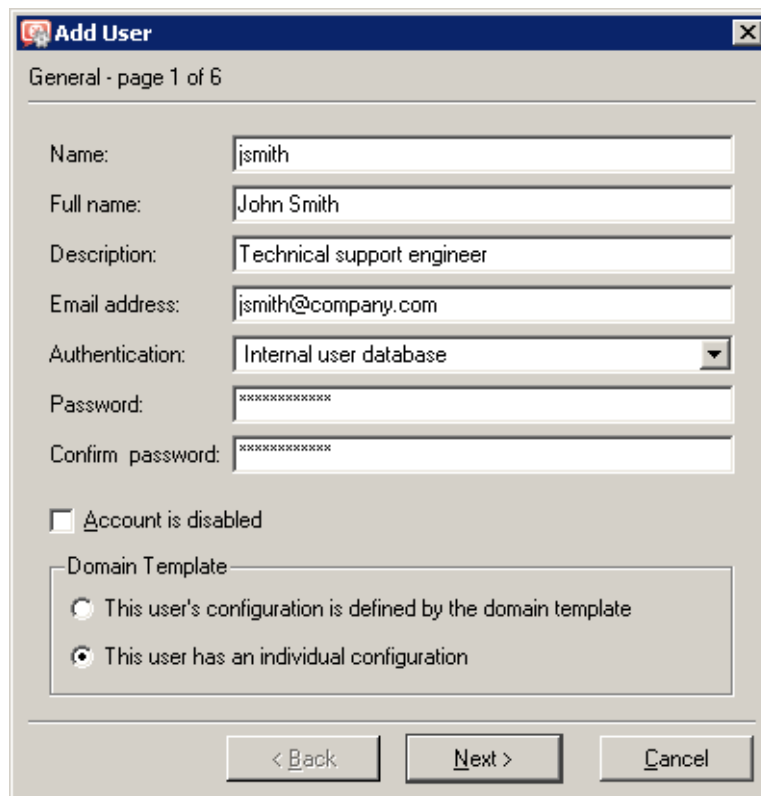


Figure 15.3 Creating a user account — basic parameters

Name

Username used for login to the account.

Warning

The user name is not case-sensitive. We recommend not to use special characters (non-English languages) which might cause problems when authenticating at the firewall's web interfaces.

Full name

A full name of the user (usually first name and surname).

Description

User description (e.g. a position in a company).

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 [19.4](#)) and other information (e.g. alert 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, refer to chapter [18.3](#).

Authentication

User authentication (see below)

Account is disabled

Temporary blocking of the account so that you do not have to remove 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).

Domain template

Define parameters for the corresponding user account (access rights, data transfer quotas and content rules). These parameters can be defined by the template of the domain (see chapter [15.1](#)) or they can be set especially for the corresponding account.

Using a template is suitable for common accounts in the domain (common user accounts). Definition of accounts is simpler and faster, if a template is used.

Individual configuration is recommended especially for accounts with special rights (e.g. *WinRoute* administration accounts). Usually, there are not many such accounts which means their configuration comfortable.

Authentication options:

Internal user database

User account information is stored locally to *WinRoute*. In such a case, specify the *Password* and *Confirm password* items (later, the password can be edited in the Web interface — see the *Kerio WinRoute Firewall — User's Guide*).

Warning

1. Passwords may contain printable symbols only (letters, numbers, punctuation marks). Password is case-sensitive. We recommend not to use special characters (non-English languages) which might cause problems when authenticating via the Web interface.
2. NTLM authentication cannot be used for automatic authentication method by NTLM (refer to chapter [25.3](#)). These accounts also cannot be used for authentication to the *Clientless SSL-VPN* interface (see chapter [24](#)).

NT domain / Kerberos 5

Users are authenticated through the *Windows NT* domain (*Windows NT 4.0*) or through the *Active Directory* (*Windows 2000/2003/2008*).

Go to the *Users* section of the *Active Directory / NT domain* tab to set parameters for user authentication through the *Windows NT* domain or/and through the *Active Directory*. If *Active Directory* authentication is set also for *Windows NT* domain, then *Active Directory* will be preferred.

Note: User accounts with this type of authentication set will not be active unless authentication through *Active Directory* or/and *NT domain* is enabled. For details, see chapter [15.3](#).

Step 2 — groups

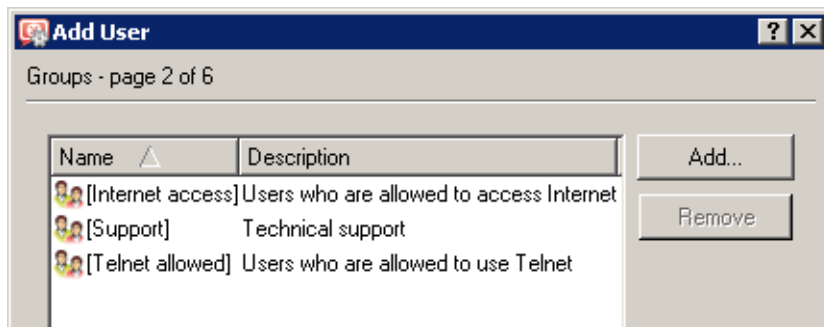


Figure 15.4 Creating a new user account — groups

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 [15.5](#)). Follow the same guidelines to add users to groups during group definition. It is not important whether groups or users are defined first.

Hint

While adding new groups you can mark more than one group by holding either the *Ctrl* or the *Shift* key.

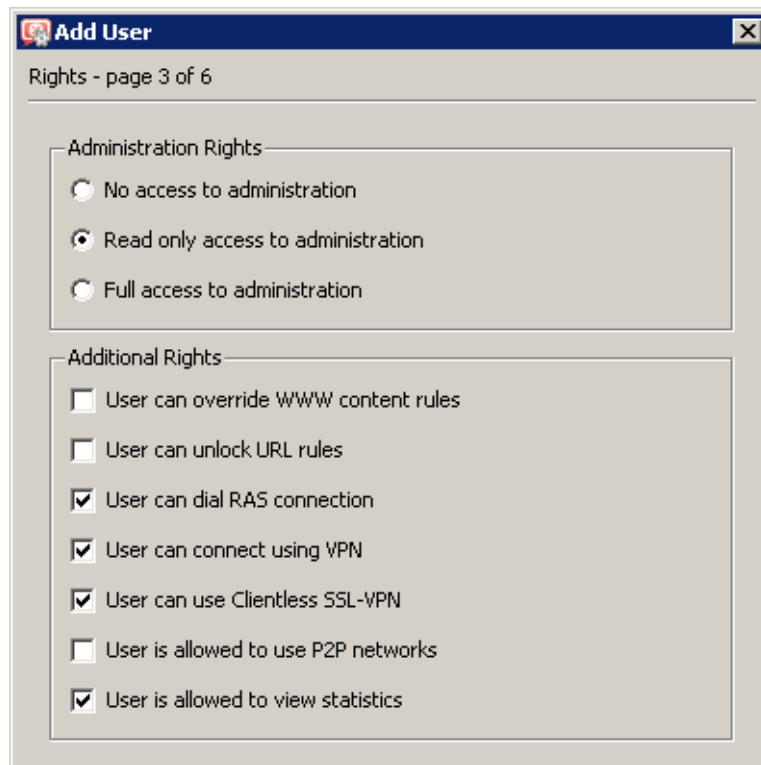
Step 3 — access rights

Figure 15.5 Creating a new user account — user rights

Each user must be assigned one of the following three levels 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

These users have full rights to administration and are equal to the Admin account. If there is at least one user with the full access to the administration, the default Admin account can be removed.

Additional rights:

User can override WWW content rules

User can customize personal web content filtering settings independently of the global configuration (for details, refer to *Step 5*).

User can unlock URL rules

If this option is checked, the user is allowed to bypass the rule denying access to the queried website — at the page providing information about the denial, the *Unlock* button

is displayed. The unlock feature must also be enabled in the corresponding URL rule (for details, refer to chapter [12.2](#)).

User can dial RAS connection

If the Internet connection uses dial-up lines, users with this right will be allowed to dial and hang up these lines in the Web interface (see chapter [11](#)).

User can connect using VPN

The user is allowed to connect through *WinRoute's* VPN server (using *Kerio VPN Client*). For detailed information, see chapter [23](#).

User can use Clientless SSL-VPN

The user will be allowed to access shared files and folders in the local network via the *Clientless SSL-VPN* web interface.

The *Clientless SSL-VPN* interface and the corresponding user right in *WinRoute* is available for *Windows* only. For details, see chapter [24](#).

User is allowed to use P2P networks

Traffic of this user will not be blocked if *P2P (Peer-to-Peer)* networks are detected. For details, see chapter [17.1](#).

User is allowed to view statistics

This user will be allowed to view firewall statistics in the web interface (see chapter [11](#)).

— Hint —

Access rights can also be defined by a user account template.

Step 4 — data transmission quota

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

Setting of daily, weekly and monthly limit of volume of transferred data for the user.

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.

Quota exceed action

Set actions which will be taken whenever a quota is exceeded:

- *Block any further traffic* — 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.)
- *Don't block further traffic (Only limit bandwidth...)* — Internet connection speed (so called bandwidth) will be limited for the user. Traffic will not be blocked but the user will notice that the Internet connection is slower than usual (this should

The screenshot shows the 'Add User' dialog box, specifically the 'Quota' page (page 4 of 6). The 'Transfer quota' section is expanded and contains three checked options:

- Enable daily limit
 - Direction: download
 - Quota: 100 MB
- Enable weekly limit
 - Direction: download
 - Quota: 300 MB
- Enable monthly limit
 - Direction: all traffic
 - Quota: 1 GB

The 'Quota exceed action' section contains:

- Block any further traffic
- Don't block further traffic (Only limit bandwidth according to Bandwidth Limiter settings.)
- Notify user by email when quota is exceeded

At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Figure 15.6 Creating a new user account — data transmission quota

make such users to reduce their network activities). For detailed information, see chapter 9.

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 18.3). If you wish that your *WinRoute* administrator is also notified when a quota is almost exceeded, set the alert parameters in *Configuration* → *Accounting*. For details, refer to chapter 19.4.

Note:

1. If a quota is exceeded and the traffic is blocked in result, 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

Don't block further traffic mode

- resetting of the data volume counter of the user (see chapter 20.1).
2. Actions for quota-exceeding are not applied if the user is authenticated at the firewall. This would block all firewall traffic as well as all local users. However, transferred data is included in the quota!

Hint

Data transfer quota and actions applied in response can also be set by a user account template.

Step 5 — web content rules and language preferences

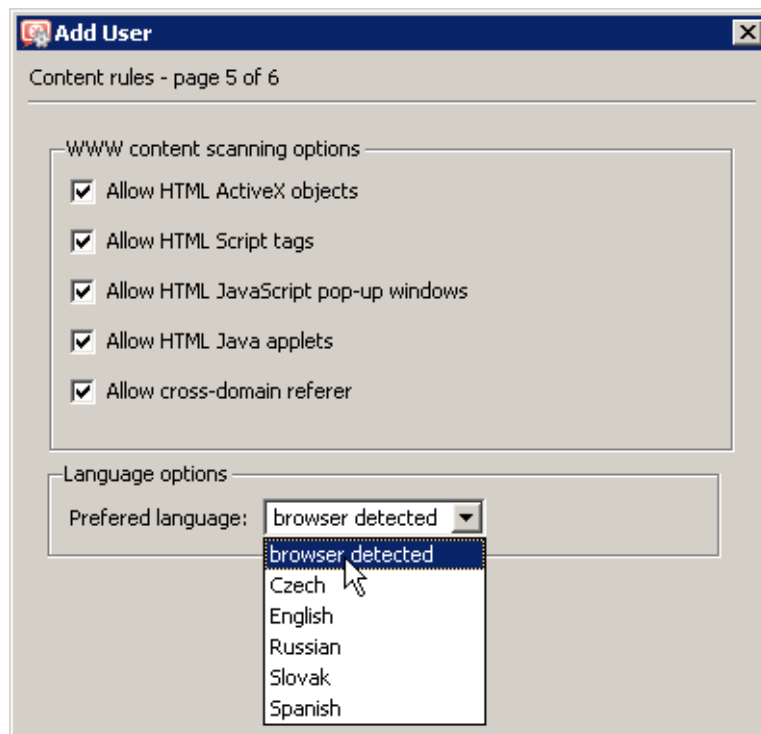


Figure 15.7 Creating a new user account — Web site content rules

In the *WWW content scanning options* section, special content filter rules settings for individual users can be defined. By default, all elements are allowed. *WinRoute* allows to block the following web elements:

ActiveX objects

Active objects at web pages. This option allows/blocks `<object>` and `<embed>` HTML tags.

<Script> HTML tags

The executive code in *JavaScript*, *VBScript*, etc.

Pop-up windows

Automatic opening of new browser windows — usually pop-up windows with advertisements.

This option will allow / block the *window.open()* method in *JavaScript*.

<Applet> HTML tags

Applets in *Java*.

Cross-domain referers

This option allows / blocks the Referer item included in an *HTTP* header.

The Referer item includes pages that have been viewed prior to the current page. This option allows to block Referer in case that it includes a server name different from the one defined in the particular HTTP request.

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

The *Language options* section allows setting of preferred language of the *WinRoute's* web interface (including the *Kerio StaR* interface). The *browser detected* option sets preferred language in accordance with settings in user's web browser and uses the language with the highest preference rate available. English will be used if none of other preferred languages is available.

Preferred language also applies to email alerts sent by the firewall (notices of reaching of data transfer quota, detected viruses, detected P2P networks, etc.). If language is detected and set by using user's web browser preferences, language set as preferred for the previous user's login to the web interface will be used. If the user has not logged into the web interface before, alerts will be in English.

Note: These settings can be customized at a corresponding page of the *WinRoute's* Web interface (see *Kerio WinRoute Firewall — User's Guide*). If the user can override content rules, any changes can be made. 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). Language preferences can always be changed.

Hint

Content rules can also be defined by a user account template.

Step 6 — user's IP addresses

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 a particular IP address. When a user account is being created, *WinRoute* automatically detects whether the specified IP address is used for automatic login or not.

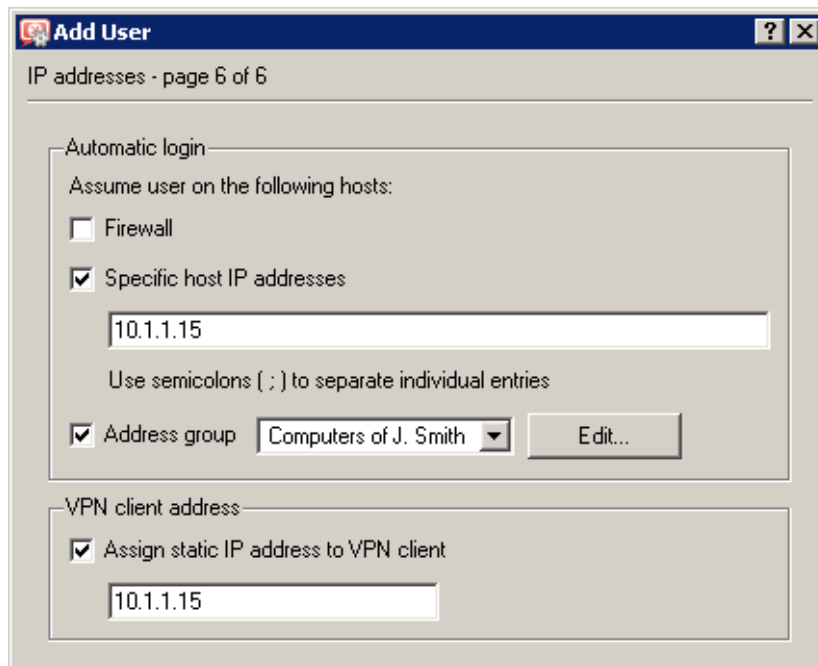


Figure 15.8 Creating a new user account — IP addresses for VPN client and automatic logins

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.). An IP address group can be used for specification of multiple hosts (refer to chapter [14.1](#)).

— **Warning** —

Automatic login decreases user's security. If an unauthorized user works on the computer for which automatic login is enabled, he/she uses the identity of the host's user who is authenticated automatically. Therefore, automatic login should be accompanied by another security feature, such as by user login to the operating system.

IP address which will be always assigned to the VPN client of the particular user can be specified under *VPN client address*. Using this method, a fixed IP address can be assigned to a user when he/she connects to the local network via the *Kerio VPN Client*. It is possible to add this IP to the list of IP addresses from which the user will be authenticated automatically.

For detailed information on the *Kerio Technologies'* proprietary VPN solution, refer to chapter [23](#).

Editing User Account

The *Edit* button opens a dialog window where you can edit the parameters of the user account. This dialog window contains all of the components of the account creation guide described above, divided into tabs in one window.

15.3 Local user database: external authentication and import of accounts

User in the local database can be authenticated either at the *Active Directory* domain or at the *Windows NT* domain (see chapter [15.2](#), step one). To apply these authentication methods, the *WinRoute* host must belong to the corresponding domain.

If *WinRoute* is installed on *Windows*, the host can be added to the domain or domain membership can be changed only in the operating system (in the computer properties).

In the *Software Appliance / VMware* appliance edition, domain membership can be set right in the firewall's administration:

- in the *Web Administration* interface, section *Domains and authentication*, the *Active Directory* tab.
- the *Administration Console*, section *Users*, the *Active Directory* tab.

WinRoute in *Software Appliance / VMware Virtual Appliance* can be connected only to the *Active Directory* domain, never to the *Windows NT* domain.

Importing user accounts

To the local user database, you can import selected accounts from the *Active Directory* or the *Windows NT* domain (import from *Windows NT* is available only in *WinRoute* on *Windows*).

Each import of a user account covers creating of a local account with the identical name and the same domain authentication parameters. Specific *WinRoute* parameters (such as access rights, content rules, data transfer quotas, etc.) can be set by using the template for the local user database (see chapter [15.1](#)) or/and they can be defined individually for special accounts. The *Active Directory / Windows NT* authentication type is set for all accounts imported..

Note: This method of user accounts import is recommended especially when *Windows NT* domain is used (domain server with the *Windows NT Server* operating system). If *Active Directory* domain is used, it is easier and recommended to use the transparent support for *Active Directory* (domain mapping — see chapter [15.4](#)).

To import user accounts, click on the *Import* button below the list of user accounts (as *Domain, Local user database* must be used, otherwise the button is inactive).

In the import dialog, select the type of the domain from which accounts will be imported and, with respect to the domain type, specify the following parameters:

- *Active Directory* — for import of accounts, *Active Directory* domain name, DNS name or IP address of the domain server as well as login data for user database reading (any account belonging to the domain) are required.
- *NT domain* — domain name is required for import. The *WinRoute* host must be a member of this domain.

Note: Import of user accounts from *Windows NT* is available only in *WinRoute* on *Windows*.

When connection with the corresponding domain server is established successfully, all accounts in the selected domain are listed. When accounts are selected and the selection is confirmed, the accounts are imported to the local user database.

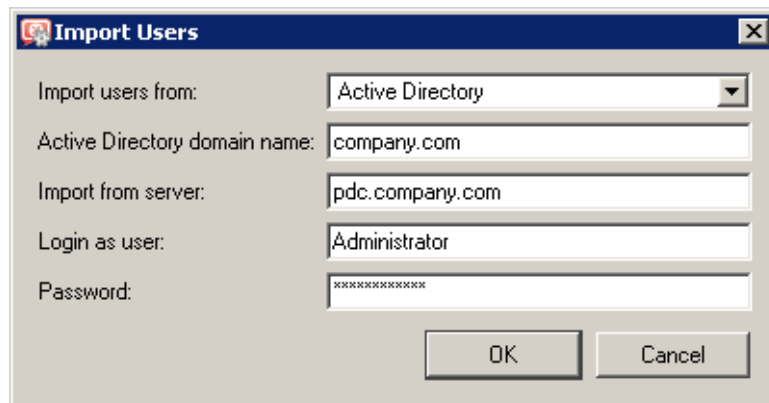


Figure 15.9 Import of accounts from Active Directory

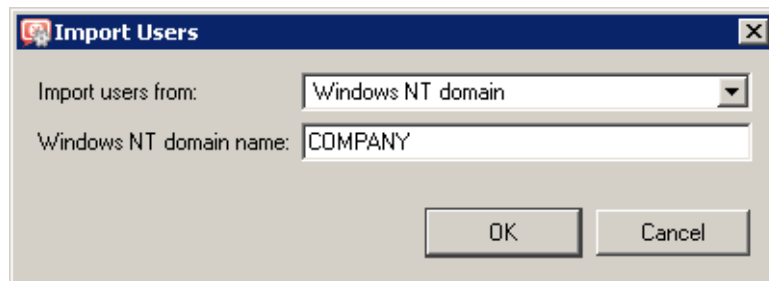


Figure 15.10 Importing accounts from the Windows NT domain

15.4 User accounts in Active Directory — domain mapping

In *WinRoute*, it is possible to directly use user accounts from one or more *Active Directory* domain(s). This feature is called either transparent support for *Active Directory* or *Active Directory* domain(s) mapping. The main benefit of this feature is that the entire administration of all user accounts and groups is maintained in *Active Directory* only (using standard system tools). In *WinRoute*, a template can be defined for each domain that will be used to set specific *WinRoute* parameters for user accounts (access rights, data transfer quotas, content rules — see chapter 15.1). If needed, these parameters can also be set individually for any accounts.

Note: The *Windows NT* domain cannot be mapped as described. In case of the *Windows NT* domain, it is recommended to import user accounts to the local user database (refer to 15.3)

Domain mapping requirements

The following conditions must be met to enable smooth functionality of user authentication through *Active Directory* domains:

- For mapping of one domain:
 1. The *WinRoute* host must be a member of the corresponding *Active Directory* domain.
 2. Hosts in the local network (user workstations) should use the *WinRoute*'s *DNS forwarder* as the primary DNS server, because it can process queries for *Active*

Directory and forward them to the corresponding domain server. If another DNS server is used, user authentication in the *Active Directory* may not work correctly.

- For mapping of multiple domains:
 1. The *WinRoute* host must be a member of one of the mapped domains. This domain will be set as primary.
 2. It is necessary that the primary domain trusts any other domains mapped in *WinRoute* (for details, see the documentation regarding the operating system on the corresponding domain server).
 3. For DNS configuration, the same rules as in mapping of a single domain are applied (the *WinRoute's DNS forwarder* is the best option).

Domain mapping settings

To set *Active Directory* domain mapping, go to:

- the *Administration Console*, section *Users and groups* → *Users*, the *Active Directory* tab,
- in the *Web Administration* interface, section *Users and Groups* → *Domains and authentication*, the *Active Directory*.

Connecting the firewall to a domain (Software Appliance / VMware Virtual Appliance)

The upper section of the *Active Directory* tab provides information about domain membership of the firewall's host.

In the *Software Appliance / VMware Virtual Appliance* edition, it is possible to add the firewall to a domain, change domain membership or disconnect the firewall from the domain.

This can be done in the easy-to-use wizard.

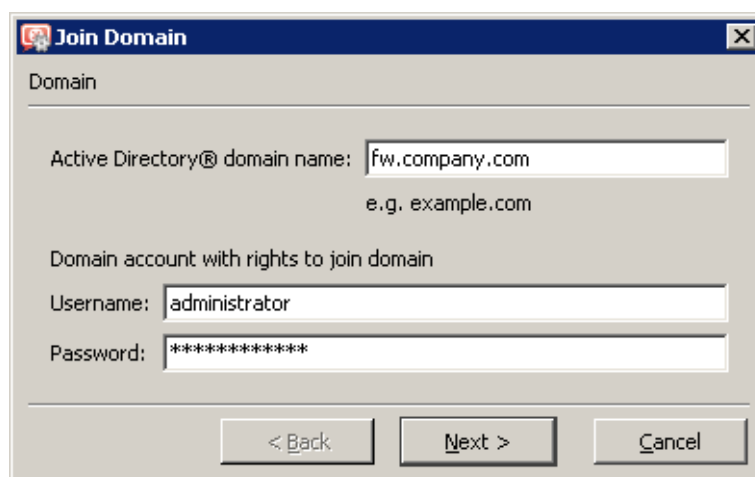


Figure 15.11 Adding firewall to a domain

The first page of the wizard requires the full name of the *Active Directory* domain (e.g. `company.com`) and name and password of a user with rights to add hosts to domains.

If *WinRoute* cannot find the domain server of the specified domain automatically, it requires specification of its IP address in the next step. Then the user gets informed about the result of the attempt to add the firewall to the domain.

Primary domain mapping

To set mapping of the primary domain (the domain of which the firewall host is a member), use option *Use domain user database*. For connection to the domain server, it is required to enter username and password of an account with read rights for the user database (any user account of the domain can be used, unless it is blocked).

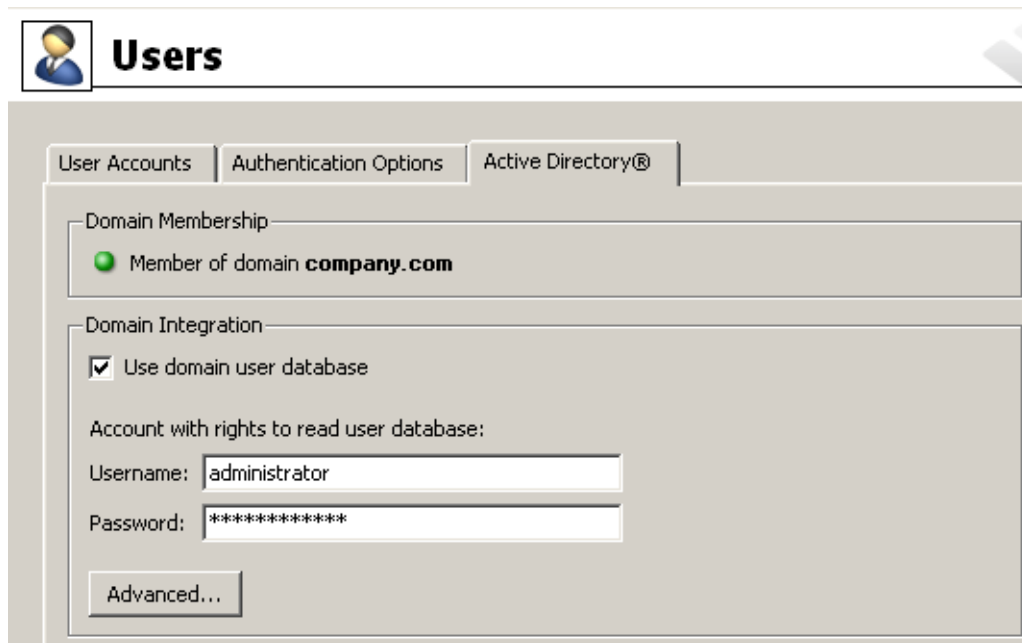


Figure 15.12 Primary domain mapping

Advanced Options

Method of cooperation between *WinRoute* and the *Active Directory* can be customized by some advanced options.

Domain mapping vs domain user authentication

The recommended method of cooperation with the *Active Directory* is domain mapping (user accounts are saved and managed only in the *Active Directory*). However, this can be undesirable under certain circumstances. For example if the *Active Directory* is implemented in a network where the *Windows NT* domain or no domain has been used, user accounts are already created in the *WinRoute's* local database. In such case, the best solution is to keep the local accounts and set only authentication in the *Active Directory* (so that users can use the same password both for the domain and the firewall).

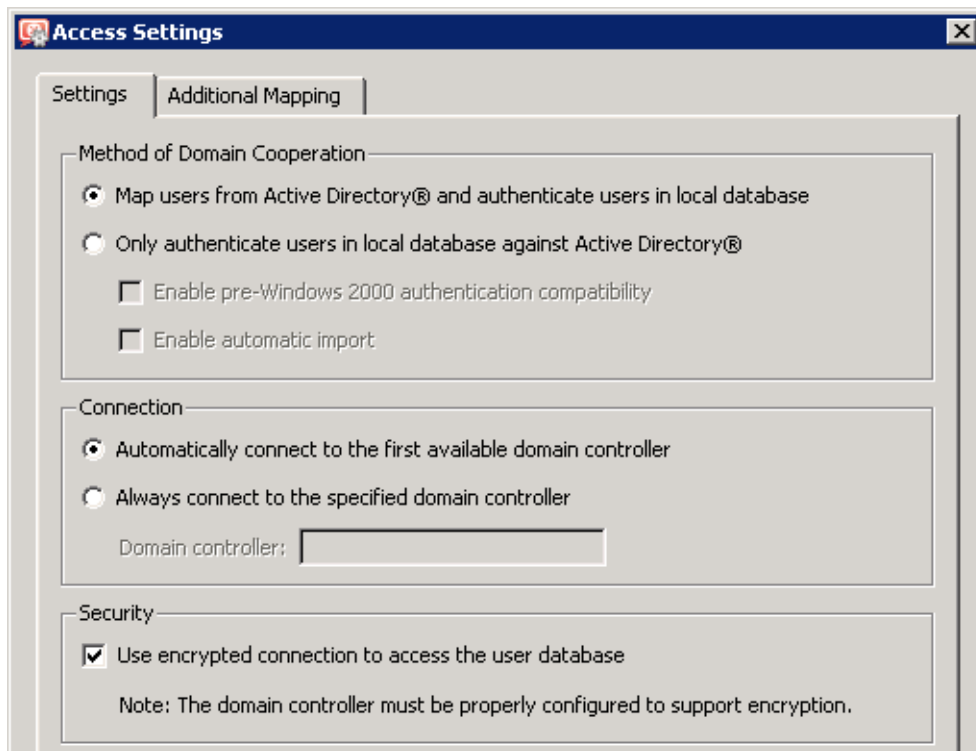


Figure 15.13 Advanced options for cooperation with the Active Directory.

If *WinRoute* is installed on *Windows*, it is possible to allow authentication compatible with older systems (i.e. authentication via the *Windows NT* domain). This option is required if the domain server uses *Windows NT* or if any of the clients in the local network uses *Windows* of older edition than *Windows 2000*. In *Software Appliance / VMware Virtual Appliance*, this option is not available (authentication in *Windows NT* domain is not supported).

Then, the settings include an option of automatic import of user accounts from the *Active Directory* to the local database (upon the first logon of user to the firewall by their domain name and password, an account with the same name will be created in the local database automatically). This option is available above all to keep the environment compatible with older *WinRoute* versions. In new installations it is strongly recommended to use domain mapping — administration of users is much more simple and much less time consuming. For details, see the *Administrator's Guide* for older versions of *WinRoute* (versions 6.7.0 or lower).

Selection of a domain server

In the default configuration, *WinRoute* automatically detects domain servers for the specified domain and uses the first detected server for connection to the *Active Directory*. Automatic detection simplifies configuration significantly (it is not necessary to specify IP addresses of individual domain servers).

If necessary, you can specify name of IP address of a specific domain server. In such case, *WinRoute* will not perform automatic detection and will always connect to the specified server only.

Secured connection to the domain server

For higher security (to prevent from tapping of traffic and exploiting user passwords), connection to the *Active Directory* can be encrypted. Enabling of encrypted connection requires corresponding settings on the particular domain server (or on all servers of the particular domain if automatic detection is used).

Mapping of other domains

To map user accounts from multiple *Active Directory* domains, add domains in advanced settings available on the *Other Mapping* tab.

Users of other domains must login by username including the domain (e.g. drdolittle@usoffice.company.com). User accounts with no domain specified (e.g. wsmith), will be searched in the primary domain or in the local database.

Add Domain

Active Directory® Mapping

Active Directory® domain name:

Description:

Domain Access

Account with rights to read user database:

Username:

Password:

Connection

Automatically connect to the first available domain controller

Always connect to the specified domain controller

Domain controller:

Security

Use encrypted connection to access the user database

Note: The domain controller must be properly configured to support encryption.

OK Cancel

Figure 15.14 Adding another Active Directory domain

Use buttons *Add* or *Edit* to open a dialog for a new domain definition and enter parameters of the mapped domain. For details, see above (Primary domain mapping and Advanced options).

Collision of Active Directory with the local database and conversion of accounts

During *Active Directory* domain mapping, collision with the local user database may occur if a user account with an identical name exists both in the domain and in the local database. If multiple domains are mapped, a collision may occur only between the local database and the primary domain (accounts from other domains must include domain names which make the name unique).

If a collision occurs, a warning is displayed at the bottom of the *User Accounts* tab. Click on the link in the warning to convert selected user accounts (to replace local accounts by corresponding *Active Directory* accounts).

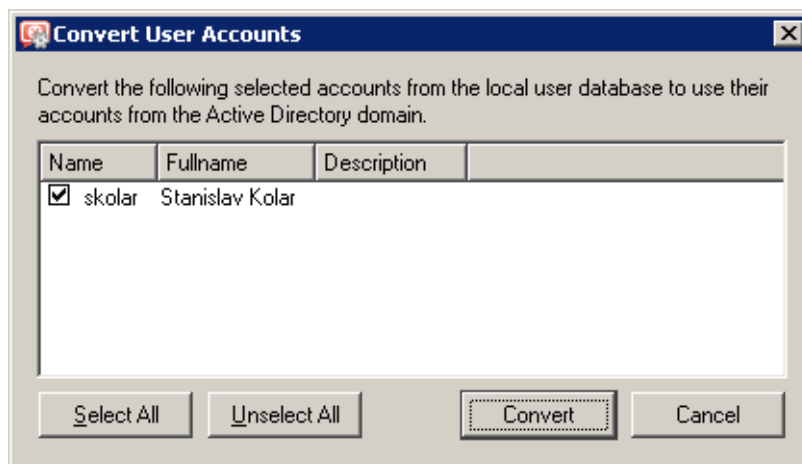


Figure 15.15 Conversion of user accounts

The following operations will be performed automatically within each conversion:

- substitution of any appearance of the local account in the *WinRoute* configuration (in traffic rules, URL rules, FTP rules, etc.) by a corresponding account from the *Active Directory* domain,
- removal of the account from the local user database.

Accounts not selected for the conversion are kept in the local database (the collision is still reported). Colliding accounts can be used — the accounts are considered as two independent accounts. However, under these circumstances, *Active Directory* accounts must be always specified including the domain (even though it belongs to the primary domain); username without the domain specified represents an account belonging to the local database. However, as long as possible, it is recommended to remove all collisions by the conversion.

Note: In case of user groups, collisions do not occur as local groups are always independent from the *Active Directory* (even if the name of the local group is identical with the name of the group in the particular domain).

15.5 User groups

User accounts can be sorted into groups. Creating user groups provides the following benefits:

- Specific access rights can be assigned to a group of users. These rights complement rights of individual users.
- 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 Definitions

User groups can be defined in *User and Groups* → *Groups*.

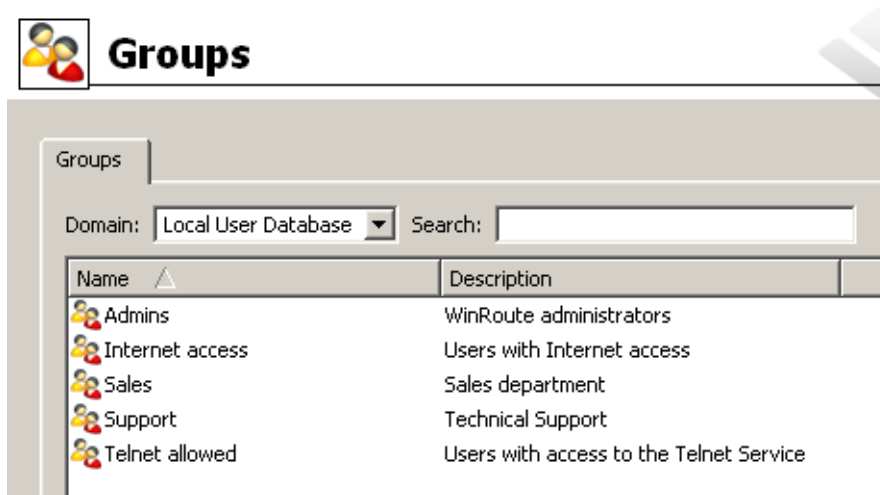


Figure 15.16 WinRoute user groups

Domain

Use the *Domain* option to select a domain for which user accounts or other parameters will be defined. This item provides a list of mapped *Active Directory* domains (see chapter 15.4) and the local user database.

In *WinRoute*, it is possible to create groups only in the local user database. It is not possible to create groups in mapped *Active Directory* domains. It is also not possible to import groups from the *Windows NT* domain or from *Active Directory*.

In case of groups mapped in *Active Directory* domains, it is possible to set only access rules (see below — step 3 of the user group definition wizard).

Search

The *Search* engine can be used to filter out user groups meeting specified criteria.

The searching is interactive — each symbol typed or deleted defines the string which is evaluated immediately and all groups including the string in either *Name* or *Description* are viewed. The icon next to the entry can be clicked to clear the filtering string and display all groups in the selected domain (if the *Search* entry is blank, the icon is hidden).

The searching is helpful especially when the domain includes too many groups which might make it difficult to look up particular items.

Creating a new local user group

In the *Domain* combo box in *Groups*, select Local User Database.

Click *Add* to start a wizard where a new user group can be created.

Step 1 — Name and description of the group

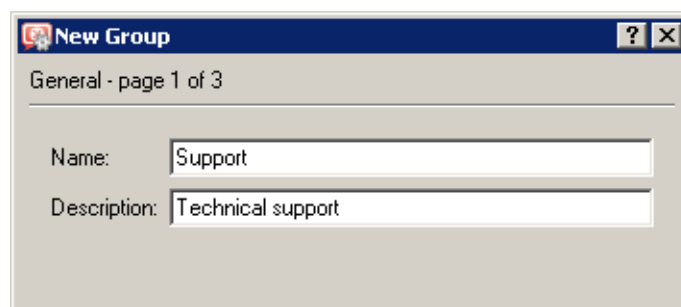


Figure 15.17 Creating a user group — basic parameters

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

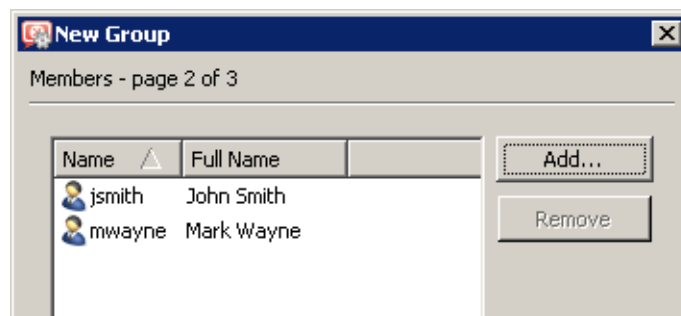


Figure 15.18 Creating a user group — adding user accounts to the group

Using the *Add* and *Remove* buttons you can add or remove users to/from the group. 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 [15.1](#)).

Hint

When adding new users you can select multiple user accounts by holding either the *Ctrl* or the *Shift* key.

Step 3 — group access rights

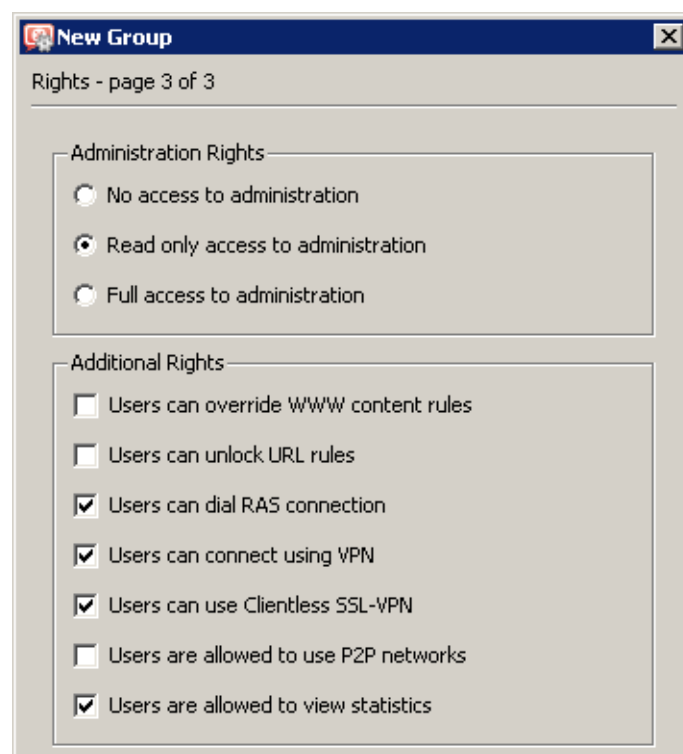


Figure 15.19 Creating a user group — members' user rights

The group must be assigned one of the following three levels of access rights:

No access to administration

Users included in this group cannot access the *WinRoute* administration.

Read only access

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 in this group have full access rights.

