

Trumpet Winsock Version 3.0











By Peter R. Tattam

Copyright (C) 1996 by Peter R. Tattam &

Trumpet Software International Pty Ltd

ACN 070 065 860

All Rights Reserved

-  [Introduction](#)
-  [Using the Trumpet Winsock over SLIP/PPP](#)
-  [Using the Trumpet Winsock over a packet driver](#)
-  [Menu Commands](#)
-  [Scripting Language](#)
-  [Trumpet Winsock Error Messages](#)
-  [Trumpet Winsock FAQ](#)
-  [Glossary](#)
-  [Contact Details and Online Services](#)
-  [Registration Details](#)

Introduction

Thank you for using the Trumpet Winsock. It is through the kind support of our users that quality networking software has been available at affordable prices to the Internet community.

You may use the Trumpet Winsock for 30 days to evaluate its usefulness. If at the end of that time you are satisfied with the Trumpet Winsock as a product, you should register it.

See [Registration Details](#) for more information.

- [☐ About this documentation](#)
- [☐ What is the Trumpet Winsock?](#)
- [☐ Capabilities of the Trumpet Winsock](#)
- [☐ What's new in Version 3.0?](#)
- [☐ License Terms and Conditions](#)
- [☐ Installation requirements](#)

About this documentation

This documentation has been revised for Trumpet Winsock Version 3.0 by Rob Torok.

Originally written as a Word for Windows 6.0 document, it is available in a variety of formats (including an online Help file, [trumpwsk.hlp](#)) from our [Web](#) and [FTP](#) sites.

Throughout this help file, “Winsock” refers to the Trumpet Winsock, and “winsock.dll” refers to the winsock.dll that comes with the Trumpet Winsock unless otherwise specified.

Note that the Trumpet Winsock’s help file has a non-standard **Menu** button. This is particularly useful for getting around from topic to topic. If you ever get lost, try using the **Menu** button to go up a level at a time.

What is the Trumpet Winsock?

The Trumpet Winsock is a Windows Sockets 1.1 compatible TCP/IP stack that provides a standard networking layer for many Windows networking applications to use, and has itself been a major vehicle in achieving widespread use of Windows Sockets 1.1.

Capabilities of the Trumpet Winsock

The Trumpet Winsock is supported under the following configurations and associated features.

1. Dial up SLIP and PPP connections, featuring:
 - Demand load dialling.
 - A Scripting Language for automatically logging in and out of your SLIP/PPP server.
 - Dynamic IP address assignment.
 - PPP options: PAP and CHAP.
- Offline use of Winsock applications.
2. Connection to local area network by way of a packet driver or shim and WINPKT under enhanced mode.
 - BOOTP and RARP.

What's new in Version 3.0?

Significant modifications made to the Trumpet Winsock since Version 2.1 include a few bug-fixes and:

- Revised documentation including online [help](#).
- Improved online [Internet registration](#) system.
- New scripting commands.
- More flexible demand loading with the [Login on network activity](#) feature.
- An installer!

Here is an exhaustive list of changes to the Trumpet Winsock since Version 2.1:

1. PPP processor modifications.
 - Added CHAP support.
 - Corrected some problems with PAP.
 - Fixed magic number bug in ECHO reply.
 - Added status message box when CHAP or PAP fails.
2. TCP protocol modifications
 - Improvements to reduce processing for out of sequence packets.
 - Blocking sockets improvements (see IP protocol modifications).
 - Correct excessive resend bugs.
 - Allow send buffer size > 32K (up to just under 65K) (fixes 32 bit Netscape problem).
3. TCP Manager (tcpman.exe) modifications
 - Extensive changes to the scripting language.
 - Changes to menus (special/register, dialler/profile, etc).
 - Changes to dialogs.
 - Added extra dialler option (dial on demand).
 - Error messages on startup displayed as dialog.
 - Improvements to the login/logout state machine for SLIP/PPP.
 - Fixed bug where tcpman crashes if wm_quit sent to tcpman.
 - Improvements to shut down operations to prevent system crashes (not 100% perfect yet, but close).
4. Packet driver module
 - Decreased minimum packet buffers from 4 buffers to 2 buffers.
 - Slightly altered error messages for error dialogs.
5. SLIP/PPP comms driver module
 - Allow reconfiguration without restarting Winsock.
 - Return PPP link failed state to WSASStartup.
 - Allow dial on demand for network activity.
 - General modifications to allow for PAP/CHAP login failure to give link failed state.
 - Improved error message reporting.
6. DNS processing.
 - Fixed bug in processing of compressed names. (only for outgoing data so not relevant to normal use of winsock)
 - Search will abort if network down.
 - Fixed gethostname to update cached hostname if IP address is changed.
 - Fixed async getXbyY calls to keep local copy of data. Fixes random Netscape host lookup errors.
7. UDP protocol modifications
 - Blocking sockets improvements (see IP protocol modifications).

8. IP protocol modifications

- Enable waitmessage processing to make the Winsock event driven rather than poll driven. (experimental, and is not 100% Winsock 1.1 compliant, does not work correctly for win32 tasks under Win95) Improves the performance of Windows when using blocking sockets.
- Win95 processing modifications to prevent reentrancy problems.
- Allow tcpman to abort bootp.
- Make sure that return IP address for ICMP ECHO REPLY is valid to prevent network storms.
- Miscellaneous alterations to blocking hook processing to be consistent with win32 multithreaded usage under Win95.

9. WINSOCK interface (winsock.dll)

- Detect more connection failed conditions (eg. PPP link failed errors).
- Reentrancy check on all socket calls for Win95 function.
- Blocking select call has waitmessage function.
- Return correct winsock information for WSAStartup errors.
- Kill all active sockets if tcpman.exe is unloaded.

License Terms and Conditions

The following terms and conditions apply to your use of the Trumpet Winsock program. Please read these terms before continuing the registration procedure. By using a registered copy of the Trumpet Winsock program you accept the terms and conditions below. If you do not accept the terms and conditions of this license, return this package to Trumpet Software International Pty Ltd and destroy all further copies of the product.

1. TSI grants the registered user of the software ("You") a non-exclusive, non-transferable and perpetual licence to use the object code comprised in the software. You must not have installed at any one time more than the number of licensed copies of the software.
2. **You acknowledge that the software is supplied "AS IS" and that TSI not making, and that no person acting on behalf of TSI has made, any warranty or representation as to the Software's functionality, performance characteristics, merchantability or suitability for any particular purpose. All such warranties and representations are hereby excluded to the maximum extent permitted by law.**
3. To the extent permitted by law, TSI, its directors, servants, agents and licensors are excluded from all forms of direct, special, indirect, or consequential loss or damage (including loss of profits, loss of data and loss or damage that may reasonably be supposed to have been in the contemplation of the parties as of the date of this agreement as a probable result of an act or omission by TSI) arising out of or in connection with Your use of the Software, including without limitation any loss or damage caused by TSI's negligence. Where relevant legislation does not permit TSI's liability to be limited in this way then, if permitted by that legislation, TSI's liability will be limited to, at TSI's option, the replacement of the relevant goods or the supplying of relevant services again, or the payment of the cost of having the services supplied again. For the purposes of this Clause 3, TSI is or will be taken to be acting as agent or trustee on behalf of and for the benefit of its directors, servants, agents and licensors.
4. The agreement formed by TSI's acceptance of Your application for registration is governed by and will be construed in accordance with the laws of the State of Tasmania, Australia. The 1980 United Nations Convention on the International Sale of Goods does not apply to that agreement.
5. You acknowledge that all rights in the Software are owned by TSI or its licensors and shall not at any time make any statement or do anything which is inconsistent with that ownership. Without limitation, Your licence does not allow You to resupply the software to any other person or to distribute the software.
6. Your licence is limited to use for Your own internal requirements and does not permit You to use the Software to process the data of any other person or a part of a bureau service. If use for these purposes is required then You should contact TSI to make arrangements for the granting of an appropriate licence covering such uses.

Installation requirements

For basic installation of the Trumpet Winsock, you will need the following:

1. About 300-400k RAM (including 50-58k of global DOS memory) under normal conditions.
2. About 1000k disk space.
3. About 5 minutes to install the files.
4. Some knowledge of your network for the Winsock's setup screens.

NOTE: If you already have some kind of Internet software installed (eg Microsoft's Dial-up Networking), it is likely that some massaging of your system configuration will be required to allow the Trumpet Winsock to function correctly. In particular, if you already have a Winsock (winsock.dll and/or wsock32.dll) on your machine, then you may need to uninstall those files before the Trumpet Winsock will work.

Using the Trumpet Winsock over SLIP/PPP

Before applications can use the Winsock to access the Internet, the Winsock must dial into your Internet Service Provider and establish a SLIP or PPP connection.

This section describes how to configure the Trumpet Winsock for use over [SLIP](#) or [PPP](#).

Have a look at the relevant sections of the [Trumpet Winsock FAQ](#), or [Trumpet Winsock Error Messages](#) for more information on this topic.

- [!\[\]\(cf5be311f7b2821912d8009884508fa2_img.jpg\) Setting up the Winsock for SLIP/PPP](#)
- [!\[\]\(9804e70d96ff9fe9899b264c06a33cd7_img.jpg\) Logging in for the first time](#)
- [!\[\]\(4f49380f3d6bce047bc47b2072cc076f_img.jpg\) Logging in manually](#)
- [!\[\]\(73944fd4f6fb83e4c64013731d1820cc_img.jpg\) Configuring your profile](#)
- [!\[\]\(d8f7165d5a8d1eba426ea452457190e5_img.jpg\) Example profiles](#)
- [!\[\]\(f608c4821f4fa8f3141b1baf96fa88f9_img.jpg\) Configuring your login script](#)

Setting up the Winsock for SLIP/PPP

To enable your Winsock to function correctly it will be necessary to enter some details in the Setup screen. If you are unclear on any of them, try to seek some help from qualified Internet support staff - it will save you a lot of time.

1. Open the Trumpet Winsock and select Setup from the File menu to open the Setup dialog.
2. Select either SLIP or PPP.
 - Some details will be greyed out now. You need not try to fill them in.
3. Enter your IP address (if you've been assigned one), DNS server, and Domain suffix.
 - If your provider allocates IP addresses dynamically (or you are unsure), leave this as 0.0.0.0 for the moment.
4. If you're feeling particularly brave, you may like to tweak Demand load timeout, MTU, TCP RWIN, TCP MSS, and TCP RTO MAX.
 - These values can be tailored to suit your needs once you have things running, but you should be able to establish a connection using MTU: **576**, TCP RWIN: **2048**, and TCP MSS: **512**.
 - You may like to talk to your Internet Service Provider for advice on optimising these.
5. Open the Dialler Settings screen by clicking the Dialler Settings button.
6. Set the Comm port to the number of the comms port to which you've connected your modem.
 - Eg. If your modem is on port 2, select: **Com2**
7. Set the Baud rate that you would like your modem to run at.
 - Eg. If you've got a Vfast 28.8k modem, select: **38400**
8. Set Hardware handshaking and VJ header compression as required.
 - Hardware handshaking is recommended if your modem and modem cable support it.
 - Under SLIP, header compression may only be used if your service provider supports it. Under PPP, header compression is optional and recommended.
9. Select the kind of Online status detection your modem supports.
 - You will need to make sure that your modem has a default power on setting of **&c1** for DCD detection to function.
10. By default, all dialling is done with standard Parity and Word Size (8 bits, no parity). This may not work for you...
 - If your service provider does not use 8 bits with no parity, you will need to select Use Control Panel settings and set these parameters in Window's Control Panel.
11. When you have finished editing these screens, click on OK and if all goes well, the Trumpet Winsock will be initialised.
12. If your Internet provider has instructed you to use PAP or CHAP, then:
 - From the File menu, select PPP Options.
 - Enable PAP or CHAP as required and enter your username and password.
 - Press OK to continue.

You should now be ready to try logging in for the first time.

For more information about the options you've set, please see the sections covering the File/Setup and Dialler/Settings screens.

Logging in for the first time

There are two methods of logging into your Internet Service Provider: manual and automatic. A manual login involves entering the modem commands and server commands each time you wish to access the Internet.

Automatic logins are a feature of the Trumpet Winsock which allow you to configure login scripts that automate the manual login process. For more information, see [What's the difference between a manual and automatic login?](#)

When logging in for first time, try the following suggestions:

1. Run the sample login script ([login.cmd](#)) by selecting Login from the Dialler menu.
 - The first time it is launched you will be asked to enter phone number, username and password. You can change these later by selecting [Profile](#) from the Dialler menu.
 - If you receive a message saying “script completed” then your script may require no further work.
 - If you receive “script aborted” (possibly preceded by another error message) then you should read on...
2. Find out if your Internet provider has a preconfigured a Trumpet Winsock login script.
 - Since many Internet providers will have prepared scripts for connecting to their systems, it may be worth contacting your provider to see if they have a suitable script.
3. Try [Logging in manually](#) and then configuring your profile or login script.

Once you have got a working login script, **we recommend that you back up your login.cmd and trumpwsk.ini files.** To be totally safe, back up all the files in your Trumpet directory.

Please note that if you are upgrading from a previous release of the Trumpet Winsock, and already have a script that works, you should stick with it.

Logging in manually

Note that the Winsock will try to dial automatically when it starts unless No automatic login from Dialler/Settings is selected. To stop the script from executing, press the <Esc> key.

1. From the Dialler menu, select Manual Login.
 - You will now be able to send commands (as described in your modem's manual) to your modem using the "at" commands.
2. Confirm that your modem is ready by typing an appropriate command (or initialisation string).
 - For example, try entering: **atz <Enter>**
 - Each character should appear on the screen as you type them. If no characters appear on the screen, make sure that the modem is turned on and plugged in and that the modem is set to echo commands as they are entered. To enable echoing, type the following command: **atze1 <Enter>**
 - The modem should return: **OK**
 - If the characters are appearing on the screen, but the <Enter> isn't being taken, trying adjusting the baud rate in the Dialler/Settings screen. Check your modem's manual to confirm which baud rates are valid. For more suggestions, see the error message Modem is not responding.
3. Enter your modem's initialisation string, if required.
 - For example, try: **at&f&c1&k3 <Enter>**
4. Instruct the modem to start dialling, by typing: **atdt <number> <Enter>**
 - The modem should now dial into your service provider. If the line is busy you will need to repeat this step until you get a free line.
 - Use **atdtp** instead of **atdt** if your phone exchange uses pulse dialling.
5. Once connected to your provider's machine, you will usually be asked to enter you username and password. Type these and press <Enter> after each one.
6. If and when you are given a choice, select either SLIP or PPP depending on whether you have SLIP or PPP selected in the File/Setup screen.
 - Your screen may now start being filled with nonsense characters.
7. Press <Esc> to enable SLIP/PPP.
8. If you are using SLIP (*and not PPP*) and your IP address is allocated dynamically you will need to:
 - Select Setup from the File menu.
 - Enter the address you've been given in the IP address field.
- Press OK to continue.
9. Try connecting to a well known host IP address using a Winsock compliant program to see if all is well.
 - The program Trumpet Ping is suitable for this.
10. Once you have determined your login sequence, you can try Configuring your profile and/or Configuring your login script. Before trying to configure your profile you may find it very helpful to make a transcript of the login. To do this:
 - Select all the text in the Trumpet Winsock's display.
 - Copy the text by selecting Copy from the Edit menu.
 - Run Notepad (or another text editor or even an email program) and select Paste from the Edit menu.
 - Save, print, or even email the transcript as required.

Configuring your profile

This section describes the process of configuring the Login profile for the standard login script and provides some examples. Configuration of the profile should allow the Winsock to successfully dial into most SLIP/PPP servers, however it may necessary to edit the login script (login.cmd) directly for some.

