CSMail 1.2 Release Notes. 1 January 2002.

New features

- Internal support for SOCKS Version 4/5 and standard pop3 and smtp proxy servers.

- Consistent proxy/firewall support.

- New SendMessageEx method on SMTPClient object allows developers to define the SMTP envelope independently of the Recipient and Originator MIME fields.

Technical notes

Version 1.2 defines new interfaces, ISMTPClient3 and IPOP3Client2, as the default interfaces on the SMTPClient and POP3Client objects. To ensure binary compatibility with the previous interfaces the new interfaces are identical to those previous interfaces with the exception of the new methods.

Quick troubleshooting notes for Visual Basic Developers

VB Error 430 'Object doesn't support Automation' when creating SMTPClient or POP3Client object
VB Error 91 'Object variable or With block variable not set' when calling any method on SMTPClient or POP3Client Object

You may be running an application developed with Version 1.2 of the library on a system with Version 1.1 or 1.0 installed. Install Version 1.2 on the system.

VB Error 438 'Object doesn't support this property or method' when calling SetProxyInfo or SendMessageEx

Your application is calling SetProxyInfo() or SendMessageEx() on one of the previous interfaces. You may have compiled your application while Version 1.1 or 1.0 of the library was installed on your system. Install version 1.2 of the library and recompile your application.

SOCKS and other proxy/firewall support

With version 1.2 CSMail provides a consistent model for proxy and firewall support. This approach is achieved through the SetProxyInfo methods on the SMTPClient and POP3Client objects together with a new object for each firewall/proxy methodology. The methodology specific objects will allow us to extend the firewall/proxy support without changing the interfaces on the main client objects.

This release provides SOCK4ProxyInfo, SOCK5ProxyInfo, POP3ProxyInfo and SMTPProxyInfo objects.

SMTPClient.SetProxyInfo Method
POP3Client.SetProxyInfo Method

Declaration: SetProxyInfo(*ProxyInfo* as Object)

Parameters:

ProxyInfo   *An object providing information required to connect to a server through a proxy server. Currently allowed objects are:*

- o   SOCK4ProxyInfo object

- o   SOCK5ProxyInfo object

- o   POP3ProxyInfo object

- o   SMTPProxyInfo object

Description

Informs the SMTPClient or POP3Client object about the proxy/firewall connection required by the caller.

Applications should call SetProxyInfo after creating the object but before a call to Connect(). The parameters to the subsequent call to Connect() should be those appropriate for the actual POP3 or SMTP server.

Return Value

There is no return value from this method. If any error occurs while a message is being sent an error will be raised and should be handled through the VB/VBA/VBScript On Error mechanism.

Example

See the examples for the SOCK4ProxyInfo, SOCK5ProxyInfo, POP3ProxyInfo and SMTPProxyInfo objects

SOCK4ProxyInfo Object

Provides configuration information for a SOCKS V4 proxy server.

Developers wishing to provide SOCKS 4 support in their application should provide a means (typically a dialog box or configuration file) for the user to provide the information necessary to connect to a SOCKS 4 server. Developers may wish to advise end users to consult their network administrators for this information.

ProxyAddress Property

Declaration: ProxyAddress as String

Description

The address of the of the SOCKS 4 proxy server. *The host may be specified either as a dotted IP address (e.g. 127.0.0.1) or as a host name (e.g. socks4.mydomain.com)*

ProxyPort Property

Declaration: ProxyPort as long

Description

The port on which to connect to the proxy server. The default is 1080 (the standard SOCKS proxy server port).

ProxyUserID Property

Declaration: ProxyUserID as String

Description

The Socks 4 UserID.

Example

```
' ------------------------------------------
' Create SOCK4ProxyInfo object and set values
' In production code these values will come
' from the user.
' ------------------------------------------

Set S4=CreateObject("CSMail.SOCK4ProxyInfo")
S4.ProxyAddress="sock4.mydomain.com"
' (The library will use sensible defaults for ProxyPort and ProxyUserId)


' ------------------------------------------
' Create POP3Client object
' ------------------------------------------
Set pop=CreateObject("CSMail.Pop3Client")


' ------------------------------------------
' Pass Proxy information to client
' ------------------------------------------
call pop.SetProxyInfo(S4)


' ------------------------------------------
' Connect to the pop server.
' ------------------------------------------
call pop.Connect("pop3.mydomain.com","username","secret")
```

SOCK5ProxyInfo Object

Provides configuration information for a SOCKS V5 proxy server.

Developers wishing to provide SOCKS 5 support in their application should provide a means (typically a dialog box or configuration file) for the user to provide the information necessary to connect to a SOCKS 5 server. Developers may wish to advise end users to consult their network administrators for this information.

ProxyAddress Property

Declaration: ProxyAddress as String

Description

The address of the of the SOCKS 5 proxy server. *The host may be specified either as a dotted IP address (e.g. 127.0.0.1) or as a host name (e.g. socks5.mydomain.com)*

ProxyPort Property

Declaration: ProxyPort as long

Description

The port on which to connect to the proxy server. The default is 1080 (the standard SOCKS proxy server port).

ProxyUsername Property

Declaration: ProxyUsername as String

Description

The Socks 5 Username.

ProxyPassword Property

Declaration: ProxyPassword as String

Description

The Socks 5 Password.

