

# **Kerio Personal Firewall 2.1**

## **User's Guide**

**Kerio Technologies**

© 1997-2002 Kerio Technologies. All Rights Reserved.

Printing date: March 27, 2002

*Windows* is a trademark of Microsoft Corporation.

# Contents

---

- 1 Introduction ..... 5**
  - 1.1 System requirements ..... 5
  - 1.2 Installation ..... 6
  
- 2 Administration ..... 7**
  - 2.1 Personal Firewall Components ..... 7
  - 2.2 Securing Access to the Administration ..... 8
  - 2.3 Administration Login ..... 8
  - 2.4 Personal Firewall Status Window ..... 9
  
- 3 Security Settings ..... 13**
  - 3.1 Introduction to TCP/IP ..... 13
  - 3.2 How does Kerio Personal Firewall work? ..... 14
  - 3.3 IP Address Groups ..... 14
  - 3.4 Levels of Security ..... 15
  - 3.5 Interaction with the User ..... 16
  - 3.6 Packet Filtering Rules ..... 18
  - 3.7 Microsoft Networking ..... 22
  - 3.8 Application MD5 Signatures ..... 24
  - 3.9 Internet Gateway Protection ..... 26
  
- 4 Firewall Logging ..... 27**
  - 4.1 Logging Configuration ..... 27
  - 4.2 Filter.log file ..... 28
  
- 5 Index ..... 31**

---

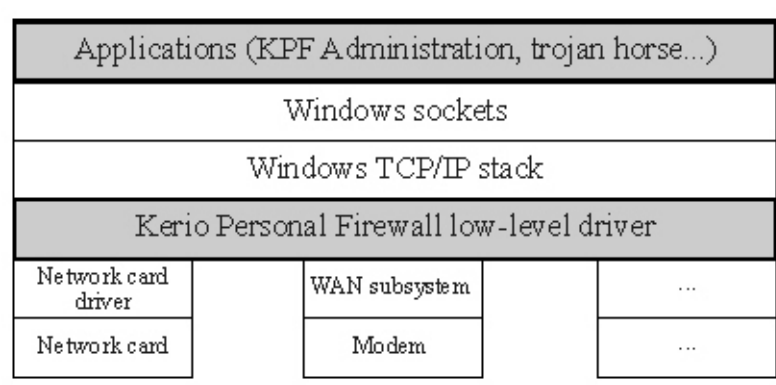
## Chapter 1

# Introduction

---

*Kerio Personal Firewall* is a small and easy to use system designed for protecting a personal computer against hacker attacks and data leaks. It is based on the ICSA certified technology used in the *WinRoute* firewall.

The firewall itself runs as a background service, using a special low-level driver loaded into the system kernel. This driver is placed at the lowest possible level above the network hardware drivers. Therefore, it has absolute control over all passing packets and is able to ensure complete protection of the system it is installed on.



## 1.1 System requirements

The following minimum configuration is recommended for *Kerio Personal Firewall*:

- CPU Intel Pentium or 100% compatible
- 32 MB RAM
- 3 MB hard drive space (for installation only; at least 10 MB of additional space is recommended for logging)
- Windows 98 / Me / NT 4.0 / 2000 / XP

*Kerio Personal Firewall* is designed for protecting computers NOT running *WinRoute Pro* or *WinRoute Lite*. These products use the same technology for security and may cause conflicts with *Kerio Personal Firewall*.

### 1.2 Installation

Installation is easy to perform by simply executing the installation archive (typically *kerio-pf-201-en-win.exe*). During installation you may choose the directory where *Kerio Personal Firewall* will be installed, or leave the default setting (C:\Program Files\Kerio\Personal Firewall). The system needs to be restarted after installation in order for the low-level driver to be loaded.

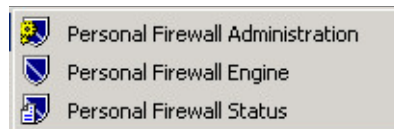
## Chapter 2

# Administration

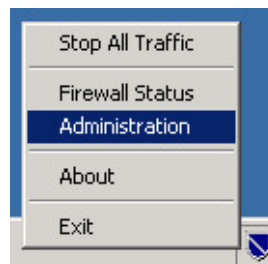
---

### 2.1 Personal Firewall Components

*Kerio Personal Firewall* consists of three programs: *Personal Firewall Engine*, *Personal Firewall Administration* and *Personal Firewall Status Window*.



Personal Firewall Engine is the program that takes care of all *Personal Firewall* functions. It runs as a background application (or as a service in Windows NT/2000) and its presence is represented by an icon in the System Tray.



If you right-click on the icon a menu is displayed, in which you can select from the following options: block all communication (*Stop All Traffic* — if selected, it reverts to *Enable Traffic* and the System Tray icon changes to indicate that traffic is being blocked), run the *Administration* application or view the *Status Window* (*Firewall Status*), program version information (*About*) or stop the *Personal Firewall Engine* (*Exit*). Stopping the *Engine* of course stops all security functions.

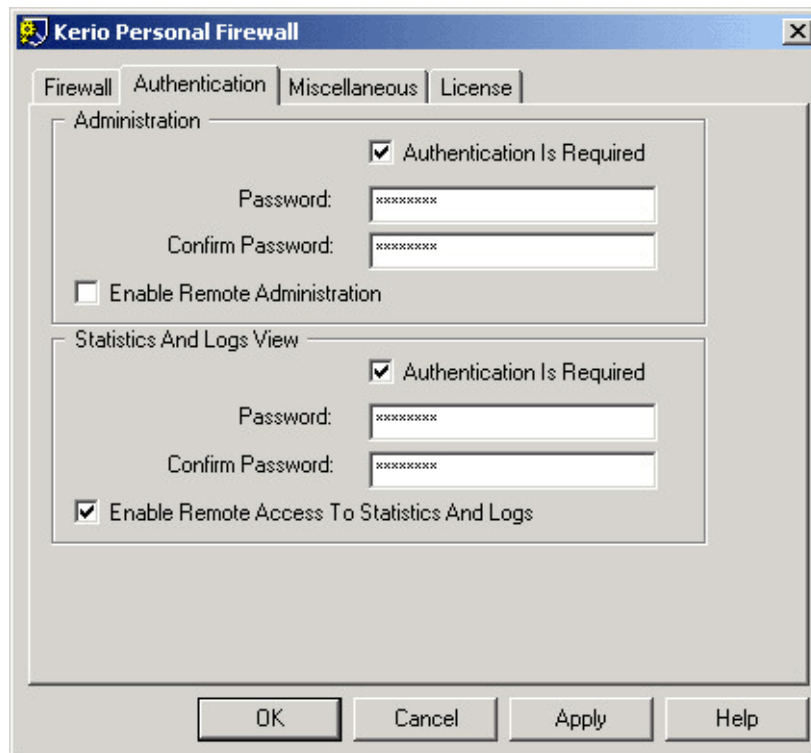
Left-double-clicking on the icon runs *Personal Firewall Status Window* program.

Personal Firewall Administration is the main configuration tool for Personal Firewall Engine. We will deal with the individual settings in the upcoming chapters of this manual.

*Personal Firewall Status Window* displays information about all running applications that communicate via TCP/IP protocol. It is also described in a special chapter.

## 2.2 Securing Access to the Administration

To ensure full security it is vital that Personal Firewall is running any time the computer is on and that only authorized personnel have access to its configuration. This can be set in the Personal Firewall Administration program, in the *Authentication* tab.