Additional rights:

Users can override WWW content rules

User belonging to the group can customize personal web content filtering settings (see chapter [15.2](#)).

User can unlock URL rules

This option allows its members one-shot bypassing of denial rules for blocked websites (if allowed by the corresponding URL rule — see chapter [12.2](#)). All performed unlock actions are traced in the *Security* log.

Users can dial RAS connection

If the Internet connection uses dial-up lines, users of this group will be allowed to dial and hang up these lines in the Web interface (see chapter [11](#)).

Users can connect using VPN

Members of the group can connect to the local network via the Internet using the *Kerio VPN Client* (for details, see chapter [23](#)).

User can use Clientless SSL-VPN

Members of this group will be allowed to access shared files and folders in the local network via the *Clientless SSL-VPN* web interface.

The *Clientless SSL-VPN* interface and the corresponding user right in *WinRoute* is available for *Windows* only. For details, see chapter [24](#).

Users are allowed to use P2P networks

The *P2P Eliminator* module (detection and blocking of *Peer-to-Peer* networks — see chapter [17.1](#)) will not be applied to members of this group.

Users are allowed to view statistics

Users in this group will be allowed to view firewall statistics in the web interface (see chapter [11](#)).

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.

Administrative settings

16.1 System configuration (Software Appliance / VMware Virtual Appliance)

In the *Software Appliance / VMware Virtual Appliance* edition, the *WinRoute* administration console allows setting of a few basic parameters of the firewall's operating system. These settings are necessary for correct functionality of the firewall and they can be found in *Configuration / Advanced options*, on the *System Configuration* tab.

Figure 16.1 System configuration — host name, date, time and time zone

Server name

Name is important both for some *WinRoute* services (e.g. secured web interface) and for the firewall's operating system's services.

The *DNS forwarder* module in *WinRoute* sets IP addresses of all the firewall's interfaces for the name automatically. If another DNS server is used in the local network, it is necessary to set corresponding DNS records on it.

Date, time and time zone

Many *WinRoute* features (user authentication, logs, statistics, etc.) require correct setting of date, time and time zone on the firewall.

Date and time can be set automatically but it is more useful to use an NTP server which provides information about the current time and allows automatic management of the

firewall's system time. The time zone also includes information about daylight saving time settings.

Kerio Technologies offers the following free NTP servers for this purpose: 0.kerio.pool.ntp.org, 1.kerio.pool.ntp.org, 2.kerio.pool.ntp.org and 3.kerio.pool.ntp.org.

16.2 Setting Remote Administration

Remote administration is connection to the firewall, its monitoring and configuration changes with the *Administration Console* or with the *Web Administration* interface from another host that the one on which *WinRoute* is installed.

If *WinRoute* includes only traffic rules created automatically by the wizard (see chapter 7.1), access to the remote administration is allowed via all trustworthy network interfaces (see chapter 5). This means that remote administration is available from all local hosts.

To allow or deny remote administration via the Internet (non-trusted networks), define a corresponding traffic rule. Traffic between *WinRoute* and *Administration Console* is performed by TCP and UDP protocols over port 44333. The definition can be done with the predefined service *KWF Admin*. the secured version of the *Web Administration* interface use TCP protocol, on port 4081 by default — predefined *KWF WebAdmin-SSL* service.

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 (see chapter 14.1).
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 *Any* or as *Internet*!)
- *Destination* — *Firewall* (host where *WinRoute* is installed).
- *Service* — *KWF Admin* (connection with the *Administration Console*) and *KWF WebAdmin-SSL* (secured version of the *Web Administration* interface).
It is not recommended to allow access via the unsecured version of the *Web Administration* interface (the *KWF WebAdmin* service)! Unsecured traffic might be tapped and misused for assaulting the firewall and local hosts behind it.
- *Action* — *Permit* (otherwise remote administration would be blocked)
- *Translation* — Because the engine is running on the firewall there is no need for translation.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Remote Administration	 Remote administration	 Firewall	 KWF Admin  KWF WebAdmin-SSL		

Figure 16.2 Traffic rule that allows remote administration

Hint

In *WinRoute*, you can use a similar method to allow or block remote administration of *Kerio MailServer* — for connection via the *Administration Console*, use the predefined service *KMS Admin*, for the *Web Administration* use *HTTPS*.

Note: Be very careful while defining traffic rules, otherwise you could block remote administration from the host you are currently working on. However, in most cases, *WinRoute* recognizes such situation and shows a warning message. Local connections (from the *WinRoute Firewall Engine's* host) works anyway. Such a traffic cannot be blocked by any rule.

16.3 Update Checking

WinRoute enables automatic check for new versions at the *Kerio Technologies* website. Whenever a new version is detected, its download and installation is offered.

Open the *Update Checker* tab in the *Configuration* → *Advanced Options* section to view information on a new version and to set parameters for automatic checks for new versions.

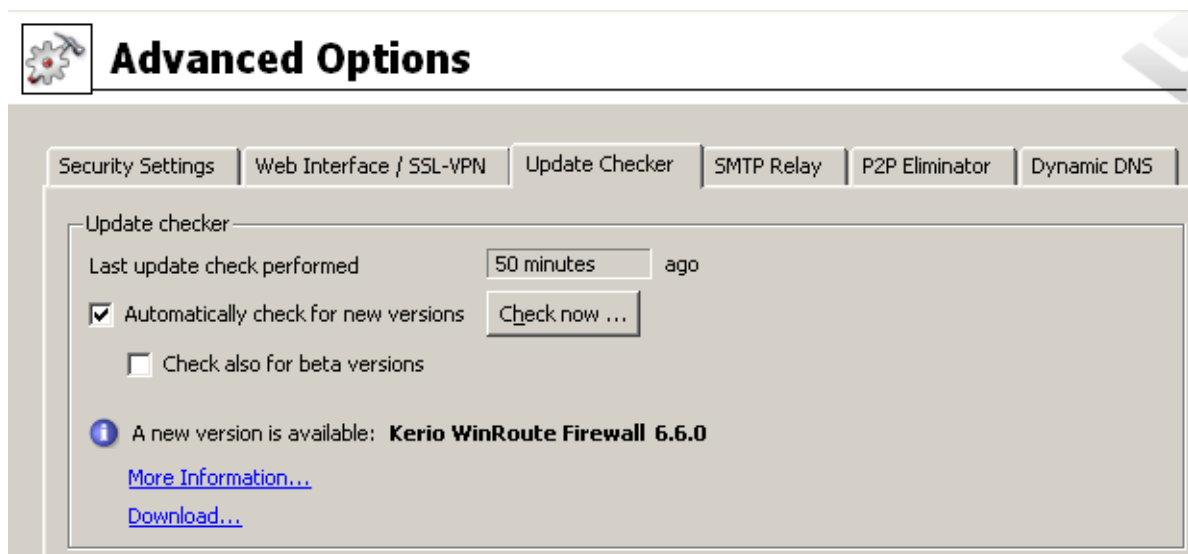


Figure 16.3 Check for new WinRoute versions

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 22.6) 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*,
- and then every 24 hours.

Results of each attempted update check (successful or not) is logged into the *Debug* log (see chapter [22.6](#)).

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* beta versions, 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 (beta versions 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 a new version is available, detailed information links and download links (links to installation files) are provided:

- *More information* — this link opens *WinRoute* changelog page in the default web browser.
- *Download* — direct link to the particular version's installation file. Click the link to download the installation file in your default browser.

For detailed information on *WinRoute* installation, refer to chapter [2.4](#).

Note: Whenever a new version is detected, this information is displayed as a link in the welcome page of the administration window (an image providing information about the application and the license). Clicking on the *Administration Console* link will take you to section *Configuration* → *Advanced Options*, the *Updates* tab.

Advanced security features

17.1 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 of big volumes of data (this sharing is mostly illegal). *DirectConnect* and *Kazaa* are the most popular ones.

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.⁶ 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.

The following restrictions can be applied to users of *P2P* networks (i.e. to hosts on which clients of such networks are run):

- *Blocking options* — it is possible to block access to the Internet for a particular host or to restrict the access only to selected services (e.g. web and e-mail),
- *Bandwidth limitation* — it is possible to decrease speed of data transmission of *P2P* clients so that other users are not affected by too much data transferred by the line.

P2P Eliminator Configuration

P2P networks are detected automatically (the *P2P Eliminator* module keeps running). To set the *P2P Eliminator* module's parameters, go to the *P2P Eliminator* tab in the *Configuration* → *Advanced Options* section.

As implied by the previous description, it is not possible to block connections to particular *P2P* networks. *P2P Eliminator* allows complete blocking of all traffic (i.e. access to the Internet from the particular host), enabling of only such services which are securely not associated with *P2P* networks or limiting of bandwidth (transfer speed) that can be used by *P2P* networks clients. The settings will be applied to all clients of *P2P* networks detected by *P2P Eliminator*.

Check the *Inform user by email* 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 /

⁶ According to thorough tests, the detection is highly reliable (probability of failure is very low).

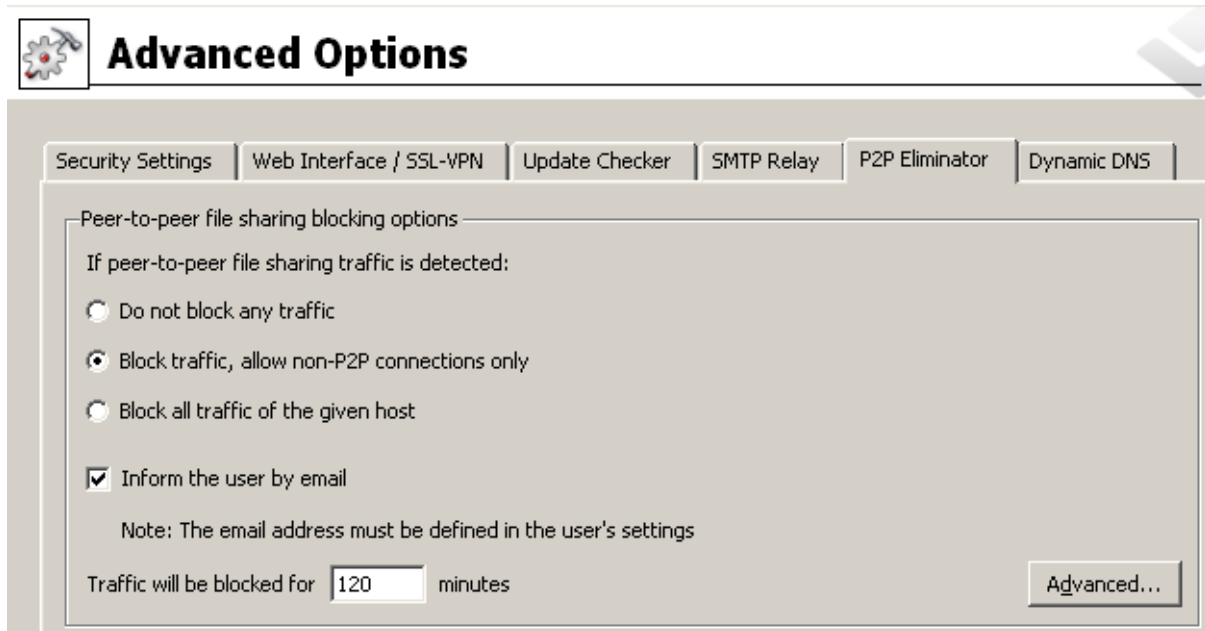


Figure 17.1 Detection settings and P2P Eliminator

allowance of only certain services and length of the period for which restrictions will be applied). The email is sent only if a valid email address (see chapter 15.1) is specified in the particular user account. This option does not apply to unauthenticated users.

The *Traffic will be blocked for* value defines time when the restriction for the particular host will be applied. 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 *P2P* networks.

If traffic of *P2P* network clients is not blocked, it is possible to set bandwidth limitation for *P2P* networks at the bottom of the *P2P Eliminator* tab. Internet lines are usually asymmetric (the speed vary for incoming and outgoing direction); therefore, this limitation is set separately for each direction. Bandwidth limitation applies only to traffic of *P2P* networks (detected by *P2P Eliminator*), other services are not affected.

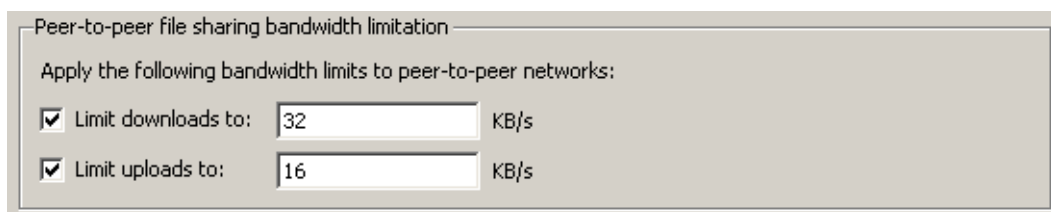


Figure 17.2 Bandwidth limits applied to P2P networks

Note:

1. If a user who is allowed to use *P2P* networks (see chapter [15.1](#)) is connected to the fire-wall 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 [19.1](#)).
3. If you wish to notify also another person when a *P2P* network is detected (e.g. the *WinRoute* administrator), define the alert on the *Alerts Settings* tab of the *Configuration → Accounting* section. For details, see chapter [19.4](#).

Parameters for detection of P2P networks

Click *Advanced* to set parameters for *P2P* detection.

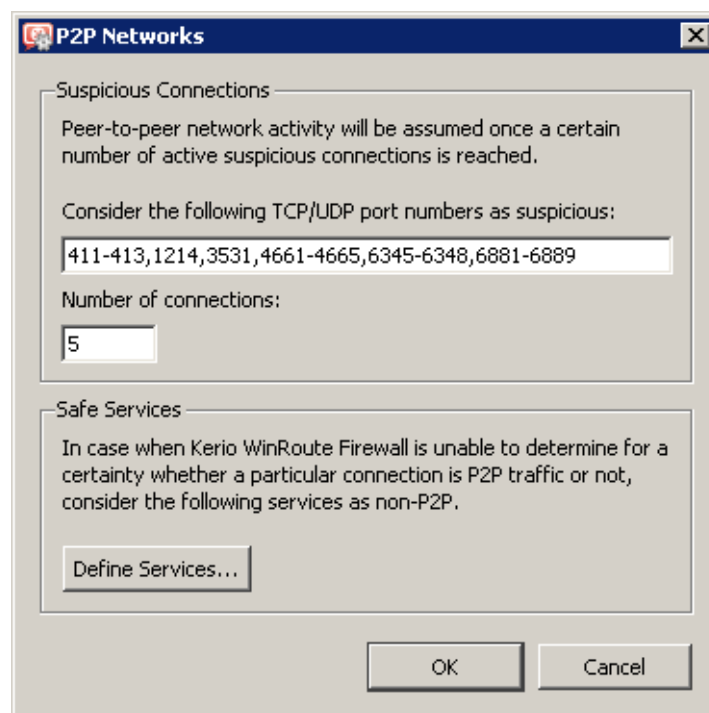


Figure 17.3 Settings of P2P networks detection

Ports of P2P networks

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. Ports in the list can be defined by port numbers or by port ranges. Individual values are separated by commas while dash is used for definition of ranges.

Number of suspicious connections

Big volume of connections established from the client host is a typical feature of *P2P* networks (usually one connection for each file). The *Number of connections* value defines maximal number of client's network connections that must be reached to consider the traffic as suspicious.

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).

Safe services

Certain "legitimate" services may also show characteristics of traffic in *P2P* networks (e.g. big number of concurrent connections). To ensure that traffic is not detected incorrectly and users of these services are not persecuted by mistake, it is possible to define list of so called secure services. These services will be excluded from detection of *P2P* traffic.

The *Define services...* button opens a dialog where services can be define that will not be treated as traffic in *P2P* network. All services defined in *Configuration* → *Definitions* → *Services* are available (for details, refer to chapter sect-services"/>).

Warning

Default values of parameters of *P2P* detection were set with respect to long-term testing. As already mentioned, it is not always possible to say that a particular user really uses *P2P* networks or not which results only in certain level of probability. Change of detection parameters may affect its results crucially. Therefore, it is recommended to change parameters of *P2P* networks detection only in legitimate cases (e.g. if a new port number is detected which is used only by a *P2P* network and by no legitimate application or if it is found that a legitimate service is repeatedly detected as a *P2P* network).

17.2 Special 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.

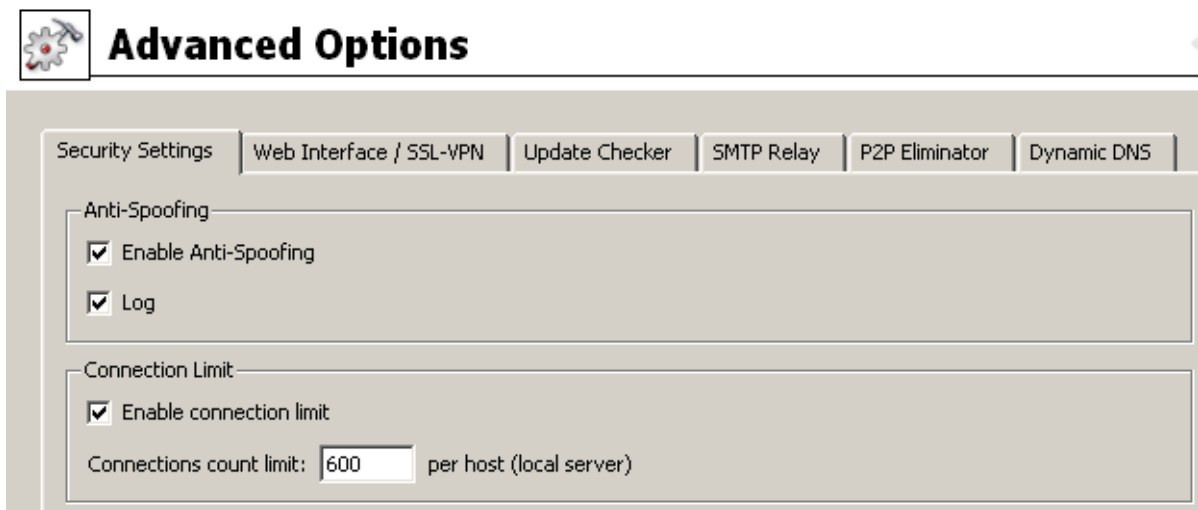


Figure 17.4 Security options — Anti-Spoofing and cutting down number of connections for one host

Anti-Spoofing

Anti-Spoofing checks whether only packets with allowed source IP addresses are received at individual interfaces of the *WinRoute* host. This function protects *WinRoute* host from attacks from the internal network that use false IP addresses (so called *spoofing*).

For each interface, any source IP address belonging to any network connected to the interface is correct (either directly or using other routers). For any interface connected to the Internet (so called external interface), any IP address which is not allowed at any other interface is correct.

Detailed information on networks connected to individual interfaces is acquired in the routing table.

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 [22.11](#)).

Connections Count Limit

This security function defines a limit for the maximum number of network connections which can be established from one local host (workstation) to the Internet or from the Internet to the local server via a mapped port.

Incoming and outgoing connections are monitored separately. If number of all connections established from/to a single local host in any direction reaches the specified value, *WinRoute* block any further connections in the particular direction.

These restrictions protect firewall (*WinRoute* host) from overload and may also help protect it from attacks to the target server, reduce activity and impact of a worm or Trojan horse.

Count limit for outgoing connections is useful for example when a local client host is attacked by a worm or Trojan horse which attempts to establish connections to larger number of various servers. Limiting of number of incoming connections can for example prevent the target from so called *SYN flood* attacks (flooding the server by opening too many concurrent connections without any data transferred).

Other settings

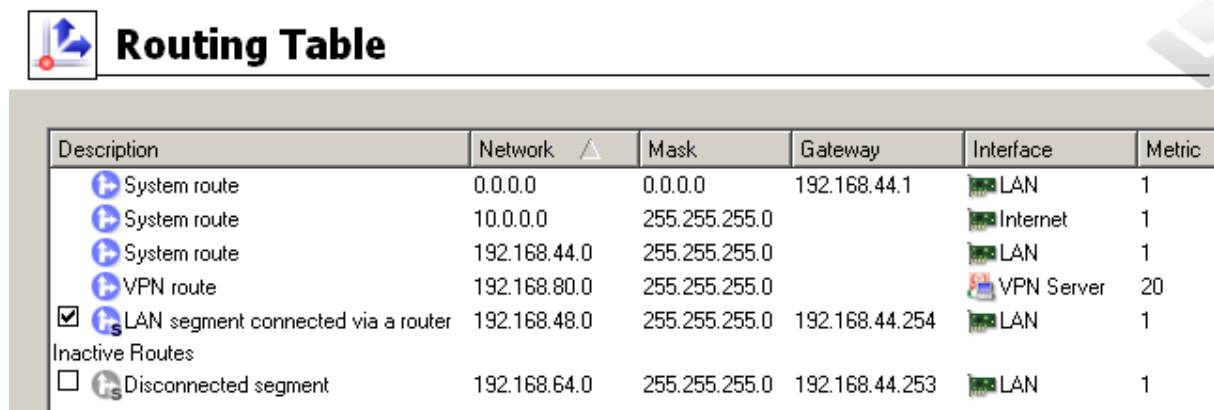
18.1 Routing table

Using *Administration Console* you can view or edit the system routing table of the host where *WinRoute* is running. 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 provides up-to-date version of the routing table of the operating system including so called *persistent routes* (routes added by the `route -p` command).

Note:

1. In the Internet connection failover mode (see chapter 6.3), only the current default route is shown (depending on which Internet interface is currently active).
2. In case of multiple Internet links in the network load balancing mode (see chapter 6.4), only a single default route will be displayed which is routed through the link with the highest proposed speed.



Description	Network	Mask	Gateway	Interface	Metric
System route	0.0.0.0	0.0.0.0	192.168.44.1	LAN	1
System route	10.0.0.0	255.255.255.0		Internet	1
System route	192.168.44.0	255.255.255.0		LAN	1
VPN route	192.168.80.0	255.255.255.0		VPN Server	20
<input checked="" type="checkbox"/> LAN segment connected via a router	192.168.48.0	255.255.255.0	192.168.44.254	LAN	1
Inactive Routes					
<input type="checkbox"/> Disconnected segment	192.168.64.0	255.255.255.0	192.168.44.253	LAN	1

Figure 18.1 Firewall's system routing table

Dynamic and static routes can be added and/or removed in section *Routing table*. Dynamic routes are valid only until the operating system is restarted or until removed by the `route` system command. Static routes are saved in *WinRoute* and they are restored upon each restart of the operating system.

Note: Changes in the routing table might interrupt the connection between the *WinRoute Firewall Engine* and the *Administration Console*. We recommend to check the routing table thoroughly before clicking the *Apply* button!

Route Types

The following route types are used in the *WinRoute* routing table:

- *System routes* — routes downloaded from the operating system's routing table (including so called persistent routes). These routes cannot be edited some of them can be removed — see the *Removing routes from the Routing Table* section).
- *Static routes* — manually defined routes managed by *WinRoute* (see below). These routes can be added, modified and/or removed.
The checking boxes can be used to disable routes temporarily —such routes are provided in the list of inactive routes. Static routes are marked with an *S* icon.
- *VPN routes* — routes to VPN clients and to networks at remote endpoints of VPN tunnels (for details, see chapter 23). These routes are created and removed dynamically upon connecting and disconnecting of VPN clients or upon creating and removing of VPN tunnels. VPN routes cannot be created, modified nor removed by hand.
- *Inactive routes* — routes which are currently inactive are showed in a separate section. These can be static routes that are temporarily disabled, static routes via an interfaces which has been disconnected or removed from the system, etc.

Static routes

WinRoute includes a special system for creation and management of static routes in the routing table. All static routes defined in *WinRoute* are saved into the configuration file and upon each startup of the *WinRoute Firewall Engine* they are added to the system routing table. In addition to this, these routes are monitored and managed all the time *WinRoute* is running. This means that whenever any of these routes is removed by the `route` command, it is automatically added again.

Note:

1. The operating system's persistent routes are not used for implementation of static routes (for management of these routes, *WinRoute* uses a proprietary method).
2. If a static connection uses a dial-up, any UDP or TCP packet with the *SYN* flag dials the line. For detailed information, see chapter 6.2.

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.

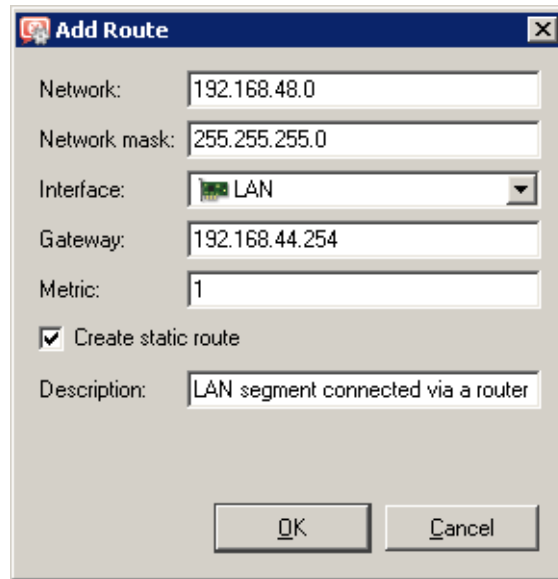


Figure 18.2 Adding a route to the routing table

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.

Create a static route

Enable this option to make this route static. Such route will be restored automatically by *WinRoute*(see above). A brief description providing various information (why the route was created, etc.) about the route can be attached.

If this option is not enabled, the route will be valid only until the operating system is restarted or until removed manually in the *Administration Console* or using the route command.

Removing routes from the Routing Table

Using the *Remove* button in the *WinRoute* admin console, records can be removed from the routing table. The following rules are used for route removal:

- Static routes in the *Static Routes* folder are managed by *WinRoute*. Removal of any of the static routes would remove the route from the system routing table immediately and permanently (after clicking on the *Apply* button).
- Dynamic (system) route will be removed as well, regardless whether it was added in the *Administration Console* or by the `route` command. However, it is not possible to remove any route to a network which is connected to an interface.
- Persistent route of the operating system will be removed from the routing table only after restart of the operating system. Upon reboot of the operating system, it will be restored automatically. 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` or the `route` command called from an execution script can be used, etc.). It is not possible to find out how a particular persistent route was created and how it might be removed for good.

18.2 Universal Plug-and-Play (UPnP)

WinRoute supports UPnP protocol (*Universal Plug-and-Play*). This protocol enables client applications (i.e. *Microsoft MSN Messenger*) to detect the firewall and make a request for mapping of appropriate ports from the Internet for the particular host in the local network. Such mapping is always temporary — it is either applied until ports are released by the application (using UPnP messages) or until expiration of the certain timeout.

The required port must not collide with any existing mapped port or any traffic rule allowing access to the firewall from the Internet. Otherwise, the UPnP port mapping request will be denied.

Configuration of the UPnP support

To configure UPnP go to the *Security Settings* folder in *Configuration* → *Advanced Options*.

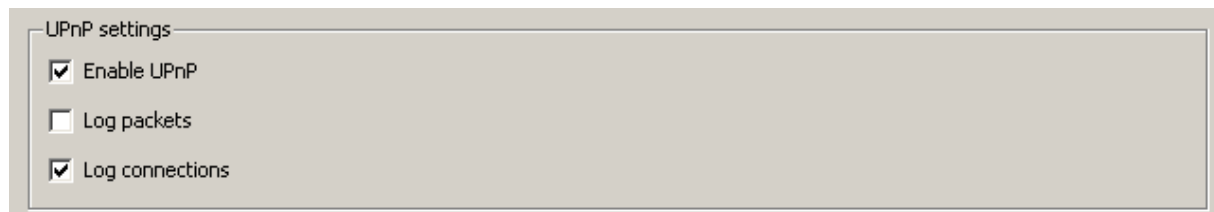


Figure 18.3 UPnP settings (the Security Settings tab under Configuration → Advanced Options)

Enable UPnP

This option enables UPnP.

Warning

If *WinRoute* is running on *Windows XP*, *Windows Server 2003*, *Windows Vista* or *Windows Server 2008*, check that 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 work with the UPnP protocol in *Windows*, and therefore they cannot be used together with *UPnP*.

Note: The *WinRoute* installation program detects the services and offers their stopping and denial.

Log packets

If this option is enabled, all packets passing through ports mapped with UPnP will be recorded in the *Filter* log (see chapter [22.9](#)).

Log connections

If this option is enabled, all packets passing through ports mapped with UPnP will be recorded in the *Connection* log (see chapter [22.5](#)).

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 [7.3](#)) you can limit usage of UPnP and enable it to certain IP addresses or certain users only.

Example:




Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Allow UPnP for selected hosts 	 UPnP clients	 Firewall	 UPnP		
<input checked="" type="checkbox"/> Deny UPnP 	 Trusted/Local	 Firewall	 UPnP		

Figure 18.4 Traffic rules allowing UPnP for specific hosts

The first rule allows UPnP only from *UPnP Clients* IP group. The second rule denies UPnP from other hosts (IP addresses).

18.3 Relay SMTP server

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 13.3), when a *Peer-to-Peer* network is detected (refer to chapter 17.1), when an alert function is set for certain events (details in chapter 15.1) or upon reception of an alert (see chapter 19.4).

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*.

Figure 18.5 SMTP settings — reports sending

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 alerts sent to users). Preset From header does not apply to messages forwarded during antivirus check (refer to chapter 13.4).

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*.

Connection test

Click *Test* to test functionality of sending of email via the specified SMTP server. *WinRoute* sends a testing email message to the specified email address.

Warning

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.

If the warning on the *SMTP server* tab is still displayed, it means that an invalid DNS name was specified or that an error occurred in the communication (DNS server is not responding). 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 [7](#).

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.

- Communication of each computer, users connected or all connections using *WinRoute* can be monitored.

Note:

1. *WinRoute* monitors only traffic between the local network and the Internet. The traffic within the local network is not monitored.
 2. Only traffic allowed by traffic rules (see chapter [7](#)) can be viewed. If a traffic attempt which should have been 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 see chapter [20](#).
 - Logs are files where information about certain activity is reported (e.g. error or warning reports, debug information etc.). Each item is represented by one row starting with a timestamp (date and time of the event). In all language versions of *WinRoute*, reports recorded are available in English only and they are generated by the *WinRoute Firewall Engine*. For details, refer to chapter [22](#).

The following chapters describe what information can be viewed and how its viewing can be changed to accommodate the user's needs.

19.1 Active hosts and connected users

In *Status* → *Active Hosts*, the hosts within the local network or active users using *WinRoute* for communication with the Internet will be displayed.

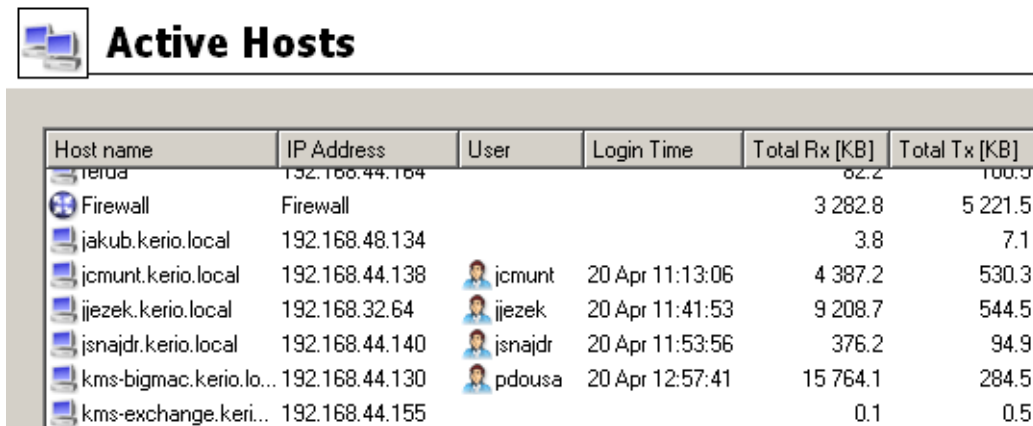
Note: For more details about the firewall user's logon see chapter [10.1](#).

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 *Active Hosts* window:

Hostname

DNS name of a host. In case that no corresponding DNS record is found, IP address is displayed instead.



Active Hosts

Host name	IP Address	User	Login Time	Total Rx [KB]	Total Tx [KB]
terua	192.168.44.164			82.2	100.9
Firewall	Firewall			3 282.8	5 221.5
jakub.kerio.local	192.168.48.134			3.8	7.1
jcmunt.kerio.local	192.168.44.138	jcmunt	20 Apr 11:13:06	4 387.2	530.3
jjezek.kerio.local	192.168.32.64	jjezek	20 Apr 11:41:53	9 208.7	544.5
jsnajdr.kerio.local	192.168.44.140	jsnajdr	20 Apr 11:53:56	376.2	94.9
kms-bigmac.kerio.lo...	192.168.44.130	pdousa	20 Apr 12:57:41	15 764.1	284.5
kms-exchange.keri...	192.168.44.155			0.1	0.5

Figure 19.1 List of active hosts and users connected to the firewall

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 [11.1](#))

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 *Internet Explorer* 5.5 or later or *Firefox/SeaMonkey* core version 1.3 or later is used)
- *VPN client* — user has connected to the local network using the *Kerio VPN Client* (for details, see chapter [23](#)).

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 [10.1](#).

Information displayed in the *Active Hosts* 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.

Active Hosts dialog options

Clicking the right mouse button in the *Active Hosts* window (or on the record selected) will display a context menu that provides the following options:

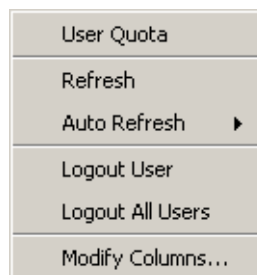


Figure 19.2 Context menu for the Active Hosts window

User quota

Use this option to show quota of the particular user (*Administration Console* switches to the *User quota* tab in *Status* → *Statistics* and selects the particular user automatically). The *User quota* option is available in the context menu only for hosts from which a user is connected to the firewall.

Refresh

This option refreshes information in the *Active Hosts* 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 *Active Hosts* 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 *Active Hosts* 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 *Active Hosts* 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.

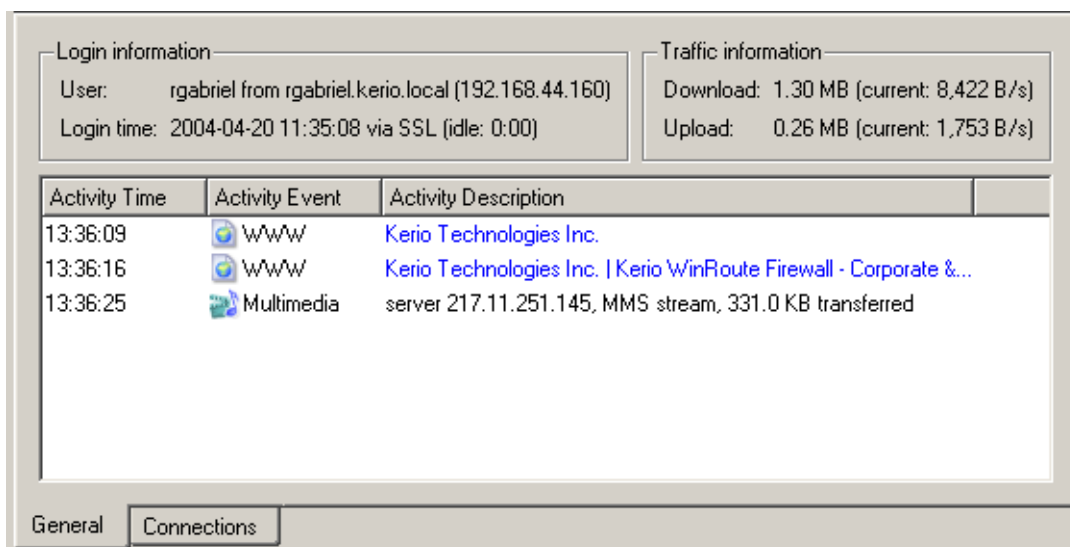


Figure 19.3 Information about selected host/user — actions overview

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.

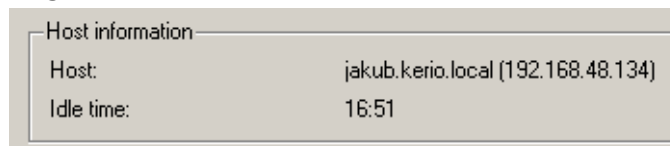


Figure 19.4 Host info (if no user is connected from it)

- *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 [17.1](#).

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.
Note: For better transparency, only the first visited page of each web server to which the user connected is displayed. For detailed information about all visited pages, refer to *Kerio StaR* (see chapter [21](#)).
- *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.

Information about connections from/to the Internet

On the *Connections* tab, you can view detailed information about connections established from the selected host to the Internet and in the other direction (e.g. by mapped ports, *UPnP*, etc.). The list of connections provides an overview of services used by the selected user. Undesirable connections can be terminated immediately.

Traffic rule	Service	Source	Source Port	Destination	Destination Port	Protocol	Info
NAT	MMS	217.11.251.145	4310	rgabriel.kerio.local	1132	UDP	Microsoft Stream
NAT	MMS	rgabriel.kerio.local	1865	217.11.251.145	1755	TCP	Microsoft Media
NAT	12774/TCP	rgabriel.kerio.local	1760	gw	12774	TCP	
NAT	ICQ	rgabriel.kerio.local	1616	64.12.24.161	5190	TCP	
NAT	1755/UDP	rgabriel.kerio.local	1866	217.11.251.145	1755	UDP	

Show DNS names

General Connections

Figure 19.5 Information about selected host/user — connections overview

Information about connections:

Traffic rule

Name of the *WinRoute* traffic rule (see chapter 7) 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:

1. 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 particular connection between the LAN and the Internet immediately.
2. The selected host's overview of connections lists only connections established from the particular host to the Internet and vice versa. Local connections established between the particular host and the firewall can be viewed only in *Status* → *Connections* (see chapter [19.2](#)). Connections between hosts within the LAN are not routed through *WinRoute*, and therefore they cannot be viewed there.

Histogram

The *Histogram* tab provides information on data volume transferred from and to the selected host in a selected time period. The chart provides information on the load of this host's traffic on the Internet line through the day.

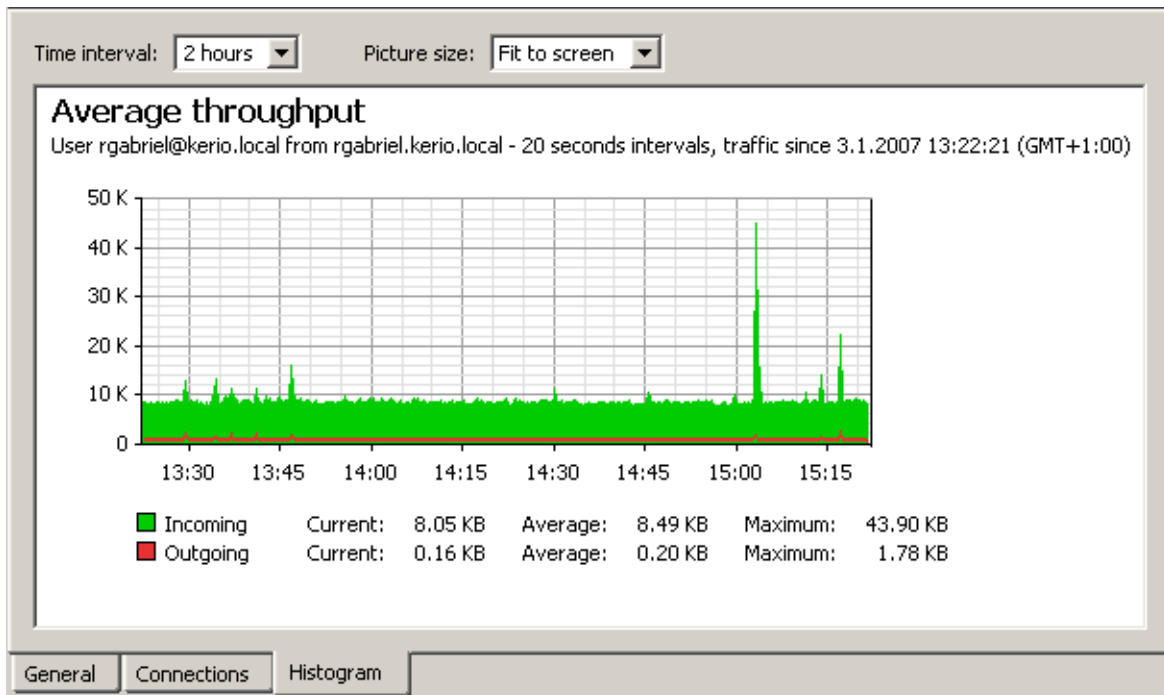


Figure 19.6 Information on selected host and user — traffic histogram

Select an item from the *Time interval* combo box to specify a time period which the chart will refer to (2 hours or 1 day). 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*).