1. Try Logging in manually first to determine what prompts are given by the server and what responses are required.
 - Be sure to make a transcript of the login, so that you have got something to refer to while editing the profile.
2. From the Dialler menu, select Profile.
3. Ensure that the username and password of your SLIP/PPP account and your service provider's dial-up phone number are entered correctly.
4. Open the Modem profile screen by clicking the Modem profile button.
 - Modify any of the relevant modem settings.
 - For example, change the init string if your modem needs a special one. Change **dt** with **dp** if pulse dialling is required.
 - Click on OK to continue.
5. Open the Server profile screen by clicking the Server profile button.
 6. Adjust the prompts, commands and responses according to the messages output by your SLIP/PPP server.
 - If the prompt that asks you for your username is "Enter your login:", you could change the Username prompt to: "login:".
 - If your server doesn't prompt for a service then leave the Command prompt field blank.
 - If you are unsure what prompts are output by your SLIP/PPP server, then compare your login transcript to the Example profiles given below.
7. Click on OK to continue when you have finished editing each of these screens.
 8. From the Dialler menu, select Login to try logging in with the new profile.
 9. If there is a problem with the login script it may indicate that one or more of the profile settings are incorrect or that the standard login script will need to be altered. See Configuring your login script for more information.
 - To obtain a full trace of the login script as it executes, edit the login.cmd file (by selecting Edit scripts from the Dialler menu) and add the following line near the top: trace on

Example profiles

Here are some examples login sessions and the corresponding Server profile settings.

Example PPP Server

Here is a typical PPP server login. In this example it is important that the username prompt be set to “ogin:” (or “login:”) rather than “sername:”.

```
SLIP DISABLED
atz
OK
at&f&c1&k3
OK
atdt123456
CONNECT 26400/ARQ

ISP login: myusername
Password:
PPP session from (123.456.789.100) to 123.456.789.156
beginning...~7F}#@!}!}!}4}"&} } } } %}&1QB}'"}
{"1}6~~7F}#@!}!}" }4}"&} } } } %}&1QB}'"}
{"&+~~7F}#@!}!}#} }4}"&} } } } %}&1QB}'"} {"o8~
PPP ENABLED
```

Note that since this server does not prompt for a service (eg SLIP or PPP) the Command prompt field must be left blank.

Server profile	value
Username prompt	ogin:
<u>Password prompt</u>	ssword:
<u>Command prompt</u>	
<u>SLIP command</u>	N/A
<u>SLIP response</u>	N/A
<u>PPP command</u>	N/A
<u>PPP response</u>	beginning..
<u>Scan for IP address</u>	N/A

Example SLIP Server

This is a fairly typical SLIP login with prompts for “Username:” and “Password:” and the requirement that “slip” be output after a “>” prompt.

```
SLIP DISABLED
atz
OK
at&f&c1&k3
OK
atdt123456
CARRIER 28800
```


PROTOCOL: LAP-M
COMPRESSION: V.42BIS
CONNECT 38400

M Y I N T E R N E T S E R V I C E P R O V I D E R

Welcome to our Internet Services. For registration or general enquiries, phone us during business hours.

Username: myusername
Password:
PORT56>slip
SLIP connected.
Server IP is 123.456.789.100, your address is 123.456.789.156
SLIP ENABLED

Note that since SLIP is being used, it is very important that the script is able to find the IP address. If the Scan for IP address field was left blank, the script would grab the first IP address it could find. In this particular example it would find the server's IP address (123.456.789.100) and incorrectly use that instead of the dynamically allocated IP address (123.456.789.156)!

The SLIP response could in fact be left blank in this situation, but is left in as a precaution to make sure that SLIP has been successfully started on the server.

Server profile	value
Username prompt	Username:
Password prompt	Password:
Command prompt	>
SLIP command	slip
SLIP response	connected.
PPP command	N/A
PPP response	N/A
Enable Scan for IP address	your address is

Configuring your login script

This section explains what changes might need to be made to the standard login script that cannot be accommodated by [Configuring your profile](#).

Your login script will be functioning correctly once it is able to login and return “script completed”. If it returns “[script aborted](#)”, your Winsock applications will not be able to connect to the Internet.

Here are some reasons why the standard login script might need to be altered:

1. The order of events when you dial in is different to the order expected in the standard login script. If this is the case you may need to try a different combination.
 - For example, some dial-up servers prompt for a service before requesting the username and password.
2. An additional password may be required.
 - For example, to output *mypassword* at the *ID Code:* prompt:

```
input %logintimeout "ID Code:"  
output "mypassword"\13
```

To access the login script, select [Edit scripts](#) from the Dialler menu, choose login.cmd, and press OK.

To obtain a full trace of the login script as it executes, insert the following command near the top of the script:

```
trace on
```

See also [Scripting Language](#).

Using the Trumpet Winsock over a packet driver




This section describes how to configure the Trumpet Winsock for use on a local area network (eg Ethernet) by way of a packet driver interface.

There are a number of ways that an application can access a network card. The most common methods are ODI, NDIS, and packet driver. Currently, the Trumpet Winsock supports a **packet driver interface only**, and **not ODI or NDIS**. To use the Trumpet Winsock over ODI or NDIS, a *packet driver shim* is required to provide a packet driver interface.

Furthermore, the Winsock requires winpkt (or a substitute, such as ndis3pkt) to make the packet driver (or packet driver shim) available within Windows.

Normally network cards will be supplied with a number of device drivers (including packet driver, ODI driver, and NDIS driver). If your network card was supplied without a packet driver and you are not using ODI or NDIS, you may like to investigate the Crynwr packet driver collection.

Have a look at the relevant sections of the Trumpet Winsock FAQ, or Trumpet Winsock Error Messages for more information on this topic.

-  [Installing a packet driver and WINPKT](#)
-  [Installing an NDIS or ODI packet driver shim](#)
-  [Setting up the Winsock for a packet driver or shim](#)

Installing a packet driver and WINPKT

To install a packet driver and winpkt, you will need lines in your autoexec.bat (or network batch file) that do the following:

1. Set up your packet driver (including vector, IRQ and I/O address).
2. Install winpkt using the same vector that the packet driver was installed on.

The following examples are intended to illustrate this.

Ne2000 packet driver using WINPKT

This example installs an ne2000 packet driver on vector 0x60 using IRQ 2 and I/O address 0x300.

```
ne2000 0x60 2 0x300
WINPKT 0x60
```

Western Digital 8-bit network card using WINPKT

This example installs a Western Digital 8-bit network card on vector 0x61 using IRQ 5 and I/O address 0x320.

```
wd8003e 0x61 5 0x320
WINPKT 0x61
```

Ne2000 packet driver with Novell NetWare access using WINPKT

Specification of the -n switch of the packet driver is important. Some packet drivers don't support this switch. In that case, you may be forced to use the ODIPKT shim instead. An example could be the Xircom Pocket Adaptor.

```
ne2000 -n 0x60 2 0x300
WINPKT 0x60
pdipx
netx
path c:\dos;c:\network\win31
f:
login
```

Installing an NDIS or ODI packet driver shim

Since the Winsock can only access a network card by way of a packet driver, a *packet driver shim* is required to provide a packet driver interface over NDIS and ODI.

Use the following table to determine which shim you will require and whether or not winpkt be need to be installed.

Network driver	Shim required	Winpkt needed	Example network OS
NDIS3 and ODI	<u>ndis3pkt</u>	no	Microsoft network (WfW or Win95) and Novell Netware
NDIS3	<u>ndis3pkt</u>	no	Microsoft network (WfW or Win95)
NDIS2	<u>dis_pkt</u>	yes	Windows for Workgroups
ODI	<u>odipkt</u>	yes	Novell Netware
ODI (over token ring)	<u>oditrpkt</u>	yes	Novell Netware

The odipkt, dis_pkt, and ndis3pkt shims are available by anonymous FTP and have directories dedicated to each of them at:

<ftp://hsdndev.harvard.edu/>

 [Ndis3pkt](#)

 [Odipkt](#)

 [Oditrpkt](#)

 [Dis_pkt](#)

Ndis3pkt

Ndis3pkt is a shim that provides a packet driver interface over NDIS3. It is available by anonymous FTP from:

`ftp://hsdndev.harvard.edu/ndis3pkt/`

Like winpkt, ndis3pkt provides a virtual packet driver interface under Windows, and in fact replaces winpkt altogether.

Odipkt

Odipkt is a shim that provides a packet driver interface over ODI. It is available by anonymous FTP from:

```
ftp://hsdndev.harvard.edu/odipkt/
```

See also winpkt and oditrpkt.

Here is an example network attach batch file and the corresponding net.cfg:

```
@echo off
cd \
lh lsl
lh \odi\ne2000
cd \net
lh ipxodi
lh odipkt
lh WINPKT 0x69
lh netx
path c:\dos;c:\net\win31
f:
echo on
login
```

The ordering of the frame protocols in your net.cfg is very important. For example:

```
Link Support
  Buffers 8 1586
  MemPool 16384
Link Driver NE2000
  Port #1 300 20
  Int #1 2
  Frame Ethernet_II
  Frame Ethernet_802.3
  Protocol IPX 0 Ethernet_802.3
```

Note that there is another version of odipkt which has been released by FTP Software.

Also note that there are two programs with the name ne2000.com. One is a packet driver and is referred to in another section. The one referred to in this section is an ODI driver - it doesn't provide a packet driver interface.

Oditrpkt

Oditrpkt is a shim that provides a packet driver interface over ODI on a token ring network. It is available by anonymous FTP from the Oklahoma State University:

`ftp://dc.cis.okstate.edu/pub/oditrpkt/`

Dis_pkt

Dis_pkt is a shim that provides a packet driver interface over NDIS2. It is available by anonymous FTP from:

ftp://hsdndev.harvard.edu/dis_pkt/

Dis_pkt currently only supports NDIS2, not NDIS3. If your network is based on NDIS3, then use ndis3pkt rather than dis_pkt.

If the Trumpet Winsock is required to operate over a Windows for Workgroups network, an installer is available to setup dis_pkt. The installer is not necessary if the Winsock is going to be used over a modem.

It is available free of charge from our FTP server:

<ftp://ftp.trumpet.com/winsock/wfwsetup/twsfwfg.zip>

This is provided AS IS with no warranty and is for personal use only.

Note that the Trumpet Winsock should be installed first, and that winpkt is still required.

Setting up the Winsock for a packet driver or shim

To enable your Winsock to function correctly it will be necessary to enter some details in the Setup screen. If you are unclear on any of them, try to seek some help from your network administrator - it will save you a lot of time.

The first time you run the Trumpet Winsock, the Setup screen will be opened automatically. You can access it later by selecting File from the Setup menu.

1. Make sure that your packet driver or packet driver shim is installed correctly.
 - If you haven't already done so, read Installing a packet driver and WINPKT or Installing an NDIS or ODI packet driver shim.
2. Enter your IP address, DNS server, and Domain suffix.
 - If you are on a network that uses BOOTP or RARP to determine its IP address, then enter **BOOTP** or **RARP** respectively.
3. Select Packet driver from the list of Driver types.
 - Some details will be greyed out now. You need not try to fill them in.
4. Enter your Vector, Netmask, and Gateway.
 - The vector is the one that your packet driver was installed under without the leading "0x". For example, if you used vector 0x60, enter **60**. Alternatively, enter **00** to instruct the Winsock to search for the packet driver.
5. If you're feeling brave, you may like to tweak MTU, TCP RWIN, TCP MSS, TCP RTO MAX, and Demand load timeout.
 - These values can be tailored to suit your needs. If you are on an Ethernet, then we suggest MTU: **1500**, TCP RWIN: **4096**, and TCP MSS: **1024**.
6. When you are done, click on OK and if all goes well, the Trumpet Winsock will be initialised.
 - You are now ready to start using the Winsock.


Menu Commands

This section describes the function of each menu command, and the options presented in each dialog box.

- File
- Edit
- Special
- Trace
- Dialler
- Help

File

 Setup

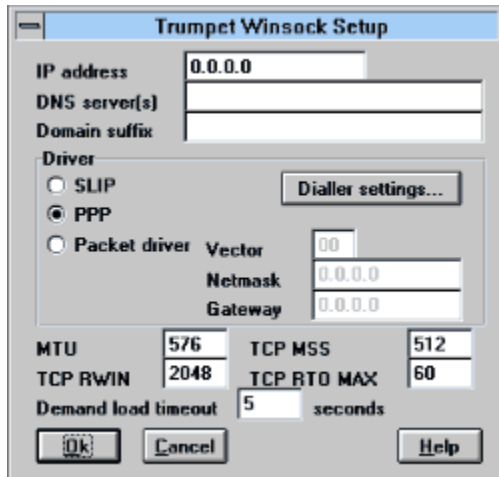
 Firewall setup

 PPP options

 Exit

Setup

Opens the Trumpet Winsock's Setup dialog box. Click on settings to see its definition.



For some guidance in how to set these see either [Setting up the Winsock for SLIP/PPP](#) or [Setting up the Winsock for a packet driver or shim](#).

- [IP address](#)
- [DNS server](#)
- [Domain suffix](#)
- [Driver](#)
- [SLIP](#)
- [PPP](#)
- [Packet driver](#)
- [Vector](#)
- [Netmask](#)
- [Gateway](#)
- [MTU](#)
- [TCP RWIN](#)
- [TCP MSS](#)
- [TCP RTO MAX](#)
- [Demand load timeout](#)

IP address

Determines the IP address of your computer.

To exist on the Internet your computer must have an IP address. This address will be allocated permanently or dynamically by your service provider.

If your service provider allocates IP addresses dynamically, this will usually be set automatically if you are using PPP. If you are using SLIP, the IP address will need to be entered either manually or by a login script.

Alternatively, the Winsock may be able to use either BOOTP or RARP to find your IP address if the services are online.

DNS server

The numerical IP address of your Internet Service Provider's domain name server.

A domain name server converts names into numeric IP addresses using the domain name system (DNS). You may provide more than one address by separating the addresses with spaces.

Domain suffix

The name of the domain in which your computer lives. This is used for looking up names via the name server.

You may provide more than one domain name by separating the names with spaces.

Driver

Determines the kind of network driver the Winsock will use to access the Internet.

SLIP

Serial Line Internet Protocol (SLIP) is a simple protocol which allows an Async serial connection to send Internet Protocol (IP).

See also PPP.

PPP

Point to Point Protocol (PPP) is a more complicated version of SLIP that includes error detection, password authentication and optional VJ header compression.

SLIP, CSLIP and PPP are all point-to-point protocols. They directly connect two computers or one and computer and a network port, usually over a modem connection. These protocols are most often used over dial up connections.

Packet driver

A packet driver is a piece of software that provides an interface between a network card (such as an ethernet adaptor) and communications software such as the Trumpet Winsock.

This is the method of network driver that the Winsock would normally use when configured for an ethernet.

Why is it called a packet driver? This is because modern networks send information using packets of information rather than sending information one byte or character at a time. For example, ethernet sends information in frames of up to 1514 bytes long. The reason for this is that information can be transmitted much more efficiently in packets.

Public domain packet drivers are available from the [Crynwr packet driver collection](#).

Vector

Sets the vector that your packet driver was installed under. This value should be entered without the leading "0x".

Eg. If you used vector 0x60, enter: **60**

Leaving the packet vector set to **00** instructs the Winsock to search for the packet driver.

A packet driver is only allowed to have a software interrupt vector in the range 0x60 to 0x7F. Normally, you will pick 0x60 as the default place to install your packet driver, but certain machine configurations may make that vector unavailable. Just choose one that is free - the packet driver should tell you if you can use it or not.

See vectors or more information.

Netmask

Sets your network mask.

A netmask is used for determining whether or not a destination IP address is on the same network.

Your netmask will often correspond to the class of your IP address. For example, if you have a Class C IP address, use: **255.255.255.0**

Gateway

Your default Internet gateway or router.

MTU

The Maximum Transmission Unit (MTU) of a network is the greatest amount of data that can be transferred in one physical frame on that network.

