ViPNet Coordinator HW/VA 3.3

Reference Guide

ViPNet®
Virtual Private Network

infotecs®

Version: 00079-06 90 01 ENU

Global contacts page http://www.vipnet.com/index_all.php

Contents

Introduction

About This Document

This document is a supplement to the main document “ViPNet Coordinator HW/VA. Administrator's Guide” and contains a description and examples of commands that are needed to be given via the command line interface for configuring ViPNet Coordinator HW/VA settings. In order to use this document effectively, you should be familiar with network technologies and the ViPNet technology in particular. For more information on ViPNet networks, see the documents “ViPNet VPN. User's Guide” and “ViPNet Technology. General Information.”

Audience

This document is intended for ViPNet VPN network administrators performing ViPNet Coordinator HW/VA setup.

Document Conventions

This document concerns the following conventions:

Table 1: Document conventions

Icon	Description
⚠	**Warning:** Indicates an obligatory action or information which may be critical for continuing user operations.
ℹ	**Note:** Indicates a non-obligatory, but desirable action or information which may be helpful for users.
💡	**Tip:** Contains additional information.

Table 2: Conventions for highlighted information

Icon	Description
Name	The name of an interface element. For instance, the name of a window, a box, a button or a key.
Key+Key	Shortcut keys. To use the shortcut keys, press and hold the first key and press other keys.
Menu > Submenu > Command	A hierarchical sequence of elements. For instance, menu items or sections in the navigation pane.
`Code`	A file name, path, text file (code) fragment or a command executed from the command line.

The following conventions are used in this document for commands' description:

- The commands you can execute only in the administrator mode are rubricated. For example:

  ```
  command
  ```

- The parameters, which should be specified by a user, are enclosed in angle brackets. For example:

  ```
  command <parameter>
  ```

- Optional parameters are enclosed in square brackets. For example:

  ```
  command <mandatory parameter> [optional parameter]
  ```

- If you can specify one of several parameters when typing a command, the available variants are enclosed in curly brackets and divided with a vertical bar. For example:

  ```
  command {variant-1 | variant-2}
  ```

Feedback

Finding Additional Information

For more information about Infotecs products and technologies, see the following resources:

- ViPNet documentation web portal http://www.infotecs.us/doc_vipnet/ENU/index.htm.
- Information about current Infotecs products http://infotecs.us/products/.
- Information about Infotecs solutions http://infotecs.us/solutions/.
- Frequently asked questions http://www.infotecs.us/doc_vipnet/ENU/index.htm#3_17014.htm.

Contacting Infotecs

We value any feedback from you. If you have any questions concerning Infotecs products and solutions, any suggestions, complains or other feedback, feel free to contact us by means of the following:

- Global contacts page http://www.vipnet.com/index_all.php
- Telephone (Germany): +49 (0) 30 206 43 66 0
- Telephone (USA): +1 (646) 589-8571

Errata

Infotecs makes every effort to ensure that there are no errors or misprints in the text of all documents supplied with ViPNet software. However, no one is perfect, and mistakes do occur. If you find an error in one of our documents, like a spelling mistake or some inaccuracy in describing user scenarios or system features, we would be very grateful for your feedback. By sending in errata you may save other reader hours of frustration, and at the same time you will be helping us provide documentation in even higher quality.

Configuring ViPNet Coordinator HW/VA Basic Settings

Configuring System Settings

Configuring Date and Time

For the ViPNet Coordinator HW/VA appliance to communicate with other ViPNet hosts correctly, you should configure the system date and time as described below.

⚠ **Warning:** If the system date and time have been specified incorrectly, encrypted connections with other ViPNet hosts may be blocked.

To configure system date and time, do the following:

1 To view the current time, execute this command:

```
machine show date
```

2 To change your time zone:

- Execute the `machine set timezone` command.
- When prompted to select a continent from the list (`"Please select a continent or ocean"`), type the number of the continent and press **Enter**.
- When prompted to select a country from the list (`"Please select a country"`), type the number of the country and press **Enter**.
- When prompted to select a time zone of the country (`"Please select one of the following time zone regions"`), type the required number and press **Enter**.
- The time zone will be defined according to the selected location and displayed on the screen. To accept the set location, type "`1`" ("Yes"). To define another location, type "`2`" ("No"). Then press **Enter**.
- After you define the time zone, the current system time will be displayed on the screen.
  - To accept the displayed time, press **Enter**.
  - To change the system time, type the date and time in the format YYYY-MM-DD hh:mm:ss, then press **Enter**.

3 If necessary, change the system time by executing the following command:

```
machine set date <date>
```

The date and time are specified in the MMDDhhmm[YYYY] format.

Configuring Swap File Settings

Note: ViPNet Coordinator HW modifications HW 100 X1/X4/X5/X6 do not support expanding virtual memory by swapping.

To configure the swap file:

- To specify the maximum swap file size, execute this command:

  ```
  machine swap set <size in megabytes>
  ```

  If you specify the swap file size that exceeds the available free disk space, the corresponding message will be displayed.

  Warning: After you specify the maximum swap file size, minimum 256 MB of free disk space should remain.

- To enable swapping, execute this command:

  ```
  machine swap mode on
  ```

- To view information about memory and swap file usage, execute this command:

  ```
  machine show memory
  ```

- To disable swapping, execute the following command:

  ```
  machine swap mode off
  ```

  After you execute this command, the swap file will be deleted.

Configuring Event Log Settings

To work with the event log, use the following commands:

- To specify a host to store the event log on or to disable event logging, execute the command:

  ```
  machine set loghost {local | <IP address> | null}
  ```

  To specify the host, choose one of the following values:

  - `local`, ViPNet Coordinator HW/VA's local disk;

- `IP address`, the IP address of the host to which the system events information will be sent;

  > **Note:** If this is not a protected host, configure a firewall rule (see Configuring Unencrypted Traffic Filtering Rules on page 55) that will allow the traffic directed to this host over UDP to port 514.

- `null`, don't log events.

- If the event log is stored on a local disk, to view the log, execute this command:

```
machine show logs
```

- You can export the event log stored on a local disk to a removable USB drive with the FAT32 or ext2 file system. To do this, connect the USB drive to the appliance and execute this command:

```
admin export logs usb
```

- To delete the event log stored on a local disk, execute this command:

```
admin remove logs
```

Connecting ViPNet Coordinator HW/VA to a Network

Connecting to an Ethernet Network

The Ethernet interfaces installed in the operating system take the names of `eth0`, `eth1`, and so on (according to the number of the interfaces in the system). To configure connection to an Ethernet network, specify the network adapters' parameters. To do this, switch to the administrator mode by using the `enable` command and do the following:

- To enable or disable a network interface, execute the following command:

  ```
  inet ifconfig eth0 {up | down}
  ```

Note: Hereinafter, specify the required network interface's name instead of `eth0`.

- To set a dynamic IP address on a network interface, execute the following command:

  ```
  inet ifconfig eth0 dhcp
  ```

- To configure a static IP address on a network interface, execute the following command:

  ```
  inet ifconfig eth0 address <IP address> netmask <network mask>
  ```

  If the interface had a dynamic IP address, then, after you set a static IP address, the data on the DNS and NTP servers received over the DHCP protocol will be lost.

- To set an alias for a network interface, execute the following command:

  ```
  inet ifconfig eth0 address add <IP address> netmask <network mask>
  ```

  If you are setting an alias, it does not matter whether the first IP address is static or dynamic.

- To delete an alias for a network interface, execute the following command:

  ```
  inet ifconfig eth0 address delete <IP address> netmask <network mask>
  ```

- To discard all the settings on an interface, execute the following command:

  ```
  inet ifconfig eth0 reset
  ```

After you execute this command, all settings of the interface will be discarded, and the interface will be disabled (its state will be set to `down`).

For discarding settings on all appliance's interfaces, execute the following command:

```
inet ifconfig all reset
```

⚠ **Warning:** If you are configuring the coordinator in a remote SSH session, use these commands cautiously. If you discard the settings of the interface via which you are accessing the coordinator, you will get disconnected from the coordinator.

- To set the default route, execute the following command:

  ```
  inet route add default gw <gateway IP address>
  ```

- To specify a static route, execute the following command:

  ```
  inet route add <destination IP address> gw <gateway IP address> [netmask
  <network mask>]
  ```

- To delete a static route, execute the following command:

  ```
  inet route delete <destination IP address> [netmask <network mask>]
  ```

- To add or delete a DNS server's address, execute the following command:

  ```
  inet dns {add | delete} <IP address>
  ```

  If DNS servers' addresses have been allocated by a DHCP server, you can't add or remove DNS servers manually.

- To add or delete an NTP server's address, execute the following command:

  ```
  inet ntp {add | delete} <IP address | DNS name>
  ```

  If NTP servers' addresses have been allocated by a DHCP server, you can't add or remove NTP servers manually.

Here is an example of the commands' usage:

```
inet ifconfig eth0 address 10.0.8.79 netmask 255.255.255.0
inet route add default gw 10.0.8.1
inet dns add 10.0.2.3
```

Connecting to a Wi-Fi Network

If the appliance has a Wi-Fi adapter (displayed as a network interface named `wlan0`), you can connect to a wireless network.

Note: A ViPNet Coordinator HW/VA host may also function as an access point (see Configuring a Wi-Fi Access Point on page 25). Functioning as a client and an access point at the same time is not supported in ViPNet Coordinator HW/VA.

To configure connecting ViPNet Coordinator HW/VA to a Wi-Fi network as a client:

1 Switch the `wlan0` network interface to the client mode by executing the following command:

```
inet wifi role client
```

2 To view a list of accessible Wi-Fi networks, execute the following command:

```
inet wifi scan
```

3 Specify the name of the Wi-Fi network you want to connect to and the logon mode. To do this, use one of the following commands:

   - If no authentication in the network is required, execute the following command:

     ```
     inet wifi client ssid <network name> authentication open
     ```

   - If you use the WPA-PSK or WPA2-PSK authentication mode, you should enter your password. To do this, execute the following command:

     ```
     inet wifi client ssid <network name> authentication {wpa-psk | wpa2-
     psk} passphrase <password>
     ```

     To learn your password, contact the administrator of the Wi-Fi network you are connecting to.

4 Enable the `wlan0` network interface by executing the following command:

```
inet wifi mode on
```

5 To verify Wi-Fi connection parameters, execute the following command

```
inet show wifi
```

Here is an example of the commands' usage:

```
inet wifi ssid mynetwork authentication wpa-psk passphrase qwerty
inet wifi mode on
```

Connecting to a 3G/LTE Mobile Network

ViPNet Coordinator HW/VA can connect to the Internet over 3G/LTE by using a USB modem. In the operating system, a 3G/LTE modem is displayed as a network interface named `pppX`.

To connect to the Internet, you can use the services of any mobile operator. To do this, buy a SIM card and enable the required services (if necessary). For more information on the terms of connecting to the mobile Internet, contact your mobile provider.

Warning: An appliance that is configured to access the Internet via a 3G/LTE modem cannot simultaneously connect to the Internet via any other interfaces (Ethernet or Wi-Fi).

If you are going to use Verizon or Vodafone as your mobile operator, connect the 3G/LTE modem to your appliance's USB port and set your mobile operator by executing the command `inet usb-modem set provider {verizon | vodafone}`.

If you are going to use another mobile operator, configure 3G/LTE connection parameters:

1. Contact your mobile operator and get the access point phone number, your user name, and password.
2. Connect the 3G/LTE modem to your appliance's USB port.
3. Specify the new operator's name by executing the command

   ```
   inet usb-modem add provider <operator name>
   ```

   Note: When you add a new provider, it is automatically set as your default provider.

4. Specify an IP address or a DNS name of the access point on the Internet by executing the command

   ```
   inet usb-modem set connection address <IP address | DNS name>
   ```

5. Specify the phone number (in the USSD command format) of the Internet access point by executing the command

   ```
   inet usb-modem set phone *99***1#
   ```

6. If necessary, specify the user name by executing the command

   ```
   inet usb-modem set user <user name>
   ```

7. If necessary, specify the password for the connection by executing the command

   ```
   inet usb-modem set password <password>
   ```

8. Set the new provider as default by executing the command

   ```
   inet usb-modem set provider <operator name>
   ```

After you have set your operator, do the following:

1 If your SIM card is PIN-protected, you can specify the PIN by executing the command

```
inet usb-modem set pin <PIN>
```

You can cancel PIN usage by executing the command

```
inet usb-modem reset pin
```

2 Enable the USB modem by executing the command

```
inet usb-modem mode on
```

Note: With some 3G/LTE modem models, it may take several minutes from the moment you have connected the modem to establish connection.

To optimize your 3G/LTE mobile network connection and to reduce the possibility of errors, we recommend you to configure the daily automatic reboot of your appliance. You can configure it to be done at a non-critical time (for example, during the night). To do this:

1 Enable the automatic daily reboot by executing the command

```
machine set dailyreboot mode on
```

2 Set the automatic daily reboot time by executing the command

```
machine set dailyreboot time <automatic reboot time>
```

The automatic reboot time should be specified in the `hh:mm` format (from `00:00` to `23:59`).

Note: To view whether the automatic reboot is enabled or disabled, as well as the current automatic reboot time, you can execute the command `machine show dailyreboot`.

Here is an example of the commands usage:

```
inet usb-modem add provider xtelecom
inet usb-modem set connection address internet.xtelecom.it
inet usb-modem set phone *99***1#
inet usb-modem set user xtelecom
inet usb-modem set password xtelecom
inet usb-modem set pin 1234
inet usb-modem mode on
machine set dailyreboot mode on
machine set dailyreboot time 03:30
```

Viewing the System Info

To obtain information on the state of ViPNet Coordinator HW/VA components or view logs, you may use the following commands:

- To view information about RAM and swap file usage, execute this command:

  ```
  machine show memory
  ```

- To view network interfaces' general parameters:

  ```
  inet show interface
  ```

  The interfaces are designated in the following way (with N standing for a number):

  - `lo`, the loopback interface.
  - `ethN`, an Ethernet interface.
  - `wlanN`, a Wi-Fi interface.

    To view the Wi-Fi interfaces' detail parameters, execute the command

    ```
    inet show wifi
    ```

  - `pppN`, one of the following:
    - a 3G/LTE USB modem,
    - an L2TP client-to-site IPsec connection.

