

SEATTLE LAB

SLnet RF

The Telnet Server for Windows

SLnet

User's Manual

Published by:

SeattleLab

17310 Red Hill
Suite 200
Irvine, CA 92614

Copyright 1995-2006 by SeattleLab. All rights reserved.

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of SeattleLab, Inc.

Emurl®, SLnet®, SLmail®, SeattleLab®, and Sealab® are registered trademarks of SeattleLab.

Other brands or products are the trademarks or registered trademarks of their respective holders and should be treated as such.

SLnet User's Manual
SLnet 60131

SEATTLELAB SOFTWARE LICENSE

1. GRANT OF LICENSE. This SeattleLab License Agreement (“License”) permits you to use one copy of SLnet (“SOFTWARE”) on any single computer, provided the SOFTWARE is in use on only one computer at any time. If you have multiple licenses for the SOFTWARE, then at any time you may have as many copies of the SOFTWARE in use as you have licenses. The SOFTWARE is “in use” on a computer when it is loaded into the temporary memory (i.e., RAM) or installed into the permanent memory (e.g., hard disk, CD-ROM, or other storage device) of that computer, except that a copy installed on a network server for the sole purpose of distribution to other computers is not “in use.” If the anticipated number of users of the SOFTWARE will exceed the number of applicable Licenses, then you must have a reasonable mechanism or process in place to assure that the number of persons using the SOFTWARE concurrently does not exceed the number of Licenses. If the SOFTWARE is permanently installed on the hard disk or other storage device of a computer (other than the network server), and one person uses that computer more than 80% of the time it is in use, then that person may also use the SOFTWARE on a portable or home computer.

2. COPYRIGHT. The SOFTWARE is owned by SeattleLab or its suppliers and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the SOFTWARE like any other copyrighted material (e.g., a book or musical recording) except that you may either (a) make one copy of the SOFTWARE solely for backup or archival purposes, or (b) transfer the SOFTWARE to a single hard disk provided you keep the original solely for backup or archival purposes. You may not copy the written materials accompanying the SOFTWARE.

3. OTHER RESTRICTIONS. This SeattleLab License Agreement is your proof of license to exercise the rights granted herein and must be retained by you. You may not rent or lease the SOFTWARE, but you may transfer your rights under this SeattleLab Agreement on a permanent basis provided you transfer this License Agreement, the SOFTWARE, and all accompanying written materials and retain no copies, and the recipient agrees to the terms of this Agreement. You may not reverse-engineer, decompile, or disassemble the SOFTWARE. Any transfer of the SOFTWARE must include the most recent update and all prior versions.

LIMITED WARRANTY

SeattleLab warrants that the SOFTWARE will perform in accordance with the accompanying written materials for a period of thirty (30) days from the date of receipt. Any implied warranties on the SOFTWARE are limited to thirty (30) days. Some states do not allow limitations on duration of an implied warranty, so the above limitation may not apply.

CUSTOMER REMEDIES. SeattleLab’s entire liability and your exclusive remedy shall be, at SeattleLab’s option, either (a) return of the price paid or (b) repair or replacement of the SOFTWARE or hardware that does not meet SeattleLab’s Limited Warranty and that is returned to SeattleLab with a copy of your receipt. This Limited Warranty is void if failure of the SOFTWARE has resulted from accident, abuse, or misapplication. Any replacement SOFTWARE will be warranted for the remainder of the original warranty or thirty (30) days, whichever is longer. Neither these remedies nor any product support services offered by SeattleLab are available for this USA-version of the product outside of the United States of America.

NO OTHER WARRANTIES. SeattleLab disclaims all other warranties, either express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the SOFTWARE and the accompanying written materials. This limited warranty gives you specific legal rights. You may have others, which vary from state to state.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. In no event shall SeattleLab or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use this SeattleLab product, even if SeattleLab has been advised of the possibility of such damages. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply.

U.S. GOVERNMENT RESTRICTED RIGHTS

The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs(c)(1) and (2) of the Commercial Computer Software--Restricted Rights at 48 CFR 52.227-19, as applicable. Contractor/manufacturer is SeattleLab.

This agreement is governed by the laws of the State of California. For more information about SeattleLab’s licensing policies, please contact SeattleLab’s Customer Service at (949) 428-9330 or email: info@seattlelab.com

Table Of Contents

Preface	1
Assumptions of this documentation	1
Typographical conventions of this documentation	1
SLnet Defined.....	2
SLnet Features.....	3
SLnet Requirements.....	4
SLnet and Windows Security	6
Authentication types.....	7
SLclient32.....	9
Secure Logon Capabilities	9
Unsecure Logon.....	9
Definitions	10
SLnet and Terminal Server	12
Installing SLnet.....	13
Welcome	13
License Agreement.....	14
Check for New Version	15
Contact Information.....	16
Enable Product	17
Evaluate Product.....	18
Permanently Enable Product	19
Configure SLnet	20
Select Components.....	21
Destination Directory.....	22
Ready to Install	24
Installation Complete	25
Registering SLnet.....	26
Configuring SLnet.....	28
Service Options	29
Client Options	30
Event Logging.....	31
Console Options.....	32
Customizable Prompts.....	36
Include/Exclude	38
Service Controls.....	40
Alternative Ways to Start and Stop SLnet.....	41
User Access Control.....	42
Adding SLnet Users.....	42
Configuring User Accounts	44
Additional Options.....	45
Windows User Accounts	47
Creating Windows NT User Accounts.....	47
Creating Windows 2000 User Accounts	49
Logging on to SLnet.....	53

Automatic Logon	56
Auto Logon against Default User	56
Auto Logon by IP Address	58
Exiting an SLnet Session	60
SLnet Terminal Emulation	61
Multiple Console Windows	64
Select Console type	65
Working with Multiple Console Windows	66
Switching Between Windows	67
Using the Menu	68
Locating and dispatching Popups	69
SLnet Logging	70
Windows Event Viewer	70
SLnet Error Codes.....	71
SLnet Frequently Asked Questions.....	73
SLnet Tips & Troubleshooting.....	75
About.....	80
Hot Key Commands Table	82
Index.....	85

Preface

We are very interested in improving SLnet documentation and welcome all criticisms and suggestions for improvement. This email address should only be used for issues relating to documentation, not for general technical questions. Customer Care (technical support) email should be addressed to support@seattlelab.com

SLnet Documentation

SeattleLab
17310 Red Hill
Suite 200
Irvine, CA 92614
Phone: (949) 428-9330
Fax: (949) 428-9331
Customer Care: (949) 428-9330
Email: support@seattlelab.com

Assumptions of this documentation

This documentation assumes that you have some familiarity with the basic Windows user interface: that you know, for example, how to select an item in a list box, what a program group is, and how to double-click. We have tried not to assume further knowledge—for example, knowledge of the Telnet protocol.

To use much of the material in this documentation, you must have administrative privileges on your Windows system—that is, you must have logged on under the user name of Administrator or be a member of the Administrator’s user group. It is assumed that if you have these privileges you also have a basic acquaintance with the essentials of Windows administration. Windows administration fundamentals are covered in the Windows Resource Guide.

Typographical conventions of this documentation

Italic type is used for items and text to select on your computer screen.

For example: To stop the service, click the *Stop* button. To restart the service, click the *Start* button.

Text in brackets < > is meant to be replaced with the (literal) information that is shown.

For example: <field1> would be replaced with the actual information that is included in field1.

Text you will be instructed to type will appear in the `Courier` font.

For example: Type: `Install`.

Important notices will be displayed entirely in **bold** text.

For example: **Note: Changes made in the Service tab will not take effect unless you click *Apply* or stop and restart the SLnet service.**

SLnet Defined

SLnet is a full featured Telnet service for Windows NT/2000/2003. If you want to run VT emulation on wireless hand-held devices connecting to a Windows server, you need SLnet.

Telnet is the standard for character-based network communications. It is a part of TCP/IP, the networking protocol of the Internet. Computer users on local area networks (LANs) and on the Internet use Telnet to connect to a remote host and execute programs on the host.

Microsoft Windows networking supports TCP/IP, and has a built-in Telnet client, but does not have a built-in Telnet service. This means that from Windows, you can Telnet *out* to run a program on other systems. But from those other systems, you cannot Telnet *in* to Windows to run programs there.

SLnet provides this functionality. When SLnet is installed on your system, you can Telnet from remote systems and run programs on your Windows system.

There are many reasons why users want to Telnet into a Windows system. Among the most popular are:

To remotely administer a Windows system

To take care of a Windows system—for example, to start and stop services—you must be at the console of that system. SLnet provides the remote user the same power as the user at a console, so a system administrator whose office is several floors from the server does not have to physically go to the server to restart a service. Indeed, over a RAS connection or over the Internet, an administrator or a developer can support a Windows system that may be physically located thousands of miles away.

To support legacy applications

With the phenomenal popularity of Windows, developers are moving applications from other platforms to run on Windows. These older platforms almost always use Telnet to provide their interface to users. They need SLnet to make Windows work more like the legacy platforms.

To run multiple users on a single Windows system

Windows is a multi-tasking operating system, completely capable of handling multiple users running different jobs at the same time. But there is no way for multiple users to log onto an NT system. Only one user can be using the system console at a time. SLnet allows multiple users to be logged on at once. One possible use of this functionality is for multiple users to run DOS applications. With SLnet, Windows is the ultimate multi-user DOS system.

To run jobs where they run most efficiently

As Wide-Area Networks and business uses of the Internet become more popular, many users are discovering that the client/server model is not the most efficient architecture for every situation. If a remote client needs to run an application that uses large amounts of data, it may be very inefficient to drag all that data over a slow link such as a telephone wire or the Internet. It can be orders of magnitude faster to run the job on the server where the data resides than to run it on the client where access to the data is slow. SLnet allows remote users to run their jobs on the server rather than on the client when it makes sense to do so.

SLnet Features

Access to Windows's Common Command Shell

Users who Telnet into SLnet can have full access to Windows's Common Command Shell. This command shell is the equivalent of the DOS command prompt, with many additional features. Users can also run OS/2 or POSIX applications from the command shell. (SLnet is completely configurable by the administrator, who can choose to disable access to the Command Shell on a user-by-user basis.)

Direct Access to Custom Applications

SLnet offers you the power to define default custom applications for users as soon as they log in.

Security

Users run in their own appropriate security context, with all their own permissions in place, just as if they had logged in from the main console. This completely preserves all Windows's C2 security features.

Secure Logon

SLclient32 encrypts user IDs and passwords for secure logon over the Internet. With SLnet's AuthTypes registry value you can specify the level of encryption (and the encryption protocol) for the entire system, or on a user-by-user basis.

Terminal emulation

SLnet provides ANSI terminal sequences (compatible with the VT-100 and VT-220) for cursor positioning and formatted screen handling.

Logging

SLnet leaves a record in the Windows event log every time a user connects or disconnects, along with a record of what process the user executed and the resulting code from that process.

Integration with SeattleLab Terminal Server for Terminal Support

By connecting SLnet to SeattleLab's Terminal Server, you can extend the complete Windows functionality to asynchronous terminals and other serial devices. On a system with both SLnet and Terminal Server installed, a user from a terminal connected to the Windows NT box can run DOS commands or any character-based application.

Remote Administration of SLnet Telnet Server

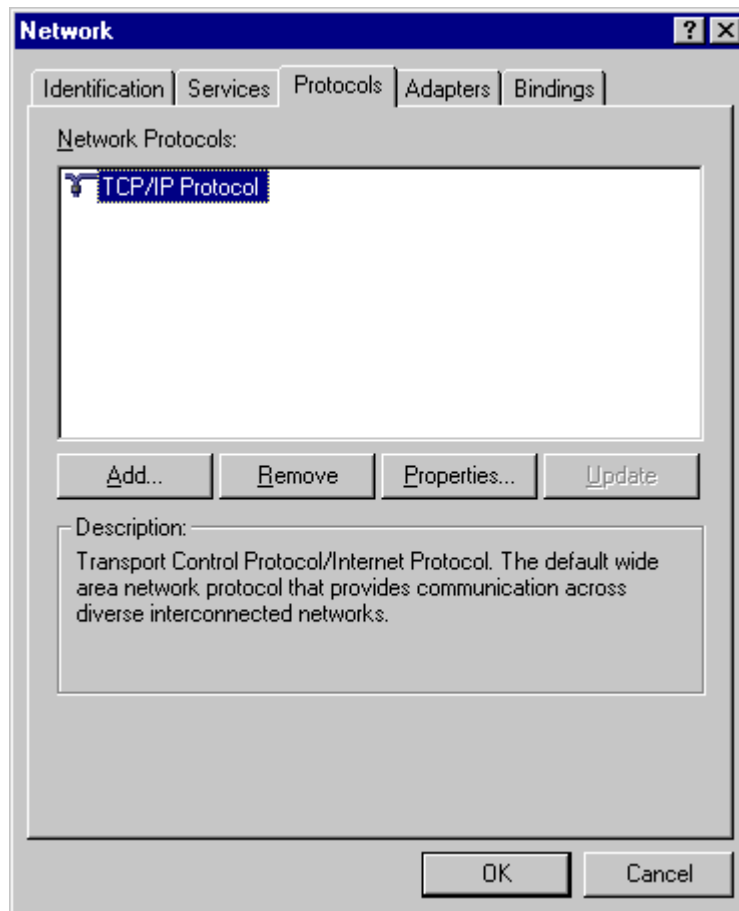
SLnet can be remotely administered by using the Run command.

SLnet Requirements

SLnet requires Windows NT (4.0 SP3+) or Windows 2000/2003. If you need to support Windows NT 3.51 or DEC Alpha CPU, please call SeattleLab.

Note: If you are using Service Pack 4 of NT 3.51, you must disable the Create Private Desktop option for SLnet to function. See the section on [Service Options](#) for details.

The TCP/IP protocol must be installed on your system for SLnet to function properly. This protocol is included with the Windows operating system but is not always installed on all systems. You can check if TCP/IP is installed on your system by choosing *Network* from the *Control Panel* and then selecting the *Protocols* tab as illustrated below. If TCP/IP is installed on your system it should show up in the list of Installed Network Software.



Network Settings dialog box in the Windows NT Control Panel

The SLnet service itself uses only a minimal amount of system resources, requiring four megabytes of disk space in your root system directory (the directory where Windows is installed on your system) and 2 megabytes of system RAM for the initial connection.

Each individual SLnet user will require two additional megabytes of system memory. The additional memory requirements of specific user applications, which may consume any amount of memory, should also be taken into account when tabulating system requirements.

SLnet–SLnet Requirements

SLnet is a CPU-intensive service. Monitoring virtual consoles to export changes across the Telnet interface consumes more and more system resources as you add more and more users.

The two crucial variables, then, in determining the total number of SLnet users your Windows system can support will be based upon a combination of your system CPU speed and memory limitations.

SLnet and Windows Security

SLnet abides by all Windows security features. It does not alter nor change any security settings or privilege levels set by normal Windows mechanisms.

When a Telnet user logs into your system via SLnet, he or she must be a valid user that you or another administrator created using the Windows User Manager or Active Directory Manager. SLnet sets that user's process to run in the security context appropriate to that user. The user then has exactly the same privileges he or she would have on your machine if they sat down in front of it and logged in on the console by pressing the *CTRL ALT DELETE* keys.

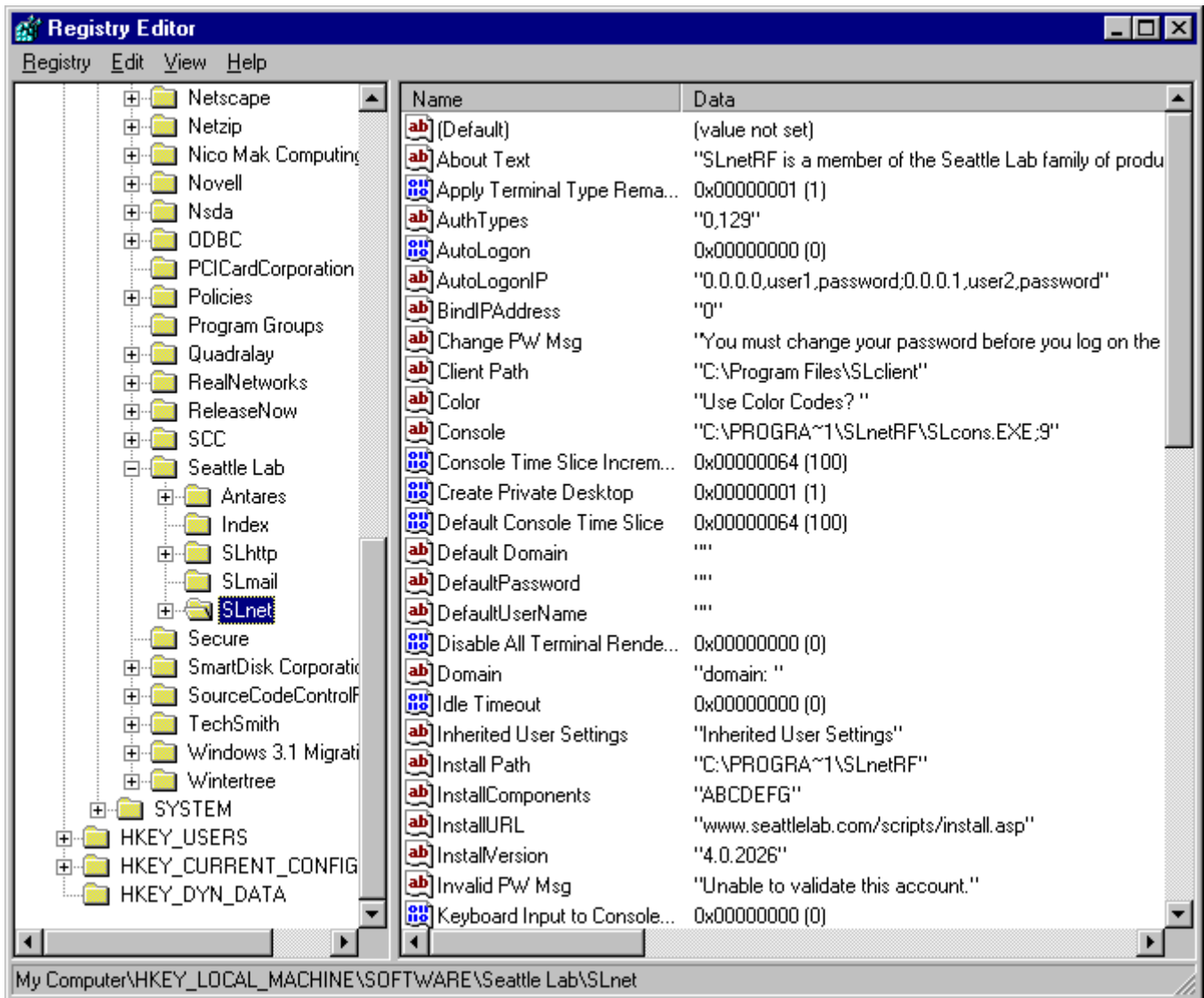
Windows allows you to grant or withhold privileges on a user-by-user basis to every object on the system. Using Windows Explorer, for example, you may choose which individual users can have access to any given file you create. SLnet abides by these exact same access restrictions. It does not and cannot alter any privileges set by Windows.

Authentication types

SLnet allows you to encrypt the user name and password in the data stream with the **AuthTypes** registry value.

You will need to modify this registry value in the root of the SLnet registry (*HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet*).

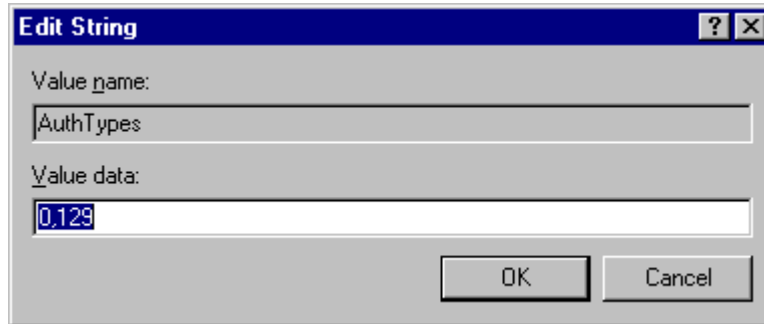
Open the Registry Editor through the Start Menu (*Start > Run* and then entering *regedit*).



Registry Editor

In the right frame of the registry editor, right-click on the *AuthTypes* string value and select *Modify* from the popup menu.

The Edit String dialog box will open allowing you to modify the Value data.



Edit String dialog box

By default the key has 0,129 (IP Address, decimal equivalent for the auth type required), which allows all users to use the SeattleLab encryption or fall back to clear text.

Enter <IP address, decimal equivalent for the authorization type; IP address,...> and then click *OK* to update the string.

