KarlBridge
&
KarlBrouter

KarlNet

Installation
&
Configuration
Software Version 2.0

This manual covers
KarlBridge and KarlBrouter
Software Version 2.0

PUBLISHED BY
KarlNet Inc.
88 East Oakland Ave.
Columbus, Ohio 43201
(614) 263-5275

FCC Statement (For U.S.A. Only)
Federal Communications Commission Radio Frequency Interference Statement

Warning: This equipment generates, uses, and can radiate radio frequency energy. If it is not installed and used in accordance with the instruction manual, it may cause interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

If this equipment causes interference to radio reception (which can be determined by unplugging the power cord from the equipment) try these measures: Re-orient the receiving antenna. Relocate the equipment with respect to the receiver. Plug the equipment and receiver into different branch circuits. Consult your dealer or an experienced technician for additional suggestions.

Software License Agreement

Introduction: It is important for Users of KarlNet Software to take time to read this License Agreement associated with this software PRIOR TO ITS USE. The End User has paid a License fee to KarlNet, Inc for the use of this software on one computer. This License does not extend to any copyrights to the program nor does it License use of the program on more than one computer no to make copies of the program for distribution or resale. A software registration card is located in the front of this manual. Please complete the card within 10 days of receipt of the software and return it to KarlNet Inc. hereafter in this License Agreement, KarlNet. Registration is required for warranty service and notification of software updates and revisions.

License Agreement: The End User is granted a non-exclusive License to use the Licensed program on a single computer subject to the terms and conditions as set forth in this agreement. The End User may not copy, modify or transfer the reference manual or other documentation or any copy thereof except as expressly provided in this agreement.

The copyright and all intellectual / industrial rights of this program and associated material remain the property of KarlNet Inc. THE END USER MAY NOT USE, COPY, SUBLICENSE, ASSIGN OR TRANSFER THE LICENSED MATERIALS OR ANY COPIES THEREOF IN WHOLE OR IN PART, EXCEPT AS EXPRESSLY PROVIDED IN THIS LICENSE AGREEMENT. The End User shall not reverse assemble or reverse compile the Licensed product or any copy thereof in whole or in part.

Upgrades and Revisions: At its sole option and discretion, KarlNet may from time to time make available for licensing to the End User, in consideration for the payment of an additional fee specified by KarlNet, future updated versions of the Licensed product. Also, at its sole discretion, KarlNet may from time to time make available for licensing to End Users, free of charge, revisions to the Licensed product.

Warranty and Liability: KarlNet warrants to the end user/purchaser, that this product will be free from defects, under normal use, in materials and workmanship under normal user and service for a period of one year from the date of original purchase. KarlNet agrees under this warranty, at its sole option, to repair, replace, or refund the purchase price of any product discovered to be defective during the warranty period. Any such replacement may be, at the sole option of KarlNet, a new or a re-manufactured product.

KarlNet has made a good-faith effort to ensure that the firewall security filters are implemented in the best way possible. The user/purchaser is solely responsible for ensuring that all firewall security filters are setup correctly and functioning correctly.

This warranty shall not apply to any product that has been modified without written approval of KarlNet, abused, misused, tampered with, damaged by other equipment or systems, or operated or stored under adverse environmental conditions.

EXCEPT AS EXPRESSLY SET FORTH ABOVE, KARLNET MAKES NO OTHER WARRANTIES OR REPRESENTATIONS, EITHER EXPRESSED OR IMPLIED (INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE). KARLNET EXPRESSLY DISCLAIMS ALL WARRANTIES NOT STATED HEREIN. YOU ASSUME THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PRODUCT. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS WHICH MAY VARY FROM STATE TO STATE.

LIMITATION OF LIABILITY

YOUR SOLE REMEDIES AND KARLNET'S AND ITS SUPPLIERS, DISTRIBUTORS, RESELLERS, AND AGENTS ENTIRE LIABILITY ARE SET FORTH ABOVE, IN NO EVENT SHALL KARLNET OR ITS SUPPLIERS, DISTRIBUTORS, RESELLERS, AND AGENTS BE LIABLE TO YOU, OR ANY OTHER PERSON, FOR ANY DAMAGES, INCLUDING ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING, LOST SAVINGS, COST OF REPLACEMENT, OR OTHER EXPENSES ARISING OUT OF THE USE OR INABILITY TO USE THIS PRODUCT, EVEN IF KARLNET HAS BEEN ADVISED OF SUCH POSSIBLE DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY. SOME STATES DO NOT ALLOW THE EXCLUSION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU.

For product returns, please call KarlNet (614) 263-5375

CONTENTS

7 APPENDIX

Introducing The KarlBridge and KarlBrouter

FEATURES AND BENEFITS

Transparent Ethernet Bridging with Advanced Filtering for Security and Network Reliability
The KarlBridge supports what is known as Transparent Ethernet Bridging with no Spanning Tree or Source Routing support. Since the KarlBridge is intended to provide network security between a local LAN and a campus or enterprise wide network, and since using multiple bridges in a Spanning Tree could compromise this security, the Spanning Tree scenario is not supported. In addition to the Transparent Ethernet Bridging the KarlBridge can drop (i.e. not forward) packets based upon the encapsulated higher layer data within the packet. It is this feature that gives the KarlBridge the ability to perform advanced firewall filtering and can add a significant measure of security and network reliability to a network surpassing that provided by modern multiprotocol routers.

Static IP Routing with Advanced Filtering for Security
The KarlBrouter supports what is known as Static IP Routing in addition to the bridging, firewall, and encryption features found on the KarlBridge. It can be used to add routing capability where an IP Router is a more appropriate choice.

Firewall Filters to Add Security to your network:
In addition to the standard Transparent Bridging and MAC layer filtering the KarlBridge has capabilities to look deep into packets and decide weather to pass or drop them. This decision is made based upon different criteria depending upon the particular protocols used. As an example for the IP protocol packets can be dropped (or blocked) that have certain IP destination addresses or are intended for certain IP server sockets, such as the SMTP server, the Telnet server, and etc. AppleTalk packets can be dropped based upon the name of the Apple printer or server that is to be used or the name of the AppleTalk Zone that the printer or server resides in. You may decide not to turn on these advanced filters in which case the KarlBridge will perform as a standard Transparent Bridge with IP/SNMP capabilities. It is recommended, however, that you set-up the KarlBridge filters to drop any protocols you don't use in order to reduce the network traffic that the Local LAN must carry. Without protocol filters a bridge will pass all Multicast and Broadcast packet into the Local LAN and each computer in the local LAN will experience a CPU interrupt for each of these packets causing it slow down.

"Tunneling" - Remote Virtual Bridging of any Ethernet protocol using an IP network as the transport mechanism
The KarlBridge can be configured to provide a virtual Ethernet connection between several LAN's using an IP network as the transport medium.

Data Encryption
The KarlBridge and KarlBrouter can be configured to provide an encrypted connection between several LAN's. These LAN's do not have to be using the IP protocol. This feature is particularly useful for companies who wish to provide connectivity by using a public network but want the security of having their data encrypted.

The KarlBridge and KarlBrouter can also be configured to provide an encrypted UDP/TCP data connection to one or more IP subnets This feature is particularly useful for companies who wish to provide encryption to their UDP/TCP connectivity when using the Internet.

FEATURE COMPARISON CHART

	Shareware / Demo Version 2.0	Commercial Version 2.0 KarlBridge	Commercial Version 2.0 KarlBrouter
HARDWARE SUPPORTED			
Standard Speed Ethernet Cards	√	√	√
Flash ROM card and Remote configuration		√	√
High Speed Ethernet Cards		√	√
WaveLAN Wireless Cards		√	√
56k / 64k Synchronous Cards (with RS232 interface)		√	√
T1 / E1 Synchronous Cards (with V.35 interface)		√	√
BRIDGING FEATURES			
Transparent Bridging	√	√	√
Filtering by Ethernet Multicast, Broadcast and Bad Packets	√	√	√
Filtering by Protocol	√	√	√
Filtering by Ethernet Address pair	L	√	√
Generic Ethernet Tunneling through IP networks		√	√
Learned Table Lock down		√	√
Expanded IP ARP Support		√	√
Automatic broadcast storm protection and notification	L	√	√
SNMP FEATURES			
IP "ping" Support	√	√	√
IP SNMP Support(MIB 2, Ethernet, Interface, SNMP and Bridge MIB)	√	√	√
IP SNMP WaveLAN and RS232 MIB Support		√	√
IP SNMP Trap Support	L	√	√
SNMP Access Lists	L	√	√
FIREWALL "SECURITY" FEATURES			
IP Net/Subnet/Host Filtering	L	√	√
Apple Zone, Server & Printer Filtering	L	√	√
Novell Network Number, Server & Service Filtering	L	√	√
DECNET Network Number and Object Filtering	L	√	√
Firewall Break-In attempt Logging		√	√
Optional IP Source Routed Packet Filtering		√	√
Optional IP Multicast Packet Filtering		√	√
Optional Suspicious IP Packet Filtering		√	√
Sending of ICMP Destination Unreachable Messages		√	√
Sending of TCP Reset Messages		√	√
Firewall Authentication Feature		√	√
IP ROUTER FEATURES (ADD-ON OPTION)			
IP Static Routing with Direct and Static Routes			√
ICMP messages, Default Router and Subnet support			√
SNMP Support for all router related MIB variables			√
ENCRYPTION FEATURES (ADD-ON OPTION)			
Data Encryption on Tunneled packets		√	√
Data Encryption on IP packets		√	√

Floppy Based Systems

LAN Light

The KarlBridge/KarlBrouter LAN light will blink whenever a packet it forwarded. It will also blink once per second just after a reboot and successful self test and before any packets have been seen on any port.

Reset Button

The KarlBridge/KarlBrouter can be hardware reset by pressing the "Reset" button on the front panel.

Floppy Exercising

The KarlBridge/KarlBrouter has several built in features for reliability. The bridge/router is designed to operate in a dusty high temperature environment for several months without problems. Since the floppy drive is a mechanical device it is susceptible to heat, dust, and humidity. When the KarlBridge/KarlBrouter is assembled great care is taken to ensure that the airflow across the floppy drive in the bridge is reduced. Also the software will turn on the floppy motor each hour to exercise the floppy drive so that it will be less likely to fail during a reboot.

Backing up the Bootable Floppy Disk

The KarlBridge/KarlBrouter "bootable" floppy has a special boot block on it that loads and runs the file "KBRIDGE.BIN". This floppy can be backed up to an identical disk by use of the standard DISKCOPY program. You must be sure to FORMAT the target disk and check to be sure that the FORMAT procedure did not discover any bad blocks before using the DISKCOPY program.

Example:

```
> Format     a: /u
> Diskcopy   b: a:
```

Flash ROM Based Systems

Flash Rom based systems are superior to floppy based systems. The Flash Rom can be reconfigured remotely via the network.

LAN Light: The KarlBridge/KarlBrouter LAN light will blink whenever a packet it forwarded. It will also blink once per second just after a reboot and successful self test and before any packets have been seen on either Ethernet port. This light will also blink rapidly and brightly when the remote configuration protections have been overridden. See "Reset Button" below.

Reset Button to Override Remote Configuration Protections: The Flash ROM version of the KarlBridge/KarlBrouter can be remotely configured through the network by use of the KBCONFIG program. The SNMP passwords must be correctly specified for KBCONFIG to remotely read and write a configuration. If the KarlBridge/KarlBrouter Flash ROM card is setup for Read Protection or Read/Write Protection then remote configuration cannot take place even if the passwords are specified correctly unless the hardware protection is disabled. You can disable the hardware protection by pressing the Reset Button located on the front panel of the KarlBridge/KarlBrouter. To indicate that hardware protection is disabled the LAN Light will blink rapidly and brightly. Hardware protection can be enabled by either pressing the Reset Button again or waiting until the automatic time out occurs, which is typically 15 minutes.

NOTE: If the Flash ROM is set to Normal mode then pressing this button will do nothing.

KarlNet

Quick Installation

Ethernet to Ethernet KarlBridge

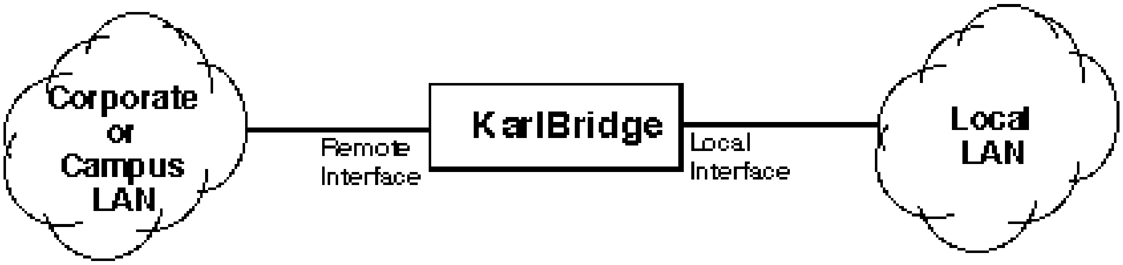
For those who wish to use the KarlBridge as a simple Thin Wire Ethernet to Thin Wire Ethernet Transparent bridge:

1) Connect up the network segments to each of the Thin Wire Ethernet ports.

2) Optionally set-up the correct 115/230 Volt power setting (most models have auto switch power supplies and need not be setup).

3) Install the KarlBridge Bootable Floppy in the floppy drive (Floppy models only).

4) Plug the bridge in.



NOTE: If you wish to change the Ethernet Port on the KarlBridge or KarlBrouter from Thin Wire Ethernet (10Base2) to either Twisted Pair (10BaseT) or AUI you must remove the cover on the KarlBridge or KarlBrouter and change the jumper on the appropriate Ethernet card.

Wireless KarlBridge

For those who wish to use the KarlBridge as a simple Thin Wire Ethernet to WaveLAN wireless transparent bridge:

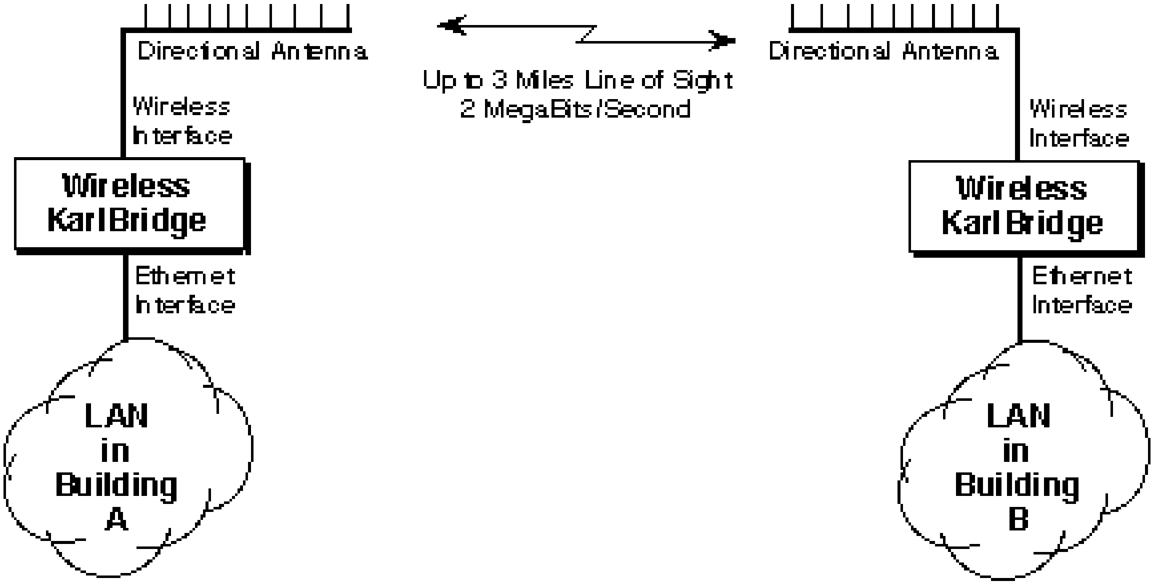
1) Connect up the Ethernet network segment to the Thin Wire Ethernet port.

2) Install the antennas and connect them up the WaveLAN port. on each Wireless bridge.

3) Optionally set-up the correct 115/230 Volt power setting (most models have auto switch power supplies and need not be setup).

4) Install the KarlBridge Bootable Floppy in the floppy drive (Floppy models only).

5) Plug the bridges in.

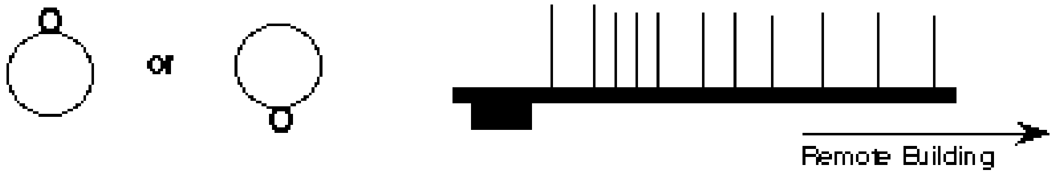


NOTE: If you wish to change the Ethernet Port on the KarlBridge or KarlBrouter from Thin Wire Ethernet (10Base2) to either Twisted Pair (10BaseT) or AUI you must remove the cover on the KarlBridge or KarlBrouter and change the jumper on the appropriate Ethernet card.

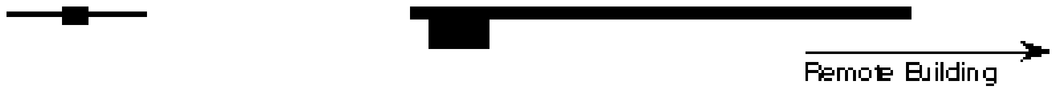
Antenna Alignment and Polarization

The Yagi type of directional antenna must be aimed so that when you look down the main barrel (shaft) of the antenna it is pointing toward the receiveing antenna on the other building. You do not have to be percise in this aiming. The radio signal shoots out the end of the antenna like a wide beamed flash light. As a general rule of thumb the 3 foot Yagi antennas have aproximatly 30 degrees of beam width. The 6 foot Yagi antennas have a slightly narrower beam width.

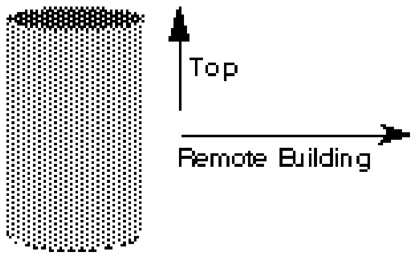
For most applications we have found that horizontally polarized antennas work best. This is because most other signals that may cause interference are vertically polarized and if you use horizontal polarization you will have better immunity to those other signals. The loop Yagi antenna is horizontally polarized when the loops are either mounted upward or downward, the standard Yagi antenna is horizontally polarized when its elements are horizontal. Following are views of horizontally polarized antennas .



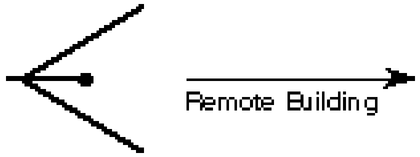
Horizontally Polarized Loop Yagi Antenna



Horizontally Polarized Standard Yagi Antenna



Horizontally Polarized
Omni-Directional Antenna

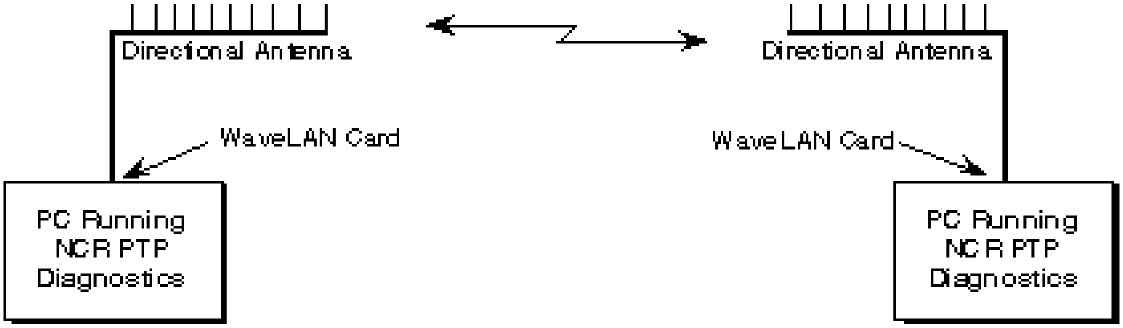


Horizontally Polarized
Corner Reflector Antenna

Running The NCR Point-to-Point Diagnostic Program

If you have trouble establishing a building to builting wireless link you may need to run the NCR Point-to-Point diagnostic program to more accurately aim the directional antennas. In order to perform this procedure you must remove the WaveLAN cards from the Wireless KarlBridge and install them in a standard PC. You then must run the program PTPDIAG.EXE which is included on the KarlBridge/KarlBrouter distribution disk. The program will display on the VGA screen a bar graph of the signal quality, signal strength and signal to noise ratio. You must run the program on each of the two WaveLAN card equiped computers.

WARNING: Do not run the PDPDIAG.EXE program in a computer that contains an Intel Etherexpress card. It will falsely determine that the Intel card is a WaveLAN card and will attempt to initialize it. This will result in your Intel card becomming unusable.



For most installations with the point to point distances under 2 miles you will simply need to install the directional antennas on the KarlBridge and will not need to fine tune the antennas by use of this diagnostic. With installations where the distance is over 2 miles, obstructed by trees, or where the antennas are mounted indoors you should use this diagnostic to fine tune the antennas.

The diagnostic will report several statistics. The key factor to look for on the diagnostic menu is the percentage of successful transmits and receives. That number should be in the 99% to 100% range. The Signal Quality bar graph displays the amount of multipath reflections and is almost always in the near 100% range. There are few reflections in out-door settings. The Signal Level bar displays the signal level and the Signal to Noise Ratio bar displays the signal to noise ratio. You should attempt to adjust the antennas to maximize the signal level and to minimize the signal to noise ratio.

Synchronous KarlBridge

For those who wish to use the KarlBridge as a simple Thin Wire Ethernet to 56K, 64K, T1 or E1 Transparent bridge:

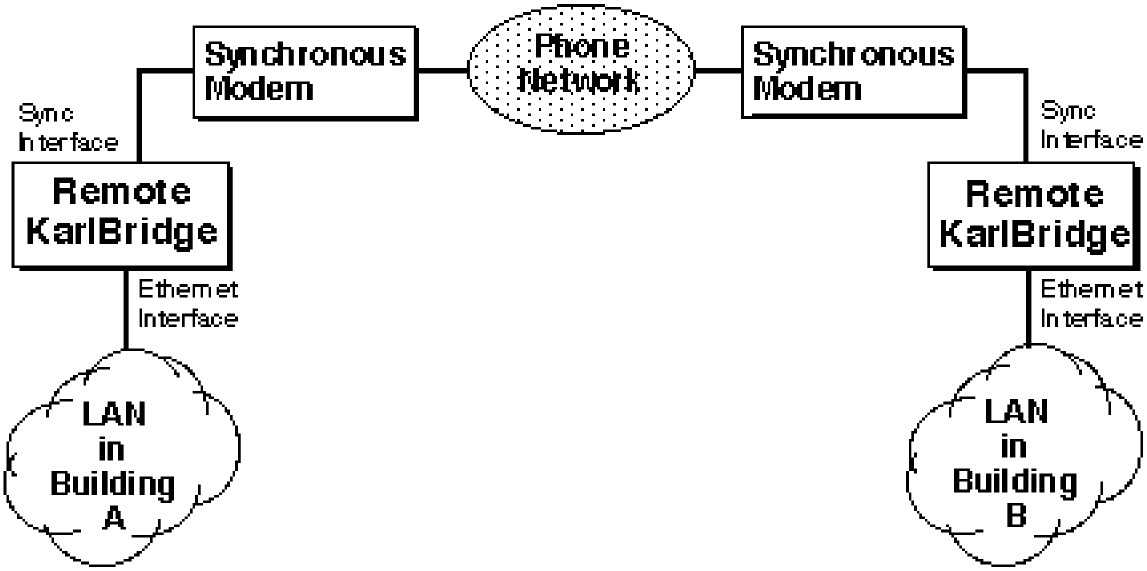
1) Connect up the Ethernet network segment to the Thin Wire Ethernet port.

2) Connect up the synchronous RS232 or V.35 cable from ths CSU/DSU to the synchronous port 0.

3) Optionally set-up the correct 115/230 Volt power setting (most models have auto switch power supplies and need not be setup).

4) Install the KarlBridge Bootable Floppy in the floppy drive (Floppy models only).

5) Plug the bridge in.



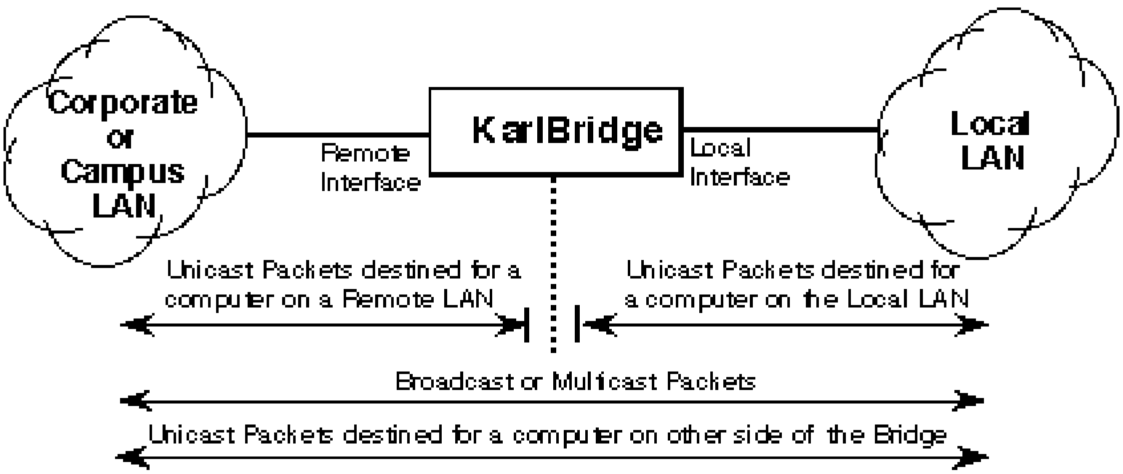
NOTE: If you wish to change the Ethernet Port on the KarlBridge or KarlBrouter from Thine Wire Ethernet (10Base2) to either Twisted Pair (10BaseT) or AUI you must remove the cover on the KarlBridge or KarlBrouter and change the jumper on the appropriate Ethernet card.

Suggested Applications

SUGGESTED APPLICATIONS AND EXAMPLES

The following section highlights several suggested applications of the KarlBridge and also the KarlBrouter. Some of these applications require the Data Encryption option. These examples are not all inclusive and can be customized and combined to fit your particular needs.

Local (Ethernet to Ethernet) Bridging


Corporate or Campus LAN
KarlBridge
Remote Interface
Local Interface
Local LAN
Unicast Packets destined for a computer on a Remote LAN
Unicast Packets destined for a computer on the Local LAN
Broadcast or Multicast Packets
Unicast Packets destined for a computer on other side of the Bridge

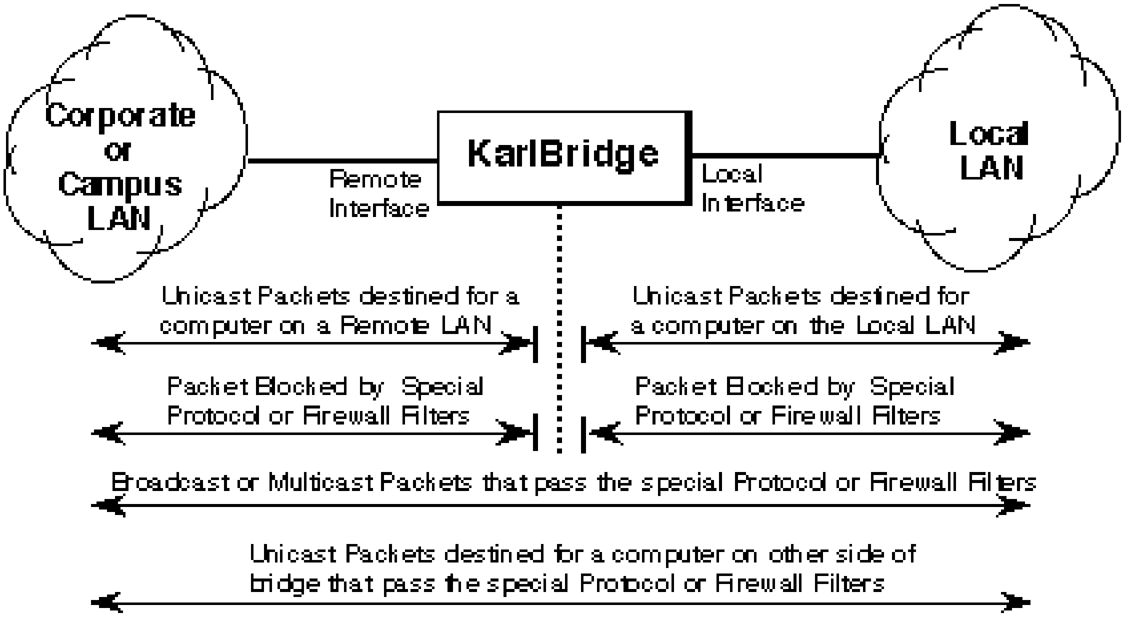
The KarlBridge supports what is known as transparent Ethernet bridging. with no Spanning Tree or Source Routing support.

It can filter and forward packets at the full Ethernet packet rate. It is an 802.1d compliant transparent learning bridge. This is the most popular type of bridge. It builds a table of Ethernet addresses as it learns which Ethernet interfaces are connected to each of its ports. Once the bridge learns that a particular Ethernet interface (computer) is on one of its ports then it will route Ethernet packets appropriately.

The transparent learning bridge will forward (pass) Ethernet packets of any protocol. (i.e. Novell IPX, Apple Talk, IP, LAN Manager, etc.) An Ethernet packet destined for a specific Ethernet address (i.e. a Unicast packet) will not be forwarded by the bridge if it can reach its destination without the bridge forwarding it. Broadcast and Multicast packets are forwarded unconditionally.

Since the KarlBridge is intended to provide network security between a local LAN and a campus or enterprise wide network, and since using multiple bridges in a Spanning Tree could compromise this security, the Spanning Tree scenario is not supported.

Local (Ethernet to Ethernet) Bridging with Special Protocol or Firewall Filters

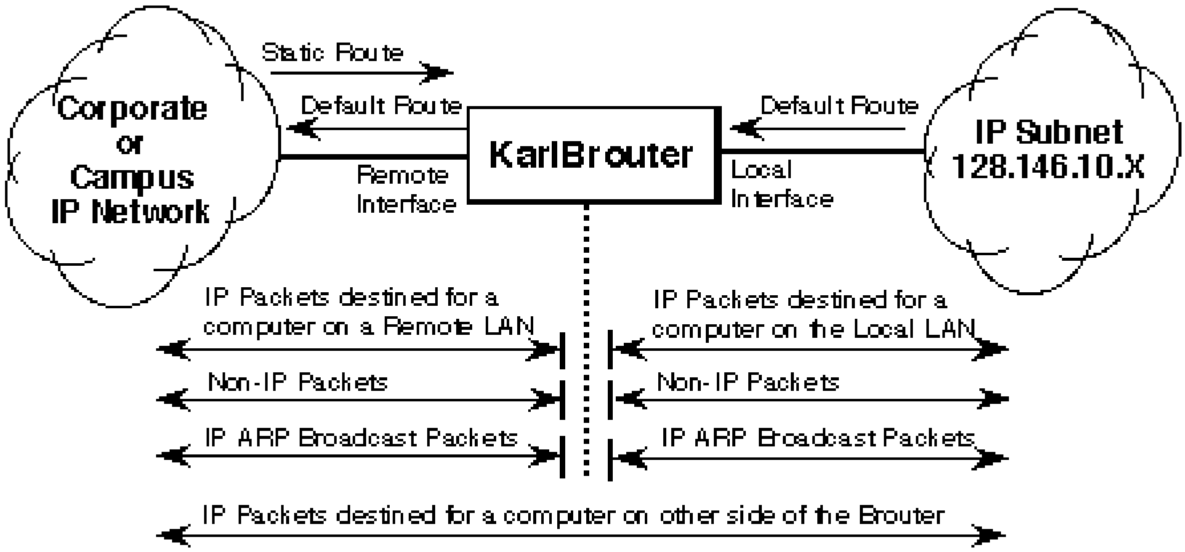

Corporate or Campus LAN
KarlBridge
Remote Interface
Local Interface
Local LAN
Unicast Packets destined for a computer on a Remote LAN
Unicast Packets destined for a computer on the Local LAN
Packet Blocked by Special Protocol or Firewall Filters
Packet Blocked by Special Protocol or Firewall Filters
Broadcast or Multicast Packets that pass the special Protocol or Firewall Filters
Unicast Packets destined for a computer on other side of bridge that pass the special Protocol or Firewall Filters

In addition to transparent Ethernet Bridging (as described on the pervious page) the KarlBridge can drop (i.e. not forward) packets based upon the encapsulated higher layer protocol and data within the packet. It is this feature that gives the KarlBridge the ability to perform advanced protocol and firewall filtering. This feature can add a significant measure of security and reliability to a network, surpassing that provided by modern multiprotocol routers.

The KarlBridge has capabilities to look deep into packets and decide weather to pass or drop them. This decision is made based upon different criteria depending upon the particular protocols used. As an example for the IP protocol; packets can be dropped (or blocked) that have certain IP destination addresses or are intended for certain IP server sockets, such as the SMTP server, the Telnet server, and etc. AppleTalk packets can be dropped based upon the name of the Apple printer or server that is to be used or the name of the AppleTalk Zone that the printer or server resides in. You may decide not to turn on these advanced filters in which case the KarlBridge will perform as a standard Transparent Bridge with IP/SNMP capabilities. It is recommended, however, that you set-up the KarlBridge filters to drop any protocols you don't use in order to reduce the network traffic.

NOTE: Without advanced filters a standard Ethernet bridge will pass all Multicast and Broadcast packet into the Local LAN and each computer in the local LAN will experience a CPU interrupt for each of these packets causing it slow down while it is processing these packets.

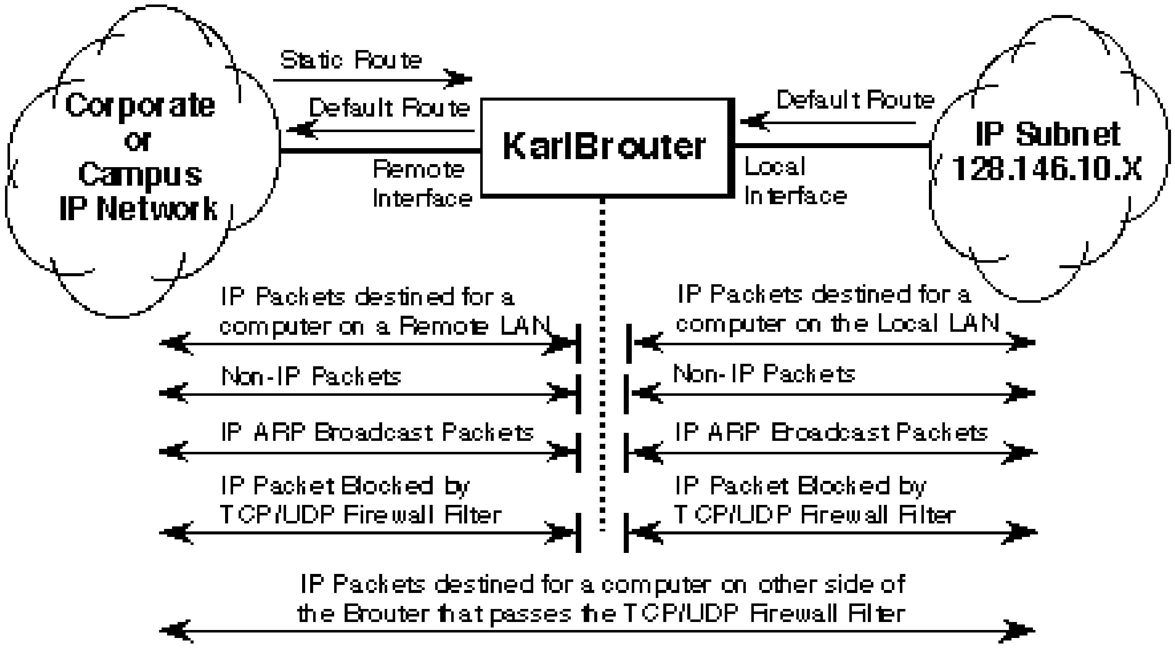
IP Router (Ethernet to Ethernet)



The KarlBrouter supports what is known as Static IP Routing in addition to the bridging, firewall, and encryption features found on the KarlBridge. All of the KarlBridge bridging, firewall, SNMP, and Encryption features are included in the KarlBrouter. The KarlBrouter simply adds IP routing functionality as described in the various applicable RFC's.

In the above example the KarlBrouter is setup to provide IP network connectivity to a Class B subnetted network. Since the KarlBrouter is a static router it's default route must be setup to point to the next IP router upstream from it. This upstream router must have a Static Route set on it to point to the 128.146.10.0 subnet via the IP address of the KarlBrouter. For those who wish to use the RIP protocol to automatically setup these routes the next version of the KarlBrouter will support RIP with the addition of RIP Access Control Lists for added security.

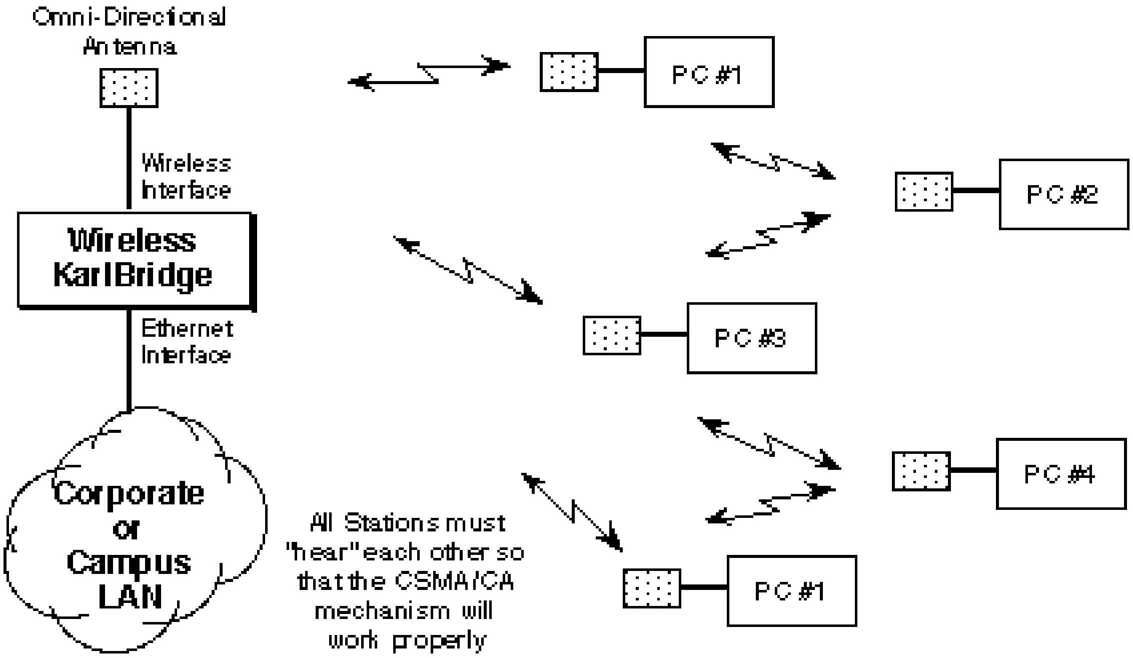
IP Router (Ethernet to Ethernet) with Special IP-TDP/UCP Firewall Filters


Static Route
Default Route
Corporate or Campus IP Network
KarlBrouter
Remote Interface
Default Route
Local Interface
IP Subnet 128.146.10.X
IP Packets destined for a computer on a Remote LAN
IP Packets destined for a computer on the Local LAN
Non-IP Packets
Non-IP Packets
IP ARP Broadcast Packets
IP ARP Broadcast Packets
IP Packet Blocked by TCP/UDP Firewall Filter
IP Packet Blocked by TCP/UDP Firewall Filter
IP Packets destined for a computer on other side of the Brouter that passes the TCP/UDP Firewall Filter

The KarlBrouter supports what is known as Static IP Routing in addition to the bridging, firewall, and encryption features found on the KarlBridge. All of the KarlBridge bridging, firewall, SNMP, and Encryption features are included in the KarlBrouter. The KarlBrouter simply adds IP routing functionality as described in the various applicable RFC's.