    To view the 3G/LTE interfaces' detail parameters, execute the command

    ```
    inet show usb-modem
    ```

- To view a network interface configuration file:

  ```
  iplir show config <network interface name>
  ```

- To check connection with an unprotected or tunneled host:

  ```
  inet ping <IP address>
  ```

- To check connection with a ViPNet host:

  ```
  iplir ping <ViPNet host identifier>
  ```

  When you type an identifier, autocomplete and prompting features work. The data for prompting is taken from the list of ViPNet Coordinator HW/VA host's links.

- To view the firewall configuration file:

  ```
  iplir show config firewall
  ```

- To view the IP packets log:

  ```
  iplir view
  ```

- To view the event log:

  ```
  machine show logs
  ```

Providing Reliable Access to Network Resources by Using Alternate Traffic Channels

About Using Alternate Traffic Channels

If you need to stabilize access to a network or to a web resource, you can use two independent channels, which will duplicate each other in case of a channel's failure or will have the traffic balanced between them. For example, to avoid connection jams and failures, you may access the Internet via two providers, which allows you to distribute your traffic and to have an Internet connection backup channel. The use of alternate channels is controlled by the *loadbalancer* service.

Note: You can use alternate channels only for unencrypted (non-VPN) connections. Currently the service supports two alternate channels of the Ethernet type.

You can enable the usage of alternate channels in following modes:

- The redundant channel mode. In this case, you set one of the two channels as the primary channel. If it fails, the traffic will be transferred via the alternate channel. In this case, the connection through the primary channel will be periodically checked, and if it recovers, the traffic will be redirected back to it.
- The traffic load balancing mode. In this case, the traffic is balanced between the two channels with the priority that you can specify. For example, you can set one channel to transfer 80% of your traffic, and the other one, to transfer the rest. If any of the channels fails, 100% of the traffic will be transferred via the operable channel, until the failed channel recovers.

When the alternate channels usage is enabled, the *loadbalancer* service periodically checks the two channels' availability by attempting to access the connection test IP address. The service tries to access the test IP address by an ICMP request via either of the channels, in turn. The address' availability via a channel means that the channel is active, and its unavailability means that the channel has failed. If both channels fail, the service keeps checking them, and when any of them recovers, the traffic is automatically redirected to it.

When you will be configuring the *loadbalancer* service, you will need to select a test IP address that is accessible via both channels, when they are available.

Commands for Managing the Loadbalancer Service

All commands available for controlling the *loadbalancer* service are listed below:

```
service loadbalancer
```

- `start` starts the loadbalancer service.
- `stop` stops the loadbalancer service.
- `mode [on|off]` enables or disables the automatic start of the service at the OS startup.
- `add provider <channel_name>` adds a new channel (provider) with the name that you specify.
- `delete provider <channel_name>` deletes the respective channel (provider).
- `set`
    - `mode [failover|balancing]` determines whether the loadbalancer works in the redundant channel mode or in the traffic load balancing mode (see About Using Alternate Traffic Channels on page 21).
    - `provider <channel_name> gateway <ip_address>` assigns the IP address of the default gateway when working via the respective channel.
    - `provider <channel_name> interface <interface>` determines the network interface of the ViPNet Coordinator HW/VA host to be used when working via the respective channel.
    - `provider <channel_name> weight <weight>` assigns the respective weight to the channel. This weight (an integer from 1 to 10) is used in the traffic load balancing mode and determines the shares of the two channels in the overall traffic transfer.

      The ratio of the weights of the two channels determines their share in the traffic. For example, if you assign weights 2 and 4 to your channels, one third of the traffic will pass via the first channel, and the rest, via the other channel. If you set weights 3 and 6, the traffic share will be the same: one third to two thirds.
    - `testip <ip_address>` sets the connection test IP address (see About Using Alternate Traffic Channels on page 21).

    - `polltime <time>` sets the test IP address poll time (from 10 to 600 seconds). The default poll time is 10 seconds.
    - `provider <channel_name> default` sets the default channel. This command is used in the redundant channel mode. When the default channel is available, all the traffic is passed over it.
- `show` displays the state, the mode, the properties of the channels, and other settings of the loadbalancer.
- `nat`
    - `add localnet <network_1,network_2,...>` sets the list of the internal networks (see Internal network on page 79), for which ViPNet Coordinator HW/VA will be performing NAT. The networks are added by specifying their IP addresses in the CIDR format, for example, `192.168.0.0/24`.
    - `delete localnet <network_1,network_2,...>` disables NAT for the specified internal networks.

Configuring Integrated Services

Configuring Network Services

A ViPNet Coordinator HW/VA appliance can provide various network services to hosts on a local network, making it easy to deploy a small office network.

Configuring a Wi-Fi Access Point

If your appliance has a Wi-Fi adapter (displayed as a network interface named wlan0), you may use ViPNet Coordinator HW/VA as a Wi-Fi access point.

Note: A ViPNet Coordinator HW/VA host may also function as a Wi-Fi client (see Connecting to a Wi-Fi Network on page 14). Functioning as a client and an access point at the same time is not supported in ViPNet Coordinator HW/VA.

If the Wi-Fi access point is switched on, the wlan0 network interface is automatically assigned with the IP address 192.168.20.1, and a DHCP server is launched on this interface. The DHCP server has the following fixed parameters you can't edit:

- The range of allocated IP addresses: 192.168.20.2–192.168.20.20.
- DNS and NTP servers' address: 192.168.20.1 (the address of the wlan0 network interface).

Note: To ensure connections between devices on a Wi-Fi network and computers on an Ethernet network, on the ViPNet Coordinator HW/VA host, in the firewall configuration file, configure forward rules allowing IP packets between these networks (see Configuring Unencrypted Traffic Filtering Rules on page 55).

To configure ViPNet Coordinator HW/VA as a Wi-Fi access point:

1. Switch the wlan0 network interface to the access point mode with this command:

   ```
   inet wifi role access-point
   ```

2. Specify the name of your Wi-Fi network and user authentication mode. To do this, use one of the following commands:

   - If no authentication in the network is required, execute this command:

     ```
     inet wifi access-point ssid <network name> authentication open
     ```

- If you need a password for authentication, choose the WPA-PSK or WPA2-PSK authentication mode and specify a password. To do this, execute this command:

  ```
  inet wifi access-point ssid <network name> authentication {wpa-psk |
  wpa2-psk} passphrase <password>
  ```

  You should tell this password to the users connecting to your Wi-Fi network.

3 If you need to specify a wireless connection standard for your Wi-Fi network, execute this command:

```
net wifi server hwmode {a | b | g}
```

The following standards are supported:

- a (IEEE 802.11a): 5 GHz, connection speed up to 54 Mbit/sec.
- b (IEEE 802.11b): 2,4 GHz, connection speed up to 11 Mbit/sec.
- g (IEEE 802.11g): 2,4 GHz, connection speed up to 54 Mbit/sec (used by default).

4 If you need to specify the number of used Wi-Fi channel, execute this command:

```
inet wifi server channel <channel number>
```

By default, channel number 1 is used. Numbers from 1 to 14 are valid.

5 To enable or disable the wlan0 network interface, execute this command:

```
inet wifi mode {on | off}
```

Configuring the DHCP Server

ViPNet Coordinator HW/VA may function as a DHCP server in a LAN (see DHCP server on page 78). If the ViPNet Coordinator HW/VA host is used as a Wi-Fi (see Configuring a Wi-Fi Access Point on page 25) access point, then, on the corresponding network interface, the DHCP server is automatically launched, and you can't change its parameters.

Note: If you want to enable the ViPNet Coordinator HW/VA integrated DHCP relay, see the section Configuring the DHCP Relay.

To run the DHCP server on an Ethernet network interface, you should configure the server manually. To do this:

1 Specify the Ethernet network interface, the DHCP server will run on, with the following command:

```
inet dhcp interface <interface name>
```

The interface you specify should be assigned a static IP address.

2 Specify the range of IP addresses, the server will allocate, with the following command:

```
inet dhcp range <start IP address> <end IP address>
```

3 To specify the default gateway IP address, execute this command:

```
inet dhcp router <IP address>
```

4 To enable or disable DHCP server autostart at ViPNet Coordinator HW/VA startup, execute the following command:

```
inet dhcp mode {on | off}
```

DHCP server autostart is disabled by default.

Note: When you enable or start the DHCP server, make sure that the DHCP relay service is disabled and stopped.

5 To start or stop the DHCP server, execute the following command:

```
inet dhcp {start | stop}
```

6 To view the current DHCP server parameters, execute this command:

```
inet show dhcp
```

Configuring the DNS Server

ViPNet Coordinator HW/VA may function as a DNS server in a LAN (see DNS server on page 78). A DNS server integrated into ViPNet Coordinator HW/VA redirects the incoming DNS requests to a superior DNS server and transfers the received responses to its own clients. By default, client requests are redirected to root DNS servers. You can specify a custom list of DNS servers the DNS requests should be redirected to.

To configure the DNS server:

1 To add the DNS server's IP address, ViPNet Coordinator HW/VA will redirect DNS requests to, execute this command:

```
inet dns add <IP address>
```

By default, client requests are redirected to root DNS servers.

2 To delete a DNS server's address from the list, execute this command:

```
inet dns delete <IP address>
```

3 To enable or disable DNS server autostart at ViPNet Coordinator HW/VA startup, execute the following command:

```
inet dns mode {on | off}
```

DNS server autostart is enabled by default.

4 To start or stop the DNS server, execute the following command:

```
inet dns {start | stop}
```

5 To view the current DNS server parameters, execute this command:

```
inet show dns
```

Configuring the NTP Server

ViPNet Coordinator HW/VA may function as an NTP server in a LAN (see NTP server on page 80). To inform its clients about exact time, an integrated NTP server automatically synchronizes the system clocks with the universal time.

To configure the NTP server:

1 To add the NTP server's IP address for ViPNet Coordinator HW/VA's system time synchronization, execute this command:

```
inet ntp add {IP address | DNS name}
```

By default, the server `pool.ntp.org` is used for the synchronization.

2 To delete an NTP server's address from the list, execute this command:

```
inet ntp delete {IP address | DNS name}
```

3 To enable or disable NTP server autostart at ViPNet Coordinator HW/VA startup, execute the following command:

```
inet ntp mode {on | off}
```

NTP server autostart is enabled by default.

4 To start or stop the NTP server, execute the following command:

```
inet ntp {start | stop}
```

5 To view the current NTP server parameters, execute this command:

```
inet show ntp
```

Configuring a Proxy Server

A proxy server facilitates secure work of corporate network users on the Internet by controlling their access to web resources. If a user addresses a web resource over the HTTP or FTP protocol, the user's request is processed by a proxy server, which can either download the requested data and transfer them to the user, or deny access to the resource.

ViPNet Coordinator HW/VA has an integrated proxy server, which has the following features:

- Data caching to speed up user access to commonly used resources.
- A transparent proxy intercepts web communication without requiring any special client configuration.
- Web content control (see Configuring Content Control on page 32).
- Virus check of the web content (see Configuring the Anti-Virus on page 34).

Note: ViPNet Coordinator HW modifications HW 100 X1/X4/X5/X6 do not support proxy server, including the anti-virus and content control filtering.

In the scheme below, you can see how a proxy server processes a user request.

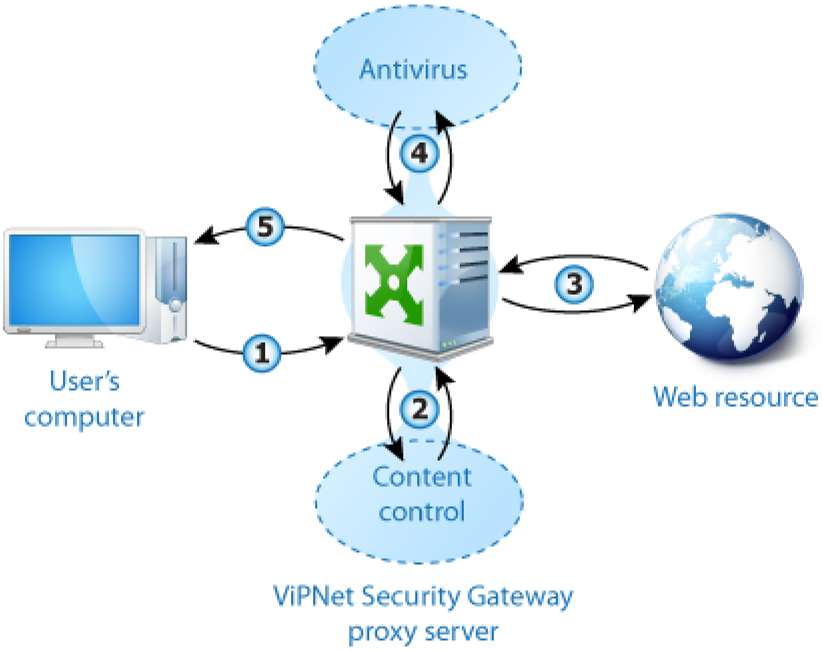


Figure 1: A proxy server with content control and virus check

If you want to enable the integrated proxy server in order to secure Internet access for your local users, you should configure general parameters of the proxy server first (see Configuring General Parameters on page 30). Optionally you can enable content filtering and antivirus protection. For more information see Configuring Content Control (on page 32) and Configuring the Anti-virus (on page 34).

Configuring the basic parameters implies specifying the external network interface of the server, the listening IP addresses, and IP addresses of the local networks that are allowed to use the proxy server. You may also enable the transparent proxy server mode.

When the proxy server functions as a 'non-transparent' proxy (the transparent mode is disabled), in user programs like a web browser, you should specify the proxy server's IP address and port.

When the proxy server functions in the transparent mode, advanced configuring of the programs is not required. The users are forced to use the proxy server. On the users' computers, specify the IP address of the proxy server (the ViPNet Coordinator HW/VA host) as the default gateway.

