

# **WinGate**

V1.3

## **Users Guide**

Copyright © 1996 Adrien de Croy

### **DISCLAIMER OF WARRANTY.**

This Software is provided on an "AS IS" basis, without warranty of any kind, including without limitation the warranties of merchantability, fitness for a particular purpose and non-infringement. The entire risk as to the quality and performance of the Software is borne by you. Should the Software prove defective, you and not Adrien de Croy or his Resellers assume the entire cost of any service and repair. You must determine that the Software sufficiently meets your requirements.

### **LIMITATION OF LIABILITY.**

Under no circumstances and under no legal theory, tort, contract, or otherwise, shall Adrien de Croy or his suppliers or Resellers be liable to you or any other person for any indirect, special, incidental, or consequential damages of any character including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. In no event will Adrien de Croy or his Resellers be liable for any damages in excess of Adrien de Croy's list price for a license to the software, even if Adrien de Croy or his Resellers shall have been informed of the possibility of such damages, or for any claim by any other party. This limitation of liability shall not apply to liability for death or personal injury to the extent applicable law prohibits such limitation.

# 1 Welcome to WinGate

Welcome to the new release of WinGate, If you are new to WinGate, or even if you are familiar with it, you should read through this document, as there have been a number of changes in the way WinGate operates which will affect you.

WinGate is basically a multiple proxy server, Telnet server, FTP Server, SOCKS (V4.0) server and extra bits rolled into one.

Set up correctly, WinGate will allow users on a Local Area Network (LAN) that is not directly connected to the Internet to access the Internet via a single machine on the LAN which is connected, either via a dial-up modem / ISDN connection, or second Ethernet interface. WinGate will allow you to use the following:

- Email
- World Wide Web
- FTP
- News
- Telnet
- WAIS
- RealAudio (new in 1.3.10)
- and others

WinGate can save you money by removing the need to get a separate dial-up account, with phonenumber and modem for each user that wants access to the Internet. Now they can all use the one link, at the same time.

## 1.1 System Requirements

The following systems are presented as a rough guideline for running WinGate:

### **Small LAN: 2-5 users**

- 486 DX2/66 8Meg RAM or better
- Windows 95 or Windows NT
- 28k8 modem
- TCP/IP installed

### **Medium LAN: 5-20 users**

- 486 DX2/66 20Meg RAM or better
- Windows NT
- 28k8 modem / ISDN connection, depending on usage
- TCP/IP installed

### **Large LAN: 20+ users**

- Pentium 90 with 32Meg RAM
- Windows NT
- ISDN connection / leased line / T1
- TCP/IP installed

## 1.2 Software Licensing

The previous versions of WinGate expired on a fixed date unless a license key was obtained and entered. This version has a different policy, outlined as follows.

WinGate 1.3.08 does not expire, however without a key it will only allow 1 user at a time to access the Internet at a time. This has the advantage in that I no longer need to issue free keys (which will save me a lot of time which I can put to better use developing WinGate further), and also it means that WinGate can be distributed to FTP archives, thereby taking the loading off the HTTP servers, and allowing more people access to the software.

You may register for a free expiring key for trial purposes. The key will be valid for unlimited concurrent users for a period of 30 days from the date of issue. I will not be sympathetic to people asking for extensions to their trial keys.

You may register with payment for a key that will not expire. Depending on your requirements, you may choose a license that suits. Keys are available for the following numbers of concurrent users.

- 2 Users
- 5 Users
- 10 Users
- unlimited

A concurrent user means any one machine that is connected to WinGate using TCP. WinGate will only allow connections from different machines up to the maximum licensed amount. So if you have a 2 user license, WinGate will accept connections from only two machines (these could be anywhere - not necessarily on your own LAN, if someone is connecting from the Internet) at any one time.

Prices may vary from time to time (normally due to exchange rate fluctuations). See below for information on how to find out more about WinGate.

## 1.3 WinGate home page

The document you are reading is updated with new releases, however, for the latest documentation, an FAQ, prices, and software updates, you should check the WinGate home page at:

*<http://www.deerfield.com/wingate/>*

Also of interest to any WinGate user will be the new WinGate Open Discussion Forum on the WWW, located here:

*<http://www.deerfield.com/wingate/hnroot.html>*

In the event that these pages move, an Internet search-engine query for WinGate should quickly locate it.

## **1.4 Updates**

Updates are generally released every 6 weeks or so. The next one is due out mid October , and contains substantial improvements and added features over the current release.