Fragmentation will occur if a packet is sent across a network that has an MTU smaller than the packet's frame length. This will lead to lower performance as fragments need to be reassembled. A work around for this would be to lower the Winsock's MTU to match the minimum MTU of the intervening networks. Since it is virtually impossible to determine the minimum MTU of the intervening networks, trial and error is usually the best way to find out.

As a starting point, try setting the MTU to the MTU of your network. For Ethernet, 1500 is the maximum, and is recommended. The unofficial MTU for SLIP servers is 1006.

TCP RWIN

The TCP Receive Window (RWIN) determines how much data the receiving computer is prepared to receive.

An RWIN set too large will result in greater loss of data if a packet is lost or damaged.

An RWIN set too small (ie $1 * MSS$) will be very slow, as each packet will have to be acknowledged before the next packet is sent. An RWIN of $2 * MSS$ causes problems with silly window avoidance.

The RWIN should usually be set to exactly 3 or 4 times the size of the TCP MSS.

TCP MSS

The TCP Maximum Segment Size (MSS) defines the largest segment of TCP data that the Winsock is prepared to receive on that connection. When a connection is established, the two ends agree to use the minimum of each end's advertised value.

If the MSS is too low, the data/header ratio will be low. Alternatively, a large MSS will lead to large IP datagrams. These will fragment when travelling across networks with small MTUs, thereby decreasing performance. For this reason, performance may sometimes be improved by dropping the value of your MTU.

Because headers are typically 40 bytes, your MSS should always be less at least 40 less than the MTU. As a rule of thumb, set the MSS to the greatest power of 2 that is at least 40 less than the MTU.

TCP RTO MAX

The TCP Retransmission Time Out Maximum (RTO MAX) sets an upper limit on the retransmission timeout.

It may be useful to lower this to perhaps 10 seconds when using interactive programs such Chat or Telnet software over a connection with a higher than average packet loss.

Demand load timeout

The demand load timeout is the number of seconds the Winsock stays loaded after all Winsock applications have finished with it. This is useful when the Winsock has been demand loaded and you wish to close the application that loaded the Winsock before loading another (and not have the Winsock logout in the meantime).

Note that this will only apply if the Winsock has been “demand loaded”. That is, the Winsock has been launched by another application.

If the Winsock hasn't been demand loaded, use the [Inactivity timeout](#).

See also [How do I get demand loading to work?](#).

Firewall setup

The Trumpet Winsock may be used with a [Socks 4.2](#) compatible [firewall](#).

Enable Firewall		Enables firewall setup.
Firewall Host:	IP address	This will be the IP address of the firewall that is visible to the local network.
	Port	The port that the firewall will be using to communicate to the Winsock with.
Firewall User ID		Your user ID string.
Local IP addresses:	Net	The address of a network that will be considered local.
	Mask	The netmask of the corresponding network.

PPP options

If your Internet Provider has instructed you to use either Password Authentication Protocol ([PAP](#)) or Challenge Handshake Authentication Protocol ([CHAP](#)) then select it from this screen and enter your username and password.

If you would like to be prompted for your password each time you login, then leave the password field blank.

The PAP/CHAP username and password should not be confused with the username and password used for a dial-up SLIP/PPP account.

Exit

Quits the Trumpet Winsock.

Edit

 Copy

 Clear





Copy

Copies selected text on the display to the clipboard.

Clear

Clears the Trumpet Winsock display.

Special

-  Info
-  Kill socket
-  Hide
-  Password registration
-  Internet registration
-  Update registration
-  License

Info

Displays a list of active sockets.

Kill socket

Allows any socket to be killed; use with caution.

Hide

Hides the Winsock from the task list. To make the Winsock reappear, select it's icon again.

Password registration

Once you have registered the Trumpet Winsock you will receive a registration name and password that should be entered here.

Since your password is generated from your registration name, **you do not get to choose your own registration password.**

See [Registration Details](#) for more information on and details of how to register.

Note that if you've registered an earlier version of the Winsock you will need to update your password. You can do this over the Internet by selecting [Update registration](#) from the Special menu.

Internet registration

This feature is for sending encrypted credit card details to Trumpet Software International via the Internet. This will only work if your Winsock is functioning and your network is visible to ours.

The Winsock will only permit you to send one registration request so make sure you get your details correct. **Do not use this feature as a test.**

You will receive a password once we have received an authorisation to debit your credit card. For this reason, **please allow five working days for your order to be processed.**

To register over the Internet:

1. Start the Trumpet Winsock (and dial into you provider if applicable), then select Internet registration from the Special menu.
2. Press OK to each message box once you have read them.
3. Fill out the details carefully and select Send registration request when you're finished.
 - If you get a "Registration Host not found" error it may be because our registration server was temporarily off line, your machine is behind a firewall, or because your domain cannot see ours for some reason. In this event, please use another method. See How to register for other options.
 - If the transfer is taking too long, it may be necessary to click on the **Abort transfer** button and try again later.

You can select **Query registration process** to see how your order is progressing. If your order has already been completed, then your password will be automatically transferred to your Winsock and the timelock will be removed. If you receive "Transaction has been declined or is invalid", then please contact our [Registration Department](#) by another method.

When your transaction has been completed, your password will be sent to you via email. You can enter you registrations name and password in the [Password registration](#) screen, which can be found under the Special menu.

If this method of registration is unsuccessful, please try another. See [How to register](#) from [Registration Details](#) for the alternatives.

Update registration

This feature allows the Winsock to automatically update your registration password over the Internet.

To use this feature:

1. Install Version 3.0 over a previously registered Trumpet Winsock.
 - OR...
2. Copy the trumpwsk.ini file from a previous registered Trumpet Winsock into the same directory as Trumpet Winsock 3.0.
3. Start the Trumpet Winsock Version 3.0 (and dial into you provider if applicable).
4. Select Update registration from the Special menu.

If you receive a “registration name not found” error message, then please contact our Registrations Department with details of how and when you originally registered.

License

Displays the License Terms and Conditions.

Trace

The following trace options provide diagnostic information about the Winsock.

Use with care since some applications may crash when the traces are active, in particular the DNS trace. Should a program crash with stack overflow (Error 202), the winsock.dll may remain loaded in memory even though tcpman.exe has exited. It is advisable to restart Windows if this happens and possibly even reboot your machine.

Also, timing measurements of the Winsock throughput will be severely affected by the trace options.

- TCP
- UDP
- IP
- ARP
- RARP
- Ethernet
- Extra detail
- Socket calls
- DNS
- Messages
- Comms
- PPP

TCP

Provides a dump of each TCP packet including:

```
time, source port->destn port, seq number, [ack number], [flags],  
wind number, [options], [data size]
```

UDP

Provides a dump of each UDP packet:

```
UDP src -> dst size
```

IP

Enables IP tracing.

ARP

Enables ARP tracing.

RARP

Enables RARP tracing.

Ethernet

Adds ethernet headers to IP, ARP and RARP traces.

Extra detail

Adds some extra detail to TCP, UDP and IP traces.

For TCP and UDP it provides a dump of the data. For UDP it also dumps broadcasts.

Socket calls

Traces each Winsock call. The subroutine parameters are displayed as well.

DNS

Traces Domain Name System operations.

Use with care, stack overflows can be frequent.

Messages

Traces async socket messages.

Comms

Traces serial port communications.

PPP

Once connected to the PPP server, the Winsock establishes a PPP link by working through a number of phases. The first, LCP, opens the connection and tests the link. The either PAP or CHAP will follow if they have been enabled in File/PPP Options (PAP and CHAP are optional). Finally, IPCP is responsible for configuring the IP addresses and options such as header compression.

C021 Link Control Protocol (LCP).



C023 Password Authentication Protocol (PAP).

C223 Challenge Handshake Authentication Protocol (CHAP).

8021 Internet Protocol Control Protocol (IPCP).

Dialler

Contains functions relevant to dial up SLIP/PPP connections.

-  Login
-  Bye
-  Manual login
-  Other
-  Settings
-  Profile
-  Edit scripts
-  Other scripts

Login

Invokes the login.cmd dialler script. The first time it is launched you will be asked to enter phone number, username and password. You can change these later by selecting Profile from the Dialler menu.

Bye

Invokes the bye.cmd dialler script. To adjust the modem hangup string run the Modem profile (modem.cmd).

Manual login

Invokes the dialler manually. Use <Esc> to exit from the manual dialler.

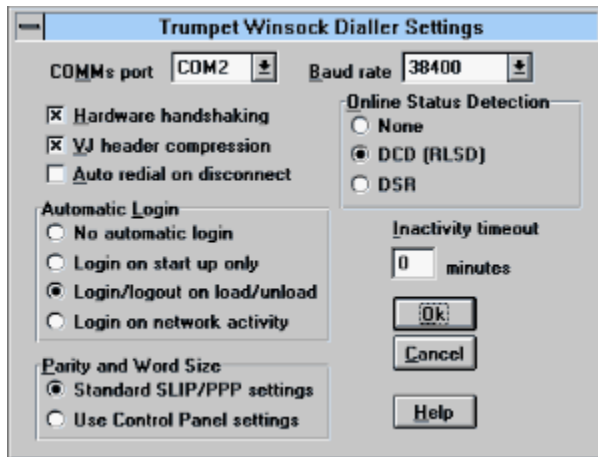
See Logging in manually.

Other










Invokes any script. A file selection dialog of *.cmd will be displayed.

Settings

The Dialler Settings dialog box allows you to adjust settings that are relevant to dial-up connections. Select a parameter to see its explanation.



Please see [Using the Trumpet Winsock over SLIP/PPP](#) for more information.

-  [COMM port](#)
-  [Baud rate](#)
-  [Hardware handshaking](#)
-  [VJ header compression](#)
-  [Automatic redial on disconnect](#)
-  [Online status detection](#)
-  [Automatic login](#)
-  [Inactivity timeout](#)
-  [Parity and word size](#)

COMM port

A communications port (or comm port) is a *serial port* that is used for connecting a device such as a modem, mouse or scanner to your computer.

The Winsock accesses the comm port by way of a comm driver (such as comm.driv).

If necessary, use the Ports screen in the Windows Control Panel to adjust the settings for your comm port. For example, changes to the IRQ. There is no setting for the IRQ in the Winsock since it uses whatever has been set in the Control Panel.

Baud rate

Determines the speed of the connection between the Winsock and your modem.

Not all baud rates listed will be valid for your modem, so please check your modem's manual for its highest baud rate.

Hardware handshaking

Enables RTS/CTS handshaking.

Hardware handshaking is recommended if your modem and modem cable support it and **&k3** is in the Init string of your Modem profile.

VJ header compression

Van Jacobsen (VJ) Header compression was developed to improve the efficiency of SLIP connections. Each SLIP packet contains header information and some actual data. Since the header information in adjacent SLIP packets is usually fairly similar, a lot less header information can be sent if the differences are sent.

SLIP with VJ Header Compression enabled is called CSLIP.

PPP supports header compression as an option.

Automatic redial on disconnect

If this option is checked, the Winsock will redial once per minute until reconnected. Otherwise, the Winsock will indicate that the connection has been lost and ask whether redial is required.

Note that this requires that Online status detection is enabled and functioning correctly.

Online status detection

Online status detection is necessary for the Automatic login and Automatic redial on disconnect features of the Trumpet Winsock to work correctly.

The Winsock can use either the Data Carrier Detect (DCD) or Data Set Ready (DSR) signal to track the online status of the modem, however DCD is more reliable.

For online status detection using DCD to work properly, **&c1** will need to be set in the modem's default profile. The modem's default profile is the group of settings that is loaded from memory when the modem is switched on.

To force the **&c1** command to be stored in the modem's memory, try adding **&w** to the Init string in Modem profile (modem.cmd), for example: **&k3&c1&w**

Similarly, online status detection using DSR requires **&s1** to be set in the modem's default profile.

If **None** is selected, then the Automatic redial on disconnect feature will be not be available, but the login script will be executed by an Automatic login regardless of the online status of the modem.

Automatic login

A login that uses a script (such as login.cmd) is called an automatic login. The following options determine when the login and bye scripts will be initiated.

No automatic login	Disables automatic login.
Login on startup only	Login.cmd is executed when the Winsock starts.
Login/logout on load/unload	Login.cmd is executed when the Winsock starts and bye.cmd is executed before the Winsock exits.
Login on network activity	Login.cmd is executed whenever a Winsock application tries to access a remote host. It may be useful to complement this with an <u>Inactivity timeout</u> to ensure that the bye script is run. In effect, this feature allows the Winsock to be used offline since the Winsock will not dial unless it really needs to.

Note: Under an Automatic login, if the Winsock is set to use online status detection (either DCD or DSR), then the Winsock will only execute login.cmd if the modem seems to be offline (and "Automatic login" in Dialler/Settings has been selected).

See also [How do I get demand loading to work?](#)

Inactivity timeout

This is the number of minutes that the Winsock keeps the line open if there is no network activity.

Under this situation the Winsock will remain loaded, so if you are going to need the Winsock to re-establish the connection, make sure you have selected Login on network activity from the list of Automatic login options.

A value of 0 disables the timeout, and therefore prevents the Winsock from disconnecting the line.

See Demand load timeout.

Parity and word size

The parity and word size is only relevant during the initial login phase of the connection (regardless of whether you are logging in manually or using a login script).

The Standard SLIP/PPP settings for parity and word size are 8 bits and no parity. If the server you are dialling into does not accept 8 bits and no parity, then you must set up the correct parity and word size under the Control Panel in Windows.

Profile

Allows you to enter the username and password of your SLIP/PPP account and your service provider's dial-up phone number.

Please note that this menu command launches a script called profile.cmd. This section refers to the standard version of this script as distributed by TSI. In some cases it may have modified by the distributor who supplied your copy of the Winsock.

See also [Configuring your profile](#).



[Modem profile](#)



[Server profile](#)

Modem profile

This dialog box allows you to adjust **modem** settings that are used by the login and bye scripts (login.cmd and bye.cmd respectively).

This screen can be reached directly by selecting modem.cmd from the Dialler menu.

- Use reset string
- Init string
- Dial string
- Hangup string

Use reset string

This your modem's reset command to be modified if necessary. Typically this will be: **z**

If these is a problem with your modem responding to the reset command as the first command it receives, then you may need to disable it completely.

Init string

The suggested modem string is **&f&c1&k3**.

Most modems are shipped with fairly standard default settings, so you may even be able to leave this string blank.

Note that &k3 is only relevant if your modem supports hardware handshaking. This command may need to be removed for modems slower than 14.4.

Dial string

Either **dt** for tone dialling or **dp** for pulse dialling.

Hangup string

This sets the init string that the bye script (bye.cmd) will use to hangup the modem. Normally this will be: **h**

Server profile

This dialog box allows you to adjust **server** settings that are used by the login script (login.cmd).

These prompts will need to be altered according to the messages your provider's machine gives when you log in and the replies you need to give. See Configuring your profile for more details and example configurations.

The difference between prompts, commands, and responses is that: *prompts* are strings that are output from the server when the server expects a reply. Once a prompt is received, the corresponding *command* will be sent by the script. Before completing the login, a script can wait for a *responses* from the server.

This screen can be reached directly by selecting server.cmd from the Dialler menu.

- Username prompt
- Password prompt
- Command prompt
- SLIP command
- SLIP response
- PPP command
- PPP response
- Scan for IP address

Username prompt

This is the server's prompt for the username. If the server does not prompt for a username, leave this field blank.

The default setting is "sername:" so that it will match with either "Username:" or "username:".

Password prompt

This is the server's prompt for the password. If the server does not prompt for a password, leave this field blank.

Command prompt

This is the server's command line prompt.

For example, `>`, `?`, or **command:**.

SLIP command

The command that is output to the server if SLIP has been selected in the File/Setup screen.

For example, this may be: **slip**, **cslip**, or **slip default**.