The string pairs must be separated by a semi colon. For example: 0,129;127.0.0.1,128;129.168.0.1,65

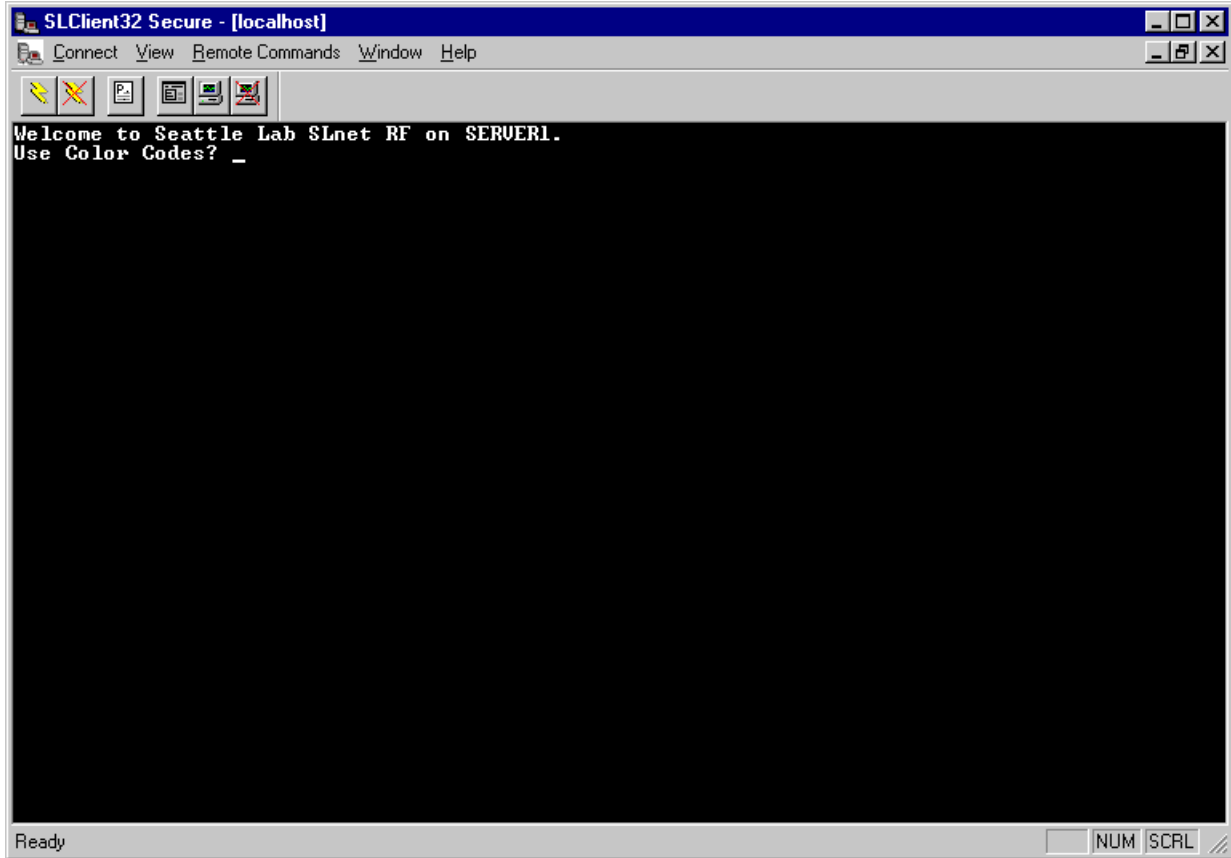
The authentication types are:

- Null (no encryption)—bit 1, decimal 1
- Microsoft NTLM—bit 7, decimal 64
- SeattleLab proprietary—bit 8, decimal 128

Note: Multiple encryption types can be specified by combining the bits. For example for Null (clear text) and SeattleLab encryption the decimal equivalent would be 129. All single encryption types and pairs are possible except for SeattleLab and NTLM.

SLclient32

SLclient is a fully featured, secure 32-bit Telnet client for Windows only. It is available from the Start menu by clicking *Start > Programs > SL Products > SLnet > SLclient*.



SLclient

Secure Logon Capabilities

SLclient32 provides secure logon for SLnet users. When a user enters his or her user ID and password, SLclient32 encrypts them and sends them to SLnet where they are decrypted. This allows complete security when sending IDs and passwords over the Internet. While anyone with the right equipment can still intercept logon information when sent over the Internet, they will not be able to read it because the information is encrypted.

In a standard dialup Internet connection using Point-to-Point Protocol (PPP), the data is encrypted with a Point-to-Point Tunneling Protocol (PPTP) wrapper. This encryption is a 40-bit international or standard process. Together SLnet and SLclient32 use a stronger encryption when sending logon data; the encryption is used only for authentication, however, not general data security.

Note: SLclient32 only works with SLnet. It will not work with any other product.

Unsecure Logon

Please note that users with the 16-bit SLclient installed on 3.1/9x machines (Windows 3.1, Windows 95 or Windows 98 operating systems) should beware that their IDs and passwords are not encrypted.

The only security risk you open to your system by installing SLnet is that when Telnet users who are **not** using **SLclient32** send passwords over their TCP/IP connection to your system, a nefarious party with a TCP/IP protocol analyzer could watch these passwords go by and use the passwords to gain unauthorized access to your system.

In most installations this is an unlikely scenario. For example, if all your users log on from your Local Area Network (LAN), evildoers would have to come onto your premises and physically attach a protocol analyzer to your Ethernet cabling in order to undermine your security.

If, however, you allow Internet connections to your network, you should be aware that malicious individuals at Internet sites between your users and you could steal your passwords and break into your system. This is the same security risk all Internet access allows, and system administrators in this position would be advised to consult the many excellent books on Internet firewalls in order to secure their sites against unauthorized access.

Definitions

Encryption is the process of converting a message into a ciphertext (encrypted message) using a key. The message appears jumbled to anyone not able to decrypt it. The intended recipient then uses the same encryption key to decrypt the message and read it.

Decryption is the process of decoding an encrypted message using the encryption key.

SLnet and Terminal Server

SLnet has a companion product named Terminal Server that extends all the functionality of SLnet to asynchronous terminals. When you add Terminal Server to any Windows system on your network, terminals and other serial devices attached to that system can use SLnet to run character-based applications on your Windows system.

By combining SLnet and Terminal Server, your users can run DOS, OS/2, POSIX, and Win32 console applications from dumb terminals. From their terminals, they can have complete control of the Windows command shell, just as if they were sitting at the console of the machine.

SeattleLab Terminal Server is Web-based software that allows any Windows computer to have the same functionality as a stand-alone hardware terminal server. One of the primary benefits of SeattleLab Terminal Server is its ability to support legacy hardware. With the world converting to Windows, which lacks native support for serial devices, SeattleLab Terminal Server provides the perfect solution.

SeattleLab Terminal Server 3.0 is the successor to SLink 2.2, SeattleLab's terminal service for Windows. It extends the capabilities well beyond what SLink provided by allowing serial devices to connect to the LAN, WAN or Internet for connectivity to a Telnet server via TCP/IP. This extends the usefulness of serial devices by connecting them to an IP network.

In addition to dumb terminals, SeattleLab Terminal Server supports any device with a serial interface, including cash registers, scanners, bar-code readers, and scientific or medical equipment.

The latest version of SeattleLab Terminal Server is script driven, using SLscript as an engine. Because of its speed, versatility and design, this scripting language can be used in situations where the only other solutions are costly C++ or Visual Basic programming solutions.

SeattleLab Terminal Server is available from SeattleLab's website: www.seattlelab.com.

Installing SLnet

Before installing SLnet, you must first log on as a member of the Administrator’s user group.

To start the SLnet installation process:

If you have purchased the CD-ROM, place the disk in the CD-ROM drive, click the *Start* button, then *Run*, select the drive that contains the media, and execute the file `SETUP.EXE`.

If you downloaded the SLnet self-extracting file from SeattleLab’s website, choose the *Run* command from either the task bar or Program Manager, choose the directory in which the self-extracting file was saved, and execute the file.

Welcome

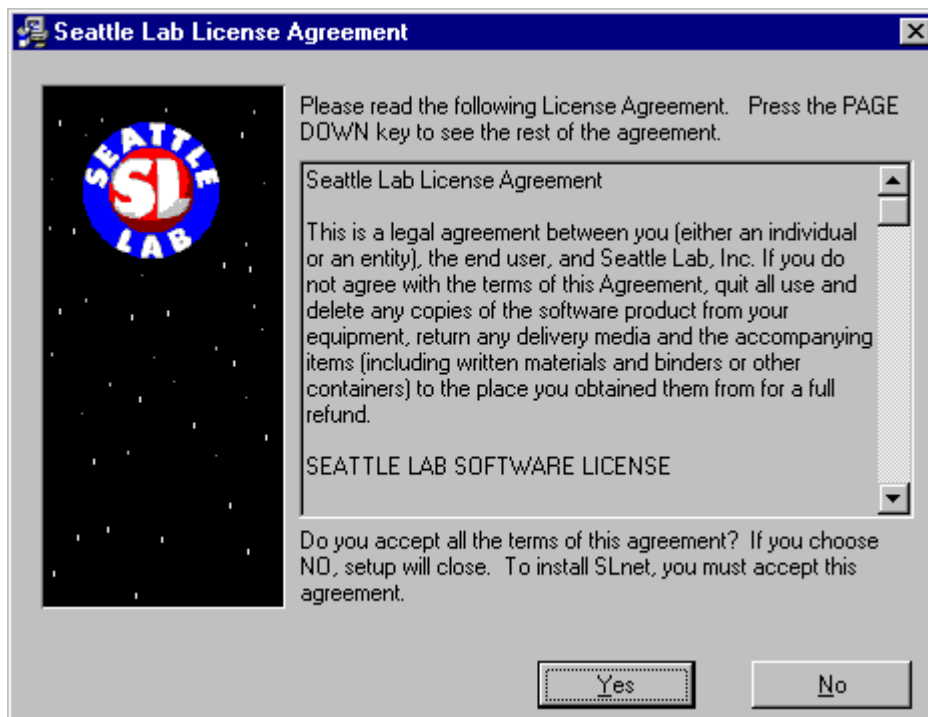
Setup will display a welcome screen identifying itself as the SLnet Installation and asking you to click *Next* to continue.



SLnet Installer Welcome Screen

The License Agreement screen will appear next.

License Agreement



SLnet License Agreement Screen

After reading the complete SLnet License Agreement, click Yes to continue the installation process.

Check for New Version

If you are connected to the Internet, the SLnet installation wizard will ask you whether you want to check with SeattleLab to ensure that you are installing the most current version of SLnet. (It is best to ensure that you are installing the most current version.)

Note: If you are not connected to the Internet, make sure that you leave the box unchecked. Otherwise, the installation process will trigger your dialup connection. If you have no way of connecting to the Internet, the installation will freeze, and you will have to quit installation and start over.

If the check box is greyed, it may indicate that the TCP/IP settings for DNS are incorrect.



Check for New Version Screen

If you are not installing the current version of SLnet, the installation wizard will ask you whether you would like to launch your web browser to the SLnet upgrade page.

Click *Next* to continue.

Contact Information

Unless you are downloading from SeattleLab's website and have already provided contact information, the installation wizard will ask you for it now. This information is not released to anybody outside of SeattleLab; it is used to inform you of updates to SLnet.

The image shows a screenshot of a Windows-style dialog box titled "SLnet Installation". On the left side, there is a vertical black bar with a starry background and a circular logo in the center. The logo has "SEATTLE" at the top, "SL" in the middle, and "LAB" at the bottom. To the right of the logo, the text "Contact Information" is displayed in a large, italicized font. Below this, a paragraph of text reads: "Please enter the following contact information. This information will be kept strictly confidential by Seattle Lab." There are three text input fields: "Name:" with the placeholder text "Your Full Name", "Email:" with the placeholder text "Email Address", and "Phone:" with the placeholder text "Phone Number including the country code". At the bottom right of the dialog box, there are two buttons: "Next >" and "Cancel".

Contact Information Screen

After filling in the text boxes, click *Next* to continue.

Enable Product

The wizard will next ask you whether you prefer to register SLnet or evaluate the product. If you already have your serial number and authorization code, click the *Register* button. Then click the *Enable* button.

Alternatively, you may register later from the Registration tab of the SLnet Configuration Control Panel. (See the section on [Registering SLnet](#) later in this documentation for details.)



Enable SLnet Screen

If you wish to visit SeattleLab’s online store at this time, check that option. The installation wizard will launch your web browser and direct it to the store. Should you choose to pay for the product at this time, you can pay on-line and the serial number and authorization code will be emailed to you within two business days.

Evaluate Product

If you wish to evaluate the product, check the *Evaluate* button and the following screen will open.

Free technical assistance is available during your 14-day evaluation period.




Evaluate Product Screen

Permanently Enable Product

Registration requires you to enter two values. The first is your *Serial Number*. You receive your serial number from SeattleLab when you purchase SLnet. In most cases, it is the same as your invoice number. You should record your serial number in a safe place because you will need it to receive technical support. The serial number consists of numerical digits.

The second number required is your *Authorization Code*. This code is provided to you by SeattleLab when you pay for your copy of SLnet. The authorization code is tied to your serial number and will not work with any other copy of SLnet. The authorization code is a string of up to eight hexadecimal digits. Hexadecimal digits include the ten decimal digits and the letters A-F. If your code includes these letters, you may enter them in either upper or lower case. You should record your authorization code with your serial number. You should not need it again unless you need to completely reinstall SLnet.



Seattle Lab Telnet Server Authorization

Permanently Enable Product

Please enter your registration information. Be sure to keep your Serial Number and Authorization Code in a safe place.

Serial Number: -1

Authorization Code: -1

Evaluation Status: New Install

I am still continuing my evaluation.

< Back Register > Cancel

Permanently Enable Product Screen

Click *Register* to continue installation.

See [Registering SLnet](#) for information on how to register the product after it has already been installed.

Configure SLnet

Whether you have chosen to register the product or evaluate it, the installation wizard will now provide a configuration screen that it has automatically filled in.

It is best not to change any of the information within the dialog boxes.

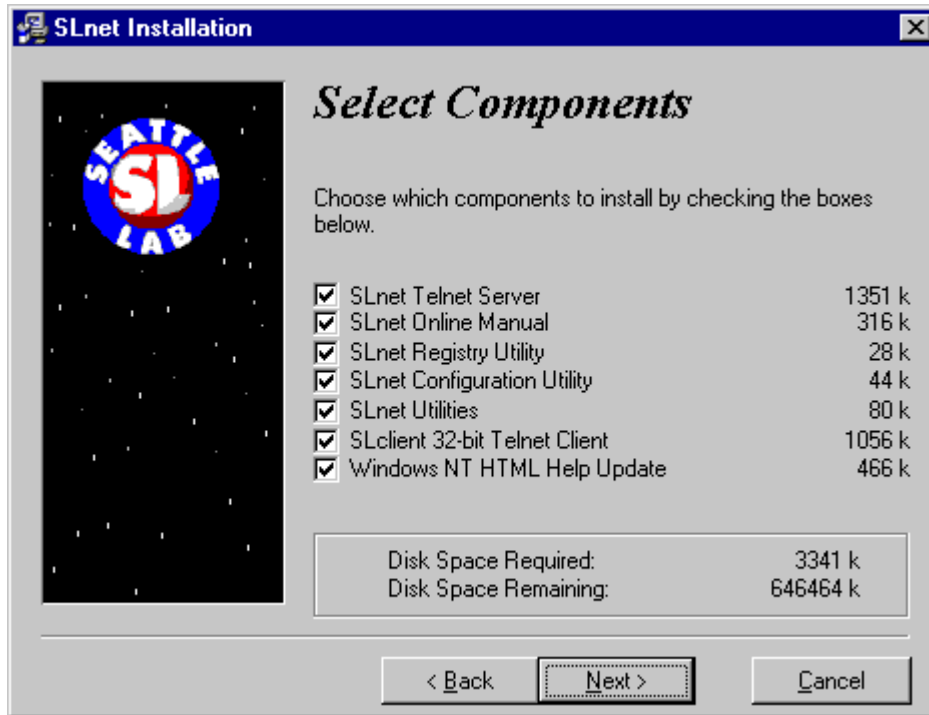


Configure SLnet Screen

Click *Next* to continue to the Select Components screen.

Select Components

The Select Components screen will allow you to choose which of the SLnet components you wish to install. By default, all components are installed. Click *Next* to continue.

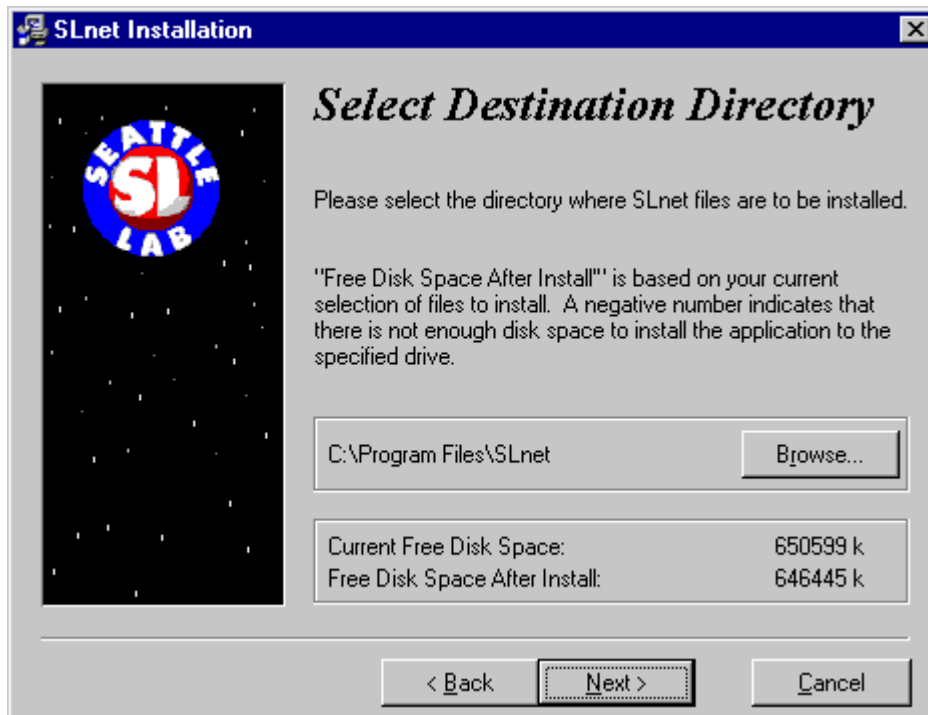


Select Components Screen

Destination Directory

If a previous installation has **not** been detected, the Select Destination Directory dialog box will open, asking you to select the directory where you want the **SLnet** files to be installed. To install the application to the default location, click *Next*.

Alternatively, you may click *Browse* to select a different destination directory. After highlighting the alternative destination directory, click *Next*.



Select Destination Directory for SLnet Files Screen

A second Select Destination Directory dialog box will open next, asking you to select the directory where you want the **SLclient32** files to be installed. Click *Next* to install the application to the default location.

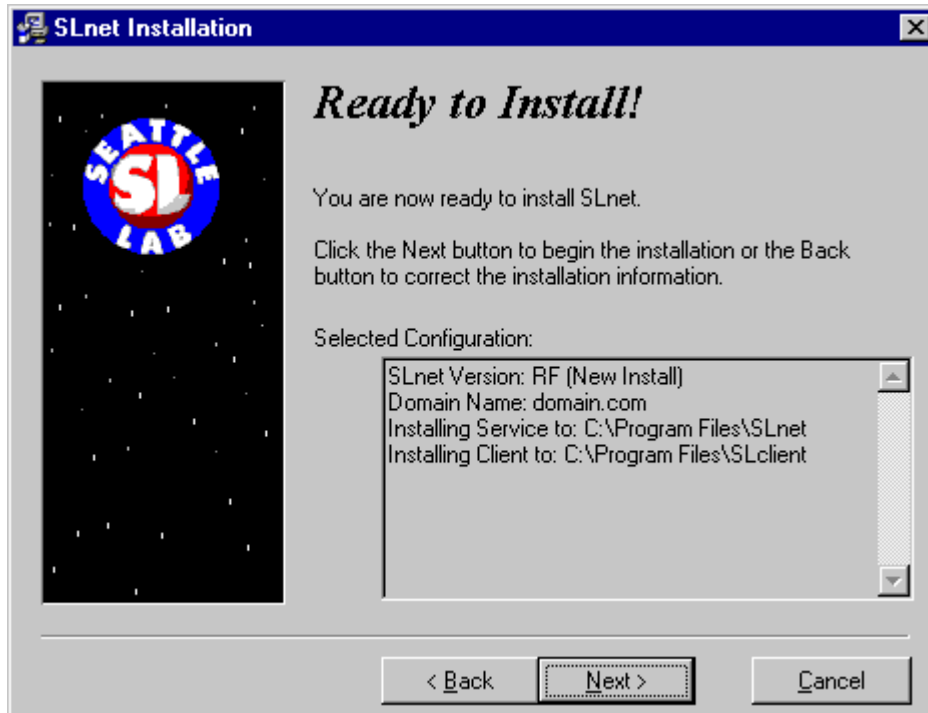


SLclient Destination Directory

Alternatively, you may click *Browse* to select a different destination directory. Click *Next* after highlighting the alternative destination directory.

Ready to Install

The Ready to Install screen provides information about the selected configuration. If you wish to change any of the information, click the *Back* button.



