



Serial/IP[®] Redirector ActiveX Interface

Tactical Software, LLC
www.tacticalsoftware.com

Copyright © 2013 Tactical Software, LLC. All Rights Reserved.

Tactical Software and Serial/IP are registered trademarks of Tactical Software, LLC.
All other trademarks are the property of their respective holders.

Table of Contents

1. Introduction	1
2. ITSRedirectorConfiguration	3
3. ITSPortList	4
4. ITSPortSettings	7
5. ITSProxySettings	11
6. ITSSSLSettings	13
7. ITSOptionSettings	17

1. Introduction

Beginning in version 4.4 of the Serial/IP Redirector, a Component-Object-Model (COM)-based API allows external programs and scripts to change all aspects of the Serial/IP configuration.

The API exports a single top-level object: `TacticalSoftware.SerialIPRedirector` which implements the `ITSRRedirectorConfiguration` interface. This object contains the following sub-objects which are accessible as properties:

Name	Type	Description
Ports	<code>ITSPortList</code>	Container of <code>ITSPortSettings</code> objects, representing the currently redirected ports, plus functions for adding and removing redirected ports.
SSL	<code>ITSSSLSettings</code>	Object which manages the global SSL/TLS-related properties.
Proxy	<code>ITSProxySettings</code>	Object which manages the global Proxy-related properties.
Options	<code>ITSOptionSettings</code>	Object which manages the global “advanced” options.

Configuration of Serial/IP is done through one of the properties on the top-level object. For example, in VBScript:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
SerialIP.Proxy.Enabled = TRUE
SerialIP.Proxy.Protocol = TSProxy_SocksV4
SerialIP.Proxy.Address = myproxy.tactical-sw.com
```

API Structure

Each interface exported by the Serial/IP API corresponds to a section of the Serial/IP Control Panel:

ITSPortList – Manages the list of redirected COM ports, implementing the functionality of the COM Port list and “Select Ports” button on the Control Panel main window.

ITSPortSettings – Manages each redirected COM port. It implements properties and functions that correspond to the port-specific fields on the right-hand side of the Control Panel main window.

ITSSSLSettings – Manages the global encryption settings. It implements properties and functions that correspond to the “SSL Encryption”, “SSL Authentication”, and “SSL Certificate” tabs in the “Advanced” dialog.

ITSProxySettings – Manages the proxy server settings. It implements properties and functions that correspond to the “Proxy Server” tab in the “Advanced” dialog.

ITSOptionSettings – Manages the global options. It implements properties and functions that correspond to the “Options” tab in the “Advanced” dialog.

Since there is such a strong correlation between the above interfaces and fields within the Serial/IP Control Panel, most of the properties are given only a brief description and its corresponding GUI field in this document. For detailed information on how each property affects the operation of Serial/IP, please refer to the Serial/IP Users Guide.

Common Errors

Accessing any of the properties or functions of the API may cause an error to occur. The following lists the common errors and their causes:

E_FAIL – Many properties and functions must communicate with the Serial/IP driver to complete. This error will occur if the driver is missing or not loaded.

E_INVALIDARG – Properties and functions which accept a COM port as a parameter will generate this error if the COM port name is invalid or specifies a COM port that is unavailable or in use. This error may also occur with other functions and properties when an invalid or out of range value is applied.

ERROR_PATH_NOT_FOUND – Functions which accept a filename as a parameter will generate this error if the file is missing or inaccessible.

Other errors may occur, which are listed in the description of specific functions and properties.

2. ITSRedirectorConfiguration

The TacticalSoftware.SerialIPRedirector object implements the ITSRedirectorConfiguration interface, which provides all the properties, functions, and sub-interfaces to control the configuration of Serial/IP.

ITSRedirectorConfiguration Properties

AdministratorOnlyMode

This is a read-only property which returns TRUE if Serial/IP was installed in Administrator-Only Mode.

AdministratorOnlyRestricted

This is a read-only property which returns TRUE if Serial/IP was installed in Administrator-Only Mode, and the current user does not have Administrator rights. If this property returns TRUE, then the API may only be used to query the current values, and any attempts to change the current values will generate an error.

ITSRedirectorConfiguration Functions

CloseGUI()

Closes all running Serial/IP GUI processes.

LaunchControlPanel()

Runs the Serial/IP Control Panel.

LaunchPortMonitor()

Runs the Serial/IP Port Monitor.

LaunchTraceWindow()

Runs the Serial/IP Port Monitor, with the Trace tab displayed.

SaveSnapshot(String *Snapshotfile*)

Saves the current configuration to the file specified by *Snapshotfile*.

RestoreSnapshot(String *Snapshotfile*)

Restores a save configuration from the file specified by *Snapshotfile*.

3. ITSPortList

The ITSPortList interface provides properties and functions to manage the list of redirected COM ports. Its primary role is as a container of ITSPortSettings objects, but it also exports some top-level properties and functions of its own.

ITSPortList Properties

Count

This is a read-only property which returns the number of redirected ports.

Available

This is a read-only property which returns an enumerator for an array of strings, each of which is a COM port designation that is not currently in use by a driver on the system, and is therefore a candidate to be redirected by Serial/IP.

IsRedirected(String *COMPortName*)

This is a read-only property which returns TRUE if the COM port specified by *COMPortName* is redirected.

IsAvailable(String *COMPortName*)

This is a read-only property which returns TRUE if the COM port specified by *COMPortName* is available to be redirected. Returns FALSE if the port is allocated for another use, either as a Serial/IP redirected port, a physical port, or a virtual port.

ITSPortList Functions

Add(String *COMPortName*)

Adds the COM port specified by *COMPortName* to the list of redirected ports. An E_FAIL error may occur if the maximum port count exceeded.

Remove(String *COMPortName*)

Removes the COM port specified by *COMPortName* to the list of redirected ports.

Set(String *COMPortList*)

Sets the comma-delimited list of ports specified by *COMPortList* to be the list of redirected ports. An E_FAIL error may occur if the maximum port count exceeded.

ITSPortList Notes

Add/Remove/Set Functions

Note that on Windows 95/98/ME, a call to any of these functions will result in a ten-second delay before the function returns. This allows the Windows plug-and-play enumerator time to “find” the new COM port layout in the device list.

PortList Container Behavior

The *Ports* object is also a container, which when enumerated returns an object which implements the *PortSettings* interface. Note that if a port is added or removed while the container is being enumerated, the results are undefined and care should be taken not to modify the list of ports while enumeration is in progress. In addition to enumeration, access by index is also supported. Indexes may either be a zero-based integer (which returns the Nth redirected port in the list), or a string which is the COM port name. For example:

```
Dim SerialIP
Dim SerialPort
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set SerialPort = SerialIP.Ports("COM5")
```

Note that any time the *Ports* attribute returns an object, it is a new object. To test for equivalence between ports, use the *Name* property. For example:

```
Dim SerialIP
Dim SerialPortA
Dim SerialPortB
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set SerialPortA = SerialIP.Ports("COM5")
Set SerialPortB = SerialIP.Ports("COM5")

` Wrong
If (SerialPortA Is SerialPortB) Then Print "Same"

` Right
If (SerialPortA.Name = SerialPortB.Name) Then Print "Same"

` Right also, since Name is the default value for PortSettings
If (SerialPortA = SerialPortB) Then Print "Same"
```

Note also that the *Ports* property itself is not assigned by reference. So consider the following:

```
Dim SerialIP
Dim PortsA
Dim PortsB
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set PortsA = SerialIP.Ports
Set PortsB = SerialIP.Ports
```

```
` Wrong
If (PortsA Is PortsB) Then Print "Same"

` Wrong also; the default value for PortList requires an index, so
` this will cause a syntax error
If (PortsA = PortsB) Then Print "Same"

` If you iterate PortsB after this next statement, it will
` NOT contain a COM20, since PortsA IsNot PortsB
PortsA.Add("COM20")
```

The PortList "ALL" Member

In addition to being able to apply functions and properties to a single port, the *Ports* container itself can be queried for a special port called "ALL". Any function or attribute applied to "ALL" is applied to all port objects in the container. This is useful for setting all ports to the same configuration. For example:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
SerialIP.Ports("ALL").ResetToDefault()
SerialIP.Ports("ALL").TelnetProtocol = TSTelnet_Telnet
SerialIP.Ports("ALL").EncryptionProtocol = TSEncrypt_Any
SerialIP.Ports("ALL").RestoreFailedConnections = FALSE
```

Note that properties are write-only on the "ALL" port, so any attempt to query the value of one of the above properties generates an error.