## 2 Features

WinGate provides the following features:

- SOCKS V4 Server
- WWW Proxy (HTTP only)
- FTP proxy
- Telnet proxy
- POP3 proxy (new in 1.3.08)
- Mapped Links
- Rules based Firewall
- Dial-on-demand
- DNS Server (new in 1.3.10)
- RealAudio Proxy (new in 1.3.10)

The SOCKS V4 server allows any SOCKS-compatible client application (such as Netscape) to operate fully, thereby allowing full FTP within Netscape, as well as Gopher access, and access to secure servers.

The WWW Proxy Gateway allows the use of WWW browsers such as Netscape or Mosaic to access the World-Wide Web. This release still only supports the use of HTTP protocol, so browsers should use the SOCKS server for other protocols.

The FTP Gateway allows use of FTP client applications that support the `username@hostname` method of firewall traversal. Examples of this are WS\_FTP, and CuteFTP, as well as command-line FTP clients. The FTP gateway also allows you to connect through another firewall using the same mechanism.

The Telnet Gateway allows use of Telnet clients to connect to remote servers.

The POP3 proxy (new in version 1.3.08) allows POP3 clients (e.g Eudora, Pegasus mail, Microsoft Exchange) to access POP3 servers to read their mail. Prior to version 1.3.08 a mapped link was used for this.

Mapped links (sometimes called “plugs”) are used to support applications which do not support traversing a firewall, or working through a proxy server. This includes applications such as most email packages and news readers.

The rules based firewall applies to all connections through WinGate. WinGate can very easily be set up to allow or deny connections to or from specified hosts, or URLs (Addresses for Web pages) containing specified words or parts of words.

You can monitor who is accessing the gateway.

Dial-on-demand allows more convenient use, by obviating the need to physically initiate a connection to an Internet Service Provider.

The DNS server allows applications that require DNS to run. Please note, this is not a full DNS implementation. WinGate DNS server implements only address record queries. This is however all that is required for full use of the access that is available through WinGate and covers well over 95% of all DNS traffic.

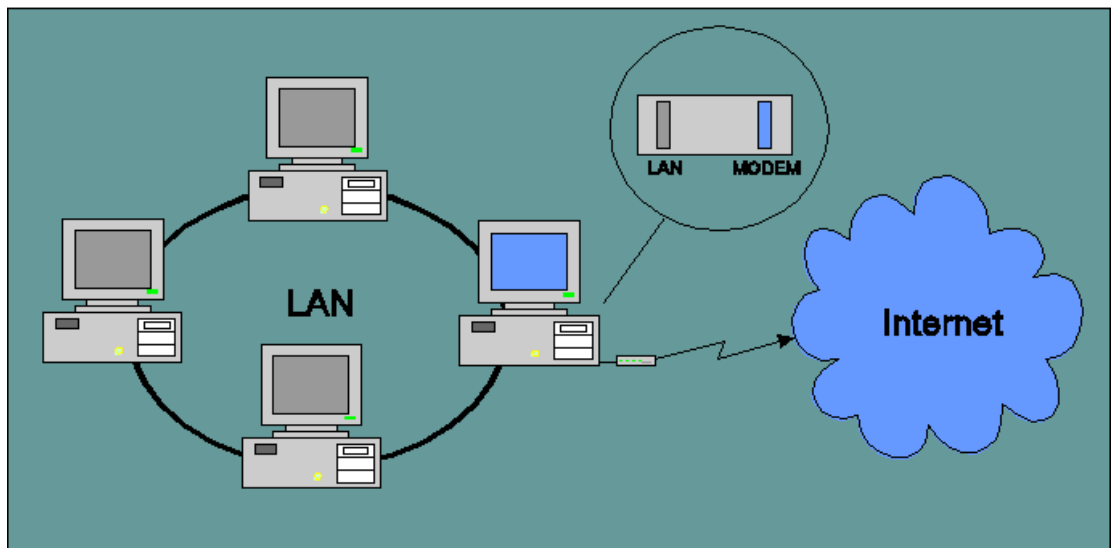
The RealAudio proxy allows you to listen to RealAudio broadcasts from a RealAudio Player 2.0 through WinGate.

## 3 Fundamentals

If you know a few fundamental things, then the concepts of WinGate become a lot easier, and it becomes correspondingly easier to set up for those services listed here, as well as other specific applications you may have.

### 3.1 How It Works

Consider the following picture. This is what in most cases the scenario will be for those that could use WinGate. There is a LAN, consisting of maybe 2 - 5 machines or more, and one machine has a modem, and a dial-up account to the internet.