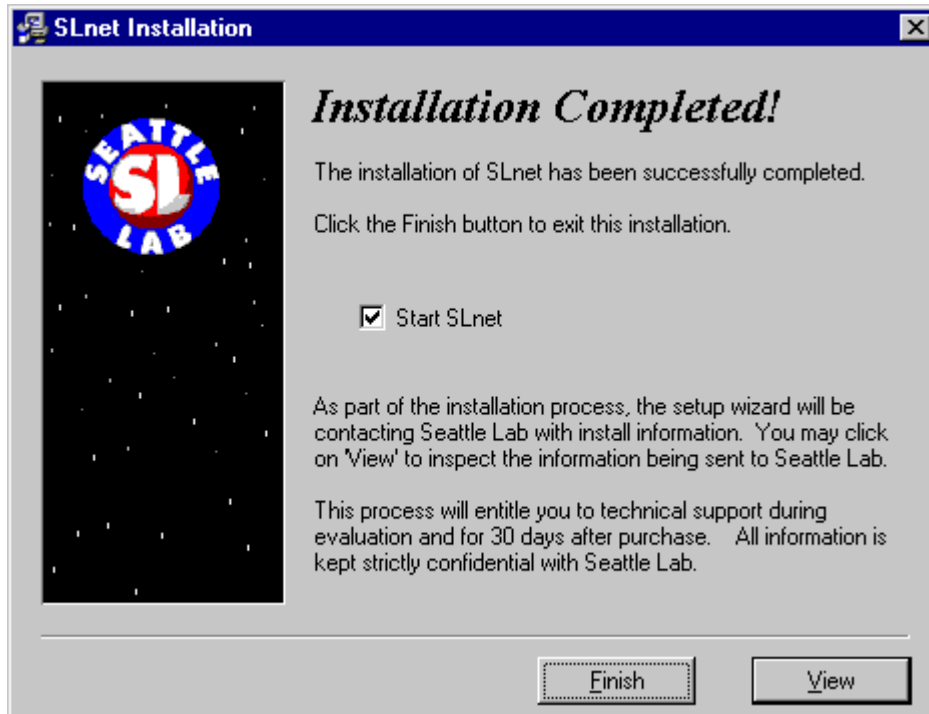
Ready to Install Screen

To complete installation, click the *Next* button. A progress indicator (not illustrated) will open.

If the installer detects that the SeattleLab Telnet Service is already running, it will ask you if you would like to stop the service and perform an upgrade. You must stop the service to perform the upgrade. Click *Yes* to continue.

Installation Complete

The following screen will open upon completion of installation. To view the information being provided to SeattleLab, click the *View* button. You have the option of starting SLnet at this time by checking the *Start SLnet* check box. Click the *Finish* button.



Installation Completed Screen

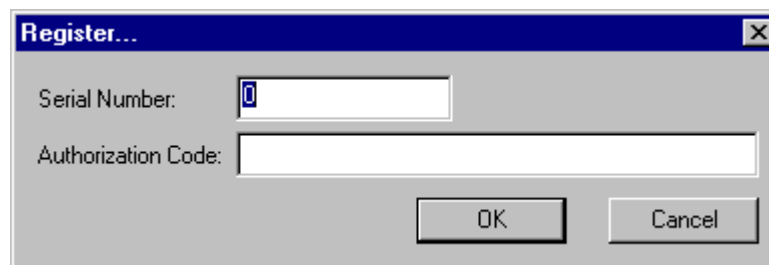
Registering SLnet

SLnet must be registered or it will cease to function two weeks after installation. If you did not register your copy of SLnet during the initial installation process, you may register it later from the Registration tab of the SLnet Configuration Control Panel, illustrated below.



Registration Tab

Click the *Register* button. The Register dialog box will open.



Registration dialog box

Enter the *Serial Number* and *Authorization Code* exactly as they were provided to you by SeattleLab and click *OK*. The number of users it is licensed for will now be displayed in the Registration Tab under the dialog box.

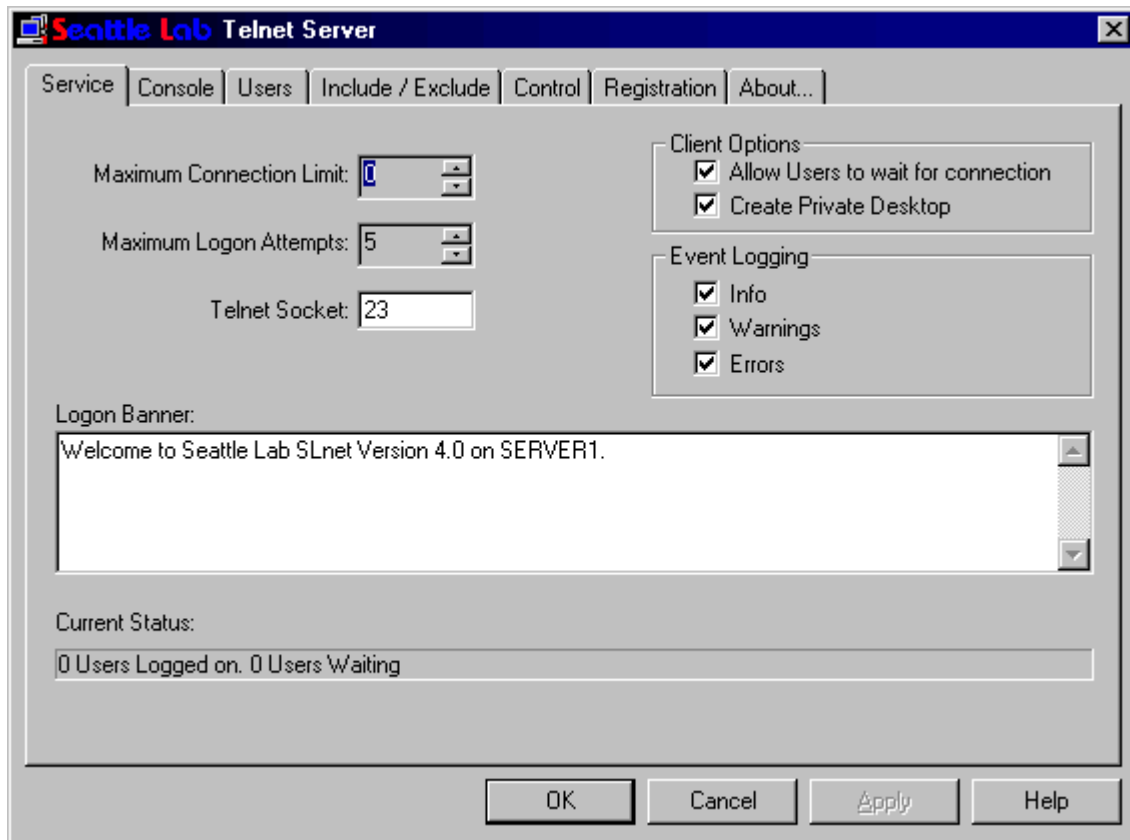
Note: Please keep a hard copy of the Serial Number and Authorization Code in a safe place. Should you ever need to reinstall SLnet, you will need to re-enter it. Once your Authorization code is entered, it will not be displayed again.

If you are using a demonstration copy of SLnet and wish to license it for permanent use, or if you wish to relicense your copy of SLnet for a greater number of users, contact SeattleLab and pay the appropriate fee to receive a new authorization code.

Configuring SLnet

All SLnet configuration and maintenance is performed centrally from the SLnet Configuration Control Panel applet. This applet may be accessed either directly from within the system control panel (simply double-click the *SLnet Configuration* icon to open the application) or from within the SLnet program group (*Start > Programs > SL Products > SLnet > SLnet Configuration*).

Note: In order for you to configure SLnet, you must be logged on with Administrator privileges.



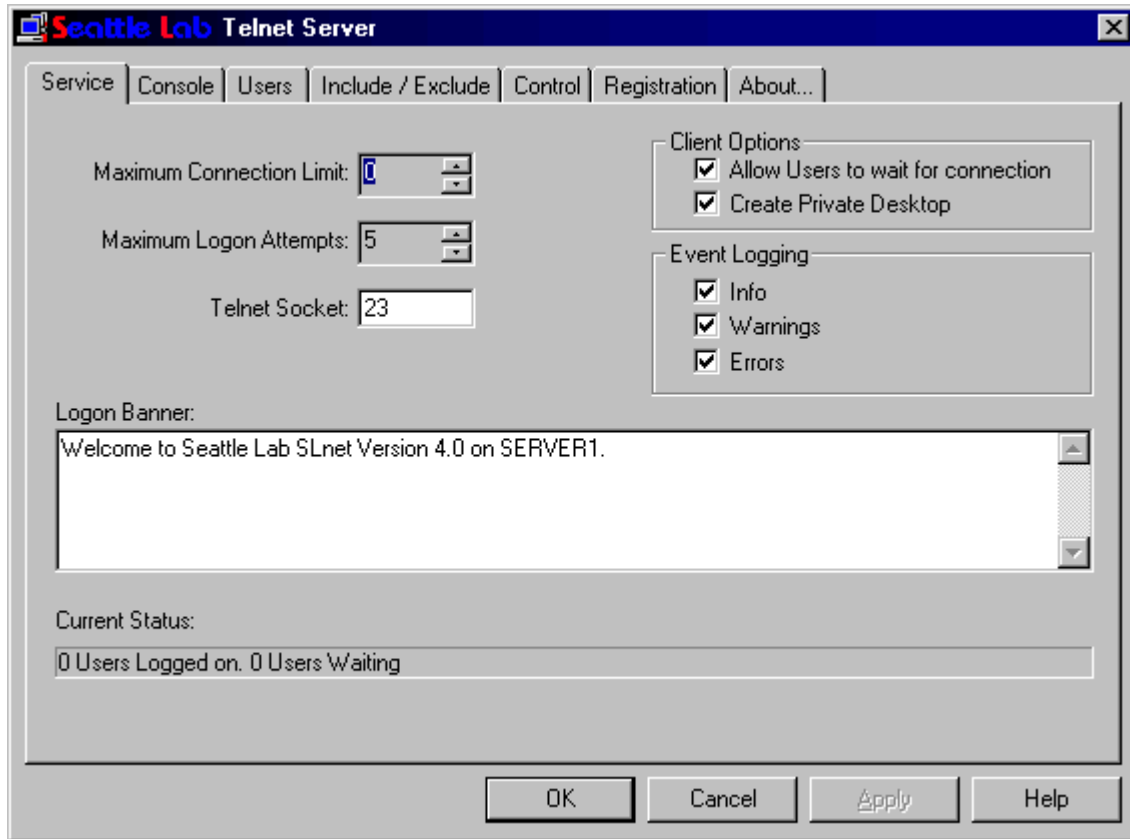
SLnet Control Panel Applet

The SLnet Configuration Control Panel applet offers complete control of SLnet operation through seven different tabs: Service, Console, Users, Include/Exclude, Control, Registration, and About. The following is a brief description of the function of each tab. In-depth documentation of the proper operation of each tab will follow later in this manual.

- The **Service** tab allows you to control service access parameters.
- The **Console** tab is used to optimize console application performance.
- The **Users** tab allows you to specify valid user accounts.
- The **Include/Exclude** tab allows you to specify ranges of IP addresses that may and may not use SLnet to connect to your system.
- The **Control** tab allows you to start, stop or pause the SLnet service.
- The **Registration** tab is used for registering or relicensing SLnet.
- Finally, the **About** tab provides information about SLnet and contacting SeattleLab.

Service Options

From the Service tab of the SLnet Configuration Control Panel applet, you can set parameters that govern the operation of the SLnet service.



Service Tab

Maximum Connections

The Maximum Connection Limit field allows you to configure how many SLnet connections are allowed at any one time. The default setting for this field is 0 (zero); this permits as many machines to connect as you are licensed for. This also allows **unlimited** maximum connections. Should you wish to change the number of machines allowed to log on, simply change the value in this field.

Maximum Logon Attempts

The Maximum Logon Attempts field allows you to set the number of times the system will allow you to enter incorrect user IDs or passwords before returning you to the Telnet prompt. The default is 5. If you leave this setting at 5, a user has five chances to enter a correct user ID and password before being returned to the Telnet prompt. It should be reiterated that SLnet will never take precedence over Windows's own security. Therefore, if you have activated NT's account lockout policy and wish to avoid locking out forgetful SLnet users from your system, ensure that the value for SLnet's *Maximum Logon Attempts* is never set higher than the NT value.

Telnet Socket

The Telnet Socket (port) field allows you to set the TCP/IP socket on your machine that the SLnet service monitors for Telnet client connections. The default is socket 23. This is the Telnet socket. Telnet services and clients use

socket 23 by default. You should only need to modify this socket if you already have another Telnet service using socket 23. If you have Telnet clients that are capable of being configured to use alternate sockets, you could configure some clients to use socket 23 to reach the other Telnet server and some clients to use an alternate socket to reach SLnet.

Note that some sockets are reserved for other services. Internet RFC 1060 lists the assigned sockets. Excerpts from this RFC are included on your Windows system in the following file:

```
%systemroot%\system32\drivers\etc\services
```

Here *%systemroot%* indicates the directory in which Windows is installed.

It is important, however, to emphasize that this is only a general guide to finding free sockets, and only experimentation will prove that an alternate socket you choose is not in use by another service on your network.

Note: Changes you make to the Telnet Socket in the Service Options do not take effect until you restart the SLnet service.

Logon Banner

The Logon Banner is displayed when users first connect to SLnet. The banner shown is an example that displays a brief welcome message and the software version followed by the machine name. The default banner is created during the install process, but can be easily altered once installation is completed. The multi-line banner can be as many as 2,048 characters long.

Current Status

At the very bottom of this tab, SLnet's current status will appear listing both the number of current users logged on and the number of users waiting for a session to open.

Client Options

Allow Users to Wait for Connection

Unless the Allow Users to Wait for Connection check box is checked, all additional users over the limit prescribed in the *Maximum Connection Limit* field who attempt to log on will see the message: "There are currently too many Telnet sessions. Please try again later." They will then be returned to the Telnet prompt.

If the *Allow Users to wait for connection* check box **is** checked, however, users will not be immediately returned to the Telnet prompt. Instead, they will be asked whether they would like to wait until a Telnet session becomes available. This option is enabled by default.

Create Private Desktop

In its normal operation, SLnet creates a private desktop for each individual user that logs on. A private desktop is essentially an invisible MS-DOS console that does not appear in the main system window nor the task list. For a user logged onto the main Windows system console, the work of SLnet users is virtually unnoticeable. For the SLnet user, work appears to take place on an MS-DOS console. Because Windows is a multi-tasking operating system, it can theoretically support an infinite number of these connections.

If the Create Private Desktop function is disabled, a command window will open on the desktop of the computer running SLnet each time a user logs onto SLnet.

The Create Private Desktop function is enabled by default.

Event Logging

SLnet uses the Windows Application Log in the Event Viewer to keep a record of everything it does by entering messages as events occur. By default, SLnet logs three types of events: Info, Warnings and Errors. Info messages (such as which user logged on) are the least serious and Errors (such as your demo copy of SLnet has expired) are the most serious. Uncheck any type of event that you do not want entered in the Application Log. (See the section on [SLnet Logging](#) in this documentation for more information.)

Note: While the Event Viewer is used to display these messages, the database it is stored in is the Windows Event Log.

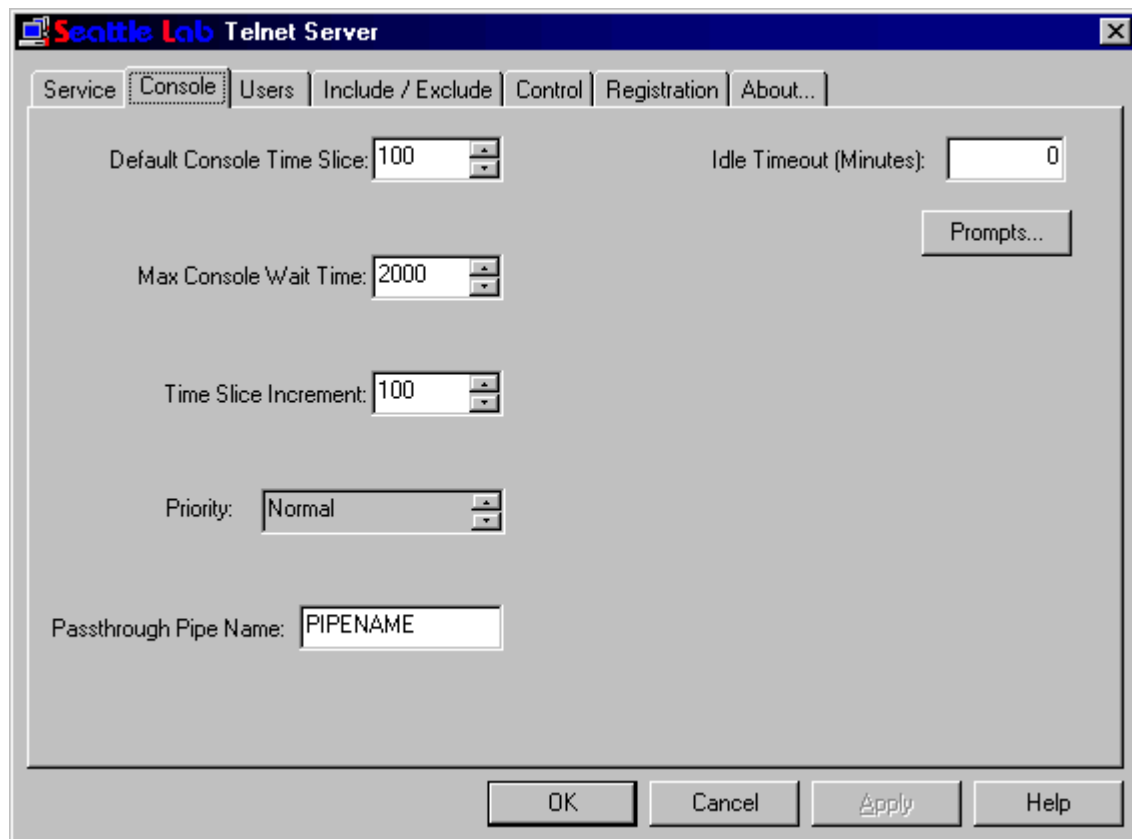
Console Options

The Console tab of the SLnet Configuration Control Panel allows you to set internal parameters used in the virtual consoles created by the SLnet service. For most use, you should not need to adjust these parameters. Doing so will change performance, latency, and throughput of SLnet and your Windows system as a whole.

When an SLnet user is running a task on a virtual console on the Windows system, the SLnet service monitors the screen buffer of that console for changes, so that those changes may be sent over Telnet to update the screen of the client.

If SLnet monitors these changes too infrequently, the user will experience sluggish and choppy performance. If SLnet monitors those changes too frequently, it will consume too much of the system's resources and reduce the performance of the SLnet user's task and all other tasks running on the system.

The algorithm used by SLnet to attempt to optimize these conflicting constraints is to begin polling relatively frequently, and to poll progressively less frequently when no screen activity is detected. The parameters used by this algorithm are those exposed in the *Console* tab of the SLnet Configuration Control Panel.



Console Options Tab

Console Time Slice, Wait Time and Increment

SLnet begins by polling at the *Default Console Time Slice*. If no screen activity is observed, it increases the polling period by the value set in the *Time Slice Increment* field until it reaches *Max Console Wait Time*. Whenever screen activity is observed, SLnet returns the polling period to the *Default Console Time Slice*. All times are in milliseconds. The defaults are those pictured above.

These parameters should provide acceptable responsiveness to SLnet users as well as an acceptable load on the system as a whole. If you wish to experiment, SLnet may be made more responsive by decreasing the Default Console Time Slice, the Max Console Wait Time, and the Time Slice Increment. (Decreasing the Max Console Wait Time gives you the lowest benefit at the highest cost, since it controls how many resources SLnet is consuming when the user is not doing anything.)

You can decrease the amount of system resources SLnet consumes by increasing the Default Console Time Slice, the Max Console Wait Time, and the Time Slice Increment. Note that if you excessively increase the Max Console Wait Time, users may be led to believe that their process is hung.

SLnet will disregard values outside a range that allows it to function. The range is set as follows:

Parameter	Minimum Value	Maximum Value
Default console time slice	100	2000
Max console wait time	2000	60000
Time slice increment	100	1000

Priority

The Priority option allows system administrators to customize response time for users connected to the SLnet service. In most cases, the default setting of “Normal” is sufficient.

You should be very cautious about changing the priority of your processes. It is possible to severely impact the performance of other processes—including the Windows desktop—if the settings are not properly set.

Pipe Name

The Passthrough Pipe Name field allows you to customize the environment variable name of the main pipe used by SLnet. The default name for this field is *pipename*.

SLnet works with unmodified off-the-shelf character applications by reading the console output buffer at periodic intervals, noting any changes made in the buffer, translating those changes into a character stream, and directing that character stream to the Telnet socket. The character stream contains ANSI escape sequences for cursor control to correctly format the screen. This design allows SLnet to support the widest possible range of third-party products, without requiring customers to make any modification to their software.

For customers who have the ability to modify their software, however, or who are writing new software, SLnet provides a more direct output connection.

This connection takes the form of a Named Pipe. There are two advantages to using direct output to the named pipe:

1. It is more efficient, allowing given hardware to support more processes in addition to allowing for each process to complete more quickly.
2. It allows support for non-ANSI terminal emulations.

The SLnet Named Pipe can be handled in the following way:

When SLnet creates a process, it places the name of the pipe it creates for that process in a variable in the process’s environment. The Pipe Name field defines the name of this environment variable. To read this environment variable from C code, a program could call the `getenv` function. The name received from the

environment can then be used in a call to the API CreateFile (which in Win32 is used to open files and other devices as well as to create them).

If your application shells out to run other applications that do not know about the Named Pipe, you should close the pipe, and SLnet will return to screen-scraping mode. When your application regains control, it can then re-open the pipe. The Win32 API CloseHandle is used to close the pipe:

Characters may be directed to the named pipe by calling the Win32 API function WriteFile.