4. ITSPortSettings

The ITSPortSettings interface provides properties and functions to manage each redirected COM port. It implements properties and functions that correspond to the port-specific fields on the right-hand side of the Control Panel main window. The tables in the following sections show the relationship between elements of the interface and the GUI, as well as default values for each element.

ITSPortSettings Properties

Property	Type	GUI Field
Name	String	
ServerAddress	String	IP Address
ServerPort	WORD	Port Number
OutboundEnabled	Boolean	Connect to server
IncomingPort	WORD	Port Number
InboundEnabled	Boolean	Accept Connections
TelnetProtocol	TSTelnetProtocolType	Connection Protocol
DCDEmulation	TSSignalEmulation	DCD Emulation
DSREmulation	TSSignalEmulation	DSR Emulation
CTSEmulation	TSSignalEmulation	CTS Emulation
DTRIsModemEscape	Boolean	DTR is Modem Escape
RestoreFailedConnections	Boolean	Restore Failed Connections
UserAuthentication	TSAAuthenticationType	Use Credentials From
Username	String	Username
Password	String	Password
EncryptionProtocol	TSEncryptionType	Use SSL/TLS Encryption with

Below is a short sample which configures COM5:

```
Dim SerialIP
Dim SerialPort
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set SerialPort = SerialIP.Ports("COM5")

` Set the Address and Port
SerialPort.ServerAddress = "10.0.1.29"
SerialPort.ServerPort = 7000

` Configure for Telnet and Prompt-On-Port-Open Authentication
SerialPort.TelnetProtocol = TSTelnet
SerialPort.UserAuthentication = TSAuth_PromptAtPortOpen

` Turn on Connection Restore
SerialPort.RestoreFailedConnections = TRUE
```

ITSPortSettings Default Values

Property	Range	Default
ServerAddress	????	(empty)
ServerPort	0-65535	0
OutboundEnabled	TRUE/FALSE	TRUE
IncomingPort	0-65535	0
InboundEnabled	TRUE/FALSE	FALSE
TelnetProtocol	TSTelnet_None (0) TSTelnet (1) TSTelnet_CrPadding (2)	TSTelnet_None
DCDEmulation	TSSignal_None (0) TSSignal_AlwaysHigh (1) TSSignal_AlwaysLow (2) TSSignal_TrackConnection (3)	TSSignal_None
DSREmulation	TSSignal_None (0) TSSignal_AlwaysHigh (1) TSSignal_AlwaysLow (2) TSSignal_TrackConnection (3)	TSSignal_None
CTSEmulation	TSSignal_None (0) TSSignal_AlwaysHigh (1) TSSignal_AlwaysLow (2) TSSignal_TrackConnection (3)	TSSignal_None
DTRIsModemEscape	TRUE/FALSE	FALSE
RestoreFailedConnections	TRUE/FALSE	FALSE
UserAuthentication	TSAuth_None (0) TSAuth_PromptAtLogin (1) TSAuth_Windows (2) TSAuth_Fixed (3) TSAuth_PromptAtPortOpen (4)	TSAuth_None
Username		(empty)
Password		(empty)
EncryptionProtocol	TSEncrypt_None (0) TSEncrypt_SSLv3 (1) TSEncrypt_TLSv1 (2) TSEncrypt_Any (3)	TSEncrypt_None

ITSPortSettings Property Notes

Name Property

The Name property (i.e., "COM5") is read-only, and has no corresponding field in the GUI. In addition, this property is returned as the default value for the object in VB/VBScript, for example:

```
Dim SerialIP
Dim SerialPortA
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set SerialPortA = SerialIP.Ports("COM5")

` Prints "COM5"
Print SerialPortA.Name

` Also prints "COM5", since the "Name" property is the default value
Print SerialPortA
```

Username/Password Properties

The Username and Password properties are only used when UserAuthentication is set to TSAuth_Fixed or TSAuth_PromptAtLogin. For security reasons, the Password property is write-only.