Configuring General Parameters

To configure the general parameters of the proxy server:

1 Specify the IP addresses and ports the proxy server will use to receive user requests:

- To add an IP address and a port, execute the following command:

```
service http-proxy listen-address add <address> <port>
```

- To view the list of specified IP addresses and ports, execute this command:

```
service http-proxy listen-address list
```

- To delete an IP address and a port, execute this command:

```
service http-proxy listen-address delete <address> <port>
```

⚠ **Warning:** We recommend you to use network interfaces with static IP addresses to receive requests. When the network interfaces' IP addresses are changed, you should stop the proxy server service, then specify current IP addresses for the addresses used to establish connection, and then start the proxy server again.

2 To specify the public IP address ViPNet Coordinator HW/VA will use to connect to the Internet, execute this command:

```
service http-proxy external-address set <IP address>
```

To view the specified public IP address, execute this command:

```
service http-proxy external-address show
```

3 Specify a list of networks, which are allowed to use the proxy server. By default, you can use the proxy server in any private network (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16).

- To add a network's IP address, execute this command:

  ```
  service http-proxy allow address add <network address>/<subnet prefix's
  length>
  ```

  For example:

  ```
  service http-proxy allow address add 192.168.10.0/24
  ```

- To view the networks list, execute this command:

  ```
  service http-proxy allow address list
  ```

- To delete an IP address from the networks list, execute this command:

  ```
  service http-proxy allow address delete <network's address>
  ```

4 If necessary, set the proxy server's cache size using the following command:

```
service http-proxy cache <size>
```

The cache size is specified in megabytes. The default value is 256 MB. Cache is used to store the copies of the often-requested data.

5 To enable or disable the transparent proxy server mode, execute this command:

```
service http-proxy transparent mode {on | off}
```

Warning: When you enable the transparent proxy server mode, in the firewall configuration file, static NAT rules are automatically created. When you disable the transparent proxy server mode, these rules are automatically deleted from the file.

If you had configured custom NAT rules, we recommend you to check them upon you have disabled the transparent proxy server mode (see Configuring Unencrypted Traffic Filtering Rules on page 55).

6 To enable or disable automatic startup of the proxy server at ViPNet Coordinator HW/VA startup, execute the following command:

```
service http-proxy mode {on | off}
```

7 To run the proxy service at once, execute the following command:

```
service http-proxy start
```

Note: Before you start the proxy server, you should specify an IP address and port to receive connections from users, as well as ViPNet Coordinator HW/VA's public IP address.

To stop the proxy service, execute this command:

```
service http-proxy stop
```

Note: If you want to provide access to the Internet via the proxy server for unprotected hosts, you should configure proper traffic processing rules (see Configuring Unencrypted Traffic Filtering Rules on page 55).

Configuring Content Control

Content control allows you to block unwanted web resources. A database of URL addresses is used for web content filtering. You can fetch this base from the Internet or from another ViPNet Coordinator HW/VA host, which is used as a URL database server.

URLs in the database can be divided into broad classes of content, such as gambling, online shopping, social networking, and so on. The web filter can block these classes of content independently from each other. If the user requests a URL that matches a blocked URL pattern, the request will be rejected and the user will be informed that the web page can't be displayed.

To configure content control parameters:

- Specify the URL database source and ViPNet Coordinator HW/VA's role using this command:

  ```
  service http-proxy redirector role {server | client}
  ```

  If you assign the `server` role, the URL database will be fetched from the Internet. ViPNet Coordinator HW/VA will function as a server, which can be used by other ViPNet Coordinator HW/VA hosts to fetch the database from.

  If you assign the `client` role, the URL database will be fetched from another ViPNet Coordinator HW/VA host, which functions as a server and updates the database from the Internet.

- If you have assigned the URL database client role to a ViPNet Coordinator HW/VA host, specify another ViPNet Coordinator HW/VA host as a server. To do this, execute the following command:

```
service http-proxy redirector client server-address <ViPNet Coordinator
HW/VA address>
```

To specify ViPNet Coordinator HW/VA's address, you can use one of the following parameters:

- An IP address.
- A hexadecimal ViPNet host identifier in the format of `0xAAAAAAAA`.

- To fetch the URL database at once, execute the following command:

  ```
  service http-proxy redirector fetch
  ```

  The database size is about 100 MB.

- To configure automatic URL database update, execute this command:

  ```
  service http-proxy redirector schedule fetch <update frequency>
  ```

  You can specify the update frequency by using one of the following values:

  - `none`, to disable the automatic database update;
  - a number from 1 to 5, to update the database 1 to 5 times a day in equal time periods.

  If you set the database to be updated once a day, it will be updated at 0:00 every day. If you set the database to be updated three times a day, it will be updated at 0:00, 8:00, and 16:00 every day.

  For example: `service http-proxy redirector schedule fetch 3`

  To view the automatic update schedule, execute this command:

  ```
  service http-proxy redirector schedule fetch list
  ```

- Specify content classes for the proxy server to ban. To do this:

  - To view the list of content classes to choose from, execute the following command:

    ```
    service http-proxy redirector category list
    ```

    A list of banned (**Blocked**) and available to ban (**Available**) content classes will be displayed as follows:

    ```
    Blocked: jobsearch, drugs, violence...
    Available: adv, aggressive, alcohol, anonvpn, automobile/bikes...
    ```

  - To add a content class to **Blocked**, execute this command:

    ```
    service http-proxy redirector category add <content class>
    ```

  - To delete a content class from **Blocked**, execute this command:

    ```
    service http-proxy redirector category delete <content class>
    ```

- If necessary, configure a list of exceptions — the hosts for which web content will not be filtered. To do this:

- To add an IP address to exceptions, execute the following command:

  ```
  service http-proxy redirector exception add <IP address>
  ```

- To view the specified exceptions list, execute this command:

  ```
  service http-proxy redirector exception list
  ```

- To delete an IP address from exceptions, execute this command:

  ```
  service http-proxy redirector exception delete <IP address>
  ```

- To enable or disable content control, execute the following command:

  ```
  service http-proxy redirector mode {on | off}
  ```

Configuring the Anti-Virus

If you use ViPNet Coordinator HW/VA as a proxy server, you can enable antivirus check of all HTTP traffic passing through the proxy server in both directions: from the Internet to the user and from the user to the Internet (for example, when you are adding an attachment to an e-mail message via a web interface).

Antivirus protection is performed by Clam AntiVirus, a free open source software developed by Sourcefire company.

To configure antivirus protection, do the following:

- To fetch the virus database at once, execute the command

  ```
  service http-proxy antivirus clamav fetch
  ```

  The database size is about 100 MB.

- To configure automatic virus database update, execute the command

  ```
  service http-proxy antivirus clamav schedule fetch <update frequency>
  ```

  You can specify the update frequency by using one of the following values:

  - `none`, to disable the automatic database update;
  - a number from 1 to 5, to update the database 1 to 5 times a day in equal time periods.

  If you set the database to be updated once a day, it will be updated at 0:00 every day. If you set the database to be updated three times a day, it will be updated at 0:00, 8:00, and 16:00 every day.

  For example: `service http-proxy antivirus clamav schedule fetch 3`

- To view the automatic update schedule, execute the command

  ```
  service http-proxy antivirus clamav schedule fetch list
  ```

- To enable scanning web content for viruses, execute the command

  ```
  service http-proxy antivirus clamav mode on
  ```

- To disable scanning web content for viruses, execute the command

  ```
  service http-proxy antivirus clamav mode off
  ```

Configuring a VoIP Server

The VoIP server implemented as a component of the ViPNet Coordinator HW/VA appliance allows you to build a corporate IP telephony system.

If you want to organize protected voice communication between several offices, deploy a VoIP server based on ViPNet Coordinator HW/VA in each office and setup trunks connecting the servers to each other. Trunks allow users of different VoIP servers to call each other.

The integrated VoIP server supports SIP protocol. Users can connect to the server by means of any software SIP client or hardware SIP phone. SIP clients should be installed according to the following rules:

- Software SIP clients should be installed on protected ViPNet hosts or tunneled hosts. In SIP client software, specify the ViPNet Coordinator HW/VA visibility address as the SIP server address.
- Hardware SIP phones should be tunneled by ViPNet Coordinator HW/VA. Specify the IP address of the external interface of ViPNet Coordinator HW/VA as the SIP server address.
- To ensure proper operation of SIP clients tunneled by ViPNet Coordinator HW/VA, configure filtering rules for unprotected traffic (see Configuring Unencrypted Traffic Filtering Rules on page 55) that allow inbound local connections of tunneled clients with ViPNet Coordinator HW/VA over TCP and UDP to port 5060.

- If you are connecting mobile devices to the ViPNet Coordinator HW/VA over the IPsec channel (see Configuring a Client-to-Site IPsec Connection on page 45), in ViPNet Network Manager, configure the ViPNet Coordinator HW/VA host to tunnel them. For this, select the ViPNet Coordinator HW/VA host and, on the **Tunneling** tab, specify the IP address range distributed among IPsec devices to be tunneled. The range is 192.168.30.1 — 192.168.30.40.

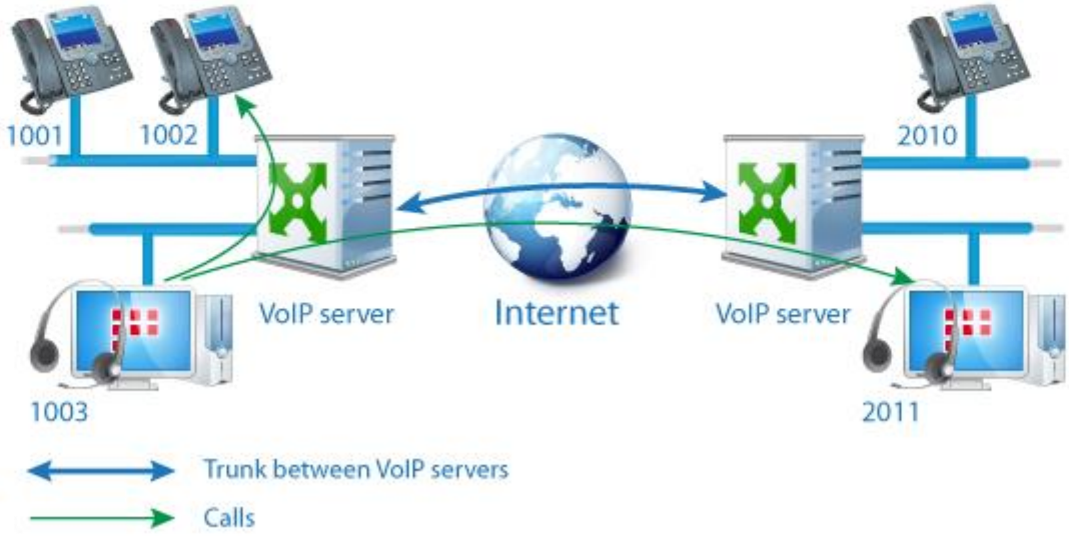


Figure 2: Using the integrated proxy server

To configure the integrated VoIP server, do the following:

1. Before you start configuring the VoIP server, log on as the administrator. To do this, execute the `enable` command and then enter the ViPNet host's administrator password.

2. Specify a network interface of ViPNet Coordinator HW/VA that SIP clients should connect to. To do this, execute the following command:

   ```
   service sip listen internal <interface name>
   ```

   Note: To display a list of network interfaces, execute this command with no interface name specified: `service sip listen internal`.

   If you specify the name of a network interface that does not have an IP address, an error message will be displayed. If the VoIP service is running, restart it after executing this command.

3. Configure the list of telephone numbers handled by your server:

   - To add a telephone number, execute this command:

```
service sip phone add number <user's number> name <user's name> surname
<user's surname> password <user's password>
```

A telephone number should consist of four digits exactly. For example:

```
service sip phone add number 1015 name John surname Smith password
qwerty
```

If you try to add a number that already exists, a message will be displayed prompting you to overwrite the number details or keep it unchanged.

- To delete a telephone number, execute this command:

  ```
  service sip phone delete <user's number>
  ```

4. If necessary, setup trunks connecting your VoIP server to remote VoIP servers:

   - Specify a network interface of ViPNet Coordinator HW/VA that remote VoIP servers should connect to. To do this, execute the following command:

     ```
     service sip listen external <interface name>
     ```

     > **Note:** To display a list of network interfaces, execute this command with no interface name specified: `service sip listen external`.

     If you specify the name of a network interface that does not have an IP-address, an error message will be displayed. If the VoIP service is running, restart it after executing this command.

   - To create a trunk connection to a remote server, execute this command:

     ```
     service sip trunk add name <server's name> address <server's IP
     address> local <local numbers> remote <remote numbers>
     ```

     Local numbers are telephone numbers that are handled by your VoIP server and should be accessible for users of a remote VoIP server. Remote numbers are telephone numbers that are handled by a remote VoIP server and should be accessible for users of your server.

     You should specify telephone numbers with a mask. For example:

     ```
     service sip trunk add name RemoteVoIP address 214.56.112.47 local 1XXX
     remote 2XXX
     ```

   - To delete a trunk, execute this command:

     ```
     service sip trunk delete <server's name>
     ```

5. To enable or disable VoIP server autostart on ViPNet Coordinator HW/VA startup, execute this command:

   ```
   service sip mode {on | off}
   ```

6 To start the VoIP server at once, execute the following command:

```
service sip start
```

To stop the VoIP service, execute this command:

```
service sip stop
```

To view the properties of the VoIP server, use the following commands:

- To view the status of the VoIP service, execute this command:

  ```
  service show sip
  ```

- To view a list of users who are active at the moment, execute this command:

  ```
  service sip phone active
  ```

- To view a list of telephone numbers handled by the server, execute this command:

  ```
  service sip phone list
  ```

- To view a list of trunks connecting your server to remote VoIP servers, execute this command:

  ```
  service sip trunk list
  ```

Configuring an IPsec Gateway

In corporate networks, you often need to protect connections with remote networks or hosts. For example, you have a corporate application server that should be accessible from the Internet via a protected channel. Sometimes you can't solve this problem using ViPNet technology. For example, it is impossible to install ViPNet software on mobile devices.

With ViPNet Coordinator HW/VA, you may protect traffic using encryption over the IPsec protocol. In this case, ViPNet Coordinator HW/VA functions as a ViPNet–IPsec gateway. ViPNet Coordinator HW/VA supports two types of IPsec connection: site-to-site (see Configuring a Site-to-Site IPsec Connection on page 41) and client-to-site (see Configuring a Client-to-Site IPsec Connection on page 45). For either of them, ViPNet Coordinator HW/VA supports authentication by a pre-shared key (PSK) or by a certificate (RSA). ViPNet Coordinator HW/VA supports up to 40 concurrent client-to-site IPsec connections.