The following C code sample demonstrates how to open the SLnet Named Pipe directly.

```
#include <fcntl.h>    /* Needed only for _O_WRONLY definition */
#include <io.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

void main (void)
{
    char PipeName[256];
    char *EnvPipeName;
    char Temp[256];
    int PipeHandle;
    int BytesRead;

    EnvPipeName = getenv ("PIPENAME");

    /*
    ** the pipename on NT is \\.\pipe\<>pipename> where <pipename> is the value stored
    **in the PIPENAME environment variable set by SLnet
    */

    if (EnvPipeName){
        sprintf (PipeName, "\\.\pipe\\%s", EnvPipeName);
        PipeHandle = open (PipeName, _O_WRONLY);
        if (PipeHandle == -1) {
            printf ("Error: Unable to open the named pipe %s.\n", PipeName);
            exit (-1);
        }

        sprintf (Temp, "This program now controls the output to the terminal\n Type
'exit' to end.\n");

        write (PipeHandle, Temp, strlen (Temp));

        while ((BytesRead = read (0, Temp, 256)) { /* File handle 0 is standard in or
the keyboard */
            if (BytesRead == -1) /* an error occurred */
                break;

            if (strnicmp (Temp, "exit", 4) == 0) /* The user chose to exit */
                break;
        }
    }
}
```

```
        write (PipeHandle, Temp, BytesRead);
    }
    close (PipeHandle);
} else {
    printf ("SLnet is not running\n");
}
exit (0);
}
```

Idle Timeout (Minutes)

This option is disabled by default. You may set *Idle Timeout* for as many minutes as you choose, however. If there is no activity from a client in the amount of time entered, SLnet will close the process.

Example: Suppose you cannot back up your system if there are processes running. If your users have a habit of not logging off at the end of the day, you can set this option in order to end the processes so that you can backup your system.

Customizable Prompts

To customize the prompts the user sees when connecting to SLnet, click on the *Prompts* button within the *Console* tab to bring up the Customizable Prompts screen illustrated below.

Customizable Prompts Screen within the Console Tab

Logon Prompt

To change the Logon Prompt that users see when they log on, simply enter the text you wish the users to see in the dialog box. For example, if your users are more accustomed to Unix systems, you can change the logon prompt to “Login” in this dialog box.

Domain Prompt and Default Domain

The Domain Prompt text box asks the user for the domain they wish to access. The user must answer the question if you have more than one domain. If you enter the domain name in the *Default Domain* box, users will not see a domain prompt of any kind, and SLnet will access the default domain automatically.

However, if your users need to access more than one domain, use the existing prompt or enter a new prompt in the text box. In this case, make sure the Default Domain box remains empty.

Note: Enter your Windows domain name, not your Internet domain name.

Password Prompt

Enter the prompt you wish the user to see when prompted for a password in this text box. For example, if users are assigned a password that coincides with their employee number, you may change the prompt to read, “Please enter employee number.”

New Password Prompt

You may change the prompt text that requests a new password from the user in this text box.

Password Confirm Prompt

You may leave the default text here or, for example, change it to “Please reenter new password.”

Logon Messages

Must Change Password informs the user that the Windows server requires them to change their password immediately.

Invalid Password informs the user that SLnet is unable to validate their password.

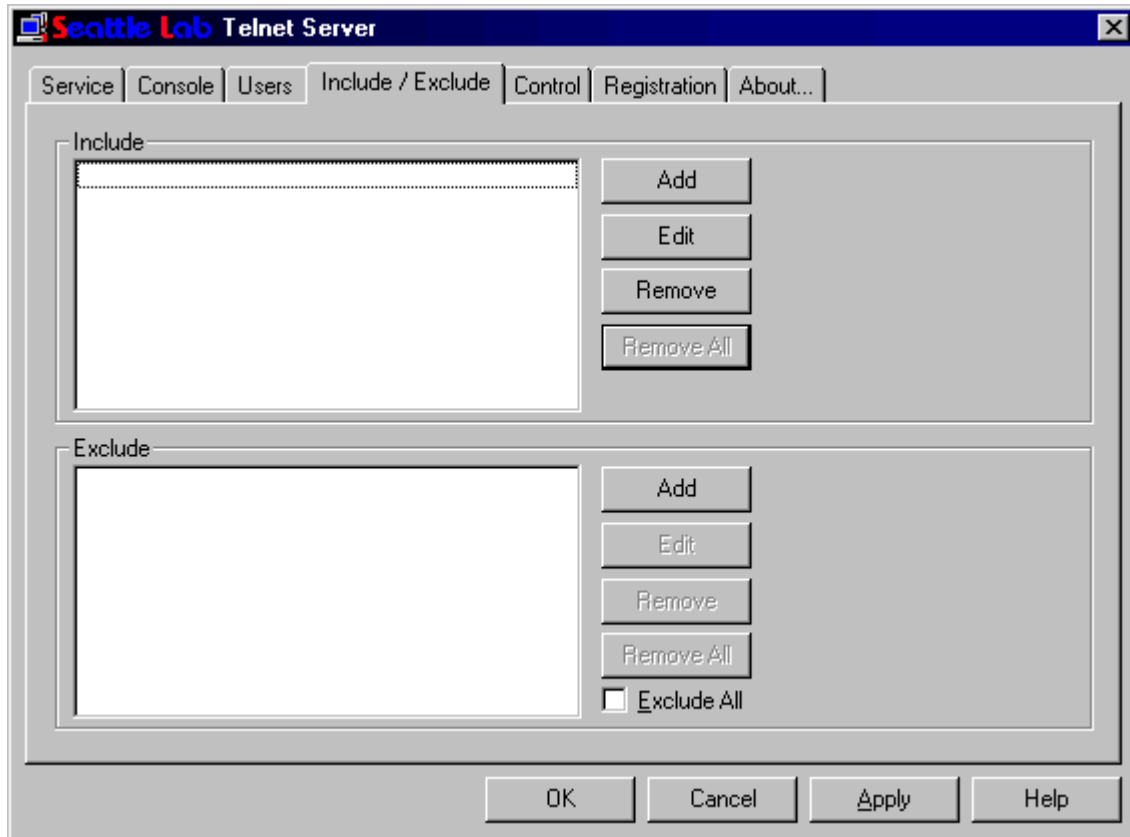
New Password Error informs the user that there was an error in changing their password.

New Password Match Error informs the user that their new password was not entered correctly the second time.

Click *OK* when you are satisfied with any changes you have made on this screen. Click *Cancel* to reset the screen to its former state.

Include/Exclude

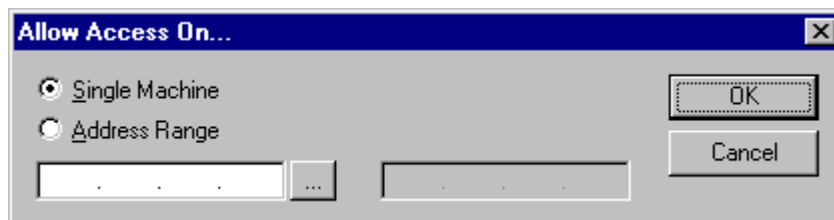
The Include/Exclude tab of the SLnet Configuration Control Panel allows you to specify ranges of IP addresses that may and may not use SLnet to connect to your system.



Include/Exclude Tab

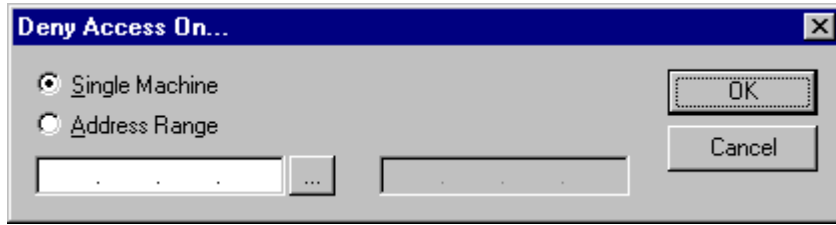
An IP (Internet Protocol) address specifies individual systems on the TCP/IP network. Using this tab, you can specify individual systems or ranges of systems for which SLnet access should be allowed or disallowed.

For example, if you wish to only allow SLnet access from a single system, perhaps that of the system administrator, check the *Exclude All* box in the exclusion section, and, using the *Add* button in the inclusion section, type <the IP address> of the single system for which SLnet access should be allowed.



Allow Access dialog box

If, to give another example, you wish to allow SLnet access from any system in the world other than that of a malicious hacker who has been vexing your organization, you would use the *Add* button in the exclusion section to type in <the IP address> of the system for which SLnet access should be disallowed.

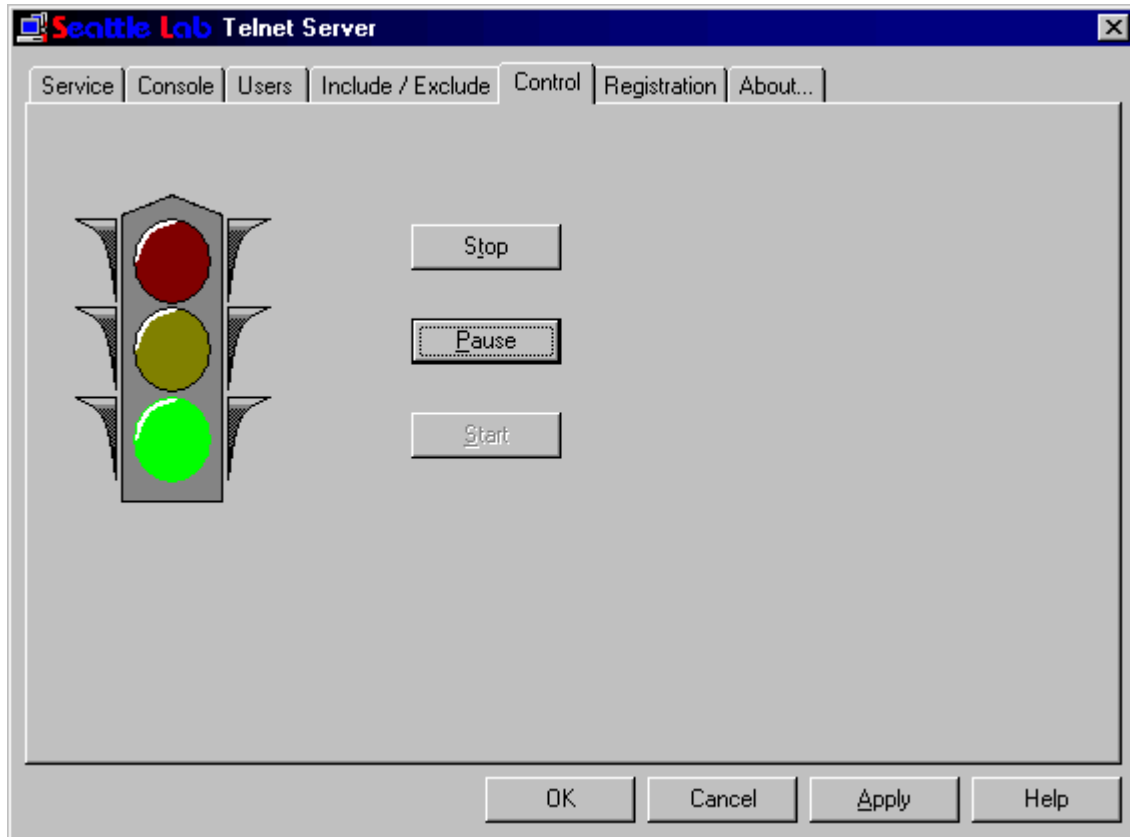


Deny Access dialog box

Note: You can specify either single systems or address ranges. If you specify an address range for exclusion, you may also specify a single address or a subrange within the excluded range for inclusion.

Service Controls

The Control tab of the SLnet Configuration Control Panel allows you to start, stop and pause the SLnet service (SeattleLab Telnet Server).



Service Controls Tab

You can stop the SLnet service by clicking *Stop*. If you stop the SLnet service, the service is completely inactivated. It is removed from memory and is no longer listening on the Telnet socket. Telnet users who attempt to connect to your Windows system will receive a connection refusal message.

Note: You should not stop the SLnet service while SLnet users are connected to your system.

You can temporarily pause the SLnet service by clicking *Pause*. A yellow light (middle) indicates that the service has been paused. When paused, all current users will be allowed to continue normally but no new users will be allowed to log in.

You can start the SLnet service by clicking *Start*.

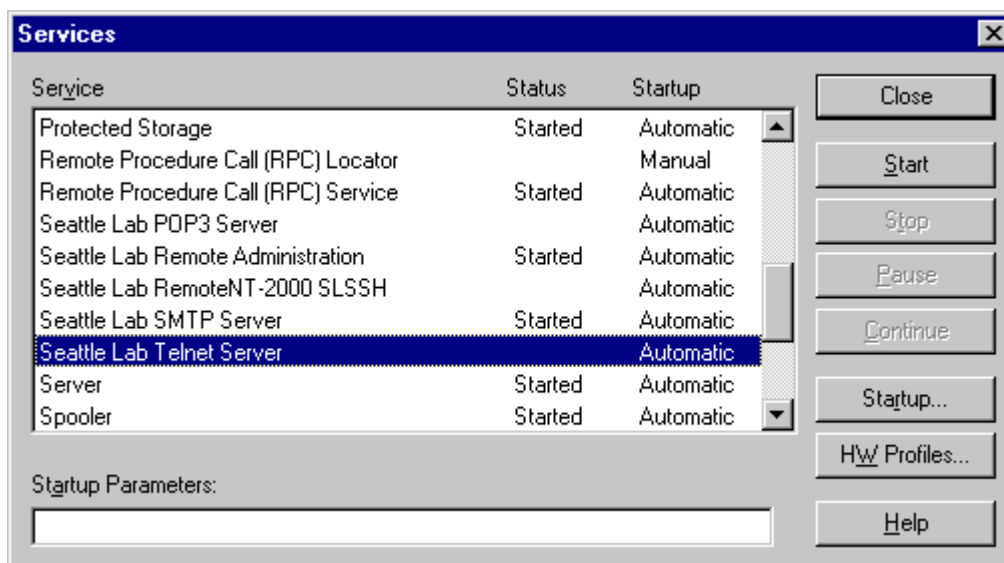
By looking at the Service tab of the SLnet Configuration Control Panel, you can tell whether the service is running. If the service is running, the *Start* button will be grayed out and the green (top) light will be on. If the service is not running, the *Stop* button will be grayed out, and the red light (bottom) will be on.

Alternative Ways to Start and Stop SLnet

The SLnet service can also be controlled from the Services applet in the system control panel. In the *Services* scroll box, choose *SeattleLab Telnet Server*. You may then start, stop or pause the service.

To set the SLnet service to start whenever the system is started, do the following: (SLnet setup sets the service to do this by default.)

- Select *SeattleLab Telnet Server* from the services scroll box.
- Click *Startup*.
- In the Services startup dialog box, select *Automatic*.
- Click *OK*.
- Click *Close* to exit the Services applet.



Control Panel Services Applet

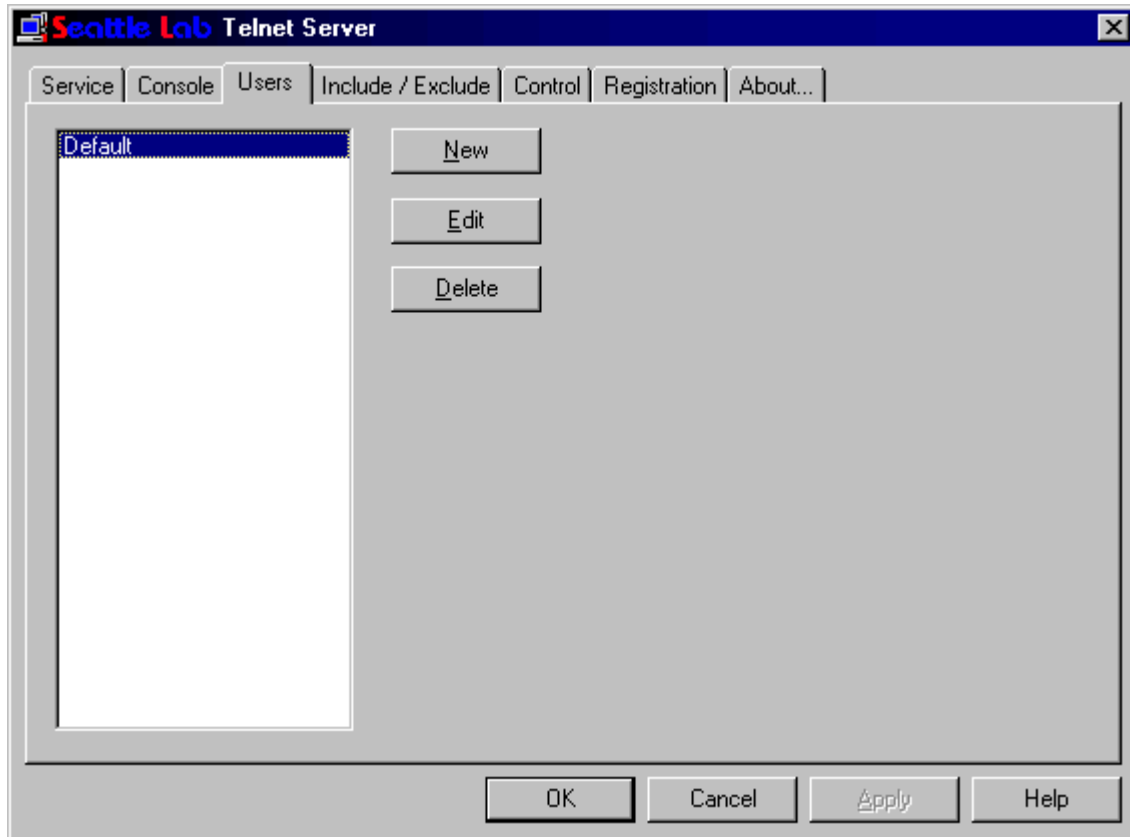
You may, of course, reverse this procedure to set SLnet to be started manually, or to completely disable the service so that it may not be started at all.

Finally, SLnet can also be started or stopped directly from the Windows Common Command Shell. To start SLnet from the Common Command Shell, type `net start slnet`. To stop the service, type `net stop slnet`.

This means that if you have the appropriate privileges, you can potentially stop the SLnet service remotely from a SLnet client. If you do so, of course, there is no way to start the SLnet service using a Telnet client.

User Access Control

The Users tab of the SLnet Configuration Control Panel allows you to decide which users should be allowed to connect to your Windows system via Telnet.



Users Tab

All valid SLnet users are based upon existing Windows user accounts. For more information on creating new user accounts, please see [Windows User Accounts](#) in this documentation.

The only further requirement to be a valid SLnet user is that you have rights to log on locally. Essentially, if a user can log onto the local system console, they can use SLnet.

Adding SLnet Users

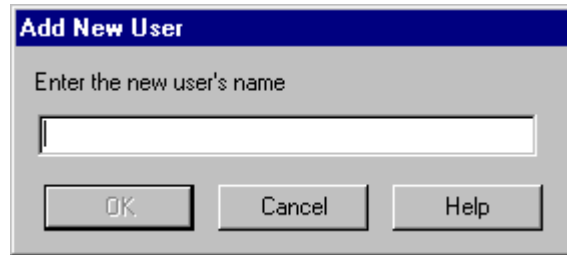
When SLnet is first installed, the users list that appears in the Users tab will default to a configuration named “Default.” This is the configuration that will be used for any user who logs onto SLnet but does not have a custom profile.

If the “Default” configuration is ever removed, only individual users who have been specifically added to the SLnet user list will be allowed to log on via SLnet.

Most system administrators will find the “Default” setting adequate. All the users the administrator has validated will now be given SLnet access. You may, however, wish to grant access to SLnet on a user-by-user basis. To do so is a simple two-step process:

First, if you wish only those specifically defined in the SLnet user list to have access rights, begin by removing the default setting by highlighting *Default* in the user list and clicking *Delete*. (At this point, because no other individual users exist in the SLnet user list, no users will be allowed to access your system using SLnet.)

Second, to add new users for later individual modification, click *New* to open the Add New User window. From this window, enter the name of any previously created system user. Each user name entered from this window will be added to the User list. Only the users that appear in this user list will be able to access your system using SLnet.



Add New User dialog box

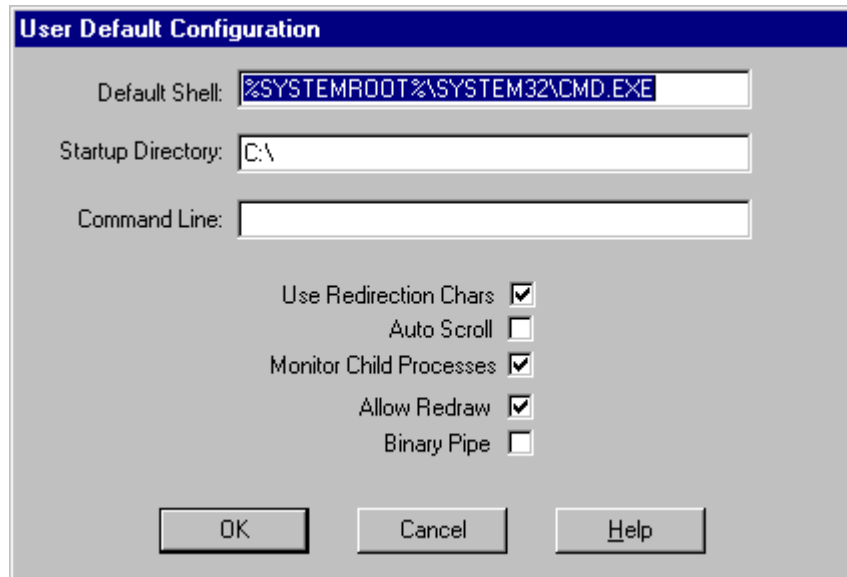
For example, if you wish to allow only Derek and Larry to access your system via Telnet, highlight the *Default* user and click *Delete*. Then click *New*. When the system presents the Add New User dialog box, type `Derek` in the text field and click *OK*. After repeating this process once more for Larry, the revised user list will be complete. Only Larry and Derek will have power to access the system through SLnet.