ITSPortSettings Functions

ResetToDefault()

Resets the properties in the ITSPortSettings object to their default values. The ServerAddress and ServerPort are not changed.

CopySettingsFrom(String *COMPortName*)

Copies all of the settings from the COM port specified by *COMPortName* to the COM port managed by the current PortSettings object. For example:

```
Dim SerialIP
Dim SerialPort6
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")
Set SerialPort6 = SerialIP.Ports("COM6")

` Copy all the settings from COM5 to COM6
SerialPort6.CopySettingsFrom("COM5")
```

As an alternative to the CopySettingsFrom() function, the *Ports* container supports assignment. I.e., all the values from one PortSettings object can be assigned to the other with a single operation:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Object Assignment of COM6 to COM5:
Set SerialIP.Ports("COM5") = SerialIP.Ports("COM6")
```

The “ALL” port may also be used in an assignment. The example below copies the settings on COM6 to all ports:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Object Assignment of COM6 to all other ports:
Set SerialIP.Ports("ALL")= SerialIP.Ports("COM6")
```

5. ITSProxySettings

The ITSProxySettings interface implements properties and functions that correspond to the “Proxy Server” tab in the “Advanced” dialog. The tables in the following sections show the relationship between elements of the interface and the GUI, as well as default values for each element.

ITSProxySettings Properties

Property	Type	GUI Field
Protocol	TSProxyProtocol	Protocol Type
Address	String	IP Address of Server
Port	WORD	Port Number
Username	String	Username
Password	String	Password

Below is a short sample which configures proxy settings on Serial/IP:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Set for HTTPS Proxy
SerialIP.Proxy.Protocol = TSProxy_HTTPS

` Set the username and password
SerialIP.Proxy.Username = "user"
SerialIP.Proxy.Password = "super-secret"
```

ITSProxySettings Default Values

Property	Range	Default
Protocol	TSProxy_None (0) TSProxy_HTTPS (1) TSProxy_SocksV4 (2) TSProxy_SocksV5 (3)	TSProxy_None
Address		(empty)
Port	0-65535	(see note)
Username		(empty)
Password		(empty)

ITSProxySettings Property Notes

Port Property

The *Port* property is set automatically by setting the Protocol attribute, but can be overridden afterwards if necessary. The default values are shown on the following table:

Protocol	Port
HTTPS	8080
Socks V4	1080
Socks V5	1080

The following sample shows overriding of the proxy port:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Set to SOCKS 5
SerialIP.Proxy.Protocol = TSProxy_SocksV5

` Use a non-standard port
SerialIP.Proxy.Port = 5572
```

Password Property

The *Password* property specifies the password to provide to the proxy server. It is ignored for Socks V4, and is write-only for security purposes.

ITSProxySettings Functions

ResetToDefault()

Resets the properties in the ITSProxySettings object to their default values.

6. ITSSSLSettings

The ITSSSLSettings interface implements properties and functions that correspond to the “SSL Encryption”, “SSL Authentication”, and “SSL Certificate” tabs in the “Advanced” dialog. The tables in the following sections show the relationship between elements of the interface and the GUI, as well as default values for each element.