This chart includes volume of transferred data in the selected direction in certain time intervals (depending on the selected period). The green curve represents volume of incoming data (download) in a selected time period, while the area below the curve represents the total volume of data transferred in the period. The red curve and area provide the same information for outgoing data (upload). Below the chart, basic statistic information, such as volume of data currently transferred (in the last interval) and the average and maximum data volume per an interval, is provided.

Select an option for *Picture size* to set a fixed format of the chart or to make it fit to the *Administration Console* screen.

19.2 Network connections 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

WinRoute administrators are allowed to close any of the active connections.

Note:

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.



Connections

Traffic rule	Service	Source	Source Port	Destination	Destination Port	Protocol
NAT	eDonkey	192.168.48.131	1870	172.184.232.10	4662	TCP
NAT	eDonkey	192.168.48.131	1724	172.206.233.246	4662	TCP
Local Traffic	DNS	gw-devel	1757	192.168.10.10	53	UDP
NAT	4246/UDP	192.168.48.131	2094	193.111.199.179	4246	UDP
NAT	4246/UDP	192.168.48.131	2094	193.111.199.183	4246	UDP
NAT	4246/UDP	192.168.48.131	2094	193.111.199.187	4246	UDP
NAT	4246/UDP	192.168.48.131	2094	193.111.199.211	4246	UDP
NAT	HTTP	192.168.44.143	2577	194.228.19.21	80	TCP
NAT	HTTP	192.168.44.143	2576	194.228.19.21	80	TCP
NAT	eDonkey	192.168.48.131	2005	195.14.200.83	4662	TCP
NAT	IMAPS	192.168.36.128	33228	195.39.55.2	993	TCP
NAT	IMAPS	192.168.44.153	2278	195.39.55.2	993	TCP
Local Traffic	HTTPS	192.168.48.131	1696	195.39.55.6	443	TCP
Local Traffic	HTTPS	192.168.48.131	1917	195.39.55.6	443	TCP
NAT	2914/TCP	192.168.44.131	2950	195.39.55.20	2914	TCP
NAT	2914/TCP	192.168.44.131	2951	195.39.55.20	2914	TCP

Figure 19.7 Overview of all connections established via *WinRoute*

One connection is represented by each line of the *Connections* window. These are network connections, not user connections (each client program can occupy more than one connection at a given moment).

The columns contain the following information:

Traffic rule

Name of the *WinRoute* traffic rule (see chapter 7) by which the connection was allowed.

Service

Name of transmitted service (if such service is defined in *WinRoute* — see chapter 14.3). If the service is not defined in *WinRoute*, the corresponding port number and protocol will be displayed instead (e.g. *5004/UDP*).

Source, 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, 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, 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.

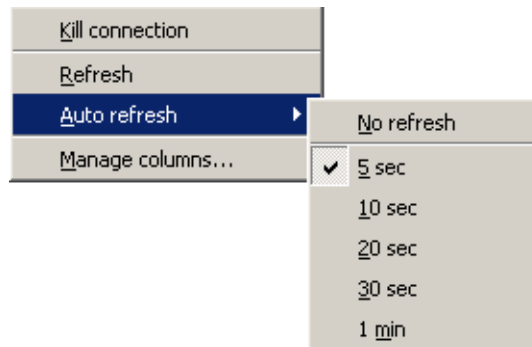


Figure 19.8 Context menu for Connections

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:

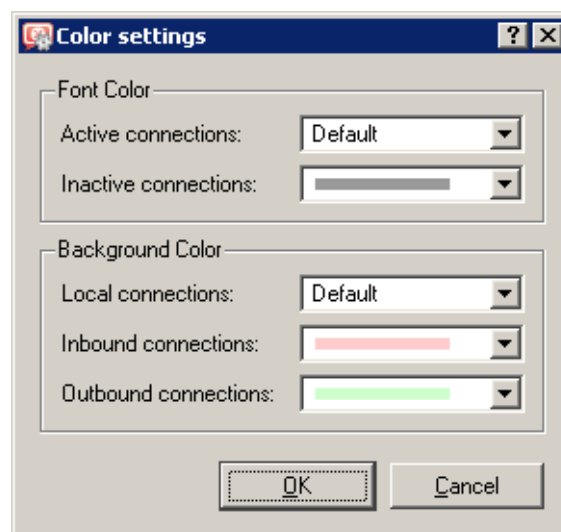


Figure 19.9 Connection colors settings

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 mis-handling)

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 details, refer to chapter 7.

19.3 List of connected VPN clients

In *Status* → *VPN clients*, you can see an overview of VPN clients currently connected to the *WinRoute*’s VPN server.

Username	Operating system	Hostname	Client IP	Login time	Version
jsmith	Mac OS X (10.4.11)	gw.there.com	10.1.1.4	8 minutes	6.6.0.5693
mwayne	Ubuntu 8.10	a185.adsl1.com	10.1.1.2	15 minutes	6.6.0.5693
lcarr	Windows Vista	95.112.211.27	10.1.1.3	34 minutes	6.6.0.5666

Figure 19.10 List of connected VPN clients

The information provided is as follows:

- Username used for authentication to the firewall. VPN traffic is reflected in statistics of this user.
- The operating system on which the user have the *Kerio VPN Client* installed.
- DNS name of the host which the user connects from. If *WinRoute* cannot resolve the corresponding hostname from the DNS, its (public) IP address is displayed instead.
- IP address assigned to the client by the VPN server. This IP address “represents” the client in the local network.
- Session duration.
- *Kerio VPN Client* version, including build number.

The following columns are hidden in the default settings of the *VPN clients* window (for details on showing and hiding columns, see chapter 3.2):

- IP address — public IP address of the host which the client connects from (see the *Hostname* column above).
- Client status — *connecting*, *authenticating* (*WinRoute* verifies username and password), *authenticated* (username and password correct, client configuration in progress), *connected* (the configuration has been completed, the client can now communicate with hosts within the local network).

Note: Disconnected clients are removed from the list automatically.

19.4 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 22.3). 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 18.3), otherwise alerting will not work.

Alerts Settings

Alerts settings can be configured in the *Alerts settings* tab under *Configuration* → *Accounting*.

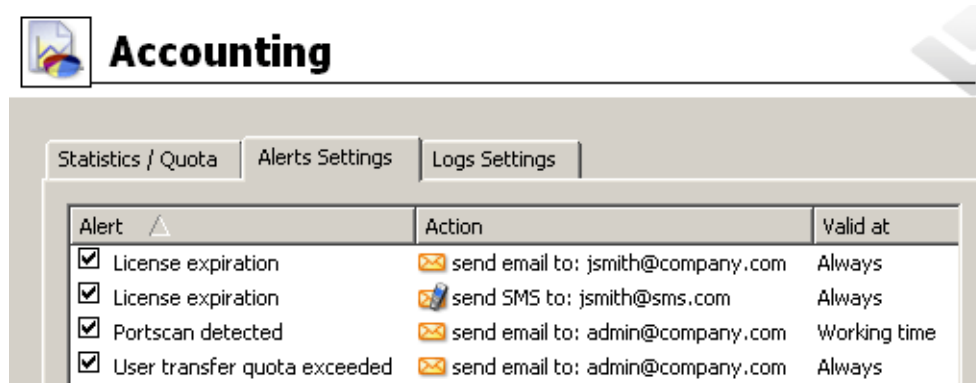


Figure 19.11 WinRoute Alerts

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.

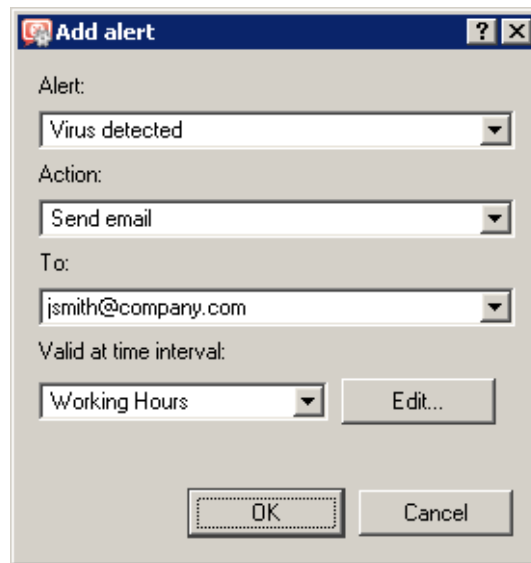


Figure 19.12 Alert Definitions

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 13).
- *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 17.2). This may indicate deployment of an undesirable network application (e.g. Trojan horse or a spyware) on a corresponding host.
- *Low free disk space warning* — this alert warns the administrator that the free space of the WinRoute host is low (under 11 per cent of the total disk capacity). WinRoute needs enough disk 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 16.3).
- *User transfer quota exceeded* — a user has reached daily, weekly or monthly user transfer quota and WinRoute has responded by taking an appropriate action. For details, see chapter 15.1.
- *Connection failover event* — the Internet connection has failed and the system was switched to a secondary line, or vice versa (it was switched back to the primary line). For details, refer to chapter 6.3.
- *License expiration* — expiration date for the corresponding WinRoute li-

cense/subscription (or license of any module integrated in *WinRoute*, such as *Kerio Web Filter*, 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 4).

- *Dial / Hang-up of RAS line* *WinRoute* is dialing or hanging-up a RAS line (see chapter 5). 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 14.2).

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* (the typical path is `C:\Program Files\Kerio\WinRoute Firewall\templates`):

- 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.

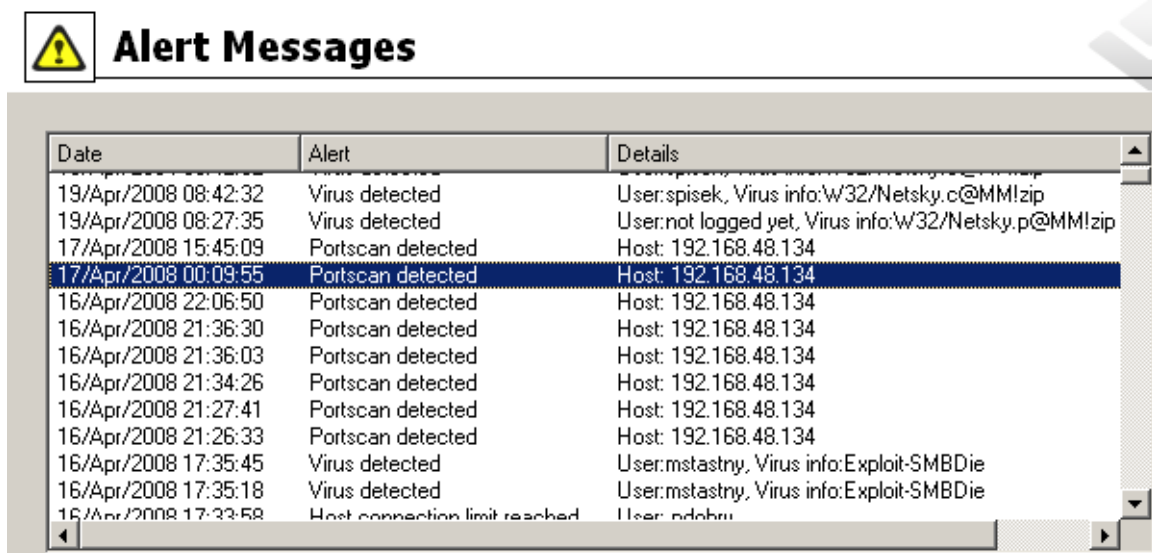
In the *Administration Console*, alerts are displayed in the language currently set as preferred (see *Kerio Administration Console — Help*). If alert templates in the language are not available, English version is used instead. Email and SMS alerts are always in English.

Note: In the current *WinRoute* version, alerts are available only in English and Czech.

Alerts overview (in Administration Console)

Overview of all sent alerts (defined in *Configuration* → *Accounting*) can be found under *Status* → *Alert Messages*. 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.



The screenshot shows a window titled "Alert Messages" with a yellow warning icon. It contains a table with three columns: Date, Alert, and Details. The table lists several alerts, with the most recent one highlighted in blue.

Date	Alert	Details
19/Apr/2008 08:42:32	Virus detected	User:spisek, Virus info:W32/Netsky.c@MM!zip
19/Apr/2008 08:27:35	Virus detected	User:not logged yet, Virus info:W32/Netsky.p@MM!zip
17/Apr/2008 15:45:09	Portscan detected	Host: 192.168.48.134
17/Apr/2008 00:09:55	Portscan detected	Host: 192.168.48.134
16/Apr/2008 22:06:50	Portscan detected	Host: 192.168.48.134
16/Apr/2008 21:36:30	Portscan detected	Host: 192.168.48.134
16/Apr/2008 21:36:03	Portscan detected	Host: 192.168.48.134
16/Apr/2008 21:34:26	Portscan detected	Host: 192.168.48.134
16/Apr/2008 21:27:41	Portscan detected	Host: 192.168.48.134
16/Apr/2008 21:26:33	Portscan detected	Host: 192.168.48.134
16/Apr/2008 17:35:45	Virus detected	User:mstastny, Virus info:Exploit-SMBDie
16/Apr/2008 17:35:18	Virus detected	User:mstastny, Virus info:Exploit-SMBDie
16/Apr/2008 17:33:58	Host connection limit reached	User:ndobru

Figure 19.13 Overview of sent alerts

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](#)

Figure 19.14 Details of a selected event

Chapter 20

Basic 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 *WinRoute*.

In the *Administration Console*, it is possible to view basic quota information for individual users (volume of transferred data and quota usage information) and statistics of network interfaces (transferred data, traffic charts). Detailed statistics of users, web pages and volume of transferred data are available in the firewall's web interface (*Kerio StaR* — see chapter [21](#)).

20.1 Volume of transferred data and quota usage

The *User Statistics* of the *Status* → *Statistics* section provides 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 [15.1](#)). 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:

1. User quota consists of three limits: daily, weekly and monthly. The *Quota* column provides the highest value of the three percentual values (if the daily usage is 50% of the daily quota, the weekly usage is 90% and the monthly usage is 70%, yellowed 90% value is displayed in the *Quota* column).
2. Monthly quota is reset automatically at the beginning of an accounting period. This period may differ from a civil month (see chapter [21.2](#)).

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. These lines do not include quota usage information.

Note:

1. Optionally, other columns providing information on volume of data transmitted in individual time periods in both directions can be displayed. Direction of data transmission

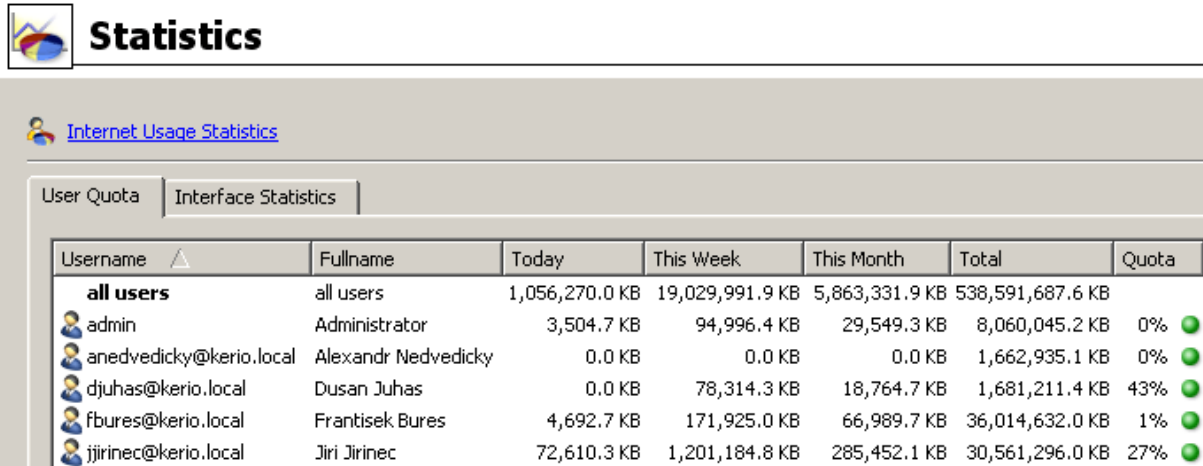


Figure 20.1 User statistics

is related to the user (the *IN* direction stands for data received by the user, while *OUT* represents data sent by the user). Hiding/showing of columns is addressed in chapter [3.2](#).

- Information of volume of data transferred by individual users is saved in the `stats.cfg` file in the *WinRoute* directory. This implies that this data will be saved the next time the *WinRoute Firewall Engine* will be started.

User Quota dialog options

Right-click on the table (or on an item of a selected user) to open the context menu with the following options:

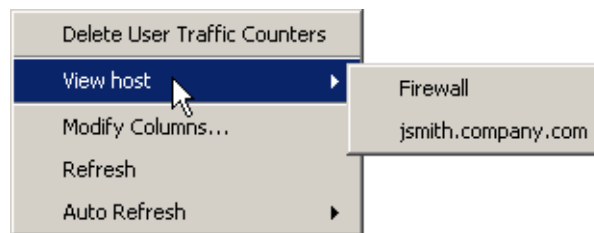


Figure 20.2 User Quota context menu

Delete User Traffic Counters

Removal of the selected line with data referring to a particular user. This option is helpful for reference purposes only (e.g. to exclude blocked user accounts from the list, etc.). Removed accounts will be added to the overview automatically when data in the particular account is changed (e.g. when we unblocked an account and its user connects and starts to communicate again).

Warning

Be aware that using this option for the *all users* item resets counters of all users, including unrecognized ones!

Note: Values of volumes of transferred data are also used to check user traffic quota (see chapter [15.1](#)). Reset of user statistics also unblocks traffic of the particular user in case that the traffic has been blocked for quota reasons.

View host...

This option is not available unless the selected user is connected to the firewall. The *View host* option switches to the *Status* → *Active Hosts* section of the host the particular user is connected from.

If the user is connected from multiple hosts, the *View host* option opens a submenu with a list of all hosts which the particular user is connected from.

Refresh

This option will refresh the information on the *User Statistics* tab 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 on the *User Statistics* tab. 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

Use this option to select and unselect items (columns) which will (not) be displayed in the table (see chapter [3.2](#)).

20.2 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).

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*.

Optionally, other columns providing information on volume of data transmitted in individual time periods in both directions can be displayed. Direction of data transmission is related to the interface (the *IN* direction stands for data received by the interface, while *OUT* represents data sent from the interface). Hiding/showing of columns is addressed in chapter [3.2](#).

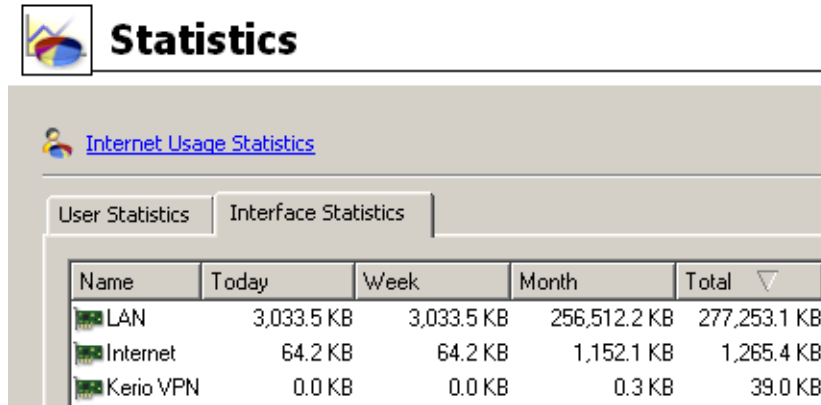


Figure 20.3 Firewall's interface statistics

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 `stats.cfg` configuration file in the *WinRoute's* installation directory. This implies that they are not reset when the *WinRoute Firewall Engine* is closed.

Interface Statistics menu

A context menu providing the following options will be opened upon right-clicking anywhere in the table (or on a specific interface):

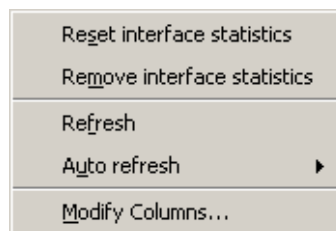


Figure 20.4 Context menu for Interface statistics

Reset interface statistics

This option resets statistics of the selected interface. It is available only if the mouse pointer is hovering an interface at the moment when the context menu is opened.

Refresh

This option will refresh the information on the *Interface Statistics* tab 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 on the *Interface Statistics* tab. 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

Use this option to select and unselect items (columns) which will (not) be displayed in the table (see chapter 3.2).

Remove interface statistics

This option removes the selected interface from the statistics. 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. Whenever a removed interface is activated again (upon connection of the VPN tunnel, etc.), it is added to the statistics automatically.

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).

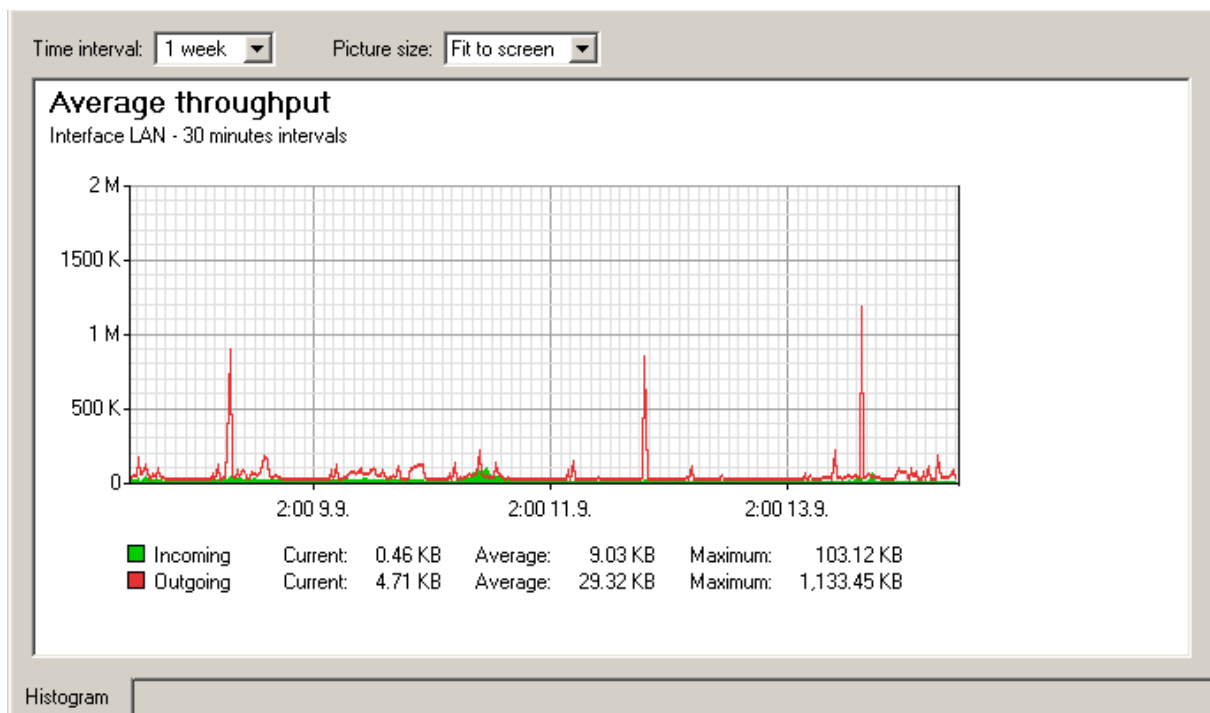


Figure 20.5 Chart informing about average throughput at the interface

The period (*2 hours* or *1 day*) can be selected in the *Time interval* box. The selected time range is always understood as the time until now (“last 2 hours” or “last 24 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*).

Select an option for *Picture size* to set a fixed format of the chart or to make it fit to the *Administration Console* screen.

The legend above the graph shows the sampling interval (i.e. the time for which a sum of connections or messages is counted and is displayed in the graph).

— **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.

Kerio StaR - statistics and reporting

The *WinRoute's* web interface provides detailed statistics on users, volume of transferred data, visited websites and web categories. This information may help figure out browsing activities and habits of individual users.

The statistics monitor the traffic between the local network and the Internet. Volumes of data transferred between local hosts and visited web pages located on local servers are not included in the statistics (also for technical reasons).

One of the benefits of web statistics and reports is their high availability. The user (usually an office manager) does not need the *Administration Console* and they even do not need *WinRoute* administrator rights (special rights are used for statistics). Statistics viewed in web browsers can also be easily printed or saved on the disk as web pages.

Notes:

1. The *WinRoute* administrator should inform users that their browsing activities are monitored by the firewall.
2. Statistics and reports in *WinRoute* should be used for reference only. It is highly unrecommended to use them for example to figure out exact numbers of Internet connection costs per user.
3. For correct functionality of the *Kerio StaR* interface, it is necessary that the *WinRoute* host's operating system supports all languages that would be used in the *Kerio StaR* interface. Some languages (Chinese, Japanese, etc.) may require installation of supportive files. For details, refer to documents regarding the corresponding operating system.

This chapter addresses setting of parameters in the *WinRoute's* administration program. The *vKerio StaR* interface is described thoroughly in the *Kerio WinRoute Firewall — User's Guide*.

21.1 Monitoring and storage of statistic data

Diverse data is needed to be gathered for the statistics. Statistic data is stored in the database (the `star\data` subdirectory of the *WinRoute's* installation directory — for details, see chapter [25.1](#)). Total period length for which *WinRoute* keeps the statistics can be set in the *Accounting* section of the *Administration Console* (see chapter [21.2](#)). By default, this time is set to *24 months* (i.e. 2 years).

For technical reasons, the *WinRoute Firewall Engine* stores gathered statistic data in the cache (the `star/cache` subdirectory) and data is recorded in the database once per hour. The cache

is represented by several files on the disk. This implies that any data is kept in the cache even if the *WinRoute Firewall Engine* is stopped or another problem occurs (failure of power supply, etc.) though not having been stored in the database yet.

The statistics use data from the main database. This implies that current traffic of individual users is not included in the statistics immediately but when the started period expires and the data is written in the database.

Note: Data in the database used for statistics cannot be removed manually (such action would be meaningless). In statistics, it is possible to switch into another view mode where data is related only to a period we need to be informed about. If you do not wish to keep older data, it is possible to change the statistics storage period (see above).

Requirements of the statistics

The following conditions must be met for correct function of all statistics:

- The firewall should always require user authentication. The statistics by individual users would not match the true state if unauthenticated users are allowed to access the Internet. For details see chapter [10](#).
- For statistics on visited websites, it is necessary that a corresponding protocol inspector is applied to any *HTTP* traffic. This condition is met by default unless special traffic rules disabling the particular protocol inspector are applied (see chapter [7.7](#)).
If the *WinRoute* proxy server is used, visited pages are monitored by the proxy server itself (see chapter [8.4](#)).
Note: *HTTPS* traffic is encrypted and, therefore, it is impossible to monitor visited sites and categories. Only volume of transferred data is included in the statistics for such traffic.
- For monitoring of web categories of visited websites, the *Kerio Web Filter* module must be enabled. In its configuration, the *Categorize each page regardless of HTTP rules* option should be enabled, otherwise web categories statistics would be unreliable. For details, see chapter [12.3](#).

Gathering of statistical information and mapped services

Connections from the Internet to mapped services on local hosts (or to services on the firewall available from the Internet — see chapter [7.3](#)) are also included in user statistics. If a user is connected to the firewall from the particular host, access to the mapped service is considered as an activity of this user. Otherwise, such connection is included in activity of unknown users (users who are not logged in).

The following example helps recognize importance of this feature. User *jsmith* is authenticated at the firewall and connected to it from a local workstation. The *RDP* service for this host is mapped on the firewall, allowing the user to work remotely on the workstation. If user *jsmith* connects from the Internet to the remote desktop on the workstation, this connection (and data transferred within the connection) will be correctly included in the user's statistics and quota.

The following example addresses case of a mapped web server accessible from the Internet. Any (anonymous) user in the Internet can connect to the server. However, web servers are usually located on a special machine which is not used by any user. Therefore, all traffic of this server will be accounted for users who are “not logged in”.

However, if any user is connected to the firewall from the server, any traffic between clients in the Internet and the web server is accounted as an activity of this user. If this user also reaches their data volume quota, corresponding restrictions will be applied to this web server (see chapters [15.2](#) and [9.2](#)).

21.2 Settings for statistics and quota

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. Therefore, *WinRoute* allows such configuration of statistics that is customized so that only useful data is gathered and useful statistics created. It is also possible to disable creation of statistics if desirable. This would save operation space of *WinRoute* as well as the disk space of its host.

Statistics settings also affect monitoring of volume of transferred data against user quota (refer to chapters [15.1](#) and [20](#)).

Use the *Statistics / Quota* tab in *Configuration* → *Accounting* to set gathering of statistical data and accounting periods for quota and statistics.

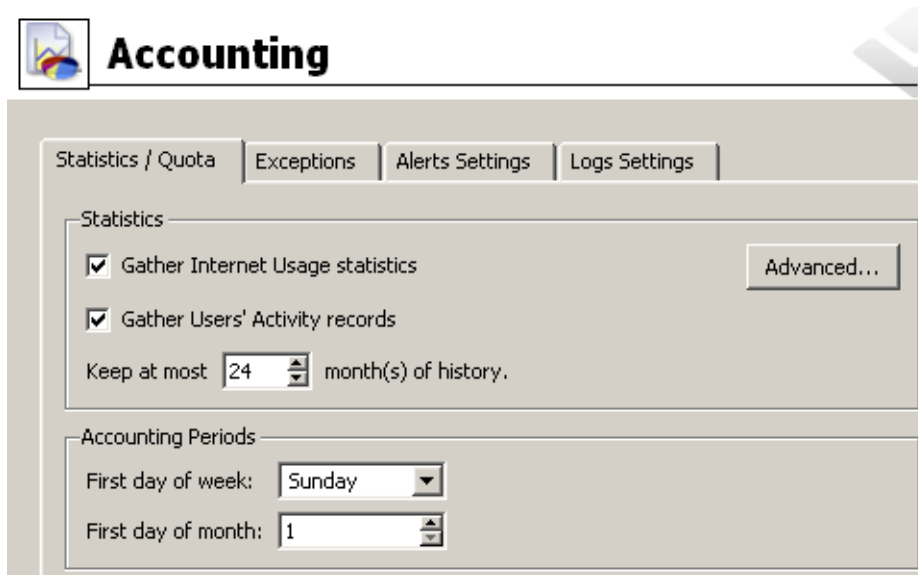


Figure 21.1 Setting of statistics and accounting periods

Enable/disable gathering of statistic data

The *Gather Internet Usage statistics* option enables/disables all statistics (i.e. stops gathering of data for statistics).

The *Monitor user browsing behavior* option enables monitoring and logging of browsing activity of individual users. If is not necessary to gather these statistics, it is recommended to disable this option (this reduces demands to the firewall and saves the server's disk space).

You can use the *Keep at most...* parameter to specify a time period for which the data will be kept (i.e. the age of the oldest data that will be available). This option affects disk space needed for the statistics remarkably. To save disk space, it is therefore recommended to keep the statistics only for a necessary period.

Advanced settings for statistics

The *Advanced* button opens a dialog where parameters can be set for viewing of statistics in the *Kerio StaR* interface (see chapter 20).

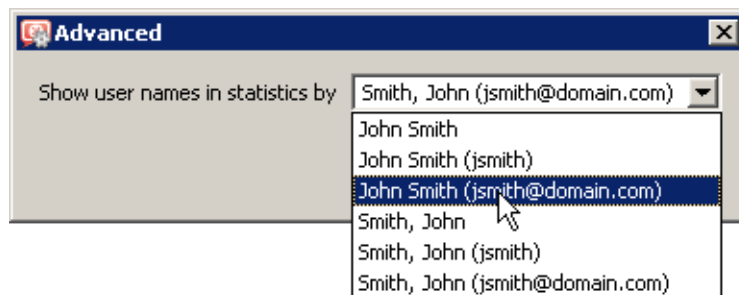


Figure 21.2 Kerio StaR advanced options

The *Show user names in statistics by...* option enables select a mode of how users and their names will be displayed in individual user statistics. Full names can be displayed as *first name second name* or *second name, first name*. Optionally, it is also possible to view full names followed by username without or with domain (if *Active Directory* mapping is used).

Statistics and quota accounting periods

Accounting period is a time period within which information of transferred data volume and other information is gathered. Statistics enable generating of weekly and monthly overviews. In *Accounting Periods*, it is possible to define starting days for weekly and monthly periods (for example, in statistics, a month can start on day 15 of the civil month and end on day 14 of the following civil month).

The parameter of first day of monthly period also sets when the monthly transferred data counter of individual users will be set to zero (for monthly quota details, see chapter 15.2).

Note: Setting of accounting period does not affect log rotation (see chapter 22.1).

Statistics and quota exceptions

On the *Exceptions* tab, it is possible to define exceptions for statistics and for transferred data quota.

This feature helps avoid gathering of irrelevant information. Thus, statistics are kept transparent and gathering and storage of needless data is avoided.

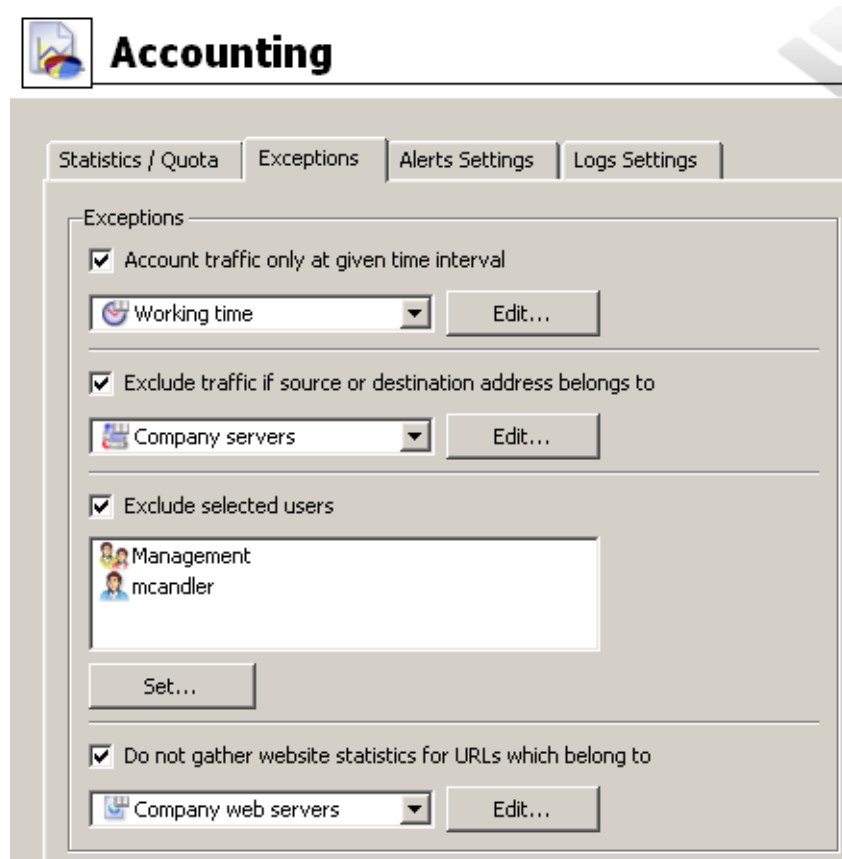


Figure 21.3 Statistics and quota exceptions

Usage of individual exceptions:

Time interval

Define a time period when information will be gathered and included in statistics and quota (e.g. only in working hours). Without this period, no traffic will be included in the statistics and in the quota neither.

For details on time intervals, see chapter [14.2](#).

IP addresses

Define IP addresses of hosts which will be excluded from the statistics and to which quota will not be applied.

The selected group may include both local or Internet IP addresses. If any of these IP addresses belongs to the local network, bear in mind that no traffic of the host will be included in the statistics or the quota. In case of addresses of Internet servers, traffic of local users with the server will not be accounted in the statistics or any user quota.

For details on IP groups, see chapter [14.1](#).

Users and groups

Select users and/or user groups which will be excluded from the statistics and no quota will be applied to them. This setting has the highest priority and overrules any other quota settings in user or group preferences.

For details on users and groups, see chapter [15](#).

Web Pages

Define a URL group. Connections to web sites with these URLs will not be accounted. Such exception can be used for example to exclude the own corporate web servers from the statistics (connection to corporate websites is usually considered a work-related activity) or to exclude ads — connection to certain pages may download advertisements automatically, it is not the user's request. For this purpose, you can use the predefined URL group *Ads/banners* (see chapter [14.4](#)).

Wildcards can be used in URL groups items. This implies that it is possible to define exceptions for particular pages or for all pages on a particular server, all web servers in a domain, etc. For details on URL groups, refer to chapter [14.1](#).

URL exceptions can be applied only to unsecured web pages (the *HTTP* protocol). Connections to secured pages (the *HTTPS* protocol) are encrypted and URL of such pages cannot be detected.

Note: Unlike in case of exceptions described above, data transferred within connections to such web pages will be included in the quota.

21.3 Connection to StaR and viewing statistics

To view statistics, user must authenticate at the *WinRoute's* web interface first. User (or the group the user belongs to) needs rights for statistics viewing — see chapter [15.1](#). *StaR* can be accessed by several methods, depending on whether connecting from the *WinRoute* host (locally) or from another host (remotely).

Note: For details on the *WinRoute's* web interface, see chapter [11.2](#).

Accessing the statistics from the WinRoute host

On the *WinRoute* host, the *StaR* may be opened as follows:

- By using the *Internet Usage Statistics* link available in the *WinRoute Engine Monitor* context menu (opened by the corresponding icon in the notification area — see chapter [2.10](#)).
- By using the *Internet Usage Statistics* link under *Start* → *Programs* → *Kerio* → *WinRoute Firewall*.

Both links open the unsecured *StaR* interface directly on the local host (by default <http://localhost:4080/star>) using the default web browser.

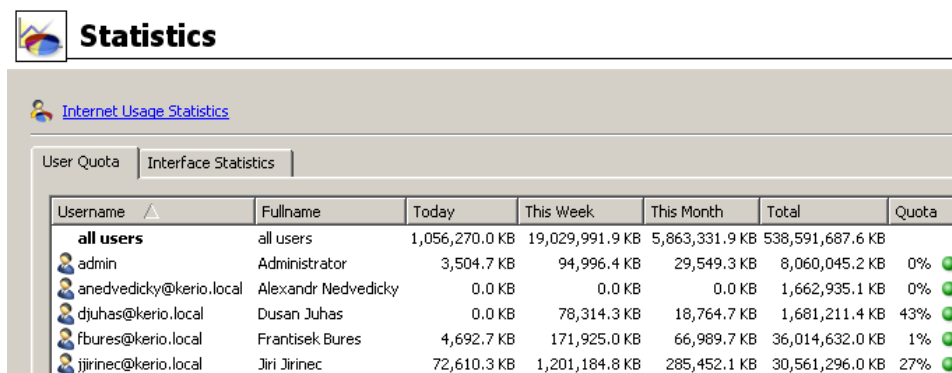
Note: Within local systems, secured traffic would be useless and the browser would bother user with needless alerts.

Remote access to the statistics

It is also possible to access the statistics remotely, i.e. from any host which is allowed to connect to the *WinRoute* host and the web interface's ports, by using the following methods:

- If the host is connected to *WinRoute* by the *Administration Console*, the *Internet Usage Statistics* link available under *Status* → *Statistics* can be used. This link opens the secured *StaR* interface for statistics in the default web browser.

Note: URL for this link consists of the name of the server and of the port of the secured Web interface defined in the configuration (see chapter 11.1). This guarantees function of the link from the *WinRoute* host and from the local network. To make *Internet Usage Statistics* link work also for remote administration over the Internet, name of the particular server must be defined in the public DNS (with the IP address of the particular firewall) and traffic rules must allow access to the port of the secured Web interface(4081 by default).



The screenshot shows the 'Statistics' page in the Administration Console. It features a navigation bar with 'User Quota' and 'Interface Statistics' tabs. Below the tabs is a table with columns for Username, Fullname, Today, This Week, This Month, Total, and Quota. The table lists several users and their usage statistics.