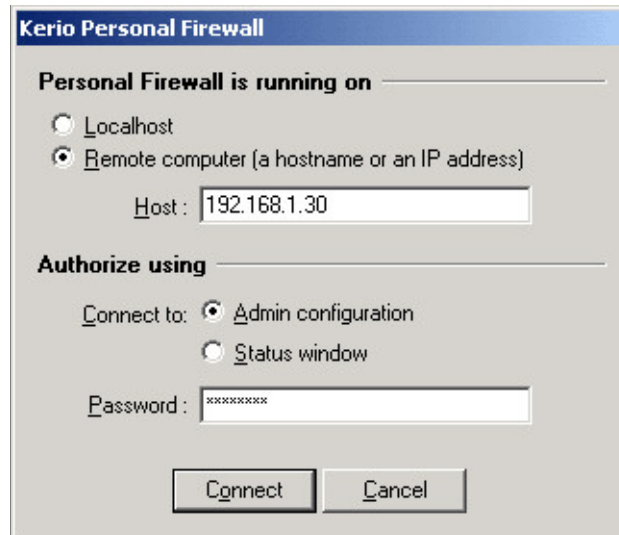
**Administration section** *Authentication Is Required* means that a password will be required upon running the *Personal Firewall Administration* program. After enabling this feature, fields will be available for you to enter the password and password confirmation. *Enable Remote Administration* allows for configuration to be performed from a remote computer.

**Statistics and Logs View section** Settings for access and remote access to logs and statistics are performed here. All fields are the same as in the previous section. Configuring these two sections separately allows for different levels of access rights — either viewing the logs and statistics only or full administration access.

## 2.3 Administration Login

To administer *Kerio Personal Firewall* or view all logs start *Personal Firewall Administration* or *Personal Firewall Status Window* applications respectively. Note that the following

connection dialog will only appear if authentication is required, otherwise you will connect directly to the local KPF engine.



Here you can choose if you want to connect to *Personal Firewall* running on the local computer (*Localhost*) or on a remote computer specified by its DNS name or IP address. You can choose whether you want to run the *Personal Firewall Administration (Admin configuration)* or *Personal Firewall Status Window (Status window)* programs. Enter your password in the corresponding field.

## 2.4 Personal Firewall Status Window

*Personal Firewall Status Window* allows monitoring of all TCP/IP activities within the operating system and displays detailed information about all communicating applications.

### **Main Window**

The main windows displays information about one local end-node in each line (the end-node is defined by its IP address, port and protocol). A local end-node can only correspond with one application. However, one application can have more end-nodes (for instance, an FTP server waits for incoming connections at the ports 20 and 21).

Individual columns then display information about the end-nodes:

**Application** The name of the application's executable that the end-node belongs to. The name can be displayed including its full path by selecting *Settings / Don't Cut Pathnames*.

Application	Protocol	Local Address	Remote Address	State	Creation Time	Rx [Bytes]	Rx
<input type="checkbox"/> INETINFO.EXE	TCP	all:80	-----	Listening	07/Mar/2002 12:06:35	0	
<input type="checkbox"/> INETINFO.EXE	UDP	all:3456	-----	Listening	07/Mar/2002 12:06:35	16	
<input type="checkbox"/> INETINFO.EXE	TCP	all:443	-----	Listening	07/Mar/2002 12:06:35	0	
<input type="checkbox"/> INETINFO.EXE	TCP	all:21	-----	Listening	07/Mar/2002 12:06:35	0	
<input type="checkbox"/> INETINFO.EXE	TCP	all:3296	-----	Listening	07/Mar/2002 12:06:35	0	
<input type="checkbox"/> ISMSERV.EXE	TCP	all:1209	localhost:389	Connected Out	07/Mar/2002 12:06:05	1112	
<input type="checkbox"/> ISMSERV.EXE	UDP	all:1200	-----	Listening	07/Mar/2002 12:06:04	3	
<input type="checkbox"/> ISMSERV.EXE	TCP	all:1201	localhost:389	Connected Out	07/Mar/2002 12:06:04	2402	
<input type="checkbox"/> ISMSERV.EXE	TCP	all:1202	localhost:389	Connected Out	07/Mar/2002 12:06:05	2431	
<input type="checkbox"/> ISMSERV.EXE	TCP	all:1203	-----	Listening	07/Mar/2002 12:06:05	0	
<input type="checkbox"/> LLSSRV.EXE	UDP	all:1657	-----	Listening	07/Mar/2002 12:21:04	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.10.1:88	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.10.1:464	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.1.17:88	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.1.17:464	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.2.40:88	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.2.40:464	-----	Listening	07/Mar/2002 12:04:42	0	
<input type="checkbox"/> LSASS.EXE	UDP	192.168.2.41:88	-----	Listening	07/Mar/2002 12:04:42	0	

TCP Listening: 52    TCP Connected: 19    UDP Listening: 58    Total Rx speed: 5.96    Total Tx speed: 5.96

**Protocol** The communication protocol (either *TCP* — connected protocol or *UDP* — unconnected datagram protocol).

**Local Address** A local IP address and port (displayed in the following format — address:port). In the *Settings* menu you can choose to display the DNS name instead of IP addresses and (standard) service.

**Remote Address** The remote IP address and port is not displayed unless a connection is established.

**State** The state of a local end-node: *Listening* — waiting for incoming connection, *Connected In* — connection established from a remote client to a local service, *Connected Out* — connection established by a local application to a remote server.

**Creation Time** The time when connection was established or when a given application started receiving connection on a given port

**Rx [bytes]** The amount of data received by a given end-node (in bytes)

**Rx speed [bytes/sec]** Average speed of data transfer (in kilobytes per second)

**Tx, Tx speed** the same as previous for outgoing data

### Main Menu

**File** *Connect...* connects to *Personal Firewall Engine* (on a local or remote system). Use *Close* to close the *Personal Firewall Status Window* application.

**Logs** Displays *Firewall Log* window or *Statistics* of transferred and filtered data.

**Settings** Contains detailed settings of what information will be displayed and how:

- *Hide Listening Sockets* — hides end-nodes that have no established connection (their status is *Listening*)
- *Hide Local Connections* — hides connections established within a local system (loopback)
- *Hide Admin-Firewall Connection* — hides connections established between individual *Personal Firewall* components
- *Don't Resolve Domain Names* — IP addresses will not be translated to DNS names
- *Don't Show Port Names* — port numbers will not be translated to services names (e.g. telnet, smtp, http)
- *Displayed Application Name* — switches application name display mode: *Whole Pathname*, *Cut Pathname* (only file name will be displayed) or *File Information* (displays application name if possible, otherwise a condensed filename will be displayed).
- *Update frequency* — changes frequency of information refreshing (*Slowest* — 5 seconds, *Slower* — 2 seconds, *Normal* — 1 second, *Fast* — 0.5 second)

**Help** Help and information about program manufacturer and version.

---

## Security Settings

---

### 3.1 Introduction to TCP/IP

In order to be able to configure *Kerio Personal Firewall* properly and make the most of its functions one needs to understand the principals of TCP/IP communication. Advanced users do not need to read this chapter, it is, however, highly recommended to beginners.

**TCP/IP** TCP/IP is a common name for communication protocols used on the Internet. Data is divided into small parts called packets within each protocol. Each packet consists of a header and a data part. The header contains control information (eg. source and destination address), while the data part carries the information being passed between applications.