⚠ **Warning:** When you use a ViPNet Coordinator HW/VA as an IPsec gateway, it needs to have a public static IP address. Therefore, you cannot deploy a ViPNet Coordinator HW/VA as an IPsec gateway if it is located behind a NAT.

If you want to use PSK authentication, we strongly recommend that you configure the IPsec connection in ViPNet Network Manager (see the document “ViPNet Coordinator HW/VA. Administrator's Guide”). RSA authentication is not supported in ViPNet Network Manager.

⚠ **Warning:** If you choose to configure the IPsec connection using the ViPNet Coordinator HW/VA command line interface, you should not change any IPsec settings for your host in ViPNet Network Manager. Otherwise, the settings made in the command line interface will be lost when you send keys to the ViPNet Coordinator HW/VA host.

To configure the IPsec connection in the ViPNet Coordinator HW/VA command line interface, do the following:

1. In ViPNet Network Manager, make sure that the IPsec gateway feature is disabled for you ViPNet Coordinator HW/VA host (see the document “ViPNet Coordinator HW/VA. Administrator's Guide”).
2. In ViPNet Network Manager, send keys to your ViPNet Coordinator HW/VA host.

3 In the ViPNet Coordinator HW/VA command line interface, configure a site-to site (see Configuring a Site-to-Site IPsec Connection on page 41) or a client-to-site (see Configuring a Client-to-Site IPsec Connection on page 45) IPsec connection.

4 Do not enable the IPsec gateway feature in ViPNet Network Manager. Otherwise, your IPsec settings will be lost.

Configuring a Site-to-Site IPsec Connection

Assume that your corporate network is protected with a ViPNet VPN. You need to create a protected communications channel with your partner company, but they don't use the ViPNet technology.

In this case, you may establish a tunnel between the two corporate networks over with the IPsec protocol. An IPsec tunnel is an encrypted traffic channel established between the two IPsec gateways deployed in each of the two networks. There is a variety of IPsec gateway software servers and appliances (Cisco appliances, servers running Linux, FreeBSD, Windows Server, and others).

You may use a ViPNet Coordinator HW/VA host as your network's IPsec gateway.

⚠ **Warning:** You can't create a protected IPsec channel between two ViPNet Coordinator HW/VA hosts that belong to the same ViPNet network. However, you may do it for hosts belonging to your partner networks (for more information, see the document “ViPNet VPN. User's Guide”, the chapter “Connecting to a Partner Network”).

Consider the following example:

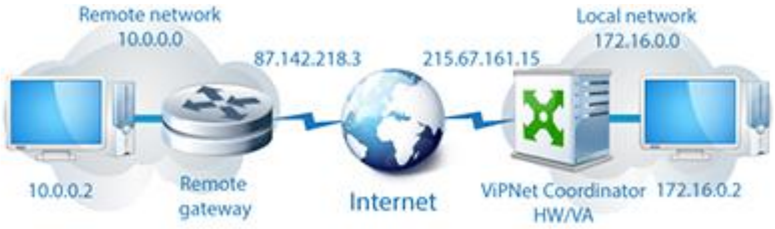


Figure 3: Connecting over the IPsec protocol

A host on the remote network with the IP address 10.0.0.2 establishes connection to a ViPNet host with the IP address 172.16.0.2. IP packets from the 10.0.0.2 host are transferred unencrypted to the remote IPsec gateway device. On the device, IP packets are encrypted and the source address is substituted with the device's public address 87.142.218.3. Then, ViPNet Coordinator HW/VA performs decryption and forwards the packets to the host 172.16.0.2 in their original state.

When configuring IPsec connection on a remote IPsec gateway and ViPNet Coordinator HW/VA hosts, make sure that the IP address spaces of the communicating networks comply with each other, as well as the devices' real addresses, the encryption protocol, and the authentication method.

To configure a site-to-site IPsec connection, on ViPNet Coordinator HW/VA:

1. Specify the IP address of the network interface, which you will use to connect to the remote network, with this command:

   ```
   service ipsec listen <IP address>
   ```

2. Specify the IP address of the device acting as an IPsec gateway in a remote network with the following command:

   ```
   service ipsec site2site peer add <remote IP address>
   ```

> **Note:** When you add an IP address of a remote network gateway, unencrypted traffic processing rules (see Configuring Unencrypted Traffic Filtering Rules on page 55) are created to allow connections between ViPNet Coordinator HW/VA and the remote network gateway over the ESP protocol and UDP connections via ports 500 and 4500.

3. Specify authentication parameters for the remote network gateway.

   If you want to use pre-shared key authentication (PSK):

   - Set the PSK authentication type:

     ```
     service ipsec site2site peer set <remote IP address> auth type psk
     ```

   - Set a password for the remote network gateway authentication with the following command. The password should be 8 to 63 characters long.

     ```
     service ipsec site2site peer set <remote IP address> psk password add
     <password>
     ```

> **Warning:** The password should not contain the following characters: the question mark (?), the backslash (\), the single quote (').

   If you want to use certificate authentication (RSA):

   - Set the RSA authentication type:

     ```
     service ipsec site2site peer set <remote IP address> auth type rsa
     ```

   - Make sure that you have the certificate and the private key of your ViPNet Coordinator HW/VA host, the certificate of the remote gateway, the certification authority root certificate, and the certificate revocation list (CRL). Import the certificates to the ViPNet Coordinator HW/VA host (see Importing Certificates and CRLs on page 49).

- Specify the required certificates and CRLs:

  ```
  service ipsec site2site peer set <remote IP address> rsa hostcert <your
  host's certificate>
  ```

  ```
  service ipsec site2site peer set <remote IP address> rsa hostkey <your
  host's private key>
  ```

  ```
  service ipsec site2site peer set <remote IP address>rsa peercert
  <certificate of the remote gateway>
  ```

  ```
  service ipsec site2site peer set <remote IP address> rsa cacert
  <certificate of the certification authority>
  ```

  ```
  service ipsec site2site peer set <remote IP address> rsa crl
  <certificate revocation list>
  ```

  When executing the commands listed above, to view the list of available certificates, leave the certificate's name empty or type a question mark (?). For example, execute this command:

  ```
  service ipsec site2site peer set 87.142.218.3 rsa hostcert
  ```

4 Set the encryption parameters to protect the connection with:

- Specify the cryptographic algorithm with the following command:

  ```
  service ipsec site2site peer set <remote IP address> crypto {3des |
  aes}
  ```

- Specify the hash calculating algorithm with this command:

  ```
  service ipsec site2site peer set <remote IP address> hash <algorithm>
  ```

  The valid values are: `md5`, `sha1`, `sha256`, `sha384`, `sha512`.

- Specify the number of a Diffie–Hellman group by executing this command:

  ```
  service ipsec site2site peer set <remote IP address> group <group
  number>
  ```

  The valid values are: 1, 2, 5, 14, 15, 16, 17, 18.

- Specify session keys lifetime in seconds:

  ```
  service ipsec site2site peer set <remote IP address> lifetime <keys
  lifetime>
  ```

  The valid values are 60–86400.

5 Specify the addresses of the networks, you are configuring a protected channel for, by using this command:

```
service ipsec site2site peer set <remote IP address> spd add localnet
<local network IP address> remotenet <remote network IP address>
```

The local network and the remote network are the networks you are establishing connection between. Network addresses are specified in CIDR notation, for example 172.16.0.0/24.

The local server's IP address is the ViPNet Coordinator HW/VA's public address, while the remote server's IP address is the other IPsec server's public address.

6 To enable autostart for the services that connect over the IPsec protocol, at ViPNet Coordinator HW/VA startup, execute the following command:

```
service ipsec site2site mode on
```

7 To start the IPsec service, execute the following command:

```
service ipsec site2site start
```

8 On the ViPNet Coordinator HW/VA host, configure a forward rule (see Configuring Unencrypted Traffic Filtering Rules on page 55) allowing traffic between the remote network and the local network you specified in ViPNet Network Manager.

Note: If you want computers from the remote network to access ViPNet hosts on your local network, configure firewall rules on your local hosts to allow inbound traffic from the remote hosts.

9 In case the ViPNet Coordinator HW/VA host is behind an external firewall, on the firewall, configure the corresponding rules, allowing the incoming UDP packets via ports 500 and 4500, whose destination is the local IP address of ViPNet Coordinator HW/VA.

10 Inform the remote network administrator about the requirement to configure ViPNet hosts' traffic routing on all remote network hosts, including the gateway. Visibility addresses should be used:

- Visibility addresses of ViPNet hosts are specified in the private network configuration file (the `accessip` parameter in the `[id]` section of each host). To view the configuration file, execute the `iplir show config` command.
- Visibility addresses of unprotected hosts are the same as their real IP addresses.

11 Make sure that the IPsec connection parameters on ViPNet Coordinator HW/VA and on the remote network gateway comply with each other.

12 To view the site-to-site connection settings, execute the following command:

```
service ipsec site2site show
```

Here is an example of site-to-site connection configuration (according to the scheme above):

```
service ipsec site2site listen 215.67.161.15
service ipsec site2site peer add 87.142.218.3
service ipsec site2site peer set 87.142.218.3 auth type psk
service ipsec site2site peer set 87.142.218.3 psk password add qwertyuiop
service ipsec site2site peer set 87.142.218.3 crypto aes
service ipsec site2site peer set 87.142.218.3 hash md5
service ipsec site2site peer set 87.142.218.3 group 5
service ipsec site2site peer set 87.142.218.3 lifetime 3600
service ipsec site2site peer set 87.142.218.3 spd add localnet 172.16.0.0/24
remotenet 10.0.0.0/24
service ipsec site2site mode on
service ipsec site2site start
```

Configuring a Client-to-Site IPsec Connection

Modern business processes require efficient use of mobile devices. With mobile devices, you can use corporate email, IP telephony, and other corporate network resources, even when you are far away from the office.

To implement protected access to corporate resources located on a ViPNet network, users may use various smartphones and tablets. In this case, traffic is protected by a combination of the IPsec and ViPNet technologies.

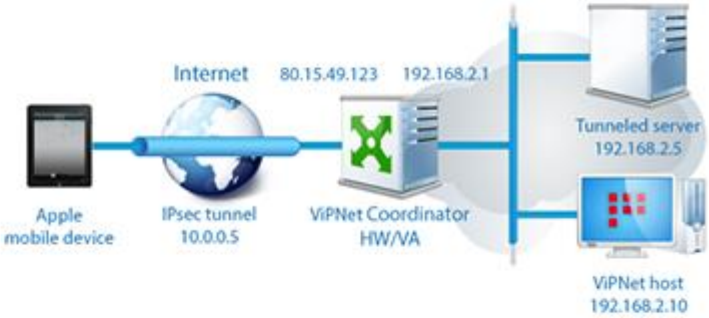


Figure 4: Apple mobile device connecting to ViPNet Coordinator HW/VA

A ViPNet Coordinator HW/VA host functions as an IPsec–ViPNet gateway providing access to tunneled and protected ViPNet hosts for mobile devices. A mobile device establishes connection to a ViPNet Coordinator HW/VA host over the IPsec protocol. A protected client-to site IPsec channel is created, and an IP address from the address pool 192.168.30.0/24 is automatically assigned to the mobile device. The mobile device's traffic is decrypted by the ViPNet Coordinator HW/VA host. Then, the unencrypted traffic is either forwarded to an unprotected host behind the ViPNet Coordinator HW/VA host or the traffic is encrypted by using ViPNet keys and is transferred to a protected ViPNet host. The hosts, located on the network protected by the ViPNet Coordinator HW/VA, are accessible from mobile devices by IP addresses.

The mobile devices connected to your ViPNet Coordinator HW/VA over the IPsec protocol have the coordinator set as their default gateway. Without proper configuration, these devices can access the resources of your corporate network, but cannot access the Internet. If you want to provide the mobile IPsec clients with access to the Internet, use the integrated proxy server or configure NAT rules on your ViPNet Coordinator HW/VA host.

ViPNet Coordinator HW/VA supports up to 40 concurrent client-to-site IPsec connections. Additionally, the number of mobile clients on your ViPNet network may be limited by your ViPNet VPN license.

If required, laptops and desktop computers running Windows or Mac OS may function as IPsec clients, too.

To configure a client-to-site IPsec connection, on ViPNet Coordinator HW/VA:

1. Specify the IP address of the network interface, which should be used by IPsec clients to connect to the ViPNet Coordinator HW/VA host, with this command:

   ```
   service ipsec listen <IP address>
   ```

2. Specify authentication parameters for L2TP connections.

   If you want to use pre-shared key authentication (PSK):

   - Set the PSK authentication type:

     ```
     service ipsec client2site peer auth type psk
     ```

   - Set a password for the remote network gateway authentication with the following command. The password should be 8 to 63 characters long.

     ```
     service ipsec client2site peer psk password add <password>
     ```

   Warning: The password should not contain the following characters: the question mark (?), the backslash (\), the single quote (').

   If you want to use certificate authentication (RSA):

   - Set the RSA authentication type:

     ```
     service ipsec client2site peer auth type rsa
     ```

   - Make sure that you have the certificate and the private key of your ViPNet Coordinator HW/VA host, the certification authority root certificate, and the certificate revocation list (CRL). Import the certificates and CRLs to the ViPNet Coordinator HW/VA host (see Importing Certificates and CRLs on page 49).

   - Specify the required certificates CRLs:

     ```
     service ipsec client2site peer rsa hostcert <your host's certificate>
     ```

```
service ipsec client2site peer rsa hostkey <your host's private key>
```

```
service ipsec client2site peer rsa cacert <certificate of the certification authority>
```

```
service ipsec client2site peer rsa crl <certificate revocation list>
```

When executing the commands listed above, to view the list of available certificates, leave the certificate's name empty or type a question mark (?). For example, execute this command:

```
service ipsec client2site peer rsa hostcert
```

3. Set the encryption parameters to protect the connection with:

    - Specify the cryptographic algorithm with the following command:

      ```
      service ipsec client2site peer crypto {3des | aes}
      ```

    - Specify the hash calculating algorithm with this command:

      ```
      service ipsec client2site peer hash <algorithm>
      ```

      The valid values are: `md5`, `sha1`, `sha256`, `sha384`, `sha512`.