The key thing to remember here is **interfaces**. In the Internet, every network interface (i.e. a LAN card, or a PPP or SLIP interface on a serial port, or a port on a router) has a unique address - an IP number of the form a.b.c.d (think of it like a phone number). Each interface in your computer is separate from all the others, so traffic appearing on one interface will not be seen on any other interface, **UNLESS** there is some specialised software which takes the traffic from one interface, and places it on another. This process is called **routing**, and enables the interconnection of networks (hence the term Internet).

Take for example the following parallel. You have 2 phones. Unless you have a PABX (equivalent of a router in the phone world), someone you are talking to on one phone can't hear what conversation is going on on the other phone, and vice versa - there is no direct means of communication between the parties at the other end of the two phone lines. If one of them wants to say something to the other one, they have to ask you to pass a message on. This is exactly the principle of a proxy server.

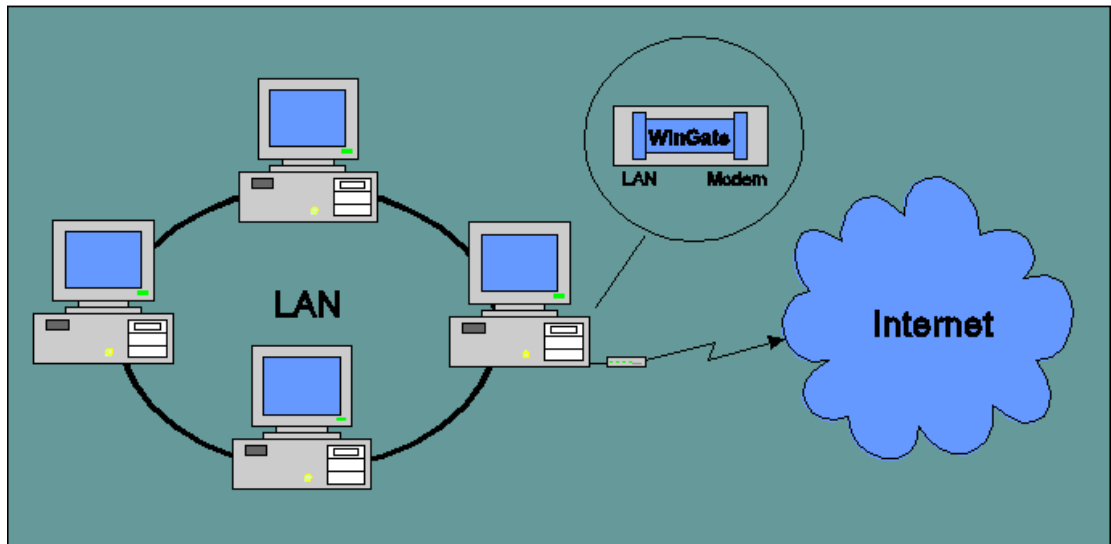
Most desktop operating systems do not support routing (notable exception here is Windows NT), so if you have a LAN and a modem, there is no way for traffic to get from your LAN to your serial port (i.e. to your modem and out onto the internet).

Historically the way to get around this was to get some specialised hardware called a router. A router is basically a very expensive computer with some very expensive software in it, which transfers packets from one interface to another, based on things



like where the packet is bound. They also do things like maintain a view of the network in order to reconfigure themselves if an interface becomes unusable (i.e a link goes down). Most people and small companies cannot justify the expense of a router in order to connect to the Internet.

Now consider the next picture



By using software such as WinGate on the machine with the Internet connection, traffic can be relayed between the interfaces. I must stress however that WinGate is NOT a true router. It does not route packets. It does however accept connections from one machine, connect to another, and relay data between them (whether the two machines involved are on the same interface or different ones). In this respect, WinGate is a firewall / proxy server. This allows machines on the LAN to communicate with machines on the Internet via the proxy.

### 3.2 Gateways

The first main thing to keep in mind is, that (under normal situations) when using the Internet through WinGate, at no time are you directly connected to any machine beyond your LAN (in fact unless you are routing on the gateway machine this would be physically impossible). You may seem to have a connection to the Internet, but this is only because WinGate is connecting out for you, and passing data back to you.

This means that in ALL circumstances, the machine wanting access to the Internet connects to the machine running WinGate (the Gateway machine).

One of the first questions that should spring to mind is “how does WinGate know where to connect to?”. The answer to this is simple - you have to tell it. Some applications can tell gateways where to connect to (i.e. Netscape, WS\_FTP) but some cannot (i.e. News, email, and many others). When the application cannot tell WinGate where to connect to, the user must preconfigure WinGate to connect to a given machine. This is where the Mapped Links feature of WinGate comes in. Mapped Links are a way of telling WinGate where you want it to connect you through to.

Most software that can talk through a proxy does not do so by default, so you will have to configure the software to use the gateway machine instead of trying to connect directly to the end destination (which it doesn't have access to).