SLIP response

If this setting is left blank then the script will switch out of terminal mode and enable SLIP (so that it can start processing SLIP packets) as soon as the SLIP command has been output.

If the Winsock is required to wait for a certain text string before enabling SLIP, then enter it in this field.

PPP command

The command that is output to the server if PPP has been selected in the File/Setup screen.

For example, this may be: **ppp** or **ppp default**.

PPP response

If this setting is left blank then the script will switch out of terminal mode and enable PPP (so that it can start processing PPP packets) as soon as the PPP command has been output.

If the Winsock is required to wait for a certain text string before enabling PPP, then enter it in this field.

Scan for IP address

This setting should be enabled if SLIP is being used and the IP address is allocated dynamically.

If enabled, the login script will scan for an IP address. This IP address will then be used for the Winsock's IP address. If the text field is left blank, then the Winsock will use the first IP address it finds, otherwise it will wait for the first IP address *after* the text string is found.

Note: This option should be disabled under a PPP login, since PPP includes IP address negotiation.

Edit scripts

Invokes Notepad, allowing you to edit a script.

See [Scripting Language](#) and [Configuring your login script](#).


Other scripts

At the bottom of the Dialler menu is a list of available scripts other than Login, Bye, and Profile. This may include modem.cmd, server.cmd and any scripts that are specific to an Internet provider that have also been installed.

Note that setup.cmd was used with earlier version of the Winsock, but has now been replaced by the Login profile (profile.cmd). This can be accessed by selecting Profile from the Dialler menu.

Help

Opens the relevant section of the online [help](#) file.

 [Contents](#)

 [Commands](#)

 [Search](#)

 [Distribution](#)

 [Using help](#)

 [About](#)

Contents

The Contents page of the online help is the top level entry point into the documentation.

Commands

Provides a listing of the menu commands.

Search

The Search index of the Trumpet Winsock Help is useful for finding references to specific topics in the documentation.

Distribution

Calls up the section of the help file dealing with distribution.

Using help

Opens the help file for Windows Help.

About

Displays the version number and copyright of the Trumpet Winsock.

Scripting Language

This section describes the Trumpet Winsock scripting language as well as providing examples of its usage.

Have a look at the relevant sections of the [Trumpet Winsock FAQ](#), or [Trumpet Winsock Error Messages](#) if you're having difficulties with a script.

- [!\[\]\(756219e9389f679d57027482aa5cf5fc_img.jpg\) Basic commands](#)
- [!\[\]\(fcb77b2d9531d23794a07d244b7a89bc_img.jpg\) String formats](#)
- [!\[\]\(8175e06aff05874f50e11ffc448e6860_img.jpg\) Structured statements](#)
- [!\[\]\(d7fb7ebced2c712ed3052caf75d30501_img.jpg\) Assignments and expressions](#)
- [!\[\]\(d0fe824e371dd316453cda01cf34ca98_img.jpg\) Predefined variables](#)
- [!\[\]\(d1a2681fe7bfd4d9525bb4cc91a3e465_img.jpg\) String functions](#)
- [!\[\]\(8b973f5add5af5083ee0afc186cd5414_img.jpg\) Commands as operands](#)
- [!\[\]\(a60687ab372714f0e840dc74a221d5b2_img.jpg\) Example script fragments](#)
- [!\[\]\(d5f663f113d4d7bec37dcda4fee36bc6_img.jpg\) Expression syntax](#)

Basic commands

Each of the following commands is a scripting primitive and will return success or failure depending on whether the command succeeded or not. Commands always return success unless otherwise specified.

Parameters enclosed in < > characters need to be replaced by a value.

For example, the parameter <timeout> corresponds to a length of time in seconds and should be replaced an integer value or integer variable. Commands with a <timeout> parameter will return failure if they have been unable to succeed within <timeout> seconds, unless otherwise specified.

Parameters enclosed in () characters are compulsory and those enclosed within [] are optional. All parameters are compulsory unless otherwise stated. The | character represents logical OR.

- [abort](#)
- [address <timeout>](#)
- [bootp](#)
- [break](#)
- [display <string>](#)
- [echo \[on | off\]](#)
- [exec <program command>](#)
- [exit](#)
- [expect <timeout> <target string>](#)
- [input <timeout> <target string>](#)
- [getprofile <variable> <str_expr> <str_expr> <str_expr>](#)
- [putprofile <variable> <str_expr> <str_expr> <str_expr>](#)
- [load <variable> \[<string variable> \[<ini file>\]\]](#)
- [message <string>](#)
- [online](#)
- [output <string>](#)
- [outpotecho \[timeout \] <string>](#)
- [password \[<prompt string> \]](#)
- [query <string variable> \[<prompt string> \]](#)
- [quit](#)
- [read <timeout> <string variable>](#)
- [readx <timeout> <string variable>](#)
- [save <variable> \[<string variable> \[<ini file>\]\]](#)
- [set \(dtr | rts \) \[on | off \]](#)
- [sleep <timeout>](#)
- [status <string>](#)
- [trace \[on | off \]](#)
- [username \[<prompt string> \]](#)
- [wait <timeout> \[dsr | cts | dcd | rlsd \]](#)

abort

Aborts the script. This command always returns failure.

See [exit](#).

address <timeout>

Parses the current stream for an IP address, and copies it into the Winsock's primary IP address. The address change will not be effective until the script finishes.

Note that this will find the first IP address in the stream. If the current stream contains alternative IP addresses (annex IP address, etc) you will first need to check the input stream with the input command.

bootp

Attempts a BOOTP when the (login) script has finished executing. This will modify any TCP/IP setup parameters that are specified in the returned BOOTP response.

break

Sends a break.

display <string>

Displays <string> in the Trumpet Winsock window.

echo [on | off]

Displays any characters received while dialling in the window. The default is on.

exec <program command>

Executes the Windows command specified. This is useful for starting another program from the script.

It is important to quote literal strings by quotes if they contain the “\”(reverse slash) character. The reverse slash has a special meaning.

exit

Stops the currently executing script but still returns success.

See [abort](#).

expect <timeout> <target string>

Checks that the current input stream contains the target string. Unlike the input command, it doesn't search. If the command timed out or the string received does not match, the command returns failure.

The "expect" scripting command differs from the input command in that it doesn't allow for spare characters before the target string.

input <timeout> <target string>

Scans the current input stream for <target string>.

getprofile <variable> <str_expr> <str_expr> <str_expr>

New in 3.0!!

putprofile <variable> <str_expr> <str_expr> <str_expr>
New in 3.0!!

load <variable> [<string variable> [<ini file>]]

Returns <variable> which may be integer or string from <ini file>.

Where <string variable> is the section name within the ini file and defaults to “default vars”, and <ini file> is the name of the Winsock ini file and defaults to “trumpwsk.ini”.

Note that the file extension must be specified with the file name.

See [save](#).

message <string>

Presents a dialog box containing <string>, and pauses until OK is pressed.

online

Instructs the Winsock to go online immediately. Starts processing SLIP packets, and adjusts the state of the comms port to normal SLIP conditions.

output <string>

Sends <string> to the output stream on the comms port. Any characters received during this process will be echoed to the Trumpet Winsock window if the echo flag is on.

See outputecho, echo.

outpotecho [timeout] <string>

Sends <string> to the output stream on the comms port. As each character is output, the script processor waits for the corresponding character to be echoed. If the received character does not match, the command will abort and a failure is returned.

If a timeout is not provided, a default of 10 seconds is used. The timeout applies to the whole command.

In addition, the carriage-return is handled specially by waiting for a linefeed to be read after the carriage return is read. Both the sequences <CR><LF> and <CR><CR><LF> are acceptable.

See [output](#).

password [<prompt string>]

Brings up a message box that allows \$password (one of the predefined variables) to be entered or modified. If the user cancels the message box, failure is returned.

Any characters typed will be echoed with a "*" character, and encrypted if saved to the .ini file.

See query, username.

query <string variable> [<prompt string>]

Brings up a message box for the <string variable> to be entered or modified. An optional prompt may be provided. If the user cancels the message box, return failure.

See password, username.

quit

Posts a quit message to the Winsock tcpman. This will start a normal quit process.

read <timeout> <string variable>

Reads a whole line (terminated by a line feed character) from the comms input stream into <string variable>. Any carriage returns are stripped.

See [readx](#).

readx <timeout> <string variable>

Reads a whole line (terminated by a carriage return) from the comms input stream into <string variable>.

See read.

save <variable> [<string variable> [<ini file>]]

Saves <variable> which may be (integer or real) to <ini file>.

Where <string variable> is the section name within the ini file and defaults to “default vars”, and <ini file> is the name of the Winsock ini file and defaults to “trumpwsk.ini”.

Note that the file extension must be specified with the file name.

See [load](#).

set (dtr | rts) [on | off]

Toggles the DTR or RTS modem control.

sleep <timeout>

Pauses for <timeout> seconds.

Always returns success.

status <string>

Puts <string> into the Dialler Status dialog box, replacing any previously entered text.

The Dialler Status box, which is initially left blank, is automatically displayed when any script is executed. The script aborts if Cancel is pressed.

trace [on | off]

Toggles the tracing of scripting commands. This provides a description of each line as it executes.

username [<prompt string>]

Brings up a message box that allows \$username (one of the predefined variables) to be entered or modified. If the user cancels the message box, failure is returned.

See query, password.

wait <timeout> [dsr | cts | dcd | rlsd]






Pauses until the given modem signal changes state or timeout occurs.

See [sleep](#).

String formats

When a string is required for a parameter, all characters up until the end of the command line are processed as the string. Strings conform to the following format.

If you wish to use the characters \, #, \$ or % inside a string, they must be quoted with “ or have a \ character in front of them. The “ (quote) character may only be formed by preceding it with a \ character.

-  “ (quote)
-  # (hash)
-  \$ (dollar)
-  % (percent)
-  \ (reverse slash)

“ (quote)

A quote character (“) means that all characters are to be taken as is without any special meaning until a corresponding closing quote (”) has been found. The string is not permitted to extend over more than one line.

(hash)

If a “#” symbol is found that is not inside quotes, then it means that the rest of the logical line is to be ignored. This can be used to annotate the script.

\$ (dollar)

If a “\$” symbol is found, it denotes a string variable. If the variable has been assigned a value when the command is executed, it’s value will be placed in the string. If not, an error message is displayed, and the script continues.

% (percent)

If a “%” symbol is found, it denotes an integer variable. If the variable has been assigned a value when the command is executed, its value will be placed in the string. If not, an error message is displayed, and the script continues. Integers are 32 bit signed integers in the range -2147483648 to 2147483647.

\ (reverse slash)






If a reverse slash “\” is found, it denotes that the character following either has a special meaning which will be placed in the string, or is to be taken literally.

Here is a list of special “\” character meanings.

<code>\b</code>	Back space (ASCII 8).
<code>\c</code>	The comm port id as a decimal number (%slippport).
<code>\e</code>	Escape (ASCII 27).
<code>\f</code>	Form feed character (ASCII 12).
<code>\i</code>	The current IP address (\$ip).
<code>\l</code>	Line feed character (ASCII 10).
<code>\n</code>	An end of line sequence (ASCII 13, then ASCII 10).
<code>\p</code>	The current password (\$password).
<code>\r</code>	Carriage return - Enter key (ASCII 13).
<code>\t</code>	Tab (ASCII 9).
<code>\u</code>	The current username (\$username).
<code>\</code>	ASCII <number> (eg. <code>\0</code> <code>\27</code> <code>\255</code>).
<code><number></code>	

Structured statements

In addition to the basic command primitives, there are the following special control statements.

-  If Statement
-  While Loop
-  Repeat .. Until Loop
-  Include
-  Dialog

If Statement

```
if <condition>  
    <statement list>  
[ else  
    <statement list> ]  
end
```

If the condition evaluates as true, the first statement list is executed. If the condition evaluates as false, the first statement list is ignored, and if an else clause is present, it is executed instead.

While Loop

```
while <condition>  
    <statement list>  
end
```

While the condition evaluates as true, the statement list is executed.

Repeat .. Until Loop

```
repeat  
    <statement list>  
until <condition>
```

The statement list is repeatedly executed until the condition evaluates as true.

Include

include <expression> ["\", " <params>]

Allows another script to be executed. For example:

```
include foobah.cmd
```

Dialog

dialog

```
{  
  <dialog item>  
}
```

end

```
<dialog item> ::= edit <stringvar> "," <id>  
| combo <stringvar> "," <id> "," <options_string>  
| select <intvar> "," <id> "," <opt-1> { "," <opt-n> }  
| toggle <intvar> "," <id>  
| button <include_file> "," <id>
```

Assignments and expressions

Assignments are used to store new values into variables. They take the following forms:

`<integer variable> = <integer expression>`

Eg. `%attempts = %attempts + 1`

`<string variable> = <string expression>`

Eg. `$name = $first + $last`

The variable part may be a string variable (either unspecified or starting with a \$), or an integer variable (starting with a %). Variables aren't case sensitive.

String expressions may be composed of the following operands:

quoted strings eg. "Fred", "OK"

string variables eg. \$name, \$response

String expressions may use the following operators: () + concatenation

Integer expressions may be composed of the following operands:

numbers eg. 1, 43, 7373

integer variables eg. %I, %count

Integer expressions use the following operators in order of priority:

() Brackets.

* / % Multiplication, division and modulo division.

%

+, - Addition and subtraction.

Also, integer operands may be formed by the comparison of strings.

eg. `$A = "FRED"` will evaluate to an integer operand of 0 or 1 depending on the value of \$A.

Conditional expressions may also be formed using conditions and boolean operators. They return 0 or 1 depending on the result of the condition.

These operators have the following priorities.

`= <> <> <= >=` Equal, not equal, less than, greater than,
less than or equal, greater than or equal.

! Boolean not operator.

& Boolean and operator.

| Boolean or operator.

Predefined variables

The names of some variables have predefined meanings.

These variables provide access to information contained in the File/Setup and Dialler/Settings screens:

Variable	Corresponding setting
%slipport	<u>Comm port</u> .
%ppp	0 = <u>SLIP</u> , 1 = <u>PPP</u> .
\$ip	<u>IP address</u> .
\$dns	<u>DNS server</u> .
\$domain	<u>Domain suffix</u> .

These variables are used by scripting commands for specific purposes:

Variable	Explanation
\$inipath	Determines the ini file in that variables will be saved to and loaded from.
\$section	Determines the section of the ini file that variables will be saved to and loaded from.
\$username	Sets a username. Used by the <u>username</u> command.
\$password	Sets a password. Used by the <u>password</u> command. Any variable with “password” in its name will be saved in an encrypted form.

String functions

The following string functions may be used as operands.

copy (\$s, %p, %l)

len (\$s)

lower (\$s)

pos (\$s1, \$s2)

upper (\$s)

copy (\$s, %p, %l)

Copy characters from string \$s starting at character %p for %l characters and return them as an expression. Note that string character counting starts at 1 - not zero.

len (\$s)

Return the length of string \$s as an integer number.

lower (\$s)

Convert string \$s to all lower case.

pos (\$s1, \$s2)

Return an integer number corresponding to the position of the first character of string \$s1 as found in string \$s2. A zero value is returned if \$s1 is not found within \$s2.

upper (\$s)

Convert string \$s to all upper case.

Commands as operands


A scripting command can be used as part of an expression by surrounding the command with [] symbols. This is called a conditional statement.


Eg. `[input 10 OK\n]`

This operand would have the value 1 if the command succeeded or 0 if it failed.