    - Specify the number of a Diffie-Hellman group by executing this command:

      ```
      service ipsec client2site peer group <group number>
      ```

      The valid values are: 1, 2, 5, 14, 15, 16, 17, 18.

    - Specify session keys lifetime in seconds:

      ```
      service ipsec client2site peer lifetime <keys lifetime>
      ```

      The valid values are 60–86400.

4. Specify the IP address range for clients connecting to the ViPNet Coordinator HW/VA over the IPsec protocol:

   ```
   service ipsec client2site peer range <start IP address>-<end IP address>
   ```

5. If you want IPsec clients to access protected ViPNet hosts, in ViPNet Network Manager, for your ViPNet Coordinator HW/VA host, add the specified IP address range to the list of tunneled addresses. Then, send key set updates to ViPNet hosts.

6. If necessary, specify the IP address of the DNS server to be used by IPsec clients with this command:

   ```
   service ipsec client2site dns set <DNS server's IP address>
   ```

   By default, the IP address 8.8.8.8 (Google Public DNS) is used.

7. Specify a list of IPsec clients that will be able to access your protected resources:

    - To add an IPsec client, specify the name and password for the new user with the following command:

```
service ipsec client2site peer user add <user's name> password <user's
password>
```

Warning: The user's password should not contain the following characters: the question mark (?), the backslash (\), the single quote (').

- To delete an IPsec client, execute the following command:

  ```
  service ipsec client2site peer user delete <user's name>
  ```

- To view the list of IPsec clients, execute the following command:

  ```
  service ipsec client2site peer user list
  ```

8. To enable autostart for the services that connect over the IPsec protocol, at ViPNet Coordinator HW/VA startup, execute the following command:

   ```
   service ipsec client2site mode on
   ```

9. To start the IPsec service, execute the following command:

   ```
   service ipsec client2site start
   ```

10. Tell the IP address of the ViPNet Coordinator HW/VA, authentication parameters (pre-shared key or certificates), user's names and passwords to the users that need access to your corporate resources. The users should configure IPsec settings on their mobile devices or computers according to the provided information.

11. To view the client-to-site connection settings, execute the following command:

    ```
    service ipsec site2site show
    ```

Here is an example of client-to-site connection configuration (according to the scheme above):

```
service ipsec client2site listen 80.15.49.123
service ipsec client2site peer set 87.142.218.3 auth type psk
service ipsec client2site peer psk password add qwertyuiop
service ipsec client2site peer crypto aes
service ipsec client2site peer hash md5
service ipsec client2site peer group 5
service ipsec client2site peer lifetime 3600
service ipsec client2site peer range 10.0.0.2-10.0.0.254
service ipsec client2site dns set 192.168.2.2
service ipsec client2site peer user add JohnB password 2jhbff42
service ipsec client2site peer user add MrSmith password 32fra24f
service ipsec client2site mode on
service ipsec client2site start
```

Importing Certificates and CRLs

If you want to configure certificate authentication (RSA) for IPsec connections, you should import the required certificates and certificate revocation lists (CRLs) to the ViPNet Coordinator HW/VA first. If you don't have a certificate for your host, you can create a request for a certificate, and then import the issued certificate.

To obtain a certificate for your host:

1 Create a private key and a certificate request. To do this, in the ViPNet Coordinator HW/VA command line interface, execute the following command and specify the required certificate name, the private key length, and the hashing algorithm:

```
service cert request create name <certificate name> bits <key length>
digest {md5 | sha1}
```

The valid values for the key length are 1024, 1536, 2048, 3072, 4096.

When you execute this command, a private key and a request file named `<certificate name>_req.pem` will be created.

2 Connect a removable USB drive to your appliance, and then export the created request to the USB drive with the following command:

```
service cert export <request file name> via usb
```

3 Transfer the certificate request to your certification authority. Then, receive the issued certificate together with the certification authority's root certificate and the CRL.

To import a certificate, a private key or a CRL:

1 Connect a USB drive with the files you want to import to your appliance.

The supported file extensions are `.pem` for private keys, `.cer` for DER-encoded certificates, and `.crl` for certificate revocation lists.

2 In the ViPNet Coordinator HW/VA command line interface, execute the following command:

```
service cert import via usb
```

A list of the files found on the USB drive will be displayed.

3 Select the file you want to import.

To view a list of installed private keys, certificates, and CRL, execute the following command:

```
service cert list
```

To view a certificate, execute the following command:

```
service cert show cert <certificate name>
```

Configuring an Integrated Firewall

About the Integrated Firewall

The ViPNet Coordinator HW/VA appliance can perform filtering and address translation for public network (unencrypted) IP traffic (on page 79). You can find processing rules for public network traffic and firewall service parameters (see Configuring Service Parameters on page 52) in the firewall configuration file.

The configuration file consists of several sections. Each of these sections contains one or several unencrypted traffic processing rules. The processing rules include:

- anti-spoofing rules (see Configuring Anti-spoofing Rules on page 53);
- IP traffic filtering rules (see Configuring Unencrypted Traffic Filtering Rules on page 55);
- network address translation (NAT) rules.

To edit the firewall configuration file, in the command line interface (on page 78), execute the `iplir config firewall` command.

Information about the events related to IP traffic processing performed by ViPNet Coordinator HW/VA's firewall is written to the IP packets log (on page 68).

Configuring Service Parameters

You can configure firewall service parameters in the firewall configuration file, in the `[settings]` section. When the configuration file is created, the `[settings]` section does not contain any parameters. The default values described below are used.

The following parameters can be configured in the `[settings]` section:

- `max-connections` is the maximum number of concurrent connections. You should take into account that the number of processed real physical connections is three times less. The default value is `300000`; this is the maximum valid parameter value. If you need to limit the number of concurrent connections, you should set a lesser parameter value.
- `dynamic-ports` is a range of ports used for dynamic NAT. The default value is `60000-65000`.
- `connection-ttl-tcp` is a timeout in seconds. When this time interval expires, and the last TCP packet is registered, the TCP connection will be broken. The default parameter value is `3600` (60 min).
- `connection-ttl-udp` is a timeout in seconds. When this time interval expires, and the last UDP packet is registered, the UDP connection will be broken. The default value is `300` (5 min).
- `dynamic-timeouts` enables or disables the mode of connections dynamic timeouts (`yes` or `no`). The default parameter value is `no`.

  The dynamic timeouts mode is used to prevent flood attacks. When the number of connections reaches a certain percentage from the maximum, timeouts of all the connections are decreased by a certain number. The closer the connections number is to the maximum, the more timeouts are decreased (but the timeouts can't be decreased to a number less than a certain minimum). When the connections number is decreased to a certain percentage of the maximum, the original values of timeouts are restored.
- `cleanup-interval` is the frequency of removing expired connections. The default parameter value is `5` (seconds).

  Greater values result in less accurate removing of expired connections, while too small values lead to extra load on CPU.

Configuring Anti-spoofing Rules

Anti-spoofing rules allow you to assign IP addresses ranges for each network interface. Only the packets coming from the specified range of IP addresses will be allowed at this network interface. The packets from the IP addresses out of the range will be blocked. Besides, if any interface receives packets from the addresses specified as accessible for another interface, such packets will also be blocked. As the name suggests, the purpose of anti-spoofing is protection against spoofing, which is one of the network attack types based on IP addresses falsification. While spoofing, a malicious user sends you a packet where the source address is compromised and changed to the one your computer recognizes. For example, a packet from the Internet can be sent to a gateway with an address defined as an address of another VPN connected to this gateway. Then, the malicious user can gain access to a service, which could be accessed only from the private network. Anti-spoofing rules let you eliminate such a possibility.

You can configure anti-spoofing parameters in the `[antispoof]` section of the firewall configuration file.

The following parameters can be configured in the `[antispoof]` section:

- `antispoof` enables or disables anti-spoofing. This parameter can take `yes` or `no` values. The default parameter value is `no`.

- Other parameters have names identical to those of the network interfaces. The value of each parameter is a list of addresses, acceptable on this interface. The addresses list can include single addresses, address ranges, address masks, and keywords, separated by a comma. All the addresses on the list belong to the subnets connected to the current interface. In the addresses list, you may also specify the following keywords:

    - `anypublic` denotes all the addresses allowed in the Internet, in other words, all the addresses except for those intended for special purposes: for the local network interface (127.0.0.0/8) and for private networks (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16);

    - `subnet` denotes the whole subnet the current interface belongs to, taking into account the interface's address and subnet mask.

      If anti-spoofing is enabled, then the list of addresses, defined by the `subnet` keyword, is generated at each control daemon's startup or each change in network parameters.

The anti-spoofing section should always contain a list of all the network interfaces except for the local one (loopback). The local interface is not affected by any traffic processing rules, and any packets are always allowed on it. Moreover, the packets from 127.0.0.0/8 source addresses are blocked on all the interfaces controlled by ViPNet Coordinator HW/VA, regardless of anti-spoofing configuration, since this is a specification requirement.

At the control daemon's startup, it is checked that anti-spoofing is enabled. At the control daemon's startup, it is checked that anti-spoofing is enabled. If it is enabled, then it is checked that all the interfaces known to the control daemon are displayed in the `[antispoof]` section. The missing interfaces are automatically added to the section with the `subnet` value.

Here is an example of an anti-spoofing section:

```
[antispoof]
antispoof= yes
eth0= anypublic
eth1= 192.168.1.0/24
```

In this example, anti-spoofing is enabled and will function as follows:

- the `eth0` interface allows IP packets from all the addresses except for the private ones (the interface is supposed to be connected to the Internet);
- the `eth1` interface allows IP packets from the addresses ranging from 192.168.1.1 to 192.168.1.255 (the interface is supposed to be connected to the Internet).

In this example, if a packet from the Internet arrives to `eth0` from a 192.168.1.0/24 or 192.168.201.0/24 network, it will be blocked. Thus, reliable anti-spoofing protection is provided.

Configuring Unencrypted Traffic Filtering Rules

Packets that have been allowed by the anti-spoofing rules are processed according to filtering rules for unencrypted traffic. Unencrypted traffic filtering rules are specified in the `[local]`, `[broadcast]`, `[tunnel]`, and `[forward]` sections of the firewall configuration file.

In the `[local]` section, you can configure local packets filtering rules. A packet is local if its source or destination is the current host.

In the `[broadcast]` section, you can specify broadcast packets filtering rules.

The configuration file must contain `[local]` and `[broadcast]` sections. At the control daemon's startup, it is checked whether these sections exist. If any of the sections is omitted, it is automatically added to the firewall configuration file with the default rules.

In the `[forward]` section, you can specify filtering rules for forward packets, in other words, the packets that pass through the current host only on their way from source to destination. An empty `[forward]` section is included in the firewall configuration file by default. If you need forward packets to go through the host, you should allow them in the `[forward]` section.

In the `[tunnel]` section, you can configure tunneled packets filtering rules. Packets are tunneled if they are transferred between the hosts that are tunneled by this host and ViPNet hosts. The configuration file must contain a `[tunnel]` section. At the control daemon's startup, it is checked whether this section exists. If this section is omitted, it is automatically added to the firewall configuration file with the default rule, which allows traffic between all the tunneled hosts and all the protected hosts linked with the current host.

TCP, UDP, and ICMP IP packets are processed on the basis of different packets' parameters analysis. Filtering rules allow you to control the source protocol, address and port, the destination address and port, and the connection’s direction. You can limit the traffic flow by the connections established, applying filtering rules in the direction of the connection: process only the requests initiating connections in the specified direction and responses to them, as well as block the requests initiating connections in the reverse direction.

For processing traffic over IP protocols other than TCP, UDP and ICMP, virtual connections are created, based on IP addresses and the protocol number. Thus, the only thing you need to do is to specify an allowing rule only for the request initiating the connection, and responses will be received automatically (if they come from the IP address the request was sent to and over the same protocol).

Each of the sections can contain one or several filtering rules. The filtering rules' syntax is the same for all the sections.

Each rule is described by the `rule` parameter. Its value consists of the following components:

```
rule= <control component> <condition> <schedule> <action>
```

or

```
rule= <control component> <action> <condition> <schedule>
```

You should specify the control component at the beginning of the rule, and the schedule should follow the condition.

Each component consists of the parts called tokens. A token is a service word, which may be followed by a parameter.

Rule components, tokens forming them, and parameters and service words within the tokens are separated by spaces.

Control Component

The control component defines the rule properties that are not related to packet processing directly. It is always specified at the beginning of the rule. It may contain the following tokens in the given order:

- `num <number>` specifies the number of the rule in the section (0 to 65535). Numbers are used to denote the rules' priority: the less the number, the higher the priority. During packet processing, the conditions of higher priority rules are checked first, and, if the conditions match, the action specified in the rule is initiated. Then processing of other rules stops.

  The `num` token may be omitted; then, ViPNet Coordinator HW/VA will try to assign a number to the rule on its own, taking into account the preceding and subsequent rules. However, to avoid failures, we recommend you to specify the number explicitly for each rule.

- `name "<name>"` specifies the rule name (rule description) enclosed in quotation marks. The `name` token may be omitted.

- `disable` indicates that the rule is temporarily disabled and is not applied. If the `disable` token is omitted, then the rule is enabled.

Condition

A condition specifies the parameters a packet should have to be processed by this rule. A condition may contain the following tokens:

- `proto <protocol>` specifies the transport protocol the packet should belong to. The supported protocols include `tcp`, `udp` and `icmp`. You may also specify numeric identifiers of any protocols. If you need to process packets of different protocols using one rule, you may specify those protocols, separating them by commas.

  You may specify the `any` keyword instead of a protocol, which means all protocols.

  In the `[broadcast]` section, you may specify only the `udp` and `icmp` values for the `proto` token.

- `type <type>` indicates the type of an ICMP message. This token can be specified only in the ICMP protocol condition (`proto icmp`). It can't be used for other protocols and when all the protocols are specified (`proto any`). For the `type` token, you can specify only one message type, which is a number from 0 to 255.

  If the `type` token is not specified for the ICMP protocol condition, it is considered that the condition is applied to ICMP messages of any type.