### 3.3 TCP/IP

There are a couple of basic things you should know about TCP/IP that will make life much easier.

#### 3.3.1 IP Numbers

There are a few good analogies between computers and telephones, this is a very good example of one. You may think of an IP number as a telephone number with all the international dialling codes attached. This means that any machine can contact any other machine using the IP number, as long as there is a path between the machines.

This also means that no two machines on the same network should be allowed to have the same number. I say on the same network, because machine on unconnected networks often do have the same addresses, for a specific reason which applies here. You can think of this in the telephone world as you might know two people who work for different companies but who have the same extension number for the phone on their desks, there is no conflict (except perhaps in your mind).

#### 3.3.2 Ports

A port can be thought of as a channel of communications to a machine. Packets of information coming into a machine are addressed not only to that machine, but to that machine on a specified port. You can think of a port as a radio channel if you like, but the fundamental difference between a radio receiver and a computer, is that the computer can listen to any / all of 65000 possible channels at once!

However, the thing to note here is that typically the computer is not listening on very many ports at all. The computer will not respond to data or connection requests that come in on a port that it is not listening to.

One other thing to note, is that there are a number of important predefined ports which are universally used for various services. The major ones of these are:

Service	Port#	Description
FTP	21	File Transfer Protocol - for transferring files
Telnet	23	for logging into an account on a Remote Host
SMTP	25	For Sending mail
Gopher	70	Text menu based browser
HTTP	80	WWW protocol - Netscape, Mosaic
POP3	110	Downloading Mail
NNTP	119	Internet Newsgroups
IRC	6667	Internet Relay Chat
CompuServe	4144	CompuServe WinCIM communications
AOL	5190	America Online
MSN	569	Microsoft Network

#### IMPORTANT NOTE

Two applications on one computer generally cannot both listen on the same port at the same time. This is relevant in that if you try to set WinGate to listen on a port that some other application is already listening on (i.e. you are already running an FTP server), it will not be able to start listening.

If you find you get the error message when you start WinGate “Cannot start socket listening”, then you can be pretty sure this is the reason. What you then need to do is check in WinGate to see which local port has come up as disabled, and change that service to listen on a different port. Port numbers above 1024 are normally safe to use. Remember to update the client applications that were relying on that service being on that port to the new port number.

### **3.4 DNS**

These three letters don't mean much to many people, but they stand for the Domain Name System. Those three words don't mean much to many people either, so enter Adrien's telephone parallel #4. In a world without telephone books, you can think of a DNS server as a telephone directory service. To talk to any machine, unless you have the number already, you are going to have to look it up. You contact the directory services, and give them the name, they give you the number. You cannot contact a machine without a number. In the same way, you can think of the “HOSTS” file in your system directories, as a personal address book. If you want to contact a machine, you first look in your personal address book for the number, if it isn't there, you ring directory services (by the way, if you don't have a number for directory services, you can't look anyone up).

OK, so now you're armed with practically everything you need to get going.

## 4 Using WinGate

If you keep in mind the concepts of the previous section, using WinGate should be easy. For setting up client software (i.e FTP clients, WWW browsers, mail clients, newsreaders - everything), **in all cases there are two main steps:**

### Step 1

you must configure WinGate to gate traffic for the service you are setting up (i.e. tell it to listen on a port corresponding to the port number that service need to use - e.g FTP is 21)

### Step 2

you must configure your client applications to use the gateway machine (i.e. tell them to talk to the gateway on the port WinGate is listening on). If your application does not support proxies, then you will have to use a mapped link (which is protocol-less) and you must tell WinGate where to pipe connections through to.

### 4.1 Setting up your LAN

The first thing you must do is set up your LAN to use TCP/IP. This is probably the hardest part of all, however, if you follow the following guidelines it should be manageable.

There are some changes here from the 1.0 release. One of the quirks of the SOCKS protocol, is that a request for a connection is made in the form of a request for connection to / from an IP number. This means that a SOCKS client needs to have DNS in order to supply the number to the SOCKS server.

For this reason, the DNS server was added to WinGate. If you already have DNS on your internal network, and it has sufficient scope to resolve all the names you wish to connect to, then you will not need to run the DNS server in order to use the SOCKS server. You should not **under any circumstances** enable the DNS server in WinGate if you are already running a DNS server on the same machine - this will mess up your DNS server.

You WILL need to enable the DNS on your LAN however.

If you are using the DNS server in WinGate, you should set the DNS Server settings for your LAN adapters (on all machines EXCEPT the gateway machine) to be the IP number of the gateway machine.

