

WIZ VSP User's Manual

(V1.0)

©2009 WIZnet Co.Ltd. All Rights Reserved.

For more information, visit our website at www.wiznet.co.kr

Document History Information

Revision	Data	Description
Ver. 1.0	December, 2009	Release with WIZ VSP Launching

Table of Contents

1. Introduction.....	4
2. Installation	7
3. Program Interface.....	11
3.1 Main window	11
3.2 Main toolbar	12
3.3 Connections tree	13
3.4 Managing connections	15
4. How to use WIZ VSP	17
4.1 Sharing Serial Port Using "Server" Connection	17
4.1.1 Creating Connection.....	17
4.1.2 Editing connection	19
4.1.3 Connection settings.....	20
4.2 Connecting serial port using "client" connection.....	24
4.2.1 Creating connection.....	24
4.2.2 Editing connection	26
4.2.3 Connection settings	27
4.3 Sharing serial port using UDP	29
4.3.1 Creating connection.....	29
4.3.2 Editing port to UDP connection	31
4.3.3 Connection settings	32
4.4 Managing signal lines in connections	34
4.5 Proxy and security settings.....	36

1. Introduction

WIZ VSP is an advanced software-based solution that allows you to share more than 255 serial port devices over the network by easily turning your computer into a low-cost terminal server. Thus, any serial port device connected to your COM port could be accessed from anywhere in the world (via Internet or LAN) as if it was attached directly to the remote PC.

When the attached serial port device sends communication data, it is actually transmitted over TCP/IP network and back from the network to your serial device.

WIZ VSP provides 3 different types of connections as listed below :

- **Share serial port for incoming connections (Server)**

The Server connection waits for incoming client's connections and shares the real/virtual serial port with the network. The Server connection can be simultaneously supports multiple client and each connected client can exchange serial data with the real/virtual serial port. Client can be called whatever application that can initialize outgoing connection to TCP/IP server (SEC, Telnet, etc).
- **Connect serial port to remote host (Client)**

In this mode, the program creates a client connection which redirects the real /virtual serial port data to the remote server using the TCP/IP protocol. The Client connection does not require the presence of a Serial to Ethernet Connector (SEC) at the remote side. All that you have to do is specify the remote server's IP address (or network name) and the TCP port to connect to. Once the connection is established, all data sent from the remote serial port device, attached to the server, are delivered to local serial port where it can be further processed.
- **Share serial port using UDP**

You can redirect the serial data from local real or virtual serial port using UDP/IP underlying protocol. This type of connection does not require SEC presence at the remote side as well. In addition, you are able to broadcast all serial data to your local network.

WIZ VSP enables you to select data transmission protocol, which will be used for connection:

RAW data transmission algorithm or Telnet (RFC 2217) protocol.

To get more information about these connections' differences, please, consult Usage scenario page.

How does WIZ VSP work?

WIZ VSP emulates real serial ports and duplicates all their functionality. The Windows OS recognizes our virtual serial port as a real serial port. Therefore, your system is no longer restricted to 1 or 2 real serial ports, and COM port connections can be created without occupying any real serial ports.

In case you are setting up the "client" connection you should specify IP address and port number to access to, when you are establishing the terminal servers you will need to specify TCP/COM options and port number to listen to.

Main Features

- Share your serial ports and devices with others over the TCP/IP network
- Both of the TCP/IP and UDP/IP protocols are supported
- Manage port signal lines states in connections
- Verify the connection's integrity by using "send command to keep connection alive" feature
- Set pauses after which all received data is sent to remote end when using UDP as an under-layer protocol
- Easily connect to remote computer using TCP/IP Telnet protocol, with the RFC 2217 Telnet extensions for COM Port Control
- Connect to several remote computers simultaneously
- Share more than 255 serial ports simultaneously for remote connections
- Ability to send data of certain size and on receiving the specified char in UDP connection type
- High speed data exchange from/to virtual serial port (up to 256 kbits)
- Virtual serial ports work like real physical ports
- Ability to enable baudrate emulation for virtual ports
- Re-naming full port names in Device Manager
- Automatically restore connection within a customizable timeout on connection brakes

- Universal function Keep Connection Alive for Telnet and Raw protocols
- Small packet transmission speed can be increased by disabling Nagle algorithm
- Dynamically change COM port parameters without necessity to reopen or re-create it
- Ports could be automatically configured every time you boot your PC
- Map your real serial port to any other virtual or real port
- Customize advanced connection settings when you switch to the new Advanced mode
- Easily track all program's activity in the new "Activity log"
- Secure your data transmission over network by authorization system and traffic encryption
- Connection speed in serial port pairs is much more reliable and faster than in physical links
- User-friendly and convenient program interface
- Fully compatible with Windows NT4/2000/XP/2003/Vista/2008/64-bit platforms
- Fully compatible with HyperThreaded and multi-processor systems
- Works under virtual machine

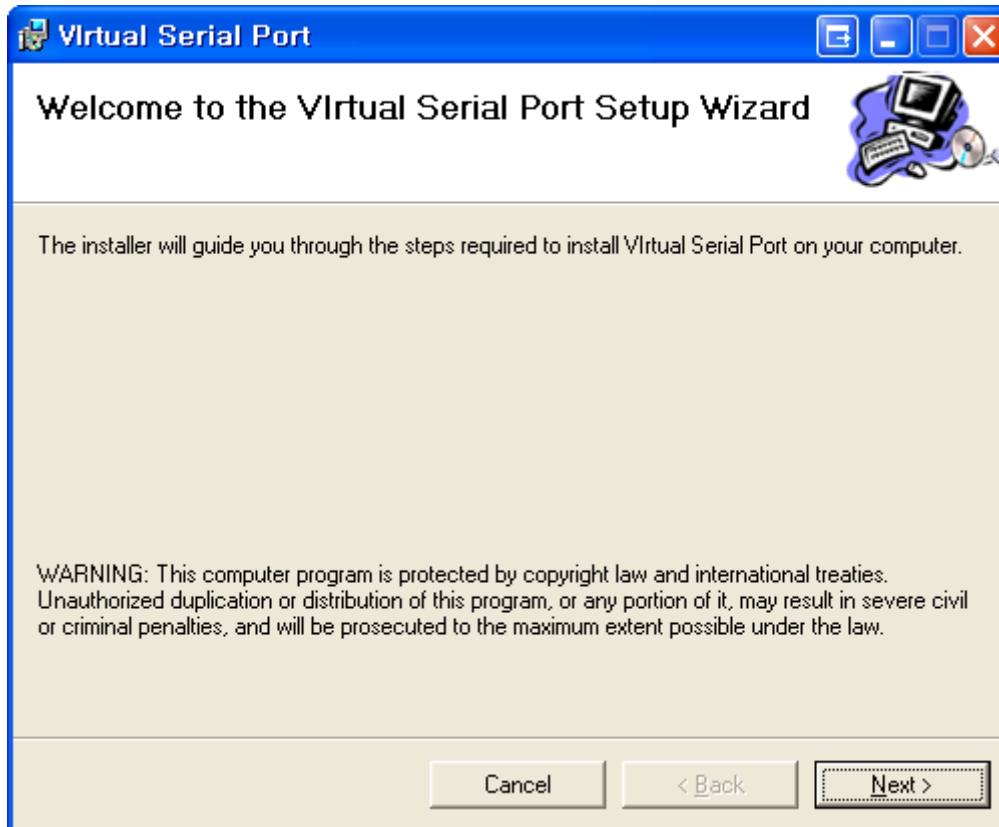
In general, WIZ VSP is a powerful tool for both business/industry and individuals, because millions of legacy serial devices without built-in network capabilities are commonly used today.

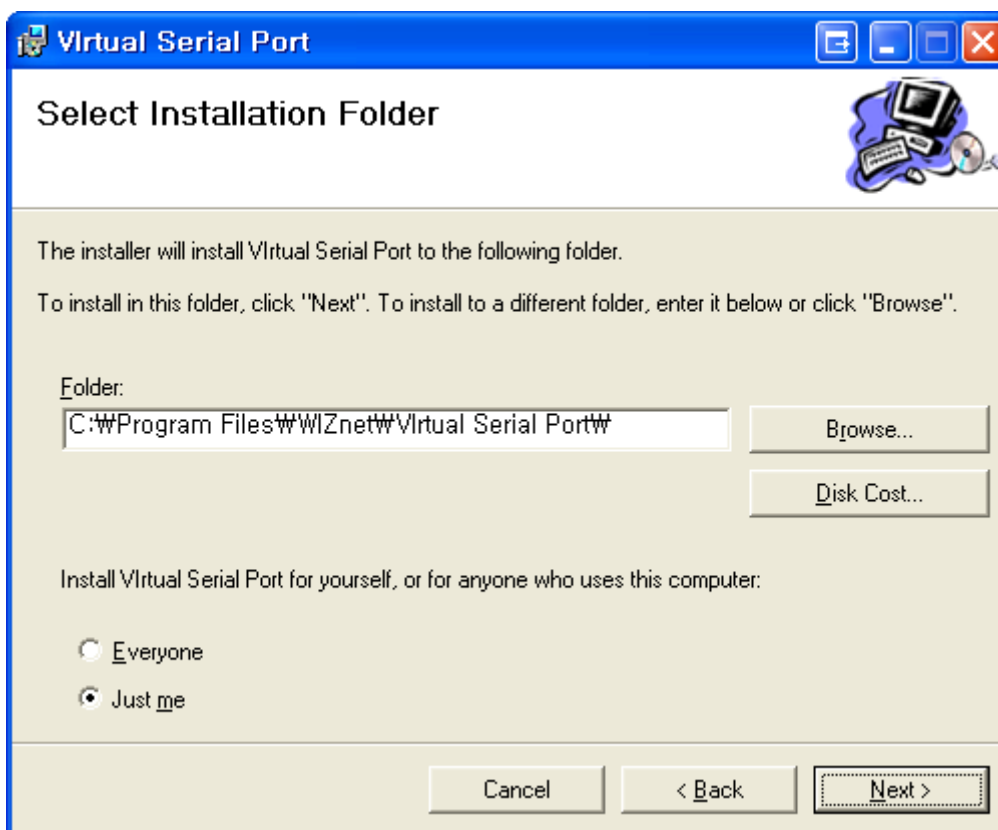
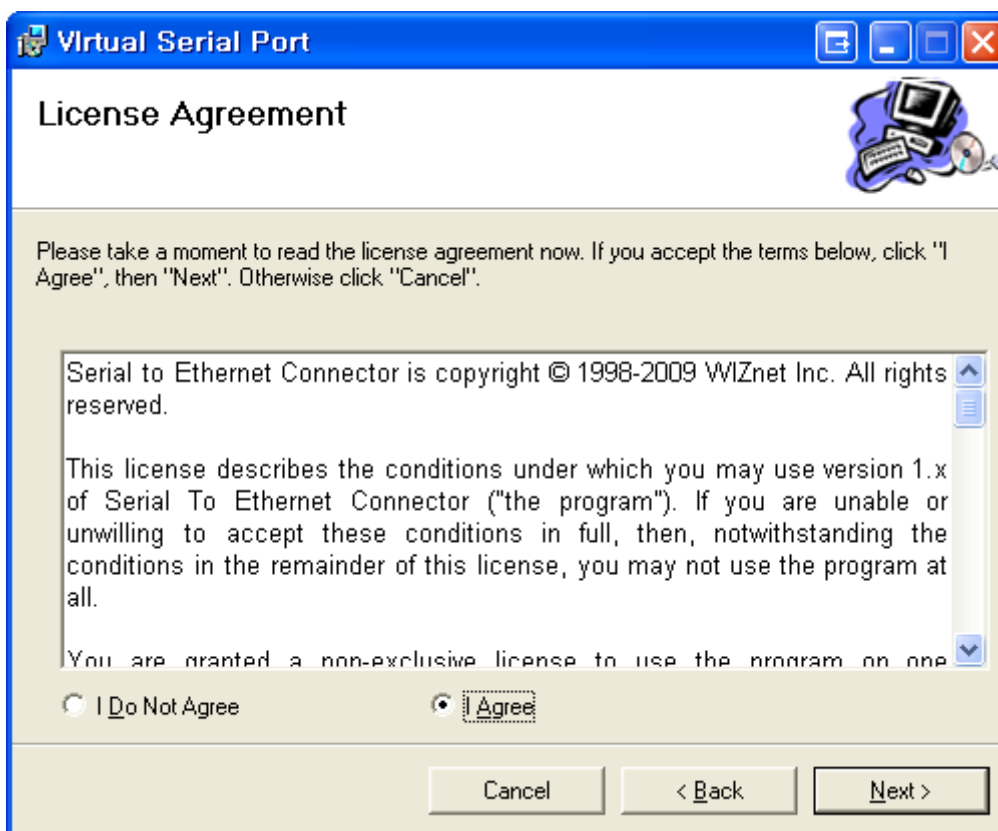
Platforms WIZ VSP is compatible with:

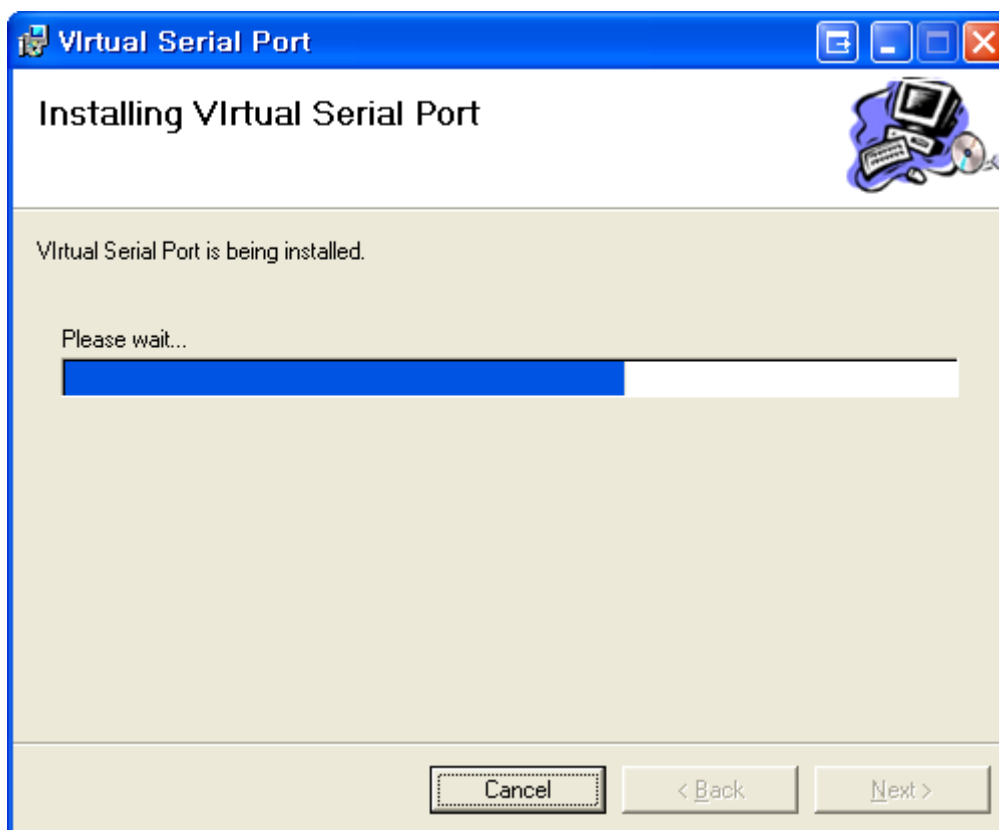
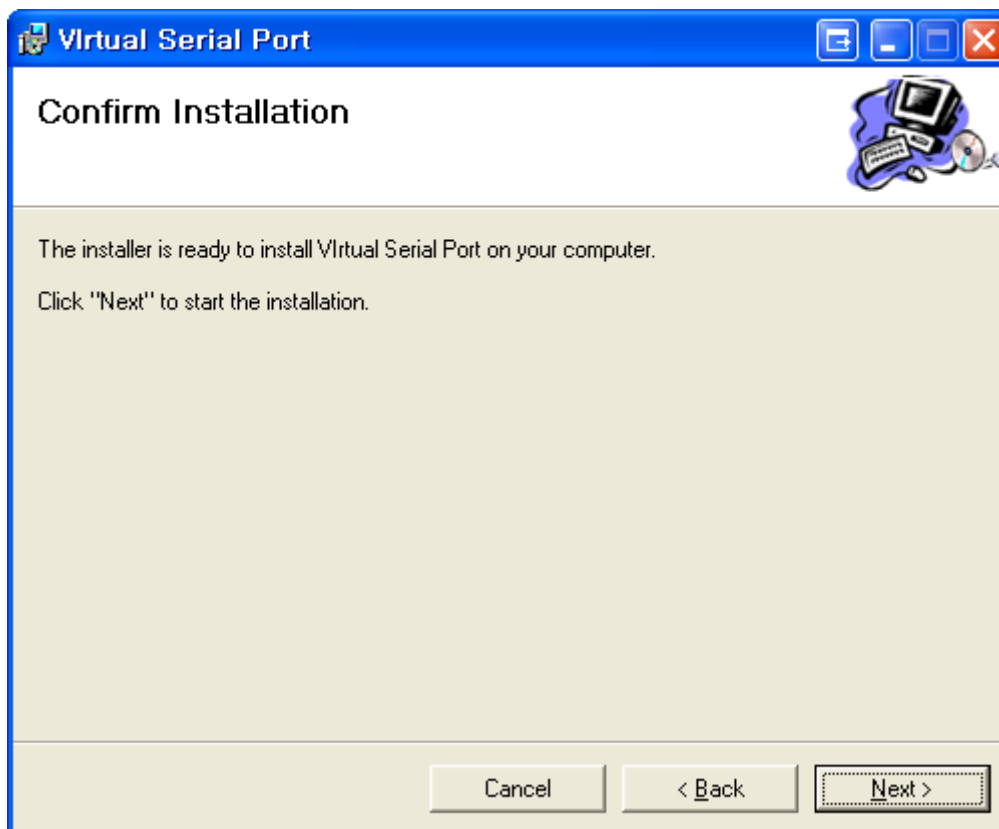
- Windows NT 4.x
- Windows 2000
- Windows 2003
- Windows XP
- Windows Vista
- Windows Server 2008
- All Windows 64-bit platforms

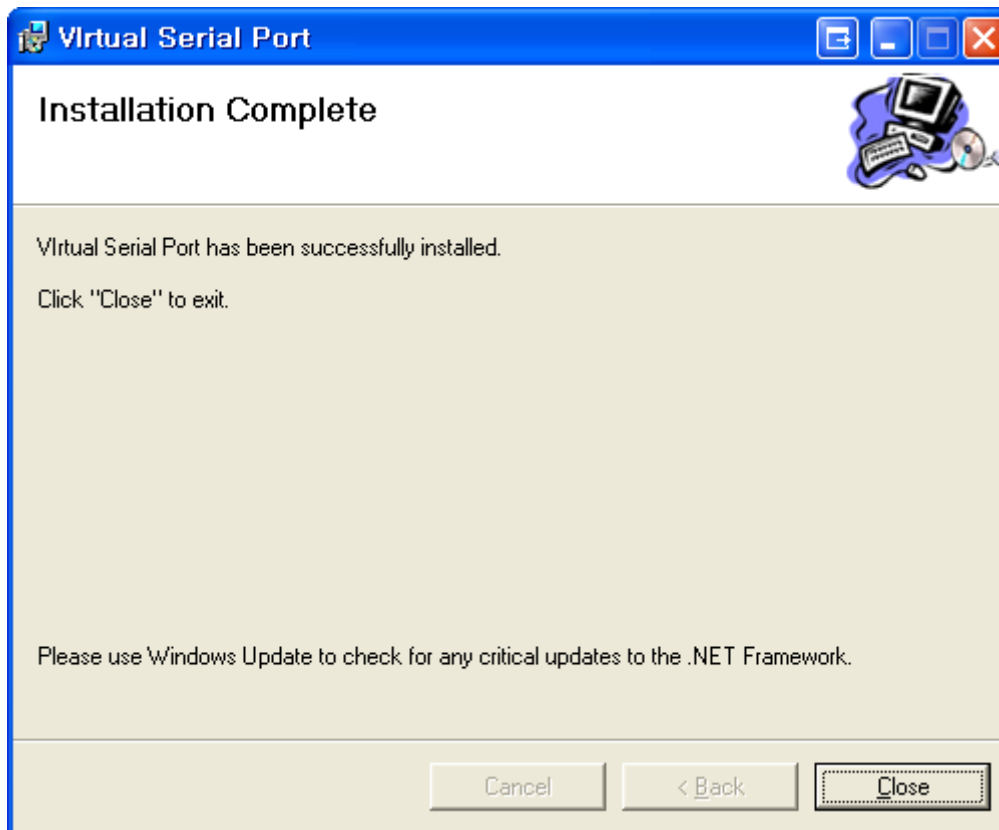
2. Installation

The WIZ VSP installation process is rather simple. All you need to do is follow the steps below:





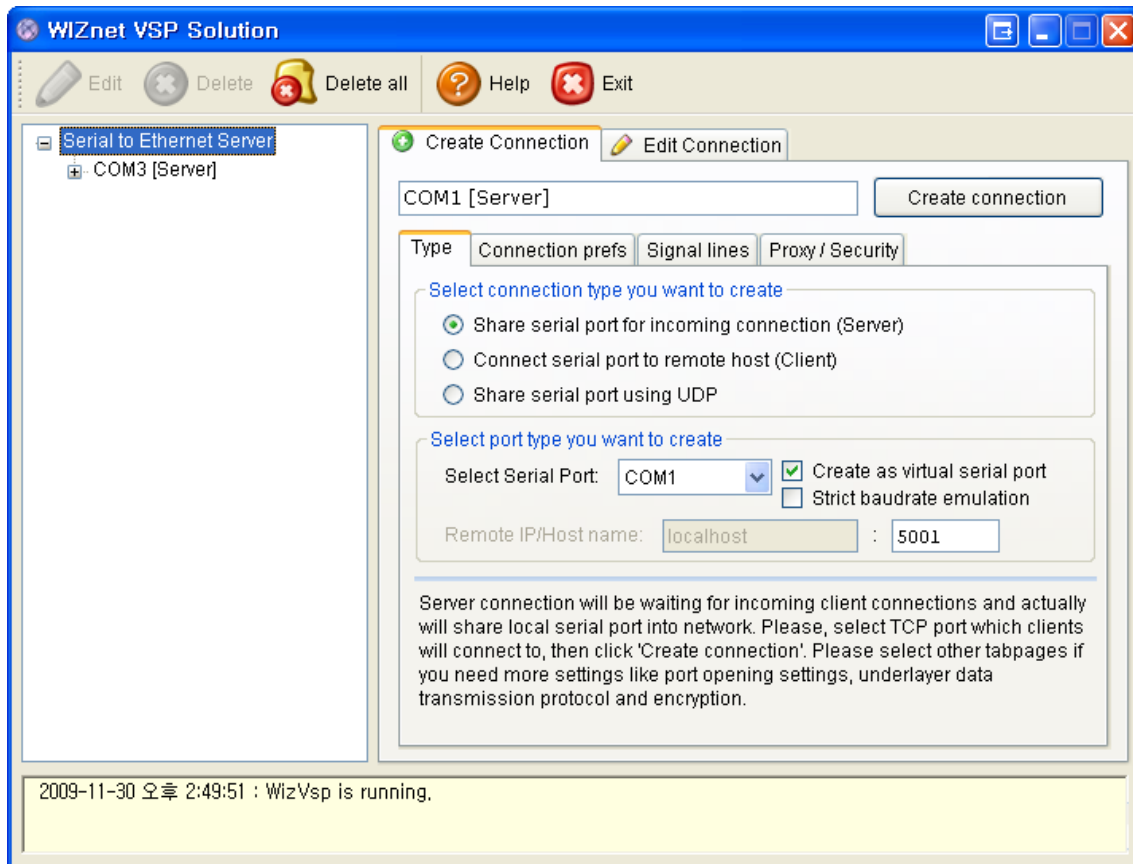




3. Program Interface

3.1 Main window

The main window is the first thing you see and access when you launch WIZ VSP.



The main windows are divided into the following parts by their functionalities :

- Main toolbar
- Connection tree
- Managing connections
- Activity log

These parts will be introduced in the following sections in details.

3.2 Main toolbar



The toolbar provides a quick and convenient access to commonly used functions :

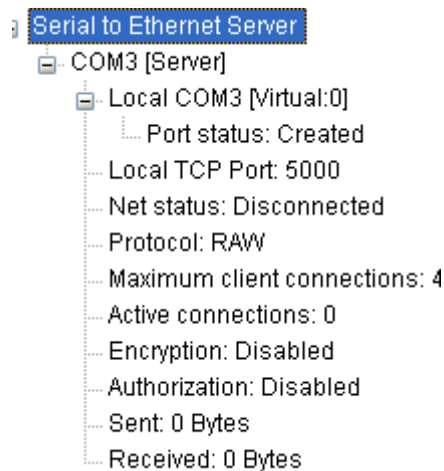
- **Edit**
Opens "Edit connection" tab to customized currently selected connection settings, and you are able to edit the selected connection (e.g. editing "Server connection", "Client connection", "UDP connection").
- **Delete**
Removes currently selected connection in the "Connections tree".
Please, note that if this connection is opened then after clicking "Delete" button it will be removed in the connections tree and close all connections of this port, and it may be exist in the system. Please be sure the virtual serial port is closed before you delete it.
- **Delete all**
"Delete all" button closes all opened connections and removes configurations from your system.
- **Help**
You can download the user manual at the WIZnet homepage.
- **Exit**
The main window will be closed, but your created connects will be remained and work well.

3.3 Connections tree

Connections tree helps visualizing all created and established connections in your system.

Each connection is represented as a single entry, which could be expanded to get detailed information about port number, connection status, remote host name, serial port, sent/received bytes, and so on.

For better representation, each connection type has its own distinctive name. Please, look what sub entries of a "server" connection has:



1. COM3 [Sever]

Connection name. You specify it in the Main window while creating new connection to identify this configuration.
2. Local COM3 [virtual]

Displays the local serial COM port which participates in connection. You can see in brackets either real or virtual serial port is used. If the port status is "Created", it means the port is created successfully and is closed. If the COM port is opened, the port status will displayed as well: Local COM3 [Virtual: 9600,N,8,1]. If the port status is "Not created/opened", it means there are some errors with this port, and you should remove it from your system and create it again.
3. Local TCP port

Shows the TCP port number which server is listening at.
4. Net status (connected/disconnected/listening/full)

Displays if you are connected/disconnected to remote host, when "listening" is displayed, server listens to all incoming connections. "Full" indicates that the maximum client connections number has been reached, and the server does not accept the further incoming connection request.

5. Protocol (RAW/Telnet)
Network protocol which is used in the connection.
6. Maximum client connections
Maximum number of clients which can be connected to server.
7. Active connections
Displays number of connections which are already established.
8. Encryption/Authorization
Encryption and authorization are not supported in current version for our serial-to-Ethernet modules.
9. Sent/Received
Shows how many bytes were sent/received within the established connection.

Additionally, Connections tree provides context menu which allows you to edit "Server/Client/UDP" connection or delete current or all connections and expand/collapse Connections tree.

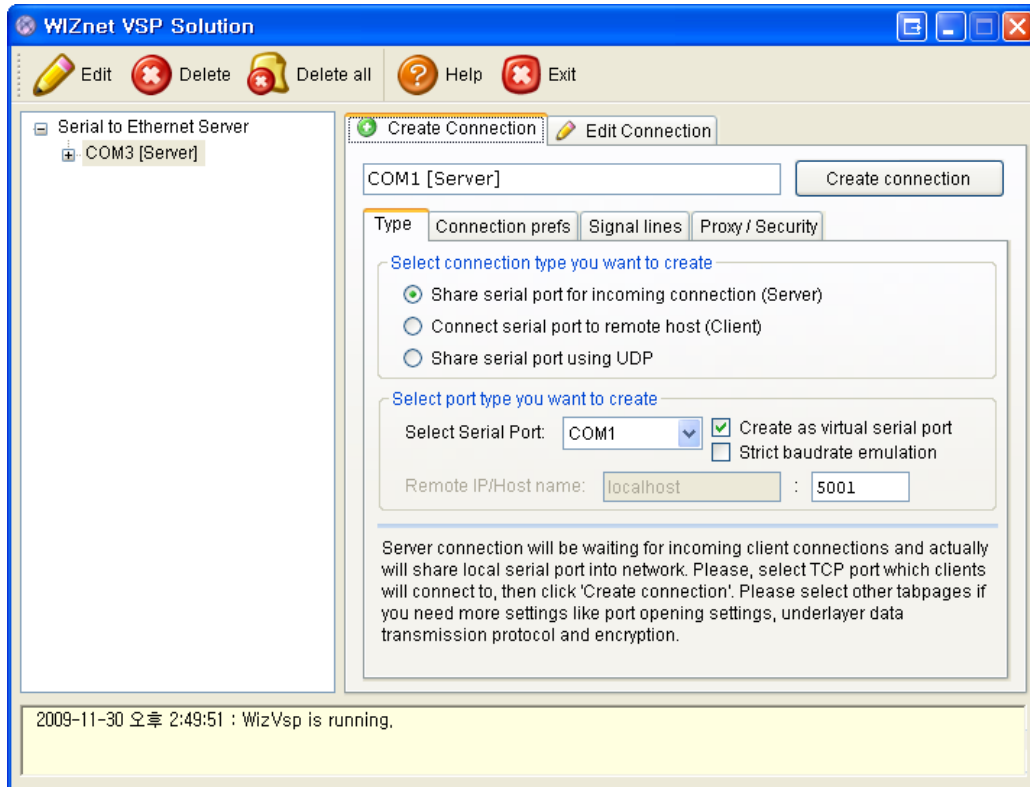
Edit connection
Delete
Delete all
Expand all
Collpase all

Connections tree always displays latest connection info which is dynamically updated on every event occurred.

3.4 Managing connections

WIZ VSP Main Menu provides convenient access to all program settings and options. All main features and options are grouped into the following tabs:

1. Create connection tab



From the “Select connection type you want to create” item, you can create connections for the three main purposes: you can either share serial port for incoming connections (server), or connect serial port to remote host (client) or share serial port using UDP.

- 1) Sharing serial port for incoming connections (refer to “Creating Server connection” section)

The Server connection will be waiting for incoming client connections and actually will share local serial port into network. All you have to do is to select a TCP port which clients will connect to and create a connection. The Server listens to all incoming connections at the defined TCP port and redirects all the input/output serial data to TCP/IP network.

- 2) Connecting local serial port to remote host (Creating “Client” connection)

The client connection will redirect all the local serial port data to the remote server. Client connection does not require SEC presence at the remote side, otherwise make

sure TCP ports are the same for "server" and "client", this will let you create a direct port-to-port connection.

All you have to do is specify remote server IP address (or network name) and TCP port to connect. Once connection is established, all data sent from remote serial port device, attached to the server, will be genuinely delivered to local serial port where it can be further processed.

3) Share serial port using UDP (creating "UDP" connection)

You can redirect all data from the local serial port using UDP/IP (User Datagram Protocol) underlying protocol instead of TCP/IP, providing additional flexibility for specific services (DNS, mail, finger, etc.). All you have to do is the same: specify remote server's IP address (or network name) and UDP port to connect. This type of connection does not require SEC presence at the remote side as well.

2. Edit connection tab

After a connection is created, you might want to modify the connection. To do this you should select the necessary connection in "Connections Tree" and click "Edit" on the main toolbar or just click one of the created connections in Connections tree and select "Edit connection".

Here you can change connection name, serial port which participates in connection, server IP-address and port number to connect to (in case you established "client" or port to UDP connection) and port number which server will listen to (in case you created "server" connection).

You can refer to editing "Server connection", "Client connection", "UDP connection" sections to get detailed information.