Note: It is required to specify the `type` token if the `code` token is specified in the condition for the ICMP protocol (see below).

- `code <code>` indicates the code of an ICMP message. This token can be specified only in the ICMP protocol condition (`proto icmp`). It can't be used for other protocols and when all the protocols are specified (`proto any`). For the `code` token, you can specify only one message code, which is a number from 0 to 255.

  If the `code` token is not specified for the ICMP protocol condition, it is considered that the condition is applied to ICMP messages with any codes.

- `from <address list>` specifies conditions for the source address and port. If both the address and port are specified, they should be separated by a colon, for example, `192.168.201.1:22`. If a port is not specified, a colon is not used after the address. In this case, the condition is applied to all the ports.

Note: Port numbers can't be specified in the `icmp` protocol condition (`proto icmp`) and in case all the protocols are specified (`proto any`).

You may specify not only a single address, but also an address range or an address mask, for example, `192.168.1.1-192.168.1.10:22` or `192.168.201.0/24:22`. You may also specify a port range, for example, `192.168.201.0/24:1024-65535`. If several conditions for the address and port are specified, they are separated by a comma: `192.168.1.1-192.168.1.10:22,172.16.1.0.24:25`.

You may combine addresses, address ranges and masks, as well as ports and port ranges into groups. The groups are enclosed in brackets, and the components are separated by commas. You may link several address ranges or masks to a single port range, without repeating it several times. For example, the entry

```
(192.168.201.0/24,172.16.1.0/24):1024-65535
```

means "all the packets from all the addresses in the 192.168.201.0/24 and 172.16.1.0/24 networks with a 1024 to 65535 source ports." Even more complex notation forms are possible, with simultaneous grouping of addresses and ports and with listing of such groups. For example, the entry

```
(192.168.201.0/24,172.16.1.0/24):(22,25,6660-6667),10.0.0.0/8:1024-
65535
```

means "all the packets from all the addresses in the 192.168.201.0/24 and 172.16.1.0/24 networks with a 22, or 25, or 6660 to 6667 source port, and the packets from the addresses in the 10.0.0.0/8 network with a 1024 to 65535 source port."

- `to <address list>` specifies conditions for the packet's destination address and port. The syntax of this token is the same as the one of the `from` token.

  In the `[broadcast]` section, for the `to` token, only the following addresses may be specified:

  - `broadcast` means the address 255.255.255.255;
  - `directed-broadcast` means broadcast addresses of all the subnets connected to the network interfaces of the computer. When the rule is loaded into the driver, this value is replaced with a list of corresponding broadcast addresses;
  - broadcast addresses of the subnetworks connected to the network interfaces of the computer. Specifying a certain broadcast address affects directed broadcasts sent in the corresponding subnet.

- `in` or `out` specifies the direction of connection establishment. Specifying these conditions, remember that they don't determine the direction of a packet. They specify only the direction of establishing connection. ViPNet Coordinator HW/VA traces relations between packets and connections established. It allows or blocks packets according to the rules. For example, let the `[local]` section have the following condition:

  ```
  proto tcp from 192.168.1.1 to anyip out
  ```

  This condition will apply to all the local packets related to the connections initiated from the 192.168.1.1 address, in other words, the packets sent from the 192.168.1.1 address to remote hosts' addresses, and responses sent to the 192.168.1.1 address. However, if a remote host tries to establish connection to 192.168.1.1, the condition described above will not apply to the packets related to this connection.

Connections over the TCP protocol are always traced. The UDP protocol does not have the concept of "connection", but still an attempt is made to trace a virtual connection established between applications mostly using UDP. For example, the condition is as follows:

```
proto udp from 192.168.1.1 to anyip:53 out
```

After a host with the IP address 192.168.1.1 sends a UDP packet to the `53` port of a remote host, a virtual connection with it is considered to be established. Then, if a response from the remote host arrives shortly at the same port that was used for sending the packet, the condition described above will also apply to this response packet, as the packet belongs to the established virtual connection. Virtual connections are considered broken if they have no traffic during the time period defined for the current protocol.

When applications exchange UDP packets using different port numbers for sending and receiving packets, there is no way to trace a virtual connection. In such cases, you should regard the `in` and `out` tokens as corresponding to the packet's direction.

The `proto`, `from`, and `to` tokens must be always specified in a condition. If any of these parameters is not important, you should specify `any` (in the `proto` token) or `anyip` (in the `from` and `to` tokens). The direction may be omitted; this means that the condition should affect connections established in both directions. The `type` and `code` tokens may be omitted for the ICMP protocol.

With tokens `from` and `to`, you may specify the following keywords instead of addresses and their ranges:

- `anyip` means all the addresses (in other words, the range from 0.0.0.0 to 255.255.255.255);
- `broadcast` means the address 255.255.255.255.
- `internet` stands for the `anyip` range with the exclusion of the three private IP address ranges:
    - `10.0.0.0-10.255.255.255`
    - `17.16.0.0-172.31.255.255`
    - `192.168.0.0-192.168.255.255`

Here are some examples of complete conditions:

```
proto any from anyip to 192.168.201.1:22 in
proto tcp,udp from anyip:53 to 192.168.0.0/16,172.16.1.0/24
proto tcp from 10.0.0.1 to (192.168.0.0/16,172.16.1.0/24):(22,25) out
proto icmp type 8 code 0 from anyip to anyip
```

Peculiarities of Specifying a Condition in Filtering Rules for Tunneled Packets

For the tunneled traffic filtering rules in the `[tunnel]` section, a condition is specified taking into account the following peculiarities:

- In one of the `from` and `to` tokens, you should specify the list of tunneled hosts' addresses. In the other one, you should specify a list of identifiers of the protected hosts that exchange traffic with the tunneled hosts.
- The list of identifiers is generated according to the same rules as the list of addresses, for example, `0x10e10000/16:(22,25)`. The `anyid` keyword is used to specify all identifiers.
- You can use the word `any` instead of the `anyid` and `anyip` keywords (herein, you can specify the ports). If the `any` keyword is specified in either the `from` or `to` token, then the word `any` should be specified in the other token as well. Otherwise, the condition returns an error. Such a rule combines two rules. The first rule: in the `from` token, the `anyip` word is specified, and in the `to` token, the `anyid` word is specified. The second rule: in the `from` token, the `anyid` word is specified, and in the `to` token, the `anyip` word is specified.

Action

An action defines what should be done to an IP packet whose parameters match the rule's condition (see below). An action is specified by one of the following two tokens:

- `pass` means that the packet should be allowed.
- `drop` means that the packet should be blocked.

Schedule

The schedule allows you to set time intervals, within which the rule is applied. When the schedule is omitted, the rule is applied permanently. The schedule is described by a single `time` token with the parameter consisting of several components, separated by commas.

```
time <schedule mode>,<schedule type>,<time interval>
```

Schedule mode can take one of the following values:

- `on` means that the rule is applied during the time intervals specified in the schedule, and is not applied at any other time;

- `off` means that the rule is not applied during the specified time intervals, and is applied at any other time (opposite to the `on` value);
- `disable` means that the schedule is disabled and the rule is always applied as if there were no schedule.

Schedule type can take one of the following values:

- `daily` means a schedule for every day. Using this schedule type, you can specify only one time interval, during which the rule is applied (if the schedule mode is `on`) or not applied (if the schedule mode is `off`).
- `weekly` means a schedule for a week. Using this schedule type, you can specify a time interval for each day of the week individually. During these intervals, the rule is applied (if the schedule mode is `on`) or not applied (if the schedule mode is `off`). This schedule type allows you to create different schedules for weekends, for example.

Depending on the schedule type, the **Time interval** can be specified as follows:

- If the schedule type is `daily`, a single interval is specified as `hh:mm-HH:MM`, where hh:mm means hours and minutes at the beginning of the interval, and HH:MM means those at its end. The interval start time is included into the schedule, but the finish time is not. Minutes can take values from 0 to 59 and hours — from 0 to 24. If the number of hours equals to 24, then the number of minutes can be equal only to 00, and this time means 12:00 a.m. of the next day.
- If the schedule type is `weekly`, then you can specify several time intervals. Specify the first three letters of an English name of the day of the week (mon, tue, wed, thu, fri, sat, sun), then put the equals sign (=) and a time interval for this day of the week in the same format as for a daily schedule, for example: `mon=9:00-18:00`. Schedules for different days of the week are separated by commas, for example: `mon=9:00-18:00,tue=10:00-18:00`. You may also specify one and the same time interval for several days of the week. In this case, they are written before the equals sign and are separated by colons, for example: `sat:sun=00:00-24:00`.

  You are free to omit some days of the week in the schedule. On the days omitted, the rule will be applied if the schedule mode is `off`, and will not be applied if the schedule mode is `on`, in other words, the omitted days are regarded as intervals not specified in the schedule.

Here are some examples of complete schedules:

```
time off,daily,9:00-18:00
time on,weekly,mon:tue:wed:thu:fri=9:00-18:00,sat:sun=00:00-24:00
```

Default Filtering Rules for Unencrypted IP Packets

The firewall configuration file is generated automatically, when ViPNet Coordinator HW/VA is installed or manually configured. The generated file consists of mandatory sections with preset rules. Some of the rules are disabled. Commenting out the corresponding lines is used to disable them, instead of the `disable` token. To enable a rule, you need to delete the comment character from the line.

The administrator may change the rules set. However, you can always return to the default rules settings in any of the mandatory sections. To do that, you need to delete the section from the firewall configuration file. At the next control daemon's startup, the omitted section will be automatically added to the firewall configuration file with default rules.

By default, the `[local]` section contains the following rules:

```
rule= proto udp from anyip:67 to anyip:68 pass
rule= proto udp from anyip:68 to anyip:67 pass
# rule= proto udp from anyip:138 to anyip:138 pass
rule= proto udp from anyip to anyip:53 pass
rule= proto udp from anyip to anyip:123 pass
```

The first two rules allow DHCP service packets (ports 67 and 68) used for IP addresses dynamic allocation. These rules are enabled.

The third rule allows NetBIOS service packets (netbios-dgm, port 138) used for data transfer between computers if NetBIOS names are used in the local network. This rule is disabled (commented out).

The fourth rule allows outgoing packets addressed to port 53 of DNS servers (see DNS server on page 78), while the fifth rule allows outgoing packets addressed to port 123 of NTP servers. These rules are enabled.

In the `[broadcast]` section, the following rules exist by default:

```
rule= proto udp from anyip:67 to anyip:68 pass
rule= proto udp from anyip:68 to anyip:67 pass
# rule= proto udp from anyip:138 to anyip:138 pass
# rule= proto udp from anyip:137 to anyip:137 pass
```

The first three rules are the same as those in the `[local]` section. The fourth rule allows NetBIOS service packets (netbios-ns, port 137) used to register and check NetBIOS names of the local network computers. This rule is disabled (commented out).

By default, the `[tunnel]` section contains the following rule:

```
rule= proto any from any to any pass
```

This rule is enabled and allows traffic between all the tunneled hosts and protected hosts connected to this ViPNet Coordinator HW/VA host. This entry is equivalent to setting the following two rules for tunneled hosts:

```
rule= proto any from anyid to anyip pass
rule= proto any from anyip to anyid pass
```

Configuring Address Translation Rules

ViPNet Coordinator HW/VA supports network address translation (NAT), in other words, changing the packet source or destination IP address according to certain algorithms. Two types of address translation are supported:

- **Source IP address translation**, also called “masquerading” or “dynamic translation”. Such address translation is used when you need to provide users having private IP addresses with access to the Internet. In this case, when packets from private users are passing through the ViPNet Coordinator HW/VA host, the source IP address is replaced with the external (public) ViPNet Coordinator HW/VA's address. When response packets arrive, the destination IP address is replaced back with the private address, and the packet is delivered to the private network as it is.

  Note: A source address can be substituted only with the IP addresses specified on the interfaces.

- **Destination IP address translation**, also called “port forwarding” or “static translation”, is used when you need to provide access from the Internet to a host located in a local network. In this case, all the packets coming from the Internet to a certain public address's port of the ViPNet Coordinator HW/VA host are forwarded to the specified address in the local network by means of substituting their destination IP addresses. The response packets from the local network host have their source IP addresses substituted.

You can set NAT rules in the `[nat]` section of the firewall configuration file.

Address Translation Rules Syntax

The control component should be specified at the beginning of the rule. Other components may follow in any order.

Note: In address translation rules, as opposed to filtering rules, there is no scheduling.

The **control component** is completely identical to the one described for filtering rules.

The **action** is defined by the `change` token with the parameter specifying addresses to be substituted and addresses to be used. Depending on the translation type, the action may be as follows:

- For translating a source IP address: `change src= {<address>:dynamic | eth<X> | modem}`

  The source IP address will be replaced according to the expression that you specify:

  - The `<address>` string defines that the packet's source address will be replaced with a particular IP address. This can be the public IP address of your ViPNet Coordinator HW/VA, when it is static, or an alias of this interface.
  - The `eth<X>` string defines that the packet's source IP address will be replaced with the current IP address of the interface X.

    For example, if the public IP address of your ViPNet Coordinator HW/VA is dynamic, you do not need to update the NAT rule each time the IP address changes. The `ethX` string in the rule updates automatically after an IP address change.
  - The `modem` string defines that the packet's source IP address will be replaced with the IP address assigned to the host's 3G/LTE modem.

  Examples:

  ```
  change src=194.87.0.8:dynamic
  ```

  ```
  change src=eth1
  ```

- For translating a destination IP address: `change dst= <address>:<port>`

  where `<address>` and `<port>` are the address and the port of the computer on your LAN, to which the packet will be forwarded. Example:

  ```
  change dst=192.168.201.1:8080
  ```

The **condition** for the address translation rules has almost the same syntax as for filtering rules, but with some specific characteristics:

- To translate destination addresses (dynamic address translation) for the `proto` token, specify `any`, and for the `to` token, specify `anyip`.

- For source address translation, the `from` token specifies a set of local network addresses to be translated, and you may specify only addresses, ranges, masks and their lists in the `from` token. You should not specify ports or port ranges there.

- For destination address translation (static address translation), the `from` token should take the `anyip` value, and the `to` token should specify the coordinator's external address and port, where the packets will arrive for forwarding. In this case, you may specify only the address or the address list, as well as the port or the port range in the `to` token. You can't specify address ranges, address masks, and port ranges.