Username	Fullname	Today	This Week	This Month	Total	Quota
all users	all users	1,056,270.0 KB	19,029,991.9 KB	5,863,331.9 KB	538,591,687.6 KB	
admin	Administrator	3,504.7 KB	94,996.4 KB	29,549.3 KB	8,060,045.2 KB	0%
anedvedicky@kerio.local	Alexandr Nedvedicky	0.0 KB	0.0 KB	0.0 KB	1,662,935.1 KB	0%
djuhas@kerio.local	Dusan Juhas	0.0 KB	78,314.3 KB	18,764.7 KB	1,681,211.4 KB	43%
fbures@kerio.local	Frantisek Bures	4,692.7 KB	171,925.0 KB	66,989.7 KB	36,014,632.0 KB	1%
jjirinec@kerio.local	Jiri Jirinec	72,610.3 KB	1,201,184.8 KB	285,452.1 KB	30,561,296.0 KB	27%

Figure 21.4 Link for viewing of the statistics in the Administration Console (status → Statistics)

- At <https://server:4081/star> or <http://server:4080/star> This URL works for the *StaR* only. If the user has not appropriate rights to view statistics, an error is reported.
- At <https://server:4081/> or <http://server:4080/>. This is the primary URL of the *WinRoute*'s web interface. If the user possesses appropriate rights for stats viewing, the *StaR* welcome page providing overall statistics (see below) is displayed. Otherwise, the *My Account* page is opened (this page is available to any user).

Warning

In case of access via the Internet (i.e. from a remote host) it is recommended to use only the secured version of the web interface. The other option would be too risky.

Updating data in StaR

First of all, the *StaR* interface is used for gathering of statistics and creating of reviews for certain periods. To *WinRoute*, gathering and evaluation of information for *StaR* means processing of large data volumes. To reduce load on the firewall, data for *StaR* is updated approximately once in an hour. The top right corner of each *StaR* page displays information about when the last update of the data was performed.

For these reasons, the *StaR* interface is not useful for real-time monitoring of user activity. For these purposes, you can use the *Active Hosts* section in the *Administration Console* (see chapter [19.1](#)).

Chapter 22

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. Each line starts with a time mark in brackets (date and time when the event took place, in seconds). This mark is followed by an information, depending on the log type. If the record includes a URL, it is displayed as a hypertext link. Follow the link to open the page in your default browser.

Optionally, records of each log may be recorded in files on the local disk⁷ and/or on the *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 includes an `.idx` file, i.e. an indexing file allowing faster access to the log when displayed in *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.

Administration Console allows to save a selected log (or its part) in a file as plaintext or in HTML. The log saved can be analyzed by various tools, published on web servers, etc.

22.1 Log settings

Log parameters (file names, rotation, sending to a *Syslog* server) can be set in the *Configuration* → *Accounting* 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.

Note: If the log is not saved in a file on the disk, only records generated since the last login to *WinRoute Firewall Engine* will be shown in the *Administration Console*. After logout (or closing of *Administration Console*), the records will be lost.

⁷ Local disk is a disk of the computer where *WinRoute* is installed, not a computer where *Administration Console* is running!

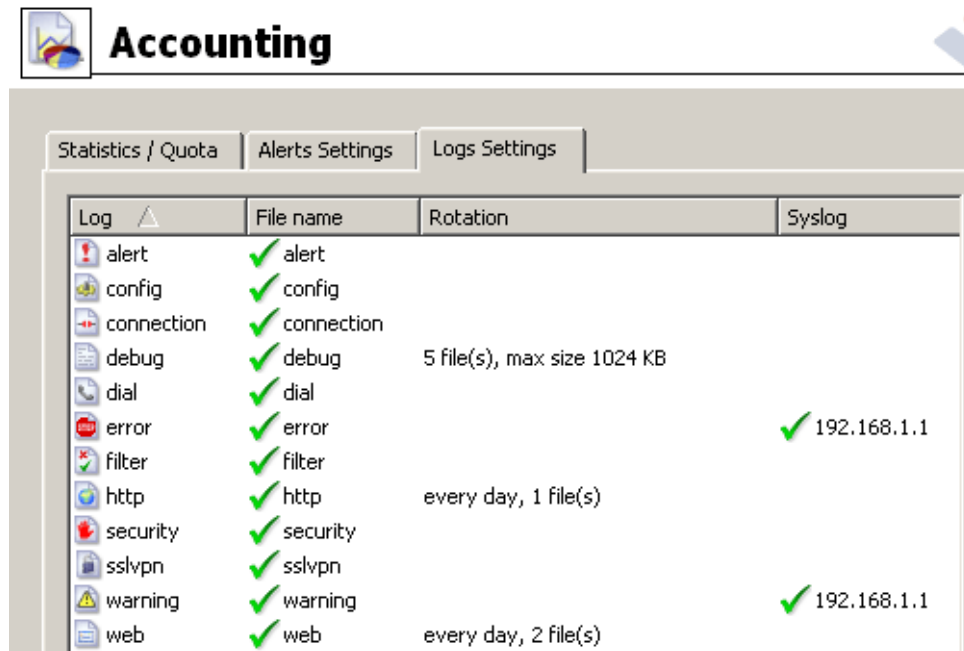


Figure 22.1 Log settings

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.

Weekly rotation takes effect on Sunday nights. Monthly rotation is performed at the end of the month (in the night when one month ends and another starts).

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 megabytes (MB).

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.

Note:

1. 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.
2. Setting of statistics and quotas accounting period does not affect log rotation (see chap-

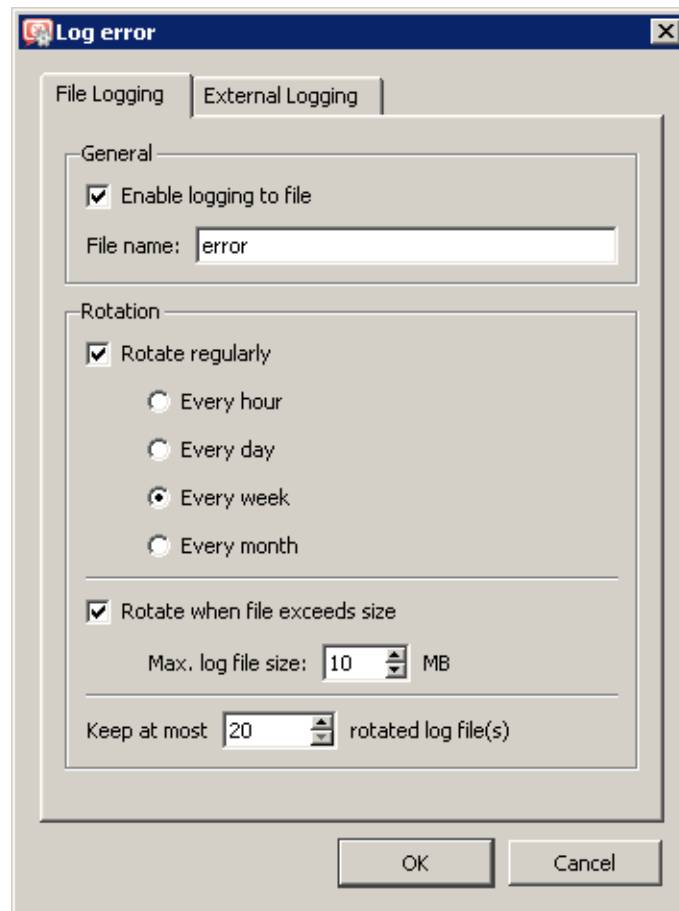


Figure 22.2 File logging settings

ter 21.2). Rotation follows the rules described above.

Syslog Logging

Parameters for logging to a *Syslog* can be defined in the *External Logging* tab.

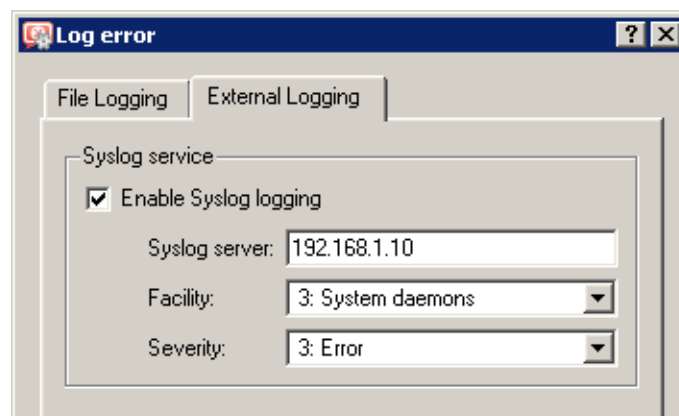


Figure 22.3 Syslog settings

Enable Syslog logging

Enable/disable logging to a *Syslog* server.

If this option is disabled, none of the following parameters and settings will be available.

Syslog server

DNS name or 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).

22.2 Logs Context Menu

When you right-click inside any log window, a context menu will be displayed where you can choose several functions or change the log's parameters (view, logged information).

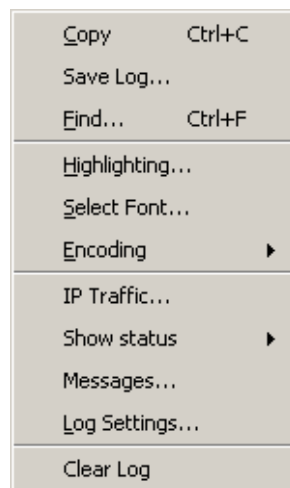


Figure 22.4 Logs Context Menu

Copy

Copies the selected text onto the clipboard. A key shortcut from the operating system can be used (*Ctrl+C* or *Ctrl+Insert* in *Windows*).

Save log

This option saves the log or selected text in a file as plaintext or in HTML.

Hint

This function provides more comfortable operations with log files than a direct access to log files on the disk of the computer where *WinRoute* is installed. Logs can be saved even if *WinRoute* is administered remotely.

The *Save log* option opens a dialog box where the following optional parameters can be set:

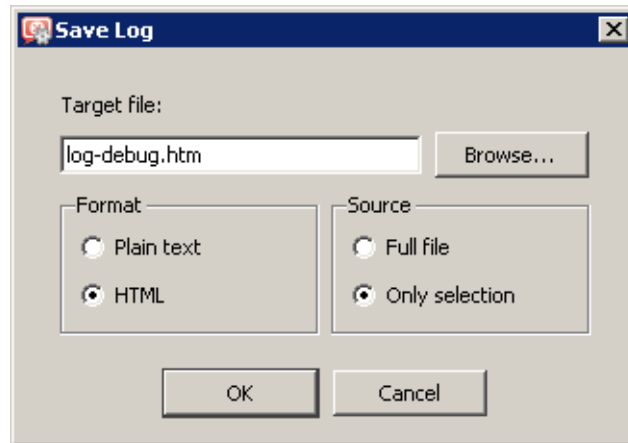


Figure 22.5 Saving a log to a file

- *Target file* — name of the file where the log will be saved. By default, a name derived from the file name is set. The file extension is set automatically in accordance with the format selected.
- *Format* — logs can be saved as plaintext or in HTML. If the HTML format is used, colors will be saved for the lines background (see section *Highlighting*) and all URLs will be saved as hypertext links.
- *Source* — either the entire log or only a part of the text selected can be saved. Bear in mind that in case of remote administration, saving of an entire log may take some time.

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.

During the first lookup (when switched to the log window), the log is searched through from the top (or the end, depending on the lookup direction set). Further search starts from the marked text (marked by mouse or as a result of the recent search).

Highlighting

Highlighting may be set for logs meeting certain criteria (for details, see below).

Select font

Within this dialog you can select a font of the log printout. All fonts installed on the host with the *Administration Console* are available.

Encoding

Coding that will be used for the log printout in *Administration Console* can be selected in this section. *UTF-8* is used by default.

Hint

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. These parameters can also be set in the *Log settings* tab under *Configuration* → *Accounting*. For details, refer to chapter [22.1](#).

Clear 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.

Note: If a user with read rights only is connected to *WinRoute* (see chapter [15.1](#)), the *Log settings* and *Clear log* options are missing in the log context menu. Only users with full rights can access these functions.

Log highlighting

For better reference, it is possible to set highlighting for logs meeting certain criteria. Highlighting is defined by special rules shared by all logs. Seven colors are available (plus the background color of unhighlighted lines), however, number of rules is not limited.

Use the *Highlighting* option in the context pop-up menu of the corresponding log to set highlighting parameters.

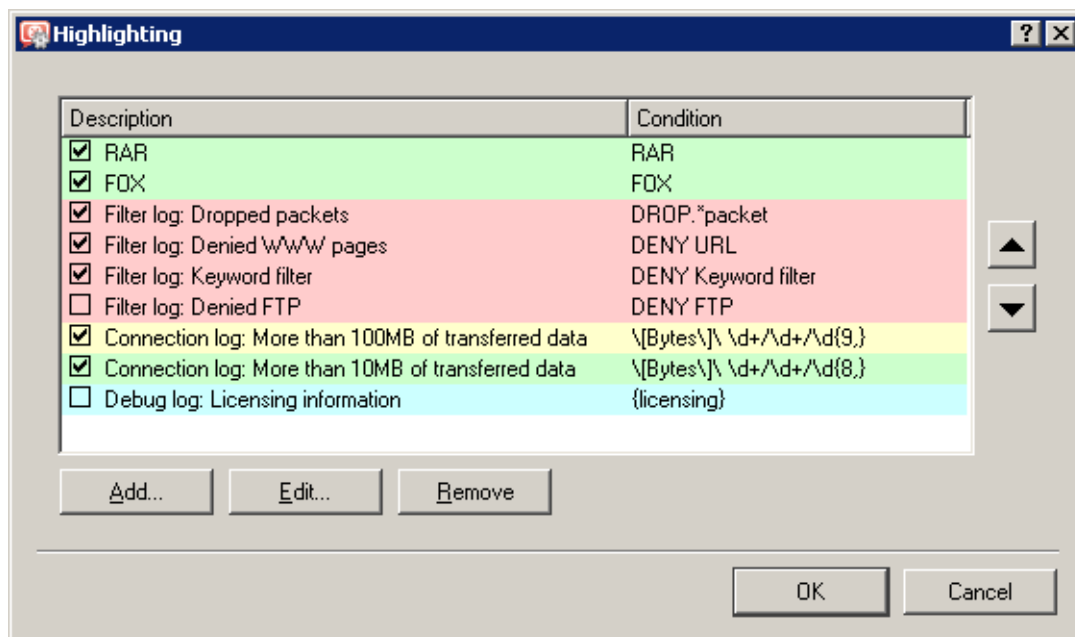


Figure 22.6 Log highlighting settings

Highlighting rules are ordered in a list. The list is processed from the top. The first rule meeting the criteria stops other processing and the found rule is highlighted by the particular color. Thanks to these features, it is possible to create even more complex combinations of rules, exceptions, etc. In addition to this, each rule can be “disabled” or “enabled” for as long as necessary.

Use the *Add* or the *Edit* button to (re)define a highlighting rule.

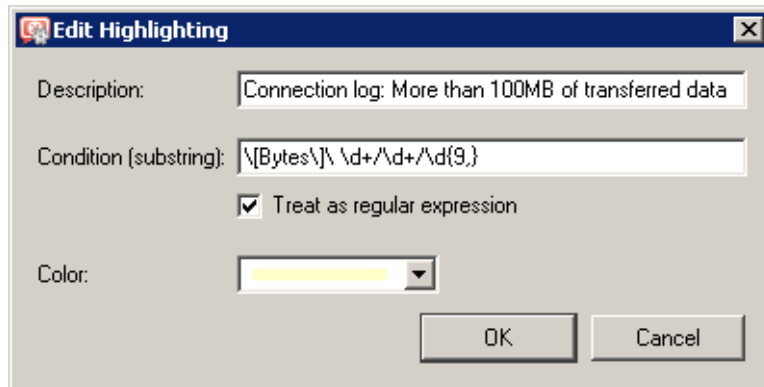


Figure 22.7 Highlighting rule definition

Each highlighting rule consists of a condition and a color which will be used to highlight lines meeting the condition. Condition can be specified by a substring (all lines containing the string will be highlighted) or by a so called regular expression (all lines containing one or multiple strings matching the regular expression will be highlighted).

The *Description* item is used for reference only. It is recommended to describe all created rules well (it is recommended to mention also the name of the log to which the rule applies).

Note: Regular expression is such expression which allows special symbols for string definition. *WinRoute* accepts all regular expressions in accordance with the POSIX standard.

For detailed instructions contact Kerio technical support. For detailed information, refer for example to

<http://www.gnu.org/software/grep/>

The Debug log advanced settings

Special options are available in the *Debug* log context menu. These options are available only to users with full administration rights (see chapter [15.1](#)).

Options of information which can be monitored by the *Debug* log are addressed in chapter [22.6](#).

22.3 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.

Hint

Email and SMS alerts can be set under *Configuration* → *Accounting*. All sent alerts can be viewed in the *Status* → *Alert messages* section (for details, see chapter [19.4](#)).

22.4 Config Log

The *Config* log stores a complete communication history between *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 *Config* window contains three log types:

1. *Information about user logins/logouts to/from the WinRoute's administration*

Example

```
[18/Apr/2008 10:25:02] james - session opened  
for host 192.168.32.100
```

```
[18/Apr/2008 10:32:56] james - session closed  
for host 192.168.32.100
```

- [18/Apr/2008 10:25:02] — date and time when the record was written to the log
 - jsmith — the login name of the user logged in the *WinRoute* administration
 - 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 *Administration Console* closed)
-

2. *Configuration database changes*

Changes performed in the *Administration Console*. A simplified form of the SQL language is used when communicating with the database.

Example

```
[18/Apr/2008 10:27:46] james - 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/2008 10:27:46] — date and time when the record was written
 - james — the login name of the user logged in the *WinRoute* administration
 - 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 changes in configuration

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/2008 12:06:03] Admin - New traffic policy set:
[18/Apr/2008 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 (see chapter [7.1](#)) is marked with `default` instead of the positional number.

22.5 Connection Log

The *Connection* log gathers information about traffic matching traffic rules with the *Log matching connections* enabled (see chapter [7](#)) or meeting certain conditions (e.g. log of *UPnP* traffic — see chapter [18.2](#)).

How to read the Connection Log?

```
[18/Apr/2008 10:22:47] [ID] 613181 [Rule] NAT
[Service] HTTP [User] james
[Connection] TCP 192.168.1.140:1193 -> hit.google.com:80
[Duration] 121 sec [Bytes] 1575/1290/2865 [Packets] 5/9/14
```

- [18/Apr/2008 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 [14.3](#)), 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 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* module cache (see chapter [8.1](#)), 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).

22.6 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, you only need to display information relating to a particular service or function. 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.

Selection of information monitored by the Debug log

The window's context menu for the *Debug* log includes (see chapter [22.2](#)) further options for advanced settings of the log and for an on-click one-time view of status information.

Note: These options are available only to users with full administration rights for *WinRoute* (see chapter [15.1](#)).

IP Traffic

This function enables monitoring of packets according to the user defined log expression.

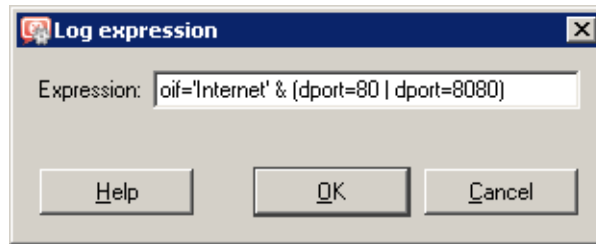


Figure 22.8 Expression for traffic monitored in the debug log

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. Logging of IP traffic can be cancelled by leaving or setting the *Expression* entry blank.

Show status

A single overview of status information regarding certain *WinRoute* components. This information can be helpful especially when solving problems with *Kerio Technologies* technical support.

Messages

This feature allows advanced monitoring of functioning of individual *WinRoute* modules. This information may be helpful when solving issues regarding *WinRoute* components and/or certain network services.

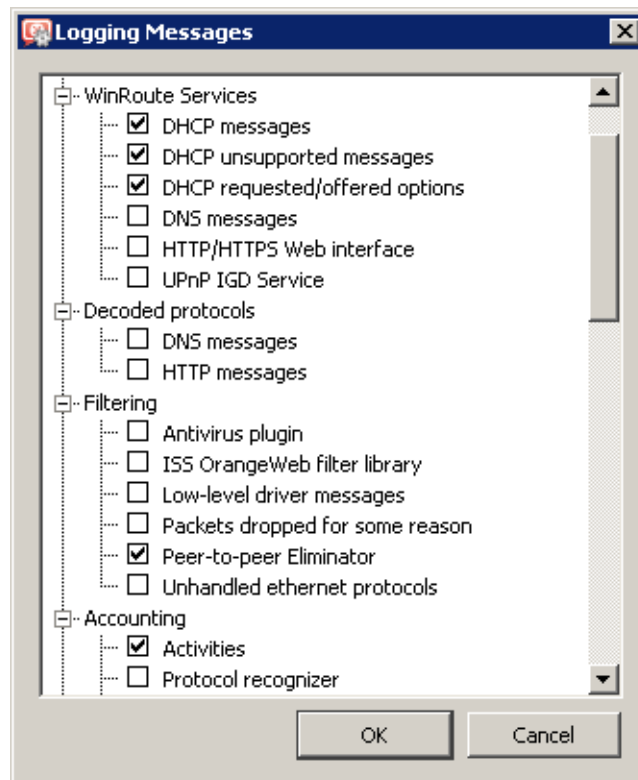


Figure 22.9 Selection of information monitored by the Debug log

- *WAN / Dial-up messages* information about dialed lines (request dialing, auto disconnection down-counter),
- *Filtering* — logs proving information on filtering of traffic passing through *WinRoute* (antivirus control, website classification, detection and elimination of *P2P* networks, dropped packets, etc.),
- *Accounting* — user authentication and monitoring of their activities (protocol recognition, statistics and reporting, etc.),
- *WinRoute services* — protocols processed by *WinRoute* services (*DHCP server*, the *DNS* module, web interface, and *UPnP* support),
- *Decoded protocols* — logs of specific protocols (*HTTP* and *DNS*),
- *Miscellaneous* — other information on miscellaneous topics (e.g. packet processing by the *Bandwidth Limiter*, Internet connection, HTTP cache, used licenses, update check, employment of dynamic DNS, etc.),
- *Protocol inspection* — reports from individual *WinRoute*'s protocol inspectors (sorted by protocol),
- *Kerio VPN* — detailed information on traffic within *Kerio VPN* (VPN tunnels, VPN clients, encryptions, exchange of routing information, web server for *Clientless SSL-VPN*, etc.).

22.7 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 5 or directly from the operating system)

```
[15/Mar/2008 15:09:27] Line "Connection" dialing,
console 127.0.0.1 - Admin
```

```
[15/Mar/2008 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 5). If the line is dialed from the *Administration Console*, the log provides this additional information

- where the line was dialed from (console — *Administration Console*,
- IP address of the client (i.e. IP address of the *Administration Console*),
- 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/2008 15:29:18] Line "Connection" hang-up,
console 127.0.0.1 - Admin
```

```
[15/Mar/2008 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/2008 15:42:51] Line "Connection" dropped,  
connection time 00:17:07, 1519 bytes received,  
2504 bytes transmitted
```

The items are the same as in the previous case (the second item — the disconnected report).

4. Requested dialing (as a response to a DNS query)

```
[15/Mar/2008 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/2008 15:51:38] Line "Connection" successfully connected
```

The first log item is recorded upon reception of a DNS request (the *DNS* module 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.

Another event is logged upon a successful connection (i.e. when the line is dialed, upon authentication on a remote server, etc.).

5. On-demand dialing (response to a packet sent from the local network)

```
[15/Mar/2008 15:53:42] Packet  
TCP 192.168.1.3:8580 -> 212.20.100.40:80  
initiated dialing of line "Connection"
```

```
[15/Mar/2008 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:

- description of the packet (protocol, source IP address, destination port, destination IP address, destination port),
- name of the line to be dialed.

Another event is logged upon a successful connection (i.e. when the line is dialed, upon authentication on a remote server, etc.).

6. Connection error (e.g. error at the modem was detected, dial-up was disconnected, etc.)

```
[15/Mar/2008 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/2008 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).

22.8 Error Log

The *Error* log displays information about serious errors that affect the functionality of the entire firewall. *WinRoute* administrator should check this log regularly and fix detected problems as soon as possible. Otherwise, users might have problems with some services or/and serious security problems might arise.

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 disk 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 (x) 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 loop in the configuration of the *DNS* module 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 certificates, etc.)
- 8000-8099 — HTTP cache errors (errors when reading / writing cache files, not enough space for cache, etc.)

- 8100–8199 — errors of the *Kerio Web Filter* module
- 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 [26](#) or to <http://www.kerio.com/>.

22.9 Filter Log

This log gathers information on web pages and objects blocked/allowed by the HTTP and FTP filters (see chapters [12.2](#) and [12.5](#)) and on packets matching traffic rules with the *Log matching packets* option enabled (see chapter [7](#)) or meeting other conditions (e.g. logging of *UPnP* traffic — see chapter [18.2](#)).

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/2008 13:39:45] ALLOW URL 'McAfee update'  
192.168.64.142 james HTTP GET  
http://update.kerio.com/nai-antivirus/datfiles/4.x/dat-4258.zip
```

- [18/Apr/2008 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
-

Packet log example

```
[16/Apr/2008 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/2008 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 matched by the packet
 - packet to — packet direction (either to or from a particular interface)
 - LAN — interface name (see chapter 5 for details)
 - proto: — transport protocol (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: — acknowledgement 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)
-

22.10 Http log

This log contains all HTTP requests that were processed by the HTTP inspection module (see section 14.3) or by the built-in proxy server (see section 8.4). 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/>). To enable or disable the *Http* log, or to choose its format, go to *Configuration* → *Content Filtering* → *HTTP Policy* (refer to section 12.2 for details).

Note:

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 chapter 22.9), if the *Log* option is enabled in the particular rule (see section 12.2).
2. The *Http* log is intended to be processed by external analytical tools. The *Web* log (see below) is better suited to be viewed by the *WinRoute* administrator.

— An example of an HTTP log record in the Apache format —

```
192.168.64.64 - jflyaway  
[18/Apr/2008:15:07:17 +0200]  
"GET http://www.kerio.com/ HTTP/1.1" 304 0 +4
```

- 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/2008:15:07:17 +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 transferred through the connection
-

— An example of Http log record in 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 milliseconds since January 1st, 1970)
 - 0 — download duration (not measured in *WinRoute*, always set to zero)
 - 192.168.64.64 — IP address of the client (i.e. of the host from which the client is connected to the website)
 - 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 — return code of the HTTP protocol
 - 0 — transferred 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
-

22.11 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 were captured by the *Anti-spoofing* module (packets with invalid source IP address — see section [17.2](#) for details)

Example

[17/Jul/2008 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, i.e. sent via the interface, or to, i.e. received via the interface)
 - LAN — interface name (see chapter 5 for details)
 - proto: — transport protocol (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: — acknowledgement 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)
-

2. FTP protocol parser log records

Example 1

[17/Jul/2008 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/2008 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 on user quotas, refer to chapters [15.1](#) and [10.1](#).

4. Information about the start and shutdown of the WinRoute Firewall Engine

a) Engine Startup:

```
[17/Dec/2008 12:11:33] Engine: Startup.
```

b) Engine Shutdown:

```
[17/Dec/2008 12:22:43] Engine: Shutdown.
```

22.12 Sslvpn Log

In this log, operations performed in the *Clientless SSL-VPN* interface are recorded. Each log line provides information about an operation type, name of the user who performed it and file associated with the operation.

Example