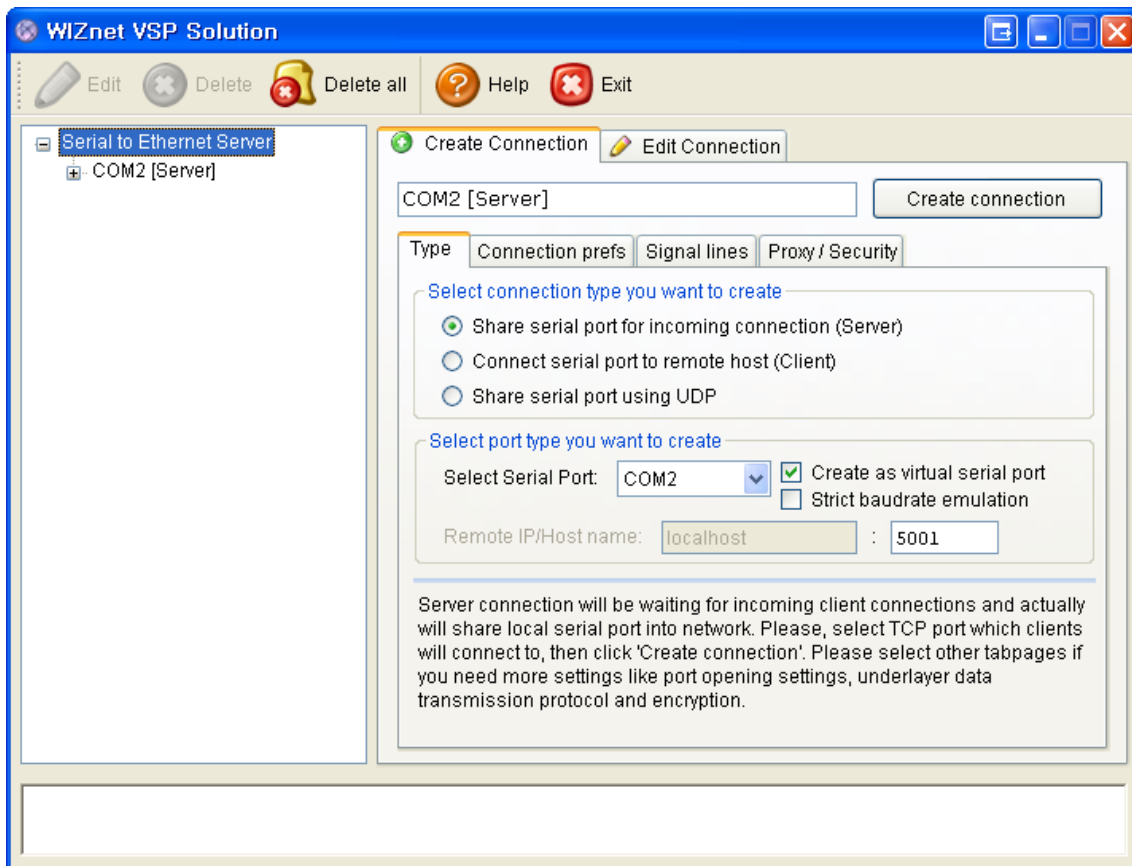
4. How to use WIZ VSP

4.1 Sharing Serial Port Using “Server” Connection

4.1.1 Creating Connection

WisVsp allows you to establish full-fledged client-server connection between two or more local real and virtual serial ports and WIZnet devices, which are connected to serial ports. So, you are able to get access to any serial port and any WIZnet serial device over network even if this device is connected to PC, which is located far from your home.

1. To create a connection in WIZ VSP, first specify the connection name to identify this configuration in “Create Connection” tab.



2. Then select a connection type which you want to create. In this case it is “Share serial port for incoming connections (server)”, then choose the local serial port which you want to share. Tick “Create as virtual serial port” option if you would like to use virtual serial ports instead of real ones.

Virtual serial ports are identical to the real serial ports and support all their features

such as parity control, data bits, stop bits, baud rate, flow control, signal lines, etc. The advantage of virtual serial ports technology is that you are not limited to the number of physical serial ports in a system and in this way you can free existing serial ports for other applications.

WIZ VSP supports serial COM ports overlapping. It means that virtual serial port can have the same name as existing physical COM port. If overlapped virtual COM port is created, it will be accessed instead of physical one.

Tick the "Strict baud rate emulation" checkbox if you want to enable baud rate emulation.

Baud rate emulation allows virtual ports to work at the same speed as the real ones. It might be useful when a virtual port is connected to a real port. The Baud rate emulation synchronizes their speed to avoid any data losses.

3. Specify a TCP port number which server will listen to. Make sure this port is not blocked by firewall (if any) and is not used by other services in your system (DNS, SMTP, IIS, etc.) If you are not sure, please, don't modify this value.
4. Finally click "Create connection" button in the Main Window. Once you have successfully created a connection, you will see your connection in the Connections tree.
5. Open the local serial port. You may use Windows HyperTerminal utility to verify the successful creation of connection.
6. Now you can start exchange data with the default settings. You can also refer to Editing "server" connection section if you are going to edit your connection settings.

You can consult **Setting up connection settings**, **Signal lines**, **Proxy and security settings** sections to find out more information on configuring all available settings in "Connection prefs", "Signal lines" and "Proxy/Security" tabs respectively of Edit Connection tab-page.

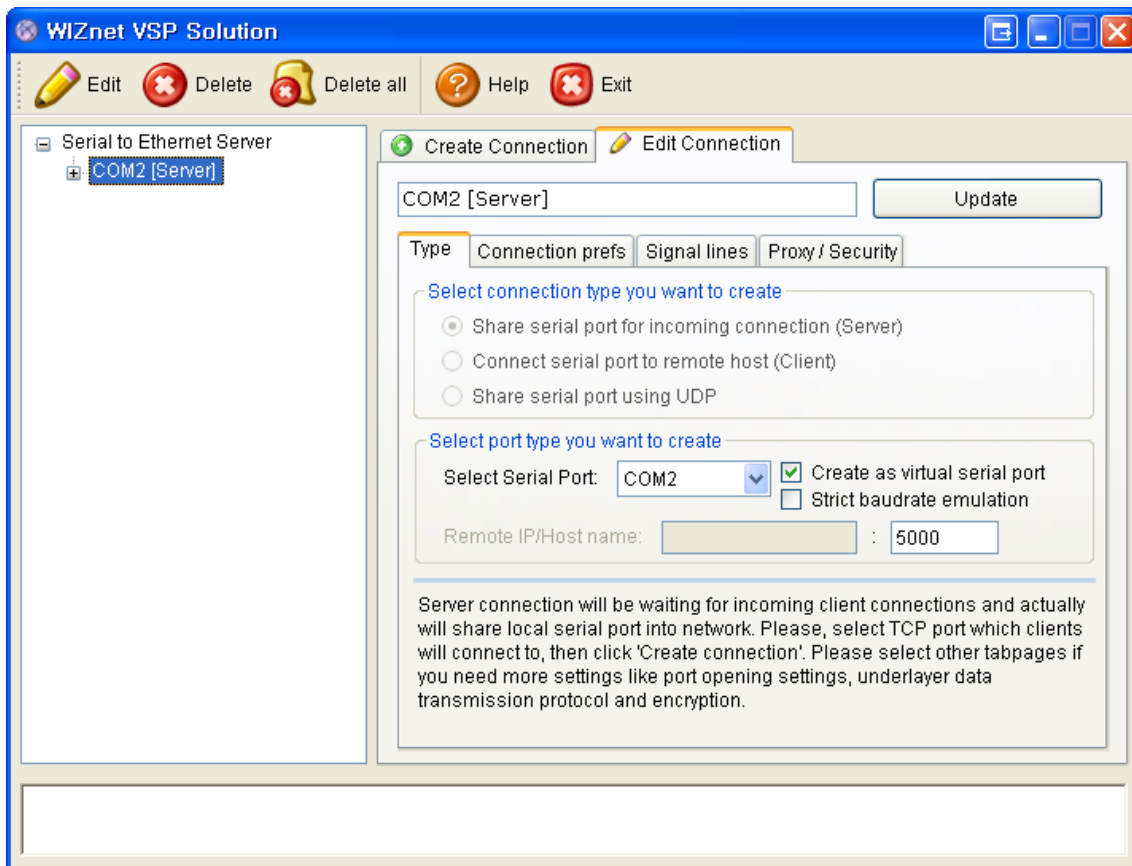
Please, refer to **Removing connections** section, if you would like to find out how to delete connections.

4.1.2 Editing connection

Once you have created a new connection, you can edit most of its settings on-the-fly in Serial to Ethernet Connector. Choose the connection which you want to modify from the branches in the connection tree and click "Edit" on the main toolbar or just right-click the connection in the "Connections Tree" and select "Edit connection".

Now you are able to change the following settings:

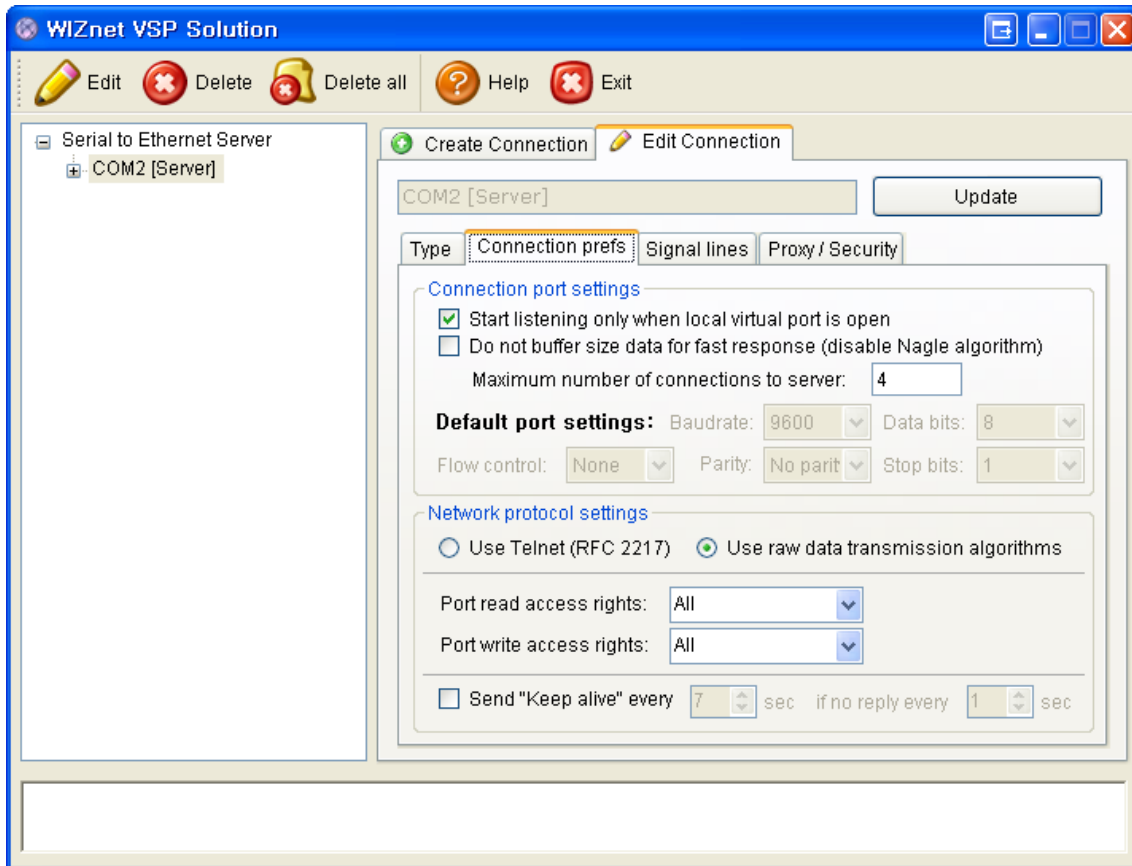
1. Connection name.
2. Local serial port which is used in connection.
3. TCP port number which server will listen to. Make sure this port is not blocked by firewall (if any) and is not used by other services in your system (DNS, SMTP, IIS, etc.) If you are not sure, please, do not modify this value.