Example script fragments

 Repeated dial

 Busy detect

 Multiple number

Repeated dial

The following segment of script could be used to attempt a repeated dial of a given number. Note the use of the `outputecho` rather than `output` so that any characters echoed from the command will be consumed.

```
%attempts = 0
repeat
    %attempts = %attempts + 1
    outputecho 60 atdt345772371\r
until [input 30 CONNECT\n] | %attempts = 10
```

This section of script is fine, but would take 30 seconds for the input function to timeout if a response other than `CONNECT` were returned. It could be refined further into the following lines

```
%attempts = 0
repeat
    %attempts = %attempts + 1
    outputecho 60 atdt345772371\r
    read 30 $result
until $result = "CONNECT" | %attempts = 10
```

This piece of script would be fine unless the modem failed to respond, in which case the script would abort. A further refinement would be the following.

```
%attempts = 0
repeat
    %attempts = %attempts + 1
    outputecho 60 atdt345772371\r
    %timeout = [read 30 $result]
    if %timeout = 0
        display "Dial up timed out."\n
    end
until $result = "CONNECT" | %attempts = 10 | %timeout
```

These portions of script are only examples to demonstrate the use of scripting. In practice, most modems do not generate simple messages after the `atdt` command. You will have to skip extra lines etc., to get a working script.

Busy detect

If you are having trouble connecting to a busy site, it would be good to be able to have Trumpet Winsock redial more quickly. Your script will need to detect a BUSY signal and redial if necessary.

In the following script segment, if a BUSY signal is detected within 5 seconds then it will redial. This time may need to be increased if it takes longer for a BUSY response. If it is not busy, then it waits for 30 seconds for a CONNECT. If it gets one, it exits the loop and continues on with the script, otherwise it will redial after 30 seconds.

```
%ok = 0
%busy = 0
repeat
  output "atdt"$number\n
  %busy = [INPUT 5 BUSY]
  if !%busy
    %ok = [INPUT 30 CONNECT]
  end
until %ok
```

Multiple number

Some service providers have more than one number to dial when they are busy. In this example, four numbers are pre-defined in the script. What it does is try each number in turn until one responds with a CONNECT. If it has already tried all four, then it goes back to the first one. In this case it would repeat indefinitely, so you may want to put a limit on the number of attempts. This can also be increased or decreased to the number of dial in numbers your service provider has.







```
$number1 = "232224"  
$number2 = "232225"  
$number3 = "232226"  
$number4 = "232227"  
$number = ""  
  
%num = 0  
%ok = 0  
%busy = 0  
  
repeat  
    %num = %num + 1  
    if %num > 4  
        %num = 1  
    end  
    if %num = 1  
        $number = $number1  
    end  
    if %num = 2  
        $number = $number2  
    end  
    if %num = 3  
        $number = $number3  
    end  
    if %num = 4  
        $number = $number4  
    end  
    output "atdt"$number\13  
    %busy = [input 5 BUSY]  
    if !%busy  
        %ok = [input 30 CONNECT]  
    end  
until %ok
```

Expression syntax

```
<expression> ::= <condterm> { "|" <condterm> }
<condterm> ::= <condfactor> { "&" <condfactor> }
<condfactor> ::= [ "!" ] <conditional>
<conditional> ::= <expr> [ ( "=" | "<" | ">" | "<" | ">=" | "<=" )
    <expr> ]
<expr> ::= [ "+" | "-" ] <term> { ( "+" | "-" ) <term> }
<term> ::= <factor> { ( "*" | "/" | "%" ) <factor> }
<factor> ::= <operand> | "(" <expression> ")"
<operand> ::= <number> | <variable> | <string> | <conditional
    statement> | <special function>
<number> ::= "0".."9" { "0".."9" } | "0x" { "0".."9","A".."F" }
<variable> ::= "%" { "A".."Z","0".."9" } | "$" { "A".."Z","0".."9" }
<string> ::= " { any character except " } "
<conditional statement> ::= "[" <command> "]"
<special function> ::= COPY "(" <string> "," <index> "," <count> ")"
    | POS "(" <string1> "," <string2> ")"
    | LEN "(" <string> ")"
    | UPPER "(" <string> ")"
    | LOWER "(" <string> ")"
    | MATCHINPUT "(" <timeout> "," <string> { "," <string> } ")"
    | PARAM "(" <index> ")"
```

Trumpet Winsock Error Messages

This section describes the causes and/or solutions for some common error messages. Have a look at our [Web site](#) for a more complete list.

-  [Windows and Misc error messages](#)
-  [Winsock API errors](#)
-  [TCP errors](#)
-  [Dialler errors](#)
-  [Comm driver errors](#)
-  [Packet driver error messages](#)

Windows and Misc error messages

- Call to undefined dynalink
- DB overflow
- Exec error: 2
- get_recv_mode failed = <number>
- Modem is not responding
- Password does not match
- Registration name not found
- Runtime error 202
- Script aborted
- TCPMAN.EXE not found
- Unable to initialise network layer
- Unable to perform bootp
- Unable to load TCP
- Unable to locate registration host
- Versions do not match

Call to undefined dynalink

This error indicates that a function call has been made to a nonexistent routine in a Windows library.

In the case of the Trumpet Winsock, this is usually caused by having multiple winsock.dll or wsock32.dll files on a machine. To overcome this, try doing a search for winsock.dll and wsock32.dll on the machine, and delete or rename any extras.

See also [TCPMAN.EXE not found](#).

DB overflow

DB overflow relates to getXbyY queries.. especially async forms.

The error message suggests one of two possibilities..

1. That the application is not providing a large enough buffer for the reply.
2. That a DNS reply is too large (perhaps due a damaged UDP packet).

A DNS trace may give some indication of the problem. It may only be specific to certain sites.

Sometimes the application may be able to resolve the problem by restarting the call with a larger buffer. In this case, one can simply ignore the message.

Exec error: 2

This error indicates “file not found”. The exec command in a script will generate this command if the file referred to either doesn’t exist or has been mistyped.

Note that the backslash “\” symbol is interpreted as a special characters, so be sure to put the path name in quotes (“”). For example:

```
exec "C:\dir\program.exe"
```

get_rcv_mode failed = <number>

This is generated by the packet driver interface, and indicates that the packet driver did not support the get receive mode call. It should not affect normal operation.

A possible cause is that multicast is being used with a packet driver that doesn't like the way the Winsock tries to use it.

Modem is not responding

Your modem should be configured so that commands can be typed in, executed and have the appropriate result code displayed on the screen. For most commands, the result should be "OK".

If this is not the case, then the default login script will give a "Modem is not responding" error message.

The most common reasons why a modem does not appear to respond (aside from the modem being turned off or not plugged in) are...

1. The baud rate set in the Dialler/Settings screen is not supported by your modem. Please consult your modem's manual to discover which baud rates are valid. The baud rate should usually be one of 9600, 19200, 38400, 57600, or 115200.
2. The modem is set to not echo characters as they are entered. To enable echoing, put the command **e1** into the modem's Init string in the Dialler/Settings screen and disable Use reset string.
3. The modem is returning numeric result codes rather than verbal codes (ie 0 instead OK). To enable verbal results, put the command **v1** into the modem's Init string in the Dialler/Settings screen and disable Use reset string.
4. The modem is not sending any result codes at all. To enable result codes, put the command **q0** into the modem's Init string in the Dialler/Settings screen and disable Use reset string.
5. The script isn't waiting long enough for the modem to return "OK". To extend the timeout of an input command, increase the first parameter. To edit your login.cmd, select Edit scripts from the Dialler menu. For example, this command waits 10 seconds for an "OK": `input 10 OK\n`

To make sure that the modem is going to echo characters and enable verbal result codes, add **e1v1q0** your modem's Init string in the Dialler/Settings screen and disable Use reset string.

Password does not match

You are given a password when you register the software. Since your password is generated as a function of your registration name, you don't get to choose it. See [How to register](#) from [Registration Details](#) for details.

If you are getting this error once you have registered the software and received your password, make sure that you are using the right password for the right version of the software and that the details are entered correctly.

If your password is for an earlier version of the Winsock, you may need to update your password. In particular, your 2.1 password will not work with 3.0.

Make sure that you enter the registration name and password exactly as provided, without the quotes (“”). When entering your password, delete the contents of the field before entering anything, and please be careful not to confuse the numbers 0, 1, and 8 with the letters O, I, and B.

For more recent information on the registration process, see:

<http://www.trumpet.com.au/wsk/faq/register.htm>

Registration name not found

This message is displayed if your registration name could not be found on the registration host.

If you have registered an earlier version of the Winsock and are having any difficulty updating your registration password, then please contact our [Registrations Department](#) with details of how and when you originally registered.

Runtime error 202

Runtime error 202 is a stack overflow, which is often caused either an application error, or by one of the traces in the Winsock being turned on (in particular DNS).

Script aborted

A “Script aborted” message will be given if the script aborts for any reason before completion.

The most common cause of an aborted script seems to be due to an input command timing out. This is usually the result of a misconfigured script (\$userprompt, \$passprompt, etc..), but may indicate a problem with the SLIP/PPP server.

The Winsock assumes that your script is configured correctly, and therefore that the SLIP/PPP connection hasn't been established correctly. An error message is passed back to any applications that try to access the Winsock, and usually relayed by that application to the screen. For example, Netscape displays, “Unable to initialise network layer”.

With versions of the Winsock earlier than 2.1, no error messages were returned to the Winsock applications if a login was unsuccessful, even though “Script aborted” was displayed. In turn, applications could freeze for several minutes before returning an error message. In fact, Netscape took ten minutes to time out before it even appeared on the task list.

This behaviour has been rectified, but not without a trade-off.

It seems that a number of 2.0b users have login scripts that return “Script aborted” despite logging in successfully. This is often the result of an incompatibility between the default login script and some SLIP/PPP servers. The problem goes like this... Once the username and password have been entered, the default login.cmd expects to be prompted for a service. However, some servers have no such prompt, they just start sending out SLIP or PPP regardless. After the script aborts (30 seconds), the Winsock would enable SLIP/PPP, and the applications would continue without a hitch.

From version 2.1 onwards, Winsock applications are not be able to connect to the Internet if “Script aborted” is returned.

Ensure that your script is suitably debugged and any error conditions have been accounted for within the script.

See [Logging in for the first time](#) for more information.

TCPMAN.EXE not found

The Trumpet Winsock consists of two main components, winsock.dll and tcpman.exe. Winsock.dll is the core of the Winsock, as such it responds to function calls from Winsock applications (such as Netscape, Ping, etc..) and maintains the TCP/IP layer. On the other hand, tcpman.exe provides an interface to the Winsock and manages the dialler. Neither of these files can operate without the other.

When the Trumpet Winsock is demand loaded (that is, when another application launches it), winsock.dll is loaded first. The first thing the winsock.dll does is to try and find tcpman.exe. If it can't, then the "TCPMAN.EXE not found" message is displayed.

To rectify this problem, make sure that:

1. You only have one copy of the Trumpet Winsock files (including winsock.dll, tcpman.exe, hosts, protocol, and services) on your machine. Rename your backup files if necessary.
2. They are all in the same directory.
3. The files tcpman.exe and winsock.dll are from the same release.

Unable to initialise network layer

Netscape will return this error message in response to an aborted login script. See [Script aborted](#).

Unable to perform bootp

No reply received to sent BOOTP requests. Either no BOOTP server available or connection not properly established. If you've got BOOTP set unnecessarily, replace your IP address with 0.0.0.0 before dialling again.

Unable to load TCP

This message will be displayed whenever the TCP/IP stack fails to load. The specific cause of the error will be displayed after this message.

If you are using a packet driver or packet driver shim, check that the packet driver loaded properly. You can do this by observing the screen as the .bat file loading the packet driver executes. This will also tell you if winpkt managed to find it.

Unable to locate registration host

If you receive this error while you are trying to register the Trumpet Winsock even though you are connected to the Internet, please contact our [Registrations Department](#) or try another method of registering. See [How to register](#) from [Registration Details](#) for the alternatives.

Versions do not match

This message will be displayed when trying to load the Winsock if the tcpman.exe and winsock.dll are from different releases. To rectify this problem, make sure that:

1. You only have one copy of the Trumpet Winsock files (including winsock.dll, tcpman.exe, hosts, protocol, and services) on your machine. Rename your backup files if necessary.
2. They are all in the same directory.
3. The files tcpman.exe and winsock.dll are from the same release.

See [Tcpman.exe not found](#) for more information.

Winsock API errors

- WSAEINPROGRESS (<number>) by <program name>
- WSAENOTINITIALIZED (<number>) by <program name>
- WSAENOTSOCK (<number>) by <program name>
- Task <program name> did not call WSACleanup

WSAEINPROGRESS (<number>) by <program name>

The message WSAEINPROGRESS is a diagnostic message generated by the Winsock. When a socket call is executed, the Winsock can possibly “block”. This means that the current application will sit in a loop examining the socket to determine whether any more data has arrived or whatever, and this is the normal operation of the Winsock. When Windows is running however, windows messages can be generated outside of the program’s normal operation, for example if the user clicks on a button or selects a menu option. If one of these operations then calls another Winsock function, the Winsock will be re-entered possibly causing problems.

While Trumpet Winsock has been designed with full reentrancy in mind, the Winsock specification insists that should this eventuality arise, that the socket call should always return an error message in this circumstance. This is because many stacks are unable to cope and can crash.

Since the usual reason for this reentrancy occurring is a programming error of some sort, Trumpet Winsock will print out this diagnostic error message when the condition arises so that the programmer can be made aware of it.

WSAENOTINITIALIZED (<number>) by <program name>

Before any socket call can be made, the calling program must call WSAStartup.

If they don't, this error is returned.

WSAENOTSOCK (<number>) by <program name>

There are two possible causes of that error.

One possibility is that a socket number has been used which is outside of the range of legal sockets numbers. Either an `INVALID_SOCKET (-1)` was passed, or the application has got a corrupt socket number.

Alternatively a legal socket number is being used, but the socket table is empty at that slot. Usually when a socket has been closed but is still trying to be used.

When the unregistered Trumpet Winsock expires, it only allows one socket to be opened. As a result, this error message will be displayed much more frequently.

Task <program name> did not call WSACleanup

This message is generated when an application exits without cleaning up after itself. WSACleanup is the last call an application should make to the Winsock. If the calling application exits abnormally (ie it crashes) it will not have a chance to make that call. In which case, the Winsock will display that error message.

It is a symptom of the calling application crashing, but not the cause of it.

TCP errors

accept returned nil

CRITICAL ERROR: TCP poll reentrancy detected - please report exact circumstance

Unacceptable segment : seg.seq = XXXXX rcv.nxt = YYYYY

accept returned nil

Accept returned nil is a diagnostic message which indicates that an incoming TCP connection was killed before a socket could be accepted.

CRITICAL ERROR: TCP poll reentrancy detected - please report exact circumstance

TCP Poll Reentrancy is an internal semaphore that the Winsock uses to detect any reentrancy to critical sections. It sometimes gets left open when a program crashes in the middle of a socket call.

This will typically be caused by a stack overflow (Runtime error 202), which may in fact be caused by one of the traces in the Winsock being turned on (in particular DNS).

Unacceptable segment : seg.seq = XXXXX rcv.nxt = YYYYY

This message is generated by the Trace/TCP option. It indicates that an unexpected TCP segment has been received. This may be caused by the TCP stack at the other end sending overlapped segments.

Ironically, this problem may be caused by the Winsock's timing being at fault as a result of the trace being enabled.

Since the segments will usually need to be resent, the transfer rate will become slower.

Dialler errors

- Comm errors [BREAK]
- Comm errors [FRAME]
- Comm errors [OVERRUN]
- Comm errors [RXOVER]
- Comm errors [RXPAR]
- Comm errors [TXFULL]
- enablecommnotification failed - Winsock will function more slowly
- getcommstate failed
- setcommstate failed
- PPP Frame check error
- Unable to open <commX>

Comm errors [BREAK]

A break signal has been received.

Comm errors [FRAME]

This error indicates that the start and stop bits of the data are in the wrong places. This is caused by a hardware failure of some description.

An occasional frame error may indicate nothing more than a slightly noisy line. This error can also be generated by turning the modem off and on.