The protocol set is further subdivided into several levels. Packets of lower-level protocols contain higher-level packets in their data part (eg. TCP protocol packets are transferred within IP packets).

**IP** IP (Internet Protocol) carries in its data part all other protocol packets. The most important piece of information contained in the header is the source and destination IP address, that is the address of the computer that sent the packet and which computer it is addressed to.

**ICMP** ICMP (Internet Control Message Protocol) transfers control messages. There are several types of these messages, eg. information about availability of a remote computer, routing request or reply request (used in *PING* command).

**TCP** TCP (Transmission Control Protocol) is used for reliable data transfers via a so-called virtual channel (connection). It is used as a carrier protocol for most application protocols, eg. SMTP, POP3, HTTP, FTP, Telnet, etc.

**UDP** UDP (User Datagram Protocol) is a so-called non-connected protocol, that means it does not create a channel — all data is transferred via individual messages (called datagrams). UDP does not ensure safe and reliable data delivery as datagrams can be lost during the transfer. In comparison to the TCP protocol, UDP is far less demanding on resources (there is no establishing and ending of connections, acknowledging of data, etc.). UDP protocol is used for DNS requests, transfers of sound or video data, etc.

**Ports** The most important piece of information in the TCP and UDP packet header is the source and destination port. While an IP address defines a computer on the Internet, a port defines an application running at this computer. Ports 1–1023 are reserved for standard services, ports 1024–65535 can be used by any application. During a typical client-server communication the destination port is known (a connection is established to this port or a UDP packet is sent there), the source port is usually assigned automatically by the operating system.

**Application protocols** Protocols which are carried within TCP/UDP packets are used for user (application) data transfer. There exist many standard application protocols (eg. SMTP, POP3, HTTP, FTP, etc.). A programmer can, however, design his own (non-standard) means of communication.

### 3.2 How does Kerio Personal Firewall work?

All communication on the Internet is carried out using TCP/IP protocol set. These protocols are usually also used for communication within local networks. The main (carrier) protocol is IP (Internet Protocol), whose packets carry all other information (they enclose other protocols). A true firewall must have complete control over all IP packets — that is, it must be able to catch them, find all necessary information within them and then let them pass or filter them. And, of course, it must be able to keep record of all performed actions, detected attacks, etc.

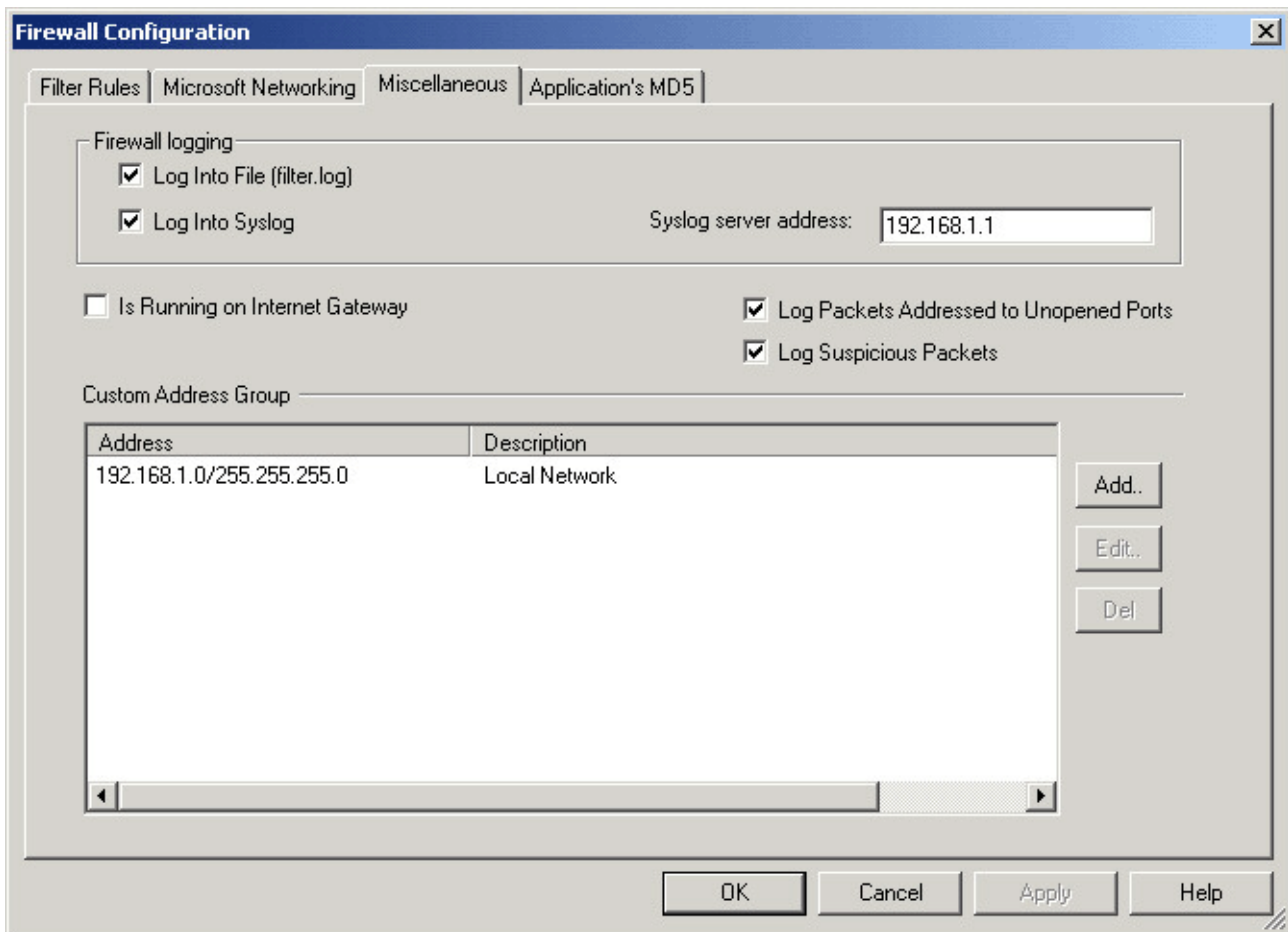
The main principle behind a firewall such as KPF is stateful inspection. This means that a record is made on every packet going from your computer and only a packet corresponding with this record is let pass back through. All other packets are dropped. This ensures that *Personal Firewall* only allows communication initiated from within the local network.

The user / administrator can further specify conditions for packet filtering in filtering rules. Only packets complying with given criteria are accepted.

### 3.3 IP Address Groups

When defining filtering rules permitting or denying certain communication a situation can arise that the same rule needs to apply for a group of IP addresses (eg. several computers within a local network).

*Kerio Personal Firewall* allows the user to define a group of IP addresses that can be easily used in filtering rules definition. A group can contain any number of IP addresses, IP address ranges or subnets.