Note: In the above example, both Larry and Derek have had user accounts previously created that Windows NT recognizes. User accounts on your system or in your domain, therefore, must have been created using the Windows User Manager or User Manager for Domains. (See [Windows User Accounts](#) in this documentation.) Nothing in SLnet bypasses Windows security. To be granted access to your system using SLnet, a user must have first been granted access to your system by the normal Windows mechanisms.

Configuring User Accounts

SLnet allows you to define system access on a user-by-user basis. It also allows you to automatically customize the start for each user. You may, for example, decide that all users are presented with the Windows Common Command Shell or, alternatively, you may specify the shell on a user-by-user basis. You may choose to allow Larry access to the command shell but deposit Derek directly into the main menu of your application.

To edit a user's settings, simply highlight the user's name in the User list box of the User tab, then click *Edit*. This will produce the User Configuration screen for that particular user.



SLnet User Configuration Screen

The *Default Shell* field should be left as configured upon installation of SLnet. This default path allows access to the system command shell, which is the same command prompt you see from your console when you click the MS-DOS icon in the Main program group. Opening the command shell allows console applications to run properly as well as giving access to the prompt in the command shell if desired.

If you would prefer that users be taken directly to your own custom application, you can specify the directory and application in the appropriate text boxes.

The *Startup Directory* field is the path to the directory that the user will first appear in immediately after logging on. For a custom application (or any application other than the command shell), enter the full path for the directory that holds the application you wish to run.

The *Command Line* text box is used to enter the application that you wish to invoke. The application must reside in the directory specified in the Startup Directory text box. Enter the filename in the text box preceded by a switch command (*/c* or */k*).

Note: You must include either the */c* or */k* switch in the *Command Line* text box for the application to run properly.

For example, if you were executing the batch file `test.bat`, which resided in the `test_2` directory in the `test` directory in the root of `C`, you would enter `c:\test\test_2` in the *Startup Directory* text box and `/c test.bat`

in the *Command Line* text box. Please see the cmd.exe help file for additional information about invoking applications from the command shell.

The complete paths for both the *Default Shell* and *Startup Directory* fields must be entered in the format of either the Universal Naming Convention (UNC) or drive letters. That is:

c:\myapp.exe	is okay, and
\\servername\sharename\myapp.exe	is okay, but
\myapp.exe	is insufficient.

Note: In both cases the paths entered must be local. Any drive letters entered must be for non-network drives and any UNC domains must be for local share points without security.

In the *Command Line* text box you can include any other parameters that would normally be included on the command line. For example, in the above case if DIR were entered in the *Command Line* field, users would automatically receive a directory of \users\default each time they logged on.

Additional Options

In the bottom left corner of the User Configuration window seven additional user options are available. Each option may be enabled or disabled by simply clicking the adjacent check box. The default installation settings for these options are adequate for the most common use. There should rarely be any need to alter these options from there defaults and doing so may risk SLnet system integrity.

Use Redirection Chars

This option determines whether non-printable characters (characters less than 32 decimal) are mapped to printable characters. The 32 non-printable characters will be converted to printable characters based upon the following table. When a non-printable character is encountered, the printable equivalent will be substituted in its place. If this option is disabled, all non-printable characters will be represented with spaces. This option will default to enabled.

Decimal	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Non-Printable Character	☺	☻	☼	☽	☾	☿	♁	♂	♀	♁	♂	♀	♁	♂	♀
RNT Equivalent	@	*	n	*	*	\$	*	o	*	o	o	o	!		*

Decimal	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Non-Printable Character	▶	◀	↑	⏏	↓	§	■	↑	↓	→	←	L	↔	▲	▼	
RNT Equivalent	>	<		#	#	\$	-		^	v	>	<	-	%	^	v

Auto Scroll

If the Auto Scroll option is enabled, when a character is output to the last position on the screen (the 80th column of the 25th row), the screen scrolls up. Most Telnet clients support auto scrolling. If the Telnet client is set to perform scrolling, the Auto Scroll option should be disabled in SLnet to prevent incorrect screen formatting. This option will default to disabled.

Monitor Child Processes

When this option is enabled any child processes that are spawned by the user’s primary application will also be monitored by SLnet. With this option, SLnet can insure that all user processes are properly stopped before exiting. It should be noted that some applications will not run if this option is enabled. If an application functions directly from the NT system console but will not function with SLnet, disable this option. This option will default to enabled.

Allow Redraw

This option, when enabled, will clear the screen if you switch from screen scraping mode to using a custom name pipe. This option will default to enabled.

Binary Pipe

If this option is enabled, all translation of characters in the input stream is disabled. This would be appropriate, for example, for users writing custom communications software such as a file transfer protocol. This option will default to disabled.

Note: Changes made in the User Configuration window do not take effect until the next time the user logs on.

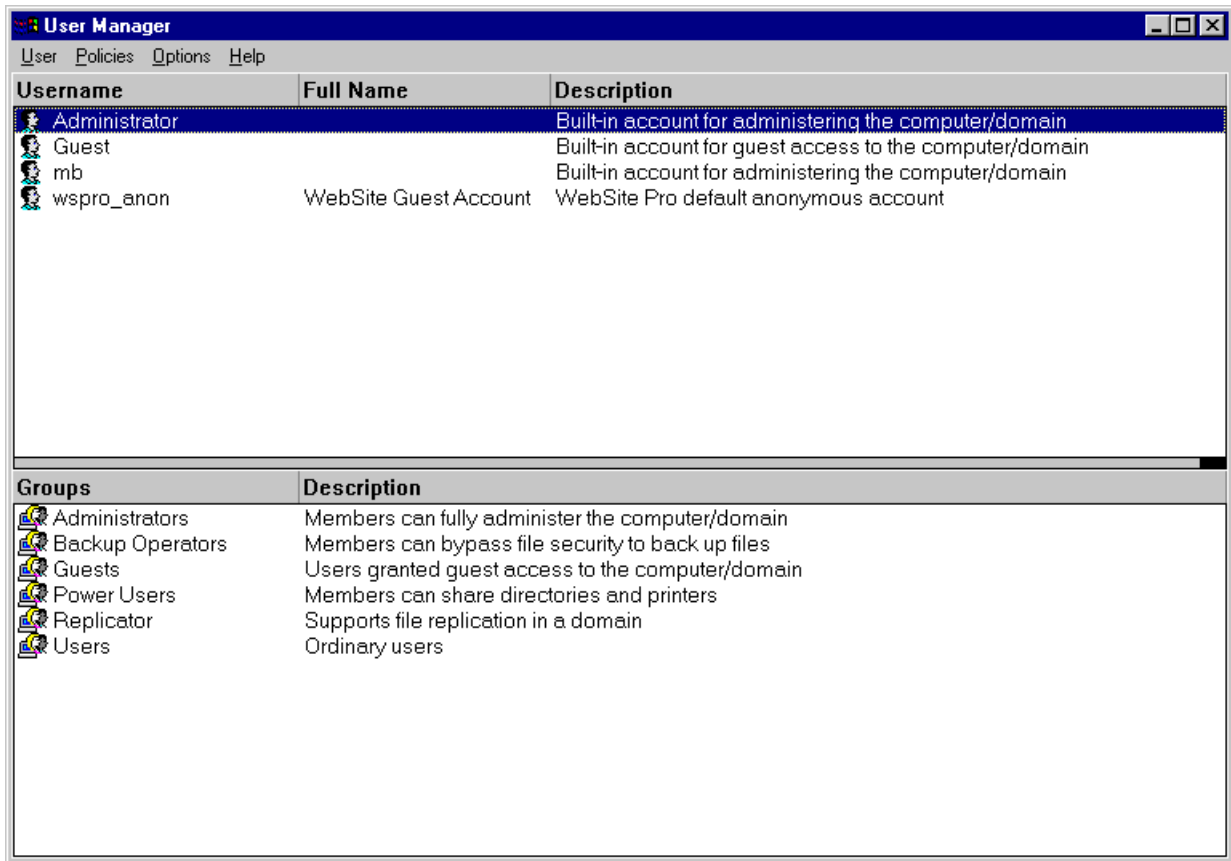
Windows User Accounts

In order to add a user account to SLnet, the user must first have a valid user account in Windows

Creating Windows NT User Accounts

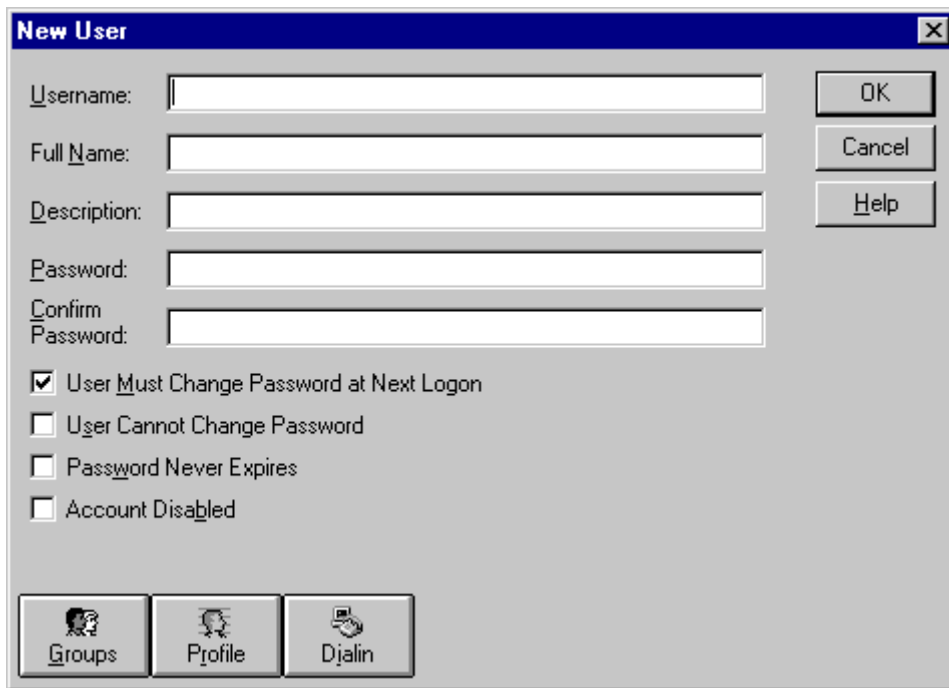
To create a new user account for Windows NT or to alter the rights of an existing user account, simply open the *User Manager* found in the Administrative Tools program group available through the (Windows) Control Panel.

Note: You must be a member of the Administrators user group to create or alter user accounts.



NT User Manager

To create a new user account, choose the *New User* option from the *User* drop-down menu. In the New User dialog box, enter the appropriate information for the new user account.



Windows NT New User dialog box

By default, all new users will automatically be members of the *Users* group that is granted access to the local console. To change the current group membership for a new user account, click *Groups* to open the Groups settings dialog box.

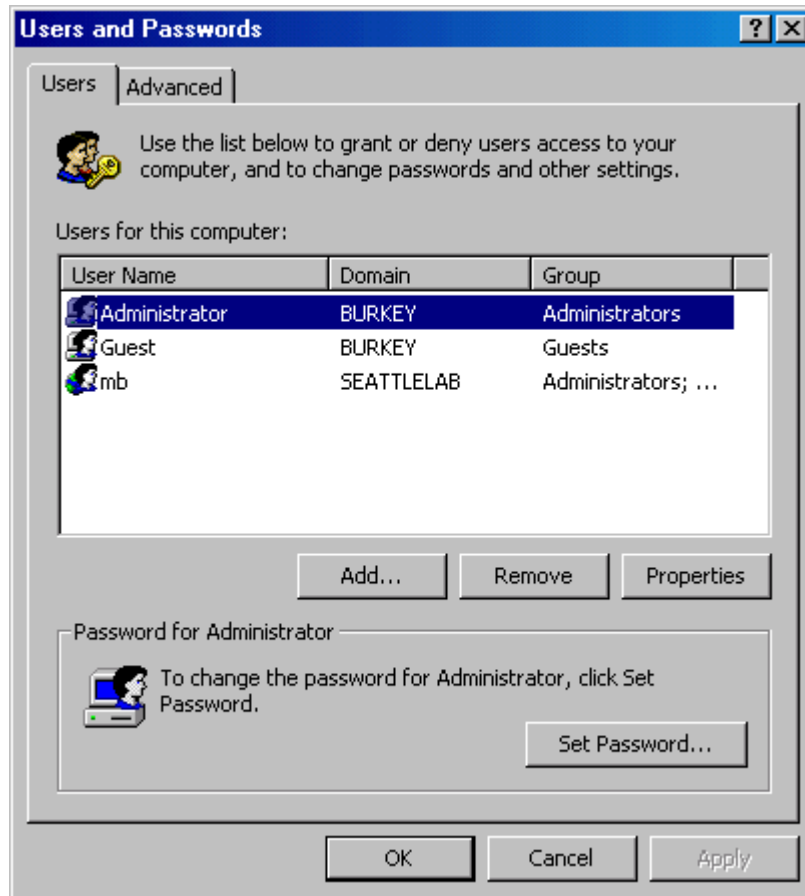
To alter the current rights for an existing user, simply double-click the desired account found in the *User Name* portion of the *User Manager* window. This will open the *User* dialog box for that specific account.

To modify the rights for a user group, choose the *User Rights* options from the *Policies* drop-down menu. From this dialog box you may mix and match various system rights, such as *Log On Locally*, with any user group.

Creating Windows 2000/2003 User Accounts

To create a new user account for Windows you must open the *Users and Passwords* window through the Control Panel.

Note: You must be a member of the Administrators user group to create or alter user accounts.



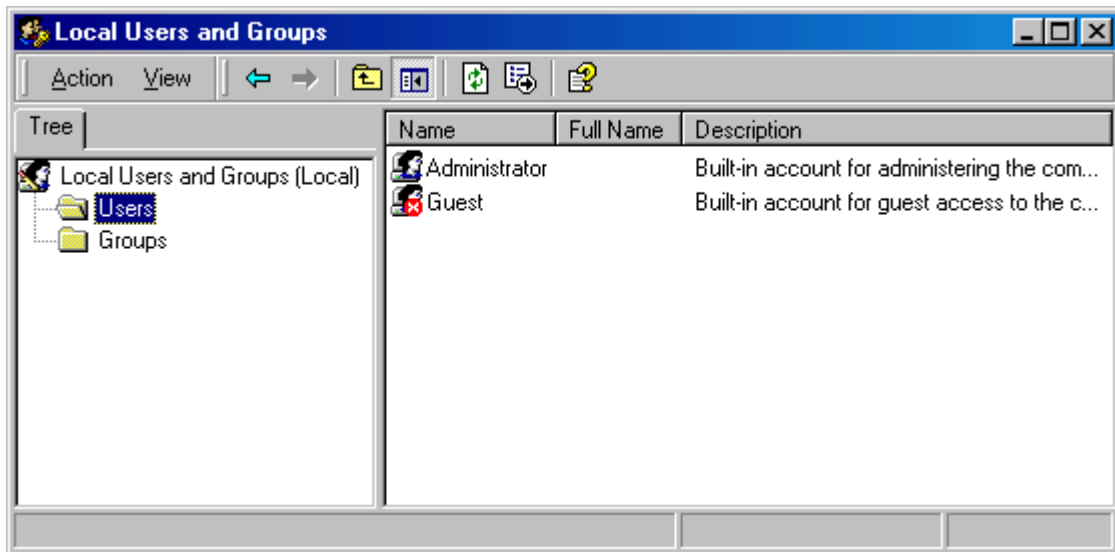
Users and Password applet

Once the User and Passwords window is open, select the *Advanced* tab and then click the *Advanced* button to open the Local Users and Groups window.



Users and Passwords Advanced tab

Click the *Users* folder in the left frame of the window to display the local users.



Local Users and Groups

Select the *New User* option from the *Action* menu to open the New User dialog box.

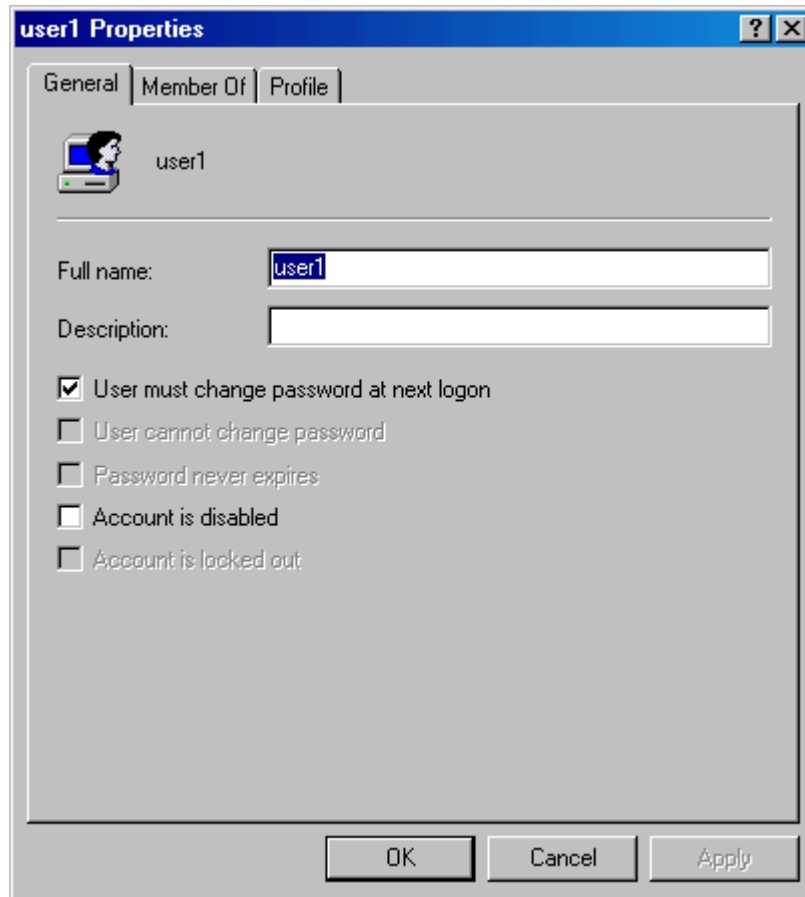
The image shows a Windows-style dialog box titled "New User". It has a blue title bar with a question mark icon and a close button (X). The dialog contains the following elements:

- Three text input fields labeled "User name:", "Full name:", and "Description:".
- Two text input fields labeled "Password:" and "Confirm password:".
- Four checkboxes:
 - User must change password at next logon
 - User cannot change password
 - Password never expires
 - Account is disabled
- Two buttons at the bottom right: "Create" and "Close".

New User dialog box

Enter the requested information in the New User dialog box and then click the *Create* button. The user account will be added to the Users group and will be displayed in the right frame of the Local Users and Groups window.

By default, all new users will automatically be members of the *Users* group that is granted access to the local console. To change the current group membership for a new user account, right-click on the new user account and select *Properties* from the popup menu or double-click on the added user to open the User Properties dialog box.



User Properties

From the *Properties* dialog box, select the *Member Of* tab. From the Member of tab click the *Add* button to add or change the group rights for the new user.

Logging on to SLnet

A Telnet client can be any of a variety of devices: a process on a UNIX machine, a Macintosh running MacTCP, a network terminal, a dumb terminal attached to SeattleLab Terminal Server (see the section on [SLnet and Terminal Server](#) elsewhere in this documentation), a Windows terminal emulator, etc. For purposes of illustration, however, assume it is the built-in Telnet applet (in the Accessories program group) on another machine. Most terminal emulators that support Telnet behave very much like the Windows Telnet application.

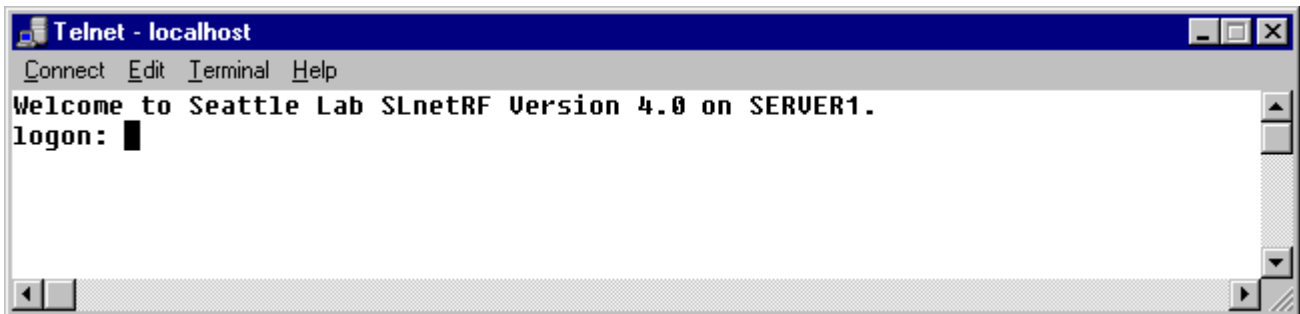
When you initially launch the Telnet client, it is not connected to anything. You need to initiate a Telnet connection. From some Telnet clients, you do so by typing commands at a Telnet command prompt. For many, the connect command is either `connect` or `open`:

```
connect hostname      or
open hostname        or
telnet hostname
```

Here hostname is the name your machine goes by on the TCP/IP network. This name may be resolved by a Windows Internet Name Service, or it may be resolved by looking in a Hosts file, or you may use a raw IP address.