The above IP routing example is setup the same as the IP Router on the previous page. In addition there are firewall filters setup to drop (not route) certain IP packets. This decision is made based upon different criteria depending upon the particular protocols used. As an example for the IP protocol packets can be dropped (or blocked) that have certain IP destination addresses or are intended for certain IP server sockets, such as the SMTP server, the Telnet server, and etc.

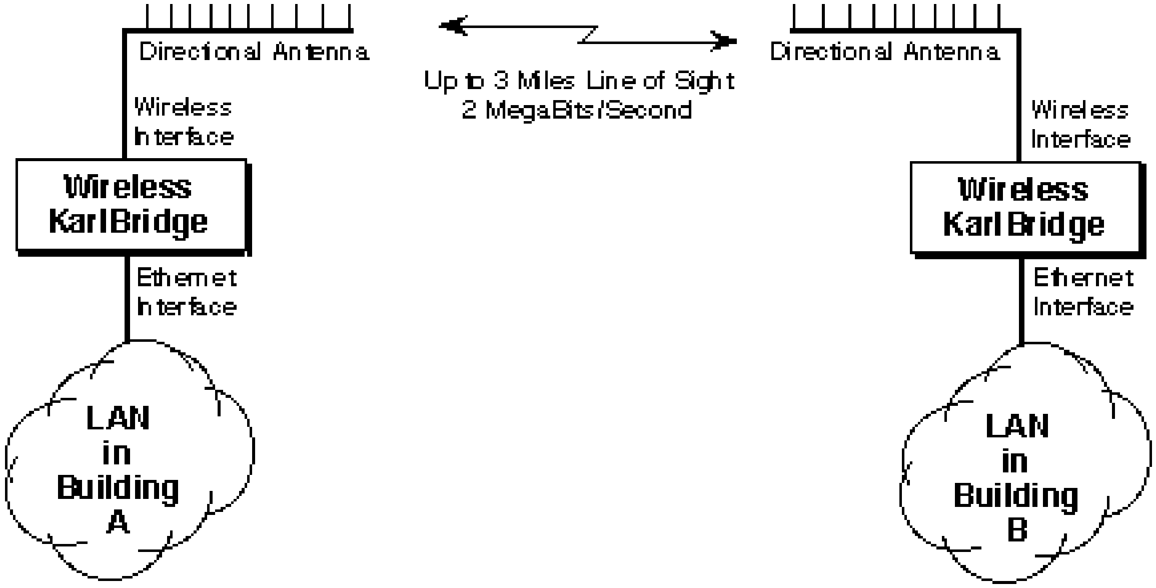
Wireless Bridge/Router/Hub (In Building)



The Wireless KarlBridge/KarlBrouter supports a standard ATT/NCR or DEC WaveLAN wireless interface card. This is the same card sold by several ATT/NCR resellers such as Solectek and Persoft in the USA and others around the world. The Wireless KarlBridge/KarlBrouter provides a wireless link to other WaveLAN wireless cards (stations) within a building. The Omni directional antenna supplied with the WaveLAN card has a range of 800 feet. With the addition of a directional antenna connections can be made between buildings that are up to several miles apart.

The one main restriction with this type of installation is that all the WaveLAN cards must be able to communicate with all other WaveLAN cards in the network. This type of wireless WaveLAN setup is compatible with all other WaveLAN based bridges available including ATT/NCR’s WavePoint bridge product. The disadvantage of this approach (along with all other wireless LANs) is that the signaling used is CSMA/CA not the more reliable CSMA/CD approach. This will result in the well know problem of packet loss on wireless LANs. This packet loss is not noticed on the traditional Novell IPX network but results in slow performance on most all other network operating systems and protocols.

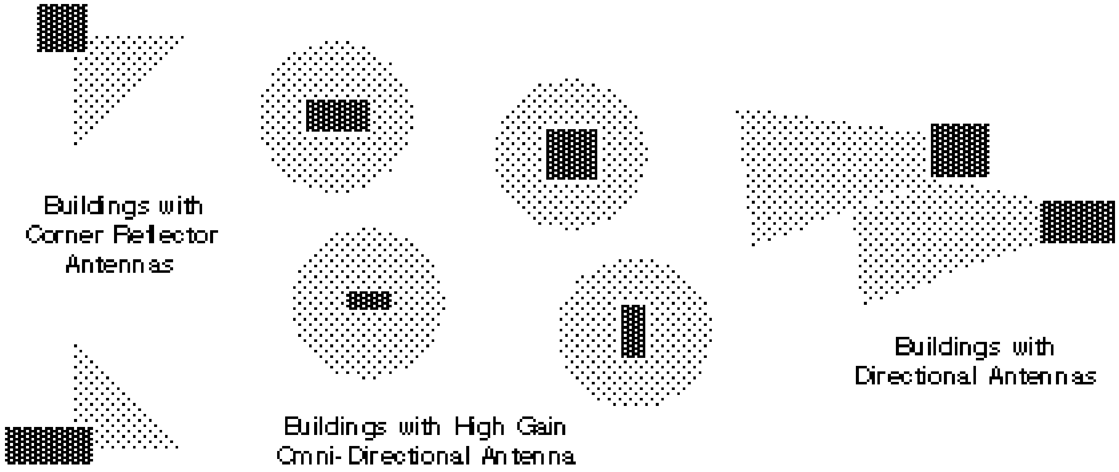
Wireless Bridge/Router (Building to Building)



The Wireless KarlBridge/KarlBrouter supports a standard ATT/NCR or DEC WaveLAN wireless interface card. This is the same card sold by several ATT/NCR resellers such as Solectek and Persoft in the USA and others around the world. With the addition of a directional antenna the Wireless KarlBridge/KarlBrouter provides a wireless link between buildings up to 3 miles apart. If a protocol other then Novell's IPX is used between building then the KarlNet pioneered reliable data communication algorithm "CellWave" is required.

Wireless Bridge/Router (Between Multiple Buildings, No Base Station)

Where all buildings can "hear" each other

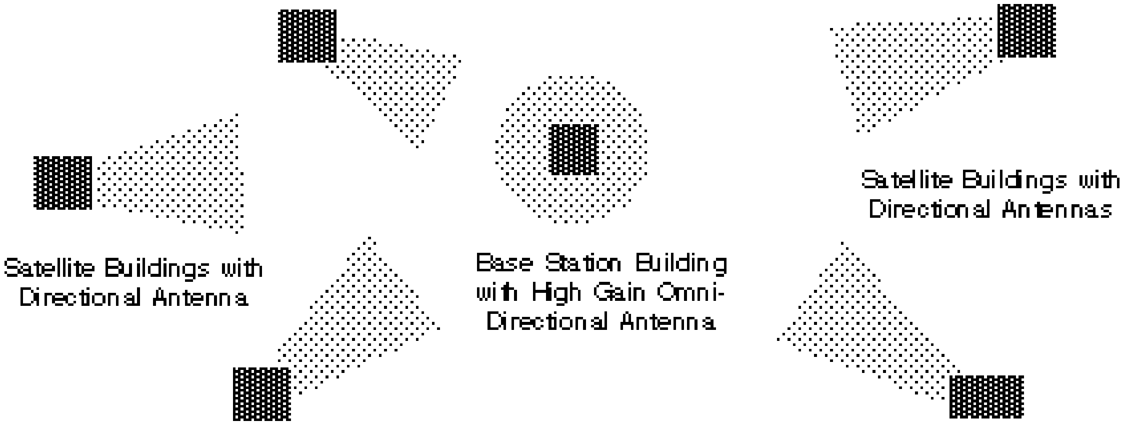


The KarlBridge/KarlBrouter has a special KarlNet pioneered reliable data communication algorithm to allow multiple buildings to network together properly. This algorithm is called CellWave and it eliminates the problems associated with wireless packet loss. One CellWave mode of operation is where all wireless buildings can communicate with each other.

The industry compatible way of transmitting and receiving data over WaveLAN (and many other) wireless networks cause data packets to be frequently lost. This is due to the fact that a wireless network does not have the ability to detect collisions like an Ethernet network. In an Ethernet network collisions can be detected by the hardware (Ethernet chip) and are automatically retransmitted. Ethernet is referred to as CSMA/CD (Carrier Sense Multiple Access with Collision Detect). Wireless networks are CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). You cannot detect collisions with radio because you cannot receive and transmit at the same time hence you cannot detect the collisions. In practice a properly operating WaveLAN point-to-point network will loose approximately 1% of the transmitted packets due to collisions. This packet loss is not noticed with Novell IPX protocol (without the burst mode NLM) but will cause networks using most other protocols to experience poor performance.

Wireless Bridge/Router
(Between Multiple Buildings, With Base Station)

Where Each Satalite Building must "hear" the
Base Station but need not "hear" each other.

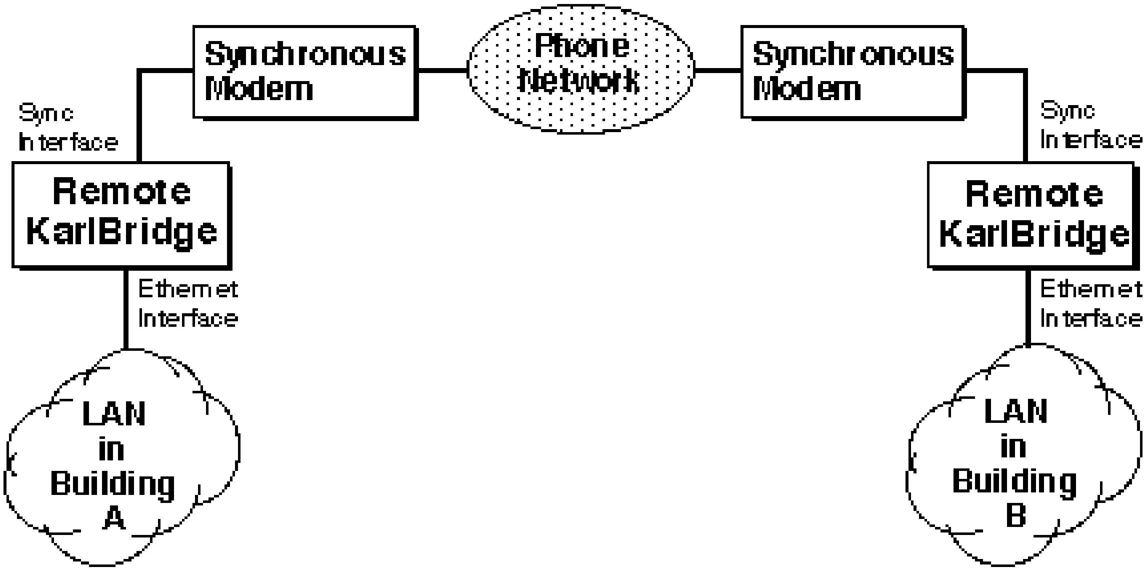


The KarlBridge/KarlBrouter has a special KarlNet pioneered reliable data communication algorithm to allow multiple buildings to network together properly. This algorithm is called CellWave and it eliminates the problems associated with wireless packet loss. One CellWave mode of operation is where all wireless buildings can communicate with a base station but not necessarily each other.

With the previously mentioned CellWave Mode (No Base Station) setting there is a requirement that all wireless stations be able to transmit to and receive from ALL other stations in the wireless network. This is not always possible due to the particular topology and terrain. The Wireless KarlBridge/KarlBrouter has a special mode where one of the wireless nodes can be setup as a “base” station and all others can be setup as “satellite” stations. In this configuration the only requirement is that each satellite station be able to communicate with the one base station. The base station is responsible for “repeating” packets that need to travel between satellite stations.

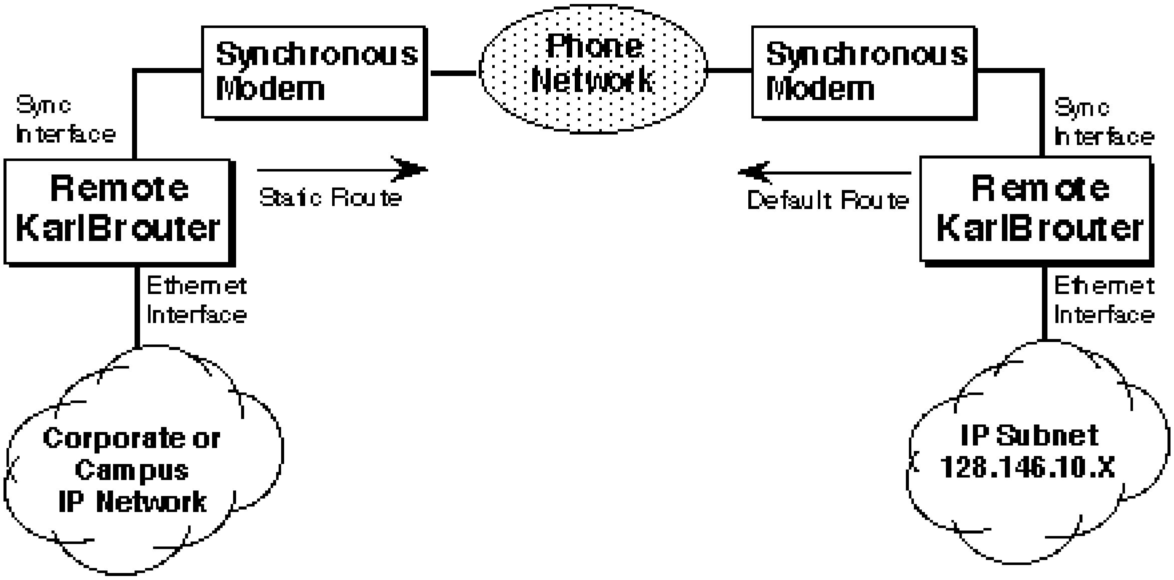
The performance of this approach is slightly improved if the base station is connected to the most heavily loaded file server or wired network access point. This is due to the fact that data flowing from one satellite to another satellite station must be repeated (retransmitted) by the base station using more of the wireless bandwidth. Data packets flowing from a satellite station to the base station are transmitted directly without the need to be repeated.

Remote Bridge
(Synchronous 56k, 64k, T1, E1)



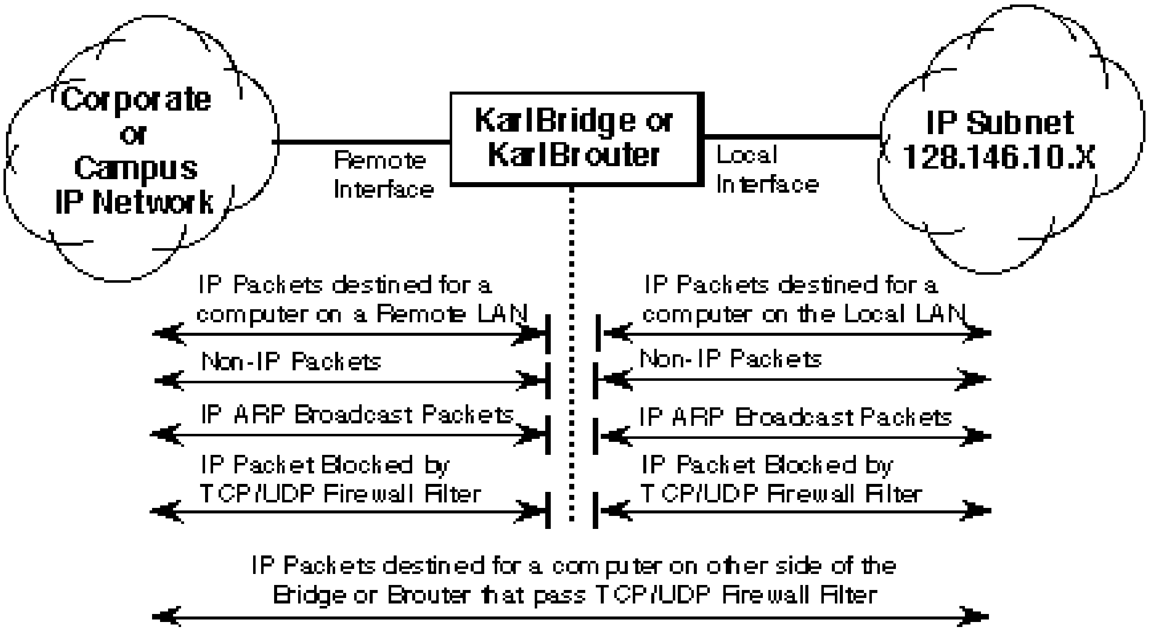
The KarlBridge can be used in conjunction with a synchronous modem or CSU/DSU to provide a remote connection between networks. The KarlBridge's synchronous interface card uses the industry standard HDLC framing and will transfer data at any rate up to 7 Mbits/second. This makes it very versatile and compatible with any kind of synchronous modem.

Remote IP Router
(Synchronous 56k, 64k, T1, E1)



The KarlBrouter can be used in conjunction with a synchronous modem or CSU/DSU to provide a remote connection between networks. The KarlBrouter's synchronous interface card uses the industry standard SLIP over HDLC framing and will transfer data at any rate up to 7 Mbits/second. This makes it very versatile and compatible with any kind of synchronous modem.

IP-TCP/UDP Firewall

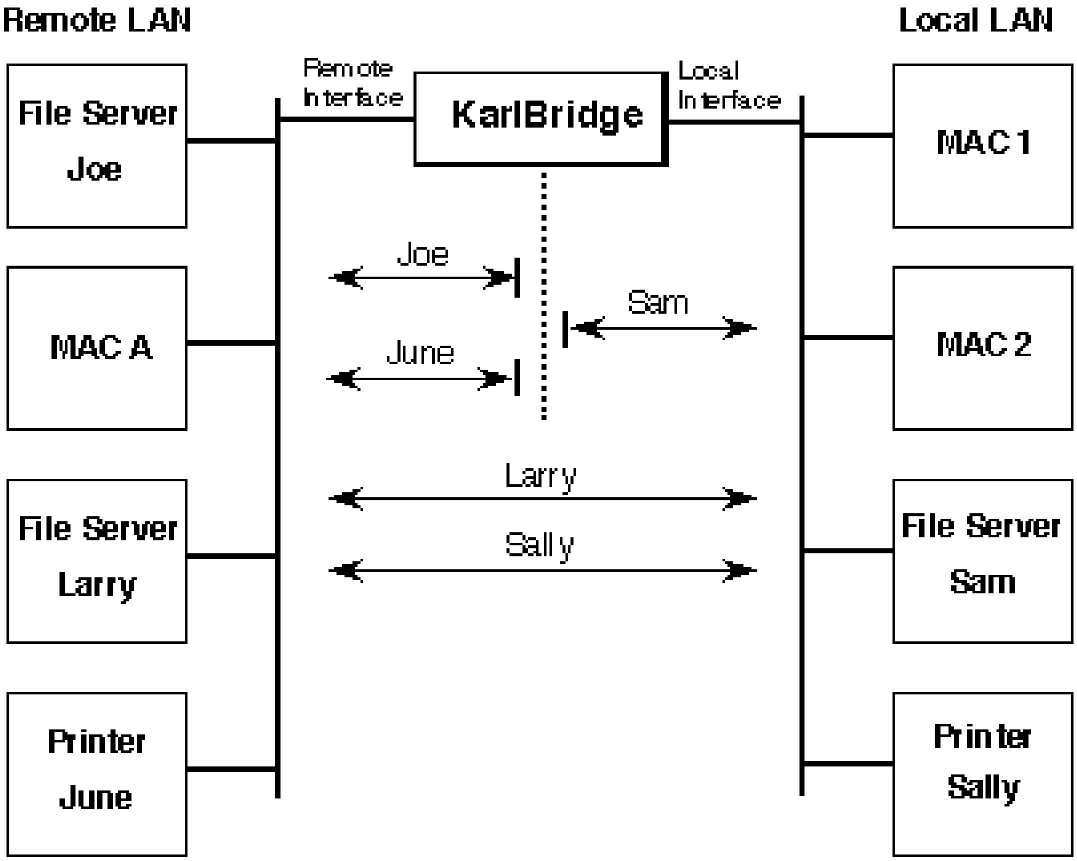

Corporate or Campus IP Network
KarlBridge or KarlBrouter
IP Subnet 128.146.10.X
Remote Interface
Local Interface
IP Packets destined for a computer on a Remote LAN
IP Packets destined for a computer on the Local LAN
Non-IP Packets
Non-IP Packets
IP ARP Broadcast Packets
IP ARP Broadcast Packets
IP Packet Blocked by TCP/UDP Firewall Filter
IP Packet Blocked by TCP/UDP Firewall Filter
IP Packets destined for a computer on other side of the Bridge or Brouter that pass TCP/UDP Firewall Filter

The KarlBridge and KarlBrouter contains special firewall filters and algorithms that can protect IP networks from intrusion by hackers and misconfigured computers. This is an extremely powerful feature that is unprecedented in the industry. You can setup filters to drop (or pass) packets that match certain criteria. As an example, you can setup the filters to allow Telnet and FTP connections out of your business while blocking them from coming back into your business from the Internet. You can allow e-mail to certain trusted machines but not others. You can also setup other filters to hide certain computers from the Internet so that no one outside your organization can communicate with those computers at all.

The KarlBridge or KarlBrouter can also perform extensive logging of network traffic. They can log all the connections made to your computers from the Internet and break-in attempts. The logging records are sent to any computer of your choice that has UNIX SYSLOG capability.

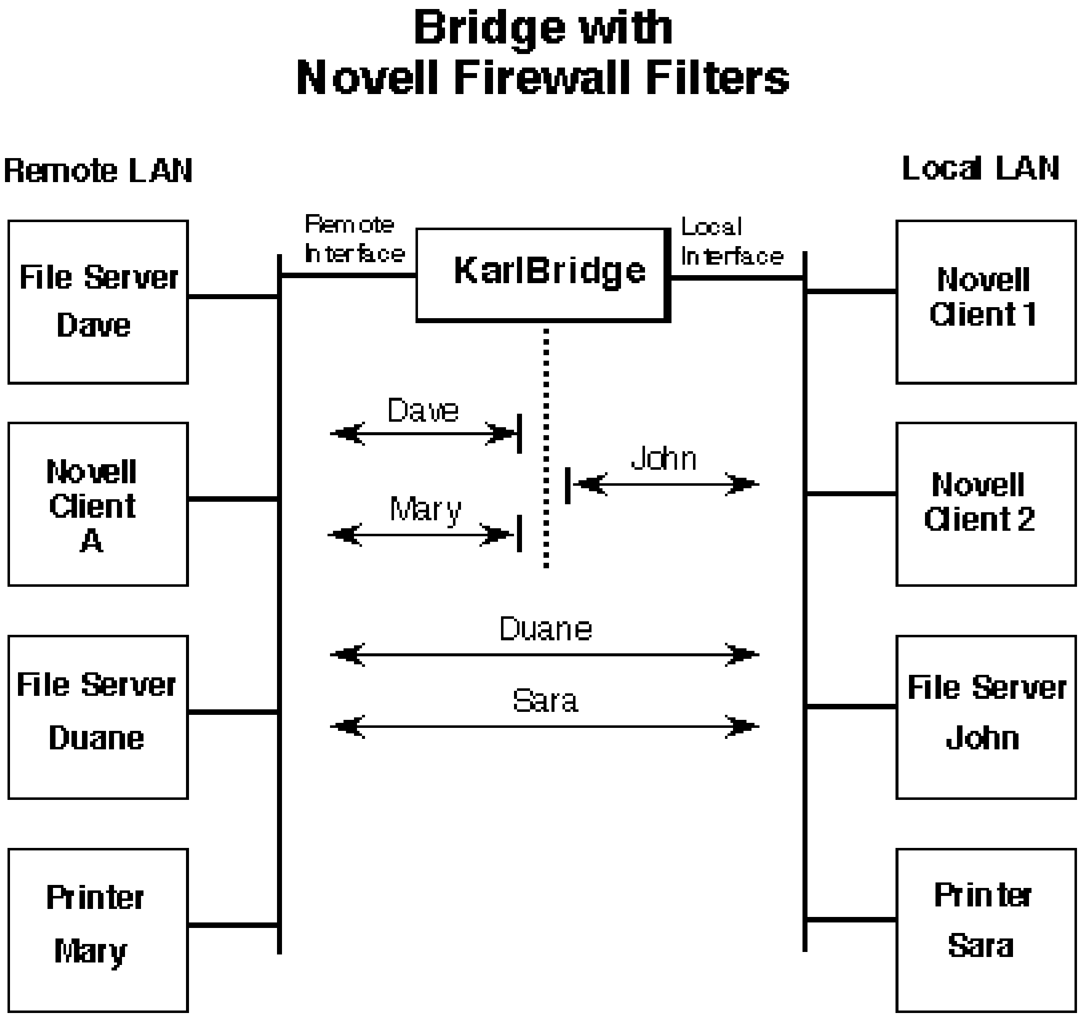
There is also a KarlBridge/KarlBrouter exclusive feature, described in a following suggested application, that will add authentication capability to the firewall. Firewall authentication is a way of dynamically punching a hole through the firewall on a case by case basis. This can be controlled by a computer setup to be an Authentication Server.

Bridge with Apple Talk Firewall Filters



When Macintosh's are networked together, one of the undesirable side effects is that all Macintosh's can "see" in their Choosers all servers and all printers that are connected to the network. If multiple zones are specified then there is some form of protection but a user needs to only specify a zone and then can choose a printer to print to anywhere in the network. The KarlBridge can be configured to selectively restrict access to specified Apple servers and/or Apple printers. The KarlBridge is not an AppleTalk router. It does not have any of the characteristics of an AppleTalk router. The KarlBridge is simply a bridge that for AppleTalk can promote or prohibit the appearance of server and/or printer names in the Chooser.

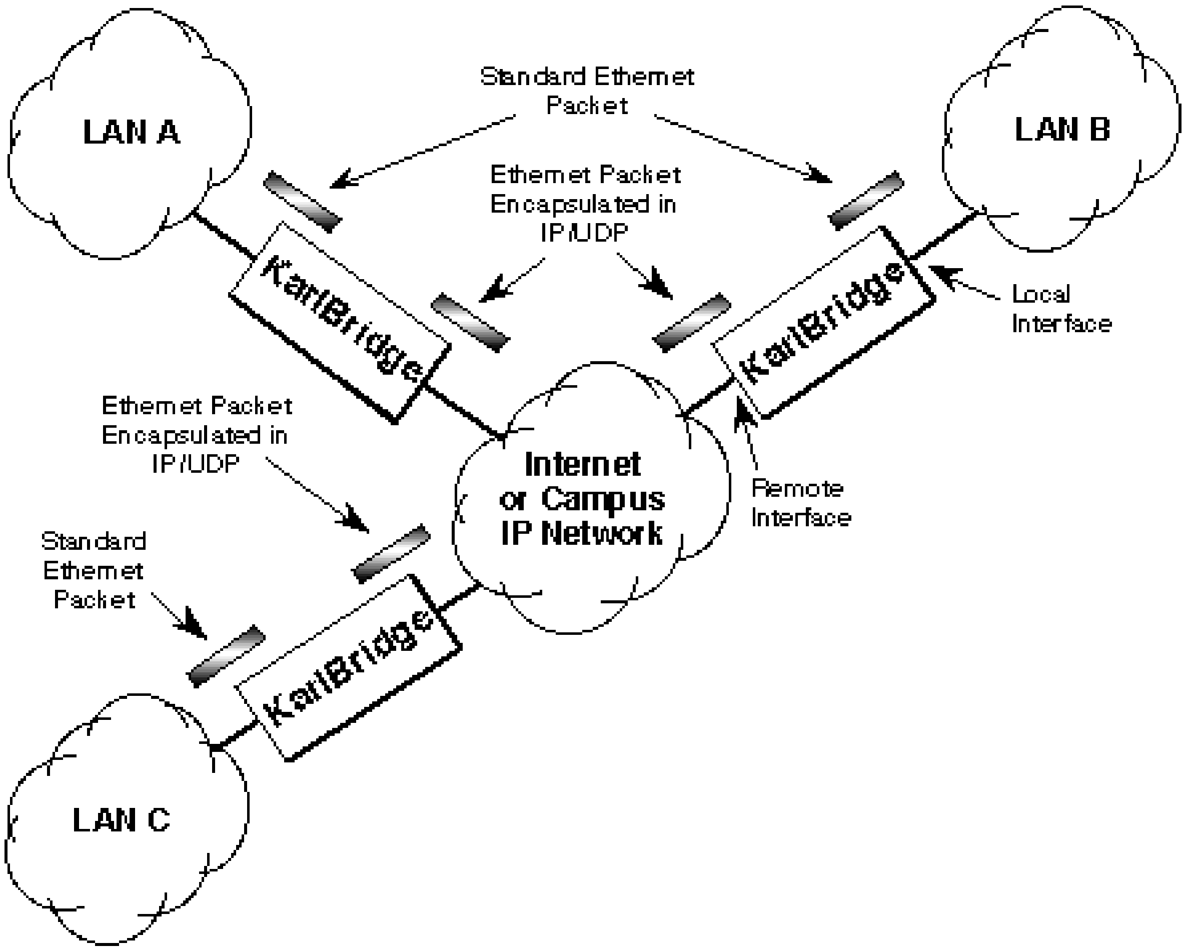
In the above example the KarlBridge has been configured so that the File Server *Joe* and the Apple Printer *June* cannot be seen by any computers on the local side of the KarlBridge. Also the file server *Sam* cannot be seen on the remote side of the KarlBridge.



When Novell servers an clients are networked together, one of the undesirable side effects is that all Novell servers and clients can “see” all servers and printers that are connected to the network.   The KarlBridge can be configured to selectively restrict access to specified Novell servers and/or printers. The KarlBridge is not an IPX router. The KarlBridge is simply a bridge that for Novell IPX can promote or prohibit the ability to connect to servers and/or printers by name and networks by number.

In the above example the KarlBridge has been configured so that the File Server *Dave* and the Novell Printer *Mary* cannot be seen by any computers on the local side of the KarlBridge. Also the file server *John* cannot be seen on the remote side of the KarlBridge.

Generic Ethernet Tunneling
(Through an IP Network)

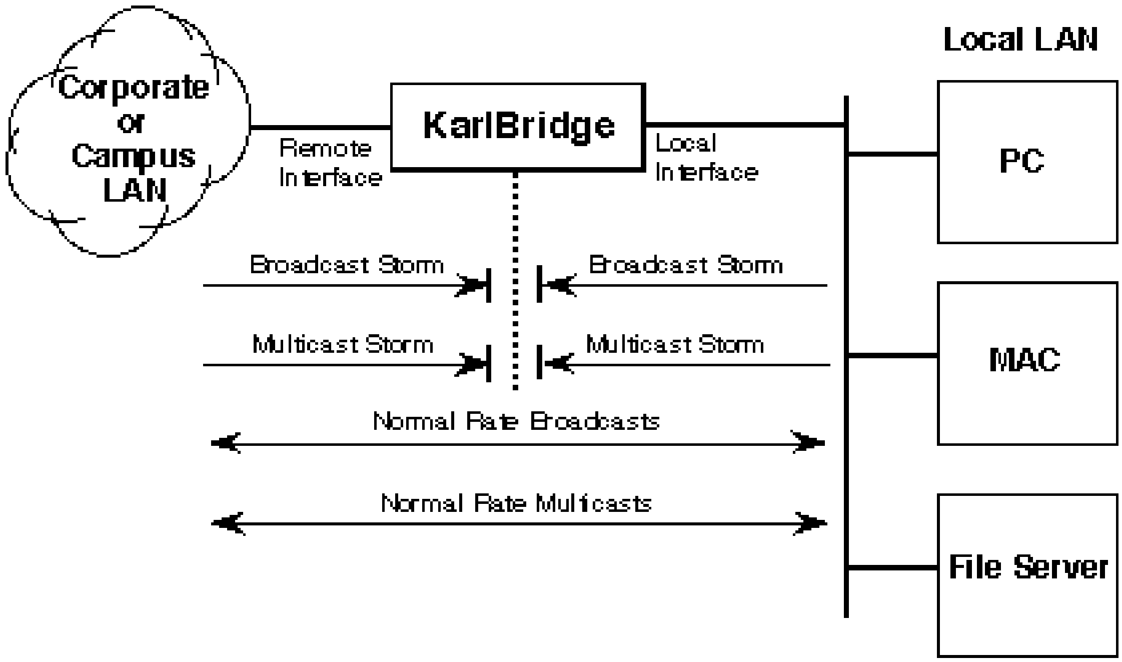


Tunneling is a method of connecting together two or more LANs that are physically connected only via an IP or Internet network. This feature allows the creation of a private virtual Ethenet network by using a standard IP or Internet network as the transport mechanism. The Ethernet packets are encapsulated in IP/UDP and travel over the IP or Internet network and can be optionally encrypted for extra security.

In the above example: The three KarlBridges are setup to tunnel one or more Ethernet protocols. This configuration allows LAN A, LAN B and LAN C to become a virtual Ethernet network with the Internet as the transport mechanism for data between them.

As an example: the KarlBridge can be setup to “Tunnel” Novell IPX or AppletTalk packets and at the same time bridge or route IP packets and also bridge NetBUI packets.

Protecting Your Network From Broadcast Storms

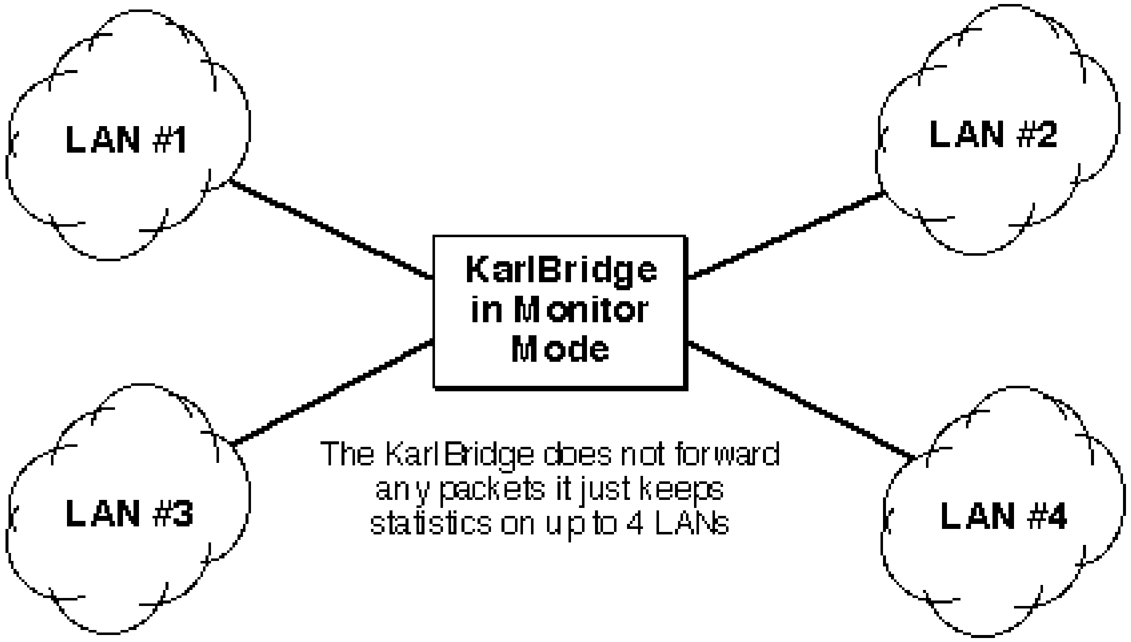


One of the unique and very useful features of the KarlBridge/KarlBrouter is its ability to keep Broadcast and Multicast storms from spreading throughout a network. Network storms can cause bridges, routers, workstations, servers and PC's to slow down or crash. Storms occur if network equipment is configured incorrectly, if network software is not functioning correctly, or if poorly designed programs such as network games are used. You can detect storms on a per port or per Ethernet address basis.

You can set the maximum number of Broadcast or Multicast packets that can occur on a particular bridge port each 1 second period before a storm condition is declared. Once it is determined that a storm is occurring then any additional Broadcast or Multicast packets received on that port will be dropped until the storm is determined to be over. The storm will be determined to be over once a 1 second period has occurred with no Broadcast or Multicast packets received on that port.

You can also set the maximum number of Broadcast or Multicast packets that can occur each one second period before a storm condition is declared for a particular Ethernet address (host). Once it is determined that a storm is occurring then any additional Broadcast or Multicast packets from that host address will be dropped until the storm is determined to be over. The storm will be determined to be over when 30 seconds has passed in which every 1 second period has less than one half the stated threshold in broadcast packets.

Network Monitoring
(Using a KarlBridge as a Network Monitor)

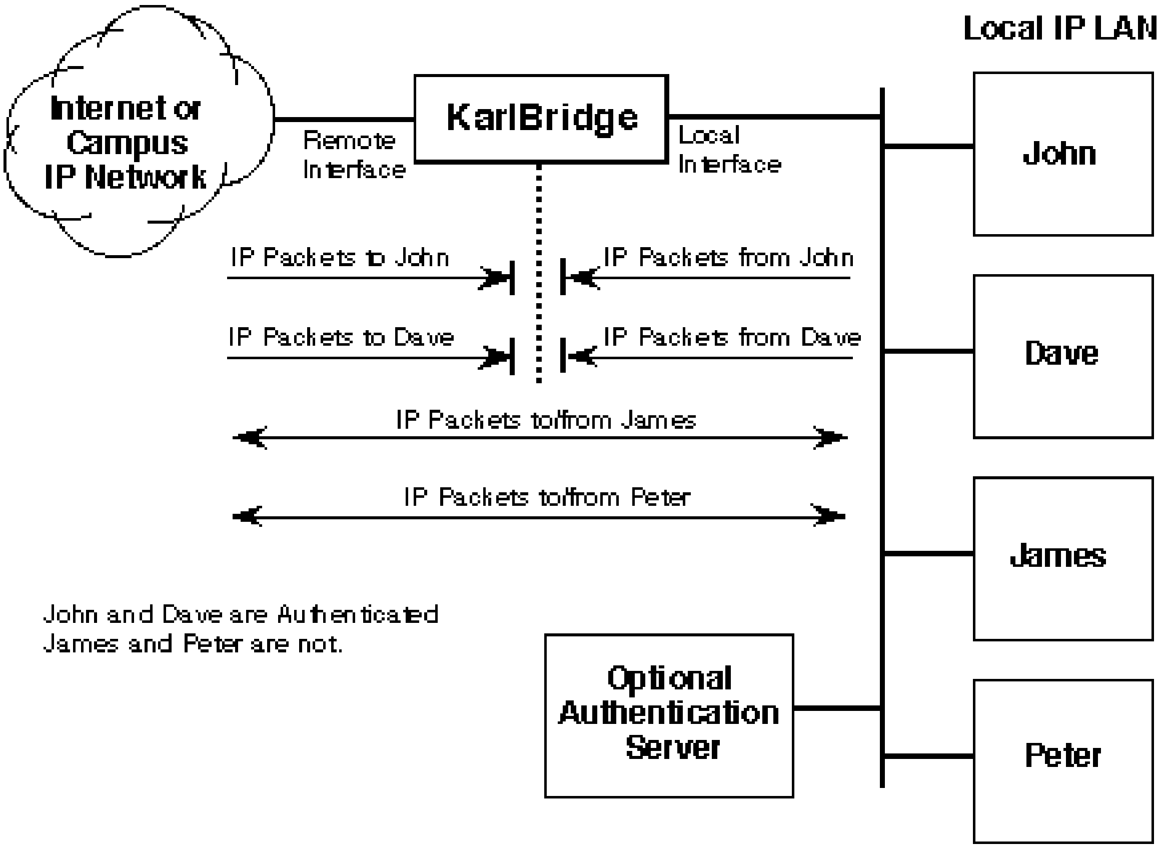

LAN #1
LAN #2
KarlBridge in Monitor Mode
LAN #3
LAN #4
The KarlBridge does not forward any packets it just keeps statistics on up to 4 LANs

The KarlBridge can be used as an SNMP network monitoring probe. It will NOT forward packets from one LAN to another. This can be accomplished by disabling the bridging and IP routing functions of the KarlBridge/KarlBrouter reducing it’s functionality to simply keeping SNMP statistics on each port.