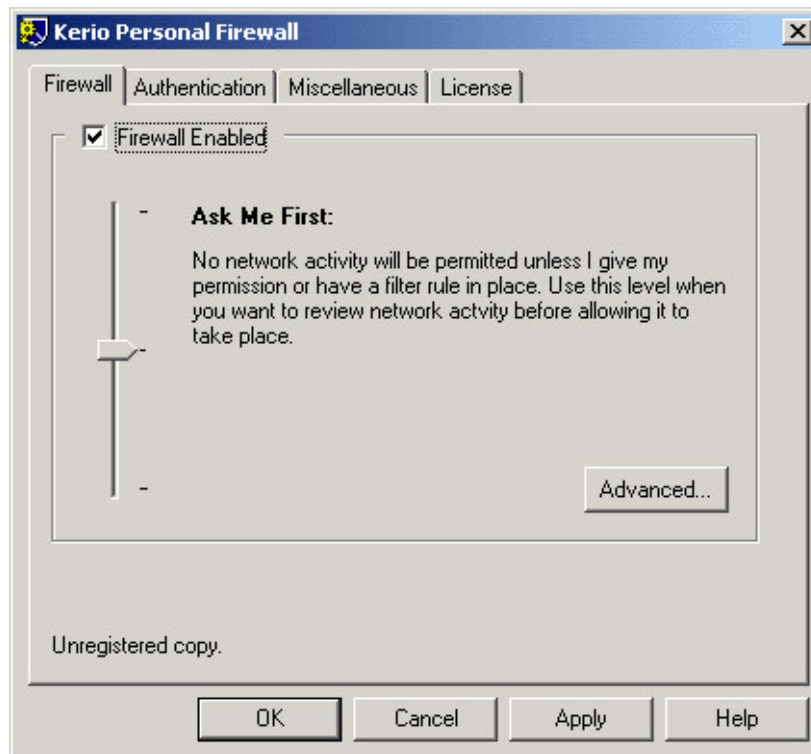
The custom address group can be defined in the *Firewall Configuration* window, *Miscellaneous* tab. Pressing *Add...* you can add a *Single IP address*, an address range (*Network / Range*) or a sub-network (*Network / Mask*). *Edit...* and *Del* buttons allow you to edit or remove individual items respectively.

### 3.4 Levels of Security

*Kerio Personal Firewall* allows for 3 basic security levels:

**Permit Unknown** Minimum security. *Personal Firewall* permits any communication, unless explicitly denied by filtering rules. *Personal Firewall* is fully transparent if there are no filtering rules set (it behaves as if it were not running at all).

**Ask Me First** All communication is denied implicitly at this level. If an application tries to communicate or somebody wants to establish a connection from outside, *Personal Firewall* stops the request and displays a dialog window asking whether you want to



permit or deny such communication. This can be allowed for once only or permanently (recommended).

**Deny Unknown** All communication is denied which is not explicitly permitted by the existing filter rules. *Personal Firewall* doesn't ask the user for anything.

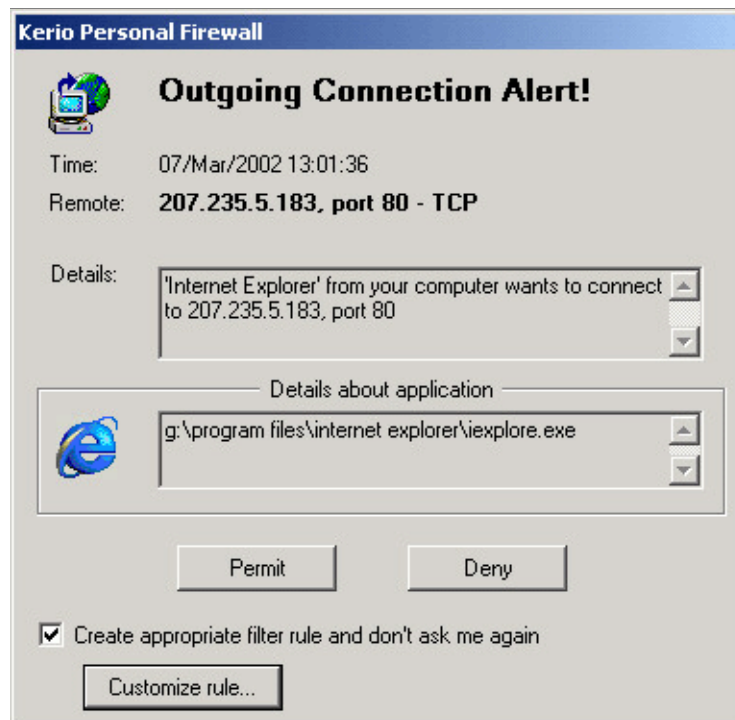
### 3.5 Interaction with the User

If *Ask Me First* security level is set *Personal Firewall* automatically permits only communication that is allowed by filtering rules. If a packet is caught that does not comply with any rule, it is assumed that the user started a new application not used before and a dialog window is displayed where the user can permit or deny such communication. Permission or denial can be either temporary or permanent (by creating an appropriate rule).

The dialog window displays the following information:

**Incoming / Outgoing Connection Alert!** Indicates whether the requested connection is outgoing (from the local network) or incoming (eg. from the Internet)

**Time** Exact time and date, when the connection was requested



**Remote** Information about the remote end-node (IP address, port and communication protocol)

**Details** Detailed information about the connection

**Details about application** Information about the local application taking part in the communication (as a client or server)

**Permit** Let the communication pass through

**Deny** Stop (filter) the communication

**Create appropriate filter rule...** If this option is selected, by pressing *Permit* or *Deny* a filtering rule is automatically created, which causes the next packet of the same type to be either permitted or denied access. This can be used in the initial configuration of *Personal Firewall* — the user does not need to define any rules, but as they run their favorite applications, rules can be created for them in this way.

**Customize rules** Here an advanced user can edit and customize the automatically created filter rule.

A filtering rule created in this way is always valid for a particular application that sent a packet or to which a packet was sent (see the *Details about application* field). An

MD5 signature is also automatically created so that subsequent executions of that same application name will be compared against the initial signature. This would prevent a trojan from spoofing its name to a trusted application such as outlook.exe. Details about MD5 signatures can be found in chapter 3.8.

By default a filtering rule for a particular application is created so that such an application can communicate at any local port with any computer on the Internet (any remote address) and also to any remote port. It is assumed that if the user permits communication for the application once, it is reliable and its communication will not be limited in the future. However, this is not always true and therefore the user can adjust the rule to their needs. Automatically created rules can always be customized or removed later from the *Filter Rules* in the *Advanced* option.

### *Customizing an automatically created rule*

The actual setting of a filtering rule is dependent on a particular situation, especially the application for which communication is permitted or denied. These are a few general principles:

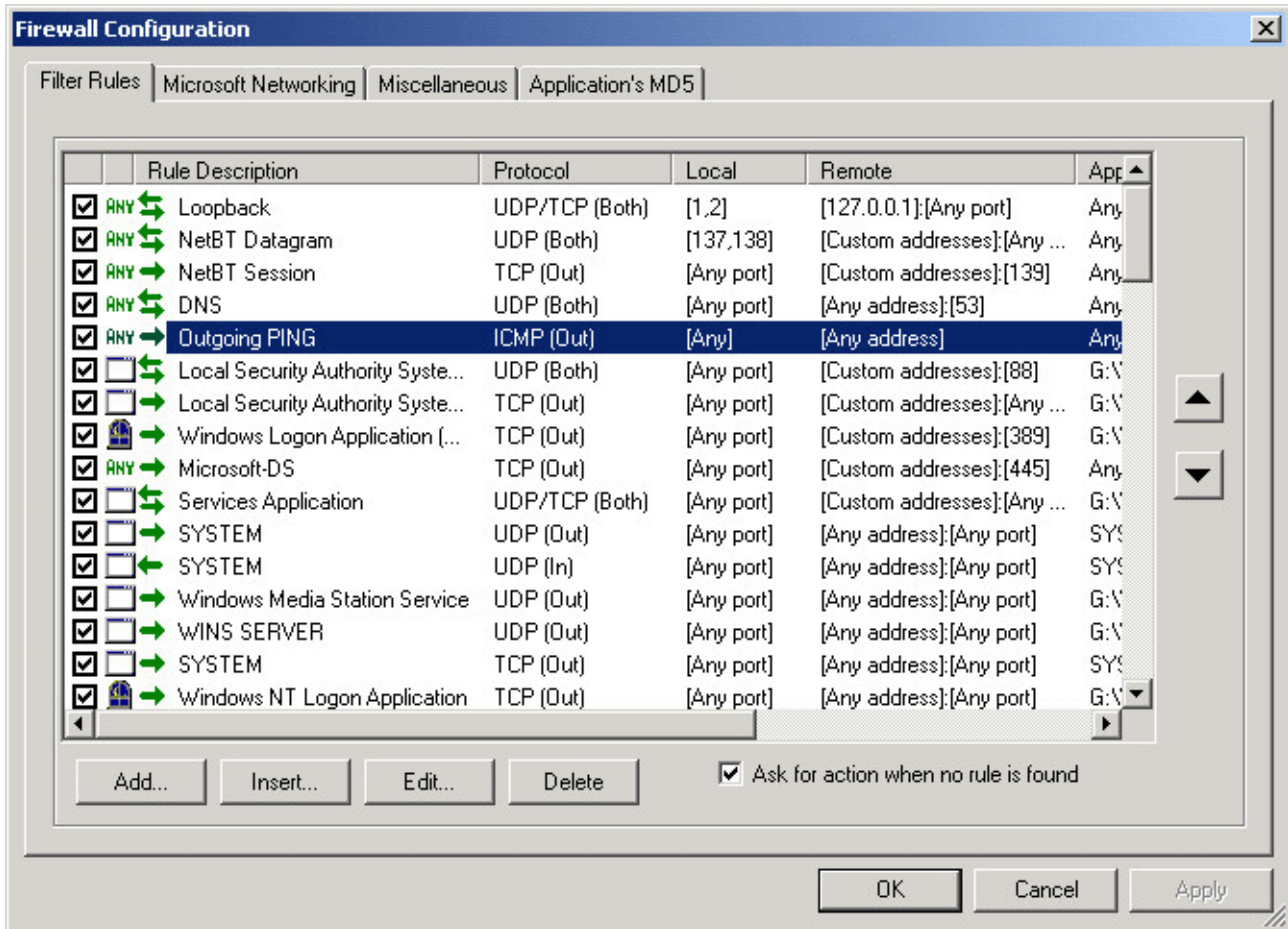
- Only experienced users, who are familiar with TCP/IP communication should alter the filtering rules settings.
- Setting the local port for an application is not recommended because the local port is assigned by the operating system and is not known in advance in most cases.
- The same applies for remote ports, if we deal with a server-type application (eg. a WWW server).

## 3.6 Packet Filtering Rules

Filtering rules define which packets should be allowed or denied communication. Without these rules *Kerio Personal Firewall* would only work in two modes: all communication allowed or all communication denied.

There exist two ways of creating the filtering rules: either automatically when detecting an unknown packet (the user must either permit or deny such a packet — see chapter 3.5) or manually in the *Personal Firewall Administration* program. Here the user can not only create rules, but also edit them, remove them or put them into order according to their priority.

Defined filtering rules are displayed in the *Filter Rules* tab (after pressing the *Advanced* button in the *Personal Firewall Administration* main window, *Firewall* tab).



#### List of filtering rules

The filtering rules are displayed in a table, in which each line represents one rule. Individual columns have the following meaning:

- **Checkbox** — indicates whether the rule is active or not. By a single click the user can activate or deactivate the rule without the need of removing or adding it.
- **Application icon** — displays the icon of the local application, to which the rule applies. If the rule is valid for all applications a special green icon saying *ANY* is displayed instead. Only in rare situations should such a rule exist.
- **Rule Description** — the direction and description of a rule. The following symbols are used for direction: right arrow (outgoing packet), left arrow (incoming packet), double (both-direction) arrow (the rule applies for both outgoing and incoming packets). The rule's description can contain anything the user wishes. For an automatically created rule the name of the application is used for its description.

- *Protocol* — used communication protocol (TCP, UDP, ICMP...). The direction of the communication (*In*, *Out* or *Both*) is also displayed in brackets following the name of the protocol.
- *Local* — local port
- *Remote* — remote IP address and port (separated by a colon)
- *Application* — the local application's executable including the full path. If the application is an operating system service, the name displayed will be *SYSTEM*.

### Controls

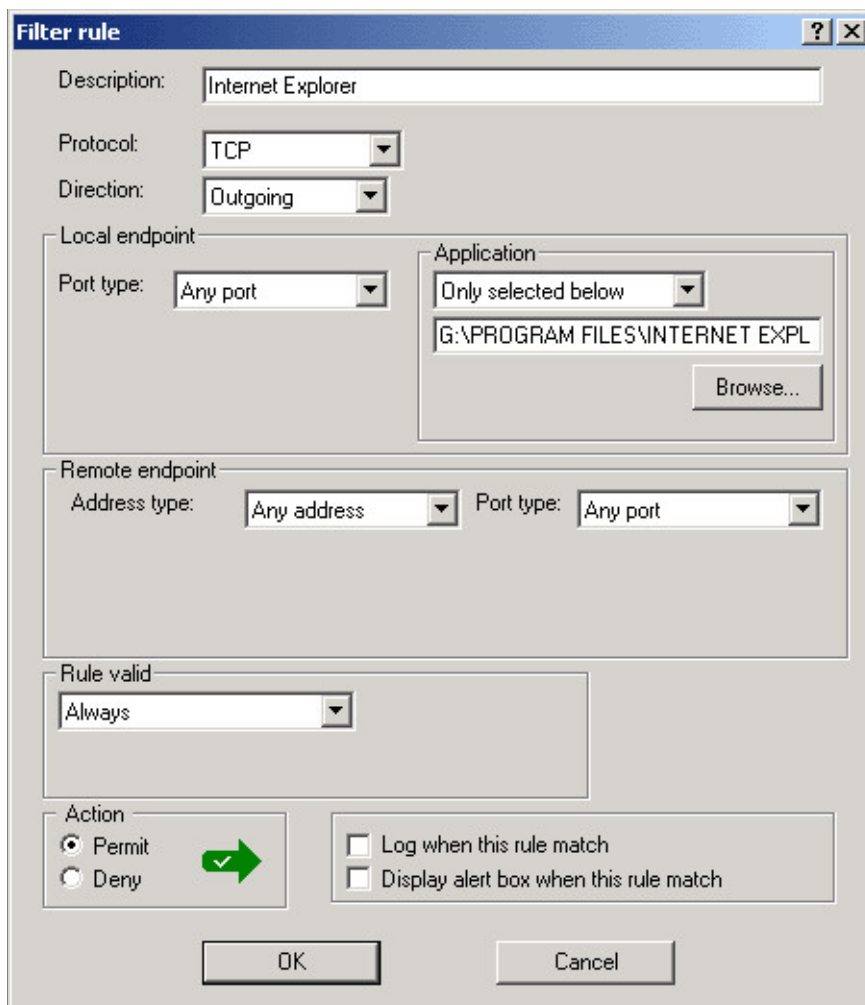
- *Add* — adds a new rule at the end of the list
- *Insert* — inserts a new rule above the selected rule. This function spares the user of moving the new rule within the list, as it allows for inserting a new rule to any desired place.
- *Edit* — edits the selected rule
- *Delete* — removes the selected rule
- Arrow buttons (to the right of the list of rules) — these enable placement of a selected rule within the list. Note that filters work from top down so the placement of a rule is very important.