```
[17/Mar/2008 08:01:51] Copy File: User: jsmith@company.com  
File: '\\server\data\www\index.html'
```

The *Clientless SSL-VPN* interface and the corresponding record is available in *WinRoute* is for *Windows* only.

22.13 Warning Log

The *Warning* log displays warning messages about errors of little significance. 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 help if for example a user is complaining that certain services are not working.

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 require user authentication, but the WWW interface is not enabled)

- 3000–3999 — warning from individual *WinRoute* modules (e.g. DHCP server, anti-virus check, user authentication, etc.)
- 4000–4999 — license warnings (subscription expiration, forthcoming expiration of *WinRoute's* license, *Kerio Web Filter* 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/2008 15:00:51] (3004) Authentication subsystem warning:
Kerberos 5 auth: user james@company.com not authenticated

[15/Apr/2008 15:00:51] (3004) Authentication subsystem warning:
Invalid password for user admin

[16/Apr/2008 10:53:20] (3004) Authentication subsystem warning:
User jflyaway 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.

22.14 Web Log

This log contains all HTTP requests that were processed by the HTTP inspection module (see section 14.3) or by the built-in proxy server (see section 8.4). 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. In addition to each URL, name of the page is provided for better reference.

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/2008 10:29:51] 192.168.44.128 james
"Kerio Technologies" http://www.kerio.com/

- [24/Apr/2008 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" — 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

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 8 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 [10.1](#)) are used for authentication.
- Statistics about VPN tunnels and VPN clients can be viewed in *WinRoute* (refer to chapter [20.2](#)).

23.1 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 details, refer to chapters [23.2](#) and [23.3](#).

VPN server is available in the *Interfaces* tab of the *Configuration* → *Interfaces* section as a special interface.

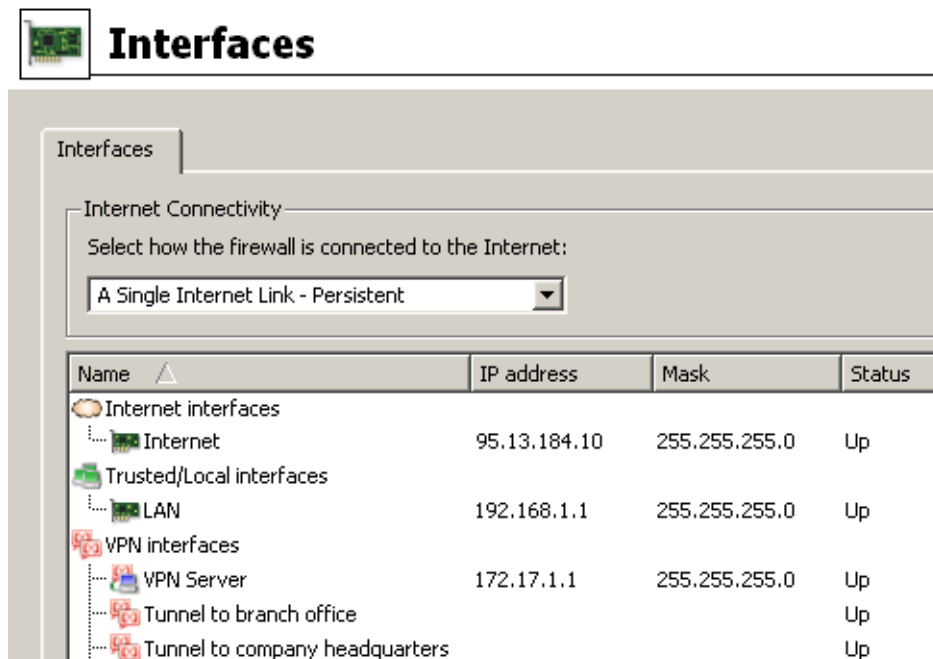


Figure 23.1 Viewing VPN server in the table of interfaces

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.

VPN subnet and SSL certificate

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.

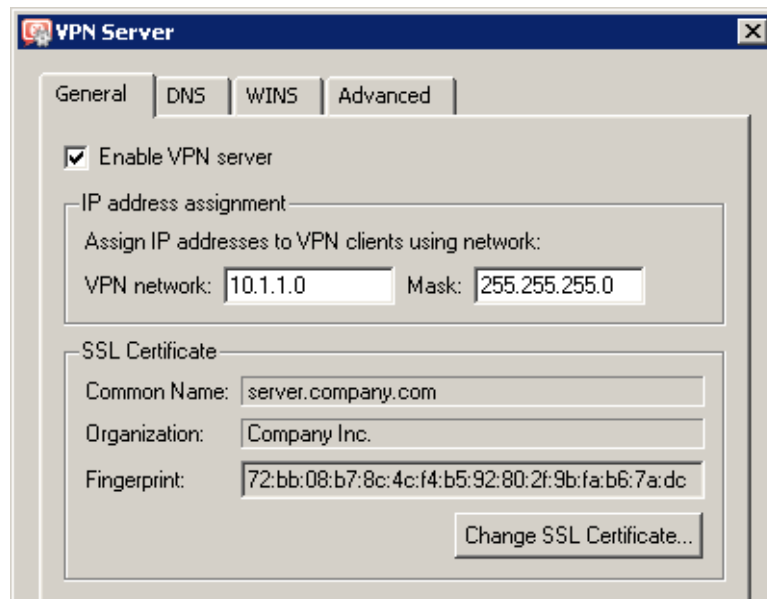


Figure 23.2 VPN server settings — basic parameters

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. After the first change in VPN server settings, the recently used network is used (the automatic detection is not performed again).

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.

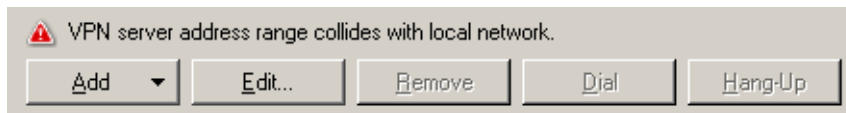


Figure 23.3 VPN server — detection of IP collision

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 startup 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).

3. VPN clients can also be assigned IP addresses according to login usernames. For details, see chapter [15.1](#).

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 [23.3](#)). The VPN server in *WinRoute* uses the standard SSL certificate.

When defining a VPN tunnel, it is necessary to send the local endpoint's certificate fingerprint to the remote endpoint and vice versa (mutual verification of identity — see chapter [23.3](#)).

Hint

Certificate fingerprint can be saved to the clipboard and pasted to a text file, email message, etc.

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. The certificate created is saved in the `sslcert` subdirectory of the *WinRoute* installation directory as `vpn.crt` and the particular private key is saved at the same location as `vpn.key`.

Methods used for creation and import of SSL certificates are described thoroughly in chapter [11.1](#).

Note: If you already have a certificate created by a certification authority especially for your server (e.g. 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 configuration for VPN clients

To allow VPN clients to access to local hosts using the hostnames, they need at least one local DNS server.

The *WinRoute*'s VPN server allows for the following options of DNS server configuration:

- *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* module; see chapter [8.1](#)). This is the default option in case that the *DNS* module is enabled in *WinRoute*.

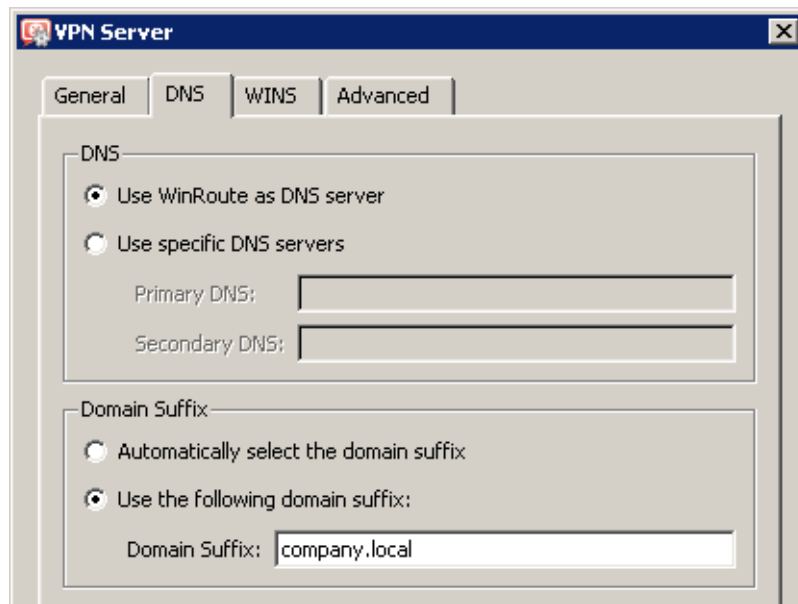


Figure 23.4 VPN server settings — specification of DNS servers for VPN clients

If the *DNS* module is already used as a DNS server for local hosts, it is recommended to use it also for VPN clients. The *DNS* module provides the fastest responses to client DNS requests and possible collision (inconsistency) of DNS records will be avoided.

- *Specific DNS servers* — primary and optionally also secondary DNS server will be set for VPN clients.

If another DNS server than the *DNS* module in *WinRoute* is used in the local network, use this option.

DNS domain extension is also assigned to VPN clients. Domain extension specifies local domain. If the VPN client's extension matches a local domain of the networks it connects to, it can use hostnames within this network (e.g. `server`). Otherwise, full name of the host including domain is required (e.g. `server.company.local`).

DNS extension can be also resolved automatically or set manually:

- Automatic resolution can be used in case that the host belongs to the *Active Directory* domain and/or in case that firewall users are authenticated in this domain (see chapter [15.1](#)).
- DNS domain must be specified in case that it is a *Windows NT* domain or a network without a domain, or in case that another domain extension is desirable (e.g. when multiple *Active Directory* are mapped).

Note: DNS servers assigned by the VPN server will be used as primary/secondary DNS server(s) on the client host. This implies that *all* DNS queries from the client host will be sent to these servers. However, in most cases this kind of “redirection” has no side effects. Upon closing of the VPN connection, the original DNS configuration will be recovered.

WINS configuration for VPN clients

The [WINS](#) service is used for resolution of hostnames to IP addresses within *Microsoft Windows* networks. Assigning of a WINS server address then allows VPN clients browse in LAN hosts (*Network Neighborhood / My Network Places*).

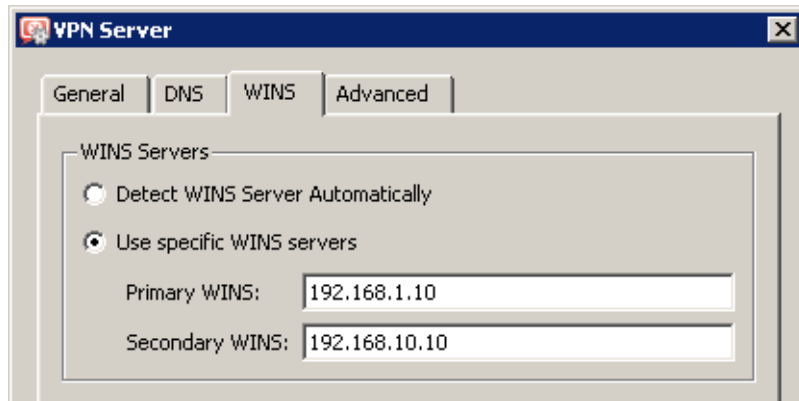


Figure 23.5 VPN server settings — specification of WINS servers for VPN clients

WinRoute can detect WINS servers either automatically (using its host configuration) or use specified addresses of primary or/and secondary WINS server(s). Automatic configuration can be used if you are sure that WINS servers on the *WinRoute* host are set correctly.

Advanced Options

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).

Note:

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 [22.8](#)) 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.

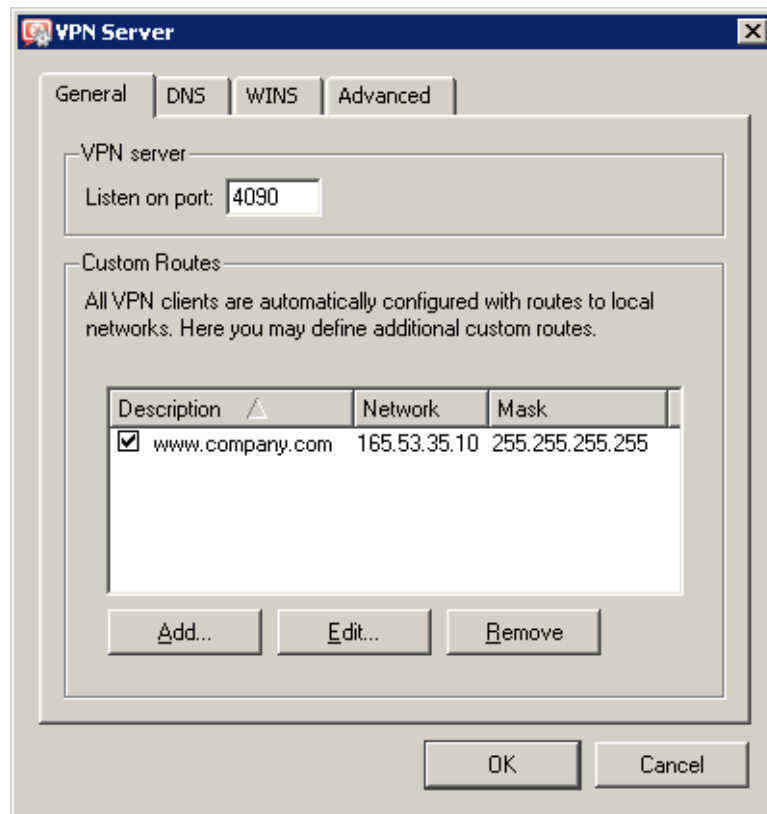


Figure 23.6 VPN server settings — server port and routes for VPN clients

Custom Routes

Other networks to which a VPN route will be set for the client can be specified in this section. By default, routes to all local subnets at the VPN server's side are defined — see chapter 23.4).

— **Hint** —

Use the 255.255.255.255 network mask to define a route to a certain host. This can be helpful for example when a route to a host in the demilitarized zone at the VPN server's side is being added.

23.2 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 15.115.1).
- 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 4 and 4.6). Be aware of this fact when deciding on what license type should be purchased (or whether an add-on for upgrade to a higher number of users for the license should be bought).

— **Hint:** —
 VPN clients correctly connected to the firewall can be overviewed in the *Administration Console*, section *Status* → *VPN clients*. For details, see chapter 19.3.

Basic configuration of traffic rules for VPN clients

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Trusted/Local	Firewall All VPN clients Trusted/Local	Any	✓	

Figure 23.7 Common traffic rules for VPN clients

- 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. A few alternatives of the restrictions settings within *Kerio VPN* are focused in chapter 23.5.

Note:

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 7.1.
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 information about traffic rules, refer to chapter 7.

23.3 Interconnection of two private networks via the Internet (VPN tunnel)

WinRoute with support for VPN (VPN support is included in the typical installation) 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 4).

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 23.1.

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.

Add VPN Tunnel

General | Advanced

General

Name of the tunnel:
Tunnel to company headquarters

Configuration

Actively connect to the remote endpoint
Select this if you can specify the hostname or IP address of the remote endpoint and the remote endpoint can accept incoming connections.
Remote endpoint hostname or IP address:
newyork.company.com

Passively accept the connection only
Select this if the remote endpoint uses dynamic IP address or is unable to accept incoming connections.

Settings for remote endpoint

Local endpoint's SSL certificate fingerprint:
da:27:e5:7f:10:18:0f:af:ae:aa:cb:44:b8:17:43:05

Remote endpoint's SSL certificate fingerprint:
72:bb:08:b7:8c:4c:f4:b5:92:80:2f:9b:fa:b6:7a:dc

The authenticity of the remote endpoint during the creation of a tunnel session is verified by checking its public SSL certificate - the fingerprint of the certificate received from the remote endpoint must match the fingerprint entered here.

Detect remote certificate...

Figure 23.8 VPN tunnel configuration

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 [7.3](#)) and interface statistics (details in chapter [20.2](#)).

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*. Passive endpoint cannot detect remote certificate.

However, this method of fingerprint setting is quite insecure — a counterfeit certificate might be used. If a fingerprint of a false certificate is used for the configuration of the VPN tunnel, it is possible to create a tunnel for the false endpoint (for the attacker). Moreover, a valid certificate would not be accepted from the other side. Therefore, for security reasons, it is recommended to set fingerprints manually.

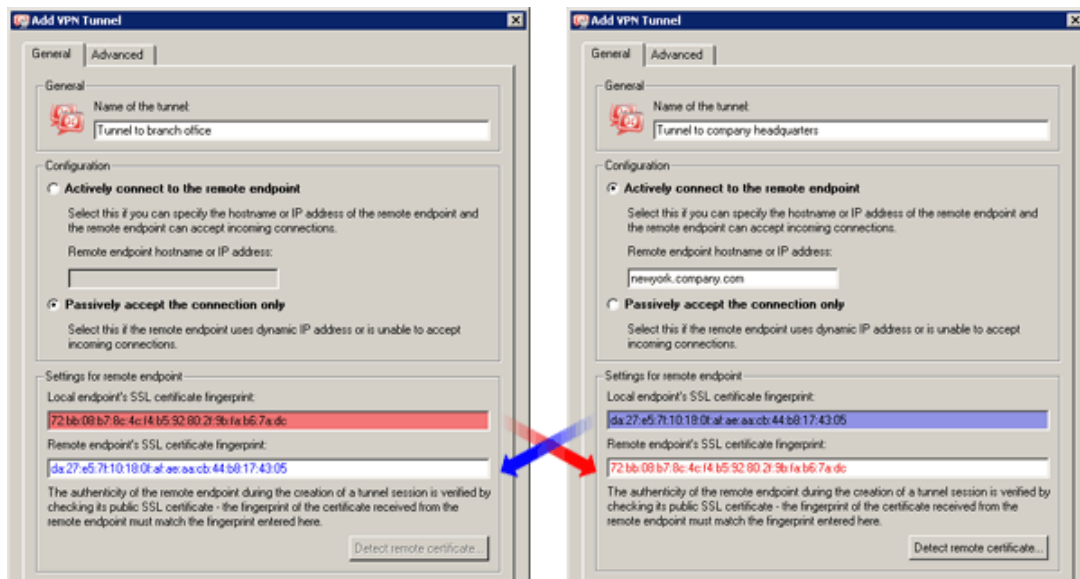


Figure 23.9 VPN tunnel — certificate fingerprints

DNS Settings

DNS must be set properly at both ends 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* module in *WinRoute* is used as the DNS server at both ends of the tunnel, DNS queries (for DNS rules, refer to chapter [8.1](#)) can be forwarded to hostnames in the corresponding domain of the *DNS* module at the other end of the tunnel. DNS domain (or subdomain) must be used at both sides of the tunnel.

Note: To provide correct forwarding of DNS queries sent from the *WinRoute* host (at any side of the VPN tunnel), it is necessary that these queries are processed by the *DNS* module. To achieve this, set the DNS server on each firewall's interface located to the local network "to its own" (i.e. use IP address of the very interface as the DNS server address).

Detailed guidance for the DNS configuration is provided in the example in chapter [23.5](#).

Routing settings

On the *Advanced* tab, you can set which method will be used to add routes provided by the remote endpoint of the tunnel to the local routing table as well as define custom routes to remote networks.

The *Kerio VPN* routing issue is described in detail in chapter [23.4](#).

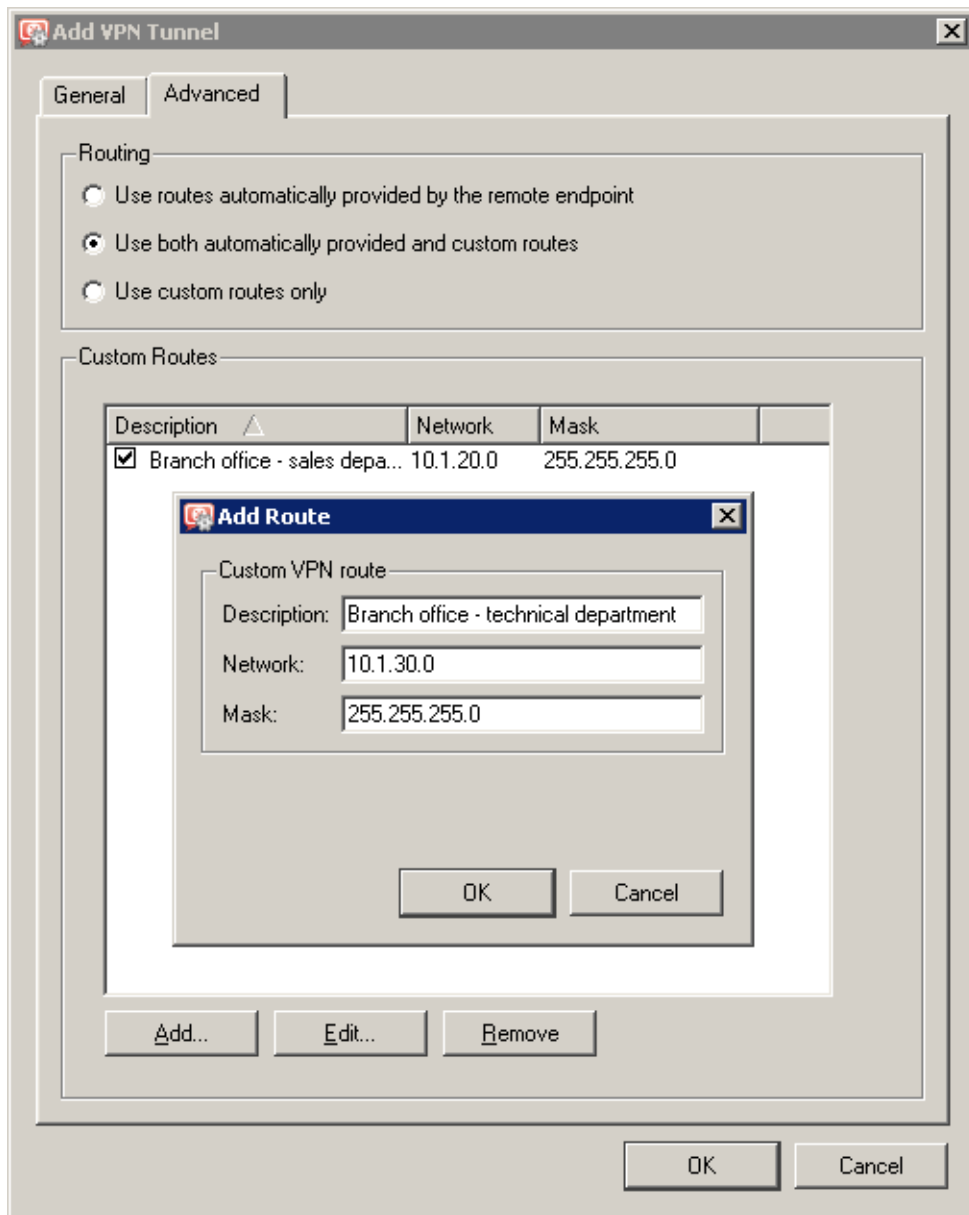


Figure 23.10 VPN tunnel's routing configuration

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*, i.e. when the corresponding traffic is allowed — see below).

VPN tunnels can be disabled by the *Disable* button. Both endpoints should be disabled while the tunnel is being disabled.

Note: VPN tunnels keep 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.

Traffic Policy Settings for VPN

Once the VPN tunnel is created, it is necessary to allow traffic between the LAN and the network connected by the tunnel and to allow outgoing connection for the *Kerio VPN* service (from the firewall to the Internet). If basic traffic rules are already created by the wizard (refer to chapter 23.2), simply add a corresponding VPN tunnel into the *Local Traffic* rule and the *Kerio VPN* service to the *Firewall traffic*. The resulting traffic rules are shown at figure 23.11.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Tunnel to branch office Trusted/Local	Firewall All VPN clients Tunnel to branch office Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	DNS HTTP HTTPS IMAP Kerio VPN POP3 Telnet	✓	

Figure 23.11 Traffic Policy Settings for VPN

Note:

- To keep examples in this guide as simple as possible, it is supposed that the *Firewall traffic* rule allows to access any service at the firewall (see figure 23.12). Under these conditions, it is not necessary to add the *Kerio VPN* service to the rule.

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Tunnel to branch office Trusted/Local	Firewall All VPN clients Tunnel to branch office Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.12 Common traffic rules for VPN tunnel

2. 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 of traffic rules are provided in chapter [23.5](#).

23.4 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). Thus, routing tables at both sides of the tunnel are always kept up-to-date.

Routing configuration options

Under usual circumstances, it is not necessary to define any custom routes — particular routes will be added to the routing tables automatically when configuration is changed at any side of the tunnel (or at the VPN server). However, if a routing table at any side of the VPN tunnel includes invalid routes (e.g. specified by the administrator), these routes are also interchanged. This might make traffic with some remote subnets impossible and overload VPN tunnel by too many control messages.

A similar problem may occur in case of a VPN client connecting to the *WinRoute's* VPN server.

To avoid the problems just described, it is possible to go to the VPN tunnel definition dialog (see chapter [23.3](#)) or to the VPN server settings dialog (refer to chapter [23.1](#)) to set which routing data will be used and define custom routes.

Kerio VPN uses the following methods to pass routing information:

- *Routes provided automatically by the remote endpoint* (set as default) — routes to remote networks are set automatically with respect to the information provided by the remote endpoint. If this option is selected, no additional settings are necessary unless problems regarding invalid routes occur (see above).
- *Both automatically provided and custom routes* — routes provided automatically are complemented by custom routes defined at the local endpoint. In case of any collisions, custom routes are used as prior. This option easily solves the problem where a remote endpoint provides one or more invalid route(s).
- *Custom routes only* — all routes to remote networks must be set manually at the local endpoint of the tunnel. This alternative eliminates adding of invalid routes provided by a remote endpoint to the local routing table. However, it is quite demanding from the administrator's point of view (any change in the remote network's configuration requires modification of custom routes).

Routes provided automatically

Unless any custom routes are defined, the following rules apply to the interchange of routing information:

- default routes as well as routes to networks with default gateways are not exchanged (default gateway cannot be changed for remote VPN clients and/or for remote end-points 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, every 10 minutes. The timeout starts upon each update (regardless of the update reason).

23.5 Example of Kerio VPN configuration: company with a filial office

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. The method described can be used in cases where no redundant routes arise by creating VPN tunnels (i.e. multiple routes between individual private networks). Configuration of VPN with redundant routes (typically in case of a company with two or more filials) is described in chapter [23.6](#).

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 London. 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.

Figure 23.13 provides a scheme of the entire system, including IP addresses and the VPN tunnels that will be built.

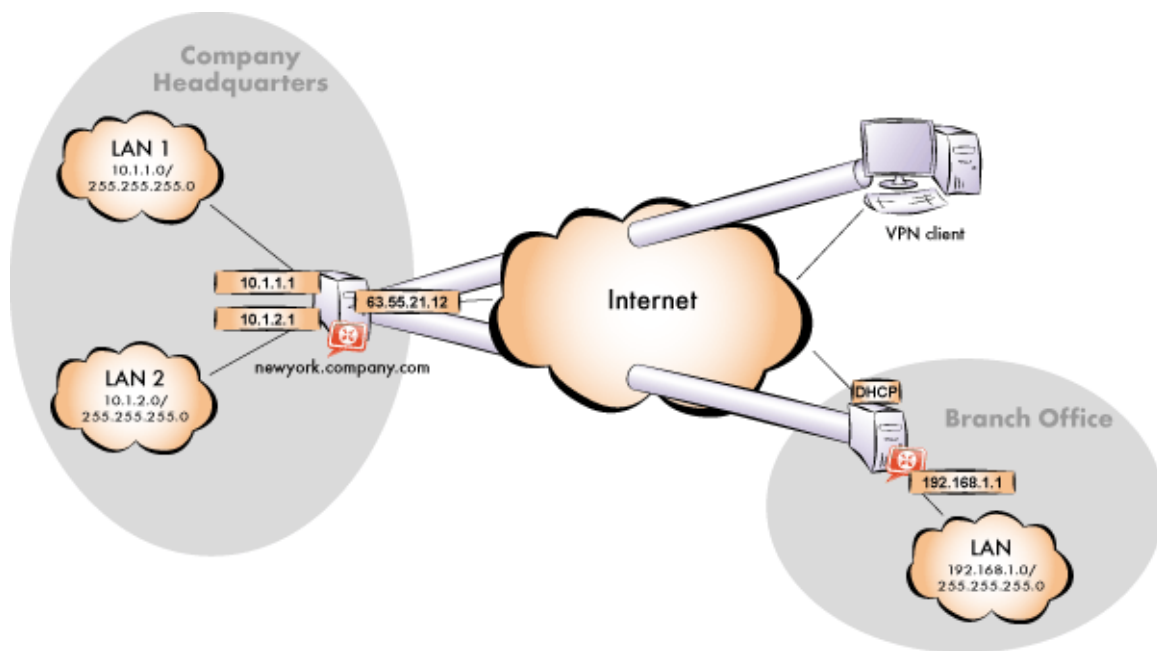


Figure 23.13 Example — interconnection of the headquarter and a filial office by VPN tunnel (connection of VPN clients is possible)

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.

Common method

The following actions must be taken in both local networks (i.e. in the main office and the filial):

1. It is necessary that *WinRoute* in version 6.0.0 or higher (older versions do not include *Kerio VPN*) is installed at the default gateway.

Note: For *each* installation of *WinRoute*, a separate license for corresponding number of users is required! For details see chapter [4](#).

2. Configure and test connection of the local network to the Internet. Hosts in the local network must use the *WinRoute* host's IP address as the default gateway and as the primary DNS server.

If it is a new (clean) *WinRoute* installation, it is possible to use the traffic rule wizard (refer to chapter [7.1](#)).

For detailed description of basic configuration of *WinRoute* and of the local network, refer to the *Kerio WinRoute Firewall — Step By Step* document.

3. In configuration of the *DNS* module set DNS forwarding rules for the domain in the remote network. This enables to access hosts in the remote network by using their DNS names (otherwise, it is necessary to specify remote hosts by IP addresses).

To provide correct forwarding of DNS requests from a *WinRoute* host, it is necessary to use an IP address of a network device belonging to the host as the primary DNS server. As a secondary DNS server, a server where DNS requests addressed to other domains will be forwarded must be specified (typically the ISP's DNS server).

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 cooperation of the *DNS* module with the DHCP server (in case that IP addresses are assigned dynamically to these hosts). For details, see chapter [8.1](#).

4. In the *Interfaces* section, allow the VPN server and set its SSL certificate if necessary. Note the fingerprint of the server's certificate for later use (it will be required for configuration of the remote endpoint of the VPN tunnel).

Check whether the automatically selected VPN subnet does not collide with any local subnet either in the headquarters or in the filial and select another free subnet if necessary.

5. Define the VPN tunnel to the remote network. The passive endpoint of the tunnel must be created at a server with fixed public IP address (i.e. at the headquarter's server). Only active endpoints of VPN tunnels can be created at servers with dynamic IP address.

If the remote endpoint of the tunnel has already been defined, check whether the tunnel was created. If not, refer to the *Error* log, check fingerprints of the certificates and also availability of the remote server.

6. In traffic rules, allow traffic between the local network, remote network and VPN clients and set desirable access restrictions. In this network configuration, all desirable restrictions can be set at the headquarter’s server. Therefore, only traffic between the local network and the VPN tunnel will be enabled at the filial’s server.
7. Test reachability of remote hosts from each local network. To perform the test, use the ping and tracert system commands. 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 the DNS.

The following sections provide detailed description of the *Kerio VPN* configuration both for the headquarter and the filial offices.

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 7.1) to configure the basic traffic policy in *WinRoute*. To keep the example as simple as possible, it is supposed that the access from the local network to the Internet is not restricted, i.e. that access to all services is allowed in step 4.

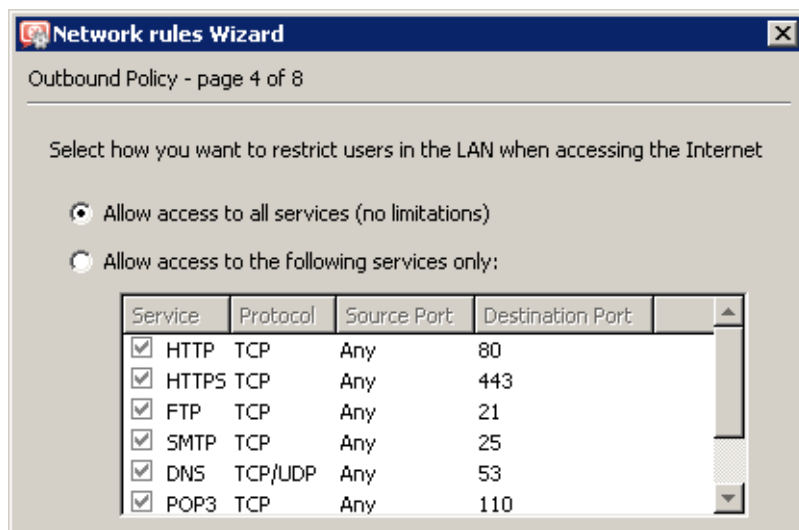


Figure 23.14 Headquarters — no restrictions are applied to accessing the Internet from the LAN

In step 5, select *Create rules for Kerio VPN server*. Status of the *Create rules for Kerio Clientless SSL-VPN* option is irrelevant (this example does not include *Clientless SSL-VPN* interface's issues).

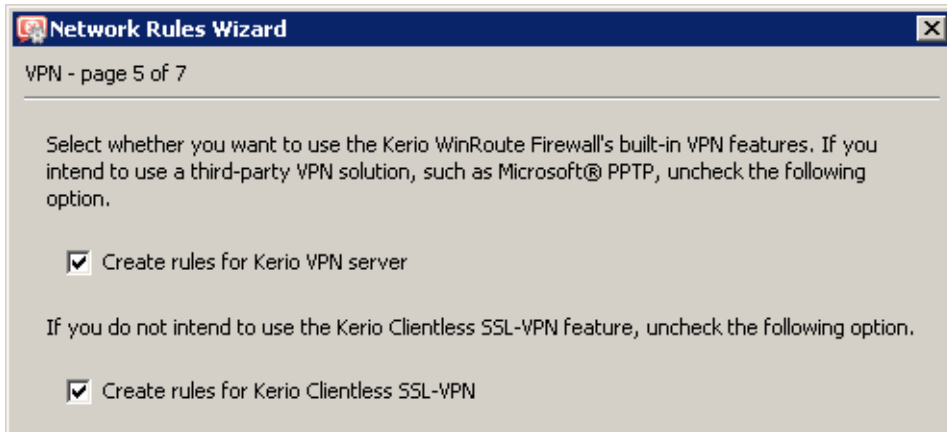


Figure 23.15 Headquarter — creating default traffic rules for Kerio VPN

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
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Trusted/Local	Firewall All VPN clients Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.16 Headquarter — default traffic rules for Kerio VPN

When the VPN tunnel is created, customize these rules according to the restriction requirements (see item 6).

Note: To keep the example as simple and transparent as possible, only traffic rules relevant for the *Kerio VPN* configuration are mentioned.

3. Customize DNS configuration as follows:
 - In the *WinRoute's DNS* module configuration, enable *DNS forwarder* (forwarding of DNS requests to other servers).
 - Enable the *Use custom forwarding* option and define rules for names in the *filial.company.com* domain. Specify the server for DNS forwarding by the IP address of the remote firewall host's interface (i.e. interface connected to the local network at the other end of the tunnel).

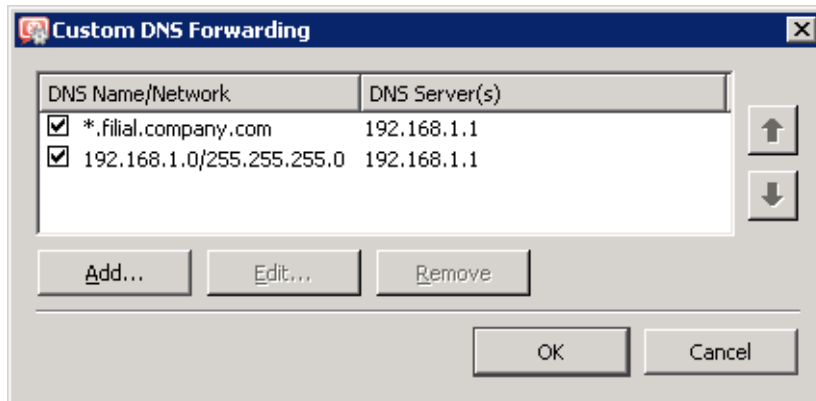


Figure 23.17 Headquarter — DNS forwarding settings

- 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 *LAN 1* local network. It is not necessary to set DNS server at the interface connected to *LAN 2* — DNS configuration is applied globally to the entire operating system.

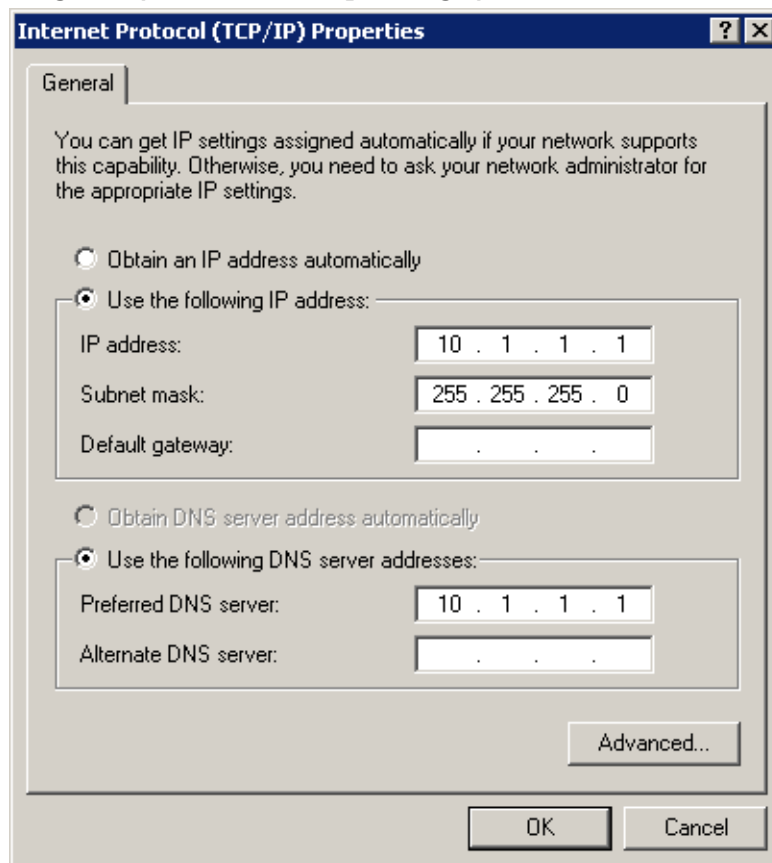


Figure 23.18 Headquarter — TCP/IP configuration at a firewall's interface connected to the local network

- 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 cooperation of the *DNS* module with the DHCP server (in case that IP addresses are assigned dynamically to these hosts). For details, see chapter [8.1](#).

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.

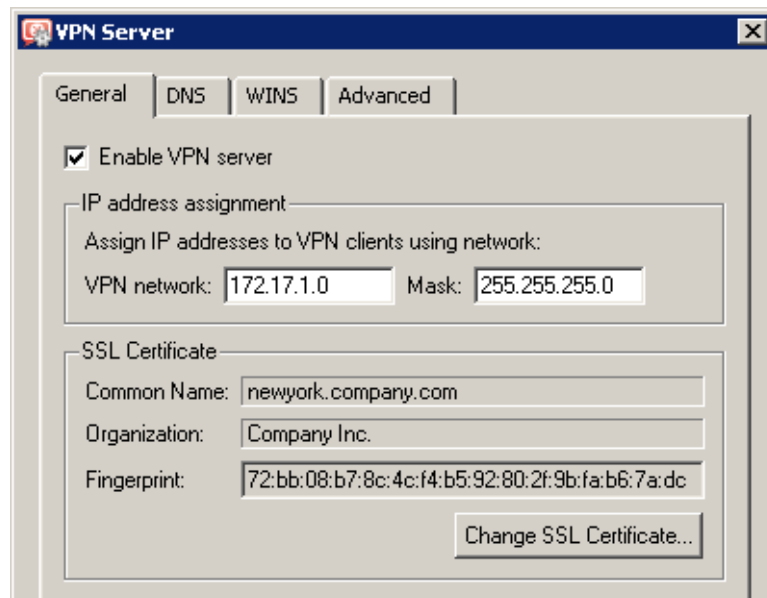


Figure 23.19 Headquarters — VPN server configuration

For a detailed description on the VPN server configuration, refer to chapter [23.1](#).

5. Create a passive end of the VPN tunnel (the server of the branch office uses a dynamic IP address). Specify the remote endpoint's fingerprint by the fingerprint of the certificate of the branch office VPN server.

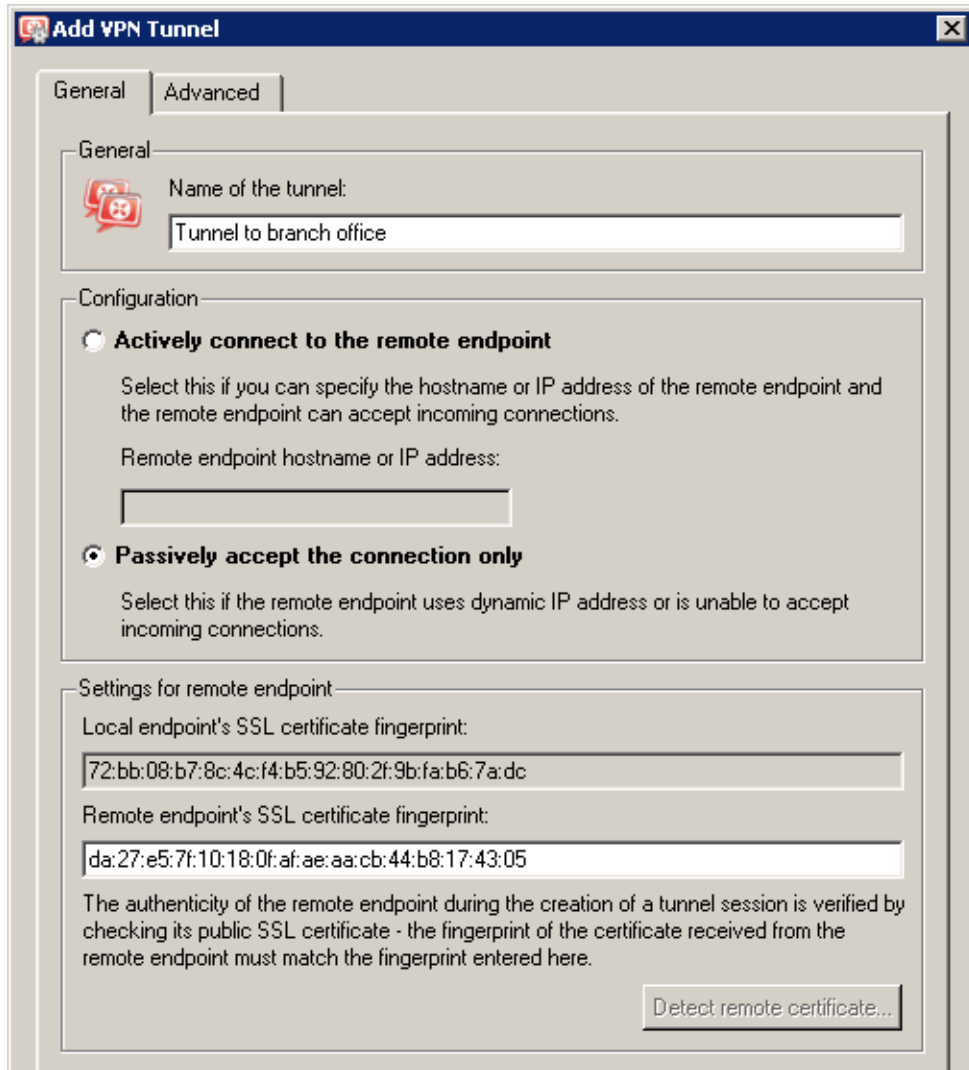


Figure 23.20 Headquarter — definition of VPN tunnel for a filial office

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).
 - 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..

23.5 Example of Kerio VPN configuration: company with a filial office

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall Trusted/Local	Firewall Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> VPN Clients	All VPN clients	LAN 1 Tunnel to branch office	Any	✓	