Example

```
' ------------------------------------------
' Create SOCK5ProxyInfo object and set values
' In production code these values will come
' from the user.
' ------------------------------------------

Set S5=CreateObject("CSMail.SOCK5ProxyInfo")
S5.ProxyAddress="sock5.mydomain.com"
S5.ProxyPort=1080
S5.ProxyUsername="proxyuser"
S5.ProxyPassword="proxysecret"


' ------------------------------------------
' Create POP3Client object
' ------------------------------------------
Set pop=CreateObject("CSMail.Pop3Client")


' ------------------------------------------
' Pass Proxy information to client
' ------------------------------------------
call pop.SetProxyInfo(S5)


' ------------------------------------------
' Connect to the pop server.
' ------------------------------------------
call pop.Connect("pop3.mydomain.com","username","secret")
```

POP3ProxyInfo Object

Provides configuration information for a standard pop3 proxy server.

Developers wishing to provide pop3 proxy support in their application should provide a means (typically a dialog box or configuration file) for the user to provide the information necessary to connect to a pop3 proxy server. Developers may wish to advise end users to consult their network administrators for this information.

ProxyAddress Property

Declaration: ProxyAddress as String

Description

The address of the of the pop3.proxy server. *The host may be specified either as a dotted IP address (e.g. 127.0.0.1) or as a host name (e.g. popproxy.mydomain.com)*

ProxyPort Property

Declaration: ProxyPort as long

Description

The port on which to connect to the proxy server. *The normal port for connection to a pop3.proxy server is 110, the same as true POP3 server.*

ProxySeperatorChar Property

Declaration: ProxySeperatorChar as String

Description

Seperator character

Example

```
' -------------------------------------------
' Create POP3ProxyInfo object and set values
' In production code these values will come
' from the user.
' -------------------------------------------

Set P3Proxy=CreateObject("CSMail.POP3ProxyInfo")
P3Proxy.ProxyAddress="pop3proxy.mydomain.com"
P3Proxy.ProxyPort=110


' -------------------------------------------
' Create POP3Client object
' -------------------------------------------
Set pop=CreateObject("CSMail.Pop3Client")


' -------------------------------------------
' Pass Proxy information to client
' -------------------------------------------
call pop.SetProxyInfo(P3Proxy)


' -------------------------------------------
' Connect to the pop server.
' -------------------------------------------
call pop.Connect("pop3.mydomain.com","username","secret")
```

SMTPProxyInfo Object

Provides configuration information for a standard smtp proxy server.

Developers wishing to provide smtp proxy support in their application should provide a means (typically a dialog box or configuration file) for the user to provide the information necessary to connect to a smtp.proxy server. Developers may wish to advise end users to consult their network administrators for this information.

ProxyAddress Property

Declaration: ProxyAddress as String

Description

The address of the of the smtp proxy server. *The host may be specified either as a dotted IP address (e.g. 127.0.0.1) or as a host name (e.g. popproxy.mydomain.com)*

ProxyPort Property

Declaration: ProxyPort as long

Description

The port on which to connect to the proxy server. *The normal port for connection to an smtp proxy server is 25, the same as true POP3 server.*

Example

```
' -------------------------------------------
' Create SMTPProxyInfo object and set values
' In production code these values will come
' from the user.
' -------------------------------------------

Set SMTPProxy=CreateObject("CSMail.SMTPProxyInfo")
SMTPProxy.ProxyAddress="smtp3proxy.mydomain.com"
SMTPProxy.ProxyPort=25
 ' -------------------------------------------
' Create SMTPClient object
' -------------------------------------------
Set smtp=CreateObject("CSMail.SMTPClient")


' -------------------------------------------
' Pass Proxy information to client
' -------------------------------------------
call smtp.SetProxyInfo(SMTPProxy)


' -------------------------------------------
' Connect to the pop server.
' -------------------------------------------
call smtp.Connect("smtp.mydomain.com")
```

SMTP Envelope specification

The SendMessage method of the SMTPClient object makes the assumption that the developer wishes to send the email to the recipients listed in the To, CC and BCC MIME records. While this is appropriate for many mail applications it is occasionally necessary for the developer to specify the addresses to which the mail is actually sent independently of these MIME fields. Perhaps the best example of this requirement of this is a simple mail forwarding application where messages need to be forwarded to another address with the message headers unchanged. The SendMessageEx method on the SMTPClient object provides a simple and fast implementation route to this functionality.

SMTPClient.SendMessageEx Method

Declaration: SendMessageEx(*Message* as IMessage, *EnvelopeTo* as String [,*EnvelopeFrom* as String])

Parameters

Message	*A message object. See the Message Object reference for details of the Message Object.*
EnvelopeTo	*The addresses to which the messages should be sent.*
EnvelopeFrom	*The address from which the message should be sent.*

Description

Send a message though the SMTP server. Connect () or ConnectESMTP () must be called before this method is called. SendMessageEx () may be called repeatedly before Close () if there are multiple messages to be sent.

Return Value

There is no return value from this method. If any error occurs while a message is being sent an error will be raised and should be handled through the VB/VBA/VBScript On Error mechanism.

Advanced

While sending a message the object will:

- Issue the SMTP RCPT TO commands for each address in the list provided in the EnvelopeTo parameter.

- Issue an SMTP MAIL FROM command for the address in the EnvelopeFrom parameter. There should be exactly one address in this list. If there is not exactly one address in the list an error will be generated.

Issue an SMTP DATA command and transmit the message body to the SMTP server, encoding attachments appropriately if required.

Example

```
' --------------------------------------------
' Assume we've picked up a message (msg) from a
' a Pop3 server and want to forward it
' verbatim to another user.
' --------------------------------------------

' Create SMTP Client and connect…
Set smtp=CreateObject("CSMail.SMTPClient")
call smtp.Connect("localhost")

' Forward verbatim (all headers intact) to a local
' user…
call smtp.SendMessageEx(msg,"localuser@localhost")

' Clean up
call smtp.Close
```