### Adding or editing the rule

After pressing the *Add*, *Insert* or *Edit* button a dialog is displayed for defining a filtering rule.

#### General options

- *Description* — a rule can be described by any text string. We recommend that the user describe all rules based on what they are intended for (e.g. dns resolution, incoming ping request...). This option is only necessary for advanced users.
- *Protocol* — the communication protocol that the rule applies to. *TCP*, *UDP*, *TCP and UDP*, *ICMP* or *OTHER* (choose *Other* and then define the protocol by the number in the IP packet header). A special option *Any* means any protocols, e.g. all IP packets.



- If *ICMP* protocol is chosen, a new *Set ICMP...* button appears. After pressing this button the user can choose the type of ICMP messages, which the rule will apply to. The chosen ICMP types are then displayed in a special text field.
- *Direction* — a direction in which the packets should be filtered (*Outgoing*, *Incoming* or *Both*)

#### Local endpoint section

- *Port type* — the port (only if TCP and/or UDP is chosen). Possible options are: Any (any port), Single Port, Port Range or List of ports (a list of port numbers, separated by commas).
- *Application* — indicates if the rule applies to all packets (*Any application*) or incoming/outgoing packets of a particular application (*Only selected below*). The

application's executable should be entered including its full path. This can be done either manually or using the *Browse* button.

### Remote endpoint section

- *Address type* — IP address of the remote computer. This can be specified as *Any address*, *Single address* (a particular computer's address), *Network/Mask*, *Network/Range* or user defined group of IP addresses (*Custom Address Group*).
- *Port type* — remote port. The options are the same as in the definition of a local port.

### Other parameters

- *Rule valid* — indicates if the rule is *Always valid* or if it is valid only at a certain time (*In this time interval only*). In the second case the user can set the time interval by pressing the *Set...* button.

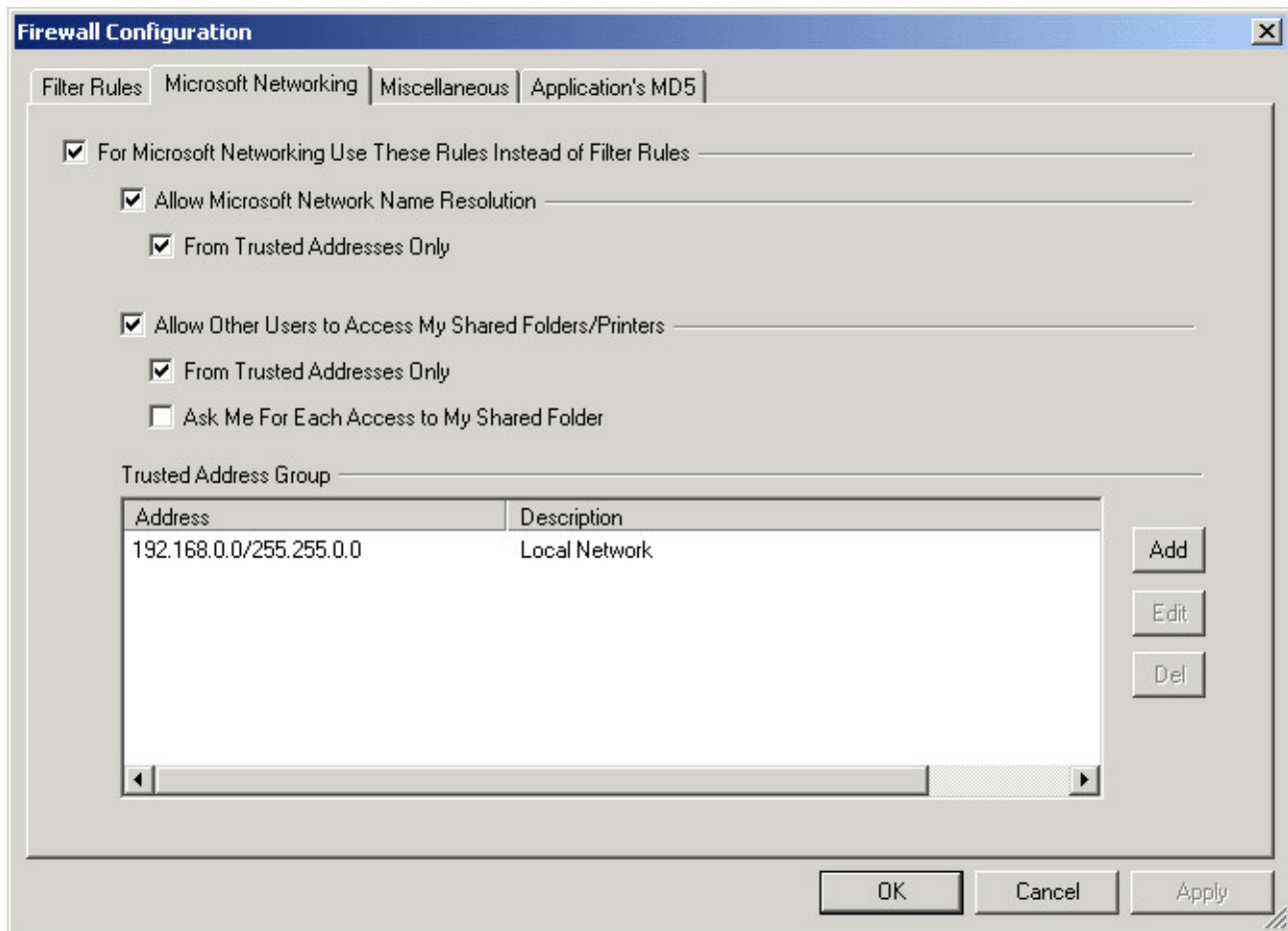
The usage of time intervals requires a correct setting of the system time!

- *Action* — the action that is to be performed — if a packet is to be *Permitted* or *Denied* communication
- *Log when this rule matches* — a packet will be logged if it complies with this rule (see logging options in the *Firewall Configuration* window, *Miscellaneous* tab)
- *Display alert box when this rule matches* — if a packet complies with this rule an information window (*Firewall Rule Alert*) will be displayed containing a detailed description of the packet and whether the packet was permitted or denied.

## 3.7 Microsoft Networking

Very common is the case where a computer running Microsoft Windows is connected to a local network running File and Printer sharing for Microsoft Networks. Several services are used for communication in this environment and setting a personal firewall for optimum performance and security under such a condition is not always easy.

*Kerio Personal Firewall* allows separate rules for a Microsoft Network environment. These settings are available in the Advanced section in the *Microsoft Networking* tab.



**For Microsoft Networking Use These Rules...** This option means that the special rules defined in this dialog will be used.

**Allow Microsoft Network Name Resolution** Enabling this option will allow the exchange of Windows network computer names.

**From Trusted Addresses Only** The protocol used for Windows name resolution will only be available to the defined address group. This group is automatically generated by gathering TCP/IP information from the local system and may be modified if necessary.