Numerous frame errors could indicate an incorrect baud rate, or a faulty modem, modem cable, phone cable, or comm card.

Comm errors [OVERRUN]

Overruns are the result of the modem going too fast for the comm port and/or Windows. There are a number of possible causes, some of which can be overcome.

If you are getting lots of overrun errors, you may be able to rectify the problem with one of the following tips:

1. If you've got a high speed modem (14400 or higher), make sure that you're using a 16540 or 16550 UART, as opposed to the older 8250 UART. Use MSD to check what UART you have on your modem's comm port. To run MSD: exit Windows, then type **msd** at the command prompt and press <Enter>.
2. If you're using Windows 3.1, check the "comm.drv=" line in the [boot] section of your system.ini file. If it reads "comm.drv=comm.drv", you are probably using the original comm driver which was designed for the 8250 UART. You might like to try another, such as Cybercom or Winfax Pro.
3. Add the following statements to the [386Enh] section of your system.ini file:
 - comXfifo=1
 - comXbuffer=1024
 - comboosttime=8
 - Where X is the number of your Comm port.
4. Some video drivers built around the s3 video chip also hold CPU cycles and therefore cause overruns as well. s3 have apparently released driver updates that cure the problem. Several OEM vendors using the s3 based design have also followed suit.
5. Unfortunately, there is one cause of overruns that cannot be avoided..
 - When accessing a local disk, Windows halts any I/O to the comm ports for a short period of time. This in turn causes the overruns, when the Winsock is unable to access the comm port. This is unfortunately a glitch in the internals of Windows.

We are interested in finding out which modem cards people have had successes and failures with. If you have any information about this or anything else that should be added to this page, please write to suggest@trumpet.com.au.

Comm errors [RXOVER]

This error message indicates an overrun of the SLIP/PPP receive buffer.

There are two ways of overcoming this, and they both involve editing your trumpwsk.ini file.

The best way is to increase the size of the buffer, by increasing the `slip-rcvbuf`:

```
slip-rcvbuf=8192
```

Increasing this parameter means that the buffer can be raised to match the incoming data flow. Try doubling it to start with.

The second way is to limit the incoming data flow by setting:

```
slip-rtflow=1
```

Comm errors [RXPAR]

Comm error [RXPAR] means that the parity is wrong. Usually this indicates that the SLIP/PPP server is using a different parity to the default Winsock setting. In this case, contact your ISP to determine what the correct settings should be.

If they are anything other than 8 bits and no parity select “Use Control Panel settings for parity and word size in the Dialler/Settings screen. You can set up the correct parity and word size under the Control Panel in Windows.

Comm errors [TXFULL]

This error indicates that the transmit buffer is full. Normally this will not occur since the Winsock checks before sending data.

If this condition does arise, it suggests that a non-standard comm driver may be installed. For example, a comm driver to modem pool redirector.

enablecommnotification failed - Winsock will function more slowly

The “enablecommnotification” call is a function call implemented in new versions of Windows that older comm drivers (for example turbocomm) don't recognise.

The solution is to either use the standard comm driver, update of your old one, or obtain a new comm driver. There are apparently plenty available (Cybercom, Winfax Pro, etc.).

If you're running Win3.11, then the standard comm driver should work fine. You can find out the name of the comm driver that is installed on your machine by checking your system.ini file. In the [boot] section of your system.ini file, there is a line that reads:

```
comm.driv=?
```

Setting it to comm.driv=comm.driv will reset it to the original comm driver, as long as the original driver hasn't been replaced by another file of the same name.

getcommstate failed

The reason is displayed following this message, and can be found under Comm driver errors.

setcommstate failed

The reason is displayed following this message, and can be found under Comm driver errors.

PPP Frame check error

To check if there are any errors when a PPP packet is transmitted, a Frame Check Sequence (FCS) is performed. The sender calculates this by adding together the binary values of all the bytes in the packet and sends it as part of the packet. The receiver makes the same calculation, and if the two values aren't the same, a frame error occurs.

PPP frame check errors generally occur when transferring a large amount of data over your PPP connection, and are occasionally caused by Comm errors [OVERRUN] errors.

Unable to open <commX>

This message indicates that the requested Comm port couldn't be opened. The reason is displayed following this message, and can be found under Comm driver errors.

Comm driver errors

These errors appear following either getcommstate failed, setcommstate failed or Unable to open <commX>.

- Baud rate invalid
- Device is already open
- Device is not open
- Hardware is not available

Baud rate invalid

The baud rate in Dialler/Settings isn't supported by the current comm driver.

Device is already open

The comm port is being accessed by another piece of software. For example, if fax software is loaded it may be accessing the modem. In this case you will need to exit the fax software before starting the Winsock.

Alternatively, the Winsock may be trying to access the wrong comm port. In this case, try selecting another comm port from the Dialler/Settings screen.

Device is not open

The open call failed, so another error message should be displayed as well.

Hardware is not available

The Winsock has tried to open a comm port that is not available. Modems are usually installed on either comm port 1 or 2, although they may occasionally be found on 3 or 4. Try another comm port in the Dialler/Settings screen.

Packet driver error messages

- Packet driver must be class 1 (Blue book) or class 6 (SLIP)
- Too many packet buffers allocated
- Unable to allocate network buffers
- Unable to locate WINPKT or PKTDRV virtual packet driver
- Warning - running with less than recommended number of network buffers (X of Y)

Packet driver must be class 1 (Blue book) or class 6 (SLIP)

This indicates that either winpkt or ndis3pkt hasn't been loaded properly or the wrong kind of packet driver is being used.

For example, a token ring driver is being used with a packet driver shim that doesn't support it.

See also [Using the Trumpet Winsock over a packet driver.](#)

Too many packet buffers allocated

Since the winsock allocates network buffers from main memory so is not too much of a problem. The main problem is when the size of the packet buffer gets too large. in that case, the winsock will try to reduce the number to a lower value. There is an eventual limit beyond which the winsock can't function.

Unable to allocate network buffers

Try to free up some special driver memory by removing windows device drivers.

See Warning - running with less than recommended number of network buffers (X of Y).

Unable to locate WINPKT or PKTDRV virtual packet driver

Check that the packet vector setting under File/Setup is correct.

See Packet driver must be class 1 (Blue book) or class 6 (SLIP).

Warning - running with less than recommended number of network buffers (X of Y)

There are two types of network buffers. This error refers specifically to *packet buffers*, which the Winsock allocates from Global DOS memory. To allocate the full number of packet buffers, more conventional memory is required. Remove unnecessary TSRs and device drivers from your autoexec.bat, and load the remaining ones into high memory if possible.

Also, when you enter Windows, try loading the Winsock before any other applications. If the Winsock is being used frequently, you may even like to add the Winsock to the StartUp group in Program Manager.

If this doesn't work, it may be because Windows has attempted to load the entire Winsock into low memory. This is not a good thing, and will inevitably result in insufficient driver memory being available. The work-around for this is to try loading the Winsock at a different time.






Trumpet Winsock FAQ

Each of the following sections covers some of the most common questions (and their answers) regarding the Trumpet Winsock, problems with menu commands, and installation difficulties.

To find information on a topic, click on the **Search** button or browse through the topics listed below.

If you can't find the answers you're looking for in either this manual or our [Web site](#) then please [contact](#) us.

For descriptions of the most common error messages, please see [Trumpet Winsock Error Messages](#).

-  [General questions](#)
-  [Registration questions](#)
-  [Dialler questions](#)
-  [Programmer's notes](#)
-  [Windows 95 questions](#)

General questions

- What files should I backup?
- Why are some entries in the File/Setup screen greyed out?
- Are there docs for Trumpet's Winsock Applications?
- Shouldn't "dialler" be spelt "dialer"?
- Where can I send bug reports?

What files should I backup?

In addition to the Winsock package that was originally supplied to you (whether a zipped file or on a disk), you should probably backup your *.cmd scripts and trumpwsk.ini settings file.

It is also recommended that you write down or print out your registration name and password, and keep them in a safe place.

Why are some entries in the File/Setup screen greyed out?

Some entries in the File/Setup screen relate specifically to the packet driver or dial-up configurations. For that reason, they are greyed out when not applicable.

Are there docs for Trumpet's Winsock Applications?

Documentation is available for the Winsock Applications (Trmptel, Winchat, Winarch, Ftpw, View, and WinIRC) on the following page:

<http://www.trumpet.com.au/wsk/winapps.htm>

These applications are available from our [Web](#) site or on the Trumpet Winsock disk.

Shouldn't "dialler" be spelt "dialer"?

"Dialer" is the American spelling, while "dialler" is Australian. Since the Trumpet Winsock was developed by an Australian company, Australian spellings have been used for the interface.

Where can I send bug reports?

If you can promise that you have searched through this help file, and not found any mention of the bug you've encountered, please write to winsock-support@trumpet.com.au.

Registration questions

For more general information regarding prices, site licenses, and distribution agreements, please see [Registration Details](#). Here are some of the topics not covered in that section.

- [Can I register over the Internet?](#)
- [How long do registrations take?](#)
- [What do I get when I register?](#)
- [I lost my password, can you send it again?](#)

Can I register over the Internet?

You may register over the Internet by using the Internet registration feature of the Trumpet Winsock. This feature is for sending encrypted credit card details to Trumpet Software International via the Internet. This will only work if your Winsock is functioning and your network is visible to ours.

How long do registrations take?

We aim to have all registrations processed within five working days. If you asked for your password to be sent to you by postal mail, then please allow an extra ten working days for delivery.

If you haven't received your password within this time, then please contact our Registrations Department (they may be unable to contact you). Please include at least two forms of contact and details of how and when you registered.

What do I get when I register?

You get a password that permits you to keep using the Trumpet Winsock after 30 days. There's actually no difference between the unregistered and registered versions aside from this (We only mail out disks when the software is registered by a phone, fax, or postal order and the disk is requested).

As a registered user, you will receive support (within the existing Capabilities of the Trumpet Winsock) for up to 12 months from the date of purchase. Support will include any upgrades or bug fixes released within that 12 months.

I lost my password, can you send it again?

If you've lost your password, then please contact our Registrations Department with details of how and when you originally registered. Also, please supply the name you registered under if you can remember it.

Dialler questions

For an overview of how to login manually and automate the login process, please see [Using the Trumpet Winsock over SLIP/PPP](#). This section provides answers to some commonly asked that may not be covered in that section.

- [What's the difference between a manual and automatic login?](#)
- [How do I get demand loading to work?](#)
- [How do I edit my login script?](#)
- [How do I change my login phone number, etc?](#)
- [How do I set the Winsock to redial automatically?](#)
- [Why is the modem connecting at 2400?](#)
- [Why does my modem get disconnected?](#)
- [How do I optimise my MTU, RWIN, and MSS?](#)
- [How can I use the Winsock offline?](#)
- [Can I use another program as a dialler?](#)
- [How do I switch between tone and pulse dialling?](#)
- [How do I send a "break" with Trumpet Winsock?](#)
- [Can I setup the Winsock for multiple Internet Service Providers?](#)
- [How do I get bye.cmd to hangup my modem?](#)

What's the difference between a manual and automatic login?

In a manual login, the user enters the modem commands and server commands at the keyboard. See [Logging in manually](#) for more information.

Logging in manually is good for seeing what is actually happening, but can be very tedious after a while. For this reason, the Trumpet Winsock allows you to configure login scripts that automate the manual login process. The Winsock can also be configured to run these scripts on demand.

A login that uses a script is called an automatic login.

We say that the Winsock has been "demand loaded" when the winsock.dll has been loaded before the tcpman.exe. This is usually how other applications start the Winsock.

How do I get demand loading to work?

1. Put the directory containing the winsock.dll in the path statement of your autoexec.bat file.
 - If you want to be really sure things are going to work, make sure that you only have one copy of any winsock.dll, wsock32.dll or tcpman.exe files (rename any backup files if necessary), and that they are all in the same directory.
2. Have login.cmd and bye.cmd scripts that do what you want them to and return "script completed".
 - For help with this, see Automating the login process.
3. Enable one of the Automatic login options from Dialler/Settings.
4. Finally, the Online status detection from Dialler/Settings will need to be adjusted to suit.
 - The default is DCD, which is usually okay. If you're unsure, try DCD first, then try the others.

How do I edit my login script?

Scripts can be edited using any text editor as long as the files are saved as text, with filenames ending in .cmd (for example, myscript.cmd). To edit your script using Notepad, go to the Dialler menu and select Edit scripts, then double-click on the script you wish to edit.

For an overview of the scripting language, please see [Scripting Language](#).

How do I change my login phone number, etc?

As long as you are using a login script that is based on the example script distributed with the Trumpet Winsock, you can edit your login phone number, username and password using the following method:

1. Go to the Dialler menu, and select setup.cmd.
2. Enter or modify then entry then press OK after each one.

How do I set the Winsock to redial automatically?

To configure the Winsock to redial automatically:

1. Select a method of Online status detection that is supported by your modem (and enabled) in the Winsock's File/Setup screen.
 - The most reliable method is DCD (Data Carrier Detect), which requires the modem command **&c1** to be set. As an alternative, DSR (Data Set Ready) can also be used, and requires **&s1**. The settings **&c1** and **&s1** are the defaults for most modems.
2. Select Automatic redial when disconnected from the Dialler/Settings screen.

Why is the modem connecting at 2400?

The most common cause of this problem has been non-standard comm drivers. A number of communications packages replace the standard Windows comm driver with their own driver to optimise performance.

You can find out what comm driver you have installed by checking the comm.driv line in the [boot] section of your system.ini file.

Setting it to comm.driv=comm.driv will reset it to the standard Windows comm driver, as long the original driver hasn't been over written.

Why does my modem get disconnected?

Here are the most common reasons why a line would drop out:

1. The inactivity timeout in the Dialler/Settings screen may have been enabled unwittingly. If so, set it to 0 to disable the timeout.
2. Most ISPs have timeouts on their server to hang up the line if no data has been transferred within a certain period of time. Your ISP should be able to tell you if this is the case.
3. Phones or fax machines on the same line as the modem can create problems. For example, some phones charge a capacitor every 15 minutes to power their memory. Remove all phones and fax machines from your line to be safe.
4. Power surges from electric equipment (such as domestic white goods) may cause modems drop out. For example, if a fridge is plugged into the same circuit as a modem with a sensitive power supply then the modem may drop out whenever the fridge cycles. To eliminate this as a possibility, try disconnecting the suspected appliance.
5. Line noise! This can be a real problem, and is more common than we'd like to believe. If line noise is the problem, then reducing the baud rate in the File/Setup screen may help.

How do I optimise my MTU, RWIN, and MSS?

These are enhanced primarily by trial and error. However, we recommended the following values as a starting point:

	SLIP/PPP	Ethernet
MTU	576	1500
TCP RWIN	2048	4096
TCP MSS	512	1024

For more information see: [MTU](#), [TCP RWIN](#), [TCP MSS](#).

How can I use the Winsock offline?

A new feature, Login on network activity, is available from Version 3.0 onwards. With this feature enabled, the Winsock loads offline and only dials when an application tries to contact a remote host.

With version 2.1f and earlier, the Winsock can be configured to operate offline by setting the following:

1. Internal SLIP in the File/Setup screen.
2. No automatic login in the Dialler/Options screen.

Can I use another program as a dialler?

Yes. The Winsock can establish a SLIP/PPP link over a connection created by another dialler.

Before exiting the dialler to start the Winsock, don't forget to issue **at&d0**. This ensures that the modem ignores DTR signal when it is set low by the exiting program, otherwise the connection may be severed when the application closes the comms port.

In the Winsock's Dialler/Settings screen, make sure that either No automatic login selected OR Online status detection is configured correctly. Otherwise the Winsock may try to run the login script.

How do I switch between tone and pulse dialling?