There are a lot of very good resources on the Internet which will help you to set this all up. In particular, the following page will most likely be able to help you if you run into difficulties:

<http://www.windows95.com/connect/lansing.html>

Remember back in Section 3 when I said that IP numbers have to be unique for machines on the same network? Well, you can think of the entire Internet as a single network. But your LAN is probably not on the same network, even if one of the

machines (i.e. the gateway machine) is on the Internet. You see, it is not so much a computer being on the Internet as a **computer interface**, whether this be a LAN card or a serial port to your modem. The Internet can see the interface that is connected, but no further.

This means that you can choose any number you like for the machines on your LAN. However it isn't a good idea to choose just any old thing, because you have to think of the situation the gateway machine is in.

The gateway machine can see the entire Internet, and your LAN. So, you don't want to confuse it by giving your LAN the same addresses that the gateway machine can see on the Internet.

Fortunately some smart person already thought of this one, and so a whole heap of addresses have been kept aside for just this purpose. These addresses are called "private addresses" and are not meant to be available anywhere on the Internet. Therefore, by using them on your LAN, there won't ever be any confusion for the gateway machine.

Depending on the number of machines on your LAN, you probably will want to use a c-class (256 interfaces) address range. A good one to use is

192.168.0.\*

the corresponding subnet mask will be 255.255.255.0

Anything in the range 192.168.0.x through to 192.168.255.x is usable for these purposes.

You should set up your LAN using numbers in this range. You should need no other settings in the TCP/IP setup of your LAN machines, except that on the gateway machine, you need an entry for DNS server, which will be the IP number given to you by your service provider. If you have been using the Internet before getting hold of WinGate, then this will have already been set up for you.

#### **Note**

For getting started, you may like to put an entry for the gateway machine in the hosts file of each of your LAN machines (or if you are running a DNS server, you should put it in there, and make sure all your LAN machines are pointing to it - except for the gateway machine, which should use the DNS server of your Internet Service Provider).

An example may be (if you are using the 192.168.0.0 private c-class addresses on your LAN, which I recommend)

```
192.168.0.1    wingate
```

remember you must put an enter at the end of the last line in your hosts file, else it may not be recognised.

#### **Important Note for users of multi-segmented LANS**

If you are running a LAN with more than one segment (i.e with your own internal router[s]) then you will need to make a few modifications to your setup on the

WinGate machine, otherwise you will not be able to use WinGate from the other segments of your LAN that aren't directly connected to the WinGate machine.

The reason for this is as follows.

When your machine dials up your service provider, the PPP login sequence allocates a new **default gateway** for the machine that connects. This is done so that the machine connecting can access machines on the Internet through a router at your ISP. The side effect of this however is that their default gateway no longer points to a router on your internal LAN which was providing access to your other LAN segments. This means you have effectively lost the route to these other subnets.

The way around this is by adding static routes to the route table on the WinGate machine. This is done using the route command - available both on Windows 95 and Windows NT.

The syntax of this command is explained in the command itself, but generally the command(s) you will issues will look something like:

```
route add <subnet> MASK <subnet mask> <router address>
```

You will need one for each subnet on your LAN/WAN that needs access.

## 4.2 SOCKS Server

The SOCKS server allows any SOCKS compatible client to access the Internet to both connect out, and listen for connections as if the client was directly connected.

### Step 1

The port number for SOCKS servers is normally 1080.

The setting "Allow Unresolved Connects" may be required to access some sites. The explanation for this is as follows. When a client supplies an IP number for the SOCKS server to connect to, the server must check to see if the client is allowed to connect to the remote site. In order to check against the rule-base, the server needs to look up the name of the host from its IP number supplied. There may be some rule which denies connections to a site, and this needs to be checked.

If the server cannot resolve the name from the IP number (i.e because of a bad DNS entry somewhere, or incorrect DNS setup somewhere for the site concerned), then if the "Allow Unresolved Connects" is not set, the request will be denied.

As mentioned above, to use SOCKS, you must have DNS available on your LAN (see below)

### Step 2

Configuring clients requires specifying gateway as the socks server on port 1080. In Netscape, you should set the socks host to gateway, and clear all the other proxies. You can continue to use the WWW Proxy server for HTTP if you like, and this will give you the benefits of more meaningful error messages, and filtering of URLs.

There are a number of SOCKS compatible clients, for things such as FTP, Telnet etc.

### 4.3 DNS Server

The DNS server feature (under the Settings tab) allows you to process DNS requests using the resolver built into the TCP/IP stack of the machine running WinGate. This feature was new in version 1.3.10, and replaced the DNS forwarding (now obsolete). The DNS server should be more stable than DNS forwarding was.