ITSSSLSettings Properties

Property	Type	GUI Field
MinKeyLength	WORD	SSL Encryption Minimum
MaxKeyLength	WORD	SSL Encryption Maximum
CipherSet	BitMask	SSL Encryption Encryption Ciphers
CipherList	String	SSL Encryption Cipher Set Preview
CertificateAuthorityFile	String	SSL Authentication Filename
RequireValidatedCertificate	Boolean	SSL Authentication Require Validated Certificate
CertificateMatch	String Container	SSL Authentication Validation Criteria
CertificateEnabled	Boolean	SSL Certificate Supply Certificate
CertificateFile	String	SSL Certificate Certificate File

Below is a short sample which configures SSL settings on Serial/IP:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Configure for a minimum of 56 bits, and no maximum
SerialIP.SSL.MinKeyLength = 56
SerialIP.SSL.MaxKeyLength = 0

` Allow only RC4, AES, or DES
SerialIP.SSL.CipherSet = TSCipher_RC4 or TSCipher_AES or TSCipher_DES

` Require a certificate and match on the IP address of the server
SerialIP.SSL.RequireValidatedCertificate = TRUE
SerialIP.SSL.CertificateMatch("CommonName") = "%i"

` Use the built-in certificate authorities
SerialIP.SSL.CertificateAuthorityFile = ""
```

ITSSSLSettings Default Values

Property	Range	Default
MinKeyLength	0-65535	128
MaxKeyLength	0-65535	0 (see notes)
CipherSet	TSCipher_RC2 (0x01) TSCipher_RC4 (0x02) TSCipher_DES (0x04) TSCipher_3DES (0x08) TSCipher_AES (0x10)	0x1f
CipherList		(read-only)
CertificateAuthorityFile		(empty)
RequireValidatedCertificate	TRUE/FALSE	FALSE
CertificateMatch	(see notes)	(see notes)
CertificateEnabled	TRUE/FALSE	FALSE
CertificateFile		(empty)

ITSSSLSettings Property Notes

CertificateAuthorityFile Property

This property is a fully-qualified pathname to the file which contains the trusted certificate authorities. May also be a null or empty string, in which case the built-in certificate authorities are used for certificate validation.

CertificateMatch Property

A container of strings, indexed by one of the following certificate field names:

```
Country
State
Locality
Organization
OrganizationalUnit
CommonName
EmailAddress
```

Assigning a string value to one of the members causes SSL to test an incoming certificate for that value. Assigning a null or empty value to one of the members causes SSL to ignore that field in an incoming certificate.

When this container is enumerated rather than accessed by index, each non-empty member is returned as a string in the form "*certificatefield = value*". Each member that is empty (and therefore is ignored when checking incoming certificates), will return "*fieldname IGNORED*".

The default for all members of the container is an empty value, except for “CommonName” which contains “%h”.

An E_INVALIDARG error will occur if the array is indexed by a name other than one of those listed above.

Code sample:

```
` Configure the certificate matching
SerialIP.SSL.RequireValidatedCertificate = TRUE
SerialIP.SSL.CertificateMatch("CommonName") = "foo.com"
SerialIP.SSL.CertificateMatch("EmailAddress") = ""
SerialIP.SSL.CertificateMatch("Country") = "USA"

` Print the matching for Country
Print SerialIP.SSL.CertificateMatch("Country")
```

MinKeyLength/MaxKeyLength Properties

If the MinKeyLength property is set to zero, it means no lower limit is in force. If the MaxKeyLength property is set to zero, it means no upper limit is in force.

CipherSet Property

This property is a bitmask which represents the list of ciphers that will be negotiated. To set more than one cipher, combine the values using bitwise OR. For example:

```
` Configure the ciphers
SerialIP.SSL.MinKeyLength = 128
SerialIP.SSL.CipherSet = TSCipher_RC4 or TSCipher_AES
```

CipherList

This read-only property is a string which enumerates the eligible ciphers based on MinKeyLength, MaxKeyLength, and CipherSet. Each cipher is terminated by CRLF.

CertificateFile

Full-qualified pathname of the certificate to supply to the remote endpoint. This property may be safely set to a NULL or empty string if clearing the filename (in addition to disabling certificates) is desired.

ITSSSLSettings Functions

SetCertificatePassword(String *password*, Boolean *save*)

This function sets the password to use when decrypting the certificate's private key to be the specified string *password*, and if the boolean parameter *save* is TRUE, the password is saved in the registry rather than just in memory (preserving it across reboots).

Similar to the *CertificateFile* property, setting a null or empty password will clear the password setting without resulting in an error.

The function will return TRUE if the password was set successfully or FALSE if the password did not work with the certificate specified in the *CertificateFile* property.

ValidateCertificate()

This function returns a success code if the certificate file and password are correct, or returns an error code if there is a problem. It will return one of the following values:

TSValidate_Success	(0)
TSValidate_CantAccessFile	(1)
TSValidate_InvalidPassword	(2)
TSValidate_Failed	(3)

TSValidate_Success – Certificate File and password are valid.

TSValidate_CantAccessFile – Unable to read certificate file, or file is not a valid certificate.

TSValidate_InvalidPassword – Certificate file is valid, but password is incorrect.

TSValidate_Failed – Validation failed (driver missing or not loaded).

ResetToDefault()

Resets the properties in the ITSSSLSettings object to their default values.

7. ITSOOptionSettings

The ITSOOptionSettings interface implements properties and functions that correspond to the “Options” tab in the “Advanced” dialog. The tables in the following sections show the relationship between elements of the interface and the GUI, as well as default values for each element.

ITSOOptionSettings Properties

Property	Type	GUI Field
DisableNagle	Boolean	Enable Nagle Algorithm
UpdateRouteTable	Boolean	Update Routing Table Upon COM Port Open
SyncPortOpen	Boolean	Synchronize with Server Upon COM Port Open
CPCKeepAliveTimer	DWORD	COM Port Control Keep-Alive Interval
MaxReconnectTimeout	DWORD	Maximum Connection Recovery Interval
DelayPortClosure	DWORD	Extend Server Connection by
UseWindowsDomain	Boolean	Include Domain in Windows Credentials
AlwaysReportCTS	Boolean	
AuthTerminator	String	
DefaultLoginPrompts	String Container	
DefaultPasswordPrompts	String Container	
LoginPrompts	String Container	
PasswordPrompts	String Container	

Below is a short sample which configures global option settings on Serial/IP:

```
Dim SerialIP
Set SerialIP = CreateObject("TacticalSoftware.SerialIPRedirector")

` Disable the Nagle Algorithm
SerialIP.Options.DisableNagle = TRUE

` Turn off port closure delay
SerialIP.Options.DelayPortClosure = 0

` Turn off Route Table updating
SerialIP.Options.UpdateRouteTable = FALSE
```

ITSOptionSettings Default Values

Property	Range	Default
DisableNagle	TRUE/FALSE	FALSE
UpdateRouteTable	TRUE/FALSE	TRUE
SyncPortOpen	TRUE/FALSE	TRUE
CPCKeepAliveTimer	0-4294967295	60000 (60 seconds)
MaxReconnectTimeout	0-4294967295	30000 (30 seconds)
DelayPortClosure	0-4294967295	8000 (8 seconds)
UseWindowsDomain	TRUE/FALSE	TRUE
AlwaysReportCTS	TRUE/FALSE	FALSE
AuthTerminator		"\n" (Linefeed)
DefaultLoginPrompts		"login: " "username: " "Username:" "login :" "Login:" "user:"
DefaultPasswordPrompts		"password:" "Password:" "Enter PASSCODE"
LoginPrompts		(empty)
PasswordPrompts		(empty)

ITSOptionSettings Property Notes

AlwaysReportCTS Property

When set to FALSE, client does not request that server send CTS status when hardware flow control is requested by the application. When set to TRUE, client always requests CTS status from server.

AuthTerminator Property

String which is appended to username and password during user authentication. Default is linefeed (decimal ASCII 10). Recognizes C-style "\r", "\n", and "\t" escape sequences.

DefaultLoginPrompts/DefaultPasswordPrompts Properties

Container of strings which lists the login or password prompts recognized by the driver for user authentication. This represents the list of prompts that is installed by default. Read-only.

LoginPrompts/PasswordPrompts Properties

Container of strings which lists additional login/password prompts recognized by the driver for user authentication. This represents the list of prompts that has either been updated by the user or an OEM customized build.

ITSOOptionSettings Functions

ResetToDefault()

Resets the properties in the ITSOOptionSettings object to their default values.