If your login script is based on the login.cmd supplied with Version 3.0, then set either **dt** for tone dialling or **dp** for pulse dialling in the modem dial string. This can be found in the modem profile screen which can be accessed directly by selecting modem.cmd from the Dialler menu.

If your login script is based on an earlier version of the login.cmd then there will be a line within the script that outputs either "atdt" or "atdp" followed by the number. The command "dt" instructs the modem to use tone dialling, while "dp" represents pulse dialling.

For example, to use tone dialling:

```
output "atdt"$number\13
```

Or, pulse dialling:

```
output "atdp"$number\13
```

How do I send a "break" with Trumpet Winsock?

The break command can be used to send a break signal. One way of using would be to write a script that consists of one line:

```
break
```

Save this in the same directory as the Trumpet Winsock's files (usually c:\trumpet), and give it an appropriate name such as break.cmd. To run this script select break.cmd from the Dialler menu.

Alternatively, you could have a script to output the modem's escape code so that it returns to command mode, then output the AT commands to transmit break ("at\b") and return modem to data mode ("ato").

For example, within a script this would appear as:

```
output "+++"  
input 10 OK\  
output "at\b"\  
output "ato"
```

Can I setup the Winsock for multiple Internet Service Providers?

It is possible to configure the Winsock for multiple Internet Service Providers (ISPs), but it requires some tweaking. There are two methods...

- Method One... Easy but ugly
- Method Two... More complicated but cute
- Method Three... Coming soon!

Method One... Easy but ugly

Using this method you will have two (or more) full copies of the Winsock in as many different directories. This "brute force" method is effective, but it means that if you update the Winsock you will have to update each copy. Also, you risk getting the various copies confused.

1. Create a new directory on your machine and copy all of the Winsock files into it.
2. Create a program item for the new copy of tcpman.exe.
 - An easy way to do this is by making a copy of your existing Trumpet Winsock (ie tcpman.exe) icon, and then selecting Properties from the File menu and modifying the copy's Command line.
3. Start the new version of the Winsock and:
4. Select Setup from the File menu and modify the parameters as required (In particular DNS server and Domain suffix).
5. Select setup.cmd from the Dialler menu to access and edit the phone number, username and password.
 - The login.cmd script may need to be altered for each provider. Do this using the instructions in the Help file.

Method Two... More complicated but cute

If you would prefer not to maintain two separate copies of the Winsock, then there is a solution.

By default, the Winsock stores all its settings in a file called trumpwsk.ini. You can use the -inipath=<path> command line option to specify an alternate *.ini file.

To do this:

1. Create a program item for the Trumpet Winsock that will use the alternate *.ini file.

An easy way to do this is by making a copy of your existing Trumpet Winsock (ie tcpman.exe) icon, and then selecting Properties from the File menu and modifying the copy's Command line.

```
c:\winsock\tcpman.exe -inipath=c:\winsock\inifile1.ini
```

```
c:\winsock\tcpman.exe -inipath=c:\winsock\inifile2.ini
```

2. Start the new version of the Winsock and:

Select Setup from the File menu and modify the parameters as required (In particular DNS server and Domain suffix).

Select Profile from the Dialler menu to access and edit the phone number, username and password.

So far this is fine, but one problem remains.... What if different login.cmd scripts are required for each ISP?

The solution is to use only one login.cmd file that contains all of the required scripts. Once you have a working *.cmd script for each provider you can integrate them using the following steps:

To indicate which *.ini file is being used (and therefore which script will be needed) add a new variable (say \$isp) to the [default vars] section of each *.ini file. For example:

```
[default vars]
$number=*****
$username=*****
$password=*****
$isp = "isp1"
```

Don't forget to set the value of \$isp differently in each *.ini file!

The login.cmd script can be modified to execute one section of itself if the \$isp is of a particular value and another section for a different \$isp value. The new login.cmd file may look something like:

```
load $isp
if $isp = "isp1"
# login script for first isp
end
if $isp = "isp2"
# login script for second isp
end
```

Method Three... Coming soon!

A third alternative is to use the new include command to simplify the process of incorporating different scripts. An example that makes use of this feature will soon be available on our Web site.

How do I get bye.cmd to hangup my modem?

The bye.cmd script that is currently supplied with the Trumpet Winsock includes two alternative methods of hanging up the modem. To try them out, edit the line that sets the %which variable to either 1 or 2.

Programmer's notes

- Where can I get information about the Windows Sockets API?
- Is there a list of "Trumpet Winsock specific" functions?
- Can I have the source to anything written by Trumpet Software?
- Do you have a Visual Basic tool kit?
- Does the Winsock support IP multicasting?

Where can I get information about the Windows Sockets API?

Stardust Technologies & Stardust Winsock Labs:

<http://www.stardust.com/>

Stardust Technologies' Winsock Resource Center:

<http://www.stardust.com/wsresource/wsresrce.html>

Microsoft:

<http://www.microsoft.com/>

Is there a list of "Trumpet Winsock specific" functions?

No. We do not make this information available for a number reasons:

These functions calls are subject to change without notice. By publishing private function calls we would be making a commitment to support these calls in future releases of the Winsock. Maintaining backward compatibility increases the baggage that comes with each new release.

By writing your software such that it conforms to the Winsock API, you ensure greater compatibility between your software and other vendors' (ie non-Trumpet) Winsocks.

Can I have the source to anything written by Trumpet Software?

No. We don't release the source code of our applications or utilities under any circumstances.

Do you have a Visual Basic tool kit?

No. There is no specific Trumpet Winsock Visual Basic tool kit.

Does the Winsock support IP multicasting?

The Trumpet Winsock does support multicasting, but it is not fully IGMP compliant.

To program for multicasting all you need to do is specify the right multicast address in the connect/bind/sendto calls.

A sample program is available at:

http://www.trumpet.com.au/wsk/faq/test_mc.txt

Windows 95 questions

- How does the layer work?
- Does the Winsock work under Windows NT?
- Compatibility with 32-bit Winsock applications
- Trouble spots
- Fixing lock ups

How does the layer work?

This release is not a full 32-bit version of the Trumpet Winsock.

Rather it includes a “thunking” layer which allows 32-bit applications to utilise the 16-bit Trumpet Winsock under Windows 95.

In fact the main parts of the Winsock, tcpman.exe (the dialler and interface) and winsoc.dll (the core of the Trumpet Winsock), will continue to operate with 16-bit applications under both Windows 3.1x as well as Windows 95.

When a 32-bit application requires the Winsock, it loads wsock32.dll which in turn loads twsk16.dll. Most functions are passed directly through to the winsoc.dll. The exceptions are the getXbyY routines, and the WSAAsyncGetXbyY routines.

Since special Windows 95 calls are used to create the layer between 32-bit applications and the 16-bit Winsock, it will not work under Windows NT (or Win3.1).

Does the Winsock work under Windows NT?

No. Version 3.0 uses special Windows 95 calls to create a layer between 32-bit applications and the 16-bit Winsock.

For more information about the layer, see [How does the layer work?](#).

Compatibility with 32-bit Winsock applications

All the functions supported by the regular Trumpet Winsock are available.

However, the GetXbyY routines, and WSAAsyncGetXbyY routines have required recoding into 32-bit and are currently cut down in function and may be less robust.

The following 32-bit Winsock applications have been tested with this Winsock:

Program	Comments
WinQvtNet 32	No apparent problems.
WinVN	No apparent problems.
MS Telnet	No apparent problems.
MS FTP	No apparent problems.
Super Mosaic	No apparent problems.
Netscape 32	No apparent problems.
WS_FTP 32	Seems to work correctly, however we are uncertain as to what will happen when the memory compacter starts to shift things around in memory. This may lead to problem when physical memory gets low.
HotJava (alpha 3)	Crashes regularly. Reason unknown - Sun acknowledges this to be a bug that is independent of the Win95 Trumpet Winsock. We have Soft Ice'd the program to find that the routine GetThreadContext may be related to the bug.

Trouble spots

Since the Winsock is required to manage it's resources with the possibility of being preempted, this version of the Winsock now contains special locks to prevent reentrancy. This has required the use of the yield() function, which may have an impact on some 16-bit programs. Only rudimentary testing of 16-bit programs has been done in this area.

Should an application crash while a Winsock lock is being held (for eg. HotJava) the lock may still be left on. The Winsock attempts to detect this deadlock condition and clears the lock. However, in the event of this happening, the Winsock may be left in an unstable state. All tests have shown that the Winsock remains stable in such a case, but stability cannot be guaranteed under those conditions. An error message will be displayed on the Trumpet Winsock's display should this arise. If the Winsock has been unable to recover from such an error condition, it may remain deadlocked.

To clear such an error, you can try to terminate the Trumpet Winsock tcpman.exe task using the Ctrl-Alt-Del combination. A Windows restart may however still be required occasionally.

WS_FTP32 uses a Winsock callback function. While it seems to work correctly, I'm not too sure what will happen when the memory compacter starts to shift things around in memory. Watch out when physical memory gets low.

Fixing lock ups

You will find that some apps will not respond, but Win95 will still work for 32-bit apps.

Press Ctrl-Alt-Del. You should get a list of applications that are not responding. (Win95 may not show these until some time has elapsed).

Try to kill off any locked 16-bit applications first. This should usually clear the problem. Be patient, Win95 may not terminate them immediately.

Then close down all other winsock applications. Chances are that the Winsock has been left in a tricky state. It will probably still work, but will operate less efficiently than normal. If the cause of the problem has been a 16-bit application, the Winsock should close normally. **IT IS RECOMMENDED THAT YOU ALWAYS CLOSE DOWN THE WINSOCK WHENEVER THERE HAS BEEN A LOCK UP.**

If you have had to kill off any 32-bit Winsock applications, you will need to restart Windows. This is because the files winsock.dll, twsk16.dll and wsock32.dll will still be locked in memory even though no Winsock apps may be running. You can do this by either rebooting normally, or shutdown/restart in MS-DOS mode and then typing "exit".

Please note the distinction between closing an application normally and killing off an application. Closing is done by normally exiting the application using the close button or similar mechanisms. Killing an application is done by using Ctrl-Alt-Del and following the Win95 terminate program dialogs.

Glossary

This section contains various bits and pieces that didn't seem to fit anywhere else.

- Command line
- Crynwr packet driver collection
- Trumpet files
- Definitions

Command line

The inipath command line option can be used to override the default trumpwsk.ini file.

This is useful if you want to use the Trumpet Winsock under a variety of situations. For example, one configuration for a packet driver setup and one for a dial-up PPP connection. Or configurations for two different Internet Providers.

The command line is accessible by high-lighting the Trumpet Winsock icon from the Program Manager and selecting File/Properties from the menu.

For example:

```
C:\winsock\tcpman.exe -inipath=C:\winsock\dialup.ini
```

Note: The entire path name must be given.



















Crynwr packet driver collection

The Crynwr (formerly Clarkson) packet-driver collection is furnished by Russ Nelson and available from the Columbia University Academic Information System's FTP site at:

<ftp://kermit.columbia.edu/packet-drivers/>

Trumpet files

Here is a list of files written by Trumpet Software International Pty Ltd that are included with the Trumpet Winsock package.

-  winsock.dll
-  tcpman.exe
-  wsock32.dll
-  twsk16.dll
-  trumpwsk.ini
-  hosts
-  services
-  protocol
-  trumpwsk.hlp
-  login.cmd
-  profile.cmd
-  modem.cmd
-  server.cmd
-  bye.cmd
-  Tcpmeter
-  Hop
-  Dig
-  Ping

winsock.dll

The core of the Trumpet Winsock's TCP/IP stack. This section does all the work, but takes its leads from [tcpman.exe](#).

Every Winsock package includes a winsock.dll, but each one is written by the vendor of that Winsock.

tcpman.exe

The part of the Trumpet Winsock that controls how the winsock.dll communicates with the Internet.

It makes use of the trumpwsk.ini file to store its settings.

wsock32.dll

The 32 bit interface to the Winsock under Windows 95. With [twsk16.dll](#), this file provides a “thinking layer”.

For more information about the layer, see [How does the layer work?](#).

twsk16.dll

Assists wsock32.dll.

trumpwsk.ini

Contains most of the settings for the Trumpet Winsock.

See [Command line](#) if you wish to use more than one .ini file.

Although editing the .ini file settings directly is not advised, experienced users may find it helpful to know about the following parameters:

ip_buffers	The number of buffers for internet frames.
pkt_buffers	The number of buffers specifically for the packet driver.
slip_logging	Takes a log of a dial up <u>SLIP/PPP</u> session, and saves it as usage.log.
font	Sets the font of the Trumpet Winsock's display.
slip_rcvbuf	Size of the outgoing comms ring buffer.
slip_sndbuf	Size of the incoming comms ring buffer.

Note that some of the network parameters may be overridden by environment variables, which have the same names as the trumpwsk.ini parameters.

hosts

Provides a list of host names. The Winsock checks this list first whenever it looks up an entry in the DNS server.

services

List of Internet services.

protocol

List of Internet protocols.

trumpwsk.hlp

The online help version of the Trumpet Winsock documentation. It uses Microsoft Help and can be run directly from the File Manager or Program Manager as required.

The Trumpet Winsock Help file was created from a Word for Windows document using Roger Hadgraft's very helpful macro, Doctor Help. Doctor Help translates any reasonable Word file into a Windows Help file (with the aid of the Help Compiler of course).

For more information regarding Doctor Help, contact roger.hadgraft@eng.monash.edu.au.

login.cmd

This is a sample login script that will work with most modems and SLIP/PPP servers once the settings in the Dialler/Profile have been set correctly.

It is significantly different to the sample script provided with earlier versions of the Trumpet Winsock. Throughout this document, *login.cmd* and *login script* refer to the standard login script supplied with Version 3.0, unless otherwise stated.

Since everyone's login sequence will be different, it will probably be necessary for you to configure your profile and/or login script.

See Logging in for the first time for more information.

profile.cmd

A script that allows you to configure the login phone number, username and password of your SLIP/PPP account.

This information is saved in your trumpwsk.ini file.

See also [Profile](#).

modem.cmd

A script that allows you to configure settings relevant to your modem.

See also [Modem profile](#).

server.cmd

A script that allows you to configure settings relevant to your SLIP/PPP server.

See also [Server profile](#).

bye.cmd

A sample bye script.

Tcpmeter

Tcpmeter provides a graphical representation of incoming and outgoing network traffic.

Incoming traffic is displayed across the top of the screen in green, and outgoing is across the bottom in red.

Each coloured bar indicates a fraction of the vertical scale. For example, if the vertical scale is 4K bytes/sec and the green bar comes $\frac{3}{4}$ of the way down the screen, then you received 3Kb in that second.

The vertical scale is displayed in the title bar of the window and can be changed by clicking the left mouse button to increase it and the right mouse button to decrease it.

Hop

Hop provides a list of the *gateways* that are traversed by a packet on its way to a destination Host address.

On the Internet, any two machines are joined by a number of physical networks. A machine that directs traffic between networks is called a gateway.

To generate a list of the machines between yours and a destination machine, enter the name of the machine in the **Host** field and select **Start**.

A list of the machines between yours and the destination machine will be displayed.

If “***” is returned, it means that although a gateway has been passed through, no information has been passed back.

Dig

Dig uses the Domain Name System (DNS) to answer queries about a domain or host on the Internet.

To use this you will need to know the address of a DNS host, or name server. If you're not sure, just use the name server from the Trumpet Winsock's File/Setup dialog.

Once you have entered the address of a name server in the **DNS Host**, enter the address of a domain (or host machine) in the **Target** field and select **Search**. You can select the **Type** of record to be requested if you are looking for specific information.

If the DNS host you entered is unreachable (it may not be functioning), the message "Timed out" will be displayed. In this case, check the address of the name server or try another one.

Parameters

Target	The address of a domain or host.
DNS host	Usually your provider's name server.
Type	ns Name server. a Address. soa Status of authority. any Usually the same as address. mx Mail exchange.