#### TECHNICAL NOTE

This DNS server is only a partial implementation of the entire Domain Name System. However it does provide all the functionality you should need. Specifically, it processes requests for Address Records in the Domain Name System. This allows the function `GetHostByName(..)` to work from client software on your LAN. This allows your client software to look up the name of a site, which is all it needs to do for the amount of access it has.

Since the DNS server in WinGate simply uses the resolver of the machine it is running on to look up names, this means that any entries in the hosts file on your WinGate machine will then be visible on your entire LAN.

The DNS server allows you to use DNS on your internal LAN, so SOCKS and utilities like `nslookup` will work.

On the LAN workstations, add an entry in the DNS servers section of your TCP/IP setup specifying the IP number of the gateway machine. Unless you have `nslookup`, the only way to test if this is working, is by testing the SOCKS gateway with a SOCKS client (i.e Netscape) or by trying to ping a site (you will get request timed out, however the IP number of the site should be displayed).

You can now also specify if a DNS request will initiate a dial-in sequence. Some machines on your LAN may have resolvers (the part of the winsock that looks up the ip addresses for host names) which periodically refresh their cache. This has been known to cause WinGate to dial out seemingly of its own accord. Disabling this option allows you to continue to use the DNS server without sacrificing dial on demand. It does mean however that to kick off the dialler in WinGate, you will need to make a request other than a DNS request. Note: Netscape when trying to use a SOCKS server for access will send a DNS request first - this will no longer start the dialler, so you will need to have the HTTP proxy configured, and click on an HTTP URL to start the dialler. As HTTP URLs account for over 80% or all URLs on the WWW, this is not normally a problem.

## 4.4 WWW Proxy

### Step 1

The WWW Proxy Gateway currently handles only HTTP requests. If you look back at the table in Section 3, you may note that HTTP figures there, associated with port 80. So unless you are already running a Web server on the gateway machine, then port 80 is as good a port as any to pick for the WWW Proxy.

### Step 2

The HTTP proxy (i.e in WinGate the WWW proxy server) will handle all requests for HTTP URLs, so this will be the majority of your Web browsing. The SOCKS server will be used for any other protocol (i.e FTP, Gopher, WAIS, HTTPS).

Using both proxy servers means you get the benefit of meaningful error messages from WinGate whenever anything goes wrong with an HTTP request. The SOCKS server cannot send back error messages that the Browser can display on your screen, because the SOCKS server can't be sure that what it is talking to will understand HTTP.

Now if you are using Netscape, you go to Options → Network Preferences → Proxies → View , and set up

HTTP Proxy:	wingate	Port:	80
SOCKS Host	wingate	Port:	1080

If you are using MSIE, go to View -> Options -> Connections -> Settings, and type in

HTTP Proxy:	wingate	Port:	80
SOCKS Host	wingate	Port:	1080

In *All* browsers, leave the other options blank, or services like FTP will not work.

## 4.5 FTP Proxy

### Step 1

Set up an FTP Gateway. A good port for this is 21, unless you are already running an FTP server on the gateway machine, in which case you should choose another port number.

### Step 2

Configure your FTP client to use the gateway. To do this in WS\_FTP you go to Options → Session Options, and select “Use Firewall”. Make sure that the “Use PASV Transfer Mode” is not checked. The FTP gateway will return an error message if you try to use an FTP client in PASV mode. Save as default, and exit options.

Under Connect→Advanced enter



Firewall: wingate  
Port: 21 or whatever port number you chose

Select "User with no logon"

Now you're ready to rock and roll.

## 4.6 Telnet Proxy

### Step 1

Set up a Telnet Gateway. Unless you are already running a Telnet server, you may like to use port 23 (this will save you some hassles in your client applications).

### Step 2

Well, since Telnet is inherently a command-line based service, there is no special setup for the Telnet client. To use it however you must **always** first Telnet to the gateway machine on the port you set up for the Telnet gateway. You will then be presented with a prompt like this:

```
WinGate>
```

now you simply type in the name of the host you wish to connect to, and optionally a port number as well. WinGate will display "Connecting to ...", when the "Connected" message comes back, you are now connected through to the remote machine.

There have been a few changes with the WinGate telnet gateway, which affect telnet clients that issue telnet commands (e.g. EWAN, simpterm, and UNIX clients). You will probably get a double-echo while you are typing in the hostname you want to connect to. Once you are connected however, things should be alright again.

Some clients may also have problems with hitting enter. Try Ctrl-J or Ctrl-enter. If that fails, try running the telnet gateway on a different port number. This sometimes tricks telnet clients into thinking they are not talking to a telnet server, and so they don't send the telnet commands which confuse WinGate.