The KarlBridge will keep many statistics including the number of packets, bytes and errors seen on each port. It will also keep a permanent record of each Ethernet address IP to Ethernet address combination that it has seen.

You can use this feature to keep statistics on what IP address is assigned to each Ethernet address. These statistics can be reported by using a standard SNMP management station capable of displaying the standard MIB II and Bridge MIB. You can also display the statistics with the KarlBridge configuration and monitoring program that is included with the KarlBridge/KarlBrouter software.

Adding Authentication
(To the IP/UDP/TCP Firewall)

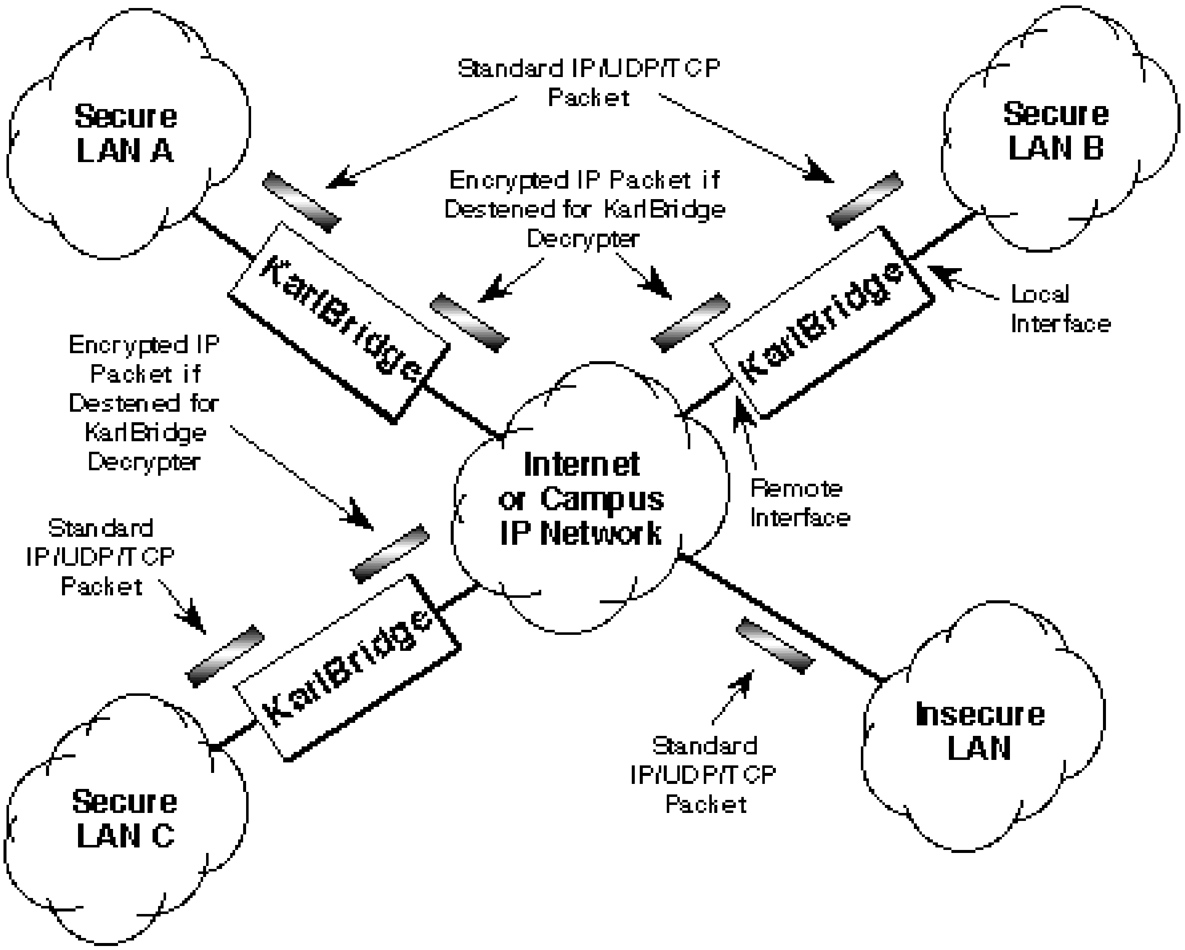


There is a KarlBridge/KarlBrouter exclusive feature that will add authentication capability to the IP/UDP/TCP firewall filters. Firewall authentication is a way of punching a hole through the firewall on a case by case basis. This can be controlled by a computer setup to be an Authentication Server as shown above.

Authentication allows the KarlBridge/KarlBrouter's UDP/TCP firewall filters to be dynamically bypassed. This feature enables data between particular subnets or hosts to flow through the firewall untouched by any security filters. This feature is very powerful and can be used to create a way to authenticate access into a particular network or host.

In the above example the computer *James* and *Peter* have been authenticated and can communicate with Internet or Campus IP Network. The other computers have not been authenticated and are subject to the strict firewall of the KarlBridge.

Adding Data Encryption
(To IP/UDP/TCP Packets)

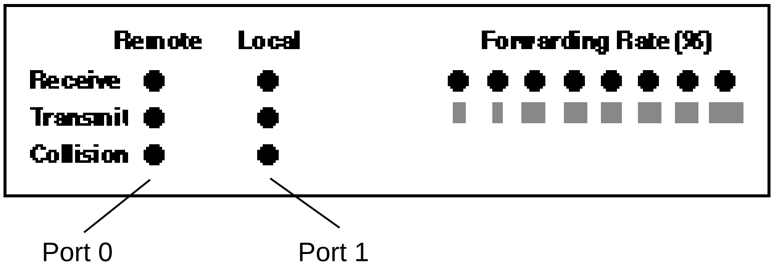

Secure LAN A
Standard IP/UDP/TCP Packet
Secure LAN B
Encrypted IP Packet if Destened for KarlBridge Decrypter
KarlBridge
KarlBridge
Local Interface
Encrypted IP Packet if Destened for KarlBridge Decrypter
Internet or Campus IP Network
Remote Interface
Standard IP/UDP/TCP Packet
KarlBridge
Insecure LAN
Standard IP/UDP/TCP Packet
Secure LAN C

In conjunction with its bridging, routing and firewall capability, the KarlBridge and KarlBrouter have the ability to selectively encrypt IP/UDP/TCP data destined for a remote IP network and the Internet. After a packets source and destination IP address matches an entry in an access list then the data portion of the UDP or TCP packet can be optionally encrypted (if destined for the remote port) or decrypted (if received on the remote port and destined for the local port).

KarlNet

Hardware Information

FRONT PANEL (ETHERNET-TO-ETHERNET)



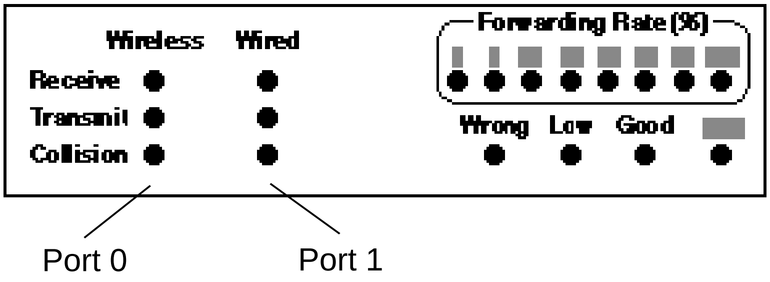
Receive: This light will blink whenever a packet is received.

Transmit: This light will blink whenever a packet is transmitted.

Collision: This light will blink whenever a collision or error is detected on the LAN.

Forwarding Rate: This will display the forwarding rate of the bridge/brouter in percent of the full theoretical Ethernet rate of 10 mega Bits per second.

FRONT PANEL (ETHERNET-TO-WAVELAN)



Wired Receive: This light will blink whenever a packet is received.

Wired Transmit: This light will blink whenever a packet is transmitted.

Wired Collision: This light will blink whenever a collision or error is detected on the LAN.

Wireless Receive: This light will blink whenever a packet is correctly received.

Wireless Transmit: This light will blink whenever a packet is transmitted

Wireless Collision: This light will blink whenever a packet is retransmitted (packets will only be retransmitted if the KellWave algorithm is being used).

Forwarding Rate: This will display the forwarding rate of the bridge/brouter in percent of the full theoretical Ethernet rate of 10 mega bits per second.

Wrong: This light will blink whenever a packet from another WaveLAN network is detected.

Low: This light will blink whenever a CellWave "hello" packet is received with a low signal to Noise Ratio.

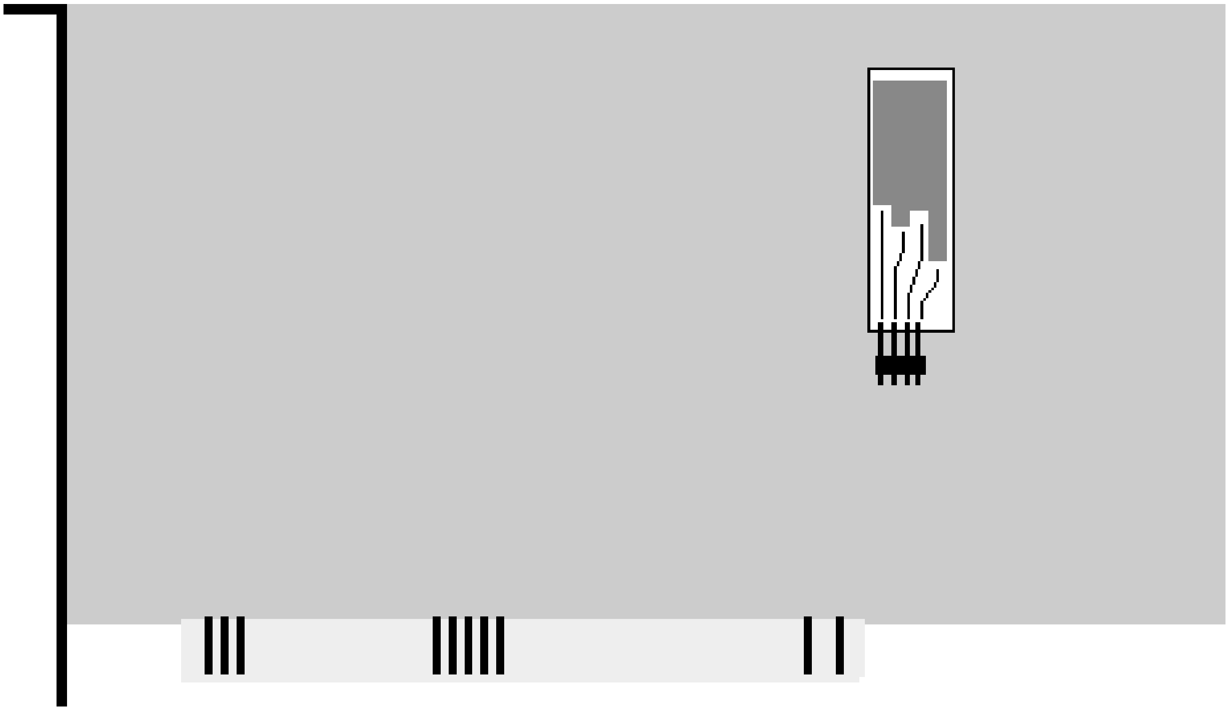
Good: This light will blink whenever a CellWave "hello" packet is received with a good signal to Noise Ratio.

Excl: This light will blink whenever a CellWave "hello" packet is received with a high signal to Noise Ratio.

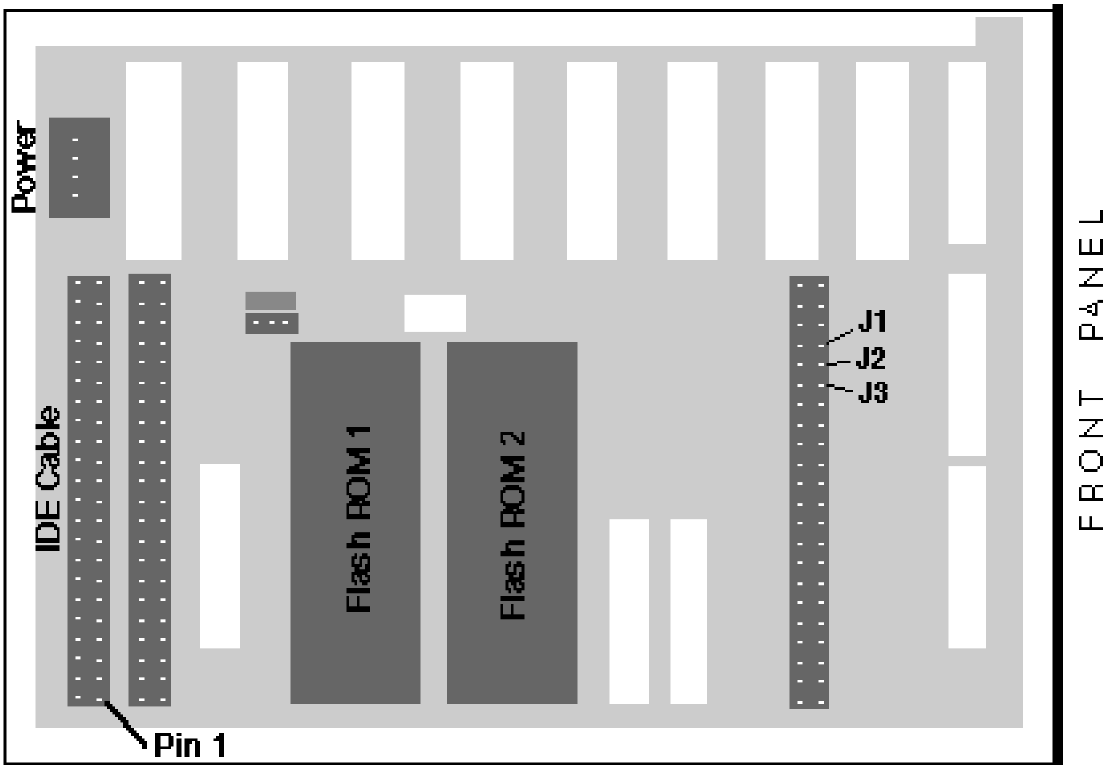
HARDWARE REMOTE CONFIGURATION PROTECTION

The Flash ROM version of the KarlBridge/KarlBrouter is configured remotely through the network using KBCONFIG via IP/SNMP. This leaves open the remote possibility that someone on the Internet could guess your SNMP read/write password and use their version of KBCONFIG to reconfigure your KarlBridge/KarlBrouter. This loophole can be completely closed by use of the SNMP Access Lists (described later in this manual) or jumpers on the Flash ROM card located inside the case. These jumpers positions are as follows:

Normal Operation	The only protection is through passwords and the SNMP Access Lists, there is no special hardware protection.
Write Protection	The configuration can be read but not written unless the hardware protections are lowered by use of the front panel protection button.
Read/Write Protection	The configuration cannot be read or written unless the hardware protections are lowered by the use of the front panel protection button.



ISA BUS FLASH CARD



KarlBridge/KarlBrouter Flash ROM Module

FUNCTION	J 1	J 2	J 3
Normal operation	ON	ON	ON
Factory Default	ON	OFF	ON
Write Protection	OFF	ON	ON
Read/Write Protection	OFF	OFF	ON
Boot on PROM	ON	ON	OFF

RESETTING TO THE FACTORY DEFAULT CONFIGURATION

The Flash ROM version of the KarlBridge/KarlBrouter is configured remotely through the network using KBCONFIG via IP/SNMP. In order for KBCONFIG to communicate through the network two things must be known; the IP Address and the read/write SNMP password (sometimes called the community name) of the KarlBridge/ KarlBrouter. When shipped from the factory the IP Address is 198.17.74.254 and the read only and read/write passwords are set to *public* and *public*. If you forget what you have changed these to you can restore them to the factory default by placing the jumper on the Flash ROM board located inside the case to the Factory Default position. You must then reboot the KarlBridge/KarlBrouter and configure it with KBCONFIG using the factory default address and passwords. Once you have changed the address and password and saved them with KBCONFIG and the KarlBridge/KarlBrouter has rebooted itself it is ready for use. You should then shut off the KarlBridge/KarlBrouter move the jumper back to Normal Operation, or one of the protection settings, and start it back up to verify that your changes have taken effect.

REMOTE AND LOCAL PORTS

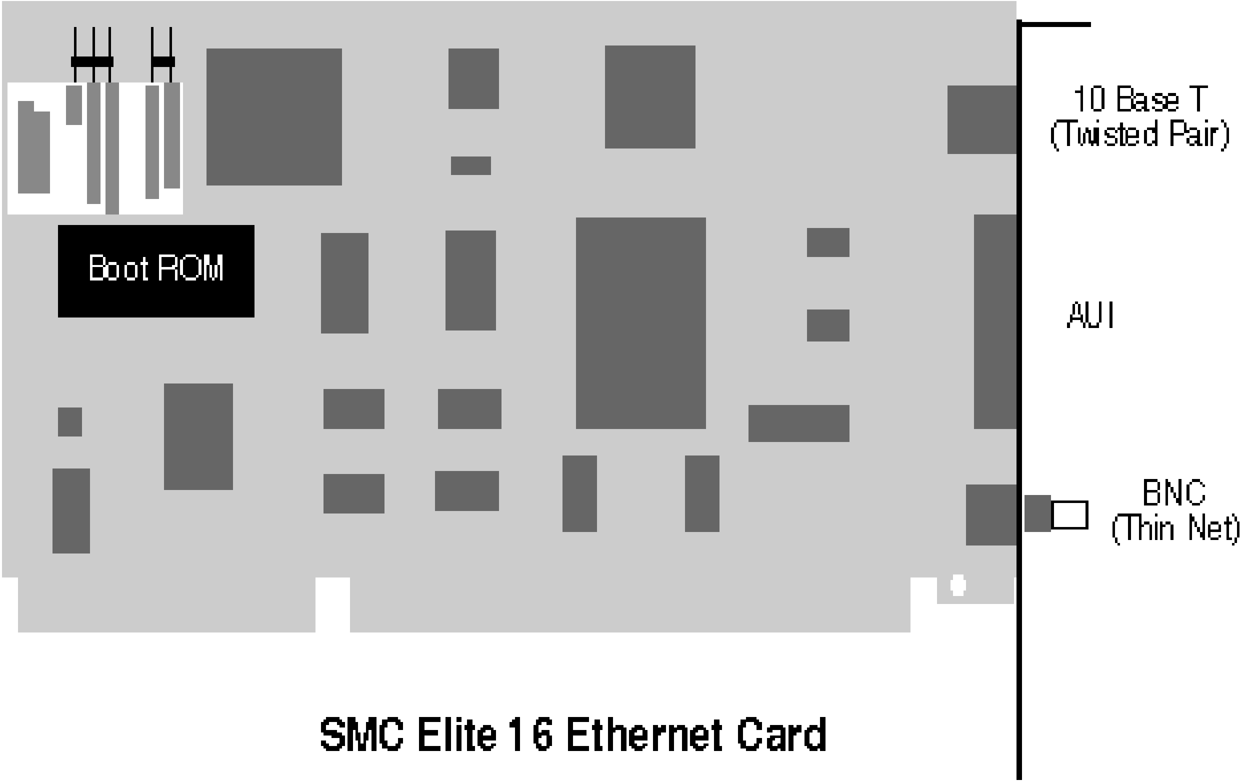
The KarlBridge and KarlBrouter's security filters provide isolation between one or more local networks and one or more remote networks. The ports on the standard 2 port KarlBridge and KarlBrouter are labeled Port 0 Remote and Port 1 Local. The work group or computer lab that you wish to isolate should be connected to the Local Port and the external network should be connected to the Remote Port. NOTE: If you have a KarlBridge/KarlBrouter that supports mixed media or more than 2 ports you will have the option in the Setup-Ports menu to change which port(s) are considered "local" and which port(s) are considered "remote".

115/230 VOLT SETTING

The Non-Auto switch KarlBridge/KarlBrouter is shipped with 115V selected. If your country uses 230V this setting should be changed. The Auto switch version of the KarlBridge/KarlBrouter automatically detects and adjusts for the proper voltage setting and no manual switch is needed or provided.

ETHERNET INTERFACE (BNC OR AUI CONNECTIONS)

The 10Base2 (Thin Wire) KarlBridge/KarlBrouter is shipped with both Ethernet cards setup for BNC (Thin Wire Ethernet). If you wish to use the AUI (transceiver) port you must open the case and change the jumpers located on the appropriate Ethernet card. These Ethernet cards have been customized for use in the commercial KarlBridge/ KarlBrouter and are not interchangeable with the standard Ethernet cards by the same manufacturer.



SMC Elite 16 Ethernet Card

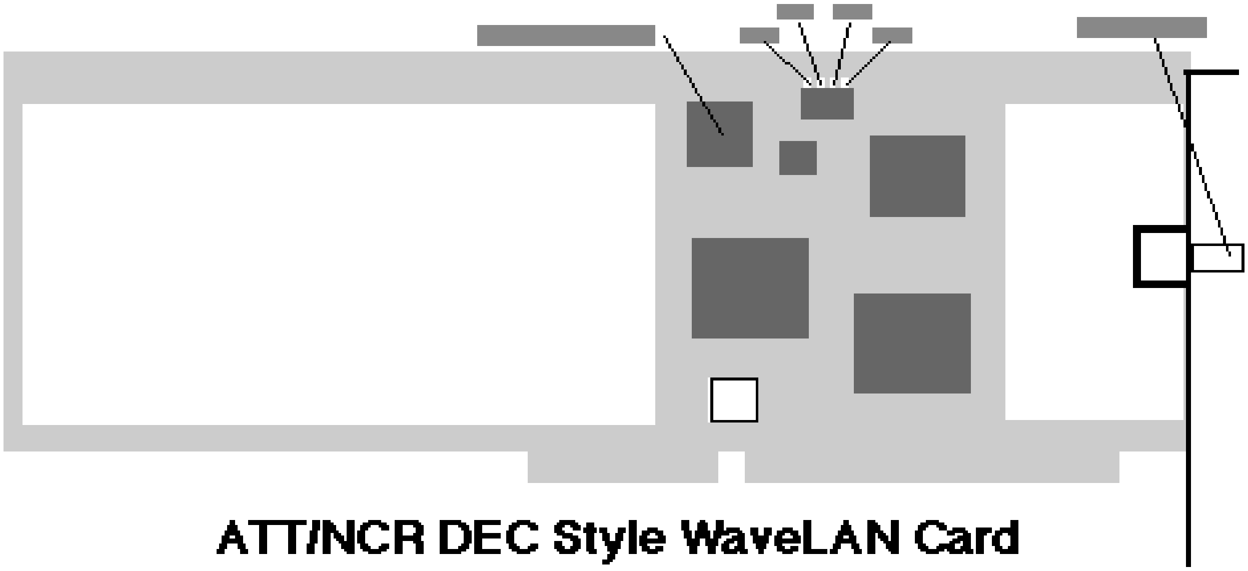
BNC
Use this setting if you are connecting your LAN to the BNC connection.

AUI & 10BaseT
Use this setting if you are connecting your LAN to either the AUI or the 10BaseT (Twisted Pair) connector.

Twisted Pair No Link
Use this setting if you are connecting your LAN to the 10BaseT (Tiwsted Pair) connector and iwsh to have No Link Integrity signal active - (This setting is not normally used).

WAVELAN INTERFACE

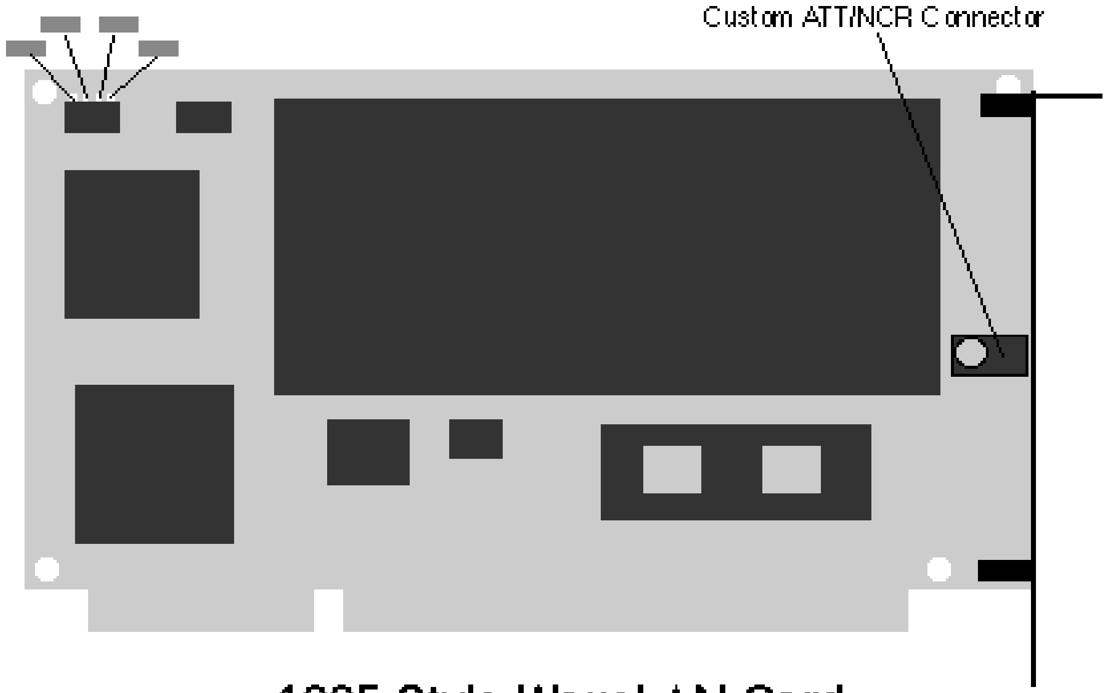
The commercial version of the KarlBridge/KarlBrouter supports a standard ATT/NCR or DEC WaveLAN wireless interface card. The card is configured in "factory default" mode (all switches in the up position). It provides a wireless link to other WaveLAN wireless cards within a building. The Omni directional antenna supplied has a range of 800 feet. With the addition of a directional antenna, (wireless network) connections can be made between buildings that are several miles apart.



ATT/NCR DEC Style WaveLAN Card

	SW1	SW2	SW3	SW4
* Port 0	off	off	off	off
Port 1	off	on	off	off
Port 2	on	on	off	off

***NOTE**: It is highly recommended that you install the WaveLAN card as KarlBridge Port 0.



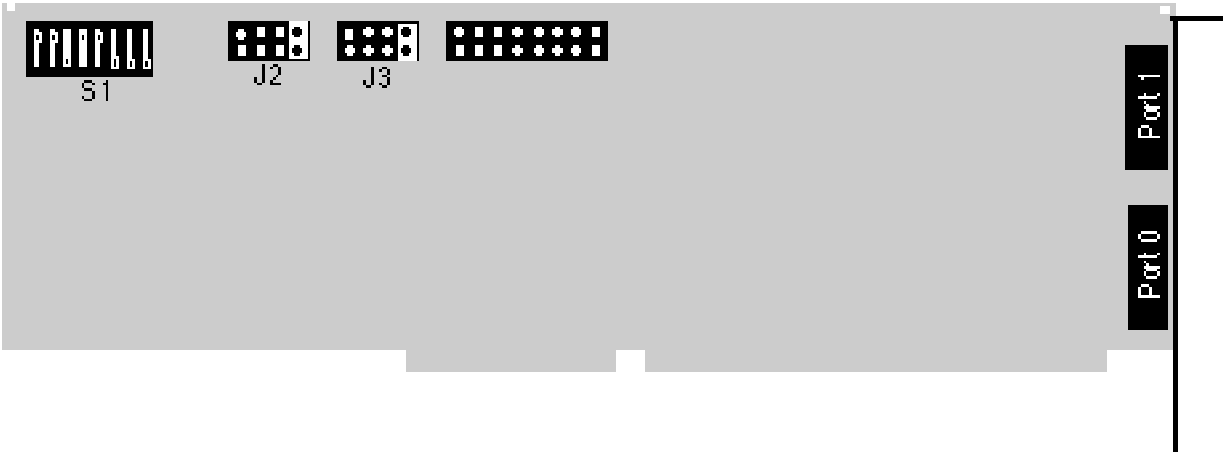
1995 Style WaveLAN Card

	SW1	SW2	SW3	SW4
* Port 0	off	off	off	off
Port 1	off	on	off	off
Port 2	on	on	off	off

***NOTE:** It is highly recommended that you install the WaveLAN card as KarlBridge Port 0.

SYNCHRONOUS INTERFACE

The KarlBridge/KarlBrouter supports one or more dual port synchronous interface cards. Each port will support a synchronous connection from 56k bps up to E1 speed (2.048 mbps). For 56/64k bps connections, an RS232 cable is provided and for T1/E1 speed connections a V.35 cable is provided.



RISCOM/H2 Synchronous Card

	S1						J2	J3
	1	2	3	4	5	6		
***** 1st & 2nd Serial Port	ON	ON	OFF	ON	ON	OFF	O	O
3rd & 4th Serial Port	OFF	ON	OFF	ON	ON	OFF	7	7
5th & 6th Serial Port	ON	OFF	OFF	ON	ON	OFF	6	6

***NOTE**: This is the typical setting.

CONFIGURATION

The KarlBridge/KarlBrouter has been designed to provide several layers of isolation and firewall security protection for many types of local area networks. You will most likely not need to use all of the features and filters provided.

RUNNING THE KBCONFIG PROGRAM
(on a floppy based KarlBridge/KarlBrouter)

Remove the KarlBridge/KarlBrouter floppy from the floppy drive and insert it into any standard PC compatible computer that is running DOS version 3 or higher with an EGA or VGA monitor. For this example it is assumed your floppy drive is drive A.

1. Copy the files KBCONFIG.EXE, KBC.EXE, KBHELP.HLP, and KBCONFIG.CFG from the "Flash ROM Remote Configuration" diskette into a directory on your hard disk.

2. Issue the command: KBCONFIG A:KBRIDGE.BIN

3. Set-up the KarlBridge/KarlBrouter features and filters by use of the menus as described in the sections later in this manual.

4. Save your new configuration back into the KBRIDGE.BIN file on the floppy by issuing the Save command under the File menu.

The KBCONFIG program modifies the KBRIDGE.BIN file which contains the bridge/router program and your filter settings. When the floppy is inserted into the KarlBridge/KarlBrouter floppy drive and the box is powered up the program KBRIDGE.BIN will boot and execute.

WARNING: The KarlBridge/KarlBrouter floppy disk boot block program will only boot the KBRIDGE.BIN file if it is contiguous. The only way to guarantee that the KBRIDGE.BIN file is contiguous is to copy it to a blank newly formatted disk with a KarlBridge boot block on it. If you copy the KBRIDGE.BIN file to a hard disk and then back to a non-blank floppy it may not be contiguous and thus will not boot properly. NOTE: When the KBCONFIG program modifies the KBRIDGE.BIN file on the floppy it does not move the KBRIDGE.BIN file and therefore will boot properly. Therefore whenever you change the configuration of the KBRIDGE.BIN file on the boot floppy always open the file on the floppy directly from KBCONFIG.

RUNNING THE KBCONFIG PROGRAM
(remotely on Flash ROM KarlBridge/KarlBrouters)

1. Ensure that a standard "Packet" driver is installed on your MS-DOS computer. It came with the software you received when you pourchased your Ethernet card. If you do not have a packet driver you can use one of the drivers that are included on the "Flash ROM Remote Configuration" diskette provided with your Flash ROM KarlBridge or KarlBrouter.

2. Copy the files KBCONFIG.EXE, KBC.EXE, KBHELP.HLP, and KBCONFIG.CFG from the "Flash ROM Remote Configuration" diskette into a directory on your hard disk.

3. If you are connected to an existing IP network then setup the KBCONFIG.CFG file to reflect your IP address, IP mask, default router, etc.

4. Issue the command: KBCONFIG.

5. Under the File menu issue an Open Remote then specify the IP address of the network connected remote KarlBridge/KarlBrouter. The factory default for the KarlBridge/KarlBrouter IP address and the IP address as shipped is 198.17.74.254.

6. Set-up the KarlBridge/KarlBrouter features and filters by use of the menus as described later in this manual.

7. Save your new configuration by issuing the Save command under the File menu.

The KBCONFIG program modifies the configuration section of the KarlBridge/ KarlBrouter Flash ROM and then the remote bridge/router will reboot.

KBCONFIG's File Menu

KBCONFIG will configure either an executable KarlBridge/KarlBrouter file or configure a remote FlashROM based KarlBridge or KarlBrouter.

CONFIGURING AN EXECUTABLE FILE

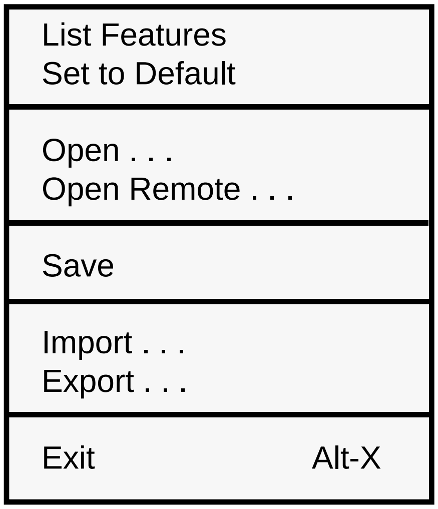
To configure an executable file you can use the Open and Save functions. The file can be either a .EXE or .BIN file. EXE files can be run under DOS and are usually the shareware demo version. BIN files can either be loaded into FlashROM or booted off of the special KarlBridge/KarlBrouter boot diskette. You must have a file open before any other KBCONFIG functions can be performed. After you have made your configuration choices you should then Save them back to the open file.

CONFIGURING A REMOTE KarlBridge or KarlBrouter

To configure a remote (network attached) KarlBridge or KarlBrouter you can use the Open Remote and Save functions. You must have a remote bridge or brouter open before any other KBCONFIG functions can be performed. After you have opened the remote device and configured it you can then Save your configuration back to the open device. When you Save back to the remote device its FlashROM will be erased and then reprogram with the new configuration.

EXPORTING AND/OR IMPORTING A CONFIGURATION

Once you have opened a remote bridge or brouter or opened an executable bridge/ brouter file you can make an ASCII file snapshot of the current configuration by using the Export function. This function will result in creating a .KBC file. The extension "KBC" is used to denote the special ASCII exported configuration file. The .KBC file once created by use of the Export function can then later be Imported into another open KarlBridge/KarlBrouter by using the Import function.



List Features

This lists the features that the open bridge supports.

Set to Default

The KarlBridge and/or KarlBrouter configuration will be set to the factory default. A factory default KarlBridge or KarlBrouter is set to bridge all protocols with no IP routing, no security filters, and a host IP address of 198.17.74.254 with the SNMP passwords of public and public.

Open

This function can be used to open a KBRIDGE.BIN or KBRIDGE.EXE file so it can be configured. The KBRIDGE.BIN file is the executable image used by the Flash ROM based or bootable floppy based commercial KarlBridge/ KarlBrouter. The KBRIDGE.EXE file is to be run under MS/DOS and is the file provided in the shareware/demo version of the KarlBridge.

Open Remote

KBCONFIG can be used to configure a network attached FlashROM based KarlBridge/ KarlBrouter. It can also be used to read the configuration from a floppy based network attached KarlBridge/KarlBrouter.

Save

Saves the current configuration back to the currently open KarlBridge/KarlBrouter executable file (.BIN or .EXE) or the remote network attached FlashROM based KarlBridge/KarlBrouter.

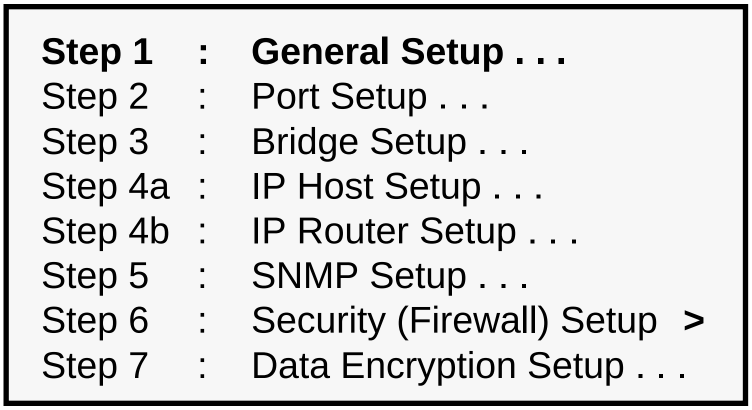
WARNING! Do not turn off the power on a FlashROM based KarlBridge/KarlBrouter until at least 30 seconds after a Save operation is performed. When a Save is performed to a remote FlashROM based KarlBridge/KarlBrouter the FlashROM will be erased and then new configurations will be programmed in. If the power is shut off during this erase/program operation the FlashROM will be corrupted. If this happens you will have to phone technical support to obtain the recovery procedure.

Import
Once either a remote FlashROM bridge/router or a local executable .BIN or .EXE file has been opened a .KBC file can then be applied to the configuration. The .KBC file is an ASCII file and can be edited with a standard editor.

Export
A snap shot of the current configuration settings can be saved into an ASCII .KBC export file for archive purposes or later importing.

Exit
The KBCONFIG program will exit back to DOS or Windows.

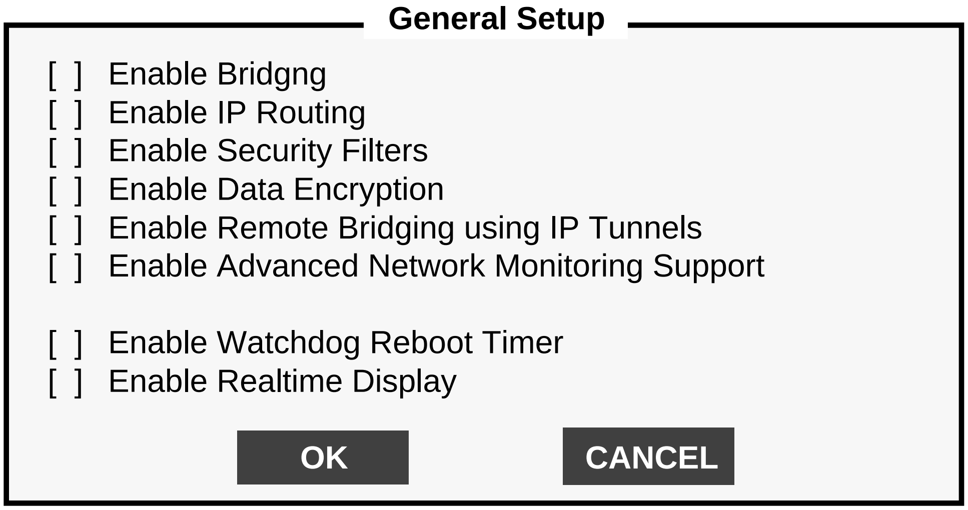
KBCONFIG'S SETUP MENU



Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Step 4b : IP Router Setup . . .
Step 5 : SNMP Setup . . .
Step 6 : Security (Firewall) Setup **>**
Step 7 : Data Encryption Setup . . .