Ping

Trumpet Ping is a diagnostic program that allows you to send test packets to a known host on the Internet. It is useful for checking your TCP/IP connection.

Parameters

Host	Enter the address of a host machine.
Size	The size of each packet in bytes.
Interval	How long to wait before sending the next packet.

To test your connection, enter a known host name or IP address in the **Host** field, and press **Start**.

If you enter the name of a machine, the first thing Ping does is to convert it into a numeric IP address. This may happen so quickly you don't even notice it.

If the status line seems to be stuck on a message saying "Looking up <host name>", this may indicate that there is a problem with the DNS system (DNS is the system convert a the name of a machine on the Internet into its numeric IP address). Check that the Name server address in your Winsock is set correctly.

An "ICMP transmit error" will usually be generated either because the network is down or because of incorrect Comm port settings.

At any time you can press **Stop** to cancel the test and display statistics.

There is no timeout, so if you're sending packets and getting no response, the status line will not change from "Trying <some numeric IP address>....". In this case, press **Stop**.

The display will then list how many packets were sent and received and a percentage of how many were

lost. Statistics are also presented that give you an idea of how long it takes for the packets to do the circuit back to your machine. This is called the round trip time (rtt).

Definitions

- Buffers
- BOOTP
- Comm driver
- CSLIP
- Firewall
- PAP
- RARP
- Shim
- Socks 4.2
- Token ring
- Vectors
- Virtual machine
- WINPKT
- Windows Sockets 1.1

Buffers

Network buffers (both IP and packet buffers) are sections of memory allocated from RAM to store packets of 1500 bytes each. There are two pools, the main IP pool (32 buffers by default) and the packet driver pool (16 buffers by default).

An IP buffer is not quite the same as a packet buffer, although they do have the same structure. Only the packet driver pool needs to be in Global DOS memory. It has to be from there so that the packet driver (which is a DOS TSR) can copy data into them when a packet arrives (and from them when packets are sent). For this reason, packet buffers are not allocated under SLIP/PPP configurations. See also Comm errors [OVERRUN].

When the Winsock is configured for SLIP/PPP, two comms ring buffers are allocated. These are the receive and send buffers. See also Comm errors [RXOVER].

BOOTP

Bootstrap Protocol (BOOTP) is an alternative to RARP that allows a machine to obtain its IP address, gateway address, and Netmask using UDP (User Datagram Protocol) and IP.

Therefore, BOOTP can be used over a dial up SLIP/PPP connection if required.

Comm driver

When the Trumpet Winsock is configured for SLIP/PPP it is normally being used over a modem. A comm driver allows the Dialler of the Trumpet Winsock to access the modem.

The comm driver (or comm.drv) is set in the [boot] section of your system.ini file.

CSLIP

Compressed SLIP (CSLIP) is a more efficient version of SLIP, that features VJ Header Compression.

CSLIP is not necessarily supported by all SLIP servers.

Firewall

A firewall is broadly defined as anything that protects one network from another.

Use of a firewall and a program such as Socks 4.2 can enable a computer to access the full resources of the Internet without actually existing on the Internet. It does this by having the public side of the firewall communicate to the Internet on behalf of any machines on the Local Area Network (LAN). Machines on the LAN speak to the local side of the firewall using TCP and know nothing about the full Internet. Socks bridges the gap between the two sides of the firewall.

Here are some reasons why a LAN might use a firewall to connect to the Internet:

- Without a firewall your machine's IP address (and name) can be seen by anybody else on the Internet. This information can be hidden by a firewall, since only the public side of the firewall is visible.
- Firewalls can free up space for IP addresses, as the public side of the firewall can represent the entire LAN with one IP address. As long as the machines on the LAN are only connected to the Internet through the firewall they can use any address. There are many "official non-Internet addresses" reserved for this purpose.
- The services that will be made available to users on the LAN can be regulated, as a program like Socks can choose what protocols to support.

PAP

Password Authentication Protocol (PAP) is an authentication system built into PPP that should only be enabled upon instructions from your internet service provider.

PAP requires a username and password and can be enabled in the File/PPP options screen.

RARP

Reverse Address Resolution Protocol (RARP) is a low level Internet protocol used by diskless machines to obtain an IP address from a server over a physical network.

An alternative to this scheme is BOOTP.

Shim

In the context of the Trumpet Winsock, a shim refers to a *packet driver shim*, a program that provides a packet driver interface to another method of network connection, such as ODI and NDIS.

See [Using the Trumpet Winsock over a packet driver](#) for more information.

Socks 4.2

Socks is a program that permits communication across a firewall. For more information regarding Socks, see:

<ftp://ftp.nec.com/pub/security/socks.cstc/>

Token ring

Since the Winsock can only use a class 1 or 6 packet driver, token ring is only available by way of a shim such as oditrpkt or the `ibmtoken` packet driver.

Vectors

The 80x86 family of processors allows programs to communicate with the operating system through what is called a "software interrupt", which always has a number in the range 0 to 255. This is termed a "vector" and is one of the key mechanisms to pass control to the MS-DOS operating system. Usually the vectors are expressed in hexadecimal, with the range 0x00 to 0xFF. The 0x in front of the number means that we are using hexadecimal numbers instead of decimal numbers. They may also be expressed in the notation 00H to FFH, or \$00 to \$FF. If you are dealing with packet drivers, hexadecimal notation is much more common, but occasionally they are expressed in decimal. Examples of software interrupts in use on PCs are 0x10 for the video BIOS, or 0x21 for calls to DOS.

Virtual machine

A virtual machine can be either the entire Windows session, or any DOS session active within Windows. Refer to the Windows system documentation for more details.

WINPKT

WINPKT is a virtual packet driver interface for Windows.

The program WINPKT was written by some clever people on the Internet to allow a packet driver interface to work correctly within Windows by making sure that packets get directed to the correct virtual machine under Windows enhanced mode.

Within Windows, packet drivers can only be used reliably using WINPKT, and even then only under enhanced mode.

It is available from our FTP site at:

<ftp://ftp.trumpet.com/winsock/winpkt.com>

Note that NDIS3PKT replaces WINPKT, since it offers exactly the same function.

Windows Sockets 1.1

For a full description of Windows Sockets, see the Stardust Technologies & Stardust Winsock Labs:

<http://www.stardust.com/>

See also [Programmer's notes](#).

Contact Details and Online Services

The following services are intended to support our clients by providing the latest news as well as gathering feedback about our products.

Our mailboxes are for questions that are not satisfactorily addressed by this document or our online services. Before writing, please check the [Trumpet Winsock FAQ](#) lists both in this document and at our [Web site](#).

- [World Wide Web](#)
- [Technical support email addresses](#)
- [General email addresses](#)
- [The trumpet.* newsgroups](#)
- [FTP](#)
- [Telephone and fax numbers](#)
- [Postal address](#)

World Wide Web

Not surprisingly, we have a Web site. Its URL is:

<http://www.trumpet.com.au/>

Our Web site is particularly useful as a supplement to this document by being the most up-to-date source of information. It also allows us to provide information that would be too bulky for this document if given in full.

Technical support email addresses

Before requesting technical support, please make sure you've searched through the [FAQ](#) lists both in this document and at our [Web](#) site.

To help us diagnose your problem quickly, it is important that you send any relevant details. For example: versions of the software that you are using, your operating system, etc.

Choose the right mailbox for your question carefully. Misplaced email may not be answered!

reg-support@trumpet.com.au	This is where registered users may send their technical questions. As a registered user you will have priority, but you must include your registered name at the start of your message.
winsock-support@trumpet.com.au	Technical questions regarding the Trumpet Winsock.
newsreader-support@trumpet.com.au	Technical questions regarding the Trumpet Newsreader.

General email addresses

Before sending a request, please make sure you've searched through the [FAQ](#) lists both in this document and at our [Web](#) site.

Choose the right mailbox for your question carefully. Misplaced email may not be answered!

info@trumpet.com.au	This is for general information about our products that isn't addressed by the other mailboxes. Typically this kind of information will be accessible from our Web site; please look there first!
sales@trumpet.com.au	Write here if you are interested in either registering your copy of one of our products, purchasing a site license, or negotiating a distribution agreement. Check our Web site first for a current price list.
registrations@trumpet.com.au	This is for following up registrations. Only write here if your query concerns an order in process, or you're having trouble with the Internet registration or Update registration features.
trumpet-bugs@trumpet.com.au	This is for informing us of bugs in our products. We don't normally reply to these directly, so if you want answers please write to one of the technical support addresses.
suggest@trumpet.com.au	As we look for ways to improve our products, we'll refer to mail sent here. Don't expect a response though!
www-admin@trumpet.com.au	Comments and queries about our Web site.

The trumpet.* newsgroups

You can ask questions, or discuss any aspect of any Trumpet program through these Usenet newsgroups. Feedback is always welcome.

If your provider's newsroom doesn't receive these groups you can access them by pointing your

newsreader at **newsroom.trumpet.com.au**.

This site maintains the following private news groups:

- trumpet.announce
- trumpet.bugs
- trumpet.feedback
- trumpet.questions

If you're using Netscape, you can access these groups by going to:

news://newsroom.trumpet.com.au/trumpet.*

FTP

The latest copies of all our freely accessible software and pre-releases are available by anonymous FTP from:

<ftp://ftp.trumpet.com/>

Or in Australia:

<ftp://ftp.trumpet.com.au/>

This site can be particularly busy at times, so if you are unable to connect please try one of our mirror sites. Our mirror sites are listed at:

<http://www.trumpet.com.au/mirror.htm>

Telephone and fax numbers

Our Australian office is open from 9am - 5pm Monday to Friday (GMT+1000).

Country or Region	Phone	Fax
USA	1800 488 4236	1800 488 3314
Canada	1800 856 8918	1800 856 8922
United Kingdom	0800 96 7294	0800 96 7295
International (until 8/96)	+61 0245 0220	+61 0245 0210
International (after 8/96)	+61 3 6245 0220	+61 3 6245 0210
Australia	1800 67 0220	1800 67 0210
Hobart, Tasmania (until 8/96)	(002) 45 0220	(002) 45 0210
Hobart, Tasmania (after 8/96)	(03) 6245 0220	(03) 6245 0210

Postal address

Trumpet Software International Pty Ltd
GPO Box 1649
Hobart
Tasmania 7001
AUSTRALIA





Registration Details

Registration of the Trumpet Winsock is encouraged since it not only funds further development of the software, but also gives you a warm fuzzy feeling deep down inside. Furthermore, it allows you to keep using the Trumpet Winsock after 30 days.

The basic registration fee for a single user version of the Trumpet Winsock is AU\$25 for Australian users and US\$25 for international (non-Australian) users (valid until 31-Dec-96). See the [price list](#) for details on multi-user site licenses.

As a registered user, you will receive support (within the existing [Capabilities of the Trumpet Winsock](#)) for up to 12 months from the date of purchase. Support will include any upgrades or bug fixes released within that 12 months.

After 12 months, upgrades for previously registered users will cost 25% of the current registration fee per annum. Arrangements will also be made for conversion of smaller licenses to larger ones.

-  [How to register](#)
-  [Price list](#)
-  [Distribution Rights](#)
-  [Order form](#)

How to register

Please fill out the order form in clear print and send it along with your payment/purchase order. We accept the following methods of payment:

- Internet registration** You may email your credit card details to us using the Internet registration feature of the Trumpet Winsock, which is located under Special menu.
- Credit card** We accept Mastercard, VISA, Bankcard, American Express and Diners Club. Credit card details may be given by fax or phone. See Contact Details for our fax and phone numbers. Credit card details may only be emailed using the automatic Internet registration option.
- Direct bank transfer** Direct bank transfers can be made using the following bank account details. Please fax both the completed order form and details of the transfer (ie deposit slip). See Contact Details for our fax number.
- Account Title: Trumpet Software International Pty Ltd
- BSB: 087 421
Account number: 64892 8805
- Bank name: National Australia Bank
7 Bayfield Street
Rosny Tasmania 7018
Australia
- Cheque** Cheques should be drawn in favour of **Trumpet Software International Pty Ltd** and sent with the order form to our postal address via Air Mail.
- All cheques from international (non-Australian) users must be in US currency and drawn on US banks.
- We are also able to accept American Express international postal money orders.
- Please note that mail from the US to Australia can take up to 15 working days to arrive.
- Purchase order** We accept purchase orders for any amount, but will only invoice orders that are greater than \$100. Prepayment is required for purchase orders that are less than \$100. Purchase orders should be accompanied by the order form and faxed to us.

Price list

All prices are in Australian dollars for Australian users, and US dollars for international (non-Australian) users. For example, a single license costs AUS\$25 for Australian users and US\$25 for everyone else.

Prices valid until 31-Dec-1996.

Single User License

1 user	\$25
--------	------

Multi-user Site License

1-99 users	\$25 per user
100-499 users	\$2500 + \$10 per additional user over 100
500-999 users	\$6500 + \$5 per additional user over 500
1000-4999 users	\$9000 + \$2 per additional user over 1000
5000-9999 users	\$17000 + \$1 per additional user over 5000
10000-19999 users	\$22000 + \$0.90 per additional user over 10000
20000-49999 users	\$31000 + \$0.80 per additional user over 20000
50000-99999 users	\$55000 + \$0.70 per additional user over 50000

Multi-user Educational Site License

1-100 users	\$25 per user
100+ users	\$2500 per campus

A multi-user educational site license covers a single campus only. If more than one campus is to be covered, a multi-user site license costing structure with a 10% discount applies. This is restricted to students and staff using campus facilities. If dialling in off campus, students and staff must dial in through the school's services only.

A site license is a license to a specific named organisation and may not be sub-licensed or assigned to anyone else. Copies of the Winsock software obtained under a site license may only be used by employees of the licensed organisation or, in the case of an educational organisation, by employees or students of the licensed organisation. Payment under a site license is based on the number of installed users. A site license cannot be used by an Internet Service Provider to provide Internet services for its clients.

Distribution Rights

Trumpet Software International Pty Ltd reserves all rights in the Winsock program, including the rights of publication and of distribution. The Winsock program may only be made available from FTP sites which have been approved by TSI for that purpose and may only be made available from those sites in the unmodified **twsk30?.zip** file distributed by TSI. All other forms of making the program available or of distribution (and all reproductions of the Winsock program in connection with such activities) are prohibited without TSI's express written agreement.

TSI has a standard distribution agreement for use with its distributors and a standard scale of distribution charges. Please contact TSI at sales@trumpet.com.au for further information on TSI's distribution arrangements.

Order form

Trumpet Winsock Version 3.0

Ship to:

Bill to:

_____	_____
_____	_____
_____	_____
_____	_____

Please supply the following items:

- Single user license.
- Multi-user site licence for [_____] users.
- Multi-user educational site licence for [_____] users.

Total cost : _____

(AUS\$25 for Australian users and US\$25 for international (non-Australian) users (valid until 31-Dec-96). See the price list for details on multi-user site licenses.)

Method of purchase:

- Purchase order. We accept purchase orders for any amount, but will only invoice orders that are greater than \$100. Prepayment is required for purchase orders that are less than \$100.
- Cheque. All cheques from international (non-Australian) users must be in US currency and drawn on US banks.
- Bank Transfer. Direct bank transfers can be made using the bank account details found in the section on How to Register.
- Credit card. We accept Mastercard, VISA, Bankcard, American Express and Diners Club.

Name on card: _____

Card number: _____

Card type: _____ Expiry Date: _____

Signature: _____

Do you require a disk with the latest version of the Trumpet Winsock?

- No disk required.
- 3.5" disk.
- 5.25" disk.

How would you like to receive your password?

- Email.
- Fax.
- Postal mail.

Your registration name as will appear on the program (maximum 84 characters including spaces):

Email: _____

Phone: _____ Country: _____

Fax: _____ Date sent: _____