From the Windows Telnet client, you connect by choosing the *Remote System* option from the *Connect* menu and typing the hostname in the text box if it does not appear in the list offered.

When you connect to a Windows system running SLnet, you will see the following prompt:



Sample SLnet Telnet Session

The greeting line specifies the host name of the system to which you have attached. In the example, it is the system named SERVER1.

The SLnet service will negotiate with the Telnet client to determine the terminal emulation type, if possible.

SLnet will then pause at the logon prompt until you enter your user name. You must be a valid Windows user on the individual system or on the domain, and you must have been granted permission to use SLnet in the Include/Exclude tab of the SLnet Configuration Control Panel.

SLnet defaults to the domain name that the host machine is logged onto. If you wish to log onto another domain, enter the domain before your user name, separated by a back slash (\). For example, if a user named Amy wanted to log onto the default domain, she would respond to the logon prompt this way:

```
logon: AMY
```

If she wanted to log onto the Engineering domain, she would respond this way:

logon: ENGINEERING\AMY

If you are logging on as a user who is not a member of the default domain, you may alternatively use the same logon format, simply replacing the domain name with the machine name. For example, if Amy were not a member of the default domain, she would respond to the login prompt this way, where Test is the name of the server:

logon: TEST\AMY

SLnet then asks you for your password. (Nothing shows on your screen as you type the password in.)

If you have successfully logged on, you will be taken to the shell specified in the Configuration Screen for your user. (The default is the Windows Common Command Shell.)

If your user name or password is not valid, SLnet will display the following message:

```
Unable to validate this account
```

It will then re-display the logon prompt. If you fail to logon correctly in the number of tries granted you by the system administrator in the Service tab of the SLnet Configuration Control Panel, your SLnet connection will be broken.

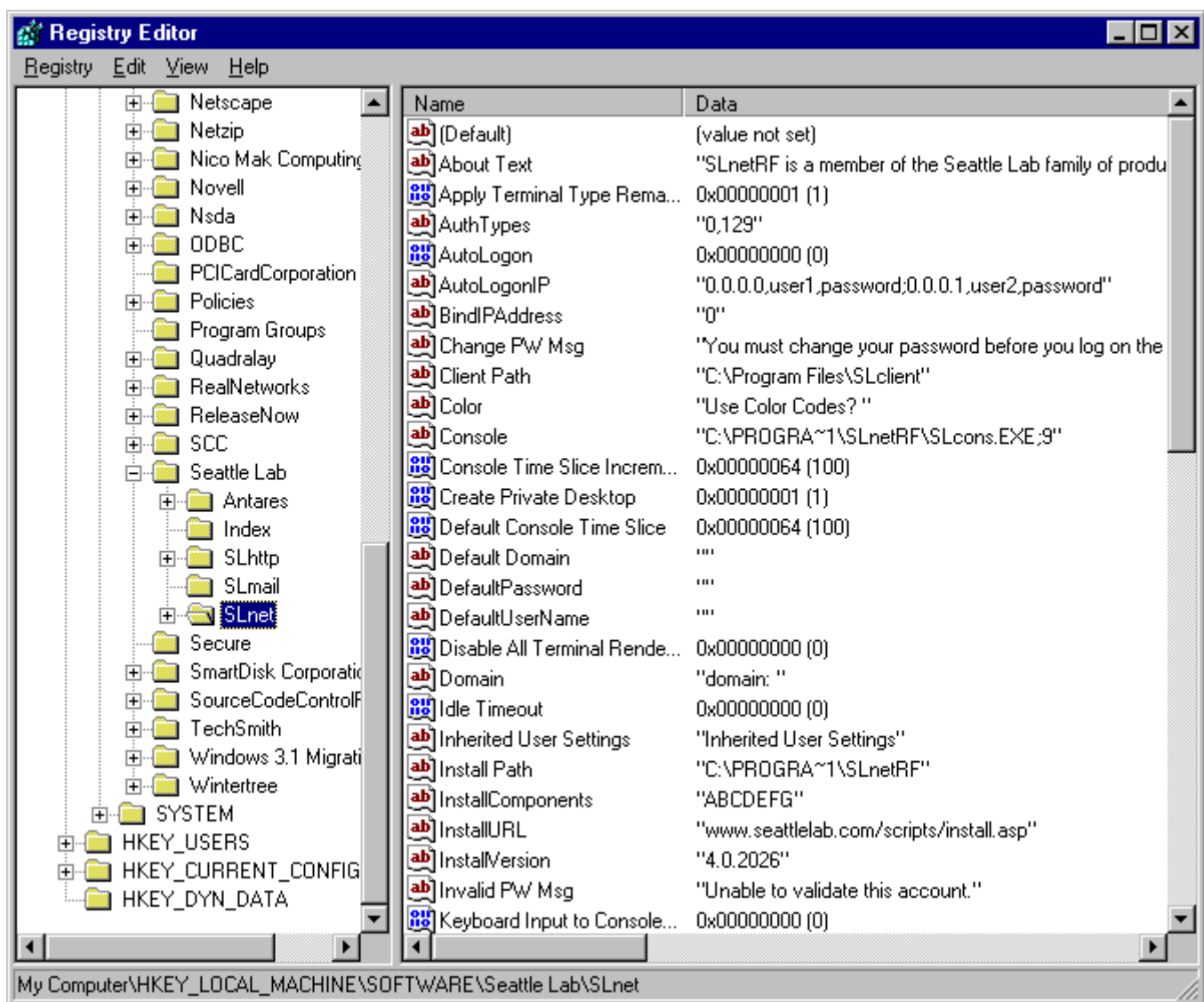
Automatic Logon

SLnet allows you to set up an auto logon using a default user account for authentication or by IP address. With auto logon anybody connecting to SLnet will automatically be logged on without having to enter their user name or password and will be deposited in the shell specified in the Configuration Screen for your user. The default location is the Windows Common Command Shell.

Auto Logon against Default User

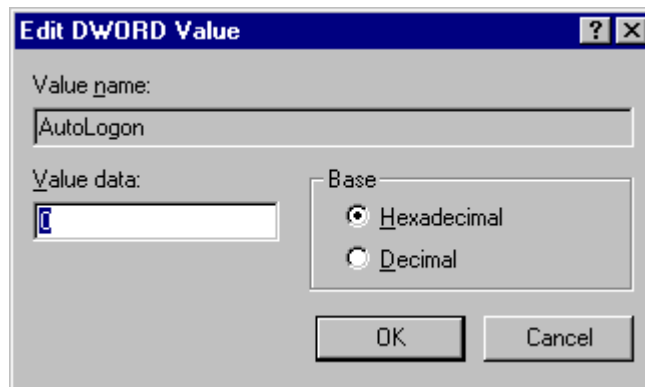
You will need to modify a registry value in the root of the SLnet registry (*HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet*) called AutoLogon.

Open the Registry Editor through the Start Menu (*Start > Run* and then entering *regedit*).



Registry Editor

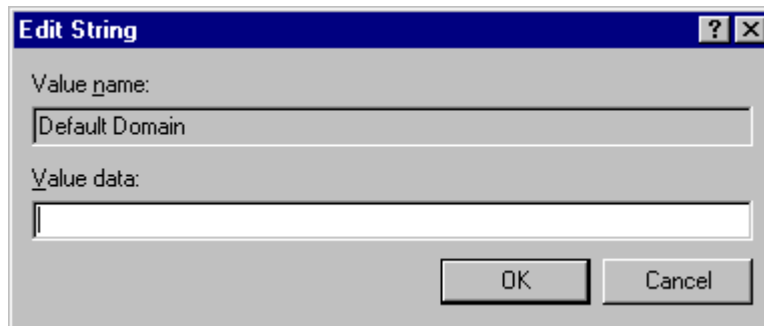
In the right frame of the registry editor, right-click on the *AutoLogon* DWORD value and select *Modify* from the pop up menu (or double-click on the DWORD value). The Edit DWORD Value dialog box will open.



Edit DWORD Value dialog box

To enable AutoLogon, change the *Value data* to 1 and then click *OK*.

You must then add the default user account information. In the right frame of the registry editor, right-click on the *Default Domain* string value and select *Modify* from the popup menu (or double-click on the string value). The Edit String dialog box will open.



Edit String dialog box

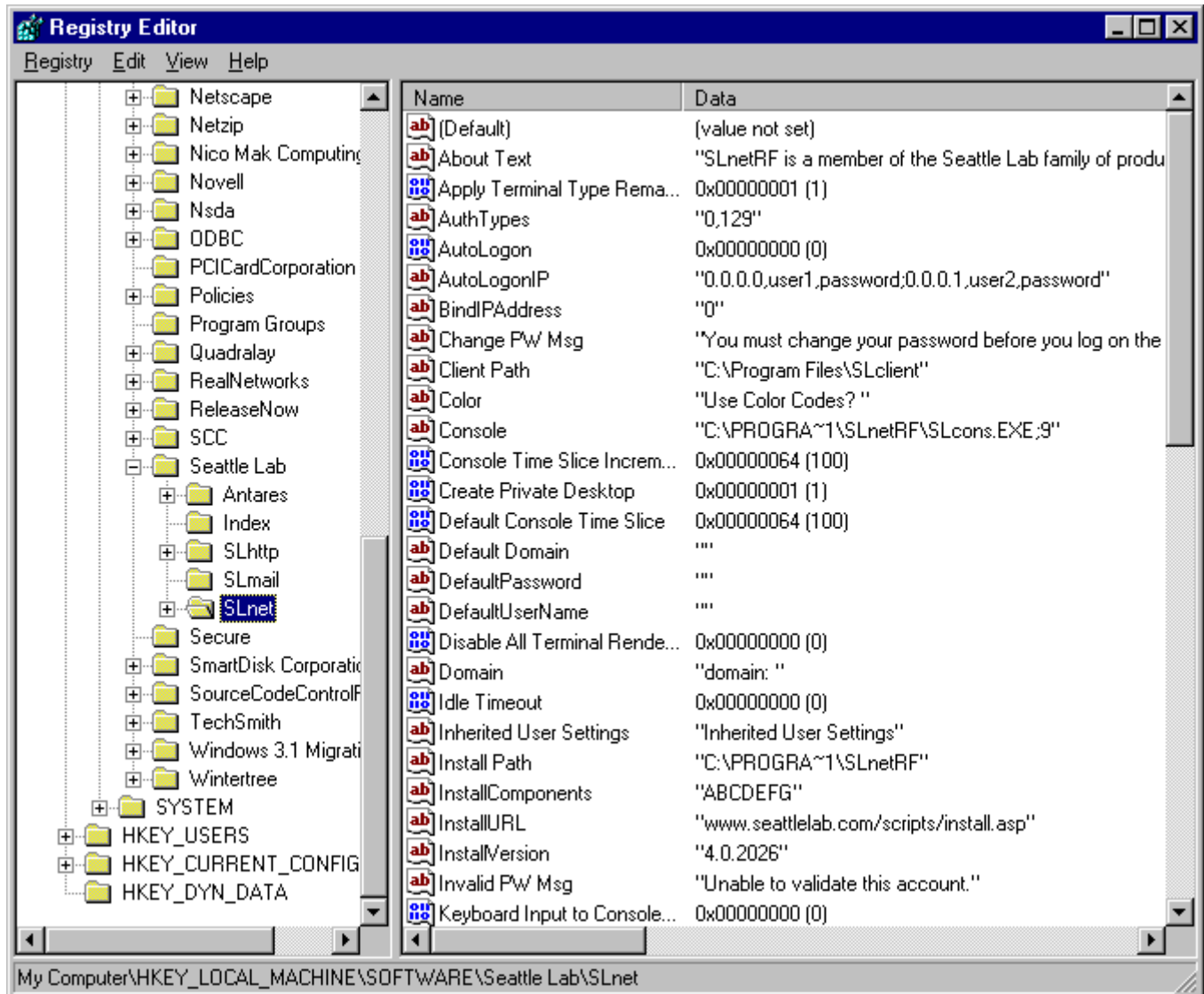
Enter the Domain that you wish to validate against for the default account and then click *OK*.

Repeat this process for the *DefaultPassword* and *DefaultUserName* strings providing the account information for the default user account. When you have completed the registry change you must stop and restart the SLnet service for the changes to take effect.

Auto Logon by IP Address

You will need to modify a registry value in the root of the SLnet registry (*HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet*) called AutoLogonIP.

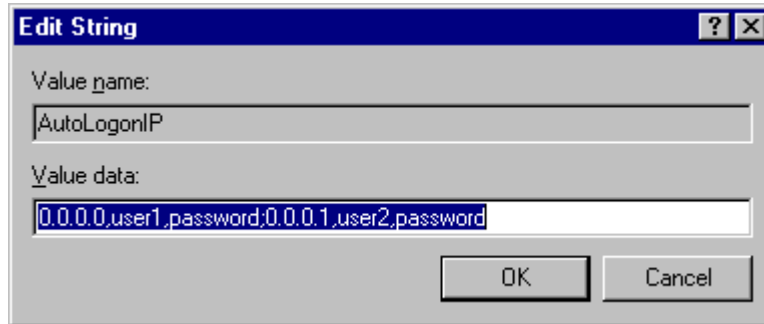
Open the Registry Editor through the Start Menu (*Start > Run* and then entering *regedit*).



Registry Editor

In the right frame of the registry editor, right-click on the *AutoLogonIP* string value and select *Modify* from the popup menu.

The Edit String dialog box will open allowing you to modify the Value data.



Edit String dialog box

Enter the <IP Address, user name, password; IP Address, . . .> for each IP Address that you wish to have auto logon access in the same form as the default entry. You may enter as many IP addresses, users, and passwords as you need.

Note: You must separate the string triplets with a semi colon as in the default configuration.

When you are finished click *OK* to update the string. When you have completed the registry change you must stop and restart the SLnet service for the changes to take effect.

Note: You must have *AutoLogon* enabled (DWORD value set to 1) in order for Auto Logon by IP to function correctly.

Exiting an SLnet Session

Once your SLnet connection is established, your Telnet client remains connected to SLnet until one of the following happens:

- You type `exit` at the command prompt.
- You disconnect the Telnet session.

You can do this, for example, by choosing *Disconnect* from the *Connect* menu in the Windows Telnet applet, or by exiting the application altogether.

Note: Ending the session in this way may leave child processes running.

- The process you are running on the Windows system terminates. (This could be an intentional termination, such as choosing *Quit* from a menu, or it could be an abnormal ending, such as having a program cause an exception.)
- Your network link is broken.

Although SLnet takes great care in an attempt to clean up after disconnections and abnormal exits, **it is best to exit cleanly whenever possible**. Educate your users to exit cleanly, too. With some Telnet clients, this means using the file exit commands rather than clicking on the “X” in the upper right-hand corner of the user interface.

SLnet Terminal Emulation

ANSI terminal emulators lack all the keys on a PC keyboard, yet software written to be run on PCs makes extensive use of these keys. To allow terminal emulator users to simulate the pressing of keys they do not have, SLnet provides mapping sequences. These sequences begin with `CTRL-A` (A for *Attention*). You first press `CTRL-A`, then another key, to simulate the pressing of a single PC keyboard key.

The keys that need to be simulated fall into three basic groups: Cursor keys (Home, Pg Up, etc.); ALT-Keys (keystrokes entered while the ALT key is held down); and PC function keys.

The following chart summarizes the CTRL-A sequences used to simulate these keys. Where ellipses (...) are used, they indicate that the pattern set by the preceding simulation is followed. For example, to simulate Function Key 2, you would use the same sequence as for Function Key 1, only replacing the 1 with a 2. The sequence must be completed in less than one second, or no key will be simulated, and the sequence that you typed will be sent to the application verbatim.

The Special Keys allow you to control processes you have generated from SLnet. It is possible from SLnet to invoke a process of which you lose control. For example, suppose that from a terminal emulator via SLnet you type `Notepad` at the command line. SLnet will invoke the Notepad applet, which is a Windows program. From your terminal emulator, you have no way of getting input to a Windows program. The Special Keys allow you to terminate this process.

`CTRL-A p` will show a process list of processes you have created. `CTRL-A k` will allow you to exit from the last process on the list. (It will prompt you before doing so.)

Sequence	Simulates
Cursor Keys	
<code>CTRL-A i</code>	Insert
<code>CTRL-A d</code>	Delete
<code>CTRL-A h</code>	Home
<code>CTRL-A e</code>	End
<code>CTRL-A u</code>	PgUp
<code>CTRL-A n</code>	PgDn
<code>CTRL-A t</code>	Back-tab
ALT Keys	
<code>CTRL-A A</code>	ALT-A
...	...
<code>CTRL-A Z</code>	ALT-Z
Function Keys	
<code>CTRL-A 1</code>	F1
...	...
<code>CTRL-A 9</code>	F9
<code>CTRL-A 0</code>	F10
<code>CTRL-A - (minus)</code>	F11
<code>CTRL-A = (equal)</code>	F12
Sticky Keys (Active until second key pressed)	
<code>CTRL-A a</code>	ALT
<code>CTRL-A c</code>	CTRL

Sequence (cont.)	Simulates (cont.)
Sticky Keys (Active until CTRL-A is repeated)	
CTRL-A s	SHIFT
Special Keys	
CTRL-A p	List Processes
CTRL-A k	Close application
CTRL-R	Refresh Screen
CTRL-A SPACE	Refresh Screen
CTRL-A l	Forced exit of application

SLnet Terminal Emulation Keys Chart

Note: The special keys affect only processes you have created. You cannot use them to list or kill another user's processes.

Multiple Console Windows

SLnet allows you to remotely open multiple console windows on the Windows desktop just as you would if you were sitting at the terminal and needed to run some programs simultaneously. This functionality is designed to be reasonably unobtrusive, displaying a menu of the processes as needed, otherwise remaining completely invisible as you work in the active window.

A single console window requires much less bandwidth and supports far more concurrent connections than multiple window do, so it is recommended that using multiple console windows (by using SLcons) only be done for administrative purposes.

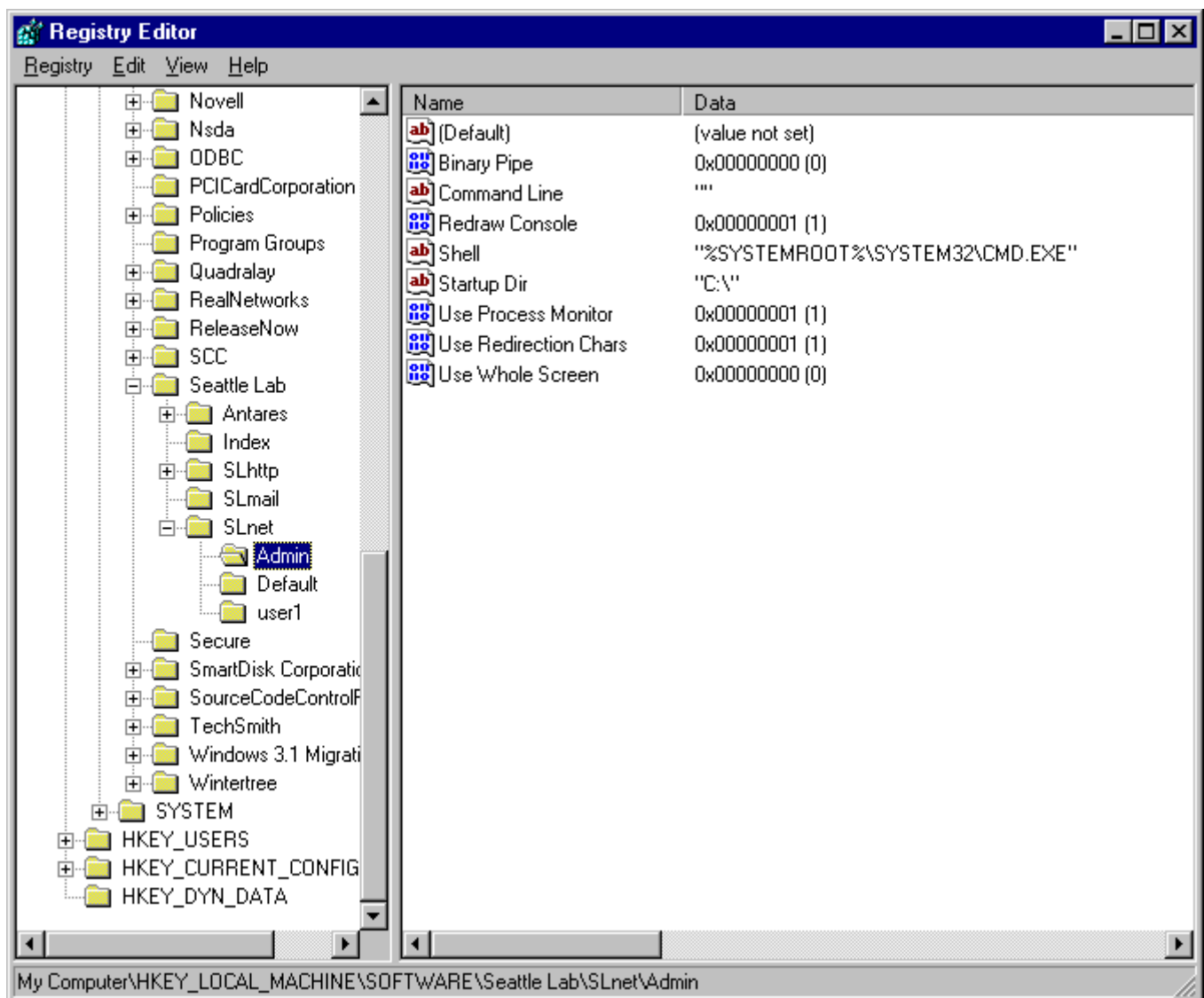
Select Console type

SLnet allows you to use SLcons.exe, which has more functionality but supports fewer concurrent connections or SLcon.exe, which has less functionality but supports more concurrent connections. For example, an administrator may want to log on with **SLcons** due to its increased functionality and limited connections while users using a hand held device may want to use **SLcon** because they may have more connections and do not need all of the functionality.