Here are some examples of address translation rules:

- For source address translation:

  ```
  rule= num 10 change src=194.87.0.8:dynamic proto any from 192.168.201.0/24
  to anyip
  ```

- For destination address translation:

-
  ```
  rule= num 100 change dst=10.0.0.7:8080 proto tcp from anyip to
  194.87.0.8:80
  ```

Interaction between Filtering and Address Translation Rules

The packets subject to address translation are also processed by filtering rules specified in the `[local]` and `[forward]` sections. When the parameters of a packet with translated addresses match the condition of a filtering rule, the following principle is applied: the source IP address is taken from the packet before translation, and the destination IP address is taken from the packet after translation. It is these addresses that are checked to match the rules' conditions.

For example, there is a ViPNet Coordinator HW/VA host with the 10.0.1.0/24 local network and 194.87.0.8 public IP address. You need to provide the local network users with access to the Internet. To do that, you should set the following rule in the `[nat]` section:

```
rule= num 10 change src=194.87.0.8:dynamic proto any from 10.0.1.0/24 to
anyip
```

You should also specify an allowing rule in the `[forward]` section:

```
rule= num 100 pass proto any from 10.0.1.0/24 to anyip
```

In this example, if you need to block TCP connections that have been established by local network users to the external address 194.226.82.50, add the following rule to the `[forward]` section:

```
rule= num 90 drop proto tcp from 10.0.1.0/24 to 194.226.82.50
```

You can see that the `from` token contains the address of the local packet source before address translation, and the `to` token contains the address of the remote recipient after address translation (in this case, it is not changed during translation).

The same principle is applied to destination address translation. Assume there are packets arriving at port 80 of the ViPNet Coordinator HW/VA's external address. You need to direct them to the address 10.0.1.1 and port 8080 in a LAN. To do that, you should configure the following rule in the `[nat]` section:

```
rule= num 10 change dst=10.0.1.1:8080 proto tcp from anyip to 194.87.0.8:80
```

You should also configure an allowing rule in the `[forward]` section:

```
rule= num 100 pass proto tcp from anyip to 10.0.1.1:8080
```

Now, if you need to block inbound connections from the external address 194.226.82.50 to this LAN computer, add the following rule to the `[forward]` section:

```
rule= num 90 drop proto tcp from 194.226.82.50 to 10.0.1.1:8080
```

IP Packets Log

Configuring IP Packets Logging

Information about the events related to processing IP packets on ViPNet Coordinator HW/VA's network interfaces is written to the IP packets log. You can specify entry detailing and maximum log size for each of your ViPNet Coordinator HW/VA's network interfaces.

To configure logging parameters for the IP packets passing through a certain network interface:

1. In the command line interface, switch to the administrator mode by executing the `enable` command.

2. To start editing the configuration file of a network interface:

   ```
   iplir config <network interface name>
   ```

   Note: To view a list of network interfaces, execute the `inet show interface` command.

3. In the `[db]` section, specify the required values of the following parameters:

   - `maxsize` is the maximum size of the log in megabytes.

     As soon as the log reaches the maximum size, the oldest entries get overwritten with newer ones.

     If this parameter's value is null, logging is disabled on this network interface.

   - `timediff` is a time period (in seconds), within which events with similar characteristics are united into one log entry. The default value is 60 seconds.

     For example, information about allowed encrypted IP packets with the same source address and port and the same destination address and port will be united within one log entry.

     If this parameter's value is null, each IP packet will have a separate log entry.

   - `registerall` allows you to enable and disable IP packets logging. This parameter may take the following values:

     - `on` — log any IP packet.
     - `off` (the default value) — log only blocked IP packets and the events related to changing ViPNet hosts' IP addresses.

- `registerbroadcast` allows you to enable or disable broadcast IP packets logging. This parameter may take the following values:
  - `on` — log broadcast IP packets.
  - `off` (the default value) — don't log broadcast IP packets.
- `registertcpserverport` allows you to enable or disable source port logging for TCP packets. This parameter may take the following values:
  - `on` — don't log a source port. In this case, log entries will be sorted by a destination port.
  - `off` (the default entry) — log a source port.

4. To save the configuration file, press **Ctrl+O**, then press **Enter**.
5. To close the file, press **Ctrl+X**.

IP Packets Log

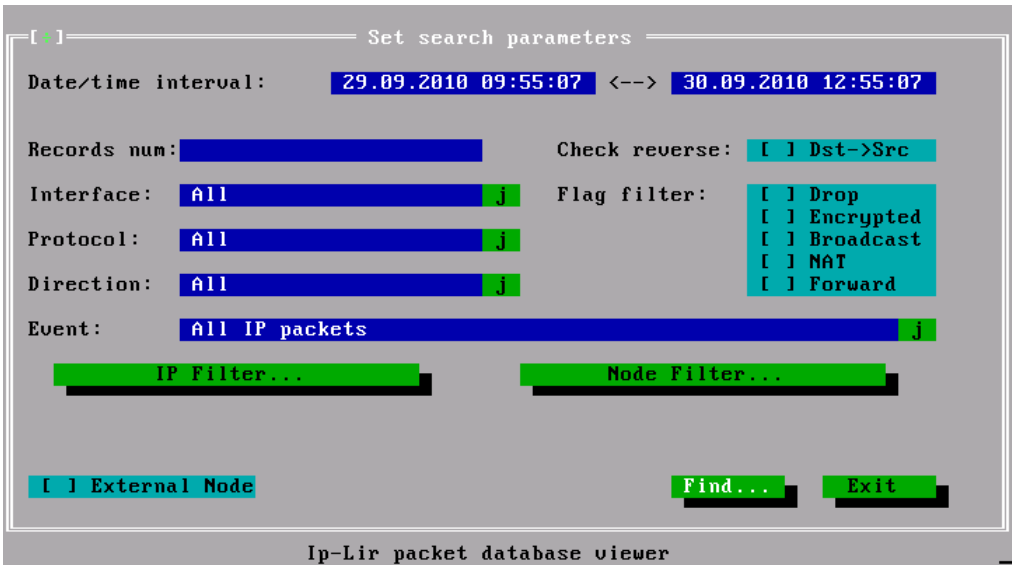
The IP packets log contains information on encrypted and unencrypted IP packets that have been processed by the ViPNet driver on the ViPNet Coordinator HW/VA appliance. The data is collected on all the appliance's network interfaces.

Take into account that the information on the packets passing through ViPNet Coordinator HW/VA's network interfaces is logged, and not the information on connections. Thus, all allowed forward packets are displayed in the log twice: first, on the interface they arrive at, then on the interface they are forwarded from. All allowed local packets are displayed once: on the interface they arrive at and leave from. All blocked packets are displayed in the log once: on the interface they arrive at and are blocked on. This is the same both for encrypted and unencrypted packets.

To view the IP packets log, use the `iplir view` command. You can filter the entries by the following search criteria:

- date interval;
- the network interface, a particular packet was processed by;
- IP protocol;
- packet direction — incoming or outgoing;
- event type;
- a range of IP addresses or one IP address of the packet's source or destination host;
- a range of local port numbers or one local port number for TCP, UDP;
- a range of remote port numbers or one remote port number for TCP, UDP;
- the ViPNet host's name (the name of the packet's source and (or) destination host).

Upon you have executed the `iplir view` command, the following window will be displayed:



Set search parameters

Date/time interval: 29.09.2010 09:55:07 <--> 30.09.2010 12:55:07

Records num:

Check reverse: [ ] Dst->Src

Interface: All

Flag filter: [ ] Drop [ ] Encrypted [ ] Broadcast [ ] NAT [ ] Forward

Protocol: All

Direction: All

Event: All IP packets

[ ] External Node

Ip-Lir packet database viewer

Figure 5: Defining search criteria in the IP packets log

By default, you are offered to view the log of your host. If you need to view the log of another ViPNet host, enable the **External Node** option. You will be prompted to enter the ViPNet network administrator's password. If you enter the correct password, on the right from the **External Node** option, the **Select** button will be available. Click it and select the required host from the displayed list of ViPNet hosts. Connection to a remote host will be established successfully if it is accessible on the TCP/IP level and the control daemon is running on it. If the connection fails to be established, you are informed about that, and the control daemon stops working.

You can specify the following search criteria for the IP packets log:

- **Date/time interval** is the date and time interval in the DD.MM.YYYY HH:MM:SS format. Within this interval, the search for IP packet registration entries is performed.
- **Records num** is the number of entries to display at once.
- **Interface** is the network interface (selected from the list of available interfaces).

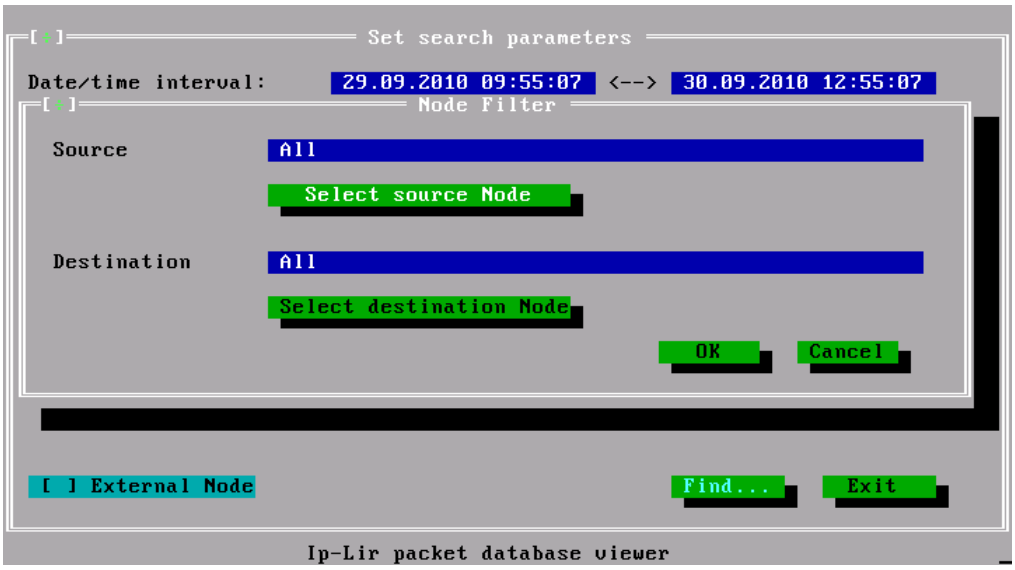
  In the current context, an interface is considered available if it has an IP packets log. Packets will be searched for in the chosen interface's log. You can specify the interface's name or set the parameter to **All** to search for IP packets in all the logs of all available network interfaces.
- **Protocol** is the name of the protocol to search only for the packets belonging to this IP protocol among all the packets.

You can specify either the protocol name or set the value to **All** to search for packets belonging to any protocol.

- **Direction** is the packet's direction; it can take the following values:
  - **All**, meaning incoming and outgoing packets;
  - **Incoming**, meaning incoming packets;
  - **Outgoing**, meaning outgoing packets.
- **Check reverse** is the flag indicating that the log includes response packets from a destination host to a source host.

  It makes sense to use this flag when you specify a particular IP address (**IP Filter**) or host (**Node Filter**) as the packet's source and (or) destination.
- **Flag filter** allows you to search for the packets having one or more flags specified below:
  - **Drop** means blocked packets;
  - **Encrypted** means encrypted packets;
  - **Broadcast** means broadcast packets;
  - **NAT** means translated packets;
  - **Forward** means forward packets.
- **Event** means the event, the packets are associated with.

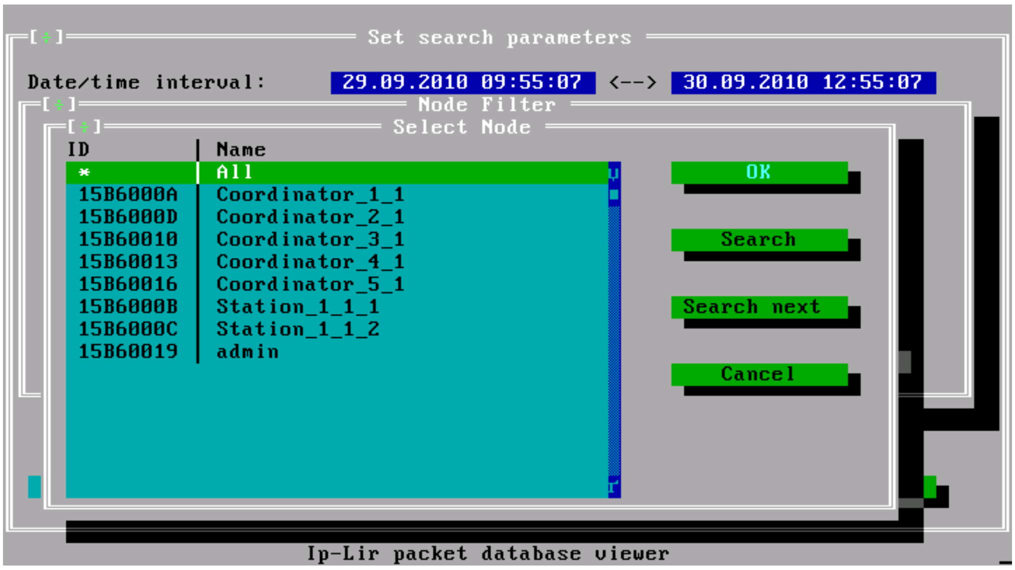
  You can select an event from the list. By default, the search is performed among all event types.
- **IP Filter** displays a window where you can specify the following query parameters:
  - **Source IP address** — **All** defines a range or a single value for the allowed source IP address of the packet;
  - **Destination IP address** — **All** defines a range or a single value for the allowed destination IP address of the packet;
  - **Source port** defines a range or a single value for the allowed source port number (0–65535) for TCP and UDP protocols.
  - **Destination port** defines a range or a single value for the allowed destination port number (0–65535) for TCP and UDP protocols.
- **Node Filter** displays a window, where you can narrow your search by specifying the source and (or) destination host: **Source** and (or) **Destination**.



```
Set search parameters
Date/time interval:   29.09.2010 09:55:07 <--> 30.09.2010 12:55:07
Node Filter
Source        All
Destination   All
[ ] External Node
Ip-Lir packet database viewer
```

Figure 6: Specifying the source and (or) destination host to narrow the search

To choose the source or destination host, use the corresponding buttons: **Select source Node** or **Select destination Node**. A searchable list of ViPNet hosts will be displayed in a new window.



```
Set search parameters
Date/time interval:   29.09.2010 09:55:07 <--> 30.09.2010 12:55:07
Node Filter
Select Node
ID        | Name
*         | All
15B6000A  | Coordinator_1_1
15B6000D  | Coordinator_2_1
15B60010  | Coordinator_3_1
15B60013  | Coordinator_4_1
15B60016  | Coordinator_5_1
15B6000B  | Station_1_1_1
15B6000C  | Station_1_1_2
15B60019  | admin
Ip-Lir packet database viewer
```

Figure 7: A list of ViPNet hosts to choose the IP packets' source or destination host from

This window displays the list of all ViPNet hosts linked with your host. The hosts are listed alphabetically. The hexadecimal ViPNet host identifier is displayed in the left column. The list also contains the **All** service element. Choose it to search through all the available ViPNet hosts.