When you finish editing connection settings, just click "Update" button and resume data exchange.

4.1.3 Connection settings

To configure other advanced connection settings, please, switch to the "Connection prefs" tab (stands for the Connection preferences) in the "Edit Connection" page in the main window, you will be shown the following set of features:



"Connection port settings"

Start listening for incoming connections only when local virtual port is open checkbox enables the SEC to start listening for all incoming connections only when local virtual port is opened. (Notice that all the incoming connections are ignored if virtual port is not opened). If this option is unchecked, all the incoming connections will be accepted, not depending on local port status.

Do not buffer size data for fast response checkbox is used for disabling the "Nagle algorithm", to achieve maximum speed of response if small packets are transmitted over network. Note that this will cause traffic increase. If you are uncertain about this option, please, do not modify it

Maximum number of connections to server option defines the maximum number of clients that can be simultaneously connected to the server at any time. Use this option if you want to restrict the number of connections (for instance if you have limited bandwidth).

Default port settings option allows you to either assign manually predefined serial port settings (for instance, if you have legacy serial port device that has specific parameters) or fetch them automatically from the remote end. Please, note this option is not available (grayed) if you are creating a virtual serial port since communication parameters are fetched during the handshaking process.

Baudrate defines the rate at which bits are transmitted (bits per second). In the serial port context, "9600 baud" means that the serial port is capable of transferring a maximum of 9600 bits per second. To be able to communicate at the maximum speed, both local and remote ends must be configured to the same baud rate and pass handshake stage before you can successfully read or write data.

Databits defines the number of data bits to transmit. Usually, the transferred bits include the start bit, the data bits, the parity bit (if used), and the stop bits. However, only the data bits carry useful information. You can configure DataBits to be 5, 6, 7, or 8. Data is transmitted as a series of five, six, seven, or eight bits with the least significant bit sent first (little-endian). At least seven data bits are required to transmit ASCII characters. Eight bits are required to transmit binary data. Five and six bit data formats are used for specialized communication equipment.

Parity defines the parity checking type. Parity can be one of the following: none, odd, even, mark, or space. If Parity is none, parity checking is not performed and the parity bit is not transmitted. If Parity is odd, the number of mark bits (1s) in the data is counted, and the parity bit is asserted or unasserted to obtain an odd number of mark bits. If Parity is even, the number of mark bits in the data is counted, and the parity bit is asserted or unasserted to obtain an even number of mark bits. If Parity is mark, the parity bit is asserted. If Parity is space, the parity bit is unasserted.

Stop bits sets the number of bits used to indicate end of a byte. Stop bits could be 1, 1.5, or 2, however almost all contemporary devices are configured to 1 Stop bit. Please, note that both ends of serial port must be configured to transmit the same number of stop bits to work properly.

Flow control defines the Flow control type. Flow control is usually used to ensure that the receiving serial port device can handle all of the incoming data sent to it. Advanced Serial Port Terminal provides the following values you can assign to Flow control:

Xon/Xoff (commonly used for asynchronous communication), Hardware and None.

You can also choose the flexible Telnet (RFC2217) protocol for port-to-port connection to provide the ability to communicate with WIZnet serial devices and applications which support Telnet (RFC2217) protocol directly.

There are four main functionality features of the Telnet (RFC 2217) protocol:

- The ability for the client to change and send serial port configuration information to the access server. This is needed to ensure the data being transmitted and received is formatted correctly at the byte level.
- The ability for the access server to inform the client of any modem or signal line changes.
- The ability to monitor and operate remotely any serial device.
- The ability to create port-to-port connection via TCP/IP protocol.

"**Network protocol settings**" let you choose the under-layer protocol to be used for data transmission in your connection:

If you are using Telnet (RFC 2217) protocol to transmit data over network you have additional options to tune:

Notify remote host on local port settings change checkbox attempts to change the local port settings (Baudrate, Data bits, Flow control, Parity, Stop bits and Signal lines) on client side, depending on your values (actual, if client connects via real serial port only). Please note that signal lines will be sent in any case, not depending on local serial port types.

Allow changing local port settings checkbox is available only when real serial ports are used in the connection. If the remote host wants to change your local port settings, he will have the permission to change them.

Send command to keep connection alive option enables you to specify period of time (in msec). After this period elapsed, Serial to Ethernet Connector sends a command to the remote end in order to verify the connection's integrity and keep the connection alive.

If Use **raw data transmission algorithms** checkbox is used, you are able to specify the client's access rights to the local COM port which lets you manage the communication direction. You can enable or disable port read/write access rights.

Send "Keep alive" every X sec ... if no reply every Y sec

Send "Keep alive" option gives an interlocutor "Keep alive" command if there is no data transfer between the sides during X seconds. An interlocutor must answer this message.

This option is used to detect connection problems and to maintain active connection. Use this option if your device disconnects automatically after a definite time of net inactivity.

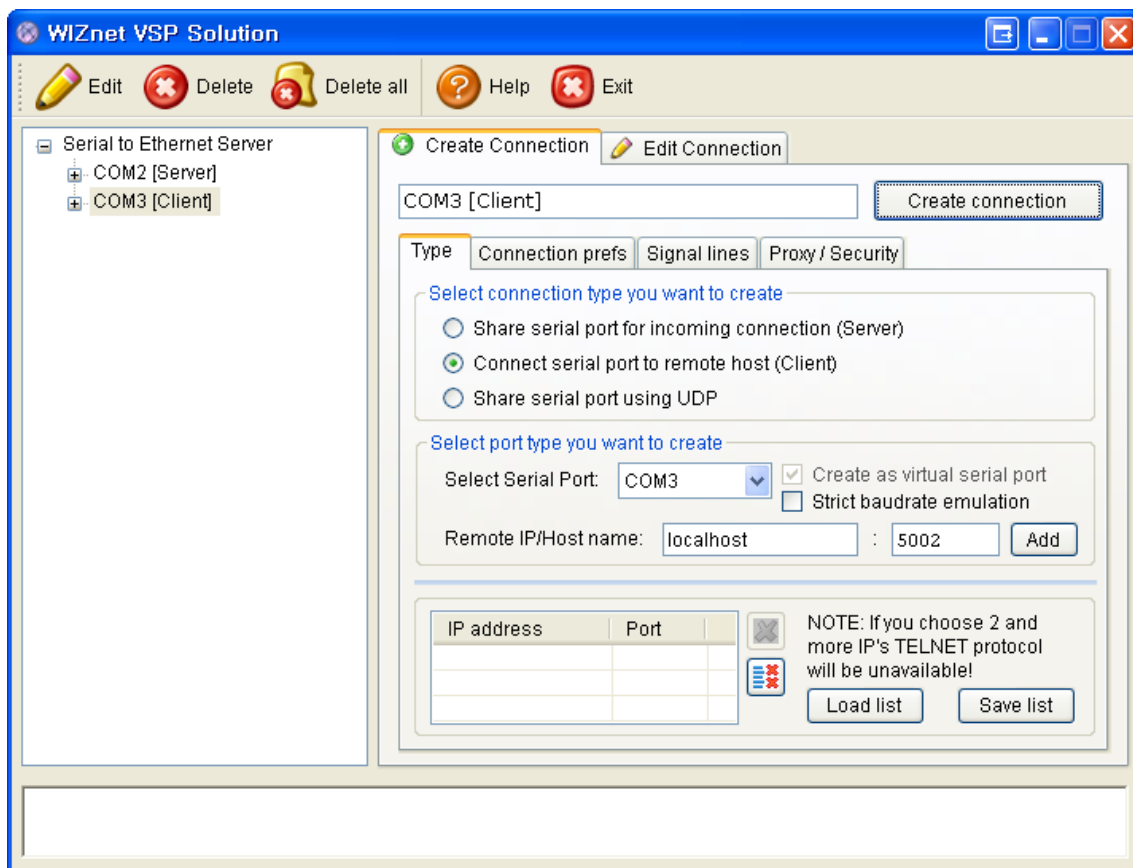
Y is an interval of sending "Keep alive" message in case an interlocutor didn't answer the last "Keep alive" command sent in X sec interval.

4.2 Connecting serial port using "client" connection

4.2.1 Creating connection

WIZ VSP provides flexible way to establish client-server between two COM ports. Virtual serial port connection is faster and more reliable than real serial port connection. In the terms of client-server application, you may create up to 255 simultaneous connections to remote server.

1. Specify a connection name to identify this configuration (COM1 [Client] is set by default) in "Create connection" tab. Default name is based on local serial port, which participates in connection and selected connection type in brackets.



2. Then select connection type which you want to create. It is "Connect serial port to remote host (Client)" in this case, then choose local serial port which will be shared in this connection. Tick "Create as virtual serial port" checkbox if you would like to use virtual serial ports instead of real ones. Tick "Strict baudrate emulation" checkbox if you want to enable baudrate emulation.
3. Specify the server's IP address (or network name) and port number to connect to. Make sure that port number matches the one at remote host and isn't blocked by firewall. Click "Add" button to add IP address to the IP's list. You are able to connect

to several remote computers simultaneously using the IP's list. Moreover you can save your IP's list to hard drive and load it using "Save list" and "Load list" buttons.

4. Finally click "Create connection" button in the Main Window. On successful creation, you will see your connection in **Connections tree**.
5. Open the local serial port. You may use Windows HyperTerminal utility to verify the successful creation of connection.
6. Now you are ready to start communication process with default settings. If you would like to edit your connection settings, please, consult Editing "client" connection section.

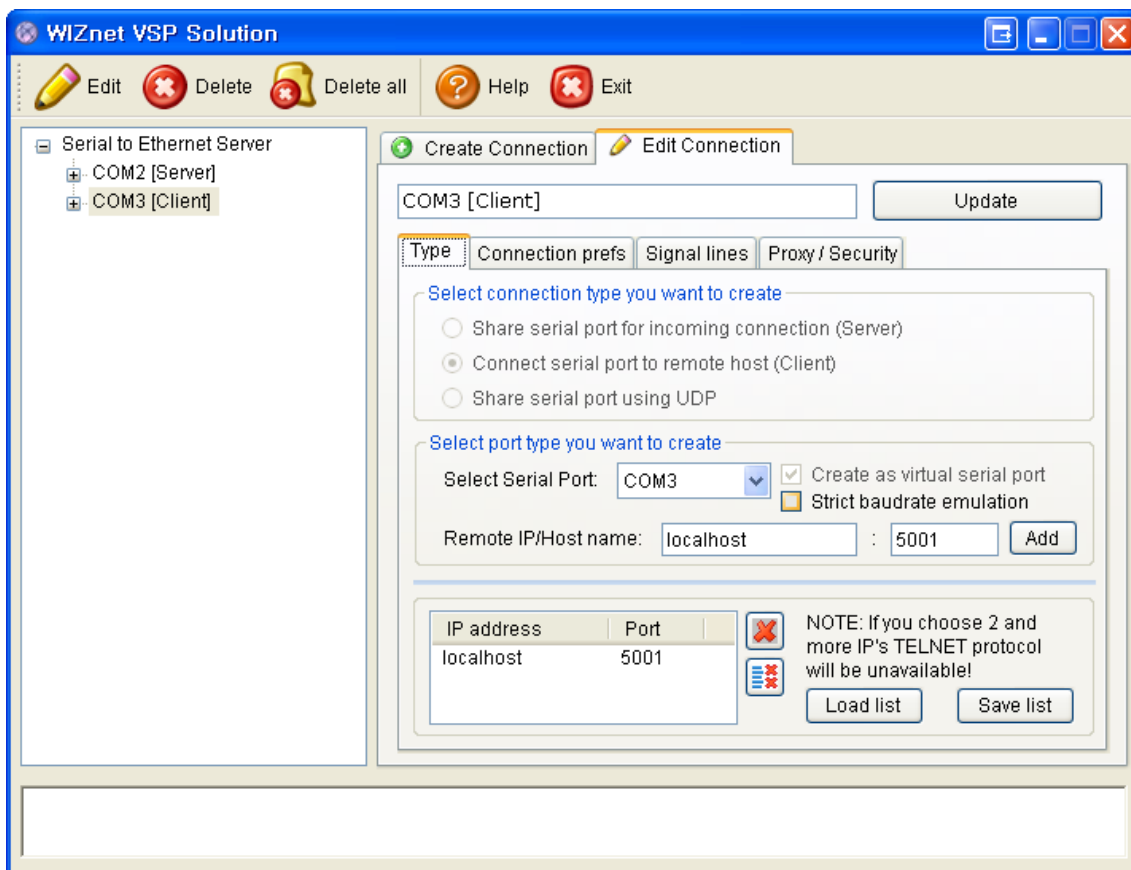
If you would like to find out how to delete the connection, please, refer to **Removing connections** section.

4.2.2 Editing connection

Once you have created a new connection you can edit most of its settings. Serial to Ethernet Connector allows you to do it on-the-fly. You need to choose the desired connection or any of its branches in Connections tree and click "Edit" on the main toolbar or just right-click the created connection in "Connections Tree" and select "Edit connection".

Now you are able to change the following settings:

1. Connection name.
2. Serial port which is shared with connection.
3. Server's IP address (or network name) and port number to connect to.

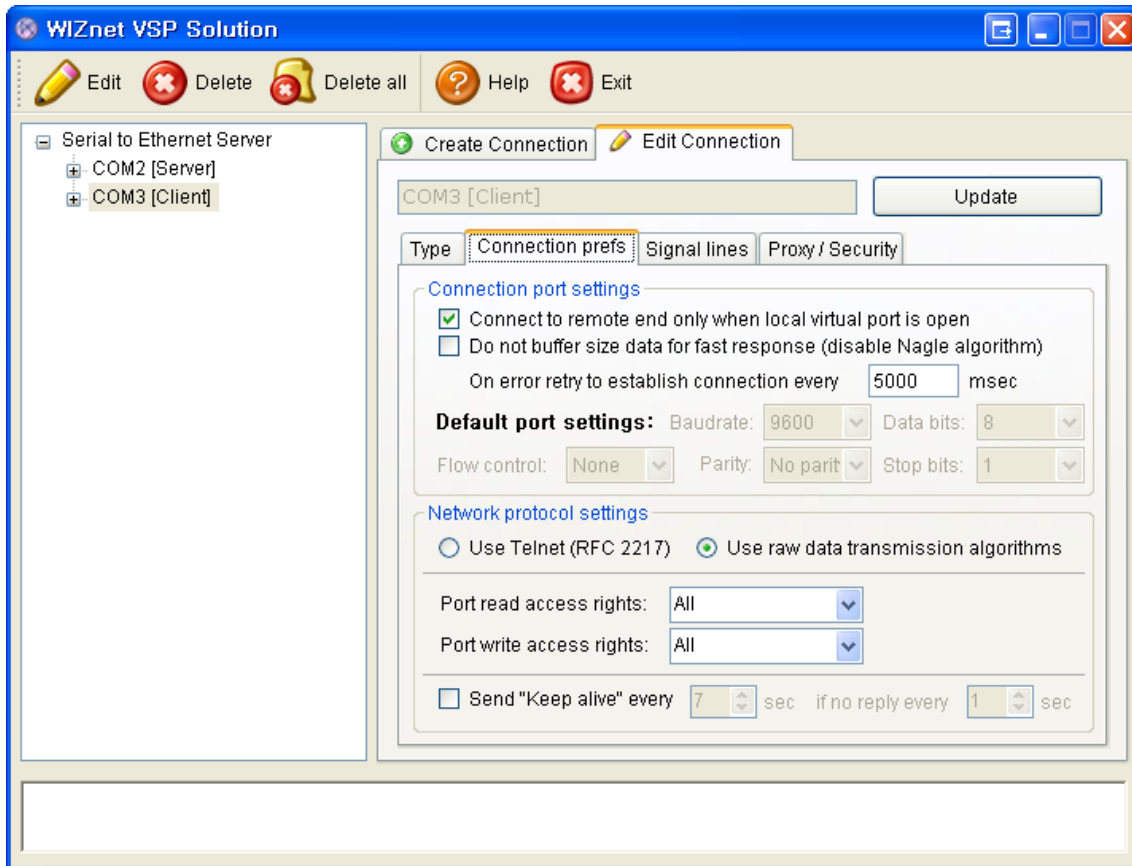


When you finish editing connection settings, just click "Update" button and resume the communication.

Note: make sure that the remote host's name you specified is the local one and the port number matches the one you set in the local end configuration.

4.2.3 Connection settings

To configure other advanced connection settings, please, switch to the "Connection prefs" tab (stands for Connection preferences) in the "Edit Connection" page in the main window, you will be shown the following set of features:



"Connection port settings"

Connect to remote end only when local virtual port is open checkbox specifies whether the connection attempts to connect or not only after virtual port is opened for communication by your application.

Do not buffer size data for fast response checkbox is used for disabling "Nagle algorithm", to achieve maximum speed of response if small packets are transmitted over network. Note that this will cause traffic increase. If you are not sure, please, do not modify this option.

Besides you can specify time to reconnect if any error occurs (in milliseconds). When connection is broken, Serial to Ethernet Connector will try to restore connection after reconnection period.

Default port settings option allows you to either manually assign predefined serial port settings (for instance you have legacy serial port device that has specific parameters) or fetch

them automatically from the remote end. Please, note this option is not available (grayed) if you are creating virtual serial port since communication parameters are fetched during handshaking process.

You can refer to **Connection settings (server)** section for more info about these parameters.

You can also choose the flexible Telnet (RFC2217) protocol for port-to-port connection to provide the ability to communicate with WIZnet serial devices and applications which support Telnet (RFC2217) protocol directly.

"**Network protocol settings**" let you choose the under-layer protocol to be used for data transmission in your connection:

If you are using **Telnet (RFC 2217) protocol** to transmit data over network, please refer to **Connection settings (server)** section for more info about these parameters.

If Use **raw data transmission algorithms** checkbox is used, you are able to specify the client's access rights to the local COM port which lets you manage communication direction. You can enable or disable port read/write access rights.

Send "Keep alive" every X sec ... if no reply every Y sec

Send "Keep alive" option gives an interlocutor "Keep alive" command if there is no data transfer between the sides during X seconds. An interlocutor must answer this message.

It permits to find possible connection problems and to maintain active connection. Use this option if your device disconnects automatically after a definite time of net inactivity.

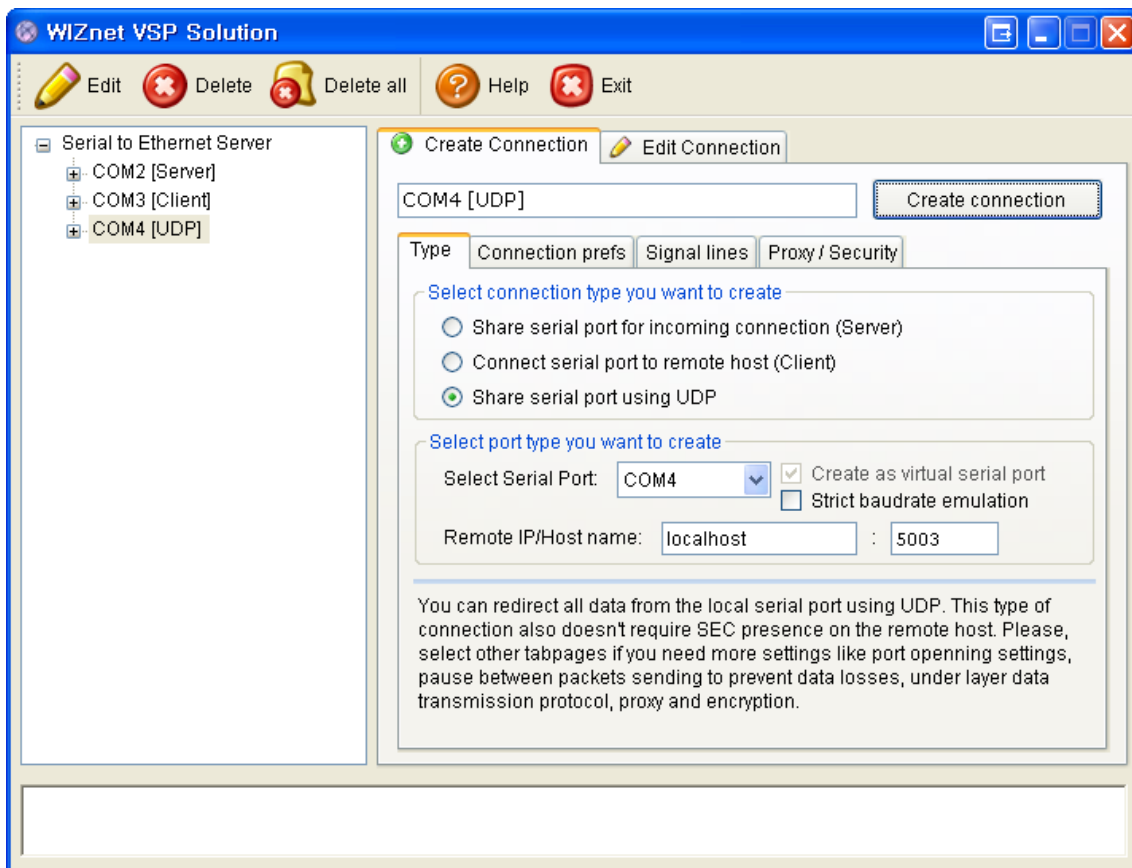
Y is an interval of sending "Keep alive" message in case an interlocutor didn't answer the last "Keep alive" command sent with X sec interval.

4.3 Sharing serial port using UDP

4.3.1 Creating connection

WIZ VSP allows you to establish UDP/IP connection between serial ports. UDP type of connection is useful for streaming big chunks of data as well as for mail, DNS, finger and other services.

1. Specify the connection name to identify this configuration (COM1 [UDP] is set by default) in "Create connection" tab. Default name is based on the local serial port, which participates in connection and selected connection type in brackets.



2. Then select connection type you want to create. In this case it is "Share serial port using UDP" and choose local serial port which will be shared with the connection. Tick "Create as virtual serial port" option if you would like to use virtual serial ports instead of real ones. Tick "Strict baudrate emulation" checkbox if you want to enable baudrate emulation.

You can find more details about the virtual serial port and baudrate emulation instructions in "Create connection (server)" section.

3. Specify the server's IP address (or network name) and the UDP port number to connect to. Make sure that UDP port number matches the one at remote end and isn't blocked by firewall.
4. Finally, verify that all settings are correct and click "Create connection" button in the Main Window. On successful creation, you may see your connection in **Connections Tree**.
5. Open the local serial port. You may use Windows HyperTerminal utility to verify the successful creation of connection.
6. Create a "UDP" connection at the remote end and make sure your host name, IP address and port number match at both end of the connection.

For instance, if you have 2 PCs with IP addresses 192.168.0.1 and 192.168.0.2 then on the one that has 192.168.0.1 you should set server address to 192.168.0.2 and vice versa. UDP ports must be the same.
7. Now you are ready to start communication process with default settings. You can refer to **Editing "UDP" connection** if you would like to edit new established connection.

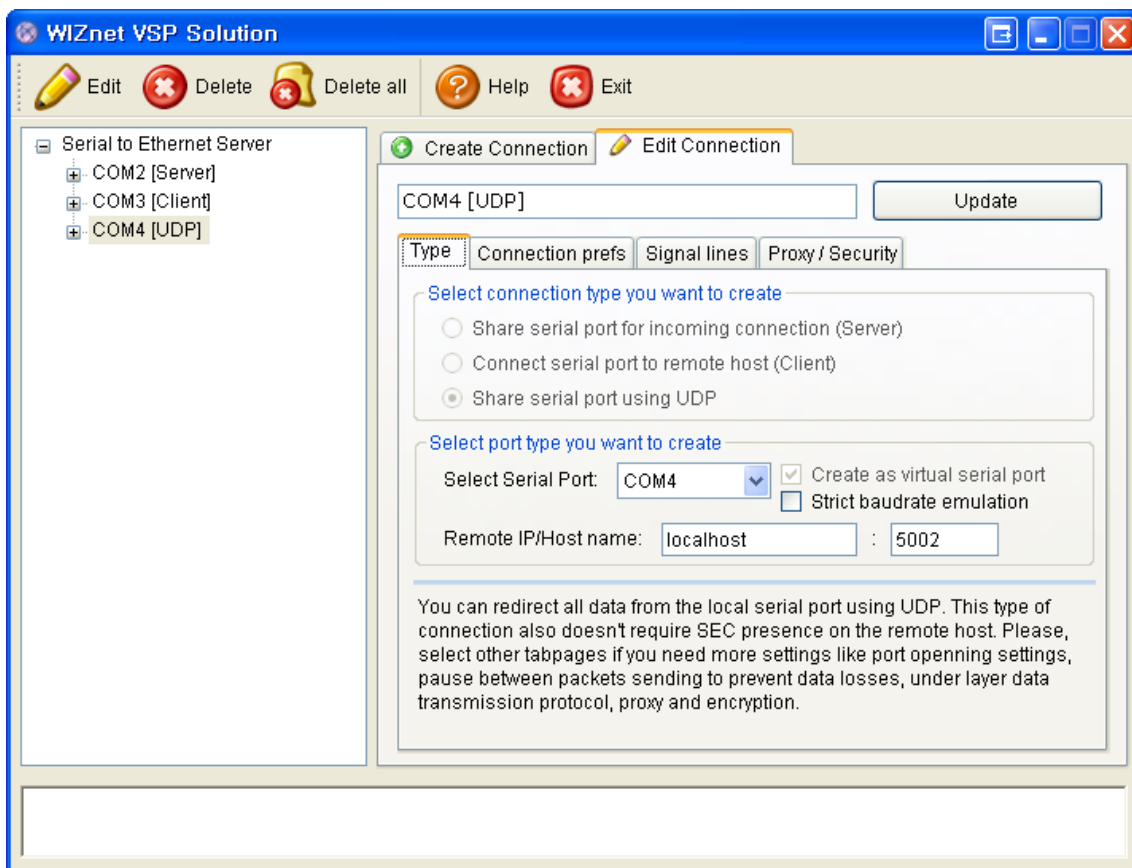
If you'd like to find out how to delete the connection, please refer to **Removing connections** section.

4.3.2 Editing port to UDP connection

When a new connection created you may want to edit it. WIZ VSP allows you to do it on-the-fly. Please, choose the connection in Connections Tree and click "Edit" on the toolbar panel or just right-click created connection in the "Connections Tree" and then select "Edit connection".

Now you are able to change such settings:

1. Connection name.
2. Serial port which shares the connection.
3. Server's IP address (or network name) and UDP port number to connect to.

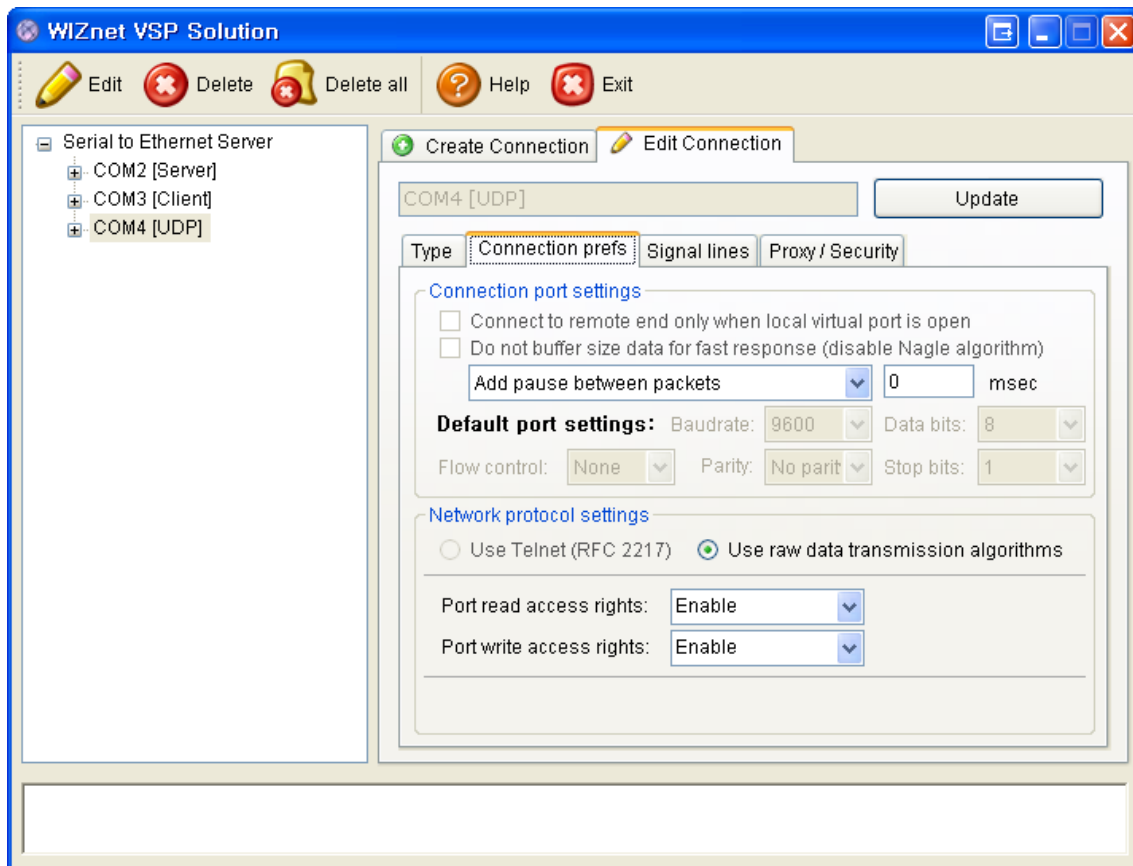


When you finished editing your connection just click the "Update" button and resume communication.

Note: make sure the hostname and port number matches on both side of the connection.

4.3.3 Connection settings

To configure other advanced connection settings, please, switch to the "Connection prefs" tab in the "Edit connection" tab-page in the main window, you will be shown the following set of features:



"Connection port settings"

When using UDP/IP connections, you are able to configure the following settings:

Connect to remote end only when local virtual port is open checkbox specifies whether the connection attempts to connect or not only after virtual port is opened for communication by your application.

Do not buffer size data for fast response checkbox is used for disabling "Nagle algorithm", to achieve maximum speed of response if small packets are transmitted over network, note that this will cause traffic increase. If you are not sure, please, do not modify this option.

From this list Add pause between packets 0 msec you are able

configure the following:

Add pause between packets option enables you to set a pause (in msec) between packets during transmission. This is necessary due to the fact that data losses will occur when queue overflow occurs in UDP oriented connection. Thus, you can prevent this from happening by adding pauses in between packets.

Before sending data wait for option is used for setting a delay (in msec) before the received serial data is sent to the remote end.

Send data when block reached the size of option is used to send all received data to the remote end when a block of reaches the size of the specified value.

Send data when received char with code option is used to send all received data to remote end when char received with specified code.

Default port settings option is located below, which allows you to either manually assign predefined serial port settings (for instance you have legacy serial port device that has specific parameters) or fetch them automatically from the remote end. Please, **note** this option is not available (grayed) if you are creating virtual serial port since communication parameters are fetched during handshaking process.

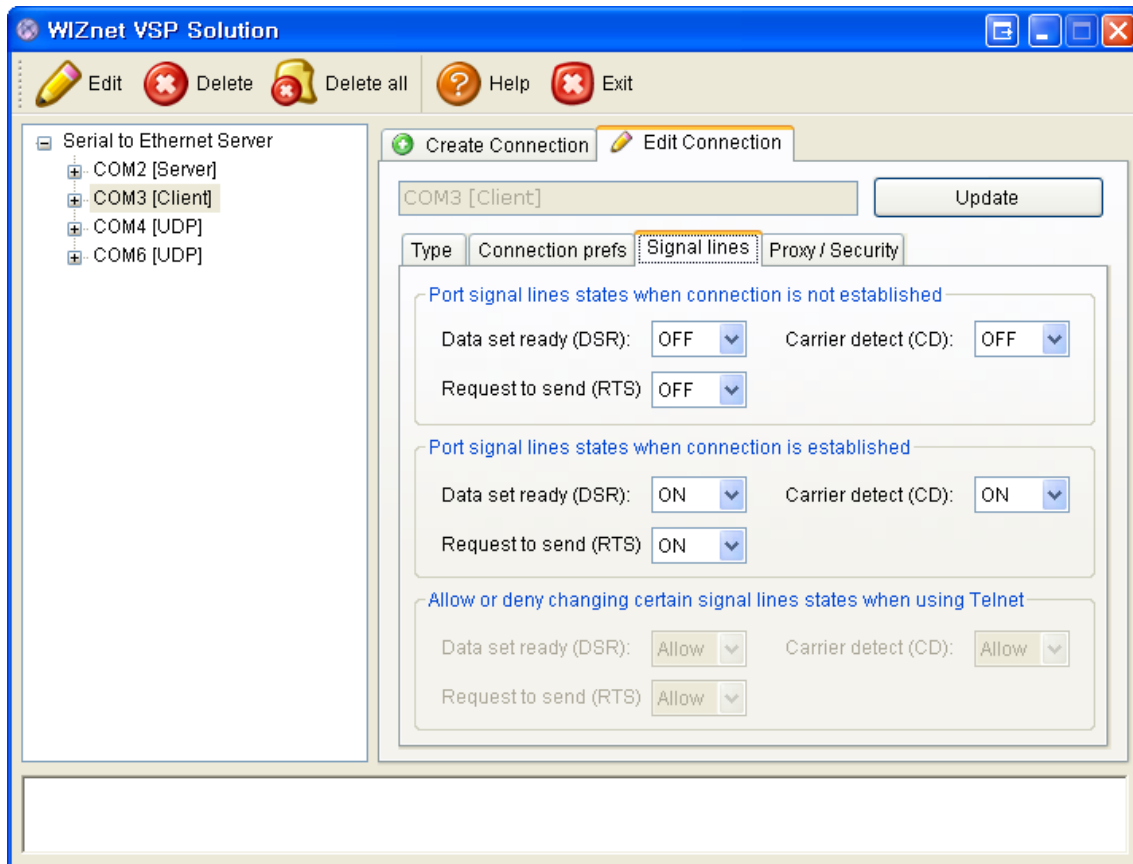
You can refer to the **Connection settings (server)** section for more info about these parameters.

If the **Use raw data transmission algorithms** checkbox is used, you are able to specify the client's access rights to the local COM port which lets you manage the communication direction. You can enable or disable port by changing the read/write access rights.

4.4 Managing signal lines in connections

To enable the signal lines settings, please switch to the "Signal lines" tab in the main window and you will be shown the following set of features:

1. When you are using virtual serial port in the connection, you are able to modify the following:



Port signal lines states when connection is not established:

Here you can change state of DSR, CTS and CD signal lines when a connection has "Disconnected" status in the "client" connection or there are no connected clients for the case of "server" connection.

Port signal lines states once connection is established:

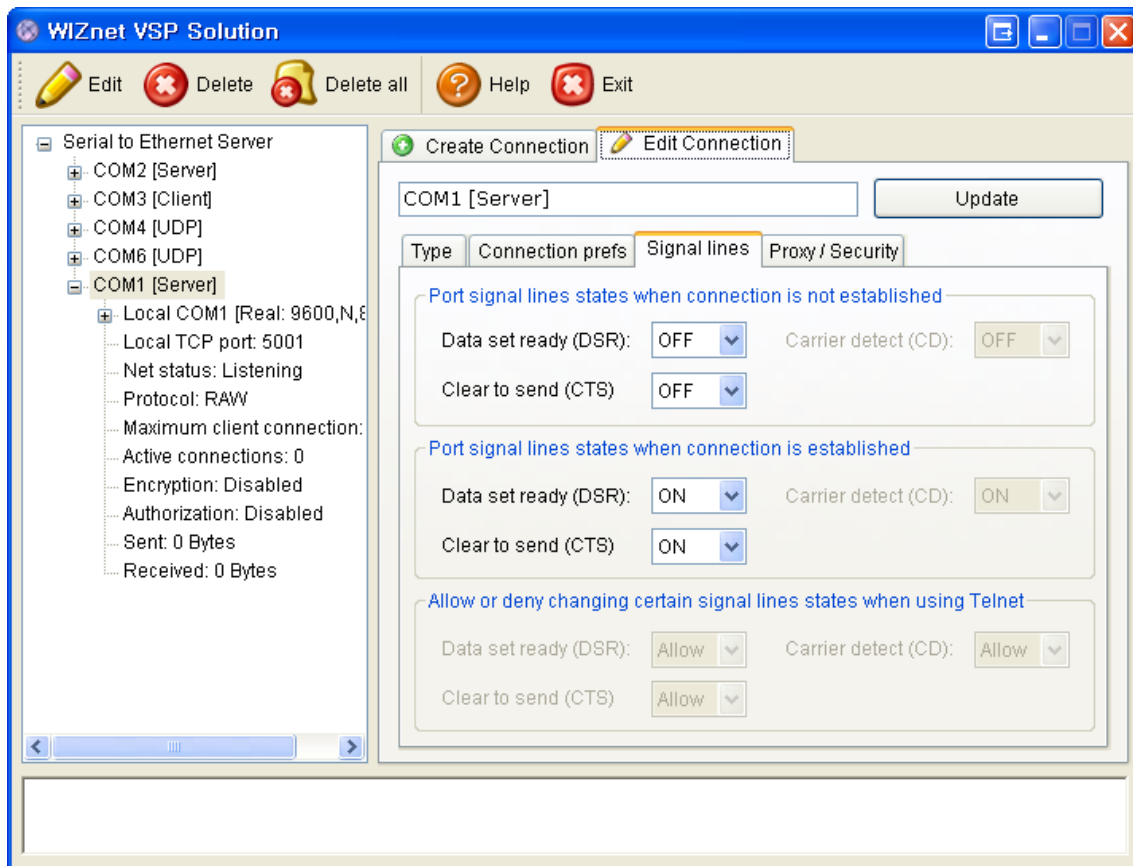
You are able to manage the same signal lines (DSR, CTS, CD) states even when the connection is already established.

Allow or deny changing certain signal lines states when using Telnet protocol:

you are using Telnet data transmission protocol in your connection you can allow or deny changing DSR, CTS and CD signals state.

2. If real (physical) serial port is used in the connection, you will be able to configure

the following signal lines settings:



Port signal lines states when connection is not established:

You can change the state of DTR and RTS signals when a connection is not established yet, in other words, when a connection has the "Disconnected" status in "client" connection or there are no connected clients for the "server" connection case.

Port signal lines states once connection is established:

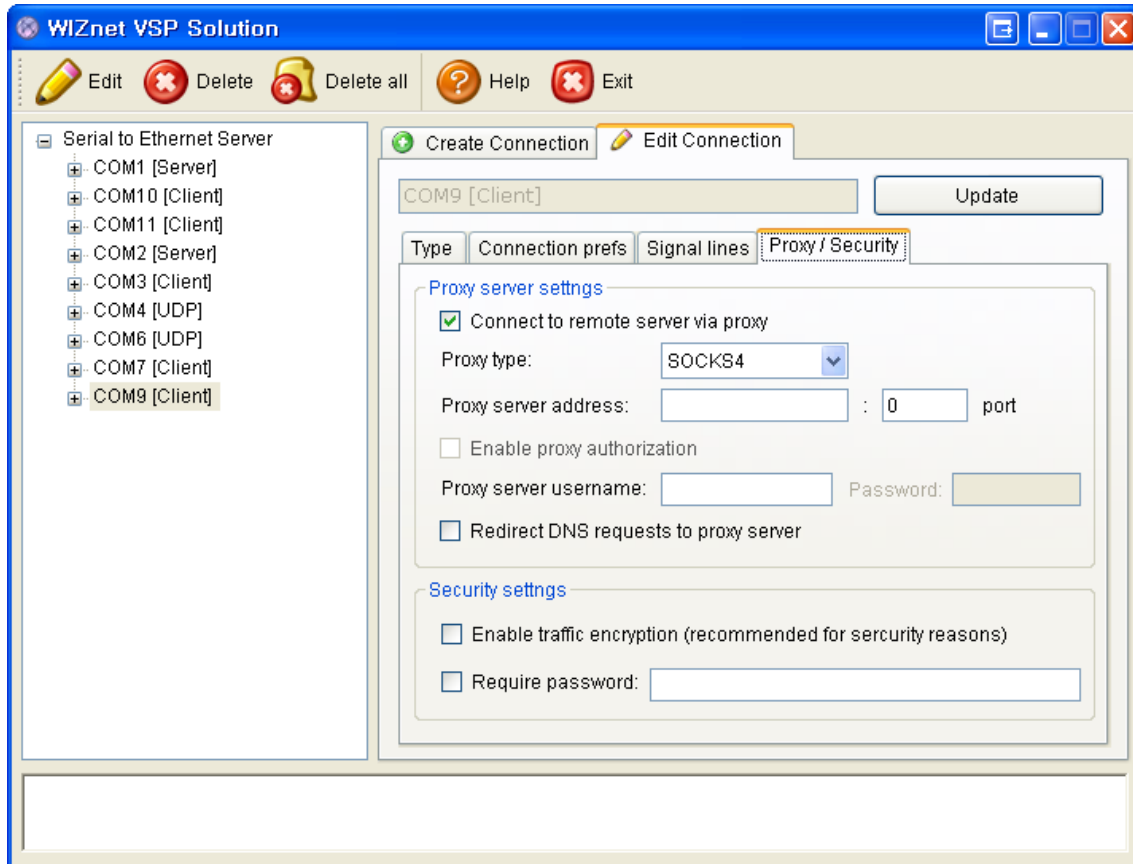
You are able to manage the same signal lines (DTR, RTS) states at the moment when connection is already established.

Allow or deny changing certain signal lines states when using Telnet protocol:

Here, you can allow or deny changing DTR, RTS and CD signal lines states when you are using Telnet protocol in your connection.

4.5 Proxy and security settings

To make proxy and security settings available, please, switch to the "Proxy and Security" tab in the main window and you will be shown the following set of features:



Now tick the "Connect to remote host server via proxy" checkbox, choose proxy type you use (SOCKS4, SOCKS5, HTTPS) and input the proxy server address and port.

If you want to enable proxy authorization (only available for SOCKS5 or HTTPS proxies), please tick the correspondent checkbox and input in proxy server username and password.

You can tick "Redirect DNS requests to proxy server" checkbox if you want the proxy to convert DNS name to IP address. If not, the local DNS is used.

Note, that these three options are available in "client" connection only.

Please note that the "Security settings" is not supported in current version!