<input checked="" type="checkbox"/> Branch office	Tunnel to branch office	LAN 1	Any	✓	
<input checked="" type="checkbox"/> Company headquarters	Trusted/Local	Tunnel to branch office	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.21 Headquarter — final traffic rules

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 7.3).

Configuration of a filial office

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 7.1) to configure the basic traffic policy in *WinRoute*. To keep the example as simple as possible, it is supposed that the access from the local network to the Internet is not restricted, i.e. that access to all services is allowed in step 4.

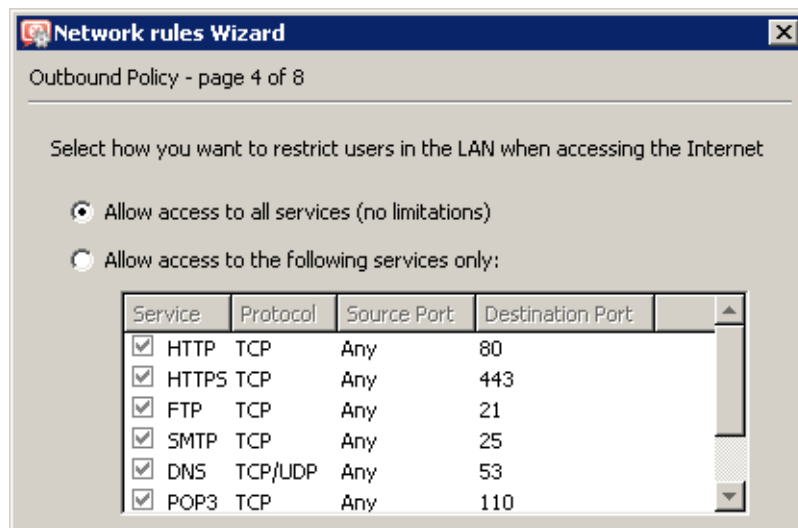


Figure 23.22 Filial — no restrictions are applied to accessing the Internet from the LAN

In this case, it would be meaningless to create rules for the *Kerio VPN server* and/or the *Kerio Clientless SSL-VPN*, since the server uses a dynamic public IP address). Therefore, leave these options disabled in step 5.

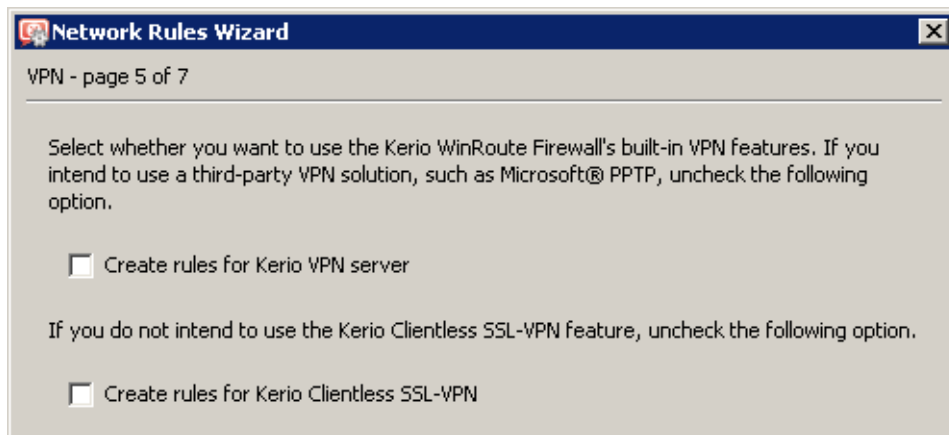


Figure 23.23 A filial — it is not necessary to create rules for the Kerio VPN server

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
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Trusted/Local	Firewall All VPN clients Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.24 Filial office — default traffic rules for 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 the *WinRoute's DNS* module configuration, enable *DNS forwarder* (forwarding of DNS requests to other servers).
- Enable the *Use custom forwarding* option and define rules for names in the *filial.company.com* domain. Specify the server for DNS forwarding by the IP address of the remote firewall host's interface (i.e. interface connected to the local network at the other end of the tunnel).
- 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.

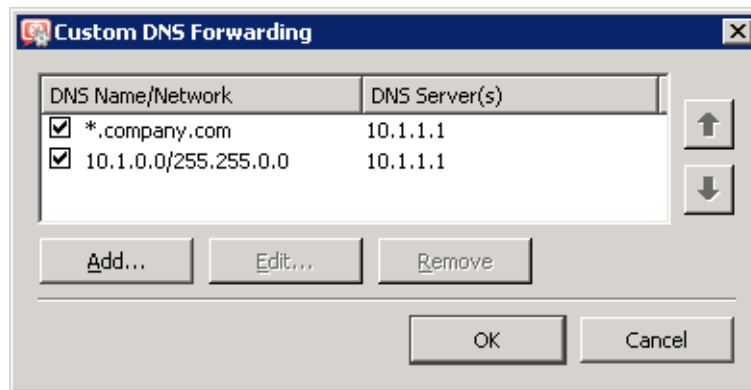


Figure 23.25 Filial office — DNS forwarding settings

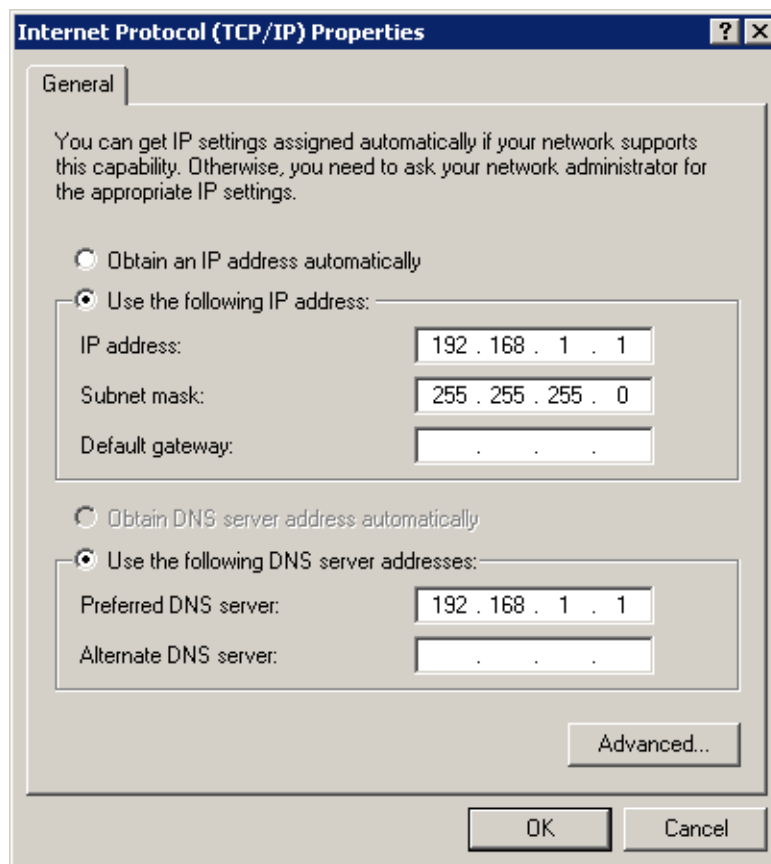


Figure 23.26 Filial office — TCP/IP configuration at a firewall's interface connected to the local network

- 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 cooperation of the `DNS` module with the `DHCP` server (in case that IP addresses are assigned dynamically to these hosts). For details, see chapter 8.1.

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.

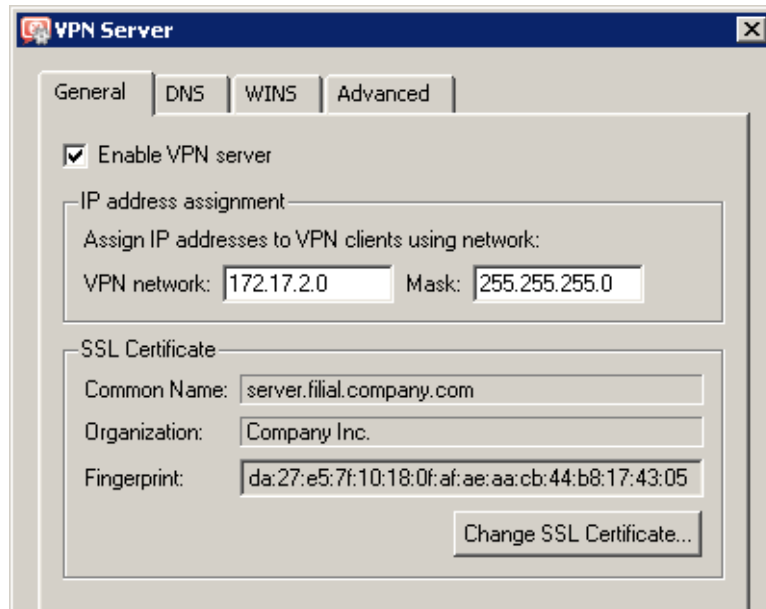


Figure 23.27 Filial office — VPN server configuration

For a detailed description on the VPN server configuration, refer to chapter [23.1](#).

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 [23.3](#).

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).

23.5 Example of Kerio VPN configuration: company with a filial office

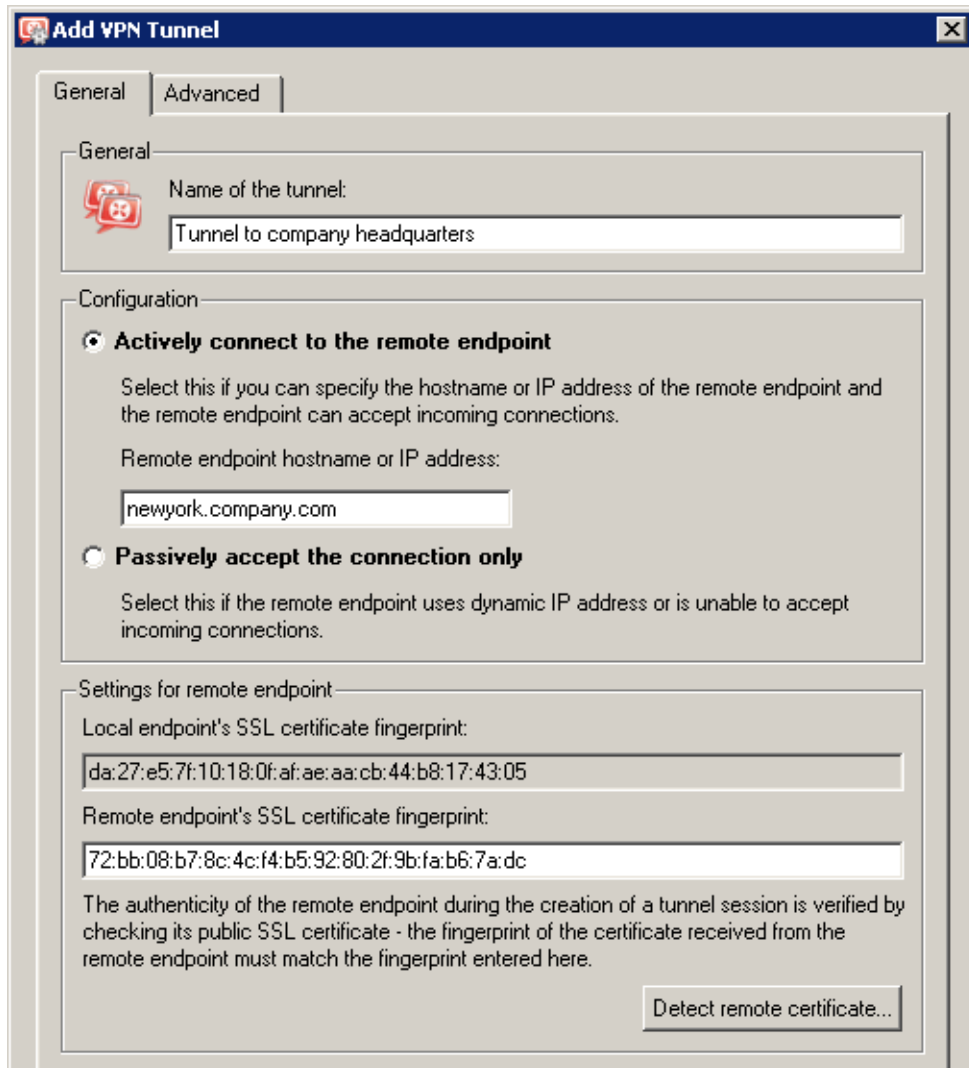


Figure 23.28 Filial office — definition of VPN tunnel for the headquarters

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall Tunnel to company headquarters Trusted/Local	Firewall Tunnel to company head Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.29 Filial office — final traffic rules

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.

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 the DNS.

23.6 Example of a more complex Kerio VPN configuration

In this chapter, an example of a more complex VPN configuration is provided where redundant routes arise between interconnected private networks (i.e. multiple routes exist between two networks that can be used for transfer of packets).

The only difference of *Kerio VPN* configuration between this type and VPN with no redundant routes (see chapter [23.5](#)) is setting of routing between endpoints of individual tunnels. In such a case, it is necessary to set routing between individual endpoints of VPN tunnels by hand. Automatic route exchange is inconvenient since *Kerio VPN* uses no routing protocol and the route exchange is based on comparison of routing tables at individual endpoints of the VPN tunnel (see also chapter [23.4](#)). If the automatic exchange is applied, the routing will not be ideal!

For better reference, the configuration is here described by an example of a company with a headquarters and two filial offices with their local private network interconnected by VPN tunnels (so called triangle pattern). This example can be then adapted and applied to any number of interconnected private networks.

The example focuses configuration of VPN tunnels and correct setting of routing between individual private networks (it does not include access restrictions). Access restrictions options within VPN are described by the example in chapter [23.5](#).

Specification

The network follows the pattern shown in figure [23.30](#).

The server (default gateway) uses the fixed IP address 63.55.21.12 (DNS name is `gw-newyork.company.com`). The server of one filial uses the IP address 115.95.27.55 (DNS name `gw-london.company.com`), the other filial's server uses a dynamic IP address assigned by the ISP.

The headquarters uses the DNS domain `company.com`, filials use subdomains `santaclara.company.com` and `newyork.company.com`. Configuration of individual local networks and the IP addresses used are shown in the figure.

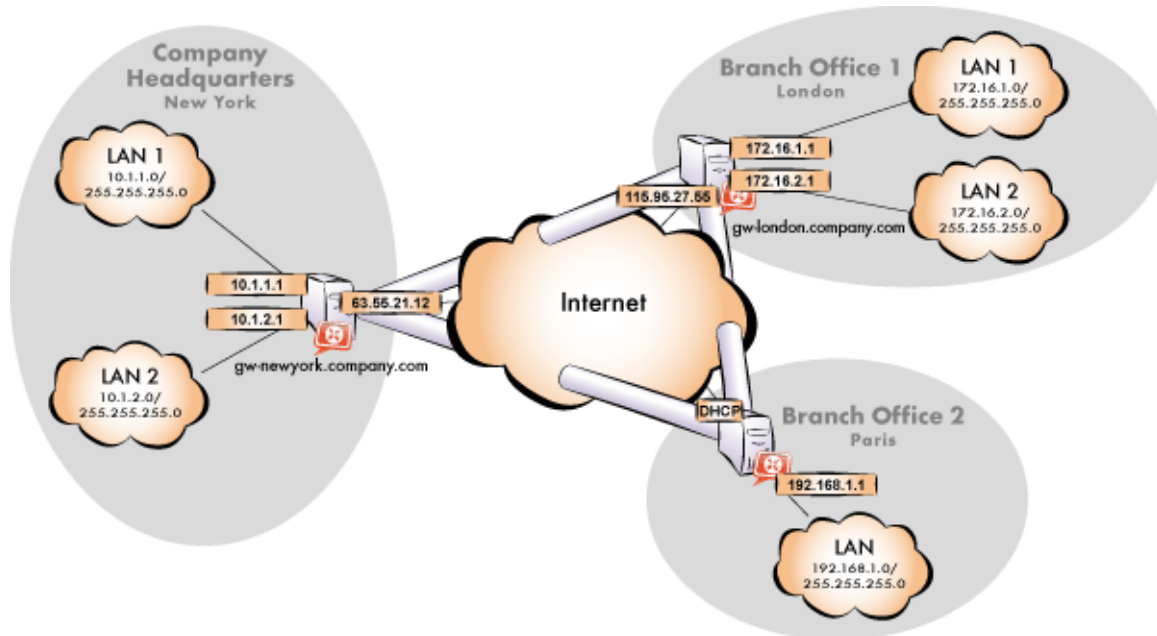


Figure 23.30 Example of a VPN configuration — a company with two filials

Common method

The following actions must be taken in all local networks (i.e. in the main office and both filials):

1. *WinRoute* in version 6.1.0 or higher must be installed at the default gateway. Older versions do not allow setting of routing for VPN tunnels. Therefore, they cannot be used for this VPN configuration (see figure [23.30](#)).

Note: For *each* installation of *WinRoute*, a separate license for corresponding number of users is required! For details see chapter [4](#).

2. Configure and test connection of the local network to the Internet. Hosts in the local network must use the *WinRoute* host's IP address as the default gateway and as the primary DNS server.

If it is a new (clean) *WinRoute* installation, it is possible to use the traffic rule wizard (refer to chapter [7.1](#)).

For detailed description of basic configuration of *WinRoute* and of the local network, refer to the *Kerio WinRoute Firewall — Step By Step* document.

3. In configuration of the *DNS* module, set DNS forwarding rules for domains of the other filials. This enables to access hosts in the remote networks by using their DNS names (otherwise, it is necessary to specify remote hosts by IP addresses).

To provide correct forwarding of DNS requests from a *WinRoute* host, it is necessary to use an IP address of a network device belonging to the host as the primary DNS server. As a secondary DNS server, a server where DNS requests addressed to other domains will be forwarded must be specified (typically the ISP's DNS server).

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 cooperation of the *DNS* module with the DHCP server (in case that IP addresses are assigned dynamically to these hosts). For details, see chapter [8.1](#).

4. In the *Interfaces* section, allow the VPN server and set its SSL certificate if necessary. Note the fingerprint of the server's certificate for later use (it will be required for configuration of the VPN tunnels in the other filials).

Check whether the automatically selected VPN subnet does not collide with any local subnet in any filial and select another free subnet if necessary.

Note: With respect to the complexity of this VPN configuration, it is recommended to reserve three free subnets in advance that can later be assigned to individual VPN servers.

5. Define the VPN tunnel to one of the remote networks. The passive endpoint of the tunnel must be created at a server with fixed public IP address. Only active endpoints of VPN tunnels can be created at servers with dynamic IP address.

Set routing (define custom routes) for the tunnel. Select the *Use custom routes only* option and specify all subnets of the remote network in the custom routes list.

If the remote endpoint of the tunnel has already been defined, check whether the tunnel was created. If not, refer to the *Error* log, check fingerprints of the certificates and also availability of the remote server.

6. Follow the same method to define a tunnel and set routing to the other remote network.
7. Allow traffic between the local and the remote networks. To allow any traffic, just add the created VPN tunnels to the *Source* and *Destination* items in the *Local traffic* rule. Access restrictions options within VPN are described by the example in chapter [23.5](#).
8. Test reachability of remote hosts in both remote networks. To perform the test, use the `ping` and `tracert` system commands. 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 the DNS.

The following sections provide detailed description of the *Kerio VPN* configuration both for the headquarter and the filial offices.

Headquarters configuration

1. Install *WinRoute* (version 6.1.0 or higher) at the default gateway of the headquarters network.
2. Use *Network Rules Wizard* (see chapter 7.1) to configure the basic traffic policy in *WinRoute*. To keep the example as simple as possible, it is supposed that the access from the local network to the Internet is not restricted, i.e. that access to all services is allowed in step 4.

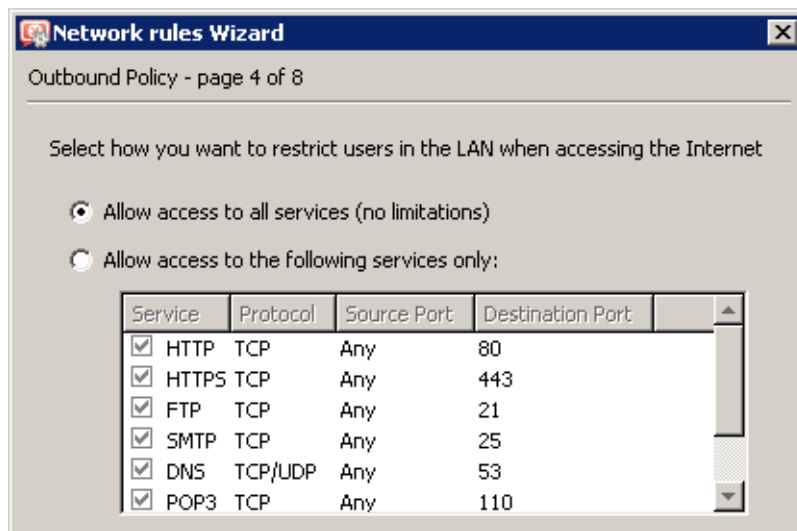


Figure 23.31 Headquarters — no restrictions are applied to accessing the Internet from the LAN

In step 5, select *Create rules for Kerio VPN server*. Status of the *Create rules for Kerio Clientless SSL-VPN* option is irrelevant (this example does not include *Clientless SSL-VPN* interface's issues).

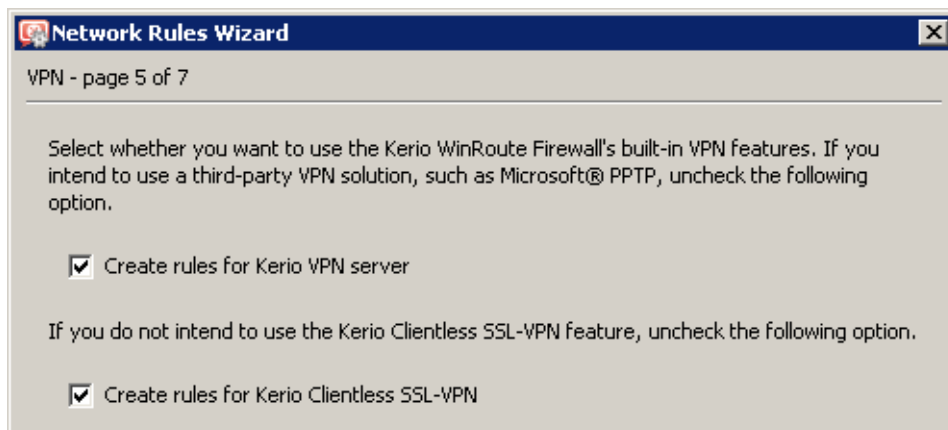


Figure 23.32 Headquarter — creating default traffic rules for Kerio VPN

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
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Trusted/Local	Firewall All VPN clients Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.33 Headquarter — default traffic rules for Kerio VPN

3. Customize DNS configuration as follows:

- In the *WinRoute's* DNS module configuration, enable *DNS forwarder* (forwarding of DNS requests to other servers).
- Enable the *Use custom forwarding* option and define rules for names in the *filial1.company.com* and *filial2.company.com* domains. To specify the forwarding DNS server, always use the IP address of the *WinRoute* host's inbound interface connected to the local network at the remote side of the tunnel.

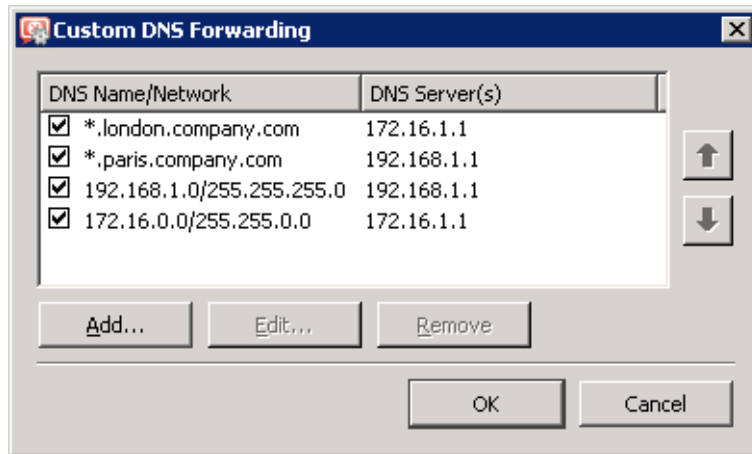


Figure 23.34 Headquarter — DNS forwarding settings

- 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 *LAN 1* local network. It is not necessary to set DNS at the interface connected to *LAN 2*.
- Set the IP address 10.1.1.1 as a primary DNS server also for the other hosts.

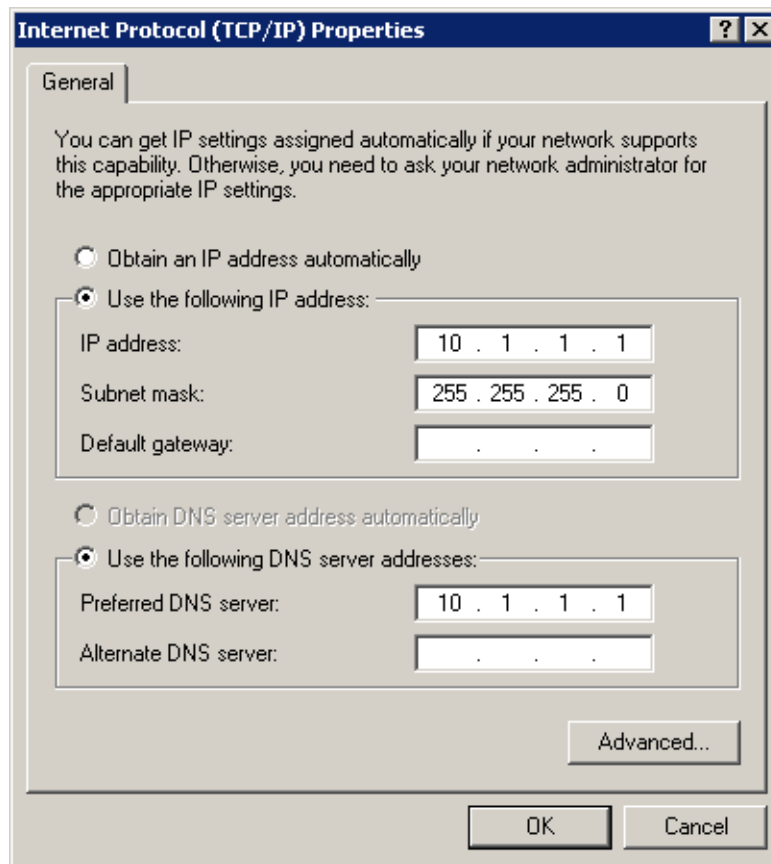


Figure 23.35 Headquarter — TCP/IP configuration at a firewall's interface connected to the local network

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. Check whether this subnet does not collide with any other subnet in the headquarters or in the filials. If it does, specify a free subnet.

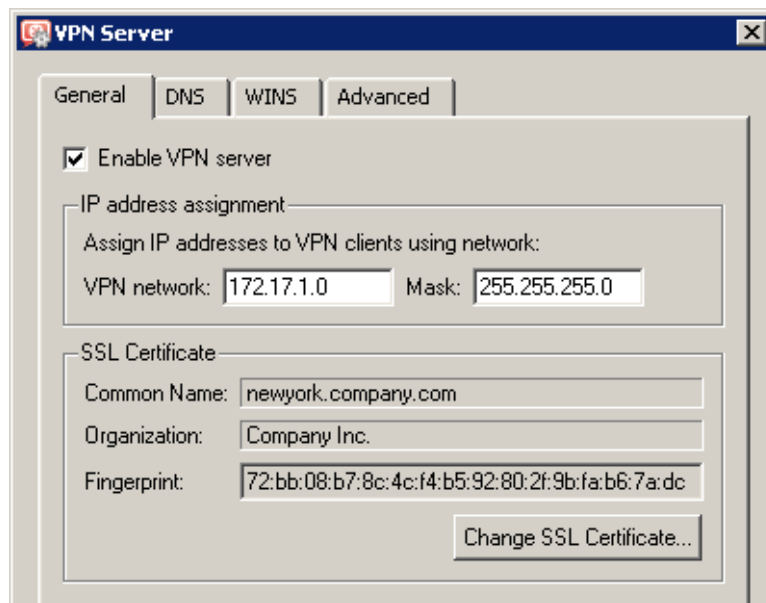


Figure 23.36 Headquarters — VPN server configuration

For a detailed description on the VPN server configuration, refer to chapter [23.1](#).

5. Create a passive endpoint of the VPN tunnel connected to the *London* filial. Use the fingerprint of the VPN server of the *London* filial office as a specification of the fingerprint of the remote SSL certificate.

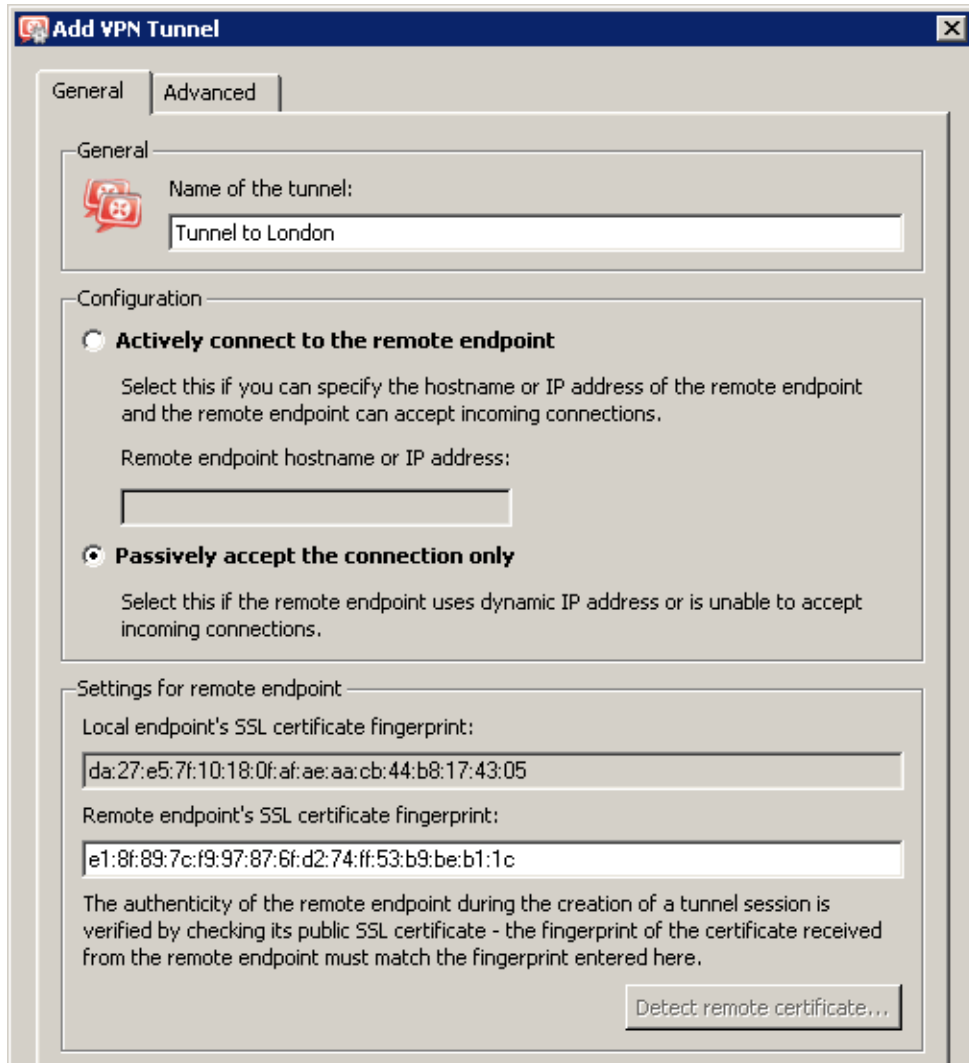


Figure 23.37 Headquarter — definition of VPN tunnel for the London filial

On the *Advanced* tab, select the *Use custom routes only* option and set routes to the subnets at the remote endpoint of the tunnel (i.e. in the *London* filial).

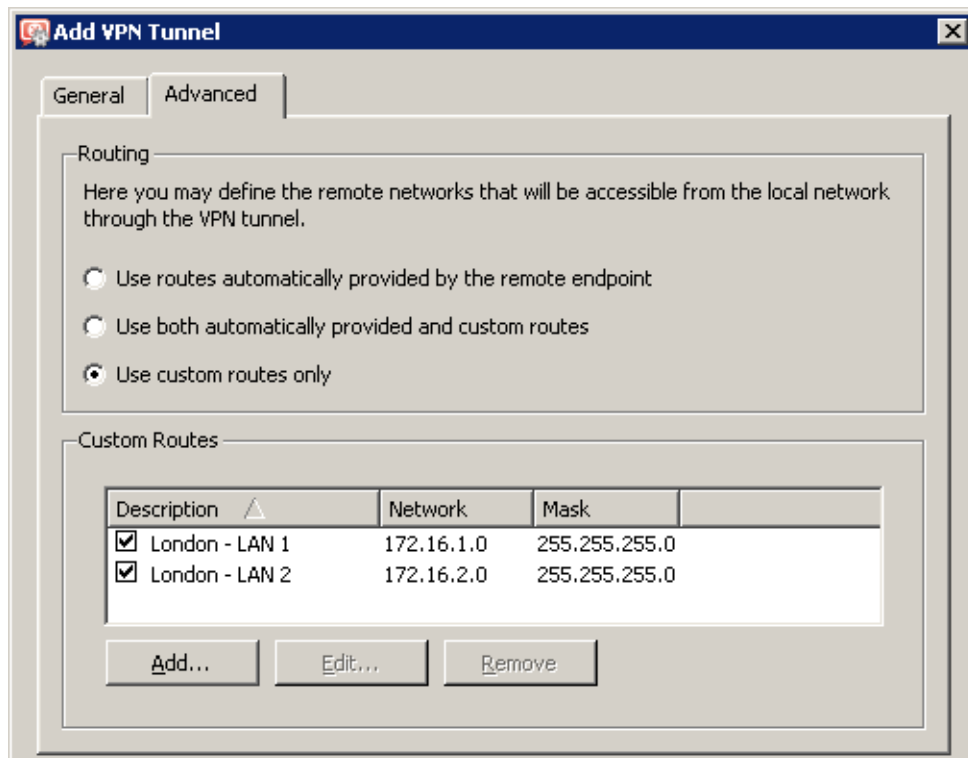


Figure 23.38 The headquarters — routing configuration for the tunnel connected to the London filial

Warning

In case that the VPN configuration described here is applied (see figure 23.30), it is *unrecommended* to use automatically provided routes! In case of an automatic exchange of routes, the routing within the VPN is not be ideal (for example, any traffic between the *headquarters* and the *Paris* filial office is routed via the *London* filial whereas the tunnel between the *headquarters* and the *Paris* office stays waste.

- Use the same method to create a passive endpoint for the tunnel connected to the *Paris* filial.

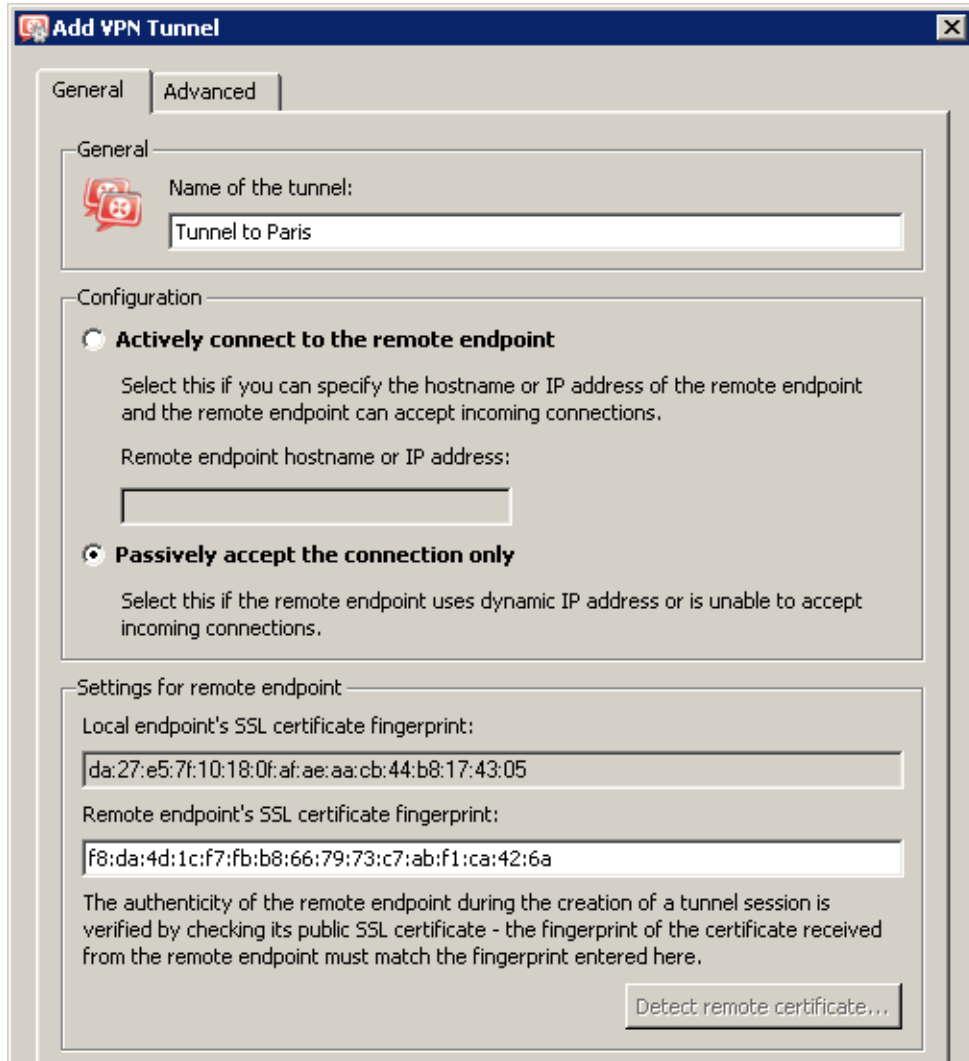


Figure 23.39 The headquarters — definition of VPN tunnel for the Paris filial

On the *Advanced* tab, select the *Use custom routes only* option and set routes to the subnets at the remote endpoint of the tunnel (i.e. in the *Paris* filial).

- Add the new VPN tunnels into the *Local Traffic* rule.

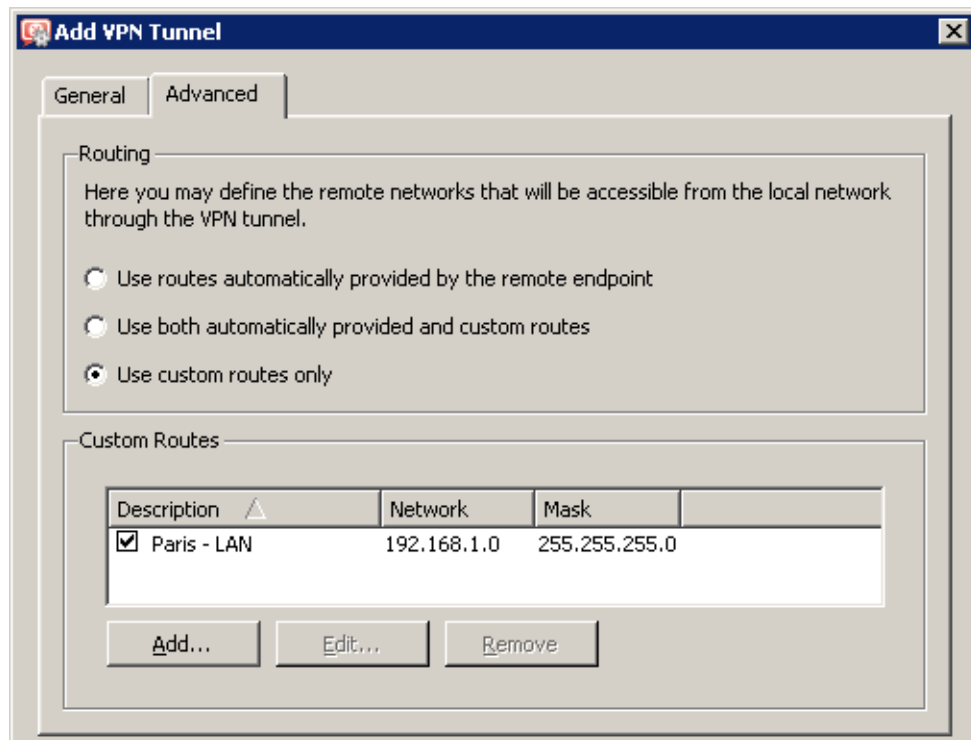


Figure 23.40 The headquarters – routing configuration for the tunnel connected to the Paris filial

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Tunnel to London Tunnel to Paris Trusted/Local	Firewall All VPN clients Tunnel to London Tunnel to Paris Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.41 Headquarter – final traffic rules

Configuration of the London filial

1. Install *WinRoute* (version 6.1.0 or higher) at the default gateway of the filial's network.
2. Use *Network Rules Wizard* (see chapter 7.1) to configure the basic traffic policy in *WinRoute*. To keep the example as simple as possible, it is supposed that the access from the local network to the Internet is not restricted, i.e. that access to all services is allowed in step 4. In step 5 of the wizard, select the *Create rules for Kerio VPN server* option (setting of the *Create rules for Kerio Clientless SSL-VPN* option is not regarded here).

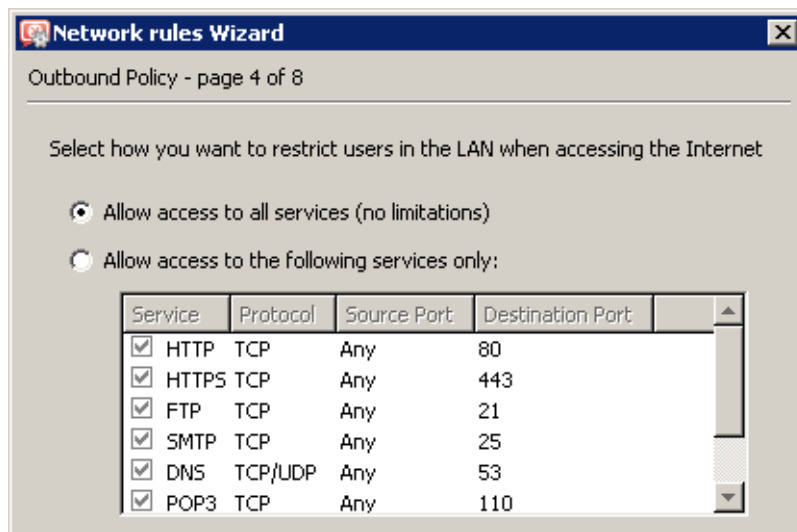


Figure 23.42 The London filial — no restrictions are applied to accessing the Internet from the LAN

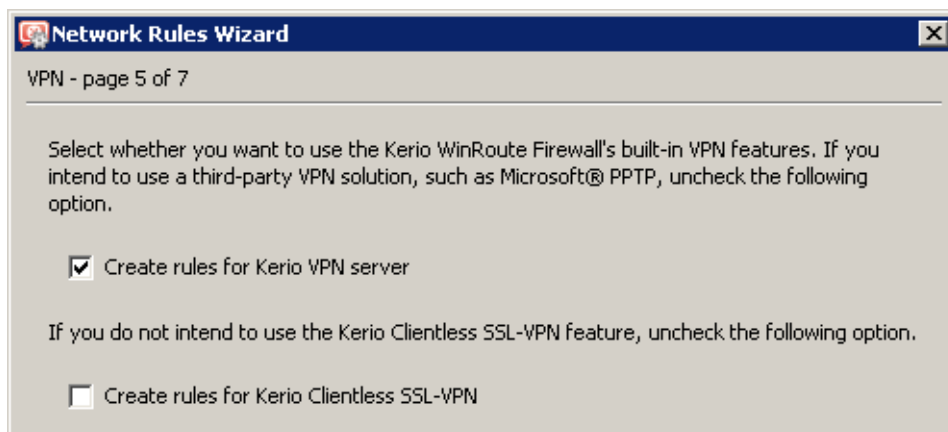


Figure 23.43 The London filial office — creating default traffic rules for Kerio VPN

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
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall All VPN clients Trusted/Local	Firewall All VPN clients Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.44 The London filial office — default traffic rules for Kerio VPN

3. Customize DNS configuration as follows:

- In the *WinRoute's* DNS module configuration, enable *DNS forwarder* (forwarding of DNS requests to other servers).
- Enable the *Use custom forwarding* option and define rules for names in the *company.com* and *filial2.company.com* domains. To specify the forwarding DNS server, always use the IP address of the *WinRoute* host's inbound interface connected to the local network at the remote side of the tunnel.

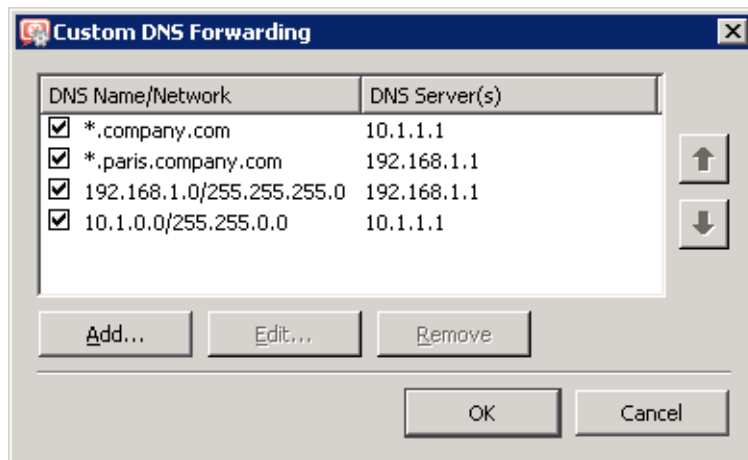


Figure 23.45 The London filial office — DNS forwarding settings

- Set the IP address of this interface (172.16.1.1) as a primary DNS server for the *WinRoute* host's interface connected to the *LAN 1* local network. It is not necessary to set DNS at the interface connected to *LAN 2*.
 - Set the IP address 172.16.1.1 as a primary DNS server also for the other hosts.
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. Check whether this subnet does not collide with any other subnet in the headquarters or in the filials. If it does, specify a free subnet.

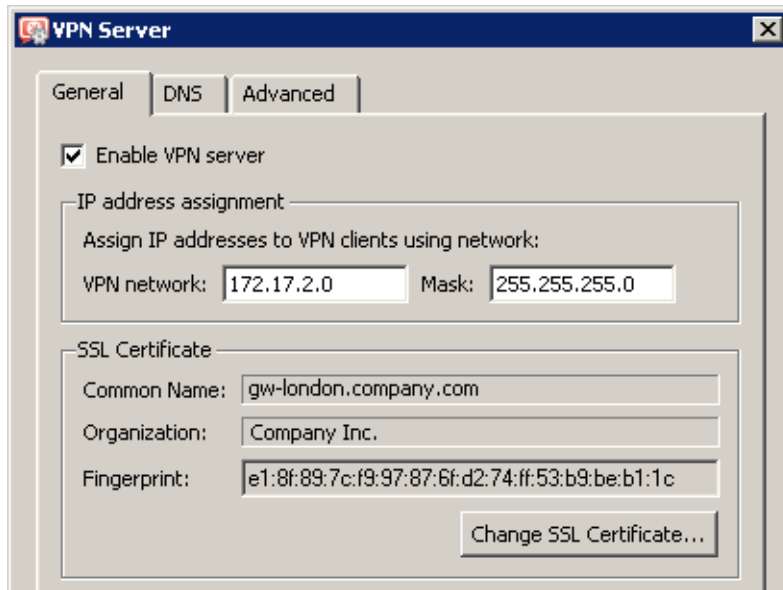


Figure 23.46 The London filial office — VPN server configuration

For a detailed description on the VPN server configuration, refer to chapter [23.1](#).

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.

On the *Advanced* tab, select the *Use custom routes only* option and set routes to *headquarters'* local networks.

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 gw-newyork.company.com` command can be used at the London

branch office server.

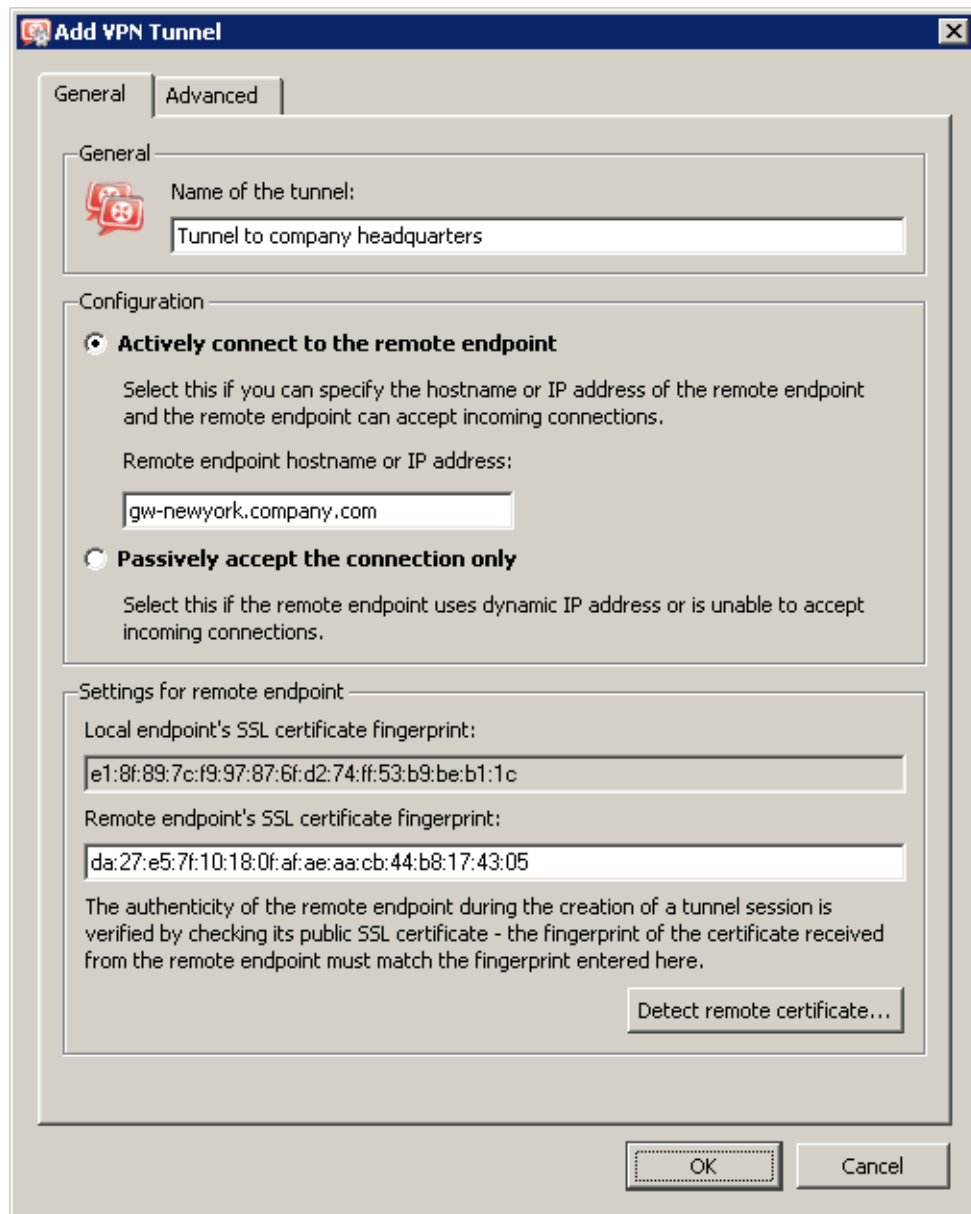


Figure 23.47 The London filial office — definition of VPN tunnel for the headquarters

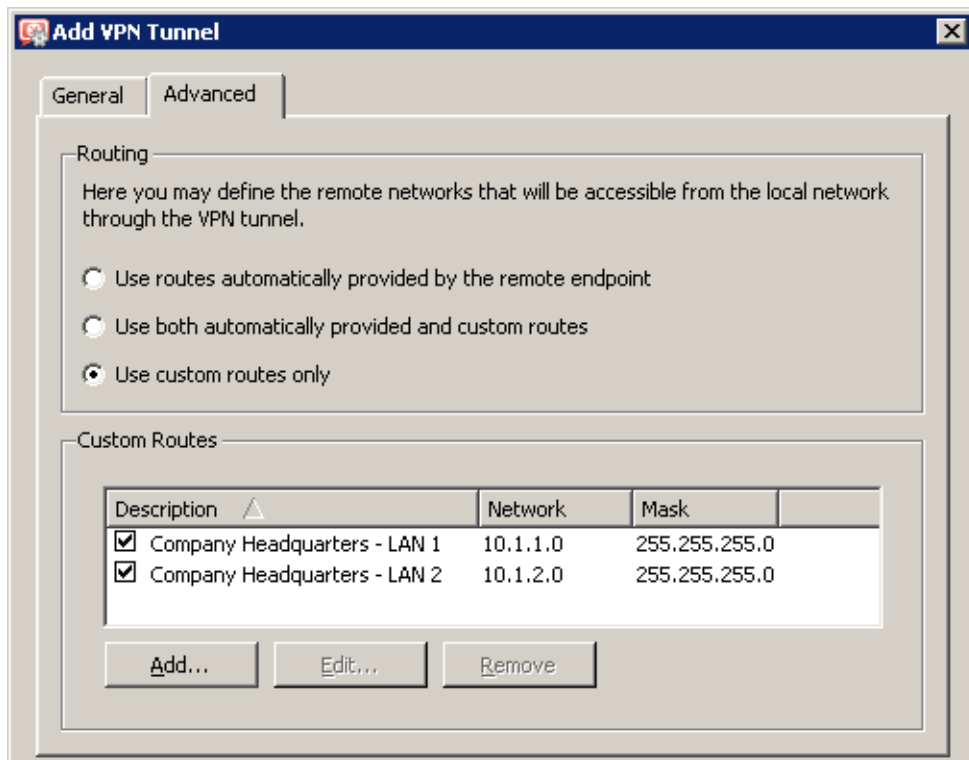


Figure 23.48 The London filial — routing configuration for the tunnel connected to the headquarters

6. Create a passive endpoint of the VPN tunnel connected to the *Paris* filial. Use the fingerprint of the VPN server of the *Paris* filial office as a specification of the fingerprint of the remote SSL certificate.

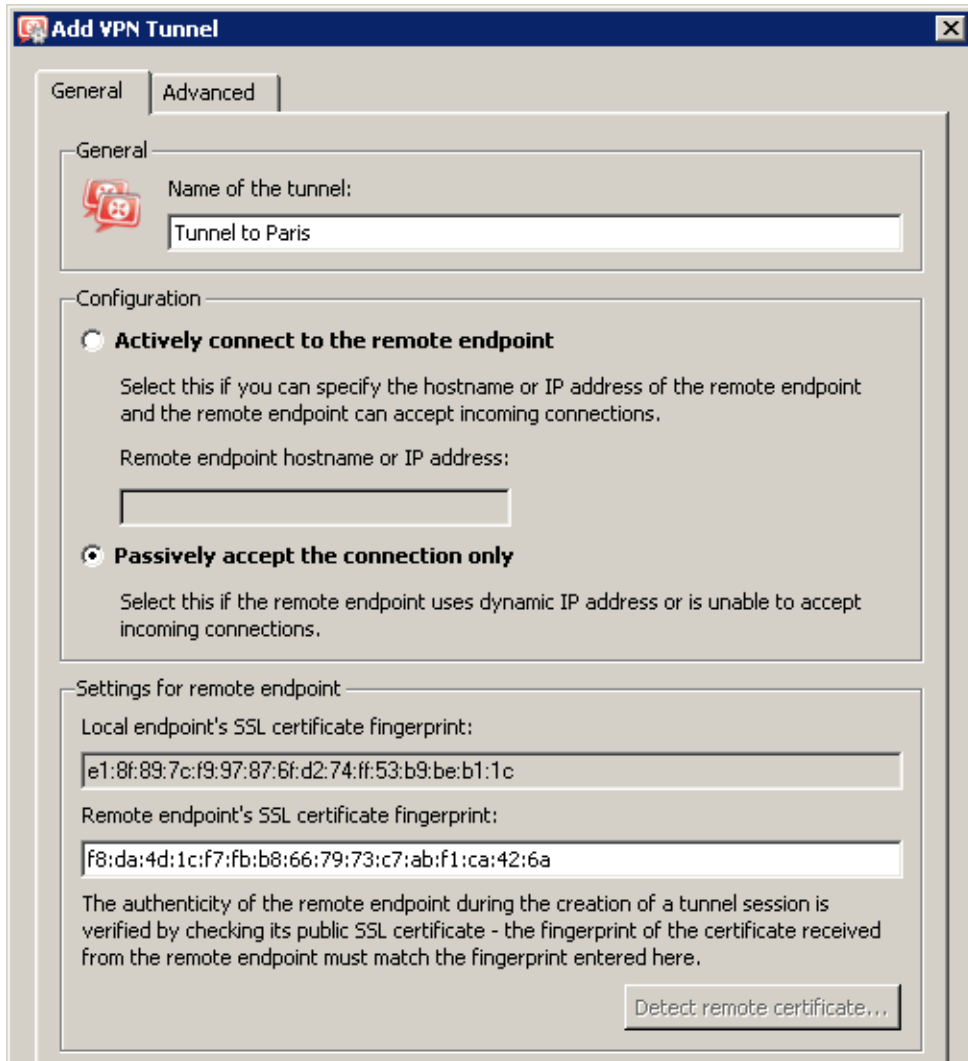


Figure 23.49 The London filial office — definition of VPN tunnel for the Paris filial office

On the *Advanced* tab, select the *Use custom routes only* option and set routes to *Paris'* local networks.

7. Add the new VPN tunnels into the *Local Traffic* rule. It is also possible to remove the *Dial-In* interface and the *VPN clients* group from this rule (supposing that all VPN clients connect to the headquarters' server).

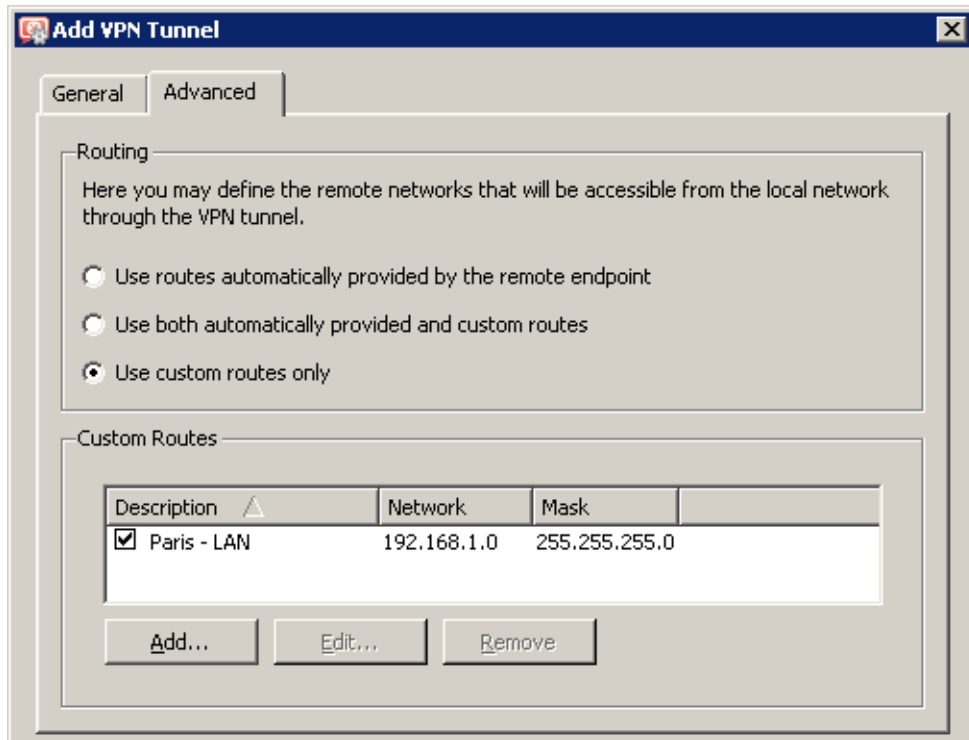


Figure 23.50 The London filial — routing configuration for the tunnel connected to the Paris branch office

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall Tunnel to company headquarters Tunnel to Paris Trusted/Local	Firewall Tunnel to company he Tunnel to Paris Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.51 The London filial office — final traffic rules

Configuration of the Paris filial

1. Install *WinRoute* (version 6.1.0 or higher) at the default gateway of the filial's network.
2. Use *Network Rules Wizard* (see chapter 7.1) to configure the basic traffic policy in *WinRoute*. To keep the example as simple as possible, it is supposed that the access from the local network to the Internet is not restricted, i.e. that access to all services is allowed in step 4.

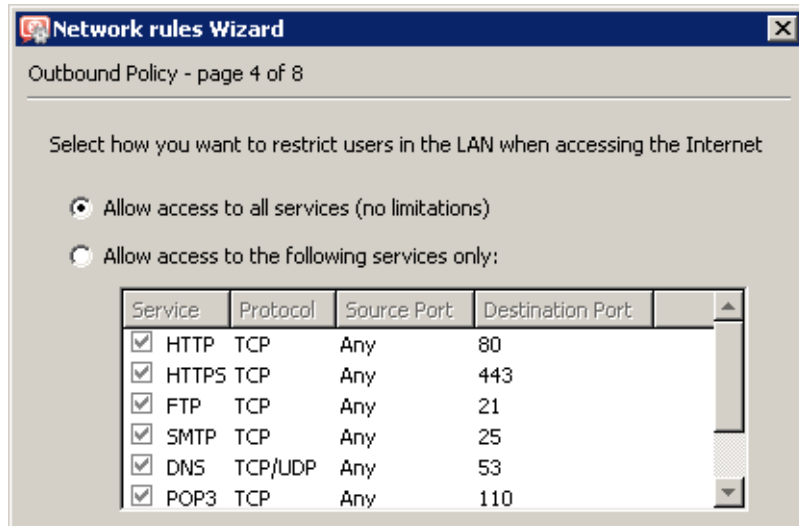


Figure 23.52 The Paris filial — no restrictions are applied to accessing the Internet from the LAN

In this case, it would be meaningless to create rules for the *Kerio VPN server* and/or the *Kerio Clientless SSL-VPN*, since the server uses a dynamic public IP address). Therefore, leave these options disabled in step 5.

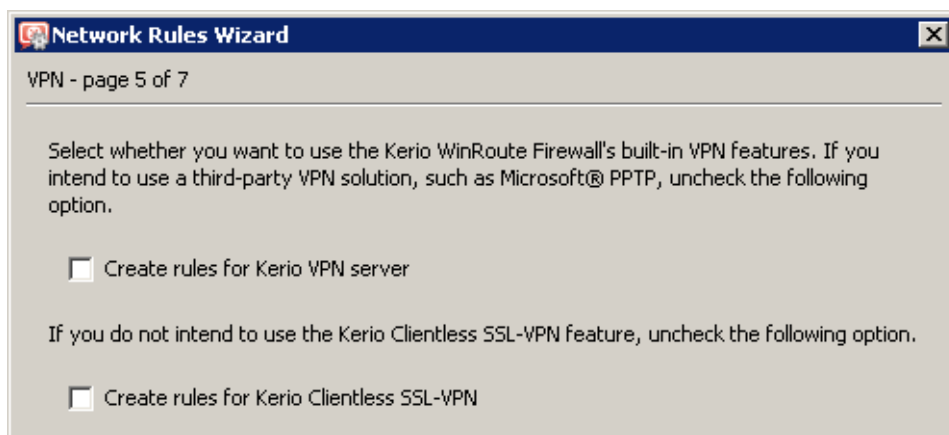


Figure 23.53 The Paris filial — default rules for Kerio VPN will not be created

3. Customize DNS configuration as follows:

- In the *WinRoute's* DNS module configuration, enable *DNS forwarder* (forwarding of DNS requests to other servers).
- Enable the *Use custom forwarding* option and define rules for names in the *company.com* and *filial1.company.com* domains. Specify the server for DNS forwarding by the IP address of the remote firewall host's interface (i.e. interface connected to the local network at the other end of the tunnel).

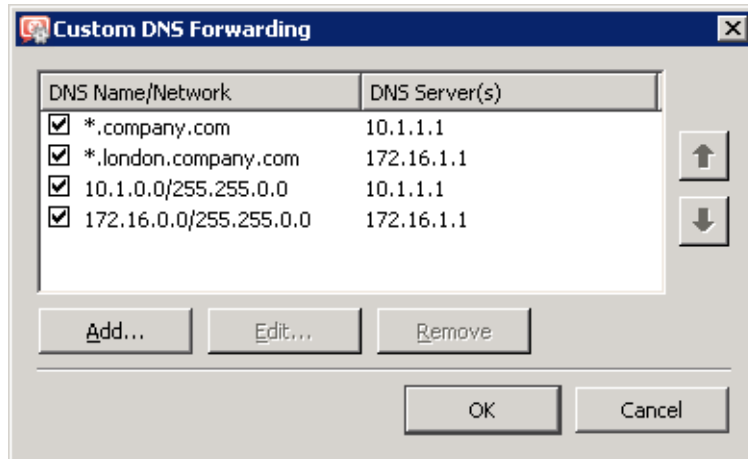


Figure 23.54 The Paris filial office — DNS forwarding settings

- Set the IP address of this interface (172.16.1.1) as a primary DNS server for the *WinRoute* host's interface connected to the *LAN 1* local network. It is not necessary to set DNS at the interface connected to *LAN 2*.
 - Set the IP address 172.16.1.1 as a primary DNS server also for the other hosts.
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. Check whether this subnet does not collide with any other subnet in the headquarters or in the filials. If it does, specify a free subnet.

For a detailed description on the VPN server configuration, refer to chapter [23.1](#).

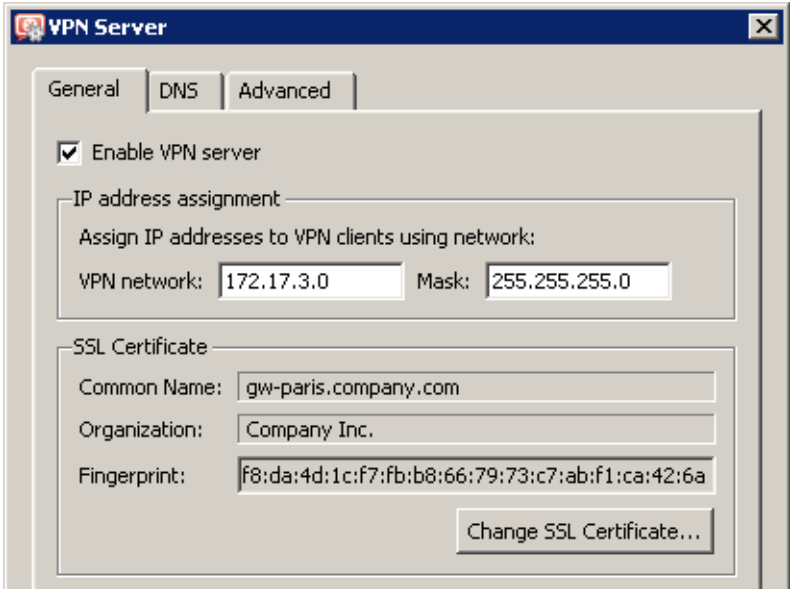


Figure 23.55 The Paris filial office — VPN server configuration

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.

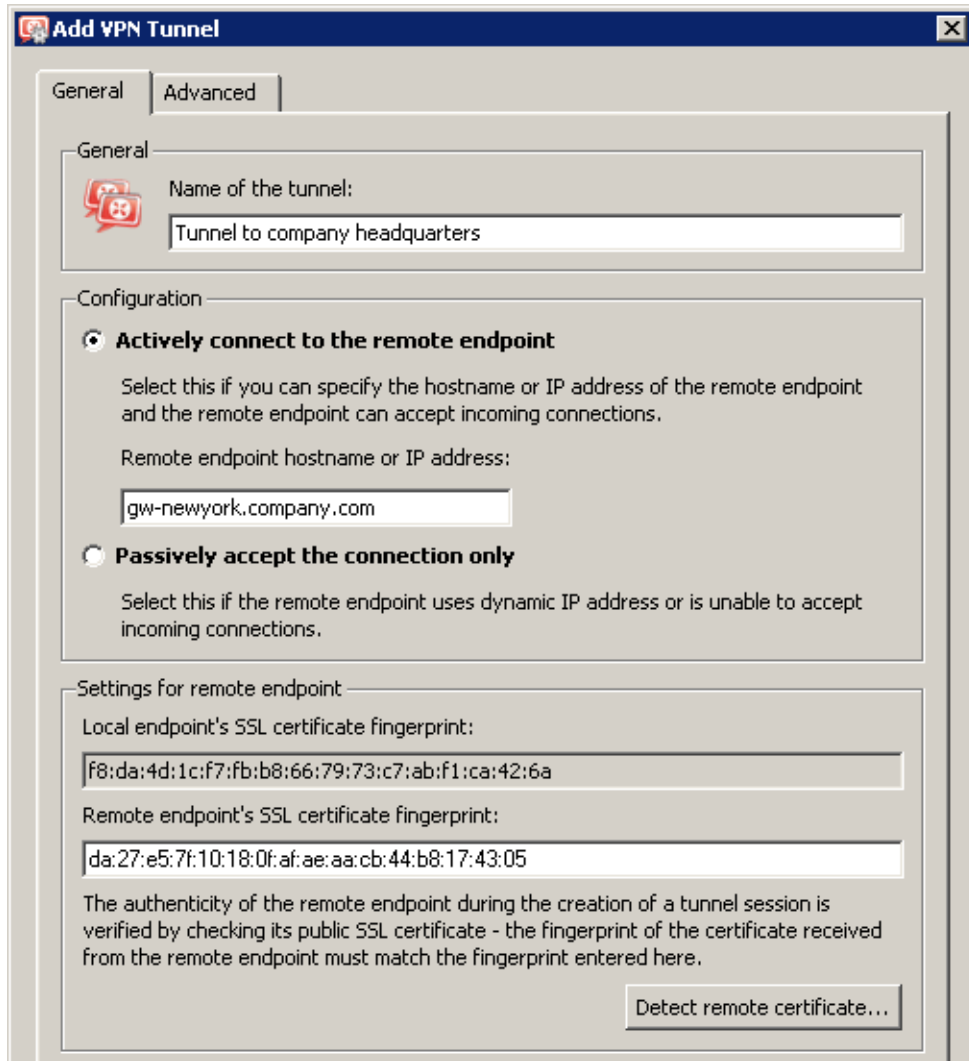


Figure 23.56 The Paris filial office — definition of VPN tunnel for the headquarters

On the *Advanced* tab, select the *Use custom routes only* option and set routes to *headquarters'* local networks.

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 gw-sanfrancisco.company.com` command can be used at the

Paris branch office server.

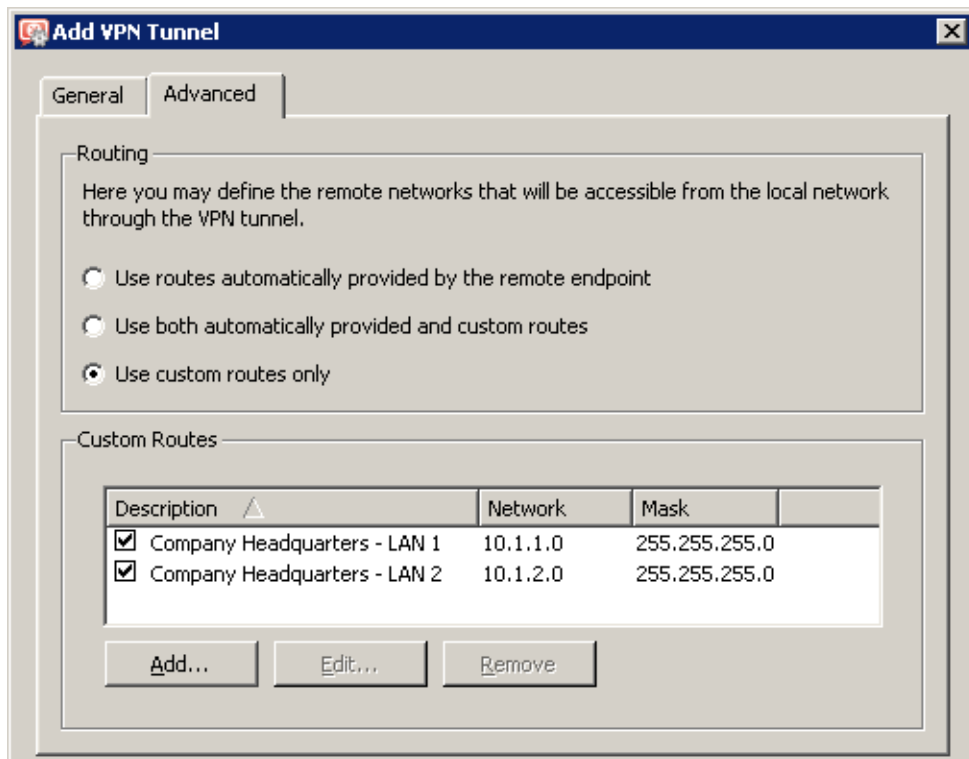


Figure 23.57 The Paris filial — routing configuration for the tunnel connected to the headquarters

6. Create an active endpoint of the tunnel connected to *London* (server gw-london.company.com). Use the fingerprint of the VPN server of the *London* filial office as a specification of the fingerprint of the remote SSL certificate.

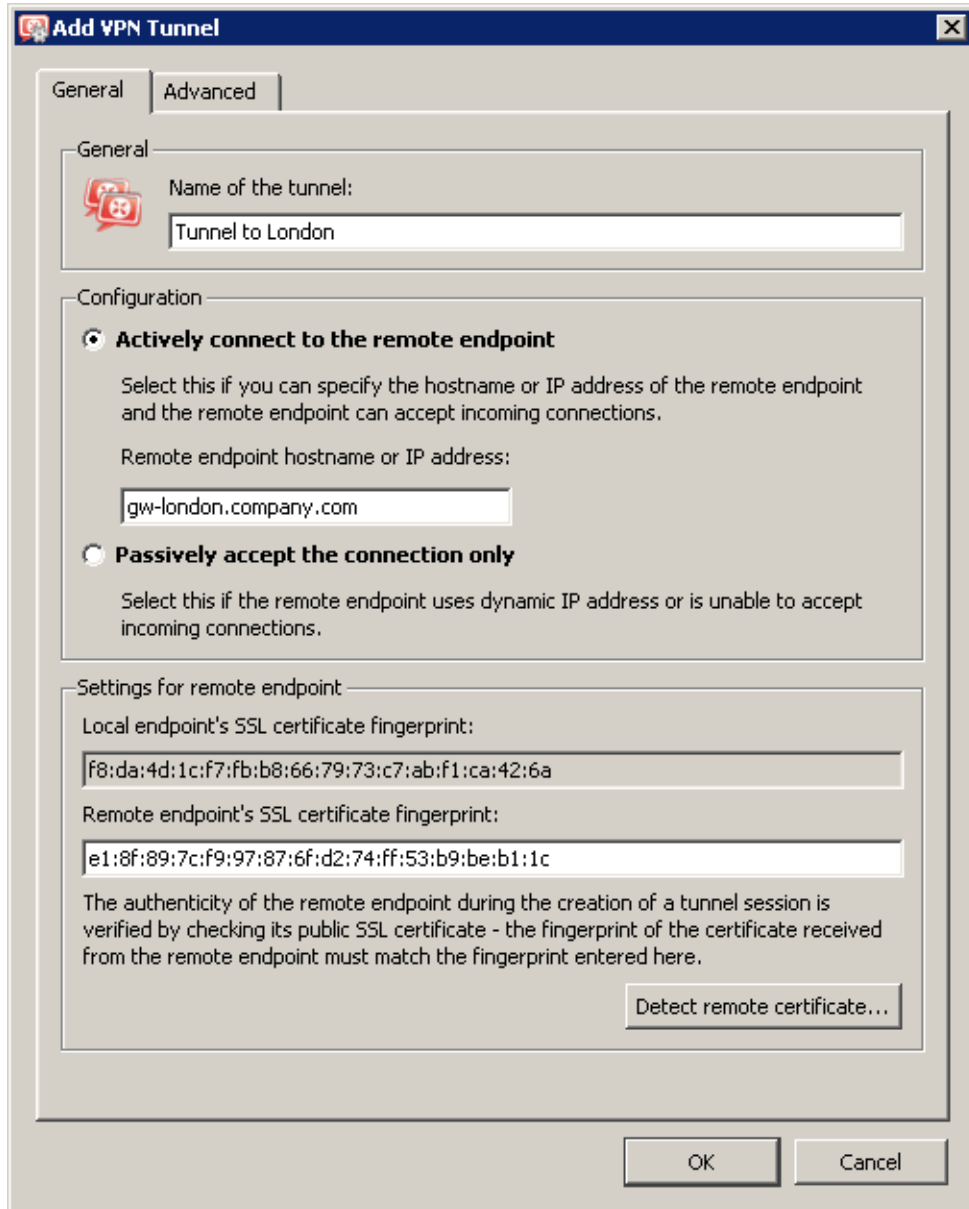


Figure 23.58 The Paris filial office — definition of VPN tunnel for the London filial office

On the *Advanced* tab, select the *Use custom routes only* option and set routes to *London's* local networks.

Like in the previous step, check whether the tunnel has been established successfully, and check reachability of remote private networks (i.e. of local networks in the *London* filial).

7. Add the new VPN tunnels 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

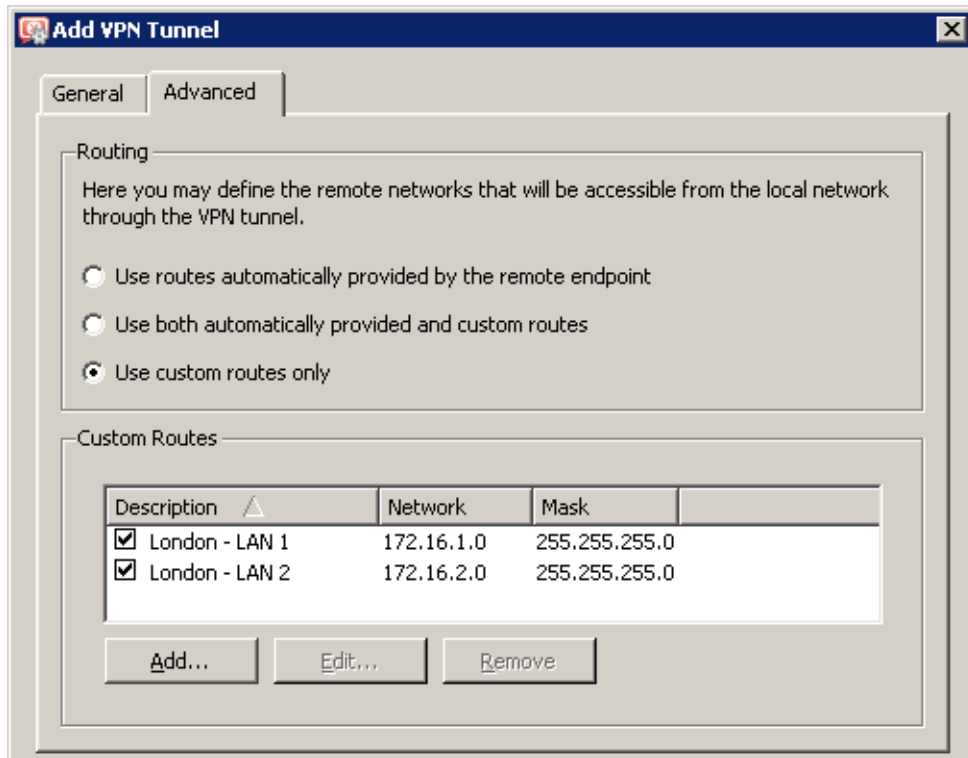


Figure 23.59 The Paris filial — routing configuration for the tunnel connected to the London branch office

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Service Kerio VPN	Any	Firewall	Kerio VPN	✓	
<input checked="" type="checkbox"/> Local traffic	Firewall Tunnel to company headquarters Tunnel to London Trusted/Local	Firewall Tunnel to company he Tunnel to London Trusted/Local	Any	✓	
<input checked="" type="checkbox"/> Firewall traffic	Firewall	Any	Any	✓	

Figure 23.60 The Paris filial office — final traffic rules

connect to this branch office).

VPN test

The VPN configuration has been completed by now. At this point, it is recommended to test reachability of the remote hosts in the other remote networks (at remote endpoints of individual tunnels).

For example, the ping or/and tracert operating system commands can be used for this testing.

Kerio Clientless SSL-VPN (Windows)

Kerio Clientless SSL-VPN (thereinafter “*SSL-VPN*”) is a special interface used for secured remote access to shared items (files and folders) in the network protected by *WinRoute* via a web browser. This interface is available only in *WinRoute* on *Windows*.

To a certain extent, the *SSL-VPN* interface is an alternative to *Kerio VPN Client* (see chapter 23). Its main benefit is that it enables an immediate access to a remote network from any location without any special application having been installed and any configuration having been performed (that’s the reason for calling it *clientless*). The main disadvantage of this alternative is that network connections are not transparent. *SSL-VPN* is, in a manner, an alternative to the *My Network Places* system tool) — it does not enable access to web servers or other services in a—remote network.

SSL-VPN is suitable for an immediate access to shared files in remote networks in such environments where it is not possible or useful to use *Kerio VPN Client*.

This chapter addresses configuration details needed for proper functionality of the *SSL-VPN* interface. The *SSL-VPN* interface is described thoroughly in the *Kerio WinRoute Firewall — User’s Guide*.

24.1 Configuration of WinRoute’s SSL-VPN

SSL-VPN interface requirements

For proper functionality of the *SSL-VPN* interface, the following conditions must be met:

1. The *WinRoute* host must be a member of the corresponding domain (*Windows NT* or *Active Directory* domain).
2. User accounts that will be used for connections to *SSL-VPN* must be authenticated at the domain (it is not possible to use local authentication). This implies that the *SSL-VPN* interface cannot be used for accessing shared items in multiple domains or to items at hosts which are not members of any domain.