SLnet allows you to select the console type on a per-user basis.

Open the Registry Editor through the Start Menu (*Start > Run* and then entering `regedit`).

You will need to add a registry value called `Console` in the specified user directory of the SLnet registry (`HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet > <user>`).



Registry Editor

With the targeted user highlighted in the left frame of the Registry Editor, right-click in any white space area of the right frame of the window. Select *New > String Value* from the pop up menu. Type `Console` for the string name and then press *ENTER*. Right-click on the newly added string name and select *Modify* from the pop up menu. Enter `<the path to SLcon.exe or SLcons.exe;number of arguments that needs to be passed to the executable>` and then click *OK* to add the string.

The number of arguments for each executable are:

- SLcon—7
- SLcons—9

For example: `C:\PROGRA~1\SLnetRF\SLcons.EXE;9`

Note: You can copy the path to SLcon(s) from the *Console* key in the root of the *SLnet* registry.

Working with Multiple Console Windows

When you log on, SLnet creates a console window for the default shell, normally the Windows Command Shell, but this can be changed in SLnet Control Panel applet for any given user. You can work in this console window as you would in any Telnet session, or you can immediately start the next process in a second console window.

To start a second process, type the command to start that process, for example: `start cmd`. On the Windows server there are now two console windows available to you but you are still looking at the Telnet client window displaying the contents of the first console window. Your keyboard input will continue to go to this first window and the second window is not displayed until you switch to it.

To switch to the second console window, press `CTRL-A 2` (hold down the `CTRL` key and press `a`, release `CTRL` and `a`, and then press `2`). By default, you have two seconds to send the `2` keystroke after pressing the `CTRL-a` combination, otherwise `^a` will be sent through verbatim. This timeout is configurable in the registry.

Proceed to work in this second window, just as you would in any normal Telnet session, by typing the commands you want to execute. SLnet forwards all keystrokes to your current console window and will forward that window's output to your terminal. The first window still contains any process you started there, but until you switch back to it, you will only see the work in the second console window.

The status line at the bottom of the screen confirms which console window is displayed as well as occasionally providing other information. The status message is displayed for 5 seconds and then disappears. For extensive online Help, press `CTRL-A h` and use the menu key sequences described there to get full SLnet Help.

The following SLnet commands are available to you while working in the console window you have created, in addition to commands available in the shell that is running:

Command	Result
<code>tlist</code>	lists the tasks with their process ids
<code>kill <PID></code>	kills the specified process

SLnet also provides Hot Key commands to work in a console window and to switch between windows. This is an abbreviated list of key sequences that are useful for working in a console window "normal mode." A second set of key sequences is available when working in "menu mode," see [Using the Menu](#) for more information.

Key Sequence	Result
CTRL-A c	Creates console window and switches to it
CTRL-A w	Displays the Menu with desktop, popups (dialog boxes or error messages) and windows. Any window can be expanded.
CTRL-A 1	Switches to console window 1 (use similar sequence for windows 2 through 9)
CTRL-A f 1	Simulates F1 key (use similar sequence for function keys 2-9)
CTRL-A h	Displays the Menu of help
CTRL-A #	Kills connection (be certain that you don't hold down shift and control key simultaneously if you are using NT's Telnet)
CTRL-A C (capital C)	Sends ^C signal to process (see caution above about holding down shift and control key simultaneously)
CTRL-A B (capital B)	Sends Ctrl-Break to process (see caution above about holding down shift and control key simultaneously)

Note: Almost all Hot Key commands that take a lowercase letter after the Ctrl-A can also be typed as a control character—for example, CTRL-a c can be typed as CTRL-a CTRL-c.

See the [Hot Key Commands Table](#) at the end of this manual for a complete listing of available key sequences.

Switching Between Windows

The ability to switch immediately between console windows allows you to multi-task using your Telnet client much as you would if you were sitting at the computer running Windows. You can “hot-switch” between windows directly (similar to using ALT+TAB to switch between all the running applications in Windows), or you can display the *Menu* and select a window from the list.

To switch directly to the first window from the second window, press ^a 1 (hold down the CTRL key and press a, release CTRL and a, and then press 1). By default, you have two seconds to send the 1 keystroke after pressing the CTRL-a combination, otherwise ^a will be sent through verbatim. This timeout is configurable in the registry.

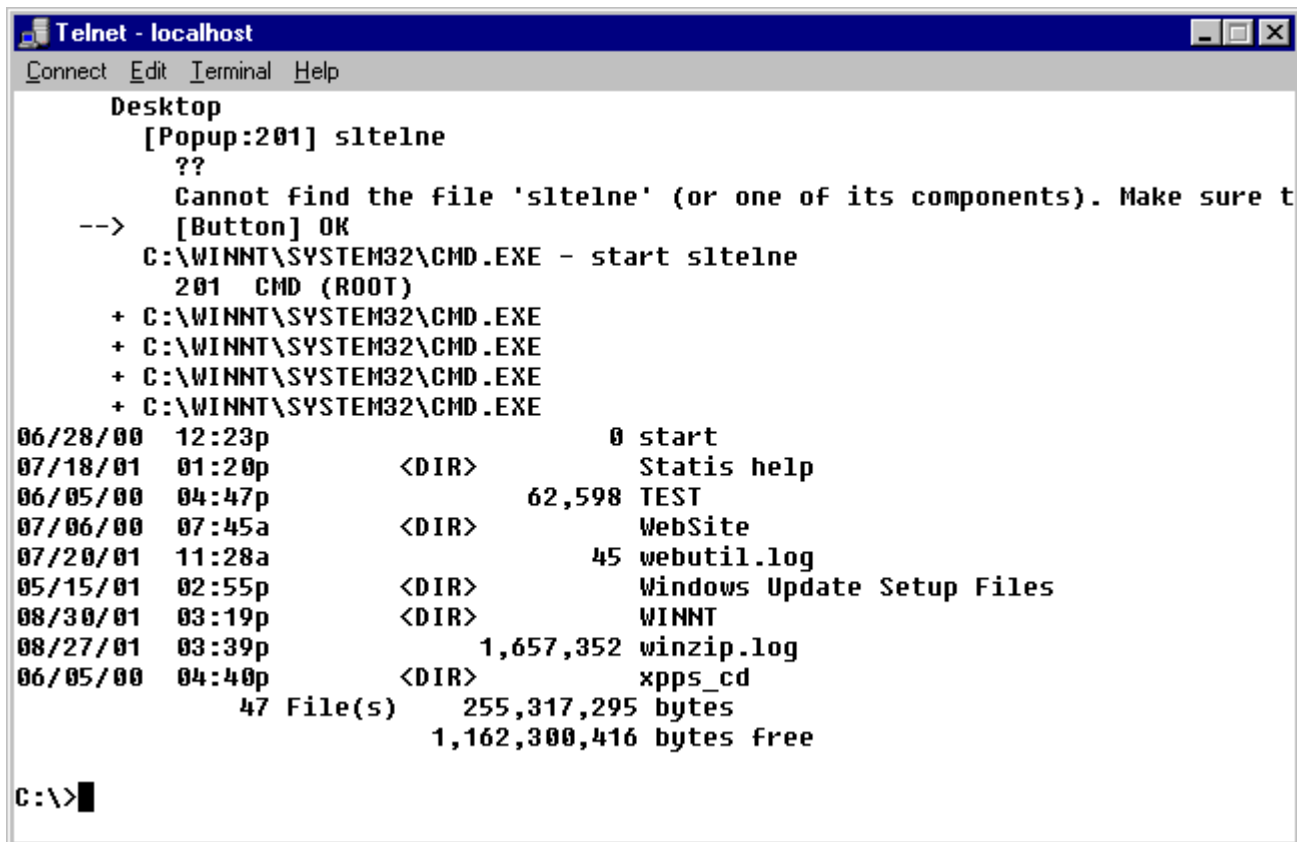
The information from the window you switched to will now be displayed in your Telnet client window and any command you type will be executed in this console.

Confirmation or error information about the window you switched to using a hot key will appear in the status line at the bottom of your screen. This status information will display for 5 seconds, then disappear and the bottom line of the screen will display the contents of your window.

The windows are numbered in the order that they appear in the menu. The ordering sequence for console windows uses reusable numbers; so if you kill a window, that window number will be assigned to the window that follows, or to the next window created if it was the last window. You can switch among as many as twelve console windows using the key sequences, see [Hot Key Commands Table](#). To switch to another window by displaying and selecting it from the Menu, see [Using the Menu](#).

Using the Menu

To gain an overview of the windows and processes you are running in the Telnet client window you can display the Menu at any time by pressing `CTRL-A w`. The Menu is displayed in the first few lines at the top of your Telnet client window and the current console window occupies the rest of the window below the Menu.



```

Telnet - localhost
Connect Edit Terminal Help
Desktop
[Popup:201] sltelne
??
Cannot find the file 'sltelne' (or one of its components). Make sure t
--> [Button] OK
C:\WINNT\SYSTEM32\CMD.EXE - start sltelne
201 CMD (ROOT)
+ C:\WINNT\SYSTEM32\CMD.EXE
+ C:\WINNT\SYSTEM32\CMD.EXE
+ C:\WINNT\SYSTEM32\CMD.EXE
+ C:\WINNT\SYSTEM32\CMD.EXE
06/28/00 12:23p 0 start
07/18/01 01:20p <DIR> Statis help
06/05/00 04:47p 62,598 TEST
07/06/00 07:45a <DIR> WebSite
07/20/01 11:28a 45 webutil.log
05/15/01 02:55p <DIR> Windows Update Setup Files
08/30/01 03:19p <DIR> WINNT
08/27/01 03:39p 1,657,352 winzip.log
06/05/00 04:40p <DIR> xpps_cd
47 File(s) 255,317,295 bytes
1,162,300,416 bytes free
C:\>

```

The Menu is displayed at the top of the Telnet Client Window

- The top line of the Menu indicates the Desktop that contains the windows that are currently running.
- The arrow (`-->`) highlights the currently selected menu item.
- The plus sign (+) indicates that collapsed sub-tree information is available under the menu item.
- Poppers (dialog boxes or error messages) are displayed directly under Desktop unless the view has been changed. See [Locating and dispatching Poppers](#) for more information.
- When the information is expanded, processes are shown under the windows they are running in with the Process ID followed by the executable name for process then the current window title for windows and console groups.
- `xxx` next to a window or process indicates it has exited or been killed, it will disappear from the list shortly after you move the highlight arrow.

The active window may be displayed in a different color (depending on your client, selected terminal type, and the terminal databases description of that terminal type).

While the Menu is displayed you are no longer in any of your console windows and all the input from the keyboard is used to navigate within the Menu. Use the arrow keys on your keyboard if available or the keystrokes in the table below to: move selection arrow; expand or collapse information; locate poppers; kill processes, poppers, or

windows; and switch to a selected window. See the [Hot Key Commands Table](#) for a complete listing of all SLnet key combinations.

This key...	has this effect when the Menu is displayed
j	Moves highlight arrow down one line
k	Moves highlight arrow up one line
b	Back one page (PgUp)
f	Forward one page (PgDn)
+	Shows sub-items hidden under the highlighted item
-	Hides sub-items
TAB	toggles collapse/expand display for selected menu item
Z (capital Z)	kills highlighted process or popup
ENTER	(when pointing to a window) closes the menu and makes the highlighted window the current console window
ENTER	(when pointing at a popup) locates popup under the process that created it
v	(when pointing at a popup) displays popup location
ESC	hides the menu and returns to selected console window
q	hides the menu and returns to selected console window

Initially, all console windows are collapsed, so that the desktop, popups, and windows are visible, but process sub trees are not. When a console window is expanded, each process tree associated with that console is listed by ascending Process ID (note that processes form a hierarchy, and can appear under other processes).

Locating and dispatching Popups

The work you do in your console window may cause an error message or dialog box requesting some response and these will appear as a popup in your session, preventing further work in that console window until the popup containing that error message or dialog box has been dispatched.

If your work in a session has been interrupted and you can no longer input to that window, it is best to check for a popup by bringing up the Menu (press CTRL a CTRL w).

If the Menu lists a popup under the desktop, use your arrow keys or equivalent keystroke combination to navigate to it. Press ENTER or v to view the popup under the associated console window. The Process Identifier (shown inside the bracket) refers to the process that created the popup.

The safest way to dispatch the popup is to move the highlight to the popup on the menu and press Z to kill it. If that fails, the second safest thing to do is to kill the console window. If you don't want to kill the window, go to the process associated with the popup and kill the process directly using Z.

Applying Z to a process is more dangerous than killing a window because the process won't have a chance to exit gracefully. Since the menu is updated every few seconds, it may take a few seconds for windows that you kill to disappear from the menu.




Killing a window or dialog box using Z is the same as clicking the "Close" button in the upper right corner of the window.

SLnet Logging

SLnet keeps a record of its activities by entering messages in the Windows Event Log as events occur. To view these messages, access the Event Viewer, click *Start > Programs > Administrative Tools (Common) > Event Viewer*.

The Event Viewer provides three event logs: System, Security and Application. To view event logs for SLnet, go to the *Log* menu on the tool bar and select *Application*.

Windows Event Viewer

The key columns in the *Event Viewer* are the *Source* column, which shows the application to which the entry relates, and the *Event* column, which shows the ID number for the event. A  symbol displayed in the Event Viewer indicates a log entry for what Windows considers a “critical error.” A blue  symbol displayed in the Event Viewer represents an “information” log entry. A yellow  symbol represents a “warning” log entry.

SLnet writes a record in the Application Log for the events described in the Error Code Table. These events are described in the section on [SLnet Error Codes](#). To determine which event took place, look up the number from the *Event Viewer*’s Event column in the following Error Code Table’s Event column.

Note: The [%1] and [%2] tags in the following examples are values that are filled in by SLnet at the time the event is generated.

Message Examples

1. If the socket SLnet is trying to use is already in use by another program the user will get an Error message. The number *1000* will appear in the Event column, and the Error message will be: “The [%1] service could not be started.”
2. If a user who does not have permission to use SLnet attempts to log on, the number *2000* will appear in the Event column, and the Warning message will be: “User [%1] attempted to log on with no profile. There was no default.”
3. When a user logs onto SLnet, the number *2002* will appear in the Event column, and the Information message for that event is: “Accepting connection from IP: [%1]. Connection Number [%2].”
4. When the Telnet service is successfully started, the number *3000* will appear in the Event column, and the Success message for that event is: “[%1] has been successfully started.”

Note: If the user is connecting via a proxy server, the IP address will always be the address of the proxy server rather than the client machine on the other side of the proxy server.

You may select which types of events you wish recorded in the Event Log by going to the SLnet Configuration Control Panel *Service* tab. In the Event Logging box, uncheck the boxes for the types of events that should not be recorded. The default setting is to record all three types of events.

SLnet Error Codes

In the Error Code Table, the Event column contains the event's ID number, the Type column specifies the type of error and its level of importance (Error, Information, or Warning). The Error Message column contains descriptions of the errors.

Note: The [%1] and [%2] tags in the Error Message column are values that are filled in by SLnet at the time the event is generated. Generally speaking, [%1] tags represent applications and [%2] tags represent machine IP addresses, although that is not *always* the case.

The following table explains error codes generated by SLnet. While many of these error codes are self-explanatory, some are not. Most of these error codes pertain to the authentication security between SLnet and Windows NT/2000/2003.

Event	Type	Error Message
1000	ERROR	The %1 service could not be started.
1001	ERROR	Unable to find socket library.
1002	ERROR	Unable to allocate memory for Socket Info Structure.
1003	ERROR	An error occurred while attempting to allocate memory.
1004	ERROR	%1 has not been installed correctly. Please re-install.
1005	ERROR	%1 %2 (generic)
1006	ERROR	%1
1007	ERROR	The user's process returned %1.
1008	ERROR	Assert in file %1.
1009	ERROR	Invalid registration data. The service is shutting down.
1010	ERROR	The demo version of %1 has expired.
10000	ERROR	The Socket error is undefined.
10004	ERROR	The Socket has been interrupted.
10024	ERROR	No more file descriptors are available.
10036	ERROR	A blocking Windows Sockets operation is in progress.
10041	ERROR	The specified protocol is the wrong type for this socket.
10043	ERROR	The specified protocol is not supported.
10044	ERROR	The specified socket type is not supported in this address family.
10047	ERROR	The specified address family is not supported.
10050	ERROR	The Windows Sockets implementation has detected that the network subsystem has failed.
10052	ERROR	The network has been reset.
10053	ERROR	The connection was aborted.
10054	ERROR	The connection was reset.
10055	ERROR	No buffer space is available. The socket cannot be created.
10060	ERROR	A timeout has occurred.
10093	ERROR	A successful WSASStartup must occur before using this API.
2000	WARNING	User %1 attempted to logon with no logon profile. There was no default.
2000	INFORMATION	The terminal type is: %1.
2001	ERROR	A connection from address %1 was rejected because it is not in the accepted address range.
2002	INFORMATION	Accepting connection from IP: %1. Connection Number %2.

2003	INFORMATION	IP %1 Disconnected.
2004	INFORMATION	The demonstration period will expire on %1.
2005	INFORMATION	%1 user version initialized.
3000	SUCCESS	%1 has been successfully started.
3001	SUCCESS	%1 has been successfully stopped.

SLnet Frequently Asked Questions

What is SLnet?

SLnet is a Telnet Server for Microsoft Windows NT/2000/2003.

What applications can SLnet run?

SLnet can run character-mode applications: DOS text applications, Win32 console applications, OS/2 character-based applications, and POSIX applications.

What applications can SLnet not run?

SLnet cannot run Windows based (gui) applications. Any applications that use a GUI will not run in SLnet.

Some Win32 applications create a second console window that is invisible to SLnet and therefore SLnet will not be able to display their output. At this time, the only applications that do this are GNU UNIX ported utilities such as vi.

How much memory does SLnet require?

SLnet requires one megabyte of system memory for the Service and less than one megabyte of memory for each logged-on user, plus whatever memory requirements the applications being run require.

The formula for determining how much memory is required is:

$$\text{NT memory requirements} + 1\text{MB SLnet} + (1\text{MB} \times \text{Number of users})$$

So for SLnet running on NT 4.0 Workstation to support two users:

$$16\text{MB (NT)} + 1\text{MB (SLnet)} + 2 \times (1\text{MB}) = 19\text{MB}$$

What terminal types does SLnet support?

SLnet supports VT100, VT220, and ANSI terminal types.

What is the maximum number of concurrent connections I can make to my server?

As a general rule, the more CPU and RAM that you have on your system the more connections you can make. The following is a table that shows the approximate maximum number of connections that SeattleLab has tested. The Operating System, CPU, and RAM for each test are listed for reference.

SLnet RF maximum number of tested connections to a single server		
Operating System	CPU/RAM	Connections
Windows 2000	K6 2-233Mhz, 256 Megs RAM	58 (default installation) 198 with MS reg tweak (3072)*
Windows NT	P Pro 200Mhz, 128 Megs RAM	51 (default installation) 206 with MS reg tweak (3072)*

* The Microsoft registry tweak involves changing a registry value. Open the registry editor by clicking *Start > Run* and then typing `regedit` then clicking *OK*, or by using RegShell shipped with SLnet. Expand `HKEY_LOCAL_MACHINE > SYSTEM > CurrentControlSet > Control > Session Manager > SubSystems`.

In the right frame you will see the list of string and DWORD values in the subsystem key. Highlight the *Windows* string value and then right-click on the string name. In the popup menu, select *Modify*. The Edit String dialog box will open.

The entire string value should look something like this (all one line).

```
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,3072  
Windows=On SubSystemType=Windows ServerDll=basesrv,1 ServerDll=winsrv:UserServerDl-  
lInitialization,3 ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off  
MaxRequestThreads=16
```

Scroll through the string until you see "SharedSection=1024,3072."

Change the first number (1024) to 3072 and add , 3072 after the 3072 that is there by default. The string should now look like this "SharedSection=3072,3072,3072." Click *OK* to accept the changes.

The full documentation of this registry change is available in the Microsoft Knowledge Base at <http://support.microsoft.com/support/kb/articles/Q142/6/76.asp>.