To search for the required host, click **Search**. In a separate window, the search box will be displayed allowing you to type a string or select it manually from a drop-down list associated with the box. You can search by a host name and by a host id. The **Name** and **ID** columns will be scanned for matches with the search string.

Click **Search next** to find the next list item matching the search string quickly.

To display the log entry matching the specified search criteria, click the **Find** button at the bottom of the main window (see figure on page 72). To close the log, click **Exit**.

All the entries found in the IP packets log are displayed in the **View results** window (see figure on page 76). The entries are sorted by packets' registration time. The list is arranged as follows:

- Event registration date and time.
- The network interface the event was registered at.
- The direction of the registered packet: “<” — outgoing, “>” — incoming, and event flags combination:
  - **C** stands for an encrypted packet;
  - **B** stands for a broadcast packet;
  - **D** stands for a blocked packet;
  - **T** stands for a forward (routed)packet;
  - **R** means that the packet will be processed by NAT rules for a public network;
  - **N** means that the packet has been processed by NAT rules for a public network.
- The protocol identifier.
- The source IP address.
- The local port for the TCP and UDP protocols.
- The destination IP address.
- Another host's port for the TCP and UDP protocols.

The following information about the selected event is displayed at the bottom of the window:

- The name of the event assigned with the IP packet.
- The network interface.
- The protocol.

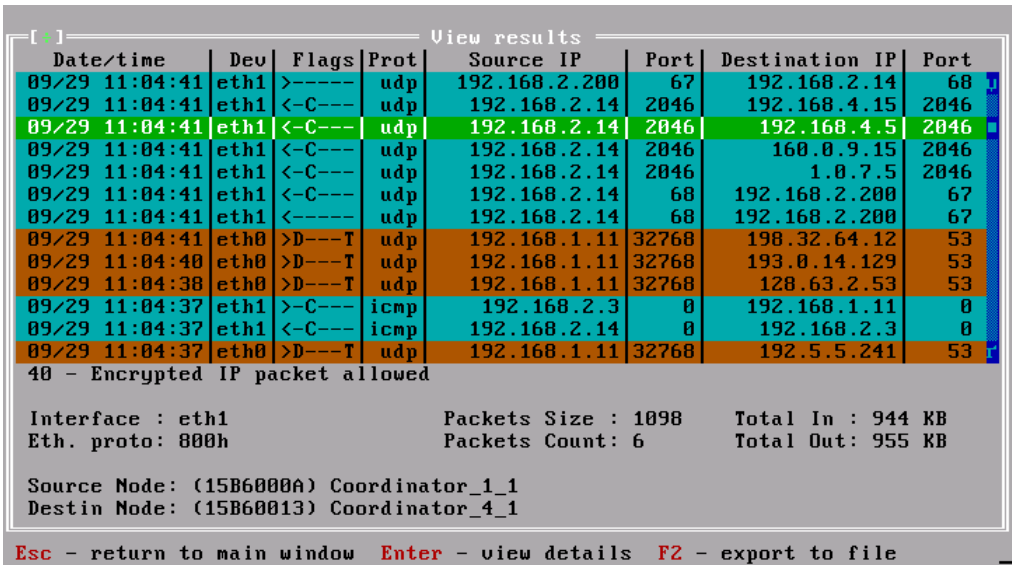
- The packet size.

  The total size of all packets related to this event (if the counter is greater than 1) is displayed. For encrypted packets, the total size includes all service headers required for the private network to operate correctly.

- The number of packets related to the event.

- The source host.

- The destination host.

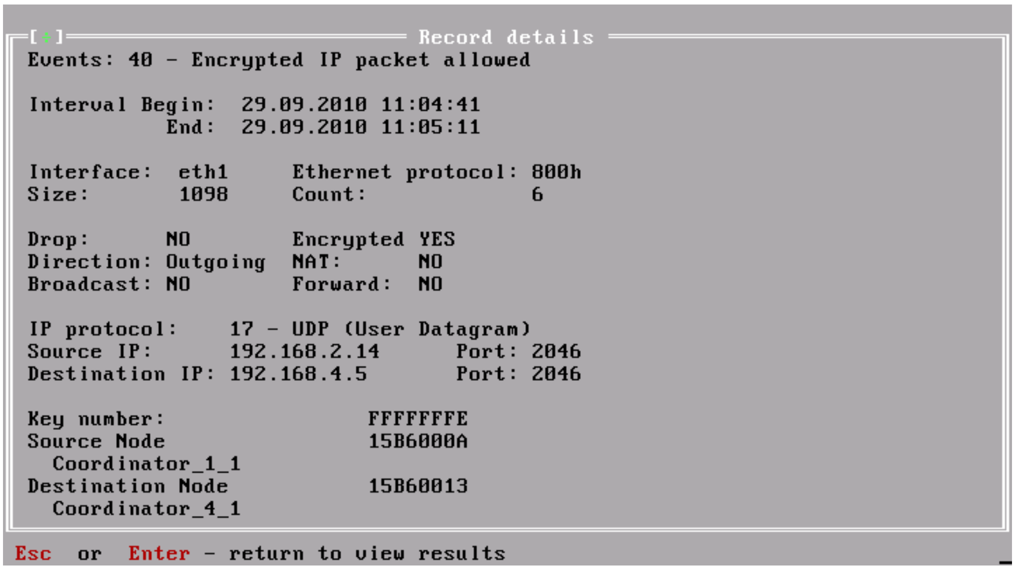


```
                                View results
   Date/time      | Dev| Flags |Prot|  Source IP   | Port| Destination IP| Port
09/29 11:04:41 |eth1|>-----  | udp| 192.168.2.200|   67|   192.168.2.14|   68
09/29 11:04:41 |eth1|<-C---  | udp|  192.168.2.14| 2046|   192.168.4.15| 2046
09/29 11:04:41 |eth1|<-C---  | udp|  192.168.2.14| 2046|    192.168.4.5| 2046
09/29 11:04:41 |eth1|<-C---  | udp|  192.168.2.14| 2046|     160.0.9.15| 2046
09/29 11:04:41 |eth1|<-C---  | udp|  192.168.2.14| 2046|        1.0.7.5| 2046
09/29 11:04:41 |eth1|<-C---  | udp|  192.168.2.14|   68|  192.168.2.200|   67
09/29 11:04:41 |eth1|<-----  | udp|  192.168.2.14|   68|  192.168.2.200|   67
09/29 11:04:41 |eth0|>D---T  | udp|  192.168.1.11|32768|   198.32.64.12|   53
09/29 11:04:40 |eth0|>D---T  | udp|  192.168.1.11|32768|   193.0.14.129|   53
09/29 11:04:38 |eth0|>D---T  | udp|  192.168.1.11|32768|    128.63.2.53|   53
09/29 11:04:37 |eth1|>-C---  |icmp|   192.168.2.3|    0|   192.168.1.11|    0
09/29 11:04:37 |eth1|<-C---  |icmp|  192.168.2.14|    0|    192.168.2.3|    0
09/29 11:04:37 |eth0|>D---T  | udp|  192.168.1.11|32768|    192.5.5.241|   53
40 - Encrypted IP packet allowed

Interface : eth1                 Packets Size : 1098     Total In : 944 KB
Eth. proto: 800h                 Packets Count: 6        Total Out: 955 KB

Source Node: (15B6000A) Coordinator_1_1
Destin Node: (15B60013) Coordinator_4_1

Esc - return to main window   Enter - view details   F2 - export to file
```

Figure 8: Displaying search results

Warning: In ViPNet Coordinator HW/VA, you can't export the log to a file.

To view details on any event selected in the list, press **Enter**.



```
[÷]                                Record details
Events: 40 - Encrypted IP packet allowed

Interval Begin:  29.09.2010 11:04:41
           End:  29.09.2010 11:05:11

Interface:  eth1     Ethernet protocol: 800h
Size:       1098     Count:             6

Drop:       NO       Encrypted  YES
Direction: Outgoing  NAT:       NO
Broadcast: NO        Forward:   NO

IP protocol:     17 - UDP (User Datagram)
Source IP:       192.168.2.14        Port: 2046
Destination IP:  192.168.4.5         Port: 2046

Key number:                 FFFFFFFE
Source Node                 15B6000A
  Coordinator_1_1
Destination Node            15B60013
  Coordinator_4_1
Esc  or  Enter - return to view results
```

Figure 9: Event details

The **Total In** and **Total Out** boxes display the total size of all the incoming and outgoing packets. You can find these boxes at the bottom of the window with the list of the found entries (see figure on page 76). The total size is calculated for all the packets returned by the query. If a returned entry does not contain information about the packet's size, it will not be counted for the total size calculation. In this case, an asterisk is displayed in the **Total In** and (or) **Total Out** box flagging that not all traffic data have been calculated. This asterisk flags that not all traffic data have been calculated. If none of the returned entries contain information about packet size, the **Total In** and (or) **Total Out** boxes will display **N/A** (without an asterisk).

When displaying the total size, the following measurement units are used:

- If the total size is less than 100 kilobytes, the size will be displayed in bytes, and the "B" suffix will be added to the size displayed in the box;
- If the total size is greater than 100 kilobytes, but less than 100 megabytes, the size will be displayed in kilobytes, and the "KB" suffix will be added to the size displayed in the box;
- If the total size is greater than 100 megabytes, the size will be displayed in megabytes, and the "MB" suffix is added to the size displayed in the box.

Glossary

C

Command line interface

A command shell you use to administer ViPNet Coordinator HW/VA with special commands.

D

DHCP (Dynamic Host Configuration Protocol)

A network protocol of the application layer that enables a server to automatically assign an IP address to a computer, as well as some other parameters necessary for it to work in a TCP/IP network. The parameters are a subnet mask, a gateway IP address, and DNS and WINS servers' IP addresses.

DHCP server

A server that automatically manages its clients' IP addresses and performs the required network setup.

DNS server

A server, containing part of the DNS database used to access computer names in the Internet domain. For example, `ns.domain.net`. As a rule, information about the domain is stored on two

DNS servers, called “primary” and “secondary” (backup is used to increase system fault-tolerance).

E

External network

A network that is separated from an internal network with a firewall.

F

Firewall

A device or software installed on a host on a network edge that checks all the incoming and outgoing IP traffic and decides whether it can be redirected to its destination point. In other words, a firewall is intended to prevent unauthorized access from one network to another. A firewall usually translates internal addresses into addresses accessible from the external network (uses NAT). In ViPNet networks, we distinguish three types of firewalls with NAT:

- ViPNet coordinator — a computer with the ViPNet Coordinator software installed that functions as a firewall and provides NAT for encrypted traffic.
- Static NAT — a firewall with static network address translation placed between a ViPNet host and an external network to provide IP traffic exchange between external hosts and your host via UDP with a specified port.
- Dynamic NAT — a firewall with dynamic network address translation placed between a ViPNet host and an external network. An additional coordinator is placed in the unprotected network to support the connections. This coordinator should have a public (routable) IP address. A firewall of this type is used when you connect to an external network via an xDSL modem functioning as a router, wireless devices, GPRS network, other ISPs assigning private IP addresses to your ViPNet hosts.

I

Internal network

A local network that is separated from an external network by a firewall.

IP traffic

The flow of data transferred on a network over the IP protocol.

IPsec protocol

A protocol, providing traffic protection with cryptographic methods.

N

Network addresses translation (NAT)

The technology that ensures translation of IP addresses and ports used on one network into addresses and ports used on another network.

Network interface

A device that connects a computer to a network. It is used in IP packets exchange. A network card, a modem and other similar devices can serve as a network interface.

Network protocol

A set of rules allowing two and more network devices to establish connection and exchange data.

NTP server

An exact time server, used to synchronize clocks on computers, workstations, servers, and other network devices. This server acts as an intermediary between the time standard and the network. It acquires time from the standard through a special channel (interface) and provides any host in the network with this information, thus ensuring synchronization of devices' clocks.

P

Protected host

A host with installed ViPNet software that can encrypt traffic in the network layer.

S

Subnet mask

A bitmask that can be used to separate the bits of the network identifier from the bits of the host identifier.

T

Tunneling

Encryption of unprotected hosts’ traffic, while the traffic is transferred via a public network.

U

Unprotected host

A host that exchanges unencrypted traffic with a ViPNet host.

V

ViPNet host

A network node with installed ViPNet software registered in ViPNet Administrator Network Control Center.

ViPNet host administrator's password

A password to enable the administrator mode on a ViPNet host. The administrator mode provides advanced options for the ViPNet program. A ViPNet host administrator’s password is created by the ViPNet network administrator in ViPNet Key and Certification Authority.

ViPNet network

A logical network that is created and maintained with ViPNet software and consists of ViPNet hosts.

A ViPNet network has a special addressing system, which provides for data exchange between its hosts. Each ViPNet network has its own unique number (host ID).

ViPNet Network Manager

A program that is a part of the ViPNet VPN software suite. It is intended to create, configure, and administer small and middle-sized ViPNet networks. ViPNet Network Manager also functions as certification and key authorities.

ViPNet user password

A personal user password for logging on to ViPNet software on a ViPNet host. Initially, this password is generated by a ViPNet network administrator in ViPNet Key and Certification Authority. A user can change his or her password on a ViPNet host.

Index