3. Users who are supposed to be allowed to access the *SSL-VPN* interface needs the right to use *Clientless SSL-VPN* in *WinRoute* (see chapter 15.2).
4. If *WinRoute* is installed on the domain server, the corresponding users need to be allowed to log on to the server locally. Local logon can be allowed under *Domain Controller Security Policy*. For details, refer to [our Knowledge Base](#).

SSL-VPN interface configuration

The *SSL-VPN* interface can be enabled/disabled on the *Web Interface* → *SSL-VPN* in the *Configuration* → *Advanced Options* section.

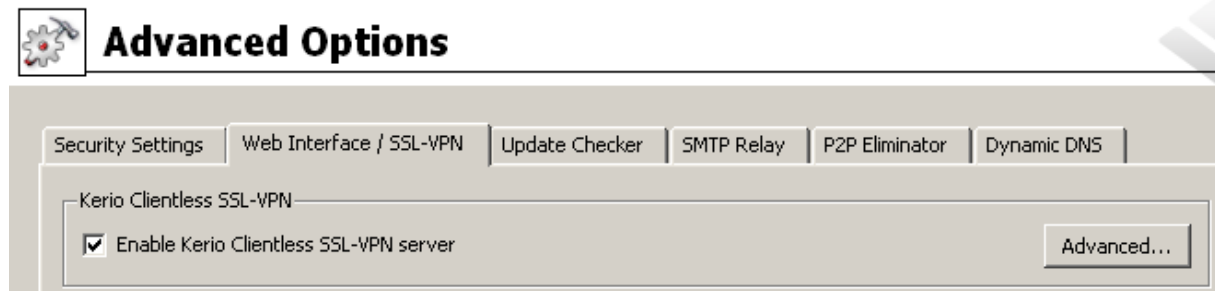


Figure 24.1 Configuration of the SSL-VPN interface

Through the *Advanced* button, you can get to configuration of a port and SSL certificate for the *SSL-VPN* interface.

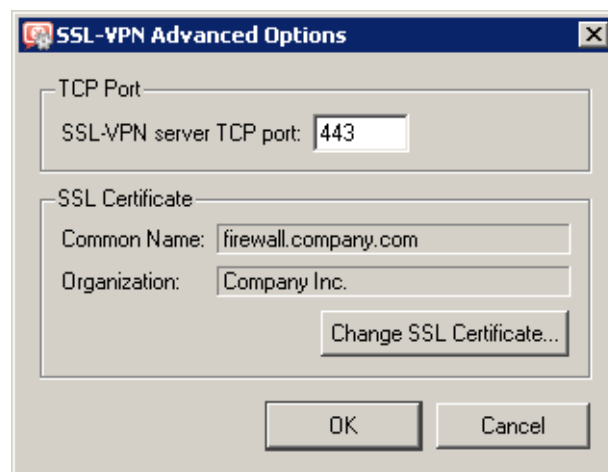


Figure 24.2 Setting of TCP port and SSL certificate for SSL-VPN

SSL-VPN's default port is port 443 (standard port of the *HTTPS* service).

Click *Change SSL Certificate* to create a new certificate for the *SSL-VPN* service or to import a certificate issued by a trustworthy certification authority. When created, the certificate is saved as `sslvpn.crt` and the corresponding private key as `sslvpn.key`. The process of creating/importing a certificate is identical as the one for *WinRoute*'s interface or the VPN server, addressed in detail in chapter [11.1](#).

Hint

Certificates for particular server name issued by a trustworthy certification authority can also be used for the Web interface and the VPN server – it is not necessary to use three different certificates.

Allowing access from the Internet

Access to the *SSL-VPN* interface from the Internet must be allowed by defining a traffic rule allowing connection to the firewall's *HTTPS* service. For details, see chapter [7.4](#).

Name	Source	Destination	Service	Action	Translation
<input checked="" type="checkbox"/> Kerio SSL-VPN 	Any 	Firewall 	HTTPS 		

Figure 24.3 Traffic rule allowing connection to the SSL-VPN interface

Note: If the port for *SSL-VPN* interface is changed, it is also necessary to modify the *Service* item in this rule!

Antivirus control

If at least one antivirus is enabled in *WinRoute* (see chapter [13](#)), all files transferred by the *SSL-VPN* interface can be scanned for viruses.

In default configuration, only files uploaded to hosts in remote private networks are scanned. For connection speed reasons, files downloaded to local hosts from remote networks are not scanned by antiviruses (files downloaded from private networks are considered as trustworthy). Settings of antivirus check can be changed in antivirus configuration — see chapter [13.5](#).

24.2 Usage of the SSL-VPN interface

For access to the interface, most of common graphical web browsers can be used (however, we recommend to use *Internet Explorer* version 6.0 or *Firefox/SeaMonkey* with the core version 1.3 and later). Specify URL in the browser in the

`https://server/`

format, where *server* represents the DNS name or IP address of the *WinRoute* host. If *SSL-VPN* uses another port than the default port for *HTTPS* (443), it is necessary to specify the used port in the URL, e.g.

`https://server:12345/`

Upon a connection to the server, the *SSL-VPN* interface's welcome page is displayed localized to the language set in the browser. If the language defined as preferred is not available, the English version will be used.

Specific settings and troubleshooting

This chapter provides description of advanced features and specific configurations of the firewall. It also includes helpful guidelines for solving of issues which might occur when you use *WinRoute* in your network.

25.1 Configuration Backup and Transfer

If you need to reinstall the firewall's operating system (e.g. in case of new hardware installation), you can easily back up your *WinRoute* configuration including local user accounts and possibly also SSL certificates. This backup can be later used for recovery of this configuration in your new installation of *WinRoute*. This may save significant amount of your time as well as help you avoid solution of problems you have already figured out.

To export or import configuration, go to the *Web Administration* (see chapter 3) and click on the corresponding link right in the welcome page.

Configuration export

Configuration is exported to a *.tgz* package (the *tar* archive compressed by *gzip*) which includes all the key *WinRoute* configuration files. Optionally, it is possible to include the web interface's VPN server's and *SSL-VPN* server's SSL certificates in the package. Exported configuration does not include *WinRoute* license key.

Configuration import

To import configuration, simply browse for or enter the path to the corresponding file which includes the exported configuration (with the *.tgz* extension).

If network interfaces have been changed since the export took place (e.g. in case of exchange of a defective network adapter) or if the configuration is imported from another computer, *WinRoute* will attempt to pair the imported network interfaces with the real interfaces on the machine. This pairing can be customized — you can match each network interface from the imported configuration with one interface of the firewall or leave it unpaired.

If network interfaces cannot be simply paired, it is desirable to check and possibly edit interface group settings (see chapter 5) and/or traffic rules (see chapter 7) after completion of the configuration import.

25.2 Configuration files

This chapter provides clear descriptions of *WinRoute* configuration and status files. This information can be helpful for example when troubleshooting specific issues in cooperation with the *Kerio Technologies* technical support department.

For backup and recovery of your firewall configuration, it is recommended to use configuration export and import tools addressed in chapter [25.1!](#)

Configuration files

All *WinRoute* configuration data is stored in the following files under the same directory where *WinRoute* is installed

(the typical path is C:\Program Files\Kerio\WinRoute Firewall).

The following files are included:

winroute.cfg

Chief configuration file

UserDB.cfg

Information about groups and user accounts.

host.cfg

Preferences for backs-up of configuration, user accounts data, DHCP server database, etc.

logs.cfg

Log configurations

Note: 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.

Files in the following directories are also considered as configuration data:

dbSSL

An automatically generated SSL certificate generated for traffic between the *WinRoute Firewall Engine* and the *Administration Console*.

For details on traffic between the *WinRoute Firewall Engine* and the *Administration Console*, refer to *Kerio Administration Console — Help* (<http://www.kerio.com/firewall/manual>).

sslcert

SSL certificates for all components using SSL for traffic encryption (i.e. the web interface, VPN server and the *Clientless SSL-VPN* interface).

license

If *WinRoute* has already been registered, the `license` folder includes a license key file (including registered trial versions). If *WinRoute* has not been registered yet, the `license` folder is empty.

Status files

In addition, *WinRoute* generates other files and directories where certain status information is saved:

Files:

dnscache.cfg

DNS files stored in the *DNS* module's cache (see chapter [8.1](#)).

leases.cfg

IP addresses assigned by the DHCP server.

This file keeps all information available on the *Leases* tab of the *Configuration* → *DHCP server* section (refer to chapter [8.2](#)).

stats.cfg

Interface statistics (see chapter [20.2](#)) and user statistics (see chapter [20.1](#)) data.

vpnleases.cfg

IP addresses assigned to VPN clients (see chapter [23.2](#)).

Directories:

logs

The *logs* directory stores all *WinRoute* logs (see chapter [22](#)).

star

The *star* directory includes a complete database for statistics of the *WinRoute* web interface.

Handling configuration files

We recommend that *WinRoute Firewall Engine* be stopped prior to any manipulation with the configuration files (backups, recoveries, etc.)! Information contained within these files is loaded and saved only upon starting or stopping the MailServer. All changes to the configuration performed while the *Engine* is running are only stored in memory. All modifications done during *Engine* performance will be overwritten by the configuration in the system memory when the *Engine* is stopped.

25.3 Automatic user authentication using NTLM

WinRoute supports automatic user authentication by the NTLM method (authentication from Web browsers). Users once authenticated for the domain are not asked for username and password.

This chapter provides detailed description on conditions and configuration settings for correct functioning of NTLM.

General conditions

The following conditions are applied to this authentication method:

1. *WinRoute Firewall Engine* is running as a service or it is running under a user account with administrator rights to the *WinRoute* host.
2. The server (i.e. the *WinRoute* host) belongs to a corresponding *Windows NT* or *Active Directory (Windows 2000/2003/2008)* domain.
3. Client host belongs to the domain.
4. User at the client host is required to authenticate to this domain (i.e. local user accounts cannot be used for this purpose).
5. The *NT domain* or the *Active Directory* authentication method (see chapter [15.1](#)) must be set for the corresponding user account under *WinRoute*. NTLM cannot be used for users authenticated only internally inside *WinRoute*.

WinRoute Configuration

NTLM authentication of users from web browsers must be enabled in *Users* → *Authentication Options*. User authentication should be required when attempting to access web pages, otherwise enabling NTLM authentication is meaningless.

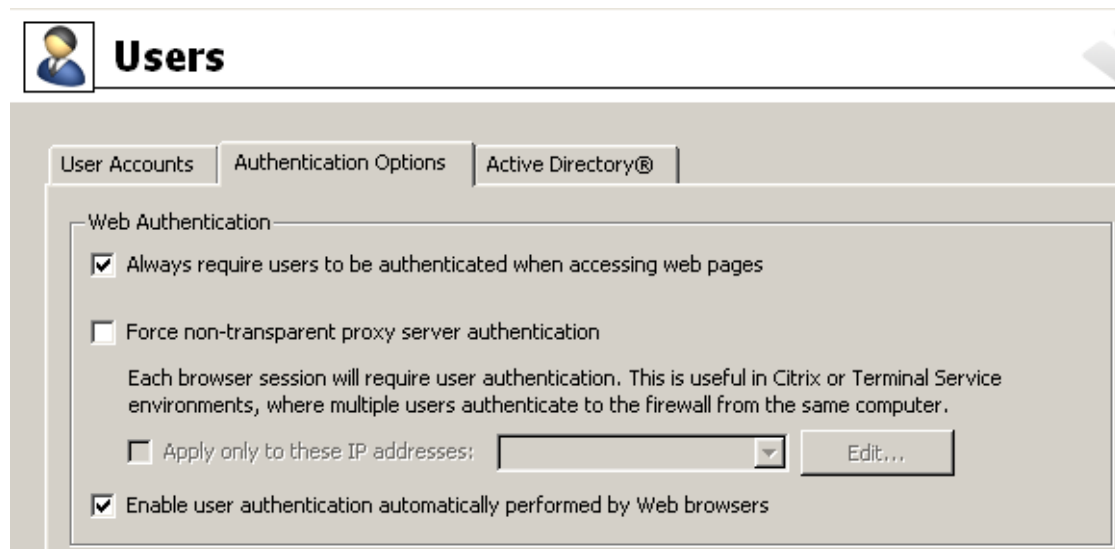


Figure 25.1 NTLM — user authentication options

The configuration of the *WinRoute*'s web interface must include a valid DNS name of the server on which *WinRoute* is running (for details, see chapter [11.1](#)).

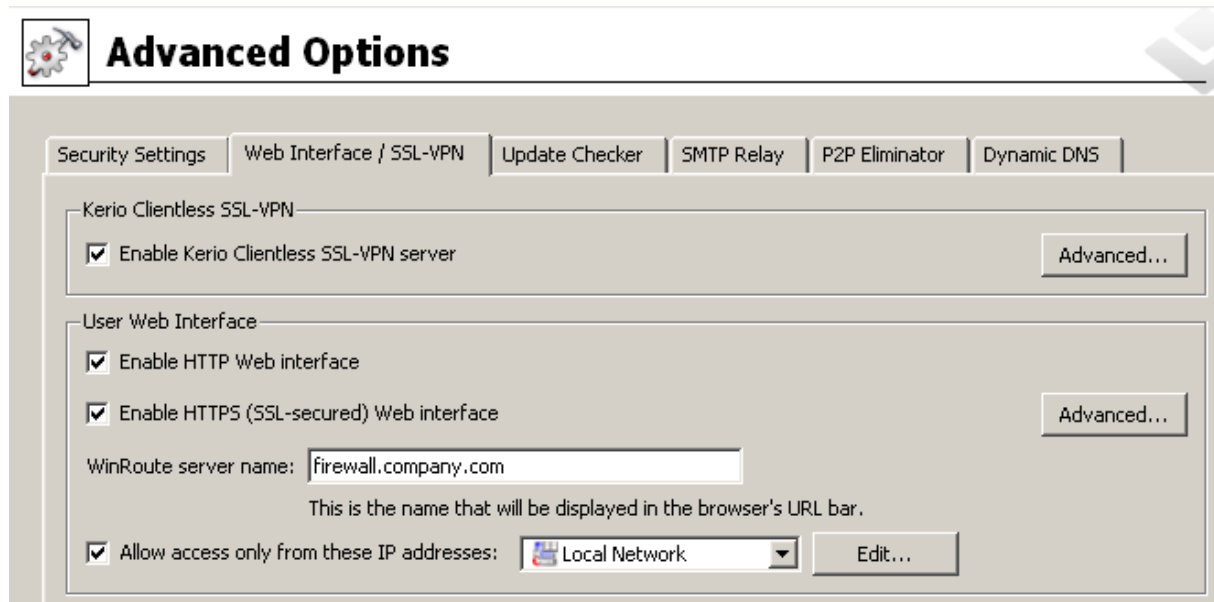


Figure 25.2 Configuration of WinRoute's Web Interface

Note: In the *Software Appliance / VMware Virtual Appliance* edition, the server name is set on the *System Configuration* tab (see chapter [16.1](#)).

Web browsers

For proper functioning of NTLM, a browser must be used that supports this method. By now, the following browsers are suitable:

- *Internet Explorer* version 5.01 or later
- *Firefox* or *SeaMonkey* with the core version *Mozilla* 1.3 or later

NTLM authentication process

NTLM authentication process differs depending on a browser used.

Internet Explorer

NTLM authentication is performed without user's interaction.

The login dialog is displayed only if NTLM authentication fails (e.g. when user account for user authenticated at the client host does not exist in *WinRoute*).

Warning

One reason of a NTLM authentication failure can be invalid login username or password saved in the *Password Manager* in *Windows* operating systems (*Control Panels* → *User Accounts* → *Advanced* → *Password Manager*) applying to the corresponding server (i.e. the *WinRoute* host). In such a case, *Internet Explorer* sends saved login data instead of NTLM authentication of the user currently logged in. Should any problems regarding

NTLM authentication arise, it is recommended to remove all usernames/passwords for the server where *WinRoute* is installed from the *Password Manager*.

Firefox/SeaMonkey

The browser displays the login dialog. For security reasons, automatic user authentication is not used by default in the browser. This behavior of the browser can be changed by modification of configuration parameters — see below.

If authentication fails and direct connection is applied, the firewall's login page is opened automatically (refer to chapter [11.2](#)). The login dialog is displayed if proxy server is used.

Note: If NTLM authentication fails by any reason, details are recorded in the *error* log (see chapter [22.8](#)).

Firefox/SeaMonkey configuration

Configuration can be changed to enable automatic NTLM authentication — leaving out the login dialog. Check the following example:

1. Insert `about:config` in the browser's address bar. The list of configuration parameters is displayed.
2. Set corresponding configuration parameter(s) using the following instructions:
 - For direct connection (proxy server is not set in the browser):
Look up the `network.automatic-ntlm-auth.trusted-uris` parameter. Use the *WinRoute* host's name as a value for this parameter (e.g. `server` or `server.company.com`). This name must match the server name set under *Configuration* → *Advanced Options* → *Web Interface* (see chapter [11.1](#)).
Note: It is not possible to use IP address as a value in this parameter!
 - If *WinRoute* proxy server is used:
Look up the `network.automatic-ntlm-auth.allow-proxies` parameter and set its value to `true`.

Configuration changes are applied right away, i.e. it is not necessary to restart the browser.

25.4 FTP on WinRoute's proxy server

Proxy server in *WinRoute*, version 6.0.2 and later (see chapter [8.4](#)), supports FTP. When using this method of accessing FTP servers, it is necessary to keep in mind specific issues regarding usage of the proxy technology and parameters of *WinRoute*'s proxy server.

1. It is necessary that the FTP client allows configuration of the proxy server. This condition is met for example by web browsers (*Internet Explorer*, *Firefox/SeaMonkey*, *Opera*, etc.), *Total Commander* (originally *Windows Commander*), *CuteFTP*, etc.

Terminal FTP clients (such as the `ftp` command in *Windows* or *Linux*) do not allow configuration of the proxy server. For this reason, they cannot be used for our purposes.

2. To connect to FTP servers, the proxy server uses the passive FTP mode. If FTP server is protected by a firewall which does not support FTP (this is not a problem of *WinRoute*), it is not possible to use proxy to connect to the server.
3. Setting of FTP mode in the client is irrelevant for usage of the proxy server. Only one network connection used by the FTP protocol is always established between a client and the proxy server.

Note: It is recommended to use FTP over proxy server only in cases where it is not possible to connect directly to the Internet (see chapter [8.4](#)).

Example of a client configuration: web browser

Web browsers allow to set the proxy server either globally or for individual protocols. In our example, configuration of *Internet Explorer 6.0* focused (configuration of any other browsers is almost identical).

1. In the browser's main menu, select *Tools* → *Internet Options*, open the *Connections* tab and click on the *LAN Settings* option.
2. Enable the *Use a proxy server for your LAN* option and enter the IP address and port of the proxy server. IP address of the proxy server is the address of the *WinRoute*'s host interface which is connected to the local network; the default port of the proxy server is 3128 (for details, refer to chapter [8.4](#)). It is also recommended to enable the *Bypass proxy server for local addresses* option — using proxy server for local addresses would slow down traffic and overburden *WinRoute*.

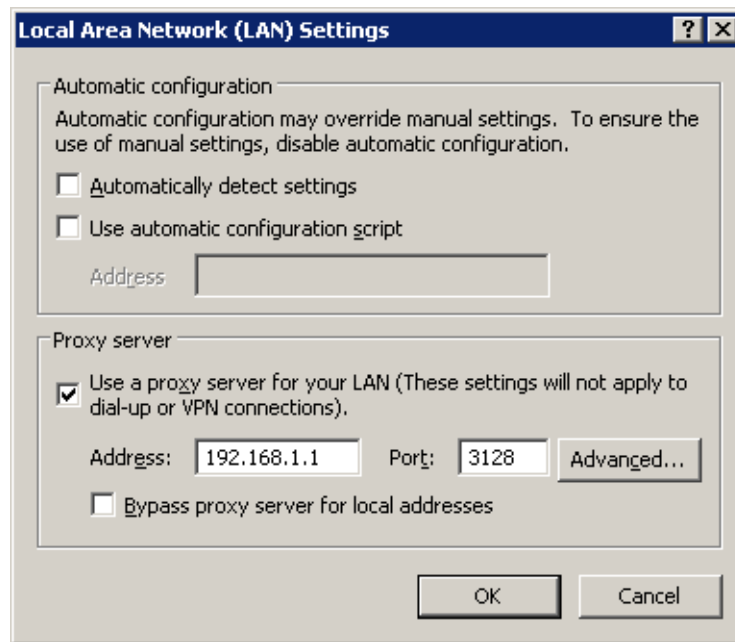


Figure 25.3 Configuring proxy server in Internet Explorer 6.0

Hint

To configure web browsers, you can use a configuration script or the automatic detection of configuration. For details, see chapter [8.4](#).

Note: Web browsers used as FTP clients enable only to download files. Uploads to FTP server via web browsers are not supported.

Example of a client configuration: Total Commander

Total Commander allows either single connections to FTP server (by the *Net* → *FTP - New Connection* option available in the main menu) or creating a bookmark for repeated connections (*Net* → *FTP - Connect*). The proxy server must be configured individually for each FTP connection (or for each bookmark).

1. In the *FTP: connection details* dialog, enable the *Use firewall (proxy server)* option and click *Change*.
2. In the *Firewall settings* dialog box, select *HTTP Proxy with FTP support*. In the *Host name* textbox, enter the proxy server's IP address and port (separated by a colon, e.g. 192.168.1.1:3128). The *User name* and *Password* entries are optional (*WinRoute* does not use this information).

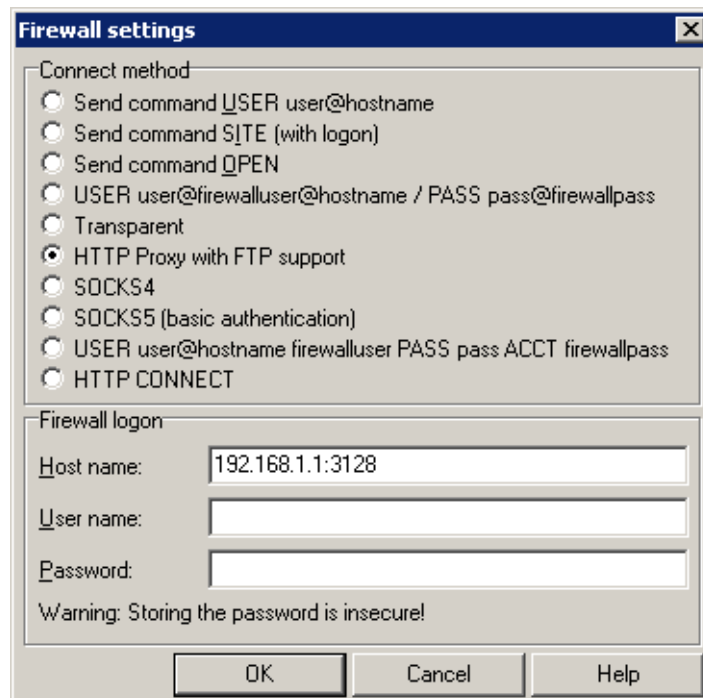


Figure 25.4 Setting proxy server for FTP in Total Commander

Hint

The defined proxy server is indexed and saved to the list of proxy servers automatically. Later, whenever you are creating other FTP connections, you can simply select a corresponding proxy server in the list.

25.5 Internet links dialed on demand

If an on-demand dial-up link is used (see chapter 6.2), consider specific behavior of this connection type. If the network and/or the firewall are not configured correctly, the link may stay hung-up even if the local network sends requests for Internet connection or it may be dialed unintentionally.

Information provided in this chapter should help you understand the principle and behavior of on-demand dial-ups and avoid such problems.

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 — see chapter 6.2).

Second, there must be no default gateway in the operating system (no default gateway must be defined for any network adapter). This condition does not apply to the dial-up line which is used for the Internet connection — this line will be configured in accordance with information provided by the ISP.

If *WinRoute* receives a [packet](#) from the local network, it will compare it with the system routing table. If the packets goes out to the Internet, no record will be found, since there is no default route in the routing table. Under usual circumstances, the packet would be dropped and a control message informing about unavailability of the target would be sent to the sender. 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.

To avoid undesired dialing of the line, line dialing is allowed by certain packet types only. The line can be dialed only by UDP or TCP packets with the *SYN* flag (connection attempts). Demand dialing is disabled for *Microsoft Networks* services (sharing of files and printers, etc.).

Since this moment, the default route exists and other packets directed to the Internet will be routed via a corresponding line. The line may be either disconnected manually or automatically if idle for a certain time period. When the line is hung-up, the default route is removed from the routing table. Any other packet directed to the Internet redials the line.

Note:

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. Only one link can be set for on-demand dialing in *WinRoute*. *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 [18.1](#)). 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 Internet line 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 traffic cannot continue.

For these reasons, the *WinRoute's* DNS module enables automatic dialing (if the DNS server cannot respond to the request itself). This feature is bound to on-demand dialing.

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 the *DNS* module because it can dial the line if necessary.

If there is a domain based on *Active Directory* in the LAN (domain server with *Windows Server 2000/2003/2008*), it is necessary to use *Microsoft* DNS server, because communication with *Active Directory* uses special types of DNS request. *Microsoft* DNS server does not support automatic dialing. Moreover, it cannot be used at the same host as the *DNS* module 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 Server* with *Active Directory* and *Microsoft* DNS are running.

4. If the *DNS* module 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 fire-wall). In *Windows*, go to TCP/IP properties in interfaces connected to the LAN and set the IP address of this interface as the primary DNS server.

- The *Proxy server* in *WinRoute* (see chapter 8.4) 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.

Unintentionally dialed link — application of on-demand dial rules

Demand dial functions may cause unintentional dialing. It's usually caused by DNS requests which cannot be responded by the *DNS* module and so it dials the line instead to forward them to another DNS server. The following causes apply:

- User host generates a DNS query in the absence of the user. This traffic attempt may be an active object at a local HTML page or automatic update of an installed application.
- The *DNS* module 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 8.1).

Note: Undesirable traffic causing unintentional dialing of a link can be blocked by *WinRoute* traffic rules (see chapter 7.3). However, the best remedy for any pain is always removal of its cause (e.g. perform antivirus check on the corresponding workstation, etc.).

To avoid unintentional dialing based on DNS requests, *WinRoute* allows definition of rules where DNS names are specified for which the line can be dialed or not. To define these rules, click on *Advanced* in *Configuration* → *Interfaces* (in the *A Single Internet Link — Dial on Demand* mode).

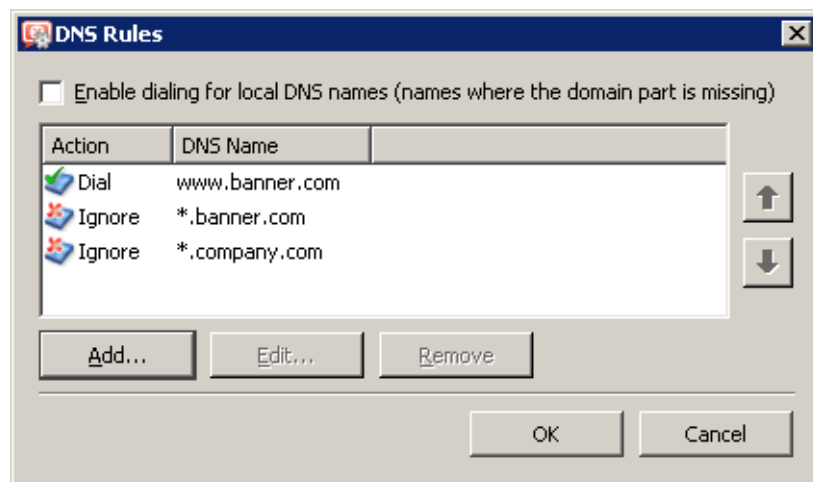


Figure 25.5 Dial on demand rules (for dialing based on DNS queries)

Either full *DNS name* or only its end or beginning completed by an asterisk (*) can be specified in the rule. An asterisk may stand for any number of characters.

Rules are ordered in a list which is processed from the top downwards (rules order can be modified with the arrow buttons 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 the *DNS* module when demanded.

In *Actions* for DNS name, you can select either the *Dial* or the *Ignore* option. Use the second option to block dialing of the line in response to a request for this DNS name. The *Dial* action can be used to create complex rule combinations. For example, dial can be permitted for one name within the domain and denied for the others (see figure [25.5](#)).

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's name is `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 the *DNS* module). Set by default, the *DNS* module 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* module 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 *Dial On Demand* dialog window). In other cases, it is recommended to leave the option disabled (again, the line can be dialed undesirably).

Technical support

Free email and telephone technical support is provided for *Kerio WinRoute Firewall*. Contacts and more information can be found at <http://www.kerio.com/support>. Our technical support staff is ready to help you with any problem you might have.

You can also solve many problems alone (and sometimes even faster). Before you contact our technical support, please take the following steps:

- Try to look up the answer in this manual. Individual chapters describe features and parameters of *WinRoute* components in detail.
- If you have not found answers here, try to find them at our website, under [Technical Support](#).

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.

26.1 Essential Information

To send a request to our technical support, use the contact form at <http://support.kerio.com/>.

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. Please specify at least the following information:

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 *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 *Administration Console*.
- If you use dial-up, connect to the Internet.
- In the *Administration Console* use the *Ctrl+S* keys.

The text file will be stored in the home directory of the logged user.

(e.g. C:\Documents and Settings\Administrator)

as `kerio_support_info.txt`.

Note: The `kerio_support_info.txt` is generated by the *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

In the directory where *WinRoute* is installed

(the typical path is `C:\Program Files\Kerio\WinRoute Firewall`)

the `logs` subdirectory is created. There you can find files `error.log` and `warning.log`. Attach these two files to your email to our technical support.

License type and license number

Please specify whether you have purchased any *WinRoute* license or if you use the trial version. Requirements of owners of valid licenses are always preferred.

26.2 Tested in Beta version

As to increase quality of our products, *Kerio Technologies* releases essential versions of our products as so called beta versions. Beta versions are product versions which include all projected new features, however, these functions and the product itself are still under development. Volunteers can test these versions and provide us with feedback to help us improve the product and fix bugs.

The feedback from beta testers is essential for the product's development. Therefore, *WinRoute* beta versions include extensions and modules helping testers communicate smoothly with *Kerio Technologies*.

For details on beta versions and their testing, refer to <http://www.kerio.com/betas>.

Appendix A

Legal Notices

Microsoft®, *Windows®*, *Windows NT®*, *Windows Vista™*, *Internet Explorer®*, *ActiveX®*, and *Active Directory®* are registered trademarks or trademarks of *Microsoft Corporation*.

Mac OS® and *Safari™* are registered trademarks or trademarks of *Apple Computer, Inc.*

Linux® is registered trademark kept by Linus Torvalds.

Mozilla® and *Firefox®* are registered trademarks of *Mozilla Foundation*.

Kerberos™ is trademark of *Massachusetts Institute of Technology (MIT)*.

Other names of real companies and products mentioned in this document may be registered trademarks or trademarks of their owners.

Appendix B

Used open source items

Kerio WinRoute Firewall contains the following open-source software (open source):

bindlib

Copyright ©1983, 1993 The Regents of the University of California. All rights reserved.
Portions Copyright ©1993 by Digital Equipment Corporation.

Firebird

This software embeds modified version of *Firebird* database engine distributed under terms of *IPL* and *IDPL* licenses.

All copyright retained by individual contributors — original code Copyright © 2000 *Inprise Corporation*.

The original source code is available at:

<http://www.firebirdsql.org/>

h323plus

This product includes unmodified version of the *h323plus* library distributed under *Mozilla Public License (MPL)*.

The original source code is available at:

<http://h323plus.org/>

KIPF — driver

Kerio IP filter driver for Linux (*WinRoute's* network interface for Linux):

Copyright © Kerio Technologies s.r.o.

Homepage: <http://www.kerio.com/>

Kerio IP filter driver for Linux is distributed and licensed under *GPL* version 2.

The complete source code is available at:

<http://download.kerio.com/dwn/libkipf.tgz>

KIPF — API

Kerio IP filter driver for Linux (API library of the *WinRoute's* network interface for Linux)

Copyright © Kerio Technologies s.r.o.

Homepage: <http://www.kerio.com/>

Kerio IP filter driver for Linux API library is distributed and licensed under *LGPL* version 2.

The complete source code is available at:

<http://download.kerio.com/dwn/libkipf.tgz>

KVNET — driver

Kerio Virtual Network Interface driver for Linux (driver for the *Kerio VPN* virtual network adapter)

Copyright © Kerio Technologies s.r.o.

Homepage: <http://www.kerio.com/>

Kerio Virtual Network Interface driver for Linux is distributed and licensed under *GPL* version 2.

The complete source code is available at:

<http://download.kerio.com/dwn/libkvnet.tgz>

KVNET — API

Kerio Virtual Network Interface driver for Linux API library (API library for the driver of the *Kerio VPN* virtual network adapter)

Copyright © Kerio Technologies s.r.o.

Homepage: <http://www.kerio.com/>

Kerio Virtual Network Interface driver for Linux API library is distributed and licensed under *LGPL* version 2.

The complete source code is available at:

<http://download.kerio.com/dwn/libkvnet.tgz>

libcurl

Copyright © 1996-2008 Daniel Stenberg.

libiconv

libiconv converts from one character encoding to another through Unicode conversion. *WinRoute* include a modified version of this library distributed upon the *LGPL* license in version 3.

Copyright ©1999-2003 Free Software Foundation, Inc.

Author: Bruno Haible

Homepage: <http://www.gnu.org/software/libiconv/>

Complete source code of the customized version of *libiconv* library is available at:

<http://download.kerio.com/dwn/kwf-iconv.zip>

libxml2

Copyright © 1998-2003 Daniel Veillard. All Rights Reserved.

Copyright © 2000 Bjorn Reese and Daniel Veillard.

Copyright © 2000 Gary Pennington and Daniel Veillard

Copyright © 1998 Bjorn Reese and Daniel Stenberg.

OpenSSL

This product contains software developed by *OpenSSL Project* designed for *OpenSSL Toolkit* (<http://www.openssl.org/>).

This product includes cryptographic software written by Eric Young.

This product includes software written by Tim Hudson.

PHP

Copyright © 1999-2006 The PHP Group. All rights reserved.

This product includes *PHP* software available for free at:

<http://www.php.net/software/>

php_mbstring

Copyright © 2001-2004 The PHP Group.

Copyright © 1998-2002 HappySize, Inc. All rights reserved.

Prototype

Framework in JavaScript.

Copyright © Sam Stephenson.

The *Prototype* library is freely distributable under the terms of a *MIT* license.

For details, see the *Prototype* website: <http://www.prototypejs.org/>

ptlib

This product includes unmodified version of the *ptlib* library distributed under *Mozilla Public License (MPL)*.

The original source code is available at:

<http://h323plus.org/>

zlib

Copyright © Jean-Loup Gailly and Mark Adler.

Glossary of terms

ActiveX

This *Microsoft's* proprietary technology is used for creation of dynamic objects for web pages. This technology provides many features, such as writing to disk or execution of commands at the client (i.e. on the host where the Web page is opened). This technology provides a wide range of features, such as saving to disk and running commands at the client (i.e. at the computer where the Web page is opened). Using *ActiveX*, virus and worms can for example modify telephone number of the dial-up.

ActiveX is supported only by *Internet Explorer* in *Microsoft Windows* operating systems.

Cluster

A group of two or more workstations representing one virtual host (server). Requests to the virtual server are distributed among individual hosts in the cluster, in accordance with a defined algorithm. Clusters empower performance and increase reliability (in case of dropout of one computer in the cluster, the virtual server keeps running).

Connections

A virtual bidirectional communication channel between two hosts.

[See also TCP](#)

DDNS

DDNS (*Dynamic Domain Name System*) is DNS with the feature of automatic update of records.

Default gateway

A network device or a host where so called default path is located (the path to the Internet). To the address of the default gateway such packets are sent that include destination addresses which do not belong to any network connected directly to the host and to any network which is recorded in the system routing table.

In the system routing table, the default gateway is shown as a path to the destination network *0.0.0.0* with the subnet mask *0.0.0.0*.

Note: Although in *Windows* the default gateway is configured in settings of the network interface, it is used for the entire operating system.

DHCP

DHCP (*Dynamic Host Configuration Protocol*) Serves automatic IP configuration of computers in the network. IP addresses are assigned from a scope. Besides IP addresses, other parameters can be associated with client hosts, such as the default gateway address, DNS server address, local domain name, etc.

DMZ

DMZ (demilitarized zone) is a reserved network area where services available both from the Internet and from the LAN are run (e.g. a company's public web server). DMZ provides an area, where servers accessible for public are located separately, so they cannot be misused for cracking into the LAN.

More information can be found for example at [Wikipedia](#).

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. Names are sorted in hierarchized domains.

Firewall

Software or hardware device that protects a computer or computer network against attacks from external sources (typically from the Internet).

In this guide, the word *firewall* represents the *WinRoute* host.

FTP

File Transfer Protocol. The FTP protocol uses two types of TCP connection: control and data. The control connection is always established by a client. Two FTP modes are distinguished according to a method how connection is established:

- *active mode* — data connection is established from the server to a client (to the port specified by the client). This mode is suitable for cases where the firewall is at the server's side, however, it is not supported by some clients (e.g. by web browsers).
- *passive mode* — data connection is established also by the client (to the port required by the server). This mode is suitable for cases where the firewall is at the client's side. It should be supported by any FTP client.

Note: *WinRoute* includes special support (protocol inspector) for FTP protocol. Therefore, both FTP modes can be used on LAN hosts.

Gateway

Network device or a computer connecting two different subnets. If traffic to all the other (not specified) networks is routed through a gateway, it is called the default gateway.

[See also default gateway.](#)

Greylisting

A method of protection of *SMTP* servers from spam. If an email message sent by an unknown sender is delivered to the server, the server rejects it for the first time (so called temporary delivery error). Legitimate senders attempt resend the message after some time. *SMTP* server lets the message in and considers the sender as trustworthy since then, not blocking their messages any longer. Most spam senders try to send as great volume in as short time as possible and stay anonymous. Therefore, they usually do not repeat sending the message and focus on another *SMTP* server.

More information (in English) can be found for example at [Wikipedia](#).

Ident

The *Ident* protocol is used for identification of user who established certain TCP connection from a particular (multi-user) system. The *Ident* service is used for example by IRC servers, FTP servers and other services.

More information (in English) can be found for example at [Wikipedia](#).

IMAP

Internet Message Access Protocol (IMAP) enables clients to manage messages stored on a mail server without downloading them to a local computer. This architecture allows the user to access his/her mail from multiple locations (messages downloaded to a local host disk would not be available from other locations).

IP address

IP address is a unique 32-bit number used to identify the host in the Internet. It is specified by numbers of the decimal system (0–255) separated by dots (e.g. 195.129.33.1). 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).

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.

Kerberos

Kerberos is a system used for secure user authentication in network environments. It was developed at the *MIT* university and it is a standard protocol used for user authentication under *Windows 2000/2003/2008*. Users use their passwords to authenticate to the central server (*KDC, Key Distribution Center*) and the server sends them encrypted tickets which can be used to authenticate to various services in the network. In case of the *Windows 2000/2003/2008* domains, function of *KDC* is provided by the particular domain server.

LDAP

LDAP (Lightweight Directory Access Protocol) is an Internet protocol used to access directory services. Information about user accounts and user rights, about hosts included in the network, etc. are stored in the directories.

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.

Detailed description (in English) can be found for example at [Wikipedia](#).

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.

P2P network

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 of big volumes of data (this sharing is mostly illegal). *DirectConnect* and *Kazaa* are the most popular ones.

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.

Policy routing

Advanced routing technology using additional information apart from IP addresses, such as source IP address, protocols etc.

[See also routing table.](#)

POP3

Post Office Protocol is an email accessing protocol that allows users to download messages from a server to a local disk. It is suitable for clients who don't have a permanent connection to the Internet.

Port

16-bit number (1–65535) used by TCP and UDP for application (services) identification on a given computer. More than one application can be run at a host simultaneously (e.g. WWW server, mail client, FTP client, etc.). Each application is identified by a port number.

Ports 1–1023 are reserved and used by well known services (e.g. 80 = WWW). Ports above 1023 can be freely used by any application.

PPTP

Microsoft's proprietary protocol used for design of virtual private networks.

[See chapters and sections 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

Protocol inspector

WinRoute's subroutine, 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*.

Proxy server

Older, but still wide-spread method of Internet connection sharing. Proxy servers connect clients and destination servers.

A proxy server works as an application and it is adapted for several particular application protocols (i.e. HTTP, FTP, Gopher, etc.). It requires also support in the corresponding client application (e.g. web browser). Compared to NAT, the range of featured offered is not so wide.

Router

A computer or device with one or more network interfaces between which it handles packets by following specific rules (so called routes). The router's goal is to forward packets only to the destination network, i.e. to the network which will use another router which would handle it on. This saves other networks from being overloaded by packets targeting another network.

[See also routing table.](#)

Routing table

The information used by routers when making packet forwarding decisions (so called routes). Packets are routed according to the packet's destination IP address. On *Windows*, routing table can be printed by the `route print` command, while on *Unix* systems (*Linux*, *Mac OS X*, etc.) by the `route` command.

Script

A code that is run on the Web page by a client (Web browser). Scripts are used for generating of dynamic elements on Web pages. However, they can be misused for ads, exploiting of user information, etc. Modern Web browsers usually support several script languages, such as *JavaScript* and *Visual Basic Script (VBScript)*.

SMTP

Simple Mail Transfer Protocol is used for sending email between mail servers. The SMTP envelope identifies the sender/recipient of an email.

Spam

Undesirable email message, usually containing advertisements.

Spoofing

Spoofing means using false IP addresses in packets. This method is used by attackers to make recipients assume that the packet is coming from a trustworthy IP address.

SSL

SSL is a protocol used to secure and encrypt network communication. SSL was originally designed in order to guarantee secure transfer of Web pages over HTTP protocol. Nowadays, it is used by almost all standard Internet protocols (SMTP, POP3, IMAP, LDAP, etc.).

At the beginning of communication, an encryption key is requested and transferred using asymmetrical encryption. This key is then used to encrypt (symmetrically) the data.

Subnet mask

Subnet mask divides an IP address in two parts: network mask and an address of a host in 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. Number one in a subnet mask represents a bit of the network address and zero stands for a host's address bit. All hosts within a particular subnet must have identical subnet mask and network part of IP address.

TCP

Transmission Control Protocol is a transmission protocol which ensures reliable and sequential 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 following 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. This version is approved by the IETF and it is accepted by all the top IT companies (i.e. *Microsoft Corporation*).

UDP

User Datagram Protocol is a transmission protocol which transfers data through individual messages (so called datagrams). It does not establish new connections nor it provides reliable and sequential 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 *PPTP (Point-to-Point Tunneling Protocol)* and *Microsoft's IPsec*.

WinRoute contains a proprietary *VPN* implementation called *Kerio VPN*.

WINS

The *WINS (Windows Internet Name Service)* service is used for resolution of hostnames to IP addresses within *Microsoft Windows* networks.

Index

A

- Active Directory [196](#)
 - domain mapping [204](#)
 - import of user accounts [203](#)
 - mapping of other domains [208](#)
- administration [27](#)
 - remote [18, 215](#)
- Administration Console [27](#)
 - columns [31](#)
 - views setup [31](#)
- alerts [243](#)
 - overview [246](#)
 - settings [243](#)
 - templates [245](#)
- anti-spoofing [222](#)
- antivirus check [11, 167](#)
 - conditions [167](#)
 - external antivirus [170](#)
 - file size limits [171](#)
 - HTTP and FTP [172](#)
 - McAfee [168](#)
 - protocols [171](#)
 - rules for file scanning [174](#)
 - settings [168](#)
 - SMTP and POP3 [176](#)

B

- bandwidth limiter [130](#)
 - configuration [130](#)
 - detection principle [135](#)
- beta version [352](#)
- BOOTP [118](#)

C

- cache
 - directory [125](#)
 - DNS [105](#)
 - size [126](#)

- URL exceptions [127](#)
- certificate
 - SSL-VPN [336](#)
 - VPN server [286](#)
 - Web Interface [144](#)
- Clientless SSL-VPN [335](#)
 - antivirus check [337](#)
 - certificate [336](#)
 - configuration [336](#)
 - deployment [337](#)
 - port [336](#)
 - traffic rule [337](#)
 - user right [198, 213](#)
- configuration files [339](#)
 - manipulation [340](#)
- conflict
 - port [10](#)
 - software [10](#)
 - system services [15](#)
- connection failover [62](#)

D

- DDNS [119](#)
- DHCP [110](#)
 - default options [111](#)
 - IP scopes [111](#)
 - lease reservations [115](#)
 - leases [116](#)
- dial-up
 - dialing scripts [61](#)
 - hangup if idle [61](#)
- dial on demand [57, 346](#)
 - unintentional dialing [349](#)
- DNS [104](#)
- DNS
 - DNS Forwarder [104](#)
 - forwarding rules [108](#)
 - hosts file [106](#)

local domain [107](#)
dynamic DNS [119](#)

F

FTP [147](#), [186](#), [343](#)
filtering rules [162](#)
full cone NAT [87](#)

G

groups
interface throughput charts [47](#)
IP address [180](#)
of forbidden words [160](#)
URL [187](#)
user groups [190](#), [196](#), [210](#)

H

H.323 [186](#)
hairpinning [102](#)
HTTP [147](#)
cache [124](#)
content rating [154](#)
filtering by words [158](#)
logging of requests [154](#)
proxy server [121](#)
URL Rules [148](#)

I

import
user accounts [203](#)
installation [11](#)
interface throughput charts [47](#)
anti-spoofing [222](#)
Dial-In [48](#)
groups [47](#)
Internet connection [53](#)
back-up [62](#)
dial on demand [57](#), [346](#)
leased line [54](#)
load balancing [66](#)
unintentional dialing [349](#)
IPSec [87](#)

K

Kerberos [196](#)
Kerio Administration Console [23](#)
Kerio Web Filter [154](#)
deployment [156](#)
parameters configuration [155](#)
website categories [156](#)

L

language
Administration Console [28](#)
of alerts [246](#)
Web Administration [28](#)
license [32](#)
expiration [43](#)
information [33](#)
license key [32](#)
license types [32](#)
number of users [33](#)
optional components [32](#)
user counter [45](#)
license key [43](#)
load balancing [66](#)
optimization [97](#)
reserved link [96](#)
localizations
Administration Console [28](#)
of alerts [246](#)
Web Administration [28](#)
log [262](#)
alert [269](#)
config [269](#)
connection [270](#)
debug [271](#)
dial [273](#)
error [275](#)
filter [276](#)
http [277](#)
security [278](#)
settings [262](#)
sslvpn [280](#)
warning [280](#)
web [281](#)

Index

M

media hairpinning 102
multihoming 93

N

NAT 84, 90
 full cone NAT 87, 101
NT domain
 import of user accounts 203
NTLM 138, 139
 configuration of web browsers 343
 deployment 340
 WinRoute configuration 341

P

P2P Eliminator 218
Peer-to-Peer (P2P) networks 218
 allow 198, 213
 deny 218
 detection 235
 ports 220
 speed limit 218
policy routing 95
port
 SSL-VPN 336
port mapping 74, 88, 91
probe hosts 65, 70
product registration 32
protocol inspector 89, 185, 186
 retirement 99
proxy server 121, 343
 parent 123

Q

Quick Setup 7
quota
 settings 256
 speed limit 130

R

ranges
 time 181, 182
RAS 118

registration
 at the Kerio website 43
 of purchased product 39
 trial version 36
relay SMTP server 229
routing table 224
 static routes 225

S

services 82, 183
SIP 186
SSL-VPN 335
 antivirus check 337
 certificate 336
 configuration 336
 deployment 337
 port 336
 traffic rule 337
 user right 198, 213
StaR 254
 conditions for statistics 255
 enable/disable gathering of statistic data 254
 overview 259
 settings 256
statistics 248
 conditions for statistics 255
 interface throughput charts 250
 in the Web interface 254
 Kerio StaR 254
 monitoring 254
 overview 259
 settings 256
 user groups 248
status information 231
 active hosts 231
 connections 238
subscription
 expiration 43
Syslog 264
system requirements 11

T

technical support 351

traffic policy [71](#)
 created by wizard [75](#)
 default rule [77](#)
 definition [78](#)
 exceptions [95](#)
 Internet access limiting [94](#)
 wizard [71](#)

transparent proxy [124](#)

Trial ID [37](#)

TTL [125, 128](#)

U

uninstallation [19](#)

update
 antivirus [168](#)
 WinRoute [216](#)

upgrade [13, 19](#)
 automatic update [216](#)

UPnP
 settings [227](#)
 system services [16](#)

user accounts [190](#)
 definition [191](#)
 domain mapping [204](#)
 in traffic rules [98](#)
 local [192, 193](#)
 mapped [192](#)
 templates [192, 195](#)

user authentication [137](#)
 authentication methods [195](#)
 automatic login [201](#)
 configuration [138](#)

V

VPN [283](#)
 client [198, 213, 289](#)
 configuration example [297](#)
 Kerio Clientless SSL-VPN [335](#)
 Kerio VPN [283](#)
 routing [296](#)
 server [48, 284](#)
 SSL certificate [286](#)
 tunnel [291](#)

VPN client [289](#)
 DNS [286](#)
 routing [288](#)
 static IP address [202](#)
 WINS [288](#)

VPN tunnel [291](#)
 configuration [291](#)
 DNS [293](#)
 routing [293](#)
 traffic policy [295](#)

W

Web Interface [141](#)

Web interface
 automatic configuration [123](#)
 configuration script [124](#)

Web Interface
 parameters configuration [141](#)
 ports [143](#)
 SSL certificate [144](#)
 user authentication [146](#)

Windows
 Internet Connection Sharing [15, 16](#)
 security center [17](#)
 Windows Firewall [15, 16](#)

WinRoute Engine Monitor [23, 24](#)

WinRoute Firewall Engine [23](#)

wizard
 configuration [17](#)
 traffic rules [71](#)