**Allow Other Users to Access My Shared...** Allows access to shared directories and printers.

**From Trusted Addresses Only** Access is only allowed from defined trusted addresses

**Ask Me For Each Access...** At every attempt of connection to a shared directory, *Personal Firewall* will ask if such a connection should be permitted or denied.

**Trusted Address Group** A group of IP addresses defined as trusted. Using the *Add*, *Edit* and *Del* buttons the user can add, change or remove an IP address, a range of IP addresses or a whole subnetwork. Validity for this IP address group is limited to the *Microsoft Networking* tab. The group cannot be used for defining other rules.

### *Examples of optimum settings*

- If you have a stand-alone computer that is not connected to a local network (e.g. a notebook connected to the Internet via a modem), only enable the option *For Microsoft Networking Use These Rules Instead Of Filter Rules*. Leave all other options off. This will disable all communication for Microsoft Networks as it is not relevant to this scenario.
- If your computer is connected to a local network where you trust your colleagues and you want them to be able to access all your shared directories and printers, enable all options except *Ask Me For Each Access to My Shared Folders*. In the field *Trusted Address Group* define your local network (e.g. as a sub-network with a corresponding mask or as a range of IP addresses).
- If you would like to grant access to your shared material but still have complete control over who can access them, do as in the previous example but also enable the option *Ask Me For Each Access to My Shared Folders*.

## 3.8 Application MD5 Signatures

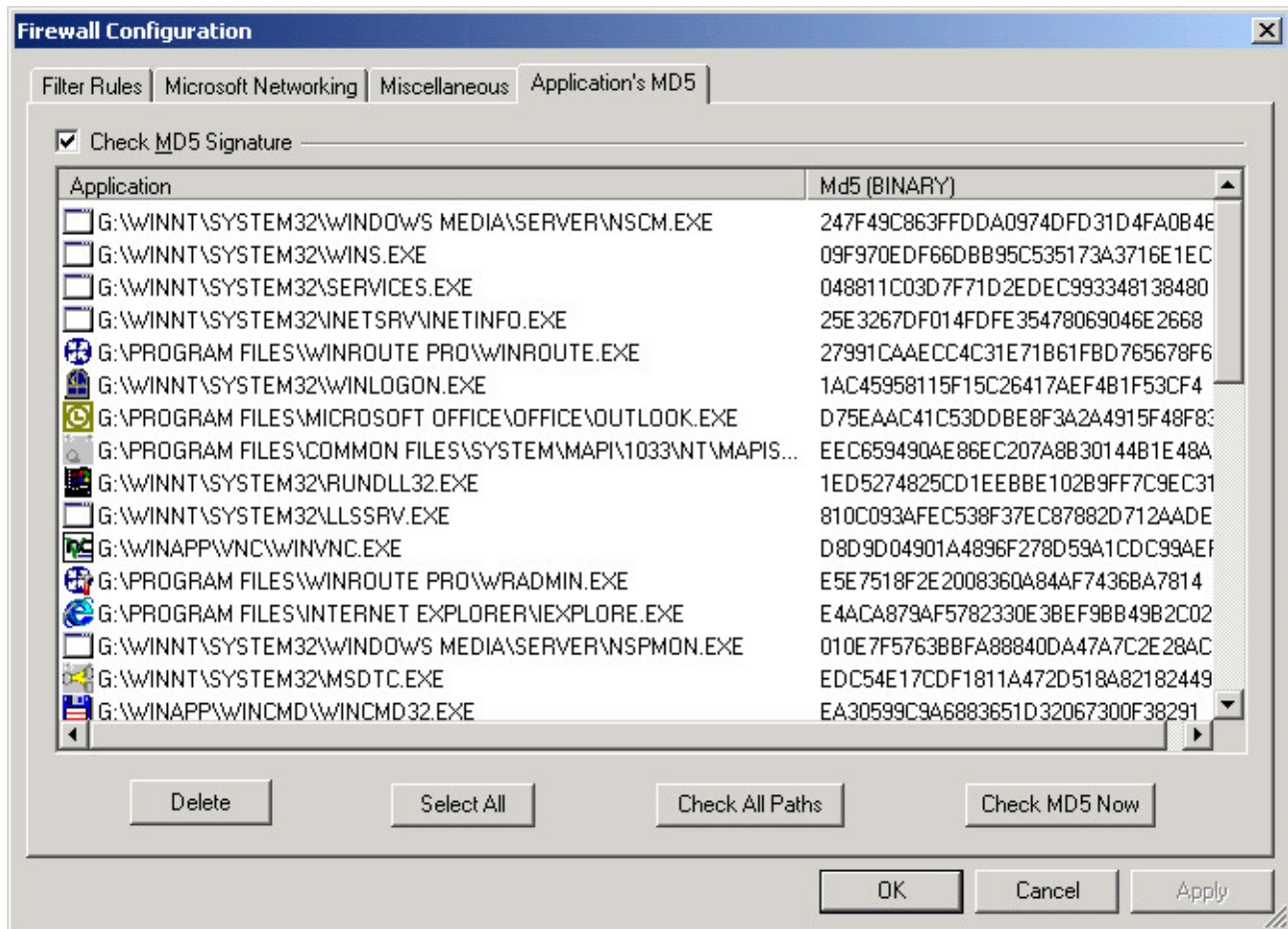
Apart from checking incoming and outgoing packets *Kerio Personal Firewall* can also detect if permitted packets are sent by authorized applications. An application could be infiltrated into your computer (e.g. by email, from a floppy disk, etc.), that acts as some regular and known program (it usually replaced by the original executable) and tries to send data out from your computer. Such an application is commonly referred to as a “Trojan horse”. Usually, they can be revealed during an anti-virus check, but this might be too late.

*Kerio Personal Firewall* uses a method of creating and checking MD5 signatures of applications. Very simply said, an MD5 signature is a checksum of the application’s executable. When the application is first run (or when the application first tries to communicate via the network) *Personal Firewall* displays a dialog, in which a user can permit or deny such communication. If the communication is permitted by the user *Personal Firewall* creates an MD5 signature for the application. This signature is checked during each subsequent attempt of the application to communicate over the network. If the application’s executable is changed (e.g. it is infected by a virus or it is replaced by another program) *Personal Firewall* denies communication for this application, displays a

### 3.8 Application MD5 Signatures

warning and asks if such a change should be accepted (e.g. in case of the application upgrade) or not.

MD5 signatures can be viewed and deleted in the *Application's MD5* tab. They can only be created automatically.



The following options are available in the *Application's MD5* tab:

**Check MD5 signature** This option enables/disables creating and checking of an applications' MD5 signature.

**Delete** Removes the MD5 signature of a selected application(s).

**Select All** Selects all applications in the list.

**Check All Paths** Checks all applications for existing executables. If the executable is not present (e.g. after application uninstall), the user is asked whether the MD5 signature should be removed for this application.

**Check MD5 Now** Checks for the validity of MD5 signatures. If a signature is not valid *Personal Firewall* asks if a change should be accepted or not.

*Note:* Several applications can be selected using the *Ctrl* or *Shift* key.