Note: You must restart your system in order for this change to the registry to take effect.

How many user licenses do I need?

There are two kinds of licenses that you can purchase for SLnet. A single license for each user (the number of users is decided by you) or a single license that allows an unlimited number of users. In either case, there is only one machine (Telnet server) on which SLnet is installed, that the users can access concurrently from their workstation.

SLnet Tips & Troubleshooting

How do I add users to SLnet?

SLnet uses the Windows user database, so adding users to SLnet is as easy as adding a new user via the User Manager.

What if I can't log on to SLnet?

Make sure you have Log on Locally rights. This means you must be able to log on with the user name and password at the machine you are sitting in front of.

If the user is a local user and not a member of the domain, you will need to specify the local machine name in the logon line.

Be sure to include a back slash (\) between the machine name and your user name.

How can I stop a program that keeps running after I log off?

SLnet normally cleans up after your user processes, exiting all processes when you log off. If you wish to disable this feature, you may disable the Monitor Child Processes option located in the Users tab of the SLnet Configuration screen. This option is set on a per user basis.

Note: If you disable this feature, you will need to manage process clean-up yourself, or your system may become littered with leftover processes.

How can I use mapped drives from SLnet?

It is important to understand that if you map a drive, the same mapping applies to ALL users. For example, if one user has drive F mapped there is no need to map it for another. If you do attempt to map it again, your attempt will fail. In a multi-user environment, the best practice is for the system administrator to set up the drive mappings globally so that individual users or individual applications do not have to.

How can I print from SLnet?

Windows maps printer ports globally, meaning that LPT1 points to the same printer whether you are logged in on the desktop or through SLnet. If you change the printer that LPT1 is mapped to locally on that machine, you change it for all users who are currently logged on. Therefore, SeattleLab recommends that the system administrator designate specific printers to specific ports. Do not allow users the ability to change these values.

How do I map Windows drives in Telnet?

Although Windows is user specific, the drive letters are global. If a user is logged onto the server's desktop locally and has drives mapped to the network, anyone who logs on via Telnet should see those mapped drives—as long as the security share permissions are set up for the user who is logging on via Telnet. If there is no user who has logged onto the desktop locally, and a user logging on via Telnet maps a drive in their Telnet session, any users logging on via Telnet will see that mapped drive as long as the first user is still connected. When the first user disconnects, that drive mapping will not be available until a current user re-maps the drive. You can also use a *.bat file to map the drive as each user logs on. SLnet's Default User Account is used when a specific user account is not already set up. In the properties for the default user at the Command Line field you can type the following path: /k c:\testpath.bat.

The following is an example of a *.bat file:

```
@echo off
net use r: \\machinename\d$
:done
```

How do I create Terminal or console beeps in SLnet?

Applications written on UNIX are by design able to run from any terminal, whether remote or local. Windows and MS-DOS applications can only be driven by a remote terminal by emulating the CRT and keyboard over the remote connection.

Because of the design of Windows, any console beeps are sent to the server via a device driver which SLnet and RemoteNT cannot intercept.

Using the C code below, it is possible to create custom applications that support terminal beeps.

```
#include <fcntl.h> /* Needed only for _O_WRONLY definition */
#include <io.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
void main (void)
{
char PipeName[256];
char *EnvPipeName;
char Temp[256];
int PipeHandle;
int BytesRead;
EnvPipeName = getenv ("PIPENAME");
/*
** the pipename on NT is \\.\pipe\

```

```
}  
exit (0);  
}
```

How can I setup an automatic logon for all users connecting to SLnet?

Open the Registry Editor by clicking *Start > Run* and then typing `regedit` and then clicking *OK*, or by using RegShell, shipped with SLnet. Expand *HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet*.

The string values for all of the SLnet registry keys will be displayed in the right frame.

You need to modify (or add) the following values to the Registry Editor in the SLnet key:

```
AutoLogon  
DefaultUserName  
DefaultPassword
```

If the registry values are present in the SLnet key (they are by default) you will need to modify the values.

Highlight the *DWORD Value* labeled *AutoLogon* and then right-click on the string name. Select *Modify* from the popup menu. The Edit *DWORD Value* dialog box will open.

In the *Value data* text box enter `1` and then click *OK* to accept the changes.

Highlight the *String Value* labeled *DefaultUserName* and then right-click on the string name. Select *Modify* from the popup menu. The Edit *String* dialog box will open.

In the *Value data* text box enter `<the default user name>` and then click *OK* to accept the changes.

Highlight the *String Value* labeled *DefaultPassword* and then right-click on the string name. Select *Modify* from the popup menu. The Edit *String* dialog box will open.

In the *Value data* text box enter `<the default password>` and then click *OK* to accept the changes.

If any of the registry values are missing they will need to be added. Right-click in an empty area of the right frame and select *New > DWORD Value* or *String Value* depending on the value type (see above) from the popup menu. Enter the appropriate value name and settings as described above ensuring that all three of the registry values are present in the SLnet key.

Note: You must stop and restart SLnet before these changes will take effect.

What do I do if I want to be prompted for terminal type when I log in?

Open the Registry Editor by clicking *Start > Run* and then typing `regedit` and then clicking *OK*, or by using RegShell, shipped with SLnet. Expand *HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet*.

The *String Values* for all of the SLnet registry keys will be displayed in the right frame.

Highlight the *DWORD Value* labeled *Negotiate Terminal Type* and then right-click on the string name. Select *Modify* from the popup menu. The Edit *DWORD Value* dialog box will open.

Change the *Base* to `Decimal` and then enter `817` in the *Value data* text box. Click *OK* to accept the changes.

Note: You must stop and restart SLnet before these changes will take effect.

How do I set the console window size manually?

If you are having difficulties with your Telnet client and would like to manually set the size of the console window (especially useful for very large or small console viewing areas) you must add some registry settings referring to negotiate about window size (NAWS).

Open the registry editor by clicking *Start > Run* and then typing `regedit` and then clicking *OK*, or by using RegShell, shipped with SLnet. Expand *HKEY_LOCAL_MACHINE > SOFTWARE > Seattle Lab > SLnet > InheritedUserSettings*.

You will need to add (or modify) two registry values in this key. The two values in question are:

```
Ignore NAWS For rendering
Ignore NAWS For sub processes
```

If these registry values are not present in the key, right-click in the white space of the right frame and select *New > DWORD Value* from the popup menu. A new value will be created in the key. Change the name of the “*New Value #1*” to one of the missing value names (above). Repeat this process until both values are present.

You need to ensure that both Values are set to 1 by right clicking on the value and selecting *Modify* then changing the *Value Data* to 1 (if necessary).

By ignoring the NAWS information for rendering you force the server to look into the registry for the console window size information for the client. The Server does not rely on the NAWS information and sends the console window size to the client monitor based on the registry settings in *Terminal Cols* and *Terminal Lines*. (The Terminal Columns and Terminal Lines keys are usually kept in individual user’s registry folders.) If Terminal Cols and Terminal Lines are both set to 0 the server will look in the terminal database to find the number of terminal lines and columns based on the Terminal Type information that was passed.

By ignoring NAWS for sub processes you are forcing the Server to send the default window size that has been set in the command shell.

Note: You must stop and restart SLnet before these changes will take effect.

How do I invoke my own application instead of being sent to the command prompt when I log in?

SLnet allows you to invoke an application when you log on or at start up if you have autologon enabled. The User Configuration lets you run virtually any console application from the command shell at logon. There are only a few quick steps needed to be made to accomplish this.

From the *User* tab of the SLnet control panel, select the desired user from the list and click the *Edit* button. The User Configuration dialog box will open.

Once in the User Configuration dialog box you can add a command line command that will invoke your application. The directions for doing so follow:

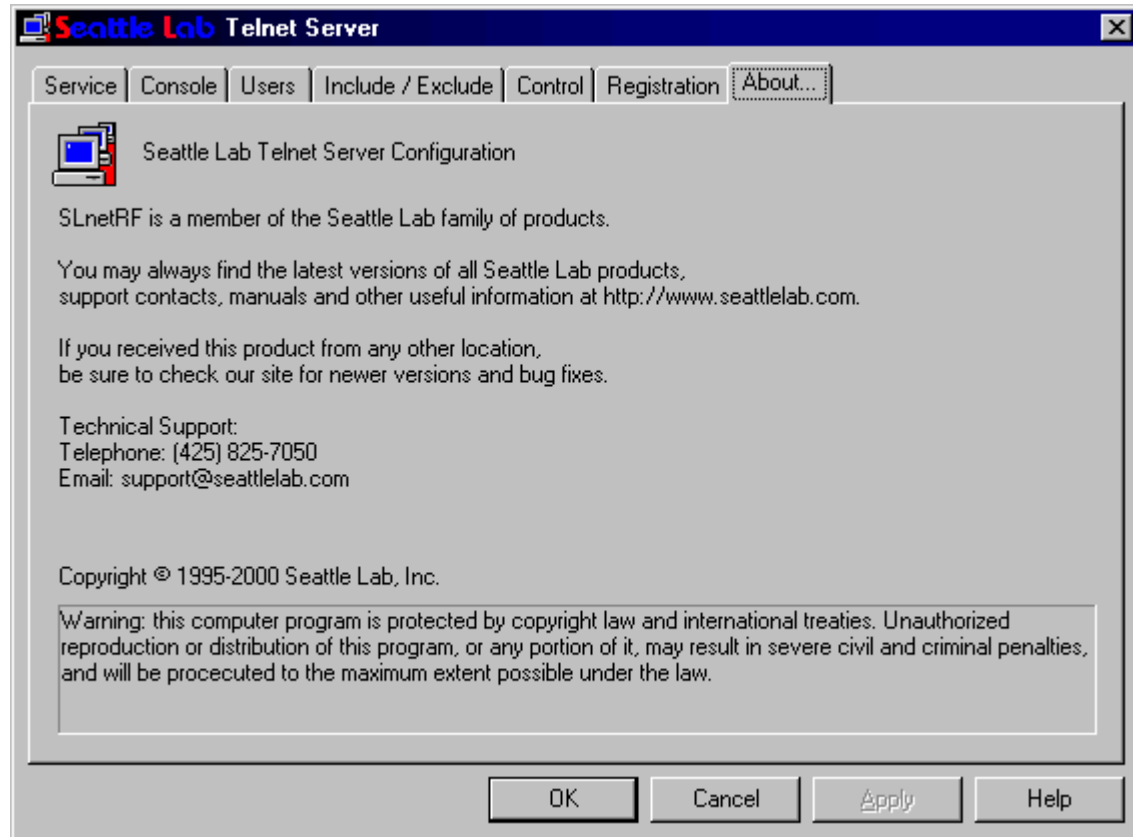
1. Do not make changes to the *Default Shell* text box should be configured to open the command shell and should be left as is.
2. The *Startup Directory* text box should be filled with the path to the directory that contains the console application that you would like to invoke.
3. The *Command Line* text box is used to enter the application that you wish to invoke. The application must reside in the directory specified in the *Startup Directory* text box. Enter the filename in the text box preceded by a switch command (`/c` or `/k`).

Note: You must include either the `/c` or `/k` switch in the *Command Line* text box for the application to run properly.

For example, if you were executing the batch file `test.bat`, which resided in the `test_2` directory in the `test` directory in the root of `C`, you would enter `C:\test\test_2` in the *Startup Directory* text box and `/c test.bat` in the *Command Line* text box. Please see the `cmd.exe` help file for additional information about invoking applications from the command shell.

About

The About tab of the SLnet Configuration Control Panel contains information about SLnet and contacting SeattleLab for Customer Care (technical support). This page also contains important copyright information.



About Tab

To determine which version of SLnet you are running, click on the SLnet icon located in the upper left-hand corner of the screen.

Hot Key Commands Table

Sequence*	Result when working in Menus
j	Move selection arrow down one line
k	Move selection arrow up one line
b	Move back one page (PgUp)
f	Move forward one page (PgDn)
+	Expand to view processes under the selected window
-	Collapse information under the selected window
TAB	Toggle collapse/expand display for selected window
BACKSPACE	Toggle collapse/expand display for selected window
Z (capital Z)	Kill a listed window or process or popup
ENTER	(When pointing to a window) close the menu and make the highlighted window the current console window
ENTER	(When pointing at a popup) locate popup under the process it is associated with
v	(When pointing at a popup) display popup location
ESC	Hide the menu and returns to selected window
q	Hide the menu and returns to selected window
CTRL-a CTRL-k u	Up
CTRL-a CTRL-k d	Down
CTRL-a CTRL-k r	Right
CTRL-a CTRL-k l	Left
CTRL-a CTRL-k p	PgUp
CTRL-a CTRL-k n	PgDn
CTRL-a h or CTRL-a H	Display Online help menu (use menu keystrokes to navigate in menu)
CTRL-a CTRL-h	Display Online help menu (use menu keystrokes to navigate in menu)
CTRL-a ? or CTRL-a /	Display Online help menu (use menu keystrokes to navigate in menu)
CTRL-a CTRL-c	Create another login process (using the same shell and parameters)
CTRL-a c	Create another login process (using the same shell and parameters)
CTRL-a 1	Switch to the first open window
...	Switch to windows 2-9
CTRL-a 0	Switch to the 10 th open window
CTRL-a -	Switch to the 11 th open window
CTRL-a =	Switch to the 12 th open window
CTRL-a CTRL-w	Display the Menu ("Windows")
CTRL-a w	Display the Menu ("Windows")
CTRL-a CTRL-d	Display the Menu ("Processes")
CTRL-a p	Display the Menu ("Desktop")
CTRL-a B	Sends CTRL+BREAK to the window***
CTRL-a C (capital C)	Sends CTRL+C to the window***
CTRL-a #	Disconnect***

*Where ellipses (...) are used, they indicate that the pattern set by the preceding simulation is followed. For example, to simulate Function Key 2, you would use the same sequence as for Function Key 1, only replacing the 1 with a 2.

**These commands affect when the SHIFT key presses and key releases are sent on to the window. They do not affect programs operating in passthrough modes, as we send these programs ASCII streams of characters, rather than keystrokes.

***WARNING: This command is bound to a shifted key. NT's Telnet client will appear to HANG if you press CTRL+SHIFT+<x>, where <x> is any key. Be sure to release CTRL (and a) before typing the next character.

SLnet–Hot Key Commands Table

Sequence* (cont.)	Result when working in Menus (cont.)
CTRL-a CTRL-r	Repaint screen
CTRL-a SPACE	Repaint screen
CTRL-a CTRL-l	Repaint screen
CTRL-a CTRL-k 4	Left
CTRL-a CTRL-k 6	Right
CTRL-a CTRL-k 7	Home
CTRL-a CTRL-k 8	Up
CTRL-a CTRL-k 9	PgUp
CTRL-a CTRL-k 0	Insert
CTRL-a CTRL-k .	Delete
CTRL-a CTRL-k t	Backtab
CTRL-a CTRL-k u	Up
CTRL-a CTRL-k d	Down
CTRL-a CTRL-k l	Left
CTRL-a CTRL-k r	Right
CTRL-a CTRL-k i	Insert
CTRL-a CTRL-k h	Home
CTRL-a CTRL-k e	End
CTRL-a CTRL-k p	PgUp (Previous Page)
CTRL-a CTRL-k n	PgDn (Next Page)
Function Keys	
CTRL-a f 1	F1
...	F2, F3, F4...
CTRL-a f 9	F9
CTRL-a f 0	F10
CTRL-a f - (minus)	F11
CTRL-a f = (equal)	F12
Shift Next Character**	
CTRL-a CTRL-x a <char>	ALT <char> (where <char> represents the character that Alt was to be applied to. EXAMPLE: for ALT F, type CTRL a CTRL x a f)
CTRL-a CTRL-x s <char>	SHIFT <char> (where <char> represents the character that SHIFT was to be applied to. EXAMPLE: for SHIFT F, type CTRL a CTRL x s f)
CTRL-a CTRL-x c <char>	CTRL <char>(where <char> represents the character that CTRL was to be applied to. EXAMPLE: for CTRL c, type CTRL a CTRL x c c)
^a^x^c ^a^x^a ^a^k.	CTRL+ALT+DEL If you set a temporary shift for Ctrl (^a^x^c), then set another temporary shift for Alt, both shifts will be pending for the next key DEL off the keypad (^a^k).

*Where ellipses (...) are used, they indicate that the pattern set by the preceding simulation is followed. For example, to simulate Function Key 2, you would use the same sequence as for Function Key 1, only replacing the 1 with a 2.

**These commands affect when the SHIFT key presses and key releases are sent on to the window. They do not affect programs operating in passthrough modes, as we send these programs ASCII streams of characters, rather than keystrokes.

***WARNING: This command is bound to a shifted key. NT's Telnet client will appear to HANG if you press CTRL+SHIFT+<x>, where <x> is any key. Be sure to release CTRL (and a) before typing the next character

Sequence (cont.)	Result when working in Menus (cont.)
Toggle Shift Lock	(applies until you type sequence again to toggle off)
CTRL a CTRL-x CTRL-x a	ALT
CTRL a CTRL-x CTRL-x s	SHIFT
CTRL a CTRL-x CTRL-x c	CTRL
Press and Release Shift Key	
CTRL-a CTRL-v <x>	Where x is a (ALT) or s (SHIFT) or c (CTRL)
CTRL-a CTRL-V S (capital V)	CAPS LOCK***
CTRL a CTRL-V C (capital C)	CAPS LOCK***
CTRL-a CTRL-V l (lower case L)	CAPS LOCK
CTRL-a CTRL-V CTRL-N	NUM LOCK
CTRL-a CTRL-V 1 (digit)	NUM LOCK

*Where ellipses (...) are used, they indicate that the pattern set by the preceding simulation is followed. For example, to simulate Function Key 2, you would use the same sequence as for Function Key 1, only replacing the 1 with a 2.

**These commands affect when the SHIFT key presses and key releases are sent on to the window. They do not affect programs operating in passthrough modes, as we send these programs ASCII streams of characters, rather than keystrokes.

***WARNING: This command is bound to a shifted key. NT's Telnet client will appear to HANG if you press CTRL+SHIFT+<x>, where <x> is any key. Be sure to release CTRL (and a) before typing the next character

Index

A

Administrators user group	1, 13
Allow Access	38
Allow Redraw	46
Arrow	68
Arrow keys	61
Asynchronous	12
Authentication	7
Authorization Code	19, 26
AuthTypes	7
Auto Scroll	45
AutoLogon	56
AutoLogonIP	58
Automatic logon	56, 77

B

Banner	30, 53
Binary Pipe	46

C

Child Processes	45
Ciphertext	10
Clear text	8
Client	9
Command Line	44
Command Shell	44
Common Command Shell	3
Connect command	53
Console beep	76
Console type	65
Console Wait Time	32
Console window	
Defined	66
Sizing	78
Control keys	61
Control Panel	41
CPU	5
Cursor keys	61

D

Decryption	10
Default Shell	44
Deny Access	39
Desktop	67

Dialog box	69
Directory	4, 44
Disconnect	67
Domain	53
Drive letters	45
Drives	
Mapping	75

E

Emulator	53, 61
Key mapping	61
Terminal type	53
Encrypt	7
Encryption	9, 10
Null	8
Error message	69, 71
Evaluate	18
Event Log	31, 70
Event Viewer	31
Exclude All	38
Exiting	60, 69

F

Function keys	61
---------------------	----

G

Greeting	30, 53
----------------	--------

H

Hosts file	53
Hot Key	66
Hot Key Commands	66

I

Idle Timeout	35
Installation	13
Internet	10
IP address	38, 53

K

Key mapping	61
Kill	66

L

legacy applications	2
License	14, 26

License Agreement	14	Security risk	10
Logging	70	Selection arrow	68
Logon	53	Serial devices	12
Banner	53	Serial Number	19, 26
Secure	9	Service	
Logon Attempts	29	Automatic startup	41
M		Disabling	41
Maximum Connections	29	Stop	40
Menu	68	Shell	54
N		Common Command	44
Named Pipe	33	SLclient32	9
Navigation in menu	68	SLcon	65
NAWS	78	SLcons	65
Nonprintable characters	45	Sockets	29
NTLM	8	Special keys	61
Null	8	Startup Directory	44
O		Status	30
Online Help	66	Status line	67
Options		Stop	40
Color	46	Switch	67
Priority	33	Switching windows	66, 67
User	44	T	
P		TCP/IP protocol	4
Passthrough Pipe Name	33	Telnet	
Password	10, 29, 54	Clients	53
Pipe Name	33	Connecting to a host	53
Polling	32	Ending a Telnet session	60
Popup	68, 69	Negotiation	53
Print	75	Port	29
Priority	33	Prompt	30
Private Desktop	30	RFC 1060	30
Privileges	1, 6	Windows NT Telnet applet	53
Process ID	68	Telnet Socket	29
Processes		Terminal beep	76
Killing	61	Terminal emulation	3
List	61	Terminal emulator	61
Monitoring Child	45	Terminal Server	12
R		Terminal type	77
Redirection chars	45	Terminals	12, 53
Registration	26	Time Slice Increment	32
S		Tlist	66
Security	6, 43	Tuning	33
		U	
		Universal Naming Convention (UNC)	45
		Unsecure Logon	9

Users

Access control	42
Adding	42
Configuring accounts	44
Domain	53
Increasing user limit	27
Limit	29
Logon	53
Privileges	6
Security	6, 43

W

Wait for Connection	30
Wide-Area Networks	2
Working Directory	44

X

XXX	68
-----------	----

Z

Z	69
---------	----