If you are always telnetting to the same host, you should use a mapped link instead - this will bypass having to type the host name in each time, and will allow full telnet negotiation.

## 4.7 POP3 Proxy

You should read this in conjunction with the following section (esp with respect to email).

### Step 1

Set up a POP3 Proxy in WinGate. If you are already running a POP3 server on this machine you will need to choose another port number for the POP3 Proxy. If your email clients do not allow you to specify the port number for POP3, then you will have to decide whether you really want to run the POP3 proxy to access POP3

servers on the Internet or not. If so, you will have to move the POP3 server you are already running onto another machine.

The one other thing you need to do to set this up is specify a delimiter character. I recommend the # key. This is used so you can specify the format of your POP3 username.

## Step 2

Configuring the email package to use the POP3 Proxy is easy. All email packages have a place in their setup for specifying the POP3 server (sometimes just called mail server) and the username and password. Eudora is a special case which has invented it's own system whereby it lumps the POP3 username with the POP3 server name into one construction which it calls POP account. Be sure not to confuse this with an email address, it most certainly is **not** an email address, although it may look just like your email address (anyone there at Qualcomm reading this - look seriously at getting rid of this construct in favour of a separate username and server field. I couldn't begin to count the number of support emails this confusion has caused me). Anyway...

What you need to do is this. Your POP3 server becomes the WinGate machine (e.g. "wingate"), and your POP3 username becomes the following:

POP3 username + delimiter you chose in step 1 + POP3 server.

For example, if I was using Netscape, EMCInt, or Pegasus mail, my POP username would become

username#domain.com

If I were to use Eudora, the POP account would become

username#domain.com@wingate.

So, if you are using Eudora, you simply replace the @ symbol with the delimiter you have chosen (e.g. #) and add @wingate to the end. That is all you need to do.

This has benefits over the previous system in that now the user can specify at will which POP3 server to connect to, without requiring a configuration change in WinGate.

## NEW in 1.3.15

The POP3 proxy now handles <user><delim><host>:<port>

so, you can now specify the port number you want WinGate to connect to. This is useful if you want to run a POP3 server on the same machine as WinGate. You can now run the server on a different port, and specify the port in your email client by using the : syntax. For example, if you were running a POP3 server on the WinGate machine on port 8110, and you were using Eudora, then your pop account in Eudora would be

user#localhost:8110@wingate

In this way, you can run a POP3 server on the same machine as WinGate, and still have access to POP3 servers on the Internet as well.

Furthermore, WinGate parses the delimiter from the end of the username, this means you could run through multiple POP3 proxies.

e.g. A pop3 account in Eudora, of

```
user#popserver#wingate2#wingate1@wingate
```

would cause Eudora to connect to wingate, and send the command

```
USER user#popserver#wingate2#wingate1
```

Which would cause wingate to connect to wingate1 and send the command

```
USER user#popserver#wingate2
```

which would cause wingate1 to connect to wingate2 and send the command

```
USER user#popserver
```

which would cause wingate2 to connect to popserver and send the command

```
USER user
```

Thereby connecting through a whole series of POP3 proxies.

## 4.8 RealAudio Proxy

### Step 1

Make sure you are running a RealAudio Proxy in WinGate. The standard port that the RealAudio Player 2.0 uses is 1080. This is normally used by the SOCKS server however, so choose another one (say 1090) for the RealAudio Proxy.

### Step 2

Configure your RealAudio Player 2.0 to use a proxy. Choose **View:Preferences**, then the **Proxy** tab. Select **Use Proxy**, set the Host to gateway, and the port number to that which you chose for the RealAudio Proxy in WinGate.

WinGate supports both TCP only, and UDP modes for receiving audio (Network Tab), The UDP option is recommended.

## 4.9 Mapped Links

You will most likely want to use at the very least 1 of these links (for your email), and probably a couple more as well.

Mapped Links are perhaps the simplest level of gatewaying, however perhaps the most difficult to grasp (still easy though).

If you think of a Mapped Link as a patch cord, then you are effectively patching machines through to remote machines, on specified ports. You can specify a remote host and port number for each individual LAN workstation, or a default remote host and port, which would be used if the machine connecting to the gateway did not have a specific map entry.

## **EMAIL**

Unless you are using the POP3 proxy (recommended), then you will need 2 Mapped Links to get email working. If you are using the POP3 proxy, you only need to set up one mapped link (on port 25) for sending mail.

If you cast your eyes back to the table in Section 3, you will see that there is a port (25) for sending mail, and a port (110) for retrieving mail.

### **Step 1**

Set up a Mapped link on port 25. The default remote host should be set to the SMTP server of your Internet Service Provider also on port 25.