### 3.9 Internet Gateway Protection

*Kerio Personal Firewall* can also be used for protecting an Internet gateway, i.e. a computer that provides access to the Internet for computers in a local network (a router or a NAT router). Typically, this can be done in combination with Microsoft's *Internet Connection Sharing* (ICS) application, a component of Windows 98 SE, Me, 2000 and XP operating systems. ICS enables access to the Internet for all local computers via a single IP address. However, it does not provide any protection from external attacks. In combination with *Kerio Personal Firewall* you can have a secure shared Internet connection.

*Personal Firewall* is designed for protecting a single computer. However, a great amount of packets pass the internet gateway (router) that are not addressed to this computer. In order not to be forced to define complex packet filters, *Personal Firewall* can be switched to a special mode designed for Internet gateways. This can be done in the *Firewall Configuration* window (after pressing the *Advanced* button) on the *Miscellaneous* tab by enabling the option *Is running on Internet gateway*.

*Note:* Do not enable this option if *Personal Firewall* does not really run on an real Internet gateway as the security level of your computer will be downgraded.

## Chapter 4

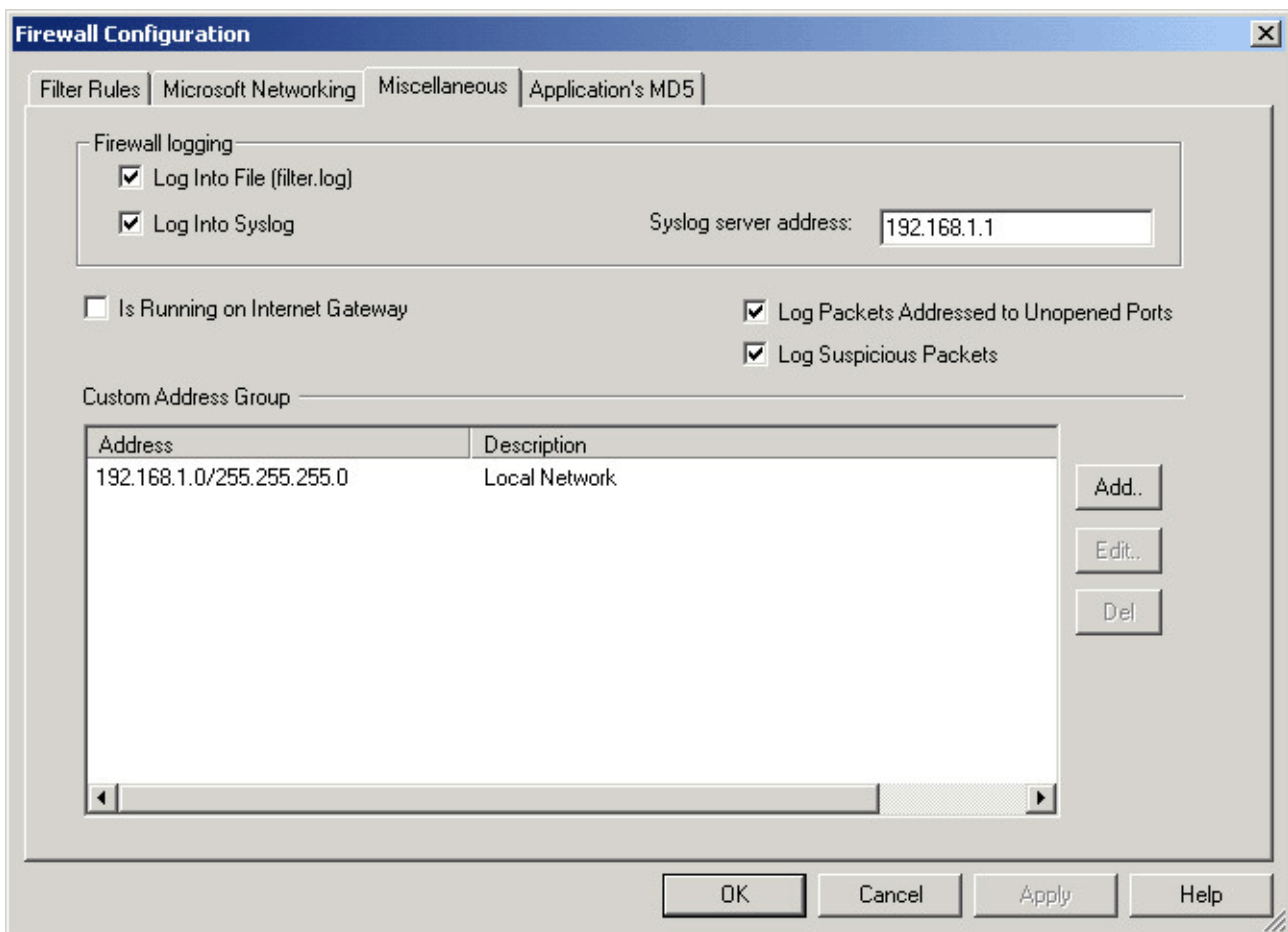
# Firewall Logging

---

### 4.1 Logging Configuration

*Kerio Personal Firewall* allows creation of detailed log files for passing or filtered packets. The user (or the administrator) has a range of options for setting what information and where it will be logged. The logs can either be saved to a file (with the name `filter.log` saved to a directory where *Personal Firewall* is installed — typically `C:\Program Files\Kerio\Personal Firewall`) or sent to a *Syslog* server.

Basic log setting are performed in the *Firewall Configuration* window, *Miscellaneous* tab in the *Firewall Logging* section.



**Log Into File (filter.log)** Logs will be saved into the filter.log file (in a directory where *Personal Firewall* is installed). The size of this file is limited only by the available space on the disk.

**Log Into Syslog** Logs will be sent to a *Syslog* server running at a specified IP address

**Log Packets Addressed to Unopened Ports** Logs packets addressed to ports from which no application is running (typically a “portscanning” type attack).

**Log Suspicious Packets** Log packets that *Kerio Personal Firewall* considers as suspicious. These are e.g. TCP packets that do not pertain to any open connection and do not initiate a new connection (the so-called “TCP PING”).

### 4.2 Filter.log file

The filter.log file is used for logging *Kerio Personal Firewall* actions on a local computer. It is created in a directory where *Personal Firewall* is installed (typically C:\Program Files\Kerio\Personal Firewall). It is created upon the first record.

Filter.log is a text file where each record is placed on a new line. It has the following format:

```
1,[08/Jun/2001 16:52:09] Rule 'Internet Information Services':  
Blocked: In TCP, richard.kerio.cz [192.168.2.38:3772]->localhost:25,  
Owner: G:\WINNT\SYSTEM32\INETSrv\INETINFO.EXE
```

How to read this line:

- 1 — rule type (1 = denying, 2 = permitting)
- [08/Jun/2001 16:52:09] — date and time that the packet was detected (we recommend checking the correct setting of the system time on your computer)
- Rule 'Internet Information Services' — name of a rule that was applied (from the *Description* field)
- Blocked: / Permitted: — indicates whether the packet was blocked or permitted (corresponds with the number at the beginning of the line)
- In / Out — indicates an incoming or outgoing packet
- IP / TCP / UDP / ICMP, etc. — communication protocol (for which the rule was defined)

- `richard.kerio.com [192.168.2.38:3772]` — DNS name of the computer, from which the packet was sent, in square brackets is the IP address with the source port after a colon
- `localhost:25` — destination IP address (or DNS name) and port (localhost = this computer)
- `Owner:` — name of the local application to which the packet is addressed (including its full path). If the application is a system service the name displayed is `SYSTEM`.



Chapter 5

## **Index**

---