This menu is used to setup the KarlBridge or KarlBrouter. It is highly recommended that you setup the bridge or brouter starting with Step 1.

STEP 1: GENERAL SETUP



General Setup

[ ] Enable Bridgng
[ ] Enable IP Routing
[ ] Enable Security Filters
[ ] Enable Data Encryption
[ ] Enable Remote Bridging using IP Tunnels
[ ] Enable Advanced Network Monitoring Support

[ ] Enable Watchdog Reboot Timer
[ ] Enable Realtime Display

[X] Enable Bridging
The transparent bridging function will be enabled when this is enabled. If you do not want the Bridge/Router to perform the bridging function then you must enable this. When bridging is enabled the Bridge Menu will be able to be used.

[X] Enable IP Routing
If you have purchased the IP Routing option then you can enable it with this button. The routing will work properly only if the routes are setup in the IP Route menu.

[X] Enable Security Filters
Enabling security filters will cause the KarlBridge/KarlBrouter to analyze each network packet to determine if it should be passed or dropped. If the KarlBridge or KarlBrouter is to be used as a simple standard transparent bridge and/or simple IP Router with no advanced filtering, then this feature should be disabled. If you wish to use the advanced filtering, firewall and security features then you must enable security filters.

NOTE: The default settings for the UDP/TCP, Novell, AppleTalk, and DECNET filters is to DROP all packets. This means that after enabling security filters you must then enable the appropriate protocol specific security filters (Step 6 under the Setup Menu).

[X] Enable Data Encryption
The Data Encryption option can be used to either encrypt/decrypt tunneled data packets that flow between KarlBridge tunnel partners or to encrypt/decrypt UDP/TCP packets that flow between KarlBridge/KarlBrouters. Since only the UDP/TCP data portion of the packet is encrypted the packet will be routed correctly by standard IP routers.

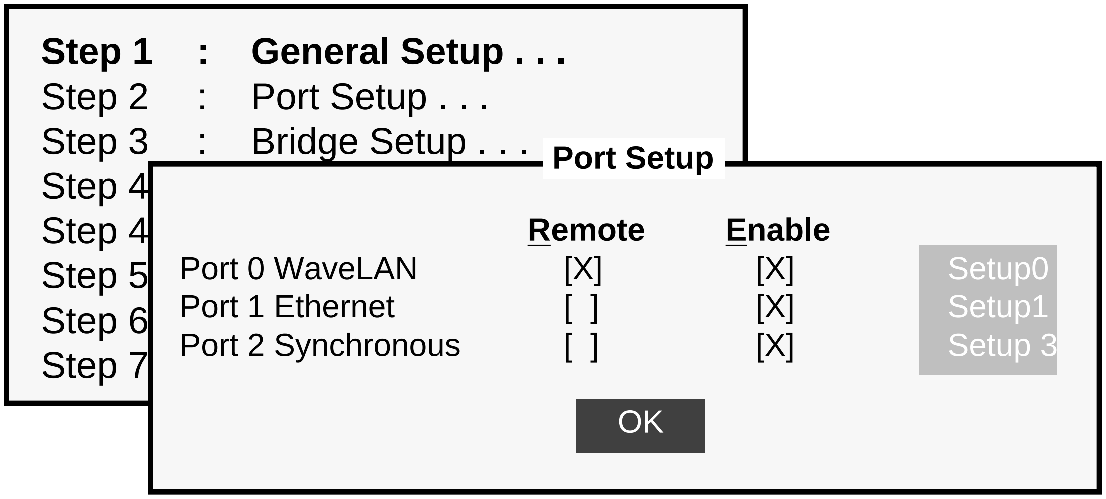
[X] Enable Remote Bridging using IP Tunnels
The KarlBridge/KarlBrouter supports a special feature which will enable Ethernet packets of any protocol type to be encapsulated in IP and then sent to other KarlBridges for de-encapsulation. This method can be used to setup "virtual" Ethernet LANs between several points using the IP network as the transport layer.

[X] Enable Advanced Network Monitoring Support
Network monitoring is not supported in this version of the KarlBridge and KarlBrouter.

[X] Enable Watchdog Reboot Timer
The KarlBridge/KarlBrouter contains a watch dog timer reboot feature. If no packets are seen on the network for more than 10 minutes (a very rare occurrence), the KarlBridge/ KarlBrouter will reboot itself. Once it has rebooted the 10 minute reboot timer will not activate again until a packet has been seen on one of the ports. This is to ensure that only one reboot will occur if the entire network is truly shutdown.

[X] Enable Real-time Display
Some KarlBridges and KarlBrouters contain a CGA, EGA or VGA controller board and display. You can enable the displaying of real-time bridge/router statistics with this option. If you have a Wireless KarlBridge/KarlBrouter then the RF signal level and quality will also be displayed. If you do not have a display then it is recommended that you disable this function.

STEP 2: PORT SETUP



Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4
Step 4
Step 5
Step 6
Step 7

Port Setup

	Remote	**Enable**	
Port 0 WaveLAN	[X]	[X]	Setup0
Port 1 Ethernet	[ ]	[X]	Setup1
Port 2 Synchronous	[ ]	[X]	Setup 3

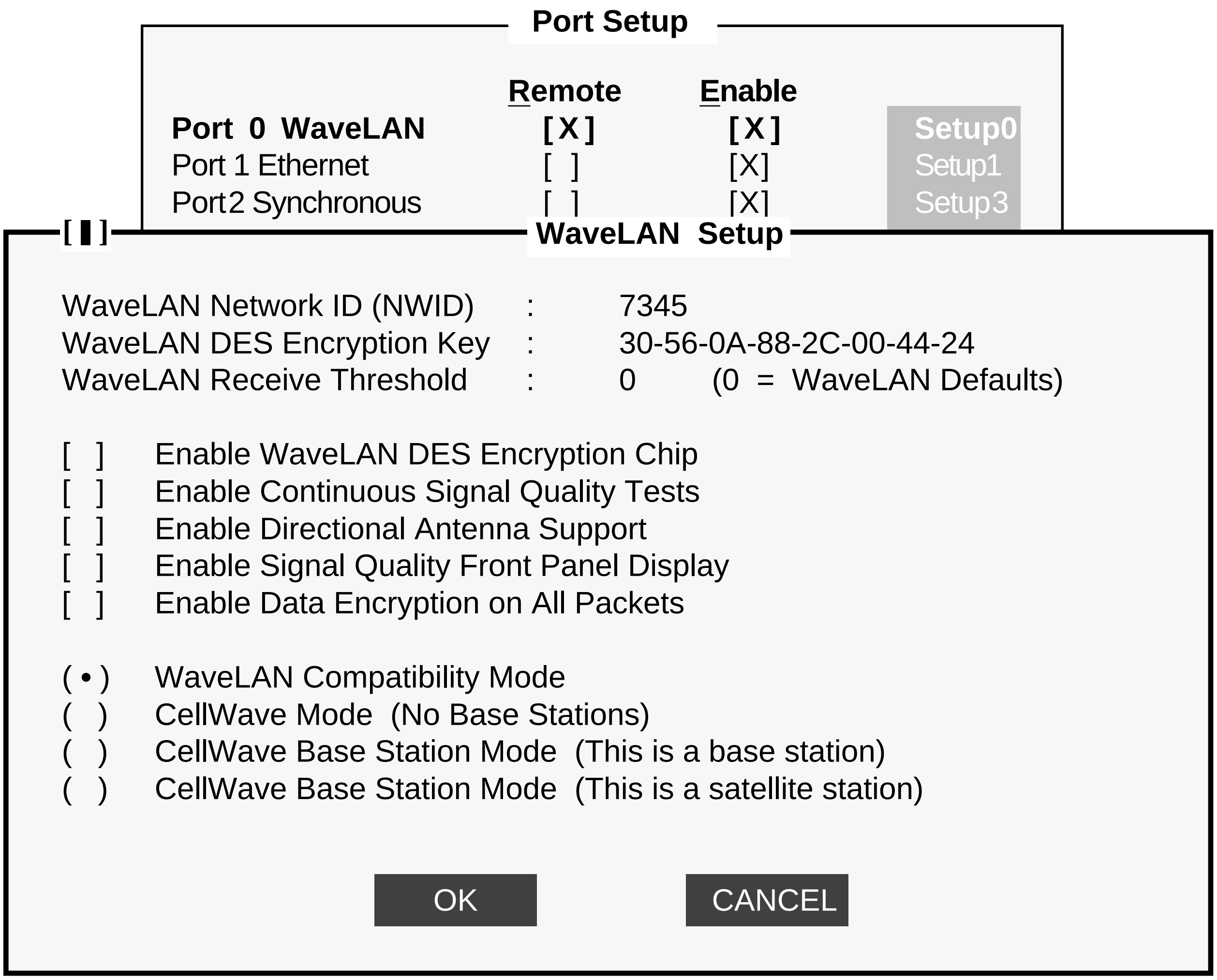
NOTE: Your particular KarlBridge may not have all the interfaces shown in this diagram (i.e., WaveLAN, Ethernet, Synchronous . . .)

[X] Remote
This setting will designate the port as being a "Remote" port. The Remote/Local designation is significant only for the Security Filters. The security filters will pass (permit) or drop (deny) packets of particular types from being forwarded between ports designated as "Local" and those designated as "Remote".

[X] Enable
On bridges or routers that have more than 2 ports this setting will enable the particular port. On 2 port bridge/routers both ports are always enabled.

WaveLAN Interface

This setup is for the ATT/NCR or DEC WaveLAN wireless card. The WaveLAN card is used in the Wireless KarlBridge and KarlBrouter products. It is a 2 Megabyte spread spectrum radio LAN card that is compatible with WaveLAN cards sold by ATT/NCR, DEC, Solectek/AirLAN and Persoft.



Port Setup

	Remote	Enable	
Port 0 WaveLAN	**[X]**	**[X]**	**Setup0**
Port 1 Ethernet	[ ]	[X]	Setup1
Port2 Synchronous	[ ]	[X]	Setup3

[▮] WaveLAN Setup

WaveLAN Network ID (NWID) : 7345
WaveLAN DES Encryption Key : 30-56-0A-88-2C-00-44-24
WaveLAN Receive Threshold : 0 (0 = WaveLAN Defaults)

[ ] Enable WaveLAN DES Encryption Chip
[ ] Enable Continuous Signal Quality Tests
[ ] Enable Directional Antenna Support
[ ] Enable Signal Quality Front Panel Display
[ ] Enable Data Encryption on All Packets

(•) WaveLAN Compatibility Mode
( ) CellWave Mode (No Base Stations)
( ) CellWave Base Station Mode (This is a base station)
( ) CellWave Base Station Mode (This is a satellite station)

NOTE: The CellWAVE Feature cannot be used if "Remote Bridging using IP Tunnels" is enabled in the General Setup Menu.

WaveLAN Network ID (NWID)

Each WaveLAN wireless network is given a different network ID. In order for WaveLAN cards, either NCR, DEC, Solectek, or Persoft to communicate with each other they must have the same network ID number. This number is a 4 digit hexadecimal number from 100 through FFFF hex.

WaveLAN DES Encryption Key

If your wireless KarlBridge/KarlBrouter WaveLAN card contains the optional DES encryption chip and if you enable the DES encryption chip then the data that is transmitted will be encrypted. The Encryption Key must be 8 even bytes separated by dashes and cannot be all zeros; (ex) 30-52-0A-88-2C-00-44-24. These bytes together specify the standard 56 bit DES encryption key and is specified the same way as the other WaveLAN vendors specify their encryption key. Note that the WaveLAN Network ID (NWID) is also encrypted. If the key you specify does not match the key specified on other WaveLAN Wireless devices the wrongly encrypted packet will be received as a Wrong Network ID packet.

WaveLAN Receive Threshold

This setting changes the WaveLAN cards receive threshold (similar to a squelch control on a two way radio). A value of 0 forces the WaveLAN card to use its default threshold value. You should use 0 when you first setup your wireless network and then consider increasing it later.

When the threshold is set to 1 the WaveLAN receiver is very sensitive to spread spectrum signals. If you set this to a higher level the WaveLAN receiver will be less sensitive to background noise, reflections, and signals from other WaveLAN network cards. If this number is set to high, the WaveLAN card will not be able to receive anything. The range of this value is 0 through 38 where 0 is the cards default value, 1 is the most sensitive and 38 is the least sensitive.

You can determine if your WaveLAN card is receiving unwanted signals by examining the "Wrong Net ID" variable on the "KarlBridge/KarlBrouter Remote Stats" menu of the KBCONFIG program. If this value is incrementing at a rate more than 10 to 50 per second you may want to increase the WaveLAN Receive Threshold. It has been our experience that a value between 5 and 20 is usually appropriate. If you set it to high you will shut off the ability to receive from all wireless stations. This is the same characteristics experienced if you set the squelch to high on a two way radio.

[X] Enable WaveLAN DES Encryption

If your wireless KarlBridge/KarlBrouter WaveLAN cards contains the optional DES encryption chip and if you have specified a DES encryption key then you can enable the encryption function with this option. All wireless data transmissions will be encrypted and all receptions will be decrypted. The WaveLAN Network ID (NWID) is also encrypted. If the key you specify does not match the key specified on other WaveLAN Wireless devices the wrongly encrypted packet will be received as a Wrong Net ID packet.

[X] Enable Continuous Signal Quality Tests

If continuous signal quality tests are enabled the wireless KarlBridge/KarlBrouter will send a special hello/test packets at a rate of one per second. This is helpful because with these tests enabled any receiving station will keep statistics on its ability to receive from this station. The cost for this feature is that these hello/test packets will take up a small amount of RF air time. If you only have a few wireless stations this is inconsequential. If you have hundreds of wireless stations in your wireless cell and all of these stations are transmitting hello/test packets the wireless LAN will be slowed down.

[X] Enable Directional Antenna Support
The WaveLAN card is designed to connect to either a special omni-directional antenna or a directional antenna. If you are using a directional antenna you should enable directional antenna support. With directional antenna support enabled, the WaveLAN card stops sending out the 10 Volt, 1 MHz square wave signal needed only by the special omni-directional antenna. Note: A DC blocking device should be connected to the WaveLAN cards antenna port if the WaveLAN card is connected to a DC grounded directional antenna such as the loop yagi.

[X] Enable Signal Quality Front Panel Display
This function will enable WaveLAN signal quality statistics on the CRT monitor or LCD front panel display.

[X] Enable Data Encryption on All Packets
Some KarlBridges and KarlBrouters contain a special software encryption algorithm that is distinct from the optional WaveLAN DES encryption chip. If Data Encryption is enabled on the General Setup menu and if an Encryption Key is setup in the Data Encryption menu then enabling encryption here will cause all packets transmitted over the WaveLAN wireless network to be software encrypted.

(•) WaveLAN Compatibility Mode
KarlNet, ATT/NCR, DEC, Persoft, Solectek and others can transmit and receive data over WaveLAN wireless networks in an industry compatible way. This setting will enable the KarlBridge/KarlBrouter to transmit and receive its WaveLAN wireless packets in this compatible way.

(•) CellWave Mode (No Base Station)
The industry compatible way of transmitting and receiving data over WaveLAN (and many other) wireless networks cause data packets to be frequently lost. This is due to the fact that a wireless network does not have the ability to detect collisions like an Ethernet network has. In an Ethernet network collisions can be detected by the hardware (Ethernet chip) and are automatically retransmitted. Ethernet is referred to as CSMA/CD (Carrier Sense Multiple Access with Collision Detect). Wireless networks are CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). The reason that collisions cannot be detected is because with radio you cannot receive and transmit at the same time hence you cannot detect the collisions. In practice a properly operating WaveLAN point-to-point network will loose, due to collisions, approximately 1% of the transmitted packets. This packet loss is not normally a problem with protocols such as Novell IPX (without the burst mode NLM) but will cause networks using most other protocols to experience poor performance.

If all of the wireless KarlBridge/KarlBrouters in your wireless cell can "hear" each other and if you are running a non-Novell IPX protocol or Novell IPX with burst mode NLM then this setting will greatly improve the performance of your wireless network.

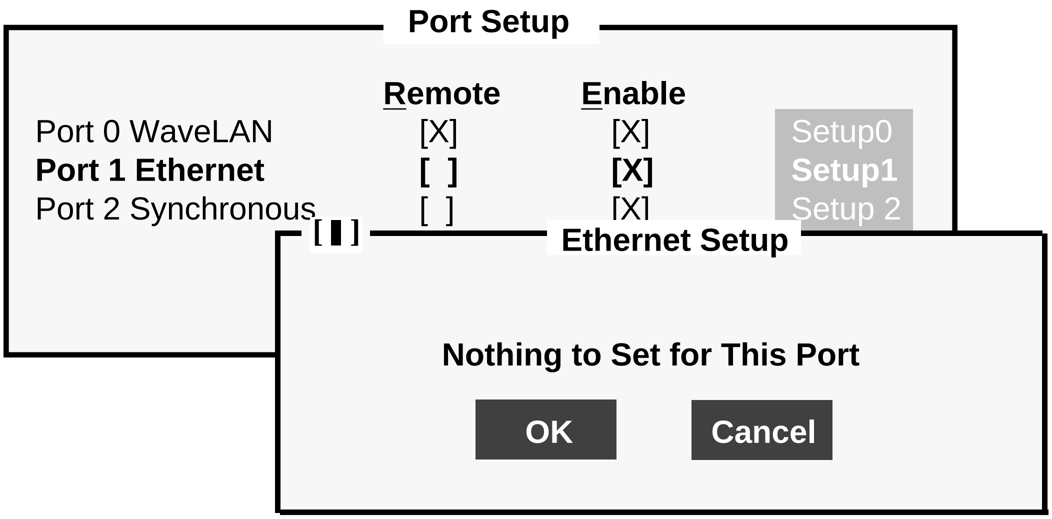
(•) CellWave Base Station Mode (This is a base station)
This setting should be used if this wireless KarlBridge/KarlBrouter is the one and only base station in the wireless network (i.e. a WaveLAN network with the same Network ID, NWID).
With the previously mentioned CellWave Mode (No Base Station) setting there is a requirement that all wireless stations be able to transmit to and receive from ALL other stations in the wireless network. This is not always possible due to the particular topology and terrain. The Wireless KarlBridge/KarlBrouter has a special mode where one of the wireless nodes can be setup as a "base" station and all others can be setup as "satellite" stations. In this configuration the only requirement is that each satellite station be able to communicate with the one base station. The base station is responsible for "repeating" packets that need to travel between satellite stations.

The performance of this approach is slightly improved if the base station is connected to the most heavily loaded file server or wired network access point. This is due to the fact that data flowing from one satellite to another satellite station must be repeated (retransmitted) by the base station using more of the wireless bandwidth. Data packets flowing from a satellite station to the base station are transmitted directly without the need to be repeated.

(•) CellWave Base Station Mode (This is a satellite station)
Set this if this wireless KarlBridge/KarlBrouter is one of the satellite stations in the wireless network. (i.e. a WaveLAN network with the same Network ID, NWID).

Ethernet Interface

There are no special hardware setups needed for Ethernet ports.



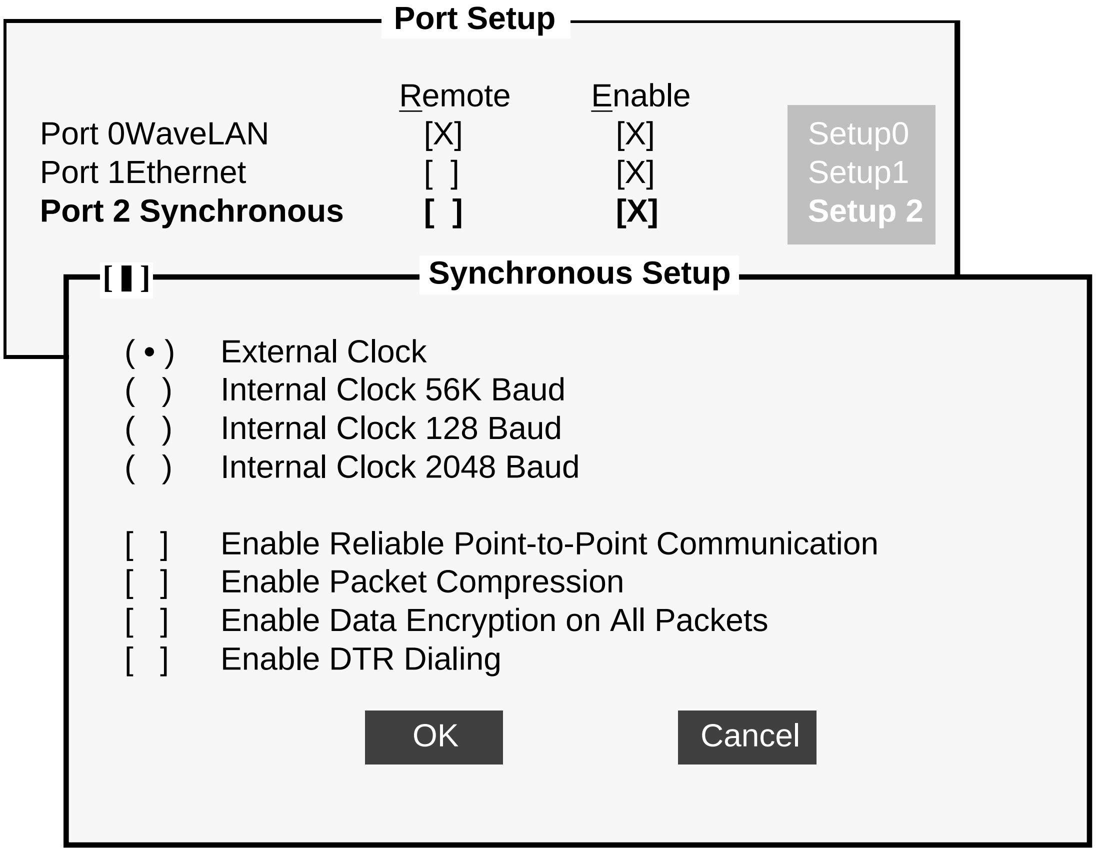
Port Setup

	Remote	Enable	
Port 0 WaveLAN	[X]	[X]	Setup0
Port 1 Ethernet	**[ ]**	**[X]**	**Setup1**
Port 2 Synchronous	[ ]	[X]	Setup 2

Ethernet Setup

Nothing to Set for This Port

Synchronus Interface



Port Setup

	Remote	Enable	
Port 0WaveLAN	[X]	[X]	Setup0
Port 1Ethernet	[ ]	[X]	Setup1
Port 2 Synchronous	**[ ]**	**[X]**	**Setup 2**

[▮] **Synchronous Setup**

(•) External Clock
( ) Internal Clock 56K Baud
( ) Internal Clock 128 Baud
( ) Internal Clock 2048 Baud

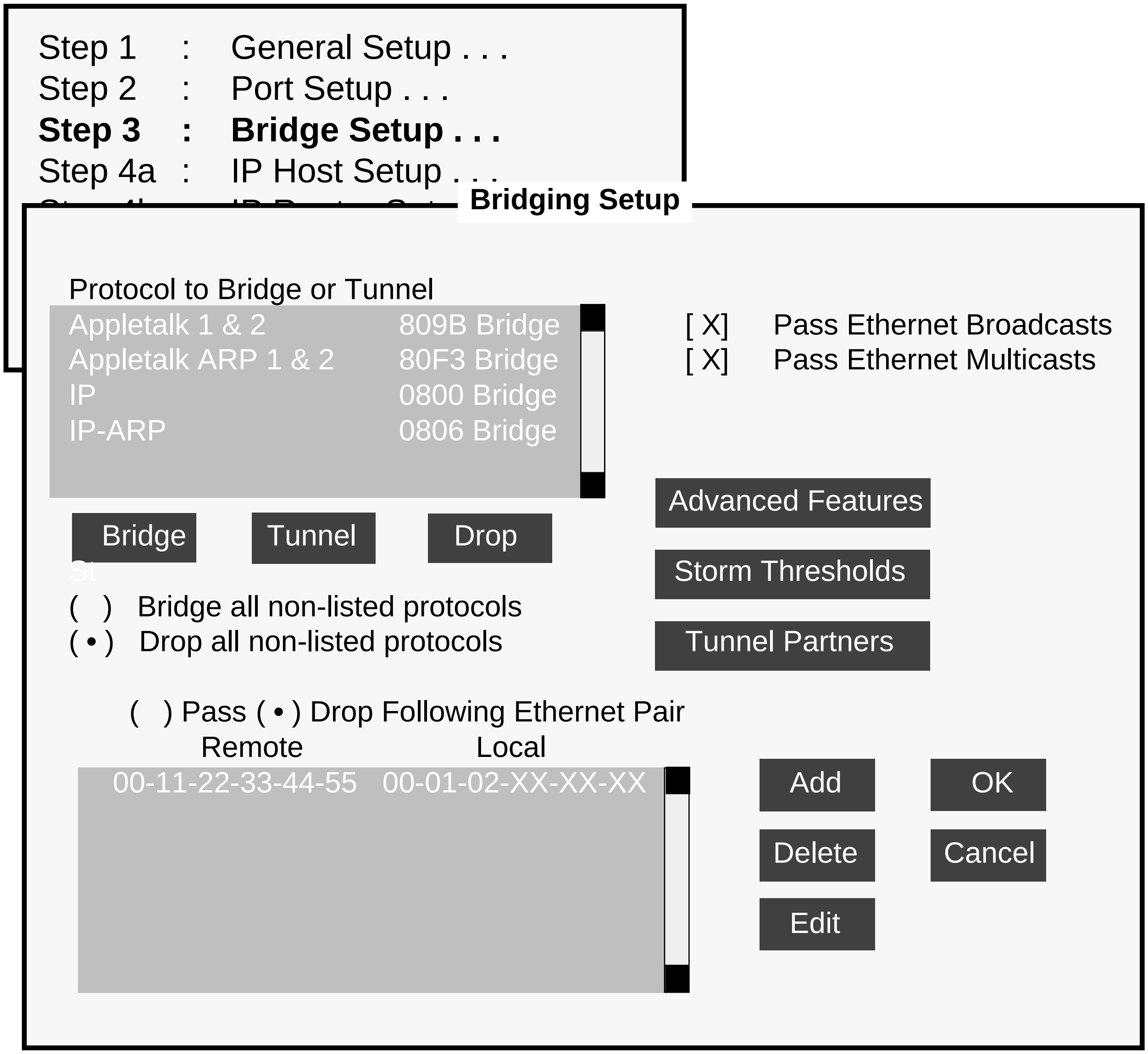
[ ] Enable Reliable Point-to-Point Communication
[ ] Enable Packet Compression
[ ] Enable Data Encryption on All Packets
[ ] Enable DTR Dialing

(•) External Clock
This setting will enable the external clock inputs and disable the internal clock source.

(•) Internal Clock
One of these settings will enable the internal clock generator to the specified bit rate.

[X] Enable Date Encryption on All Packets
Some KarlBridges and KarlBrouters contain a special software encryption algorithm that is distinct from the optional WaveLAN DES encryption chip. If Data Encryption is enabled on the General Setup menu and if an Encryption Key is setup in the Data Encryption menu then enabling encryption here will cause all packets transmitted over the synchronous port to be encrypted.

STEP 3: BRIDGE SETUP


Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Bridging Setup
Bridge
Tunnel
Drop
St
Advanced Features
Storm Thresholds
Tunnel Partners
Add
OK
Delete
Cancel
Edit

Protocol to Bridge or Tunnel

```
Appletalk 1 & 2         809B Bridge
Appletalk ARP 1 & 2     80F3 Bridge
IP                      0800 Bridge
IP-ARP                  0806 Bridge
```

☒ Pass Ethernet Broadcasts
☒ Pass Ethernet Multicasts

○ Bridge all non-listed protocols
● Drop all non-listed protocols

○ Pass ● Drop Following Ethernet Pair

```
     Remote              Local
00-11-22-33-44-55   00-01-02-XX-XX-XX
```

NOTE: The Tunnel and Tunnel Partners Buttons will not appear unless "Remote Bridging using IP Tunnels" is enabled in the General Setup Menu.

Protocol to Bridge or Tunnel

This menu specifies the Ethernet protocols to Bridge, Drop or optionally Tunnel. Each protocol can be bridged (a synonym for passed) or can be dropped as selected with the Bridge or Drop button. All other protocols not specified in the menu are then either bridged or dropped depending upon the mode selected by the radio buttons labeled "Bridge all non-listed protocols" or "Drop all non-listed protocols".

It is recommended that you bridge only the protocols that you absolutely need and drop all non-listed protocols. If you elect to bridge IP, DECNET, Novell, or AppleTalk then you will have the opportunity to setup additional filters under the Setup - Security menus. You will be given the opportunity to specify in more detail the types of services you wish to promote (pass) or restrict (drop) for the particular protocols selected.

Tunneling is a method of encapsulating Ethernet packets, received from the "Local" port in a IP/UPD packet and sending them to one or more tunnel partners. Tunneling can be used to setup virtual Ethernet networks. You can tunnel some protocols, bridge other protocols and drop other protocols all simultaneously.

(•) Bridge ( ) Drop all non-listed protocols
This setting will determine what is to happen to packets that are not listed in the "Protocol to Bridge or Tunnel" menu.

[X] Pass Ethernet Broadcast
Standard Ethernet bridges will always forward broadcast packets. Many protocols do not use broadcasts (e.g. AppleTalk Phase II, DECNET and others). However, IP/ARP does use broadcasts. If you do not use IP or any other protocol that requires broadcasts then you can drop them. Shutting off broadcast packets will reduce the traffic on your network and will also greatly reduce the number of interrupts that each computer connected to your network experiences. Networks with a high number of broadcasts will slow down the processing of each attached computer even if it is not using the network.

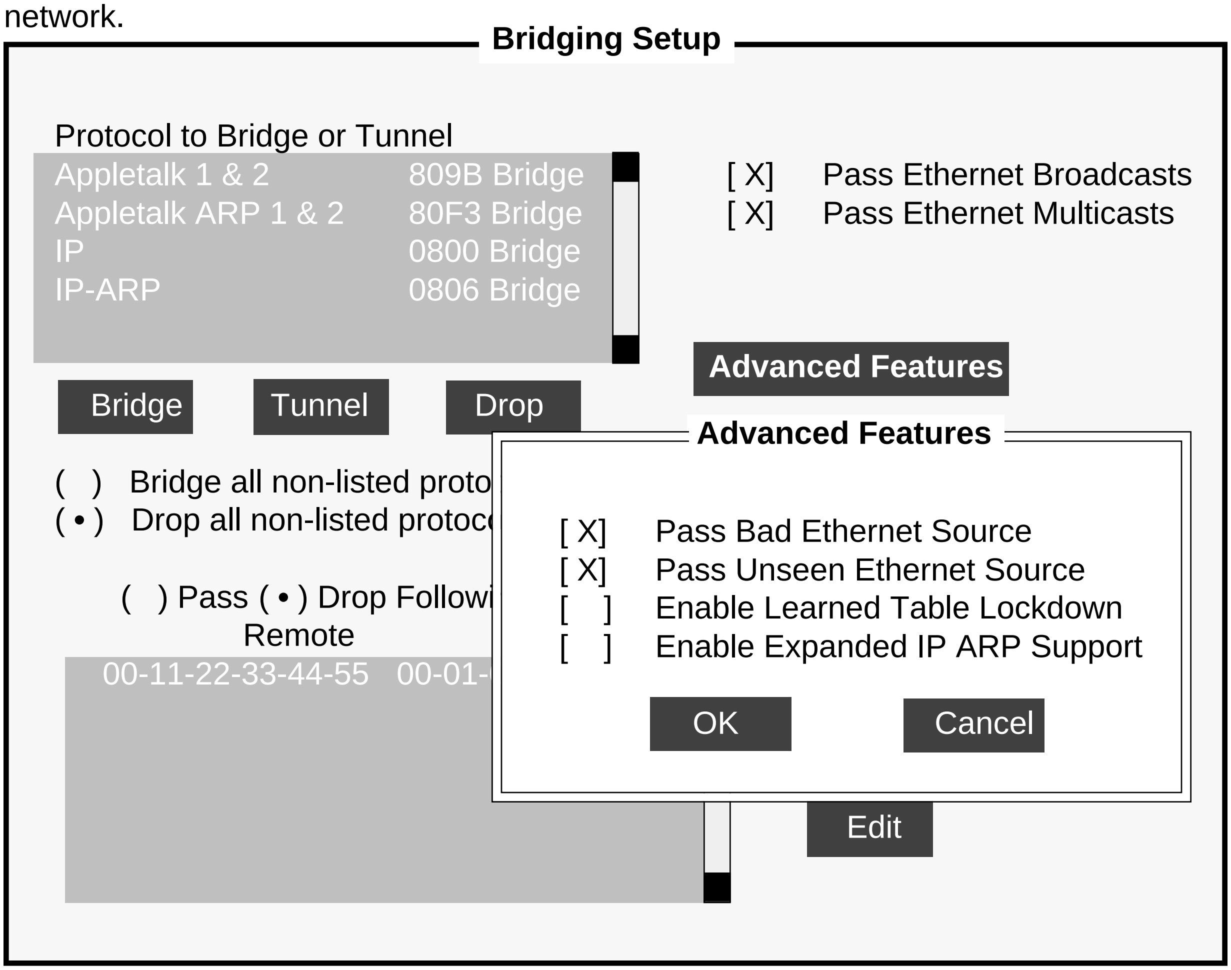
[X] Pass Ethernet Multicasts
Standard Ethernet bridges will always forward multicast packets. Some protocols do not use multicast packets, such as IP and Novell IPX. If you do not use protocols that use multicast packets then you can drop them by shutting off multicasts on the KarlBridge. Shutting off multicast packets will reduce the traffic on your network and will also reduce the number of interrupts that each computer connected to your network experiences.

(•) Pass ( ) Drop Following Address Pair
This menu specifies the Ethernet addresses that should be either Passed or Dropped both the source and destination address are checked against this filter. An entire 6 byte Ethernet address can be filtered or just portions of it. This menu can be used to inhibit or promote communication with a several particular Ethernet addresses or groups of Ethernet addresses. This approach of specifying Ethernet addresses is similar to a standard bridge that supports Ethernet address filtering. We have found this approach to not be very useful, however, support it for completeness.

As an example if the menu is set to "Drop following Pair" and an address pair of: 00-11-22-33-44-55 & 00-01-02-XX-XX-XX is specified then data packets from the address 00-11-22-33-44-55 to any addresses that start with 00-01-02 will be dropped.

Advanced Features

This menu contains advanced bridging options. These options should be changed from their default only if you clearly understand their functions and how they may impact your network.



Bridging Setup

Protocol to Bridge or Tunnel

```
Appletalk 1 & 2        809B Bridge
Appletalk ARP 1 & 2    80F3 Bridge
IP                     0800 Bridge
IP-ARP                 0806 Bridge
```

[X] Pass Ethernet Broadcasts
[X] Pass Ethernet Multicasts

( ) Bridge all non-listed proto
(•) Drop all non-listed protoc

( ) Pass (•) Drop Followi
Remote
00-11-22-33-44-55 00-01-

Advanced Features

[X] Pass Bad Ethernet Source
[X] Pass Unseen Ethernet Source
[ ] Enable Learned Table Lockdown
[ ] Enable Expanded IP ARP Support

[X] Pass Bad Ethernet Source

The standard Ethernet bridges we have tested will pass Ethernet packets with a broadcast or multicast address as their source (i.e. the first bit set to 1). The Ethernet specification for Transparent (i.e. Non-Source Routing) bridges does not allow these types of packets and are considered as "bad" packets. Our studies have shown that a common failure mode of many Ethernet interfaces and networking software is to transmit packets like these. If you do not need the KarlBridge to pass Source Routing packets it is suggested that you set it to drop these packets. Default: Pass

[X] Pass Unseen Ethernet Source

Standard Ethernet bridges will always forward packets with destination addresses that have not been "learned" (i.e. not been seen as a source address of a packet). This characteristic is needed for the proper operation of an Ethernet bridge. The down side to this is that our studies have shown that the failure mode of many Ethernet interface cards is to send out erroneous packets with good CRC's but with random Ethernet destination and source addresses. Standard bridges will pass these erroneous packets since they have not "learned" the random destination address and then add this packets random source address to their finite "learned" table. This situation is not uncommon and can greatly hinder the operation of standard bridges. If you chose to Drop unlearned packets then the KarlBridge will not forward unicast packets to Ethernet addresses that have not already been seen as a source address. This scheme works for most protocols because it relies on the characteristics of most upper-layer protocol to transmit ARP requests or Hello packets. It should be set to Drop with care by a qualified network engineer. Default: Pass

[X] Enable Learned Table Lock down

A standard bridge watches the source addresses of each packet it receives on any of its ports. As new addresses are seen, entries are added in the "learned table" that contain the particular source address and the port number that address was received on. If that source address is later seen on a different port the bridge will immediately change the port number in the learned table entry. This condition could happen in a correctly functioning network if someone moved the computer to a different part of the network. This could also happen if someone was trying to capture network packets by spoofing the bridge. Enabling learned table lock down will prevent the port number from being changed once the source address has been seen.

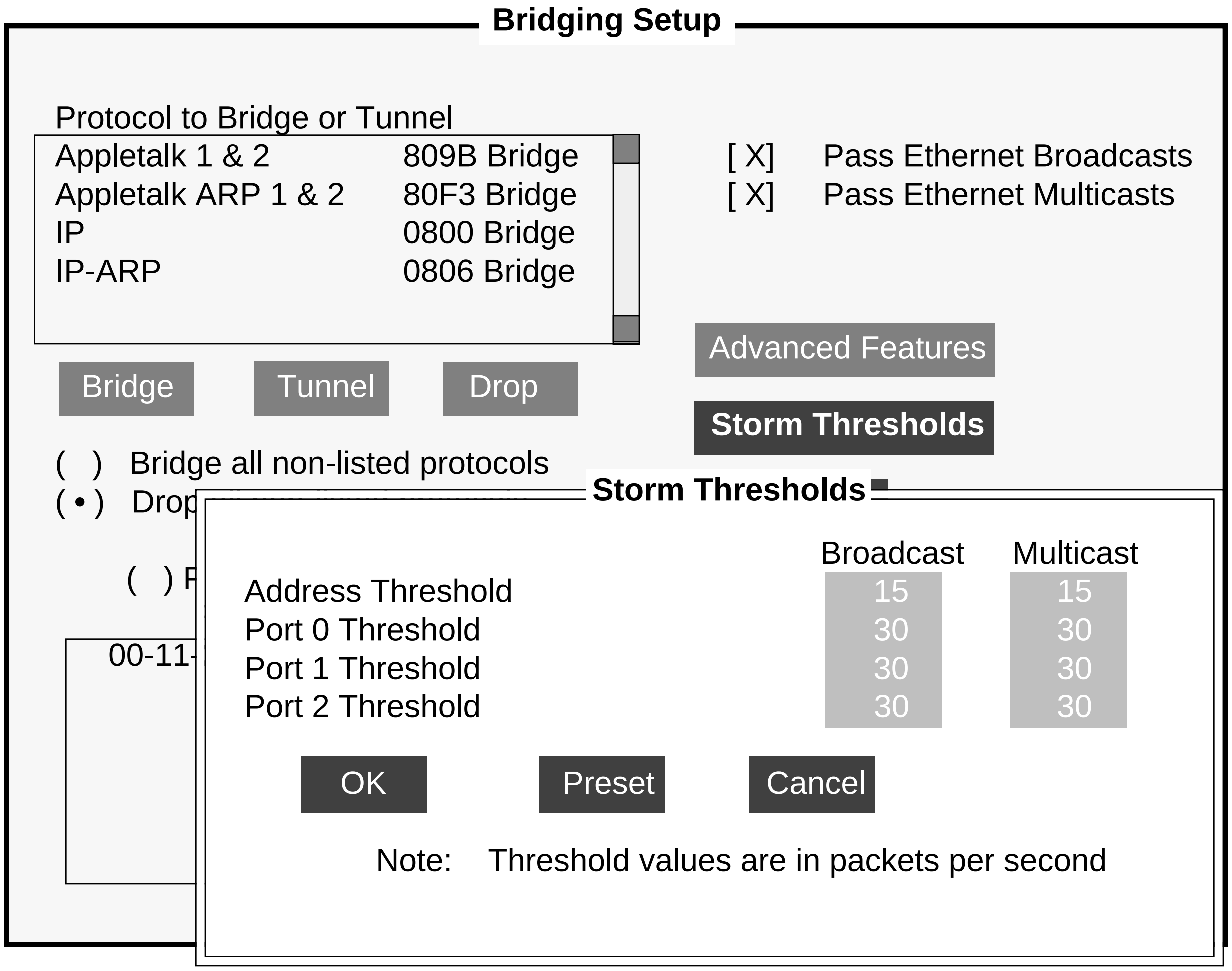
A standard bridge will also time-out the learned table records every 10 minutes. If learned table lock down is enabled then these records will not be timed out, once a record is learned it will not change or be deleted until either the bridge reboots or the learned table becomes completely filled and needs to be reset. Note: A typical KarlBridge learned table can contain over 12,000 records. Default: Disabled

[X] Enable Expanded IP ARP Support

Enabling this feature will cause the bridge to also watch the IP/ARP packets that occur on the network. No action is taken in response to an IP/ARP packet (since that is the role of an IP router) other than the bridge will add the IP address to it's IP/ARP table. This feature is helpful on an IP network because it will build a database of MAC layer address to IP address pairs. An SNMP monitoring program such as KBCONFIG can at any time extract this information. NOTE: 1) The IP/ARP table is never timed out in this mode. 2) This feature is not available if the KarlBrouter is routing IP. Default: Disabled

Storm Thresholds

One of the unique and very useful features of the KarlBridge/KarlBrouter is its ability to keep Broadcast and Multicast storms from spreading throughout a network. Network storms are common and can cause bridges, routers, workstations, servers and PC's to slow down or crash. Storms occur if network equipment is configured incorrectly, if network software is not functioning correctly, or if poorly designed programs such as network games are used.



Bridging Setup

Protocol to Bridge or Tunnel

Appletalk 1 & 2	809B Bridge
Appletalk ARP 1 & 2	80F3 Bridge
IP	0800 Bridge
IP-ARP	0806 Bridge

[X] Pass Ethernet Broadcasts
[X] Pass Ethernet Multicasts

( ) Bridge all non-listed protocols
(•) Drop

( ) F

00-11-

Storm Thresholds

	Broadcast	Multicast
Address Threshold	15	15
Port 0 Threshold	30	30
Port 1 Threshold	30	30
Port 2 Threshold	30	30

Note: Threshold values are in packets per second

Address Threshold > Broadcast

This setting determines the maximum number of broadcast packets that can occur each one second period before a storm condition is declared for a particular Ethernet address (host). Once it is determined that a storm is occurring then any additional broadcast packets from that host address will be dropped until the storm is determined to be over. The storm will be determined to be over when 30 seconds has passed where every 1 second period has less then the stated threshold in broadcast packets.

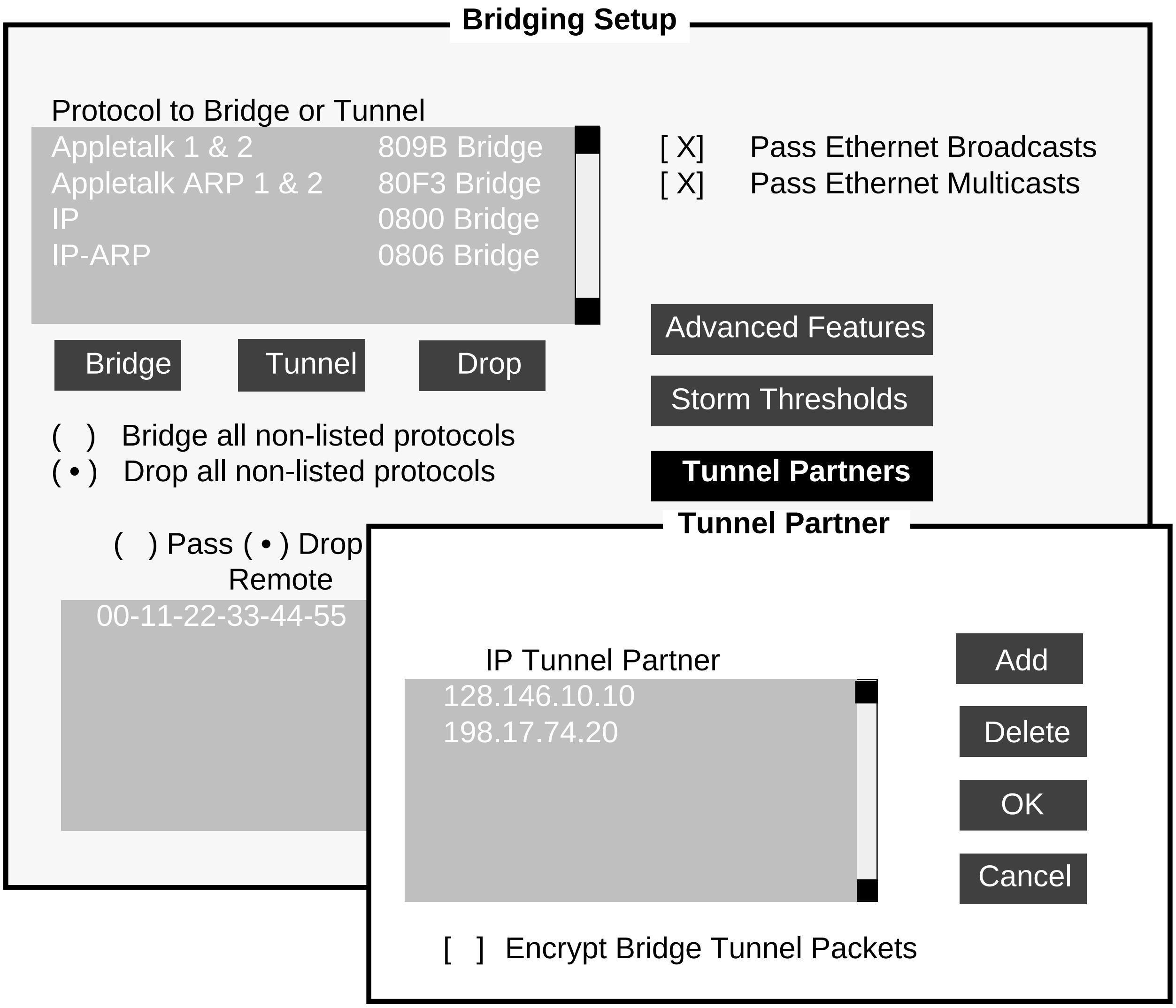
Address Threshold > Multicast
This setting determines the maximum number of multicast packets that can occur each one second period before a storm condition is declared for a particular Ethernet address (host). Once it is determined that a storm is occurring then any additional multicast packets from that host address will be dropped until the storm is determined to be over. The storm will be determined to be over once 30 seconds has passed where every 1 second period has less then the stated threshold in multicast packets.

Port Threshold > Broadcast
This setting determines the maximum number of broadcast packets that can occur each 1 second period before a storm condition is declared for a particular port. Once it is determined that a storm is occurring then any additional broadcast packets received on that port will be dropped until the storm is determined to be over. The storm will be determined to be over once a 1 second period has occurred with no broadcast packets received on that port.

Port Threshold > Multicast
This setting determines the maximum number of multicast packets that can occur each 1 second period before a storm condition is declared for a particular port. Once it is determined that a storm is occurring then any additional multicast packets received on that port will be dropped until the storm is determined to be over. The storm will be determined to be over once a 1 second period has occurred with no multicast packets received on that port.

Preset Button
This button sets the Broadcast and Multicast storm thresholds to the recommended values. These values have been determined to offer good protection without interfering with the operation of the typical network. These values may need to be tuned for your particular network.

Tunnel Partners



Protocol to Bridge or Tunnel

Appletalk 1 & 2	809B Bridge
Appletalk ARP 1 & 2	80F3 Bridge
IP	0800 Bridge
IP-ARP	0806 Bridge

[X] Pass Ethernet Broadcasts
[X] Pass Ethernet Multicasts

( ) Bridge all non-listed protocols
(•) Drop all non-listed protocols

( ) Pass (•) Drop
Remote
00-11-22-33-44-55

IP Tunnel Partner
128.146.10.10
198.17.74.20

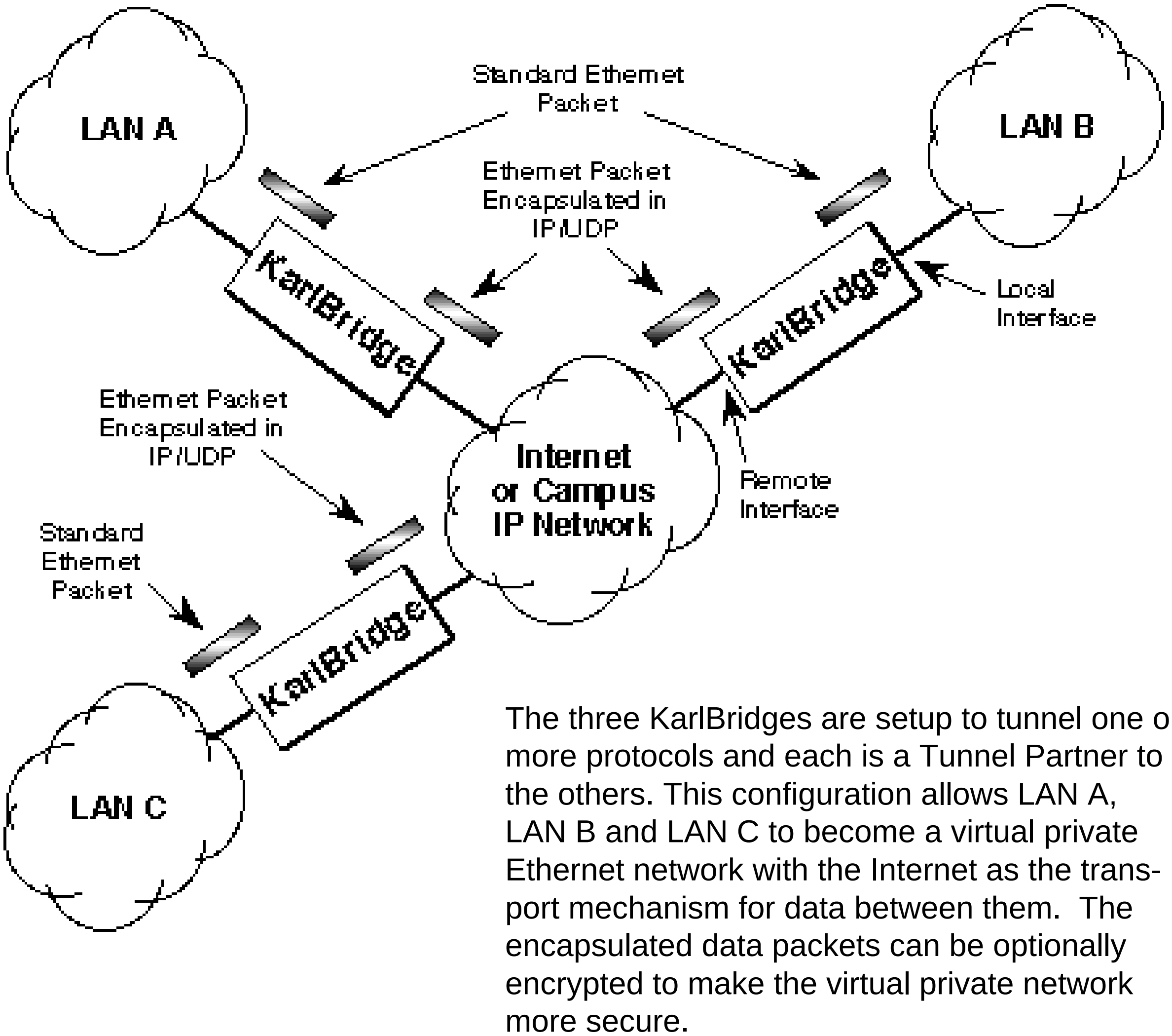
[ ] Encrypt Bridge Tunnel Packets

Tunneling is a method of encapsulating Ethernet packets, received from the "Local" port in an IP/UPD packet and sending them to one or more tunnel partners. Tunneling can be used to setup virtual Ethernet networks.

Tunnel Partners
In the General Setup menu if the "Remote Bridging using IP Tunnels" is enabled then Tunnel Partners can be setup. This menu specifies the IP addresses of each of the KarlBridge/KarlBrouters that are setup to participate in the tunnel group. Specify the addresses of all the bridges that are participating in the tunnel group but DO NOT specify the IP address of this bridge.

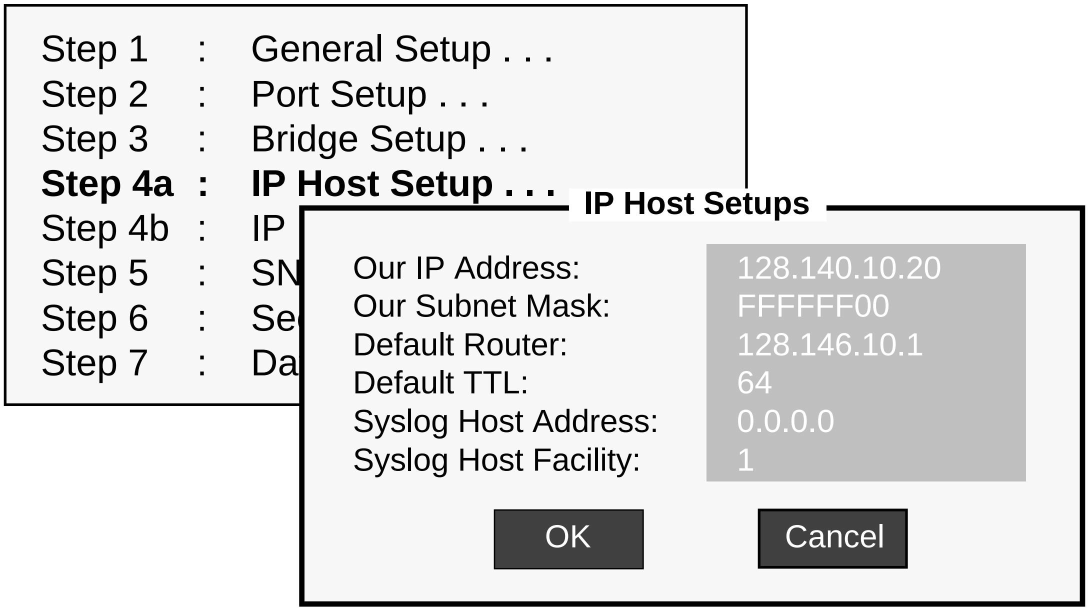
[X] Encrypt Bridge Tunnel Packets
Some KarlBridges and KarlBrouters contain a special software encryption algorithm that is distinct from the optional WaveLAN DES encryption chip on Wireless KarlBridge/KarlBrouters. If Data Encryption is enabled on the General Setup menu and if an Encryption Key is setup in the Data Encryption menu then enabling encryption here will cause all packets transmitted to tunnel partners to be encrypted and any packets received from tunnel partners to be decrypted.

Generic Ethernet Tunneling
(Through an IP Network)



The three KarlBridges are setup to tunnel one or more protocols and each is a Tunnel Partner to the others. This configuration allows LAN A, LAN B and LAN C to become a virtual private Ethernet network with the Internet as the transport mechanism for data between them. The encapsulated data packets can be optionally encrypted to make the virtual private network more secure.

STEP 4a: IP HOST SETUP



Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Step 4b : IP
Step 5 : SN
Step 6 : Se
Step 7 : Da

IP Host Setups

Our IP Address: 128.140.10.20
Our Subnet Mask: FFFFFF00
Default Router: 128.146.10.1
Default TTL: 64
Syslog Host Address: 0.0.0.0
Syslog Host Facility: 1

<u>NOTE</u>: IP Routing in the General Setup Menu must be disabled for this menu to be used.

Our IP Address

This is the IP address of the KarlBridge itself. If you wish to configure or monitor your KarlBridge or if your network supports IP and you wish to enable the Ping support and IP/SNMP support of the KarlBridge set this to a valid IP address. Setting this address to 0.0.0.0 will disable bridges Ping and IP/SNMP support. Please note that unless you enable IP Routing the KarlBridge is not an IP router. It has only one IP address and that address applies to both the Remote and Local networks (i.e. both sides of the bridge). Having two Ethernet interfaces with the same IP address is different than a standard IP host, but is appropriate for a Transparent Bridge. It is interesting to note that the Ethernet address of both ports is also the same.

Our Subnet Mask

Every IP network has what is referred to as a Subnet mask. This should be set to the appropriate mask for your network. Note that this is a hex number, hence the mask 255.255.255.0 should be specified as FFFFFF00.

Default Router

Most every IP network has a default IP router and that address should be specified here.

Default TTL

IP hosts on the Internet send out packets with a default time to live parameter. If you wish to override the factory default of 64 you can specify your new default here.

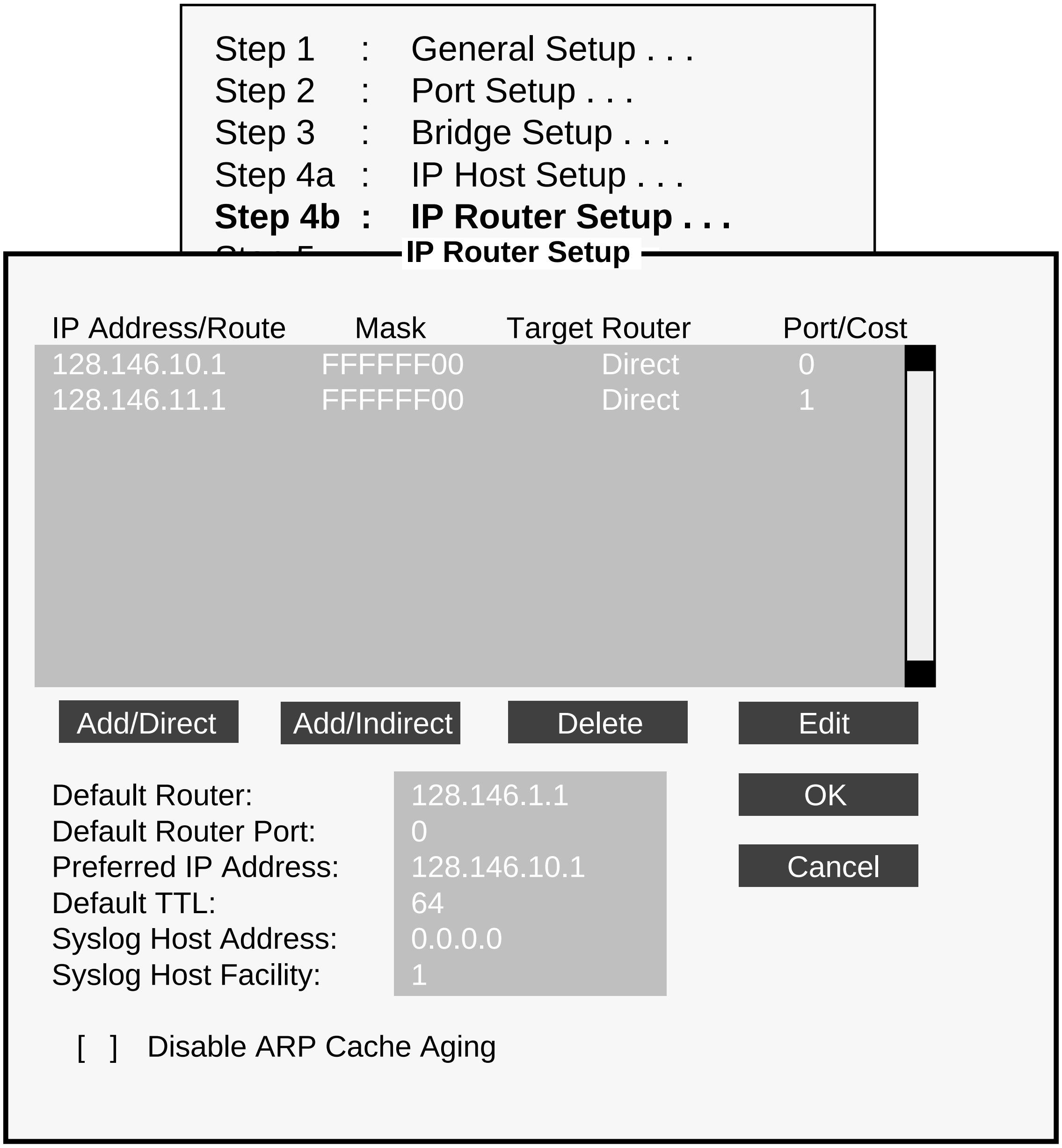
Syslog Host Address

There are many events that the KarlBridge/KarlBrouter can log. One of the places these events can be logged is on a computer equipped with the standard UNIX Syslog facility. If you want logs of this type to be kept then the IP address of the host that will take the logs must be entered here.

Syslog Host Facility

On computers that you are using to log KarlBridge/KarlBrouter events there are 7 categories of syslog messages available to you. This number specifies which category will be used.

STEP 4b: IP ROUTER SETUP


Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Step 4b : IP Router Setup . . .
Step 5
IP Router Setup
Add/Direct
Add/Indirect
Delete
Edit
OK
Cancel

IP Address/Route	Mask	Target Router	Port/Cost
128.146.10.1	FFFFFF00	Direct	0
128.146.11.1	FFFFFF00	Direct	1

Default Router:	128.146.1.1
Default Router Port:	0
Preferred IP Address:	128.146.10.1
Default TTL:	64
Syslog Host Address:	0.0.0.0
Syslog Host Facility:	1

[ ] Disable ARP Cache Aging

NOTE: IP Routing in the General Setup Menu must be enabled for this menu to be used.

Default Router (IP Address)
This entry should be set to the IP Address of the default router that this KarlBrouter is to use when it does not know where to route a particular IP packet. If the port that the default router is connected to is a serial port then this entry is ignored.

Default Router Port
This entry should be set to the port that the default router is connected to. If the port that the default router is connected to is a serial port then this defines the port that is used for the default router.

Preferred IP Address
From time to time the KarlBrouter will transmit unsolicited IP packets such as SNMP Traps, Syslog, RIP or IP ARP packets. Most routers randomly use one of the IP addresses from one of the router ports as the source IP address for these packets. On the KarlBrouter you can specify the source IP address that you prefer to use for these packets.

Default TTL
IP hosts on the Internet send out packets with a default time to live parameter. If you wish to override the factory default of 64 you can specify your new default here.

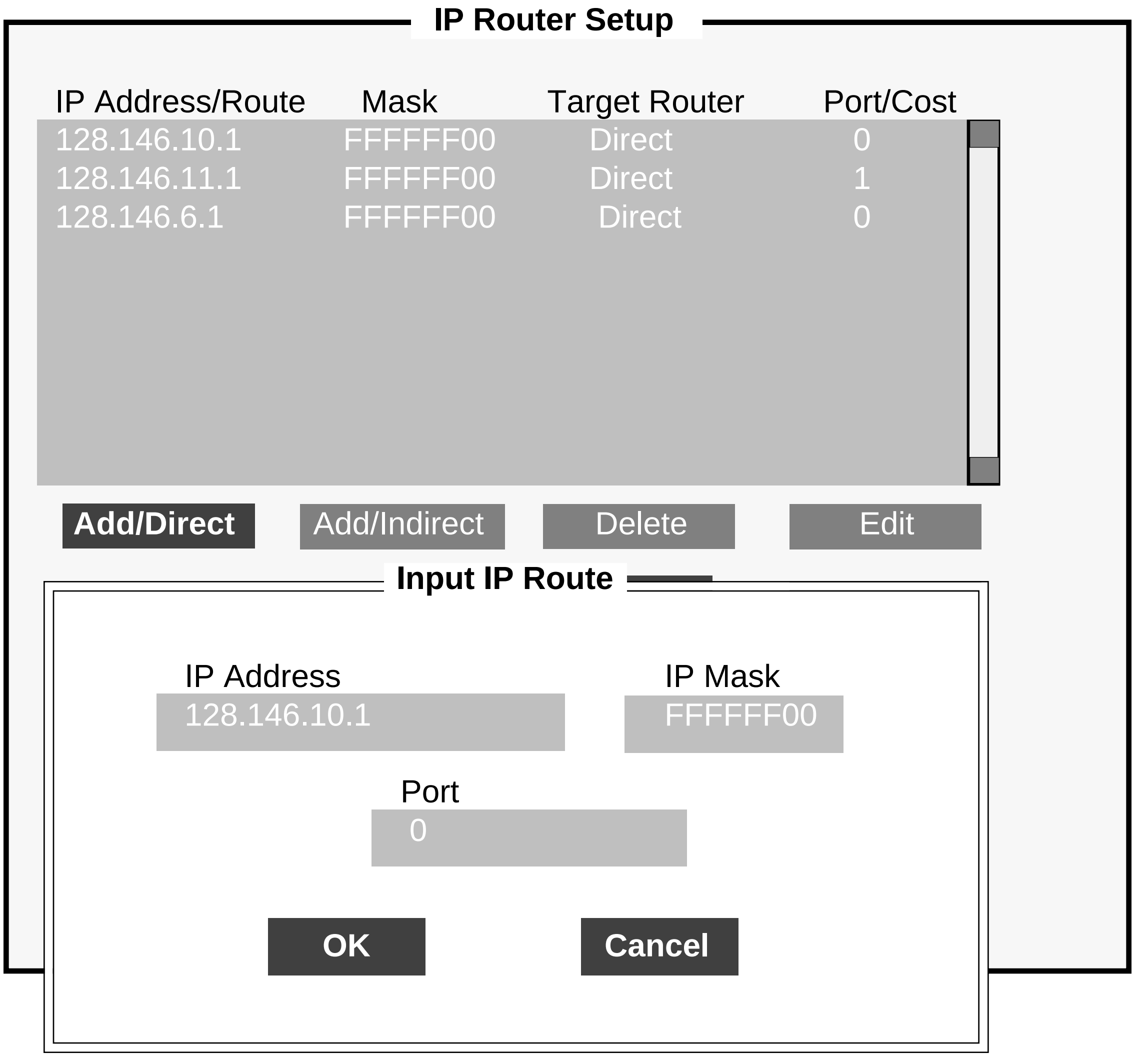
Syslog Host Address
There are many events that the KarlBridge/KarlBrouter can log. One of the places these events can be logged is on a computer equipped with the standard UNIX Syslog facility. If you want logs of this type to be kept then the IP address of the host that will take the logs must be entered here.

Syslog Host Facility
On computers that you are using to log KarlBridge/KarlBrouter events there are 7 categories of syslog messages available to you. This number specifies which category will be used.

[X] Disable ARP Cache Aging
Use this option if you want to keep a permanent record of the IP to Ethernet addresses table for each computer directly connected to a port on this KarlBrouter. This feature is helpful when used in conjunction with a corporate wide SNMP monitoring tool to create a database of all Ethernet to IP address combinations on your network. A standard IP router and the KarlBrouter will age it's ARP cache entries. It will time-out and delete the ARP entries after a certain specified period (usually 10 minutes). The KarlBrouter has the option of not aging (deleting) any ARP cache entries. This will not normally cause any IP network problems but could result in a large ARP cache table. Since the typical KarlBrouter can hold over 10,000 ARP entries this is not normally a problem.

Add/Direct

This button activates a menu which is used to specify the "direct" routes for each of the ports on the KarlBrouter. Direct routes are those that are directly connected to the ports. As an example if port 0 is to have subnet 128.146.6.X connected to it and an IP address of 128.146.6.1 with a subnet mask of 255.255.255.0 then an entry in this menu should be setup as: IP Address = 128.146.6.1; IP Mask = FFFFFF00; and Port = 0.



IP Router Setup

IP Address/Route	Mask	Target Router	Port/Cost
128.146.10.1	FFFFFF00	Direct	0
128.146.11.1	FFFFFF00	Direct	1
128.146.6.1	FFFFFF00	Direct	0

Input IP Route

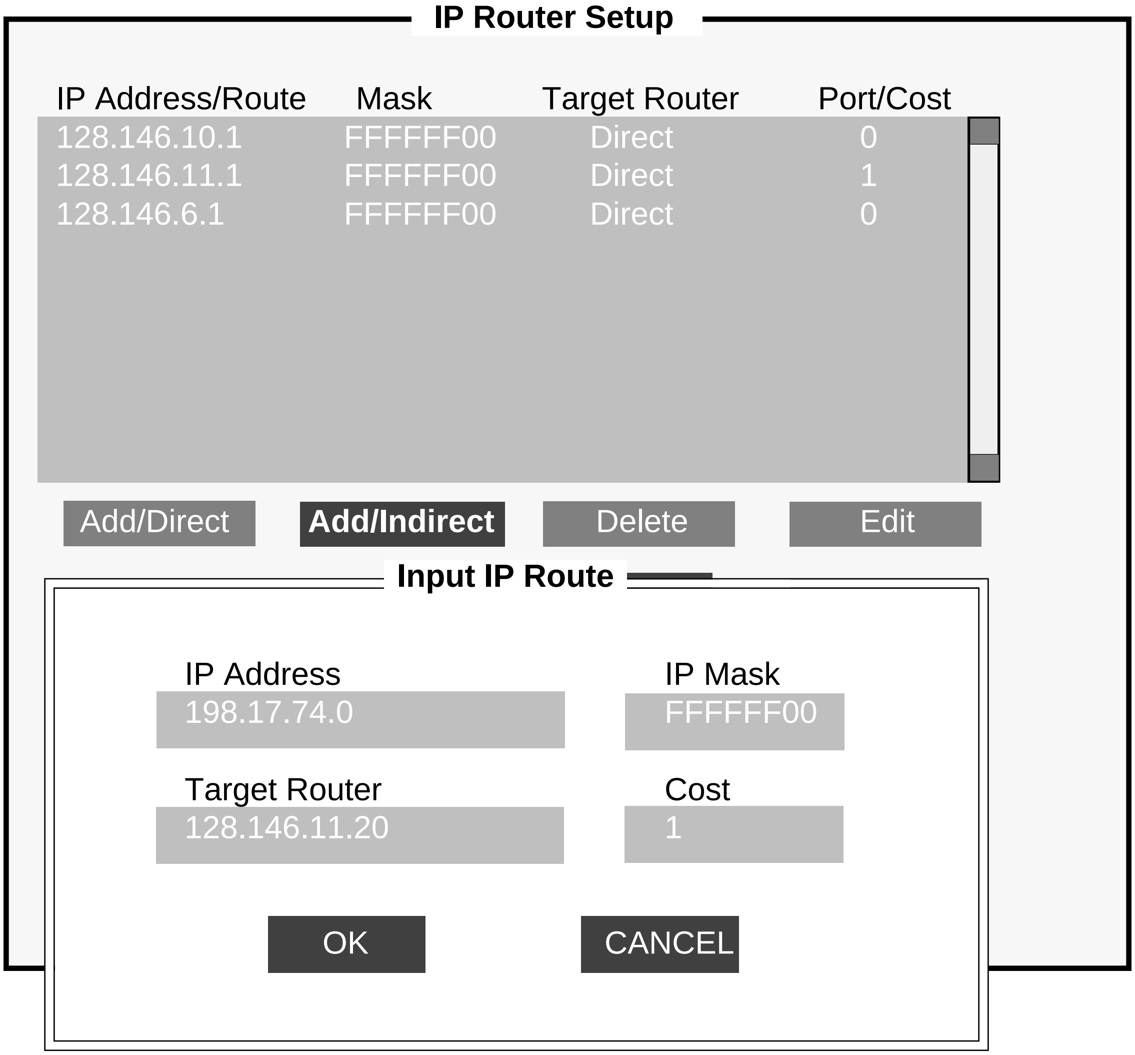
IP Address: 128.146.10.1

IP Mask: FFFFFF00

Port: 0

Add/Indirect

This button activates a menu which is used to specify the "indirect" routes for this KarlBrouter. These routes are sometime referred to as static routes. You can use indirect routes to define the way to get to subnets that are attached to other routers in your network. As an example, if subnet 198.17.74.0 is attached to a router 128.146.11.20 in order for this KarlBrouter to route packets to 198.17.74.1 you should specify an entry that is setup as: IP Address = 198.17.74.0; IP Mask = FFFFFF00; Next Hop = 128.146.11.20 with a Cost = 1.



IP Address/Route	Mask	Target Router	Port/Cost
128.146.10.1	FFFFFF00	Direct	0
128.146.11.1	FFFFFF00	Direct	1
128.146.6.1	FFFFFF00	Direct	0

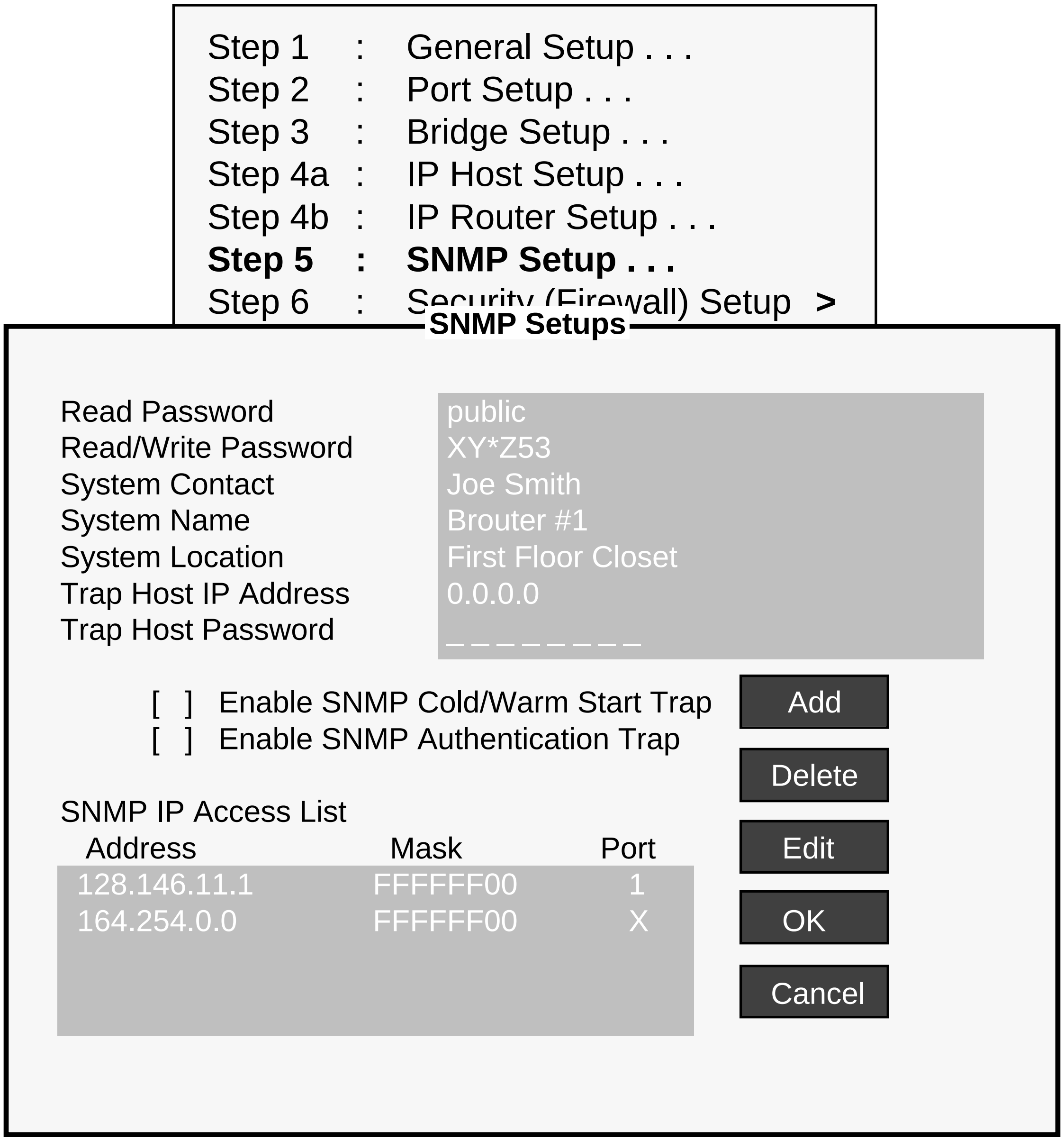
IP Address: 198.17.74.0

IP Mask: FFFFFF00

Target Router: 128.146.11.20

Cost: 1

Step 5: SNMP SETUP


Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Step 4b : IP Router Setup . . .
Step 5 : SNMP Setup . . .
Step 6 : Security (Firewall) Setup >
SNMP Setups
Add
Delete
Edit
OK
Cancel

Read Password: public
Read/Write Password: XY*Z53
System Contact: Joe Smith
System Name: Brouter #1
System Location: First Floor Closet
Trap Host IP Address: 0.0.0.0
Trap Host Password: _ _ _ _ _ _ _ _

[ ] Enable SNMP Cold/Warm Start Trap
[ ] Enable SNMP Authentication Trap

SNMP IP Access List

Address	Mask	Port
128.146.11.1	FFFFFF00	1
164.254.0.0	FFFFFF00	X

Read Password
This is the read only password used for SNMP support. It is the SNMP password needed to read the Flash ROM Configuration and SNMP MIB Variables. The factory default value for this variable is the string *public*.

Read/Write Password
This is the read/write password used for SNMP support. It is the SNMP password needed to write the Flash ROM configuration and SNMP MIB variables in to the bridge/ router. The string should be set to a value that is known only by you. The factory default value for this variable is the string *public* and should be changed to a string known only to you.

System Contact

This field should contain the identification of the contact person for this SNMP managed node, (i.i., this bridge/router) together with information on how to contact this person.

System Name

This field should contain the administratively assigned name for this managed node. By convention, this is the node's fully-qualified Internet domain name(ex: bridge20.karlnet.com).

System Location

This field should contain the physical location of this node (e.g.,`telephone closet, 3'rd floor').

Trap Host IP Address

This is the IP address of a network connected host that is setup to receive SNMP Trap messages from this bridge/router. If you do not have an SNMP Trap host then set this to 0.0.0.0.

Trap Host Password

This is the SNMP read/write password (community name) of the host that is setup to receive SNMP Trap messages. This field is ignored if the Trap Host IP Address described above is 0.0.0.0.

[X] Enable SNMP Cold/Warm Start Trap

If Cold/Warm Start traps are enabled then an SNMP Trap will be sent to the trap host whenever this bridge/router powers up, is restarted because of an internal software error, has just completed a Flash ROM reprogram and restart cycle, or reboots because the watchdog timer expired. Please see "Enable Watchdog Reboot Timer" under the General Setup Menu.

[X] Enable SNMP Authentication Traps

If SNMP authentication Traps are enabled adn a Trap Host is setup properly then an SNMP Trap will be sent to the to the trap host whenever an SNMP request is made of the bridge/router where the password (community name) is wrong.

SNMP IP Access List

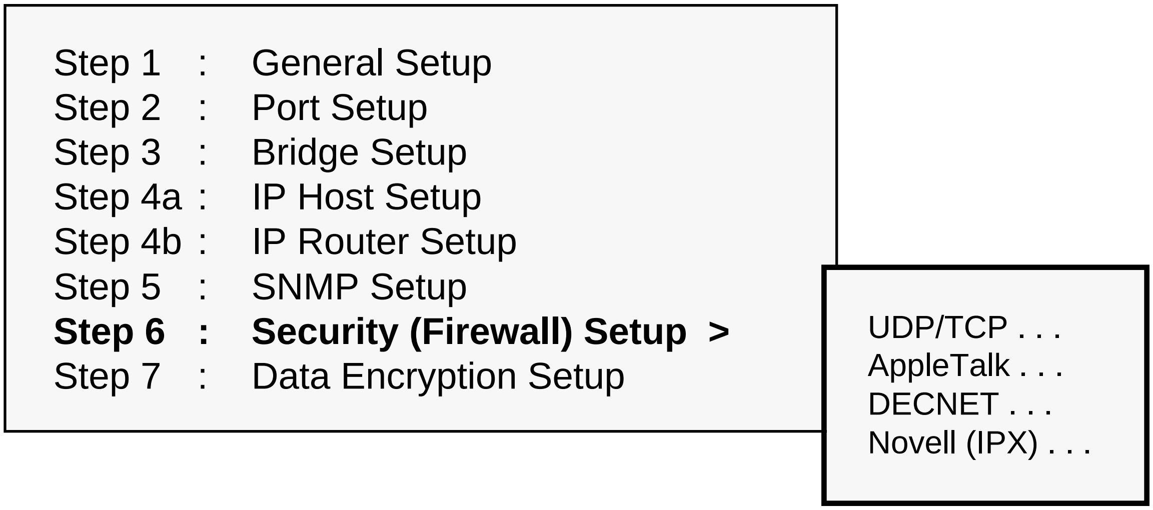
You can optionally setup a list of networks, subnets and hosts that are authorized to access the KarlBridge/KarlBrouter via SNMP. SNMP access lists are used in conjunction with well picked SNMP passwords and the special SNMP hardware protection jumpers to prohibit unauthorized access into the Flash ROM configuration database of this bridge/router.

Examples:

1. IP Address: 128.146.11.0 Mask: FFFFFF00   Port: 1 will only allow SNMP access from the Network 128.146.11.x and only if the SNMP request was made from the portion of   the network attached to Port 1.

2. IP Address: 164.254.0.0 Mask: FFFF0000 Port: X will only allow SNMP access from the network 164.254.x.x received from any port.

STEP 6: SECURITY (FIREWALL) SETUP



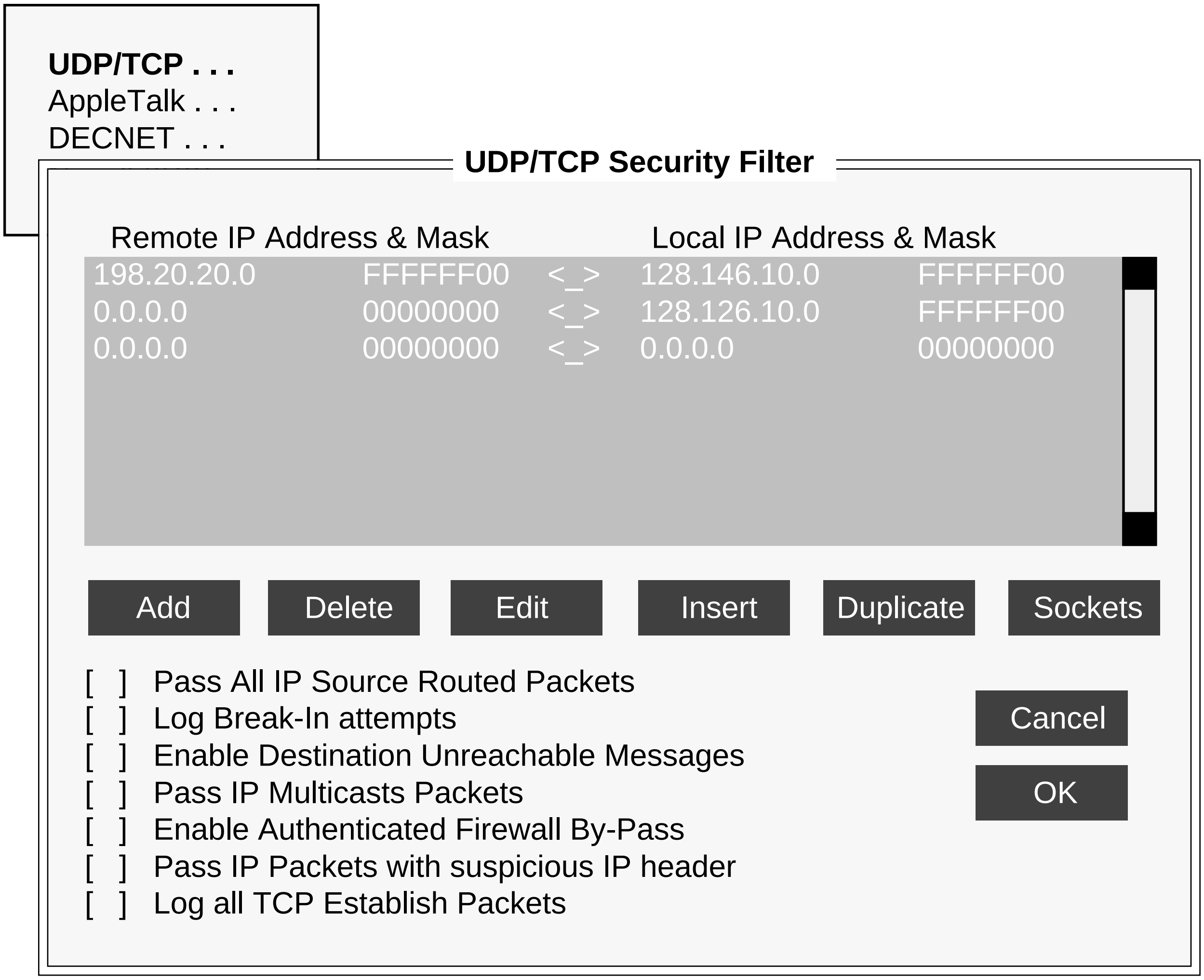
Security firewalls are enabled in the "General Setup" menu. If Security Filters are enabled and if the protocols that have security firewall capability (i.e. IP/UDP/TCP, AppleTalk, DECNET, or Novell IPX) are enabled to be passed through the bridge/ brouter then additional protection is added with these protocols. Security filters will cause the KarlBridge/KarlBrouter to analyze on the application level each packet to determine if it should be passed or dropped.

Remote & Local Menus
Some of these menus are marked "Remote" and some are marked "Local". Remote menus configure filters that pertain to networks, subnets, and/or hosts that are connected to the Remote network (i.e. the Remote port of the KarlBridge/KarlBrouter). Local menus configure filters that pertain to network, subnets, and/or hosts that are connected to the Local network (i.e. the Local port of the KarlBridge/KarlBrouter). You can determine weather a port is remote and local by looking at the Port Setup Menu.

Pass or Drop Menu modes:
The menus can be in a mode to either pass (permit) or drop (deny) their items. The concept is that in most situations one wants to either drop a few selected items or to pass a few selected items of each type. If the menu is EMPTY and is set-up to "Pass Following..." then all packets of that type will be dropped. This is because you are passing an empty menu therefore nothing will be passed. If the menu is EMPTY and is set-up to "Drop Following..." then all packets of that type will be passed.

IP/UDP/TCP Security Filter

(This will only appear if IP is being bridged or routed)



```
Remote IP Address & Mask               Local IP Address & Mask
198.20.20.0       FFFFFF00     <_>    128.146.10.0       FFFFFF00
0.0.0.0           00000000     <_>    128.126.10.0       FFFFFF00
0.0.0.0           00000000     <_>    0.0.0.0            00000000
```

[ ] Pass All IP Source Routed Packets
[ ] Log Break-In attempts
[ ] Enable Destination Unreachable Messages
[ ] Pass IP Multicasts Packets
[ ] Enable Authenticated Firewall By-Pass
[ ] Pass IP Packets with suspicious IP header
[ ] Log all TCP Establish Packets

Remote/Local IP Address Menu & Mask

This menu specifies the IP network, subnet, and/or single machine that is to have its IP packets passed, dropped, logged, or encrypted. Each packet's IP source and destination address is checked against each entry in the list to determine what action should be performed on the packet. *Matching is performed on the first entry first* and then goes down the list looking for the first match. When a match is found the action specified by the socket menus for that line is performed immediately. The packet's IP addresses are logically "anded" with the mask and then compared with the IP address to determine if a match has occurred.

NOTE: This menu specifies the IP networks, IP subnets and IP Hosts on the remote network that hosts on the local network can communicate with. This menu does not specify IP routes and is not used to setup IP Routing.

[X] Pass All IP Source Routed Packets
Source routed packets are special IP packets that are rarely used. There are certain situations where they can also be used by hackers to spoof firewalls. You should set this to *drop* unless you know you need to pass source routed packets.

[X] Log Break-In Attempts
Enabling the logging of break-in attempts will cause a Syslog packet to be sent to the Syslog server each time the security filter module detects and drops a packet.

[X] Enable Destination Unreachable Messages
Destination unreachable messages are normally sent by routers when a packet is unable to be delivered to it's final destination due to one of several reasons. If the dropped packet is a UDP packet then usually an ICMP Destination Unreachable packet is sent to the originator of the dropped IP packet. If the packet is a TCP packet then a TCP Reset packet is usually sent. If you enable this feature then the KarlBridge/KarlBrouter's security module will send either an ICMP destination unreachable packet or a TCP Reset packet to the originator of the dropped packet. This feature is helpful because software such as telnet will quickly detect that a connection cannot be made. This feature is helpful but can also tip off a potential hacker that a security firewall is being used.

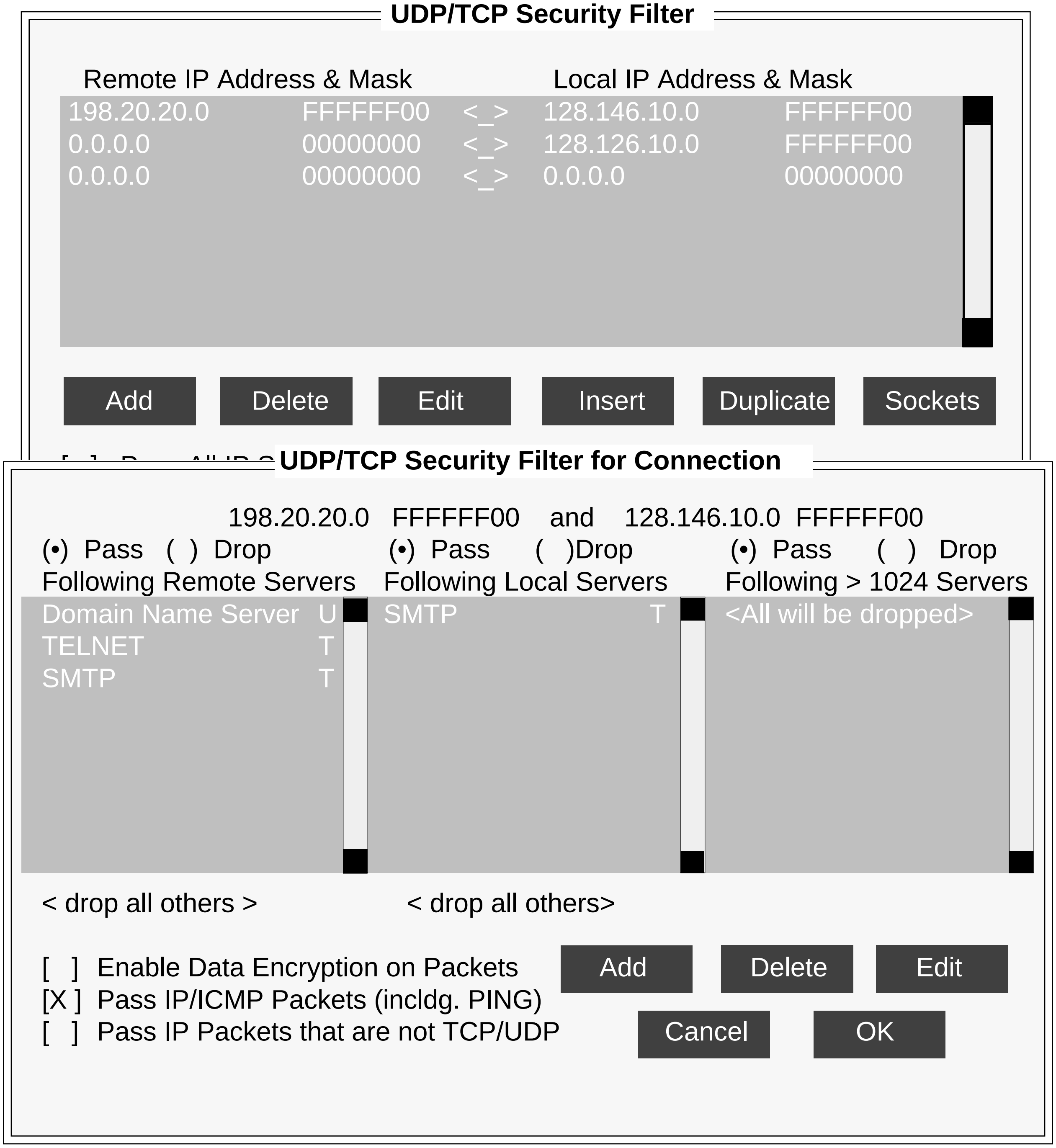
[X] Pass IP Multicast Packets
IP multicast packets are normally used for M-Bone audio and video data transmissions on a local network. IP multicast packets will penetrate through bridges and can cause abnormal behavior on some network attached computers. It is recommended that you Drop IP multicast packets unless you know you need them.

[X] Enable Authenticated Firewall By-Pass
The KarlBridge/KarlBrouter's UDP/TCP firewall filters can be dynamically bypassed. This feature enables data between particular subnets or hosts to flow through the firewall untouched by any security filters. This feature is very powerful and can be used to create a way to authenticate access by logging into a particular network or host. If enabled this feature can also be used by a hacker to gain unauthorized access to your network. If you enable this feature you must take great care to setup SNMP passwords and access lists to prevent such unauthorized tampering with your firewall.

[X] Pass IP Packets with suspicious IP header
If you set this to "drop" then each IP packet that passes through the KarlBridge/KarlBrouter is checked for inconsistencies in its IP header. If an anomaly is found the packet is dropped.

[X] Log all TCP Establish Packets

Each IP/TCP packet that travels through the bridge/router is checked to see if it is the special TCP/IP SYN packet. This type of packet is always sent in a TCP/IP network to initiate a TCP connection. As an example when the Telnet client attempts to connect to a Telnet server it sends a TCP SYN packet. If you enable this setting a SYSLOG message will be sent to the SYSLOG server each time a TCP program attempts to connect to another TCP program such as the Telnet or FTP server.



UDP/TCP Security Filter

Remote IP Address & Mask			Local IP Address & Mask	
198.20.20.0	FFFFFF00	<_>	128.146.10.0	FFFFFF00
0.0.0.0	00000000	<_>	128.126.10.0	FFFFFF00
0.0.0.0	00000000	<_>	0.0.0.0	00000000

[ ] Pass All IP S

UDP/TCP Security Filter for Connection

198.20.20.0 FFFFFF00 and 128.146.10.0 FFFFFF00

(•) Pass ( ) Drop Following Remote Servers		(•) Pass ( )Drop Following Local Servers		(•) Pass ( ) Drop Following > 1024 Servers
Domain Name Server	U	SMTP	T	<All will be dropped>
TELNET	T			
SMTP	T			
< drop all others >		< drop all others>		

[ ] Enable Data Encryption on Packets
[X] Pass IP/ICMP Packets (incldg. PING)
[ ] Pass IP Packets that are not TCP/UDP

Once a packets source and destination IP address matches an entry in the Remote/ Local IP Address Menu the UDP/TCP sockets are tested against this menu to determine if the packet is to be passed or dropped.

Following Remote Servers

This menu specifies which sockets with values less then 1024 on computers connected to the remote port are to be passed and which are to be dropped.

Following Local Servers

This menu specifies which sockets with values less then 1024 on computers connected to the local port are to be passed and which are to be dropped.

Following > 1024 Servers

This menu specifies which sockets with values greater then or equal to 1024 on computers connected to either the local or remote port are to be passed and which are to be dropped.

[X] Enable Data Encryption on Packets

After a packets source and destination IP address matches an entry in the Remote/ Local IP Address Menu then he data portion of the UDP or TCP packet can be optionally encrypted (if received on the local port and destined for the remote port) or decrypted (if received on the remote port and destined for the local port). You can specify the encryption/decryption key on the Setup - Data Encryption Menu.

[X] Pass IP/ICMP Packets (including Ping)

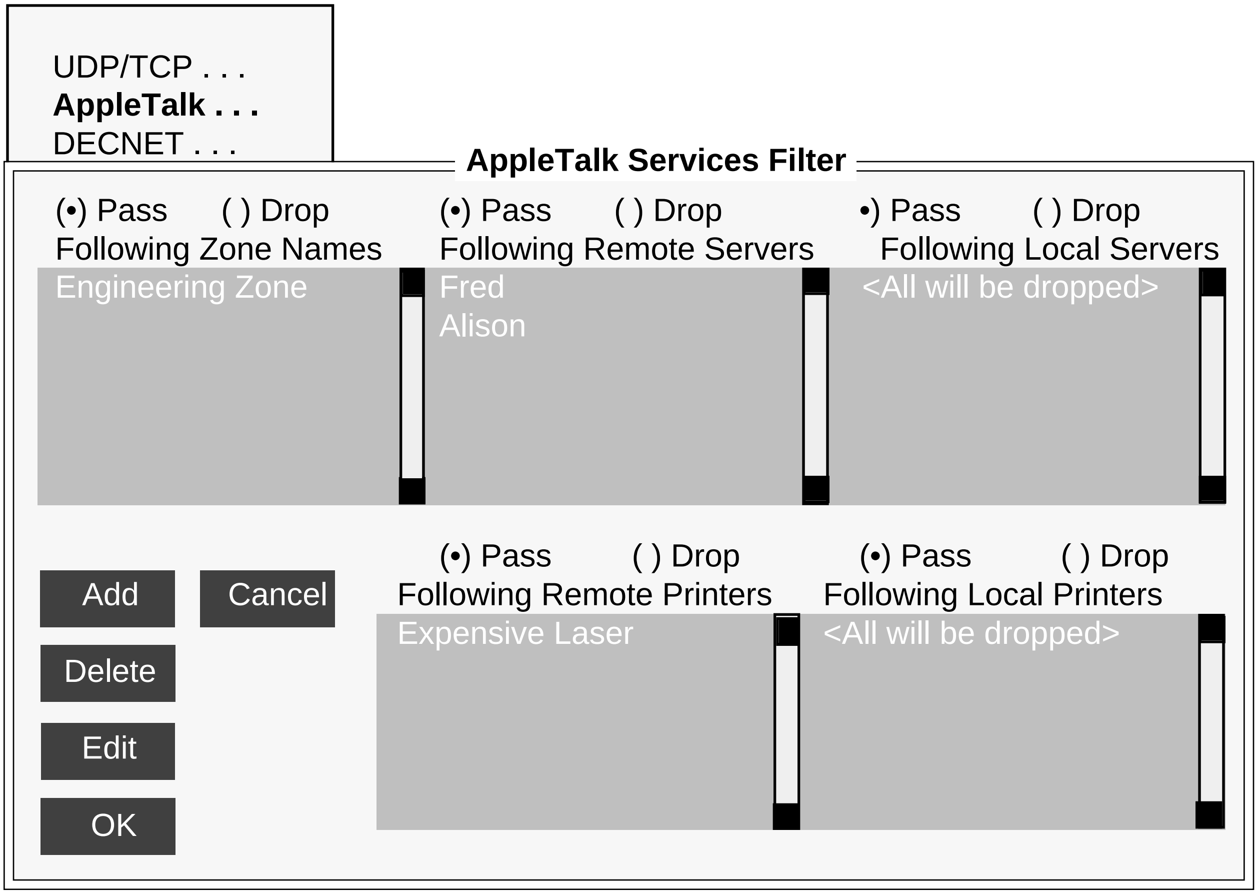
After a packets source and destination IP address matches an entry in the Remote/ Local IP Address Menu then it can be tested to see if it is an ICMP packet. You can optionally drop any ICMP packets to/from the matched IP addresses. This is helpful if you wish to allow ping packets to pass through the firewall. You can drop all ICMP (including Ping) packets if you wish to hide the computers on the other side of the firewall from potential hackers using ping to discover their existence.

[X] Pass IP Packets that are not TCP/UDP

If a packets source and destination IP address matches an entry in the Remote/Local IP Address Menu and if it is either TCP or UDP its socket number will be tested to see if it should be passed or dropped. If the packet is not UDP nor TCP then a decision must be made what to do with the packet since it does not have a socket number. Most IP packets are UDP or TCP with the exception of IGP. Since most LANs do not use IGP it is best to drop packets that are not UDP/TCP. This is helpful so keep hackers from sending non-UDP and non-TCP packets through the firewall.

APPLETALK FILTERS

(Will only appear if AppleTalk is being bridged)



AppleTalk Services Filter

(•) Pass ( ) Drop
Following Zone Names
Engineering Zone

(•) Pass ( ) Drop
Following Remote Servers
Fred
Alison

•) Pass ( ) Drop
Following Local Servers
<All will be dropped>

(•) Pass ( ) Drop
Following Remote Printers
Expensive Laser

(•) Pass ( ) Drop
Following Local Printers
<All will be dropped>

When Macintosh's are networked together, one of the undesirable side effects is that all Macintosh's can "see" in their Choosers all servers and all printers that are connected to the network. If multiple zones are specified then there is some form of protection but a user needs to only specify a zone and then can choose a printer to print to anywhere in the network. These menus will configure the KarlBridge to selectively restrict access to specified Apple servers and/or Apple printers. The KarlBridge is not an AppleTalk router. It does not have any of the characteristics of an AppleTalk router. The KarlBridge is simply a bridge that for AppleTalk can promote or prohibit the appearance of server and/or printer names in the chooser.

CAUTION: It is common characteristic of AppleTalk networks with multiple routers to have configuration problems if all of the routers do not agree on zone names and networks numbers. The KarlBridge is not an AppleTalk Router, it does not contribute to this problem. These menus will not, however, remedy this problem. If you wish to isolate a local AppleTalk network from a remote AppleTalk network you must be sure to drop AppleTalk and AppleTalk ARP in the "Ethernet Protocol Menu".

(•) Pass ( ) Drop Apple Zone Name Menu:
This menu specifies the AppleTalk Zone names that are to be passed or dropped. Each of the Apple Zones can be named in this menu. The menu entry * (single asterisk) is the standard AppleTalk code that means "my Zone". As an example; if the Local LAN's Zone name is Tiger and if you wish to see in your chooser printers and servers from a Remote LAN with the Zone name Tiger, then two entries must appear in this menu, the string Tiger and on the next line an *. This is because sometimes AppleTalk explicitly asks for printers and servers in the Zone Tiger and sometimes it uses the * as shorthand for Tiger (i.e. "my Zone").

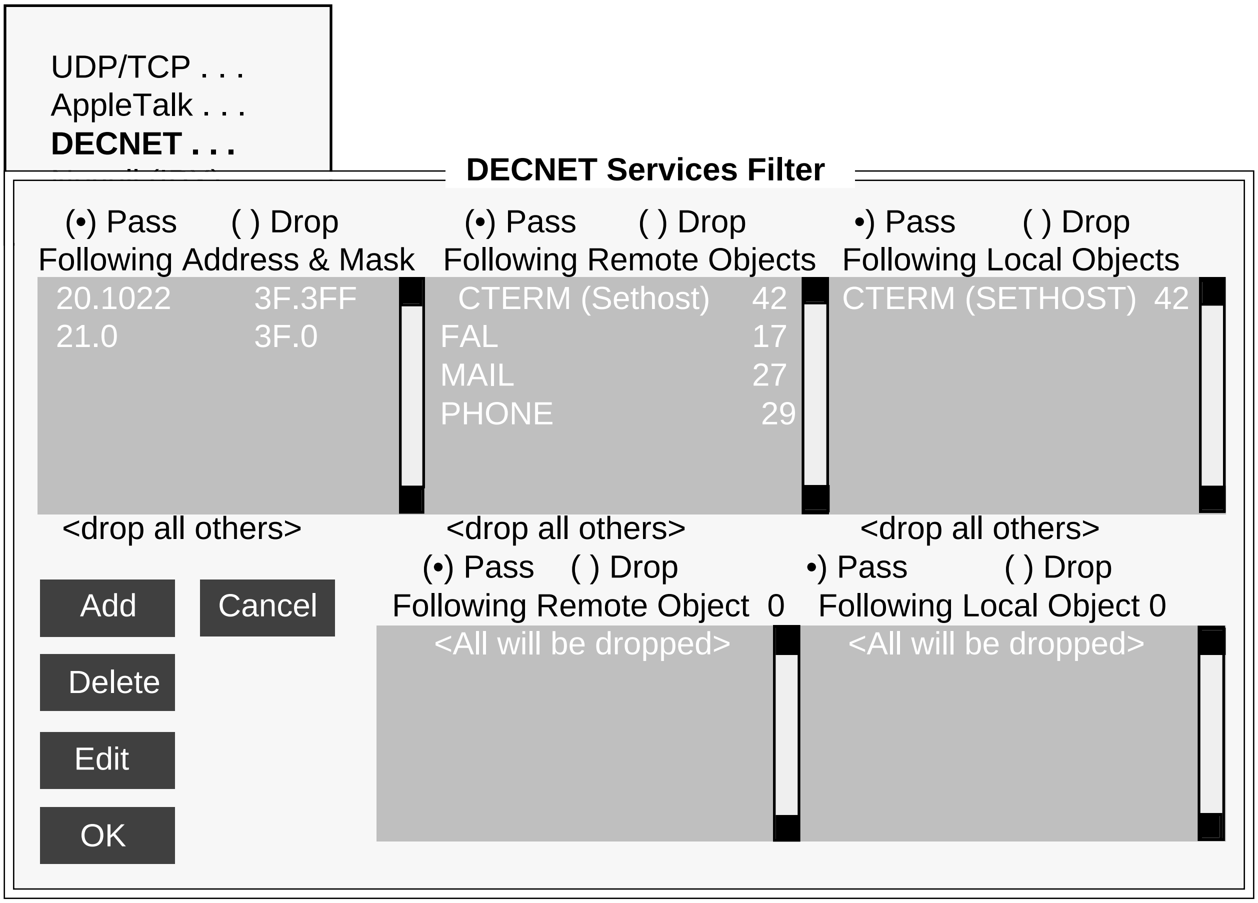
(•) Pass ( ) Drop Apple Remote Servers Menu:
This menu specifies the Remote file servers that are to appear in the Local LAN's Macintosh Choosers, regardless of Zone. If the Local LAN's Macintoshes are not to see any Remote file servers then this menu should be set to "Pass Apple Remote Servers" with no entries in it. This will force the KarlBridge to pass none of the Remote file server names to the Local LAN. If all Remote file servers are to be seen by the Local LAN then this menu should be empty and set to "Drop Apple Remote Servers".

(•) Pass ( ) Drop Apple Local Servers Menu:
This menu specifies the Local file servers that are to appear in the Remote LAN's Macintosh Choosers, regardless of Zone. If the Remote Macintoshes are not to see any Local file servers then this menu should be set to "Pass Apple Local Servers" with no entries in it. This will force the KarlBridge to pass none of the Local LAN's file server names to the Remote network. If all of the Local file servers are to be seen by the Remote network then this menu should be empty and set to "Drop Apple Local Servers".

(•) Pass ( ) Drop Apple Remote Printers Menu:
This menu specifies the Remote printers that are to appear in the Local LAN's Macintosh Choosers, regardless of Zone. If the Local LAN's Macintoshes are not to see any Remote printers then this menu should be set to "Pass Apple Remote Printers" with no entries in it. This will force the KarlBridge to pass none of the Remote printer names to the Local LAN. If all Remote printers are to be seen by the Local LAN then this menu should be empty and set to "Drop Apple Remote Printers".

(•) Pass ( ) Drop Apple Local Printers Menu:
This menu specifies the Local printers that are to appear in the Remote LAN's Macintosh Choosers, regardless of Zone. If the Remote Macintoshes are not to see any Local printers then this menu should be set to "Pass Apple Local Printers" with no entries in it. This will force the KarlBridge to pass none of the Local LAN's printer names to the Remote network. If all of the Local printers are to be seen by the Remote network then this menu should be empty and set to "Drop Apple Local Printers".

DECNET FILTERS
(Will only appear if DECNET is being bridged)



UDP/TCP . . .
AppleTalk . . .
DECNET . . .

DECNET Services Filter

(•) Pass ( ) Drop
Following Address & Mask

```
20.1022        3F.3FF
21.0           3F.0
```

<drop all others>

(•) Pass ( ) Drop
Following Remote Objects

```
CTERM (Sethost)    42
FAL                17
MAIL               27
PHONE              29
```

<drop all others>

•) Pass ( ) Drop
Following Local Objects

```
CTERM (SETHOST)  42
```

<drop all others>

(•) Pass ( ) Drop
Following Remote Object 0
<All will be dropped>

•) Pass ( ) Drop
Following Local Object 0
<All will be dropped>

(•) Pass ( ) Drop Following Address & Mask Menu:
This menu specifies the DECNET Areas and Hosts that are to be passed or dropped. Each entry consists of a DECNET Address and an special Mask; a packet that matches is then either passed or dropped as specified. Each DECNET packet's source and destination address is checked against each entry in the list to determine if the packet is to be passed or dropped. Matching is performed on the first entry first and then goes down the list. When a match is found the action specified on that line is performed immediately. The packet's DECNET addresses are logically "anded" with the mask and then compared with the IP address to determine if a match has occurred. Addresses are specified in the standard DECNET syntax: Area.Host. The special mask is a hexadecimal number that specifies a bit mask to be "anded" with the packet's DECNET address prior to being comparing with the specified DECNET address.

NOTE: The KarlBridge is not a DECNET Router. This menu specifies the DECNET hosts and/or DECNET areas that hosts on either the local or remote network can communicate with.

(•) Pass ( ) Drop Remote Objects Menu:
This menu specifies the DECNET Objects on remote DECNET hosts that are to be passed or dropped. Each DECNET connect packet is checked against each entry in the list to determine if the packet is to be passed or dropped.

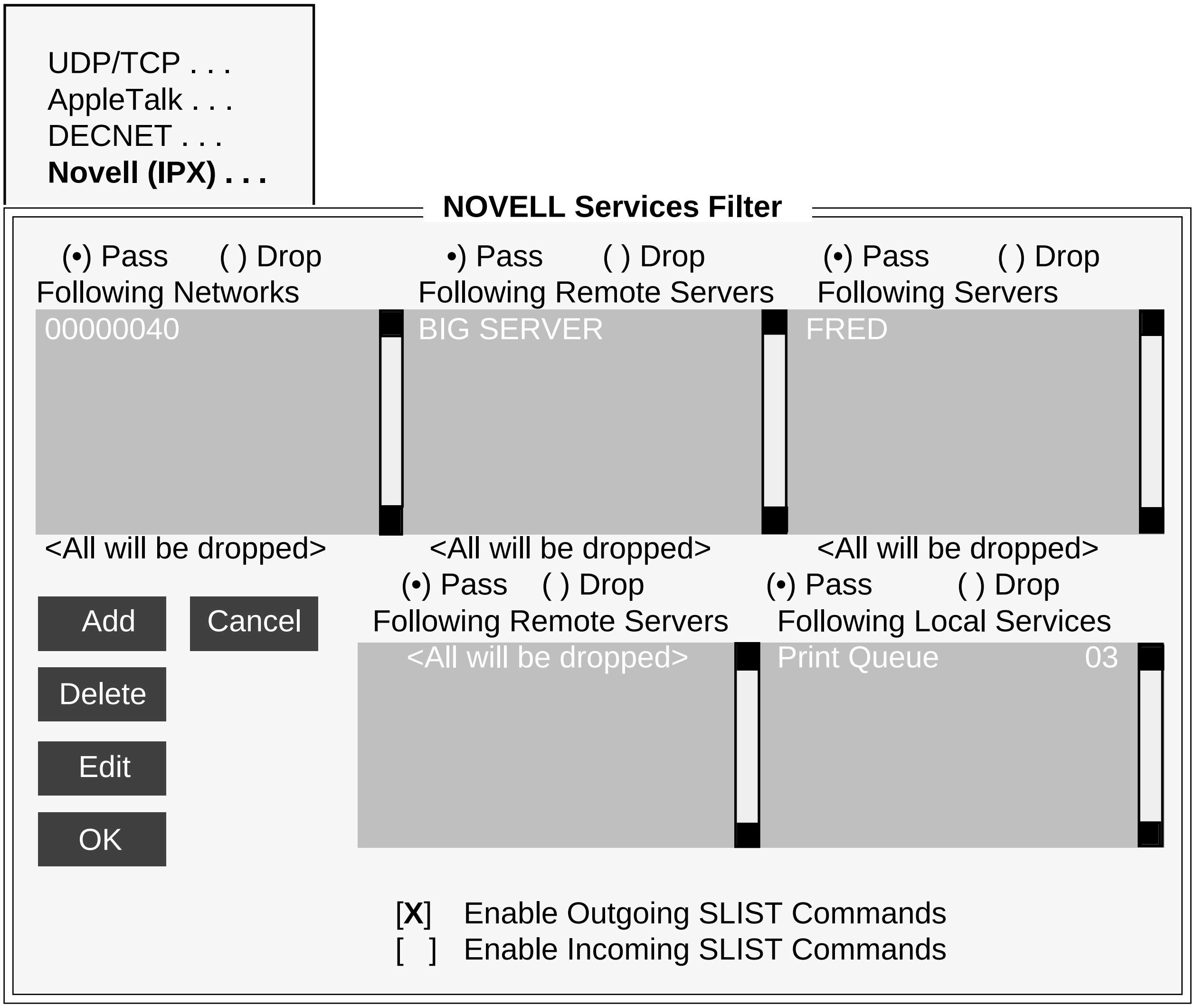
(•) Pass ( ) Drop Remote Object 0 Menu:
This menu specifies the DECNET Object 0 names on remote hosts that are to be passed or dropped. Each DECNET connect packet to DECNET Object 0 is checked against each entry in the list to determine if the packet is to be passed or dropped.

(•) Pass ( ) Drop Local Objects Menu:
This menu specifies the DECNET Objects on the local hosts that are to be passed or dropped. Each DECNET connect packet is checked against each entry in the list to determine if the packet is to be passed or dropped.

(•) Pass ( ) Drop Local Object 0 Menu:
This menu specifies the DECNET Object 0 names that are to be passed or dropped. Each DECNET connect packet to DECNET Object 0 is checked against each entry in the list to determine if the packet is to be passed or dropped.

NOVELL (IPX) FILTERS

(Will only appear if Novell is being bridged)



UDP/TCP . . .
AppleTalk . . .
DECNET . . .
Novell (IPX) . . .

NOVELL Services Filter

(•) Pass ( ) Drop
Following Networks
00000040
<All will be dropped>

•) Pass ( ) Drop
Following Remote Servers
BIG SERVER
<All will be dropped>

(•) Pass ( ) Drop
Following Servers
FRED
<All will be dropped>

(•) Pass ( ) Drop
Following Remote Servers
<All will be dropped>

(•) Pass ( ) Drop
Following Local Services
Print Queue 03

[X] Enable Outgoing SLIST Commands
[ ] Enable Incoming SLIST Commands

When Novell systems are networked together, one of the undesirable side effects is that all Novell servers can be seen by all other Novell servers and clients that are connected to the network. These menus will configure the KarlBridge/KarlBrouter to selectively restrict access to specific Novell networks, servers and/or services. The KarlBridge/KarlBrouter is not a Novell router. It does not have any of the characteristics of a Novell router. The KarlBridge/KarlBrouter is simply a bridge that for Novell IPX can promote or prohibit specific services.

Following Networks

This menu specifies the Novell networks that will be passed (permitted) or dropped (denied) through the KarlBridge/KarlBrouter. You can use it to firewall off specific Novell networks from other Novell networks.

Following Remote Servers

This menu specifies the Remote Novell servers that are to be accessible by the Local LAN's.

Following Local Servers

This menu specifies the Local Novell servers that are to be accessible by the Remote LAN's.

Following Remote Services

This menu specifies the Remote Novell services that are to be accessible by the Local LAN's.

Following Local Services

This menu specifies the Local Novell services that are to be accessible by the Remote LAN's.

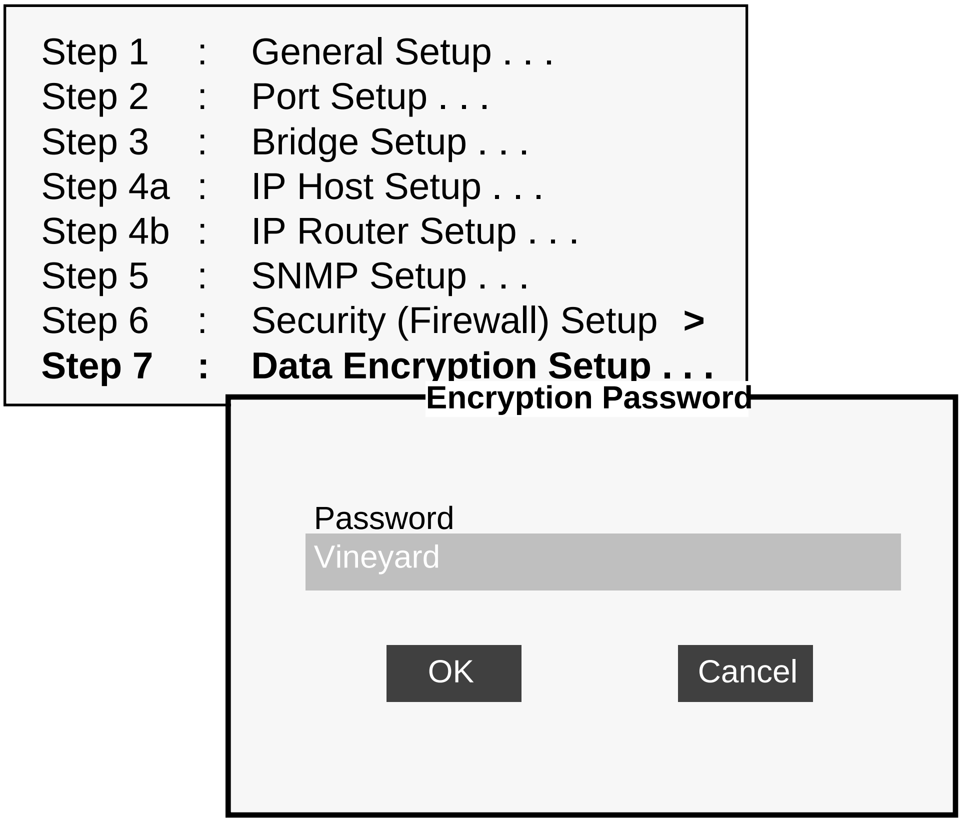
[X] Enable Outgoing SLIST Commands

The Novell SLIST and related commands bypass the normal Novell Remote Server KarlBridge/KarlBrouter filters. This is a special filter that enables or disables the Novell server listing commands from local clients to remote servers.

[X] Enable Incoming SLIST Commands

The Novell SLIST and related commands bypass the normal Novell Remote Server KarlBridge/KarlBrouter filters. This is a special filter that enables or disables the Novell server listing commands from remote clients to local servers.

STEP 7: DATA ENCRYPTION SETUP

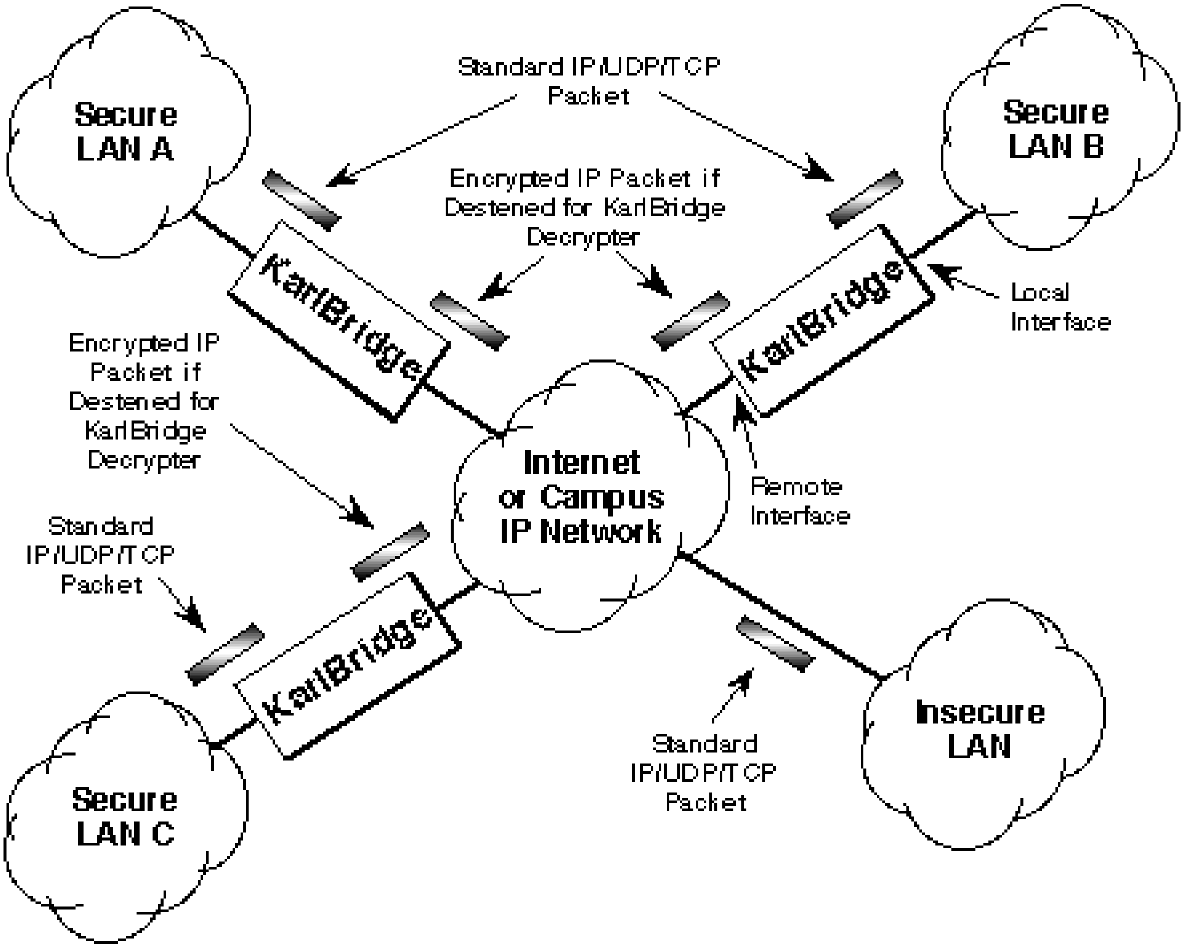

Step 1 : General Setup . . .
Step 2 : Port Setup . . .
Step 3 : Bridge Setup . . .
Step 4a : IP Host Setup . . .
Step 4b : IP Router Setup . . .
Step 5 : SNMP Setup . . .
Step 6 : Security (Firewall) Setup >
Step 7 : Data Encryption Setup . . .
Encryption Password
OK
Cancel

Password
Vineyard

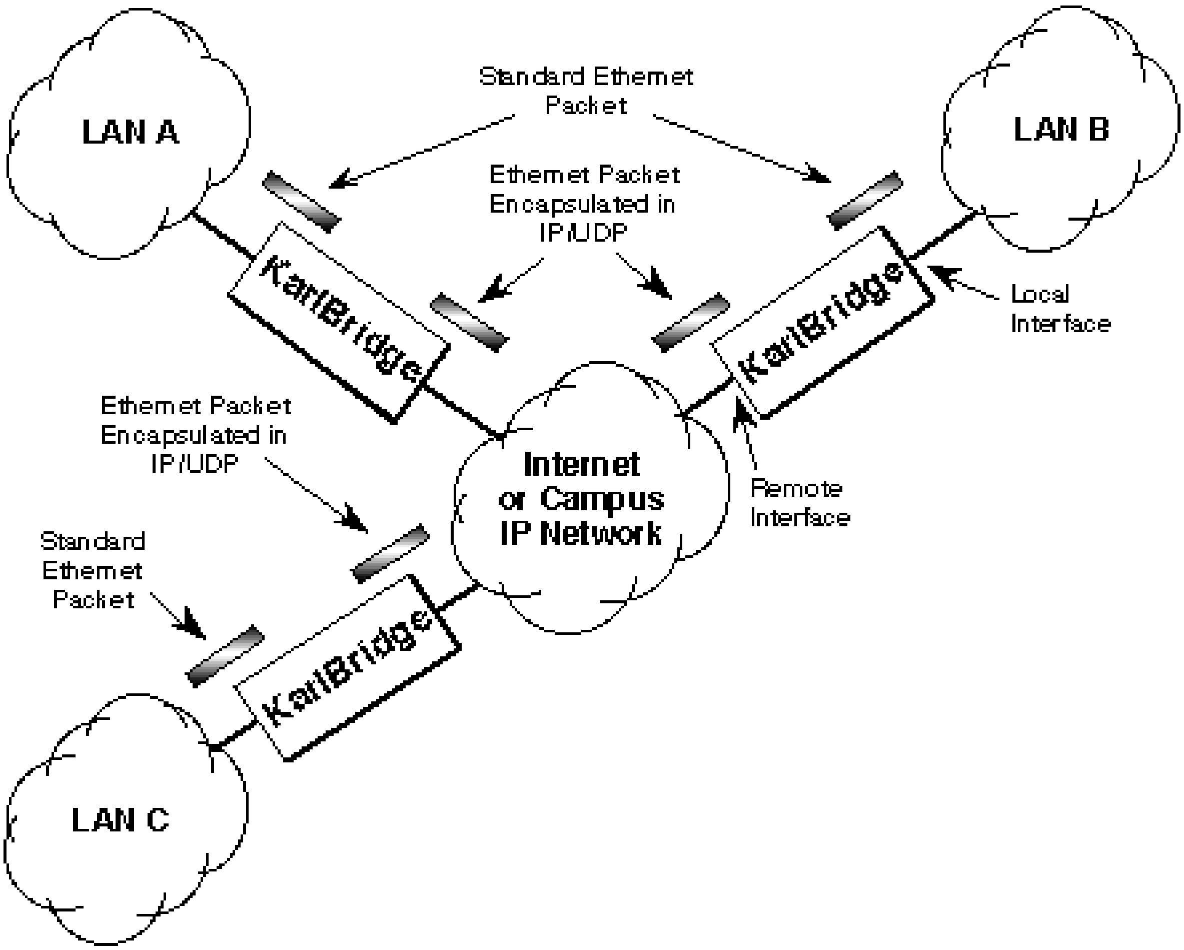
Data Encryption

The KarlBridge/KarlBrouter contains a proprietary software encryption algorithm developed in the United Kingdom. This encryption algorithm can be applied to KarlBridge Tunneled packets, IP UDP/TCP packets or all packets sent to or received from a particular non-Ethernet port.

Adding Data Encryption
(To IP/UDP/TCP Packets)

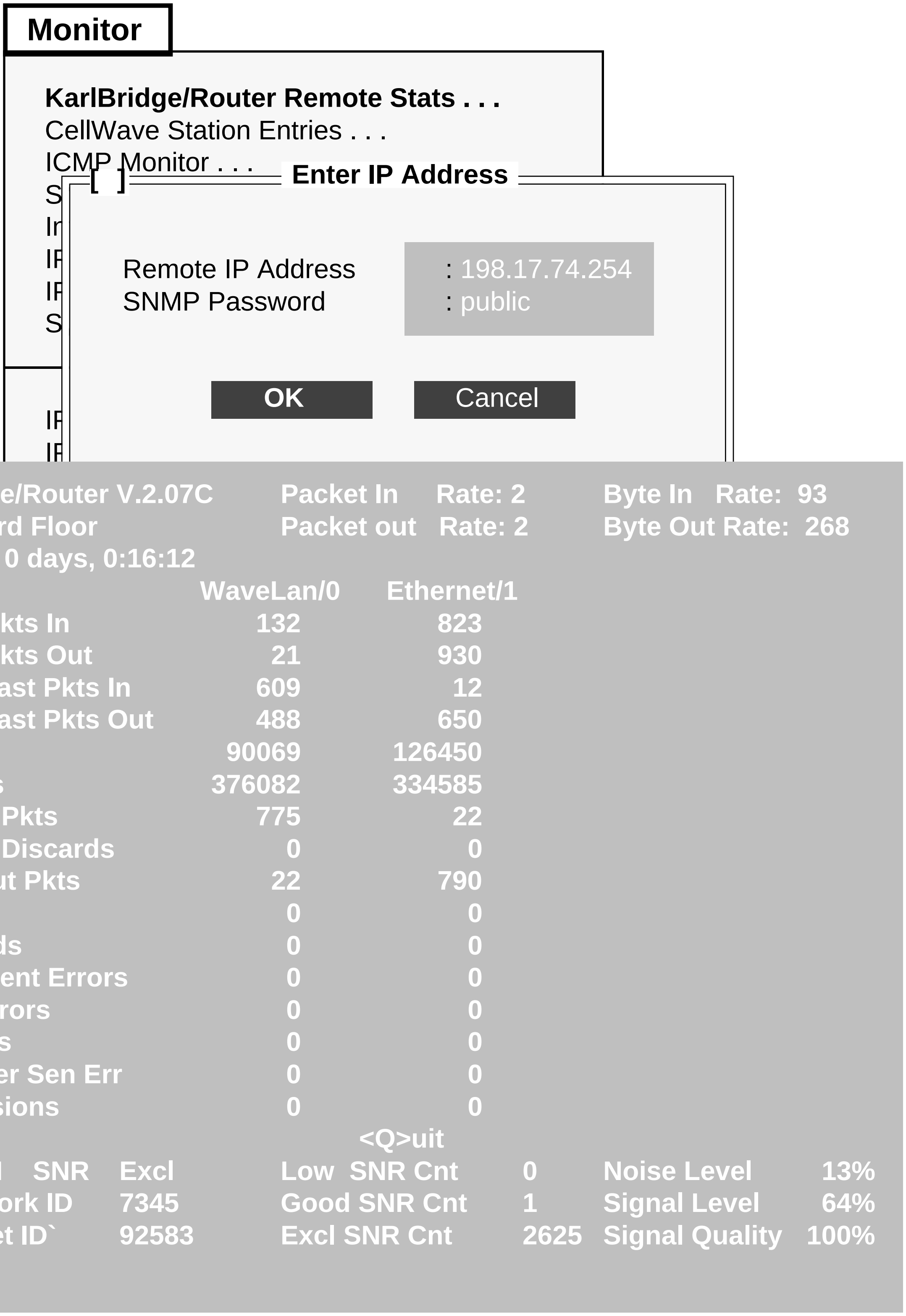

Secure LAN A
Standard IP/UDP/TCP Packet
Secure LAN B
Encrypted IP Packet if Destened for KarlBridge Decrypter
KarlBridge
KarlBridge
Local Interface
Encrypted IP Packet if Destened for KarlBridge Decrypter
Internet or Campus IP Network
Remote Interface
Standard IP/UDP/TCP Packet
KarlBridge
Insecure LAN
Standard IP/UDP/TCP Packet
Secure LAN C

Generic Ethernet Tunneling
(Through an IP Network)



SNMP MONITOR

KarlBridge/Router Remote STATS


Monitor
KarlBridge/Router Remote Stats . . .
CellWave Station Entries . . .
ICMP Monitor . . .
[ ]
Enter IP Address
OK
Cancel

Remote IP Address : 198.17.74.254
SNMP Password : public

```
KarlBridge/Router V.2.07C       Packet In     Rate: 2       Byte In   Rate:  93
Doug's 3'rd Floor               Packet out    Rate: 2       Byte Out Rate:  268
upTime:   0 days, 0:16:12
                           WaveLan/0      Ethernet/1
Unicast Pkts In                  132             823
Unicast Pkts Out                  21             930
Non-Unicast Pkts In              609              12
Non-Unicast Pkts Out             488             650
In Bytes                       90069          126450
Out Bytes                     376082          334585
Bridge In Pkts                   775              22
Bridge In Discards                 0               0
Bridge Out Pkts                   22             790
In Errors                          0               0
In Discards                        0               0
In Alignment Errors                0               0
In FCS Errors                      0               0
Out Errors                         0               0
Out Carrier Sen Err                0               0
Out Collisions                     0               0
                                      <Q>uit
WaveLAN    SNR    Excl          Low  SNR Cnt      0      Noise Level       13%
Our Network ID    7345          Good SNR Cnt      1      Signal Level      64%
Wrong Net ID`     92583         Excl SNR Cnt      2625   Signal Quality   100%
```

The remote console monitor displays a selected set of SNMP variables that provide a simple overview of the operation of the KarlBridge or KarlBrouter.

Packet In Rate: is the total number of packets received on all interfaces in packets per second.

Packet Out Rate: The total number of packets transmitted on all interfaces in packets per second.
Byte In Rate: The total number of bytes received on all interfaces in bytes per second.

Byte Out Rate: The total number of bytes transmitted on all interfaces in bytes per second.

Unicast Packets In: The number of subnetwork-unicast packets delivered to a higher-layer protocol.

Unicast Packets Out: The total number of octets (bytes) transmitted out of the interface, including framing characters.

Non-Unicast Packets In: The number of non-unicast (i.e., subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.

Non-Unicast Packets Out: The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.

In Bytes: Total number of octets (bytes) received on the interface, including framing characters.

Out Bytes: The total number of packets that higher-level protocols requested be transmitted to a non-unicast (i.e., a subnetwork-broadcast or subnetwork-multicast) address, including those that were discarded or not sent.

Bridge In Packets: The number of frames that have been received by this port from its segment. Note that a frame received on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function.

Bridge In Discards: The count of valid frames received which were discarded (i.e., filtered) by the Forwarding Process.

Bridge Out Packets: The number of frames that have been transmitted by this port to its segment. Note that a frame transmitted on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function.

In Errors: The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.

In Discards: The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.

In Alignment Errors: A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the FCS check.

In FCS Errors: A count of frames received on a particular interface that are an integral number of octets in length but do not pass the FCS check.

Out Errors: Number of outbound packets that couldn't be transmitted because of errors.

Out Carrier Sense Errors: The number of times that the carrier sense condition was lost or never asserted when attempting to transmit a frame on a particular interface.

Out Collisions: A count of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one or more collisions plus the number of times that a collision is detected on a particular interface later than 512 bit-times into the transmission of a packet.

WaveLAN SNR: The current signal to noise ratio of the WaveLAN interface. Excellent, Good, Low, or Unknown.

Our Network ID: The WaveLAN network ID (NWID) of the WaveLAN interface in the monitored device.

Wrong Network ID: The number of packets received on the WaveLAN interface from the wrong network ID.

WaveLAN Low SNR: The number of times that the WaveLAN boards receiver was found to be receiving packets that had a Low signal to noise ratio.

WaveLAN Good SNR: The number of times that the WaveLAN boards receiver was found to be receiving packets that had a Good signal to noise ratio.

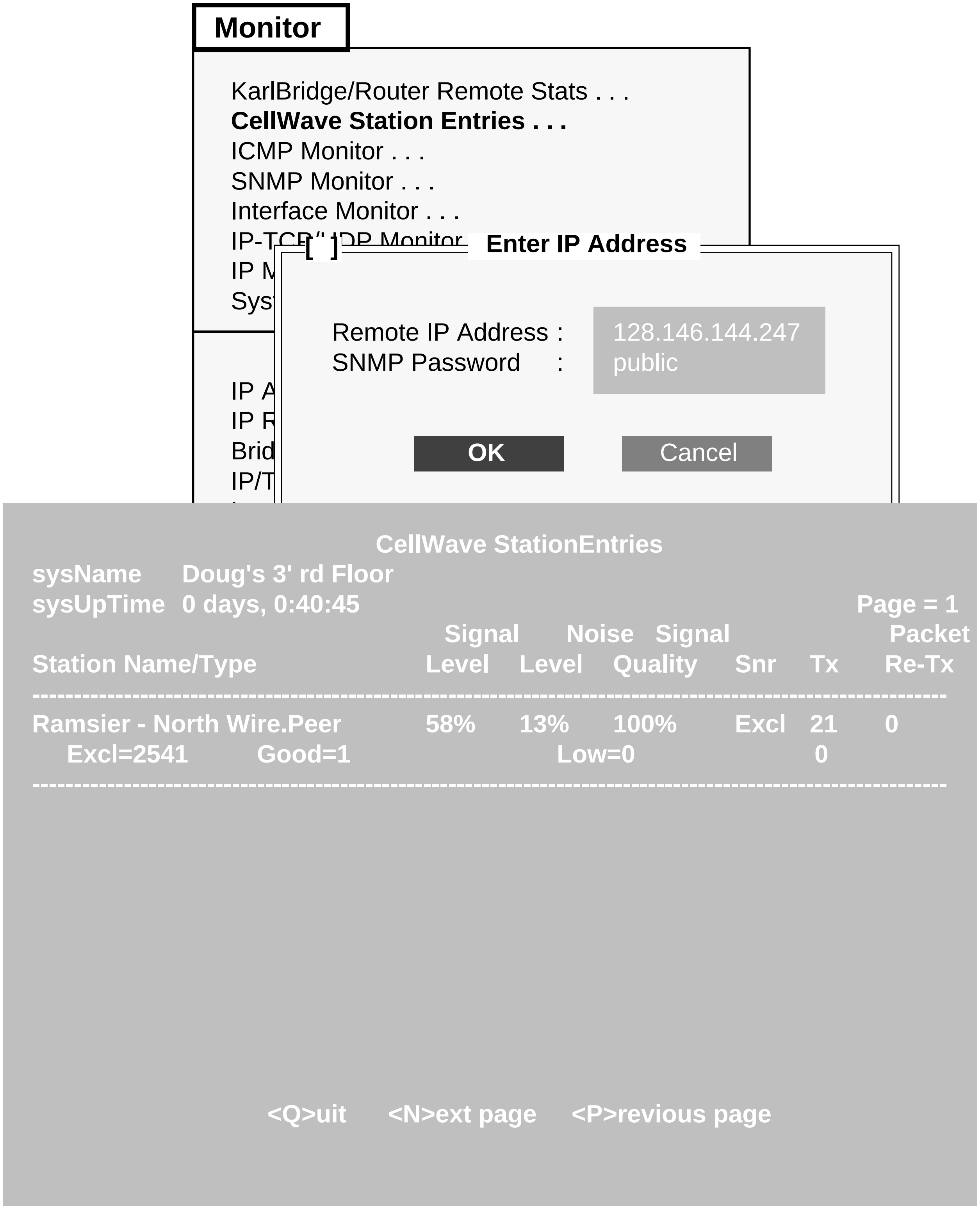
WaveLAN Excl SNR: The number of times that the WaveLAN boards receiver was found to be receiving packets that had an Excellent signal to noise ratio.

WaveLAN Noise Level: The current background RF noise level on the WaveLAN receiver in percent.

WaveLAN Signal Level: The current RF signal level on the WaveLAN receiver in percent.

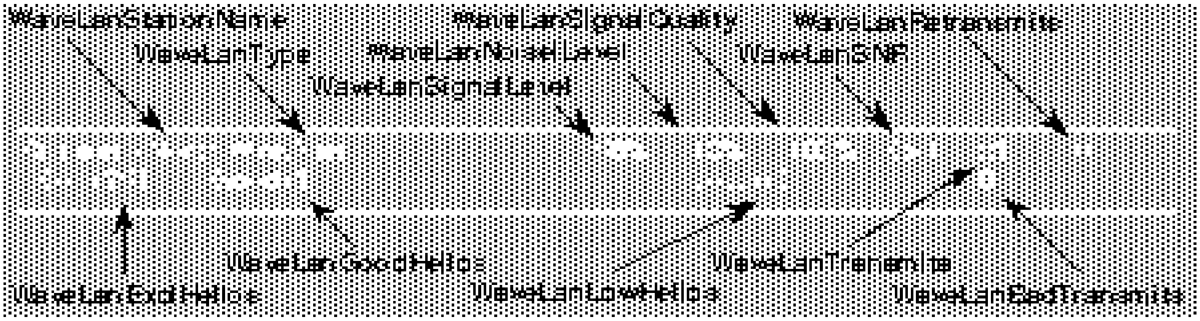
WaveLAN Signal Quality: The current RF quality on the WaveLAN receiver in percent.

CellWave Station Entries



Remote IP Address : 128.146.144.247
SNMP Password : public

```
                              CellWave StationEntries
sysName     Doug's 3' rd Floor
sysUpTime   0 days, 0:40:45                                          Page = 1
                                   Signal     Noise   Signal                Packet
Station Name/Type                  Level     Level    Quality    Snr    Tx     Re-Tx
--------------------------------------------------------------------------------------
Ramsier - North Wire.Peer          58%       13%      100%       Excl   21     0
   Excl=2541       Good=1                   Low=0                        0
--------------------------------------------------------------------------------------









                   <Q>uit     <N>ext page     <P>revious page
```



WaveLanStationName: The ASCII name of the remote station that sent this Hello packet.

WaveLanExclHellos: A count of the number of Hello packets that were received from this remote station with an Excellent signal to noise ratio.

WaveLanGoodHellos: A count of the number of Hello packets that were received from this remote station with a Good signal to noise ratio.

WaveLanLowHellos: A count of the number of Hello packets that were received from this remote station with a Low signal to noise ratio.

WaveLanSignalLevel: The signal level of the most recently received Hello packet.

WaveLanNoiseLevel: The noise level that was mesured after the most recently received Hello packet was received.

WaveLanSignalQuality: The signal quality of the most recently received Hello packet.

WaveLanTransmits: The number of CellWave packets offered for transmit.

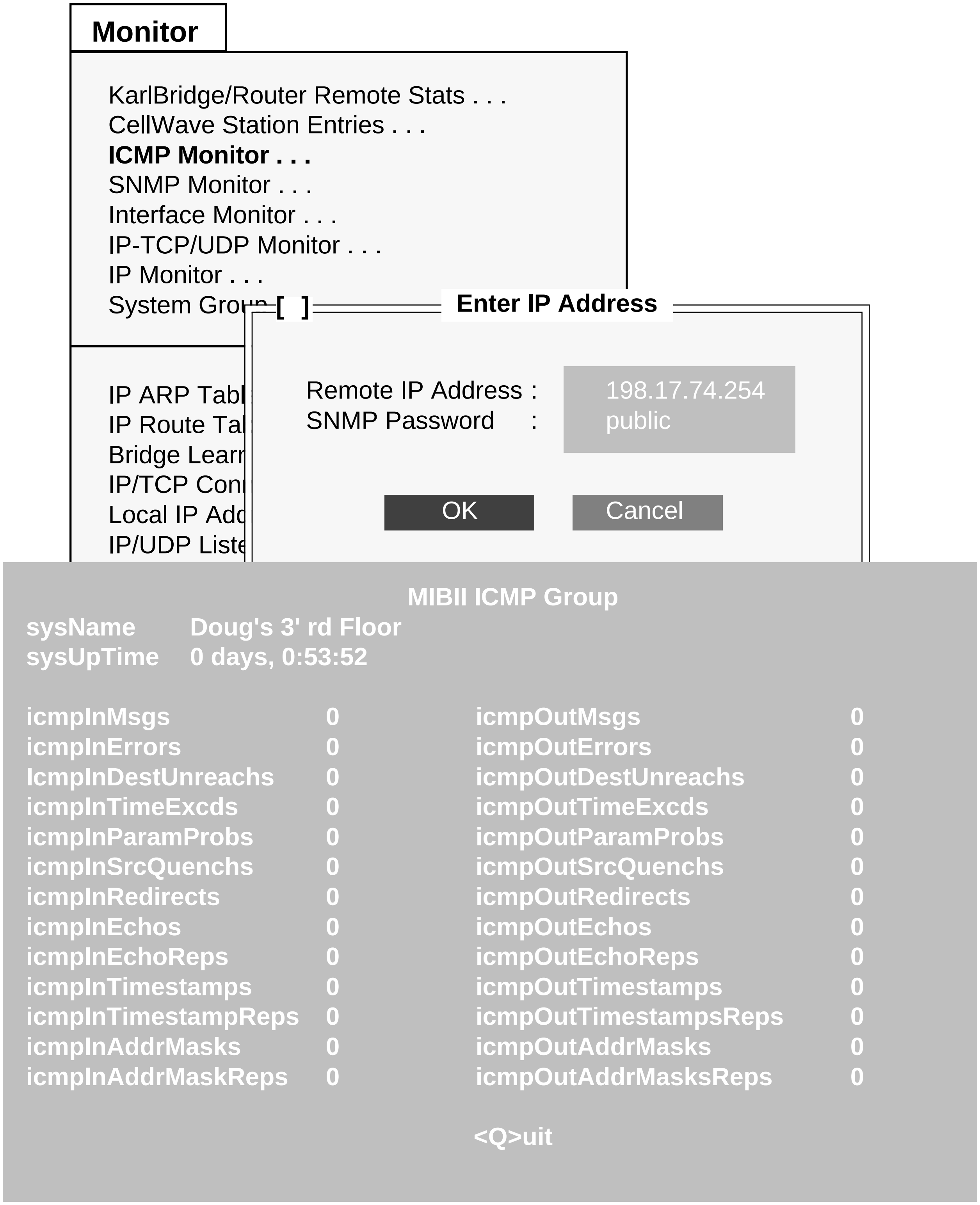
WaveLanRetransmits: The number of times a CellWave packet had to be retransmitted.

WaveLanBadTransmits: The number of times the CellWave transmit module gave up retransmitting a particular packet. It gives up after 10 attempts.

WaveLanType: The type of remote station heard from. The type are: Comp; a station in compatibility mode, Cell; the remote station is in CellWave - No Base Station mode, Base; the remote station is a CellWave Base station, Satl; the remote station is a CellWave Satellite station.

WaveLanSNR: A string representing the signal to noise ratio of the last received Hello packet. It will have a value of Unkn; Unknown SNR, Low; Low SNR, Good; Good SNR, Excl; Excelent SNR.

ICMP MONITOR



Remote IP Address : 198.17.74.254
SNMP Password : public

```
                          MIBII ICMP Group
sysName        Doug's 3' rd Floor
sysUpTime      0 days, 0:53:52

icmpInMsgs                0        icmpOutMsgs                 0
icmpInErrors              0        icmpOutErrors               0
IcmpInDestUnreachs        0        icmpOutDestUnreachs         0
icmpInTimeExcds           0        icmpOutTimeExcds            0
icmpInParamProbs          0        icmpOutParamProbs           0
icmpInSrcQuenchs          0        icmpOutSrcQuenchs           0
icmpInRedirects           0        icmpOutRedirects            0
icmpInEchos               0        icmpOutEchos                0
icmpInEchoReps            0        icmpOutEchoReps             0
icmpInTimestamps          0        icmpOutTimestamps           0
icmpInTimestampReps       0        icmpOutTimestampsReps       0
icmpInAddrMasks           0        icmpOutAddrMasks            0
icmpInAddrMaskReps        0        icmpOutAddrMasksReps        0

                               <Q>uit
```

The KarlBridge/KarlBrouter keeps the standard SNMP MIB II statistics on IP/ICMP type protocols as follows:

icmpInMsgs: The total number of ICMP messages which the entity received. Note that this counter includes all those counted by icmpInErrors.

icmpInErrors: The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.).

icmpInDestUnreachs: The # of ICMP Destination Unreachable messages received.

icmpInTimeExcds: The number of ICMP Time Exceeded messages received.

icmpInParamProbs: The number of ICMP Parameter Problem messages received.

icmpInSrcQuenchs: The number of ICMP Source Quench messages received.

icmpInRedirects: The number of ICMP Redirect messages received.

icmpInEchos: The number of ICMP Echo (request) messages received.

icmpInEchoReps: The number of ICMP Echo Reply messages received.

icmpInTimestamps: The number of ICMP Timestamp (request) messages received.

icmpInTimestampReps: The number of ICMP Timestamp Reply messages received.

icmpInAddrMasks: The number of ICMP Address Mask Reply messages received.

icmpInAddrMaskReps: The number of ICMP Address Mask Reply messages received.

icmpOutMsgs: The total number of ICMP messages which this entity attempted to send. Note that this counter includes all those counted by icmpOutErrors.

icmpOutErrors: The number of ICMP messages which this entity did not send due to problems discovered within ICMP such as a lack of buffers. This value doesn't include errors discovered outside the ICMP layer such as the inability of IP to route the resultant datagram. In some implementations there may be no types of error which contribute to this counter's value.

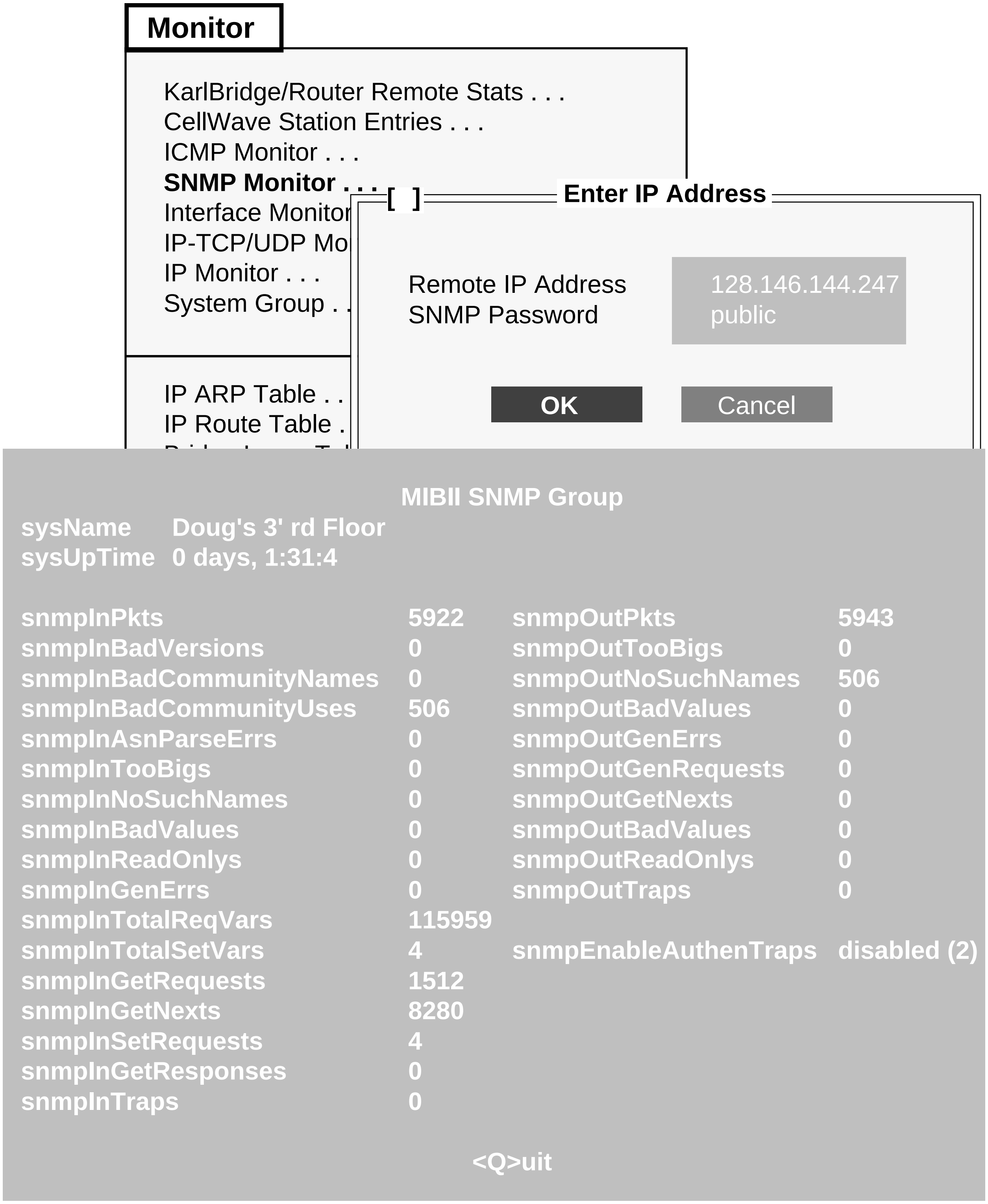
icmpOutDestUnreachs: The number of ICMP Destination Unreachable messages sent.

icmpOutTimeExcds: The number of ICMP Time Exceeded messages.

icmpOutPramProbs: The number of ICMP Parameter Problem messages.

icmpOutSrcQuenchs: The number of ICMP Source Quench messages sent.

SNMP STATISTICS



Remote IP Address 128.146.144.247
SNMP Password public

```
                              MIBII SNMP Group
sysName     Doug's 3' rd Floor
sysUpTime   0 days, 1:31:4

snmpInPkts                   5922      snmpOutPkts             5943
snmpInBadVersions            0         snmpOutTooBigs          0
snmpInBadCommunityNames      0         snmpOutNoSuchNames      506
snmpInBadCommunityUses       506       snmpOutBadValues        0
snmpInAsnParseErrs           0         snmpOutGenErrs          0
snmpInTooBigs                0         snmpOutGenRequests      0
snmpInNoSuchNames            0         snmpOutGetNexts         0
snmpInBadValues              0         snmpOutBadValues        0
snmpInReadOnlys              0         snmpOutReadOnlys        0
snmpInGenErrs                0         snmpOutTraps            0
snmpInTotalReqVars           115959
snmpInTotalSetVars           4         snmpEnableAuthenTraps   disabled (2)
snmpInGetRequests            1512
snmpInGetNexts               8280
snmpInSetRequests            4
snmpInGetResponses           0
snmpInTraps                  0

                                  <Q>uit
```

These statistics are gathered on the SNMP agent that resides in the target machine. Note that the objects defined below will be zero-valued in those SNMP implementations that are optimized to support only those functions specific to either a management agent or a management station.

snmpInPkts: The total number of Messages delivered to the SNMP entity from the transport service.

snmpInBadVersions: The total number of SNMP Messages which were delivered to the SNMP protocol entity and were for an unsupported SNMP version.

snmpInBadCommunityNames: The total number of SNMP Messages delivered to the SNMP protocol entity which used a SNMP community name not known to said entity.

snmpInBadCommunityUses: The total number of SNMP Messages delivered to the SNMP protocol entity which represented an SNMP operation which was not allowed by the SNMP community named in the Message.

snmpInAsnParseErrs: The total number of ASN.1 or BER errors encountered by the SNMP protocol entity when decoding received SNMP Messages.

snmpInTooBigs: The total number of SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is `tooBig'.

snmpInNoSuchNames: The total number of SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is `noSuchName'.

snmpInBadValues: The total number of SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is `badValue'.

snmpInReadOnlys: The total number valid SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is `readOnly'. It should be noted that it is a protocol error to generate an SNMP PDU which contains the value `readOnly' in the error-status field, as such this object is provided as a means of detecting incorrect implementations of the SNMP.

snmpInGenErrs: The total number of SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is `genErr'.

snmpInTotalReqVars: The total number of MIB objects which have been retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Get-Request and Get-Next PDUs.

snmpInTotalSetVars: The total number of MIB objects which have been altered successfully by the SNMP protocol entity as the result of receiving valid SNMP Set-Request PDUs.

snmpInGetRequests: The total number of SNMP Get-Request PDUs which have been accepted and processed by the SNMP protocol entity.

snmpInGetNexts: The total number of SNMP Get-Next PDUs which have been accepted and processed by the SNMP protocol entity.

snmpInSetRequests: The total number of SNMP Set-Request PDUs which have been accepted and processed by the SNMP protocol entity.

snmpInGetResponses: The total number of SNMP Get-Response PDUs which have been accepted and processed by the SNMP protocol entity.

snmpInTraps: The total number of SNMP Trap PDUs which have been accepted and processed by the SNMP protocol entity.

snmpOutPkts: The total number of SNMP messages which were passed from the SNMP protocol entity to the transport servic e.

snmpOutTooBigs: The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status field is `tooBig.'

snmpOutNoSuchNames: The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status is `noSuchName'.

snmpOutBadValues: The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status field is `badValue'.

snmpOutGenErrs: The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status field is `genErr'.

snmpOutGetRequests: The total number of SNMP Get-Request PDUs which have been generated by the SNMP protocol entity.

snmpOutGetNexts: The total number of SNMP Get-Next PDUs which have been generated by the SNMP protocol entity.

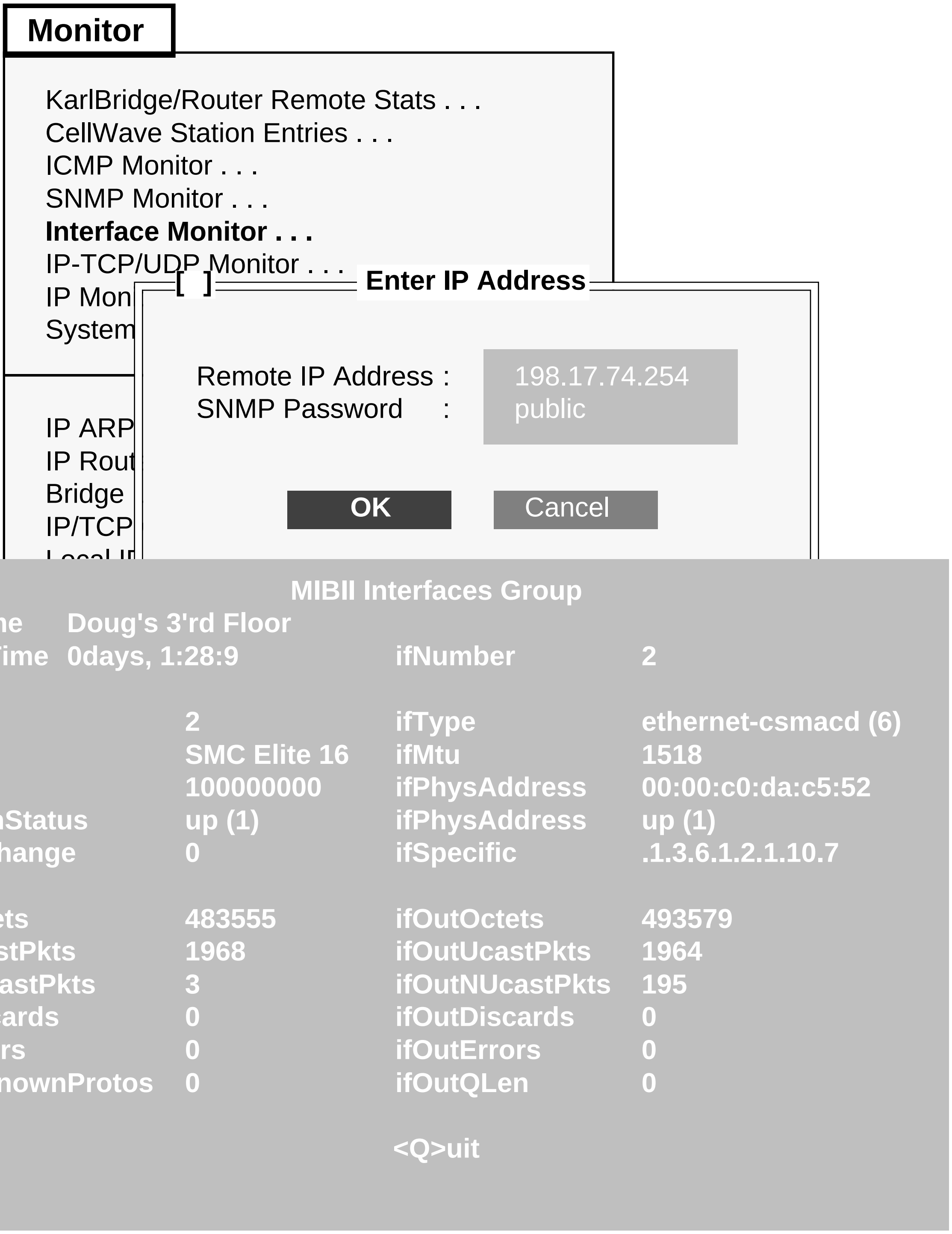
snmpOutSetRequests: The total number of SNMP Set-Request PDUs which have been generated by the SNMP protocol entity.

snmpOutGetResponses: The total number of SNMP Get-Response PDUs which have been generated by the SNMP protocol entity.

snmpOutTraps: The total number of SNMP Trap PDUs which have been generated by the SNMP protocol entity.

snmpEnableAuthenTraps: Indicates whether the SNMP agent process is permitted to generate authentication-failure traps.

INTERFACE MONITOR



Remote IP Address : 198.17.74.254
SNMP Password : public

```
                         MIBII Interfaces Group
sysName     Doug's 3'rd Floor
sysUpTime   0days, 1:28:9                ifNumber         2

ifIndex             2                    ifType           ethernet-csmacd (6)
ifDescr             SMC Elite 16         ifMtu            1518
ifSpeed             100000000            ifPhysAddress    00:00:c0:da:c5:52
ifAdminStatus       up (1)               ifPhysAddress    up (1)
ifLastChange        0                    ifSpecific       .1.3.6.1.2.1.10.7

ifInOctets          483555               ifOutOctets      493579
ifInUcastPkts       1968                 ifOutUcastPkts   1964
ifInNUcastPkts      3                    ifOutNUcastPkts  195
ifInDiscards        0                    ifOutDiscards    0
ifInErrors          0                    ifOutErrors      0
ifInUnknownProtos   0                    ifOutQLen        0

                                <Q>uit
```

The Interfaces table contains information on the entity's interfaces. Each interface is thought of as being attached to a `subnetwork'. Note that this term should not be confused with `subnet' which refers to an addressing partitioning scheme used in the Internet suite of protocols.

ifNumber: The number of network interfaces (regardless of their current state) present on this system.

ifIndex: A unique value for each interface. Its value ranges between 1 and the value of ifNumber. The value for each interface must remain constant at least from one re-initialization of the entity's network management system to the next reinitialization.

ifDescr: A textual string containing information about the interface. This string should include the name of the manufacturer, the product name and the version of the hardware interface.

ifSpeed: An estimate of the interface's current bandwidth in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth.

ifAdminStatus: The desired state of the interface. The testing(3) state indicates that no operational packets can be passed. up(1), ready to pass packets; down(2); testing(3) -- in some test mode.

ifLastChange: The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last reinitialization of the local network management subsystem, then this object contains a zero value.

ifInOctets: The total number of octets (bytes) received on the interface, including framing characters.

ifInUcastPkts: The number of subnetwork-unicast packets delivered to a higher-layer protocol.

ifInNUcastPkts: The number of non-unicast (i.e., subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.

ifInDiscards: The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.

ifInErrors: The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.

ifInUnknownProtos: The number of packets received via the interface which were discarded because of an unknown or unsupported protocol.

ifType: The type of interface, distinguished according to the physical/link protocol(s) immediately `below' the network layer in the protocol stack. The following possibilities are: other(1), regular1822(2), hdh1822(3), ddn-x25(4), rfc877-x25(5), ethernet-csmacd(6), iso88023-csmacd(7), iso88024-tokenBus(8), iso88025-tokenRing(9), iso88026-man(10), starLan(11), proteon-10Mbit(12), proteon-80Mbit(13), hyperchannel(14), fddi(15), lapb(16), sdlc(17), ds1(18), e1(19), basicISDN(20), primaryISDN(21), propPointToPointSerial(22), ppp(23), softwareLoopback(24), eon(25), ethernet-3Mbit(26), nsip(27), slip(28), ultra(29), ds3(30), sip(31), frame-relay(32)

ifMtu: The size of the largest datagram which can be sent/received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, this is the size of the largest network datagram that can be sent on the interface.

ifPhysAddress: The interface's address at the protocol layer immediately `below' the network layer in the protocol stack. For interfaces which do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

ifOperStatus: The current state of the interface. The testing(3) state indicates that no operational packets can be passed. up(1), ready to pass packets; down(2); testing(3) -- in some test mode.

ifSpecific: A reference to MIB definitions specific to the particular media being used to realize the interface. For example, if the interface is realized by an ethernet, then the value of this object refers to a document defining objects specific to ethernet. If this information is not present, its value will be set to 0.

ifOutOctets: The total number of octets (bytes) transmitted out of the interface, including framing characters.

ifOutUcastPkts: The total # of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.

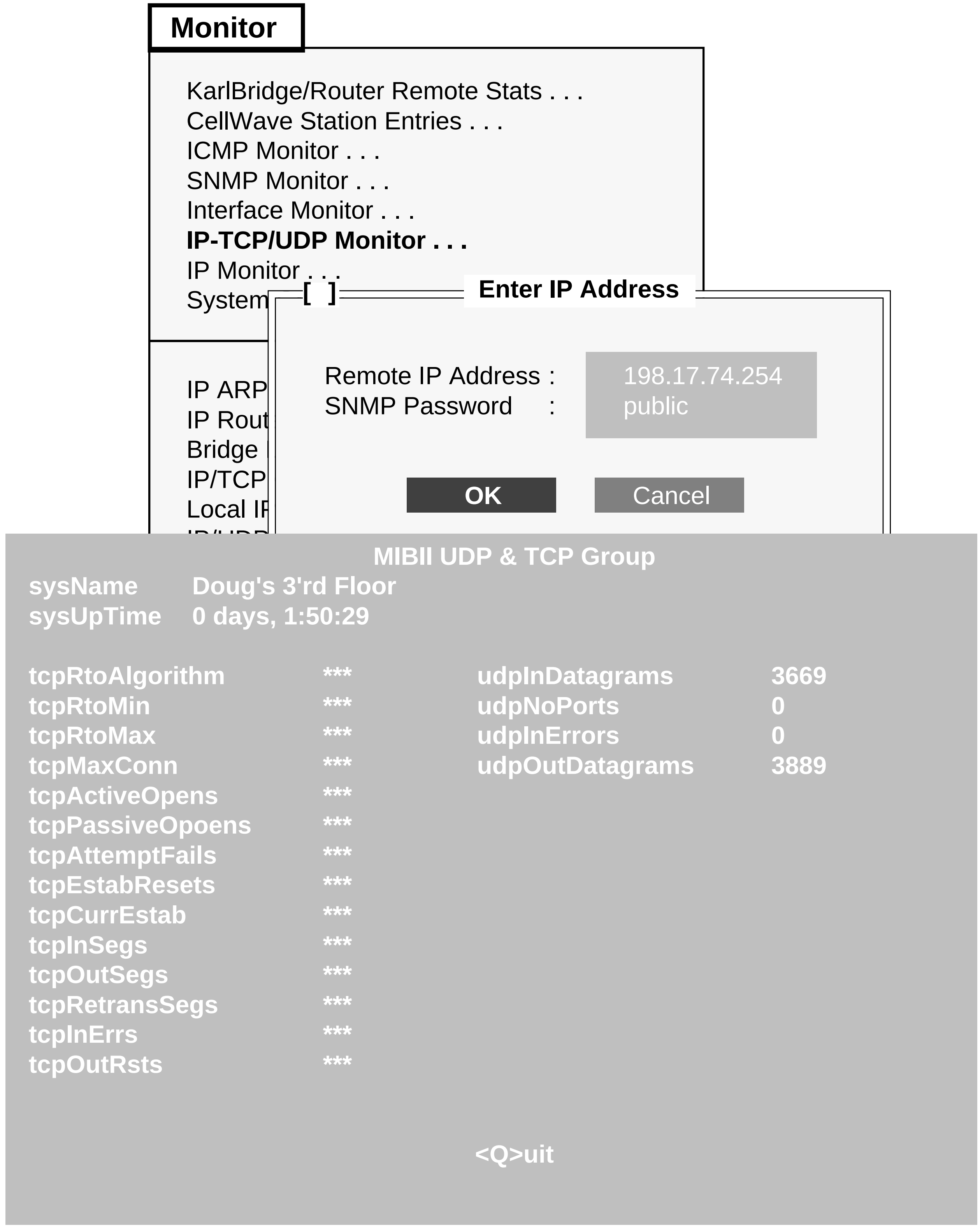
ifOutNUcastPkts: The total number of packets that higher-level protocols requested be transmitted to a non-unicast (i.e., a subnetwork-broadcast or subnetwork-multicast) address, including those that were discarded or not sent.

ifOutDiscards: The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.

ifOutErrors: The number of outbound packets that could not be transmitted because of errors.

ifOutQLen: The length of the output packet queue (in packets).

IP-TCP/UDP MONITOR



Remote IP Address : 198.17.74.254
SNMP Password : public

```
                      MIBII UDP & TCP Group
sysName      Doug's 3'rd Floor
sysUpTime    0 days, 1:50:29

tcpRtoAlgorithm       ***        udpInDatagrams     3669
tcpRtoMin             ***        udpNoPorts         0
tcpRtoMax             ***        udpInErrors        0
tcpMaxConn            ***        udpOutDatagrams    3889
tcpActiveOpens        ***
tcpPassiveOpoens      ***
tcpAttemptFails       ***
tcpEstabResets        ***
tcpCurrEstab          ***
tcpInSegs             ***
tcpOutSegs            ***
tcpRetransSegs        ***
tcpInErrs             ***
tcpOutRsts            ***

                            <Q>uit
```

The KarlBridge/KarlBrouter keeps the standard SNMP MIB II statistics on IP/TCP and IP/UDP type protocols as follows:

tcpRtoAlgorithm: The algorithm used to determine the timeout value used for retransmitting unacknowledged octets. other(1)-- none of the following; constant(2)-- a constant rto; rsre(3)-- MIL-STD-1778, Appendix B; vanj(4)-- Van Jacobson's algorithm

tcpRtoMin: The minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission timeout. In particular, when the timeout algorithm is rsre(3), an object of this type has the semantics of the LBOUND quantity described in RFC 793.

tcpRtoMax: The maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission timeout. In particular, when the timeout algorithm is rsre(3), an object of this type has the semantics of the UBOUND quantity described in RFC 793.

tcpMaxConn: The limit on the total number of TCP connections the entity can support. In entities where the maximum number of connections is dynamic, this object should contain the value -1.

tcpActiveOpens: The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.

tcpPassiveOpens: The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.

tcpAttemptFails: The number of times TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.

tcpEstabResets: The number of times TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.

tcpCurrEstab: The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.

tcpInSegs: The total number of segments received, including those received in error. This count includes segments received on currently established connections."

tcpOutSegs:The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.

tcpRetransSegs:The total number of segments retransmitted - that is, the number of TCP segments transmitted containing one or more previously transmitted octets.

tcpInErrs: The total number of segments received in error (e.g., bad TCP checksums).

tcpOutRsts: The number of TCP segments sent containing the RST flag.

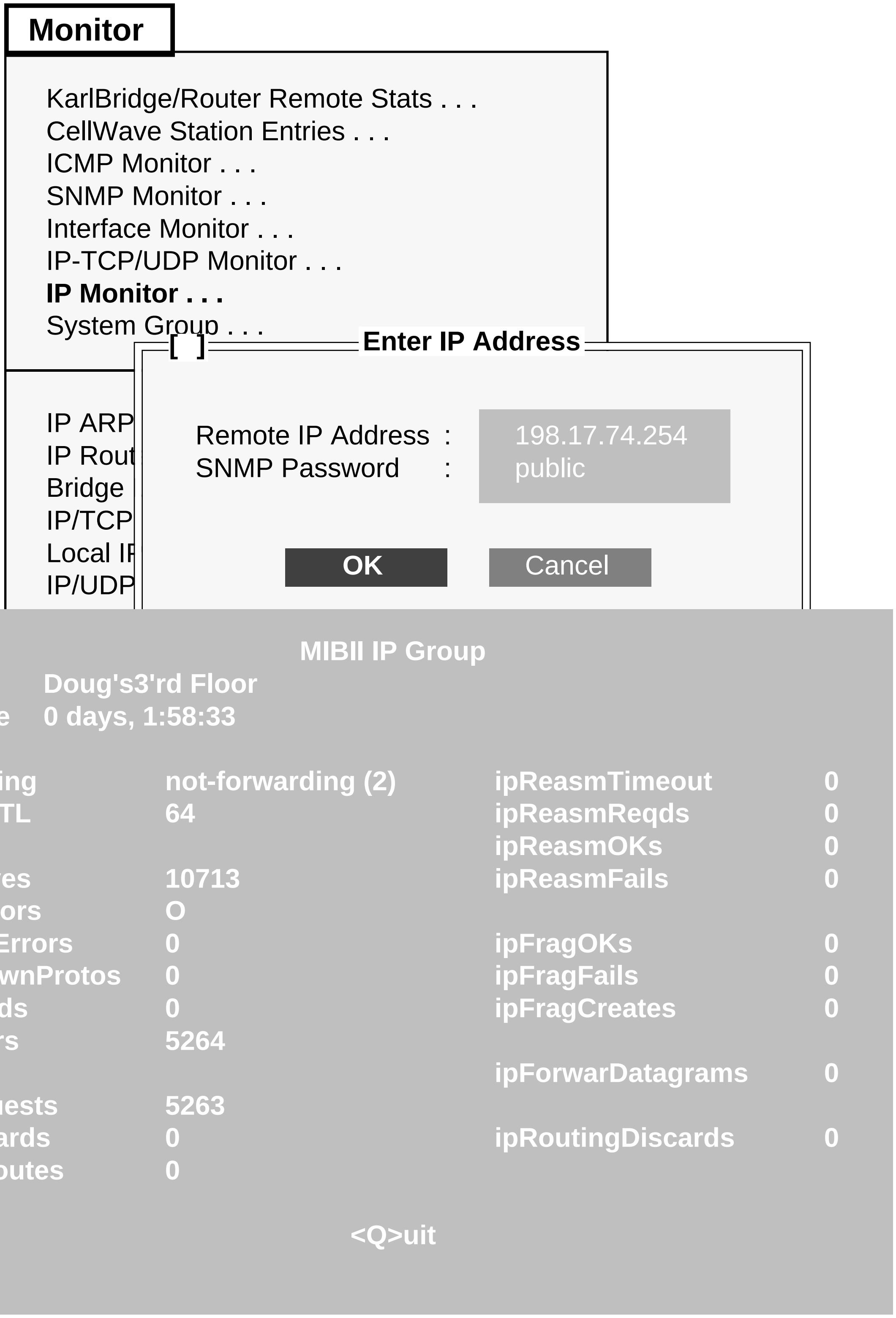
udpInDatagrams: The total number of UDP datagrams delivered to UDP users.

udpNoPorts: The total number of received UDP datagrams for which there was no application at the destination port.

udpInErrors: The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port.

udpOutDatagrams: The total number of UDP datagrams sent from this entity.

IP MONITOR



```
Remote IP Address  :   198.17.74.254
SNMP Password      :   public
```

```
                              MIBII IP Group
sysName       Doug's3'rd Floor
sysUpTime     0 days, 1:58:33

ipForwarding          not-forwarding (2)     ipReasmTimeout        0
ipDefaultTTL          64                     ipReasmReqds          0
                                             ipReasmOKs            0
ipInReceives          10713                  ipReasmFails          0
ipInHdrErrors         O
ipInAddreErrors       0                      ipFragOKs             0
ipInUnknownProtos     0                      ipFragFails           0
ipInDiscards          0                      ipFragCreates         0
ipInDelivers          5264
                                             ipForwarDatagrams     0
ipOutRequests         5263
ipOutDiscards         0                      ipRoutingDiscards     0
ipOutNoRoutes         0

                                 <Q>uit
```

The KarlBridge/KarlBrouter keeps the standard SNMP MIB II statistics on IP type protocols as follows:

ipForwarding: The indication of whether this entity is acting as an IP gateway in respect to the forwarding of datagrams received by, but not addressed to, this entity. IP gateways forward datagrams. IP hosts do not (except those source-routed via the host). Note that for some managed nodes, this object may take on only a subset of the values possible.

ipInReceives: The default value inserted into the Time-To-Live field of the IP header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol.

ipInHdrErrors: The number of input datagrams discarded due to errors in their IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, errors discovered in processing their IP options, etc.

ipInAddrErrors: The number of input datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (e.g., 0.0.0.0) and addresses of unsupported Classes (e.g., Class E). For entities which are not IP Gateways and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.

ipInUnknownProtos: The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol."

ipInDiscards: The number of input IP datagrams for which no problems were encountered to prevent their continued processing, but which were discarded (e.g., for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly.

ipInDelivers: The total number of input datagrams successfully delivered to IP user-protocols (including ICMP).

ipOutRequests: The total number of IP datagrams which local IP user-protocols (including ICMP) supplied to IP in requests for transmission. Note that this counter does not include any datagrams counted in ipForwDatagrams.

ipOutDiscards: The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space). Note that this counter would include datagrams counted in ipForwDatagrams if any such packets met this (discretionary) discard criterion.

ipOutNoRoutes: The number of IP datagrams discarded because no route could be found to transmit them to their destination. Note that this counter includes any packets counted in ipForwDatagrams which meet this `no-route' criterion. Note that this includes any datagarms which a host cannot route because all of its default gateways are down.

ipReasmTimeout: The maximum number of seconds which received fragments are held while they are awaiting reassembly at this entity.

ipReasmReqds: The number of IP fragments received which neededto be reassembled at this entity.

ipReasmOKs: The number of IP datagrams successfully reassembled.

ipReasmFails: The number of failures detected by the IP reassembly algorithm (for whatever reason: timed out, errors, etc). Note that this is not necessarily a count of discarded IP fragments since some algorithms (notably the algorithm in RFC 815) can lose track of the number of fragments by combining them as they are received.

ipFragOKs: The number of IP datagrams that have been successfully fragmented at this entity.

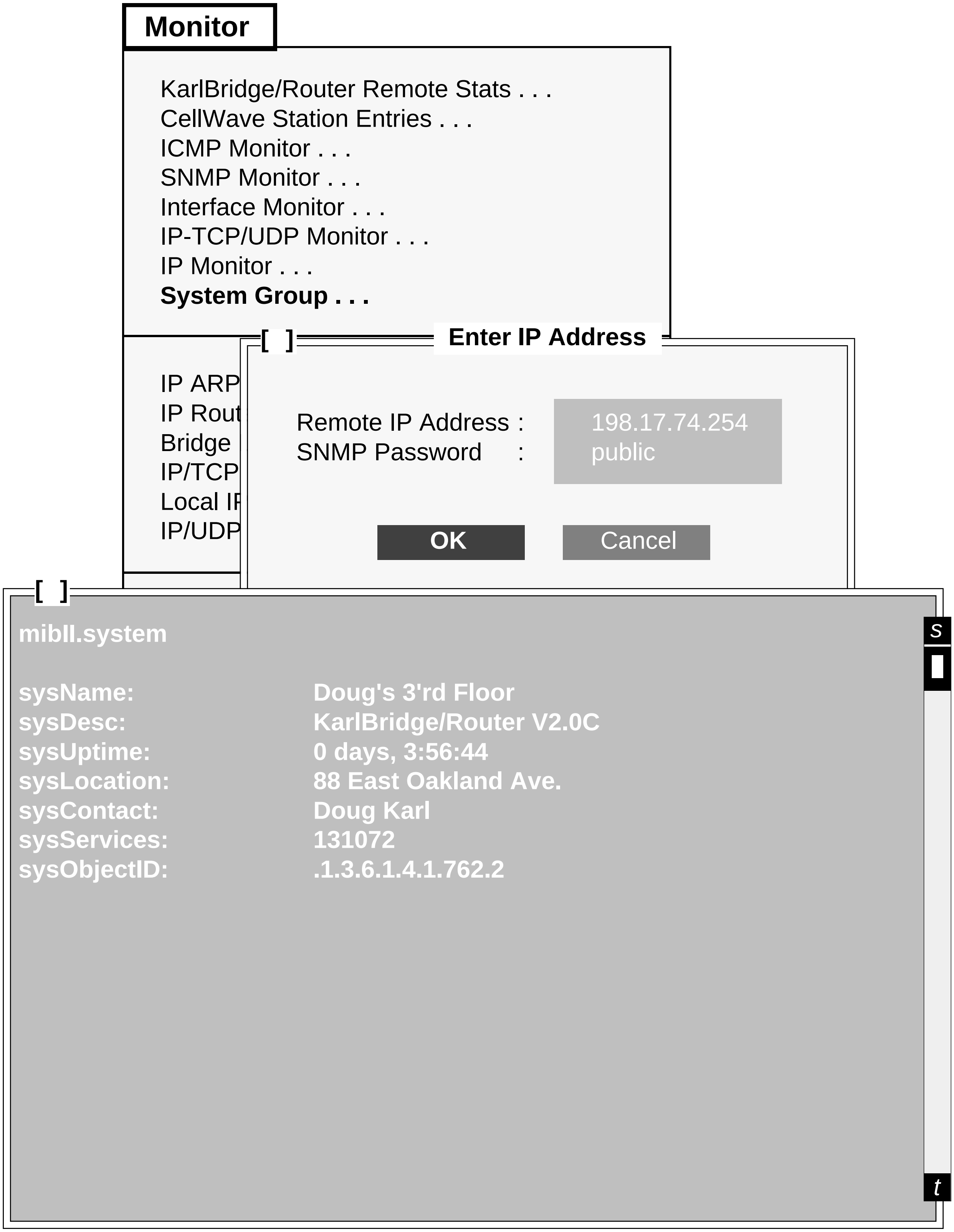
ipFragFails: The number of IP datagrams that have been discarded because they needed to be fragmented at this entity but could not be, e.g., because their Don't Fragment flag was set.

ipFragCreates: The number of IP datagram fragments that have been generated as a result of fragmentation at this entity.

ipForwDatagrams: The number of input datagrams for which this entity was not their final IP destination, as a result of which an attempt was made to find a route to forward them to that final destination. In entities which do not act as IP Gateways, this counter will include only those packets which were Source-Routed via this entity, and the Source- Route option processing was successful.

ipRoutingDiscards: The number of routing entries which were chosen to be discarded even though they are valid. One possible reason for discarding such an entry could be to free-up buffer space for other routing

SYSTEM INFORMATION



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.system

sysName:        Doug's 3'rd Floor
sysDesc:        KarlBridge/Router V2.0C
sysUptime:      0 days, 3:56:44
sysLocation:    88 East Oakland Ave.
sysContact:     Doug Karl
sysServices:    131072
sysObjectID:    .1.3.6.1.4.1.762.2
```

The KarlBridge/KarlBrouter keeps the standard SNMP MIB II statistics on system related information as follows:

Name: An administratively-assigned name for this managed node. By convention, this is the node's fully-qualified domain name.

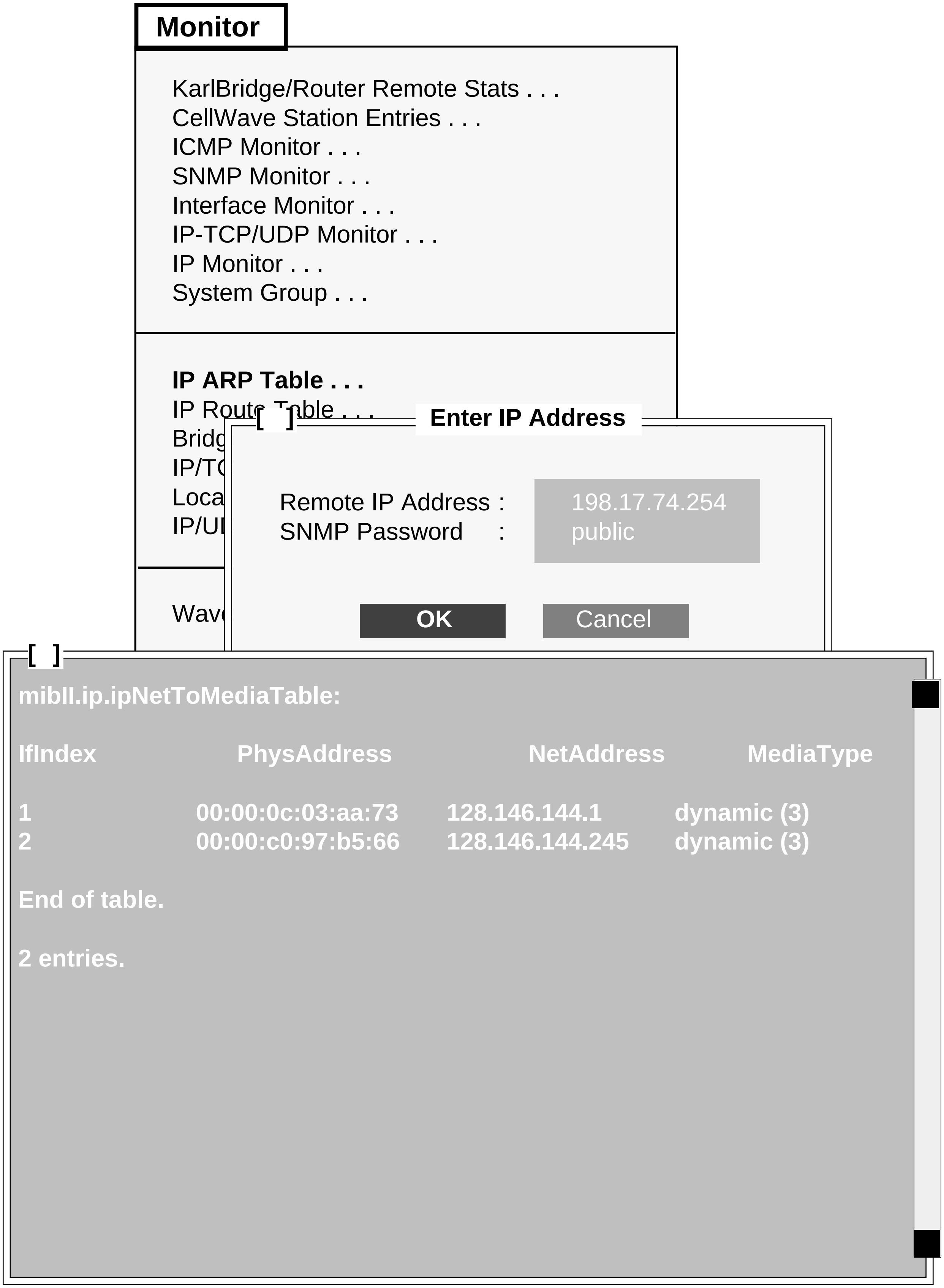
Description: This value contains the full name and version identification of the system's hardware type, software operating-system, and networking software.

Uptime: The time since the network management portion of the system was last re-initialized.

Location: The physical location of this node (e.g.,`telephone closet, 3rd floor').

Contact: The textual identification of the contact person for this managed node, together with information on how to contact this person.

IP ARP TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.ip.ipNetToMediaTable:

IfIndex          PhysAddress           NetAddress        MediaType

1            00:00:0c:03:aa:73     128.146.144.1       dynamic (3)
2            00:00:c0:97:b5:66     128.146.144.245     dynamic (3)

End of table.

2 entries.
```

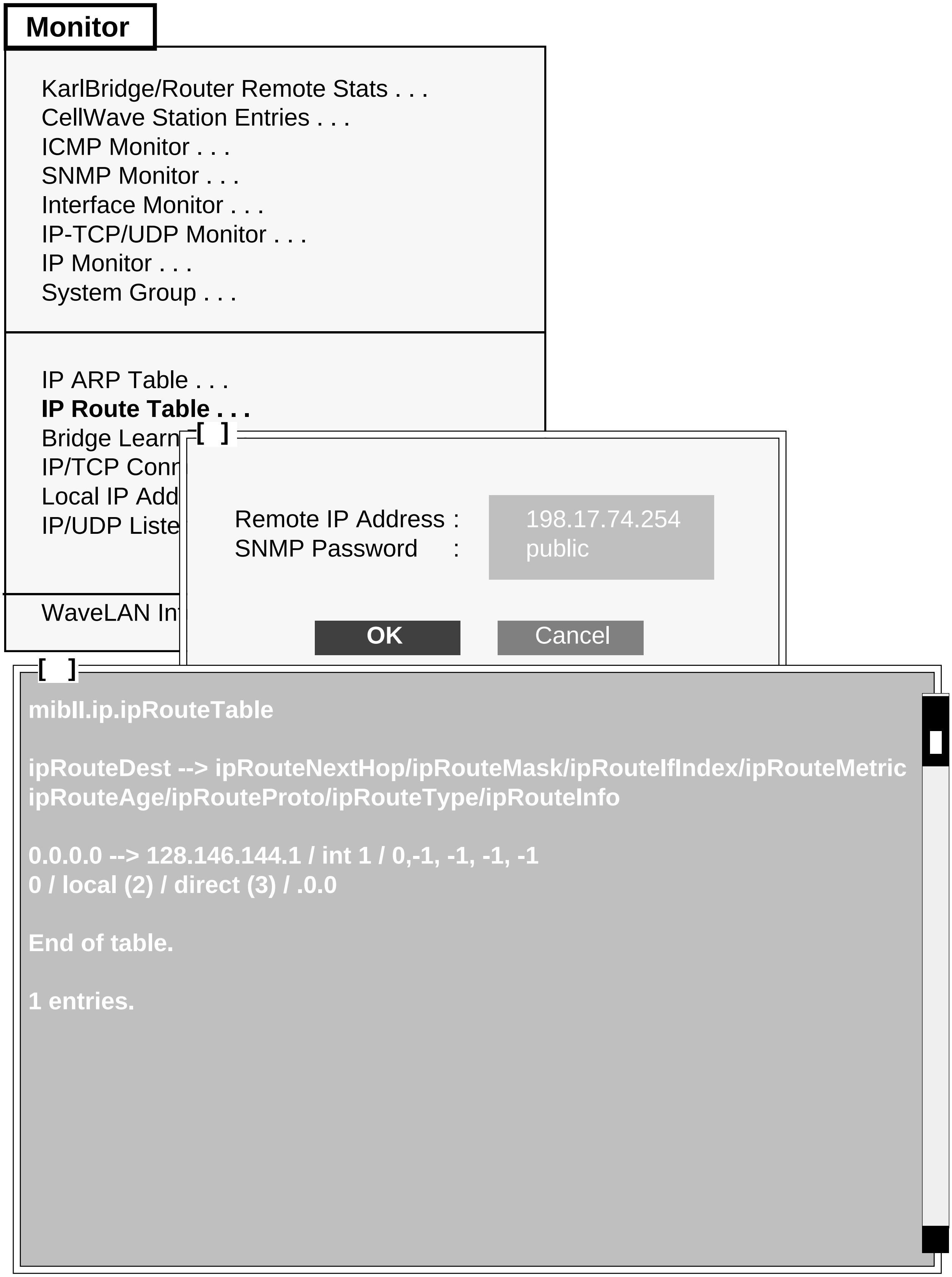
The IP address translation table contain the IpAddress to `physical' address equivalences.

IfIndex: "The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex."

PhysAddress: The media-dependent `physical' address. An example would be the address of the Ethernet or WaveLAN board.
NetAddress: The IpAddress corresponding to the media-dependent `physical' address."

MediaType: "The type of mapping. other(1) -- none of the following; invalid(2) -- an invalidated mapping; dynamic(3); static(4)

IP ROUTE TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.ip.ipRouteTable

ipRouteDest --> ipRouteNextHop/ipRouteMask/ipRouteIfIndex/ipRouteMetric
ipRouteAge/ipRouteProto/ipRouteType/ipRouteInfo

0.0.0.0 --> 128.146.144.1 / int 1 / 0,-1, -1, -1, -1
0 / local (2) / direct (3) / .0.0

End of table.

1 entries.
```

The KarlBridge/KarlBrouter keeps the standard SNMP MIB II statistics on the IP routing table which contains an entry for each routes presently known.

ipRouteDest: The destination IP address of this route. An entry with a value of 0.0.0.0 is considered a default route. Multiple routes to a single destination can appear in the table, but access to such multiple entries is dependent on the table access mechanisms defined by the network management protocol in use.

ipRouteNextHop: The IP address of the next hop of this route. (In the case of a route bound to an interface which is realized via a broadcast media, the value of this field is the agent's IP address on that interface.)

ipRouteMask: Indicate the mask to be logical-ANDed with the destination address before being compared to the value in the ipRouteDest field. For those systems that do not support arbitrary subnet masks, an agent constructs the value of the ipRouteMask by determining whether the value of the correspondent ipRouteDest field belong to a class-A, B, or C network, and then using one of:

mask network	
255.0.0.0	class-A
255.255.0.0	class-B
255.255.255.0	class-C

If the value of the ipRouteDest is 0.0.0.0 (a default route), then the mask value is also 0.0.0.0. It should be noted that all IP routing subsystems implicitly use this mechanism."

ipRouteIfIndex: The index value which uniquely identifies the local interface through which the next hop of this route should be reached. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

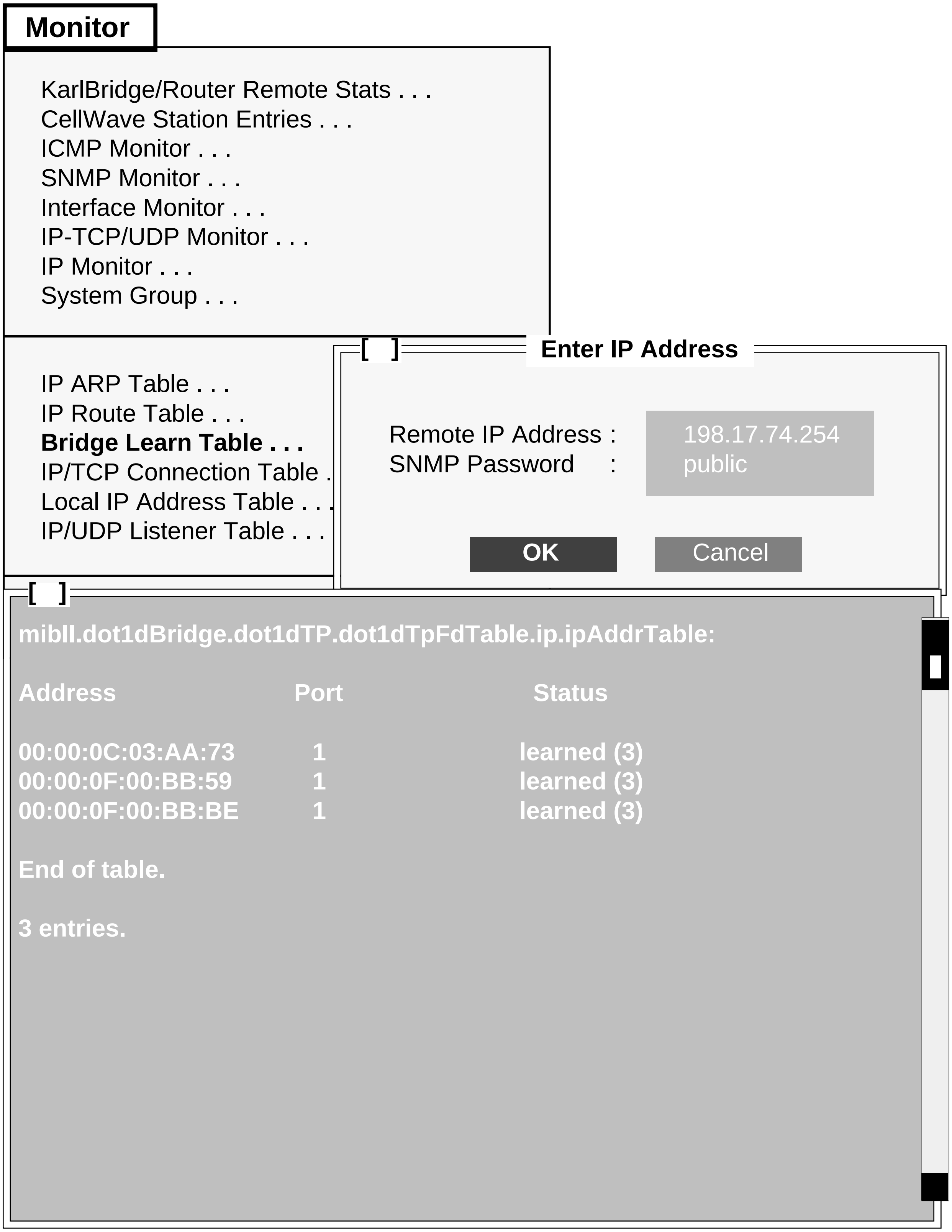
ipRouteMetric: The primary routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route's ipRouteProto value. If this metric is not used, its value should be set to -1.

ipRouteAge: The number of seconds since this route was last updated or otherwise determined to be correct. Note that no semantics of `too old' can be implied except through knowledge of the routing protocol by which the route was learned."

ipRouteProto: The routing mechanism via which this route was learned. Inclusion of values for gateway routing protocols is not intended to imply that hosts should support those protocols. The values are as follows: other(1) -- none of the following; local(2) -- non-protocol information, e.g., manually configured; netmgmt(3) -- entries set via a network management protocol; icmp(4) -- obtained via ICMP e.g. Redirect, icmp(4); Note: the remaining values are all gateway routing protocols; egp(5), ggp(6), ipRouteType: The type of route. Note that the values direct(3) and indirect(4) refer to the notion of direct and indirect routing in the IP architecture. If this object has the value invalid(2) the corresponding entry is invalid. That is, it effectively dissasociates the destination identified with said entry from the route identified with said entry. It can have the following values: other(1) -- none of the following; invalid(2) -- an invalidated route; direct(3) -- route to directly connected (sub-)network; indirect(4) -- route to a non-local host/network/sub-network

ipRouteInfo: A reference to MIB definitions specific to the particular routing protocol which is responsible for this route, as determined by the value specified in the route's ipRouteProto value. If this information is not present, its value is set to 0.

BRIDGE LEARN TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.dot1dBridge.dot1dTP.dot1dTpFdTable.ip.ipAddrTable:

Address                  Port                Status

00:00:0C:03:AA:73          1                 learned (3)
00:00:0F:00:BB:59          1                 learned (3)
00:00:0F:00:BB:BE          1                 learned (3)

End of table.

3 entries.
```

A table that contains information about unicast entries for which the bridge has forwarding and/or filtering information. This information is used by the transparent bridging function in determining how to propagate a received frame.

Address: A unicast MAC address for which the bridge has forwarding and/or filtering information.

Port: Either the value '0', or the port number of the port on which a frame has been seen. A value of '0' indicates that the port number has not been learned but that the bridge does have some forwarding/filtering information about this address.

Status: The status of this entry. The meanings of the values are:

other(1) None of the following.