Set up a Mapped link on port 110. The default remote host should be set to the POP3 server of your Internet Service Provider also on port 110.

If different users on your LAN need to connect to different mail hosts, then you need to set up a mapping for their machine, else they will use the default (if enabled).

### **Step 2**

In your email software,

Set your SMTP server (sometimes called your mail relay host) to “wingate”

Set your POP3 server (sometimes called mail server) also to “wingate”

if you are using Eudora, set your POP account to username@wingate (see section 4.7 if you are using the POP3 proxy)

## **NEWS**

### **Step 1**

Set up a Mapped link on port 119. The default remote host should be set to the news server of your Internet Service Provider also on port 119.

### **Step 2**

In your newsreader software,

Set your NNTP server (sometimes called your news server) to “wingate”

## **Notes**

If you do not specify a mapping for a machine, and do not have a default remote host, then a connection from that machine will not be accepted.

For specific mappings, the incoming host must be an IP number. In general, all WinGate knows about a machine connecting to it is its IP number (so this applies for rules for incoming connections as well). So if you want WinGate to do something based on which machine is connecting to it, you must specify an IP number.

The use of the name “gateway” in the previous examples assumes you have set up an entry in the hosts file for each LAN workstation that will be using the gateway. If you have not done this, then in the examples above, you need to substitute the IP address of the LAN adapter of the gateway machine.

## 4.10 Rules

Rules were incorporated in order to protect your LAN from people connecting back through your gateway. There are three ways you can limit access through the gateway.

- Based on connecting host
- Based on requested remote host
- Based on URL (for WWW Proxy Gateway)

You can set default rules for all of these, as well as specifying specific allow / deny rules for particular circumstances.

One standard example of this would be to set up default rule for inward connections to deny all, and then individually allow inward connections from the machines on your LAN, and any specific machine on the Internet you wanted to allow access.

One thing to remember, with the current WinGate release, all comparisons are done on a text basis. This means you could set up a rule which denied connections to hosts containing “.com”. However if a user knew the IP number of a forbidden host, they may try to use the number instead of the hostname, in which case if you hadn’t also set up a rule to deny the IP number as well, the request would be granted.

## 4.11 Dial-on-demand

Dial-on-demand requires that Remote Access Service (RAS - for Windows NT) or Microsoft Dial-up Networking (DUN - for Windows 95) be installed. If the appropriate service is not installed, the Dialler setup page will not be displayed, and no dialler features will be enabled.

The setup for the Dialler page is relatively straightforward. It however assumes that you have had a dial-up profile (phonebook entry) previously configured in RAS / DUN. If you have not used either of these before, it is likely that your phonebook file (from which WinGate gets all its dial-up settings) will be empty, and you will have to run and configure DUN / RAS until you have a profile which allows a complete login without user intervention (save entering a password and pressing the connect button).

All you need to get going then is to select the phonebook entry you wish to use in WinGate, and specify what you want the dialler to do.

## Dialler Options

The Options dialog deserves some more explanation. The Username and Password fields may not be required by all PPP servers / phonebook entries. If you choose Save Password, your password will be encrypted and saved in the registry, otherwise you may need to enter a password here every time you run WinGate.

There were previously (prior to 1.3.08) two advanced options you could specify.

- When to dial
- What to do if there is an existing connection

These have been removed (as they caused more trouble than they were worth)

There is however a list where you can specify locally accessible hosts. If WinGate is asked to connect to a host on this list or a host which contains an entry in this list, it will not initiate the dialler.

The “Use Synchronous RASDial” option is to use a blocking RASDial call instead of the asynchronous call. Don’t worry if you don’t understand what this means.

If you are having trouble with WinGate not completing a connection process, try setting this option and see if the problem persists.

## 4.12 Settings Tab

There is now (WinGate 1.3.11) an option for specifying what sort of socket calls to use in WinGate. The socket calls are the functions calls that WinGate makes to your winsock, which provides TCP/IP connectivity, and allows it all to work.

Various different setups have been problematic with the same setup for WinGate. In previous versions, I have tried to remedy this by changing the way in which WinGate makes sockets calls. No one solution has been found to work for everybody though. So now there is an option. This is how the options relate to previous versions.

**Blocking Send/Recv:** This was introduced in WinGate 1.3.8. API blocking-mode sockets calls are used for sends and receives.

**Non-Blocking Send/Recv:** This was in WinGate versions prior to 1.3.7. API non-blocking-mode sockets calls are used for and receives.

**Readfile / WriteFile API:** This was introduced in WinGate 1.3.10, and was called optimised sends and receives. This only works on Windows NT, and is disabled if you are running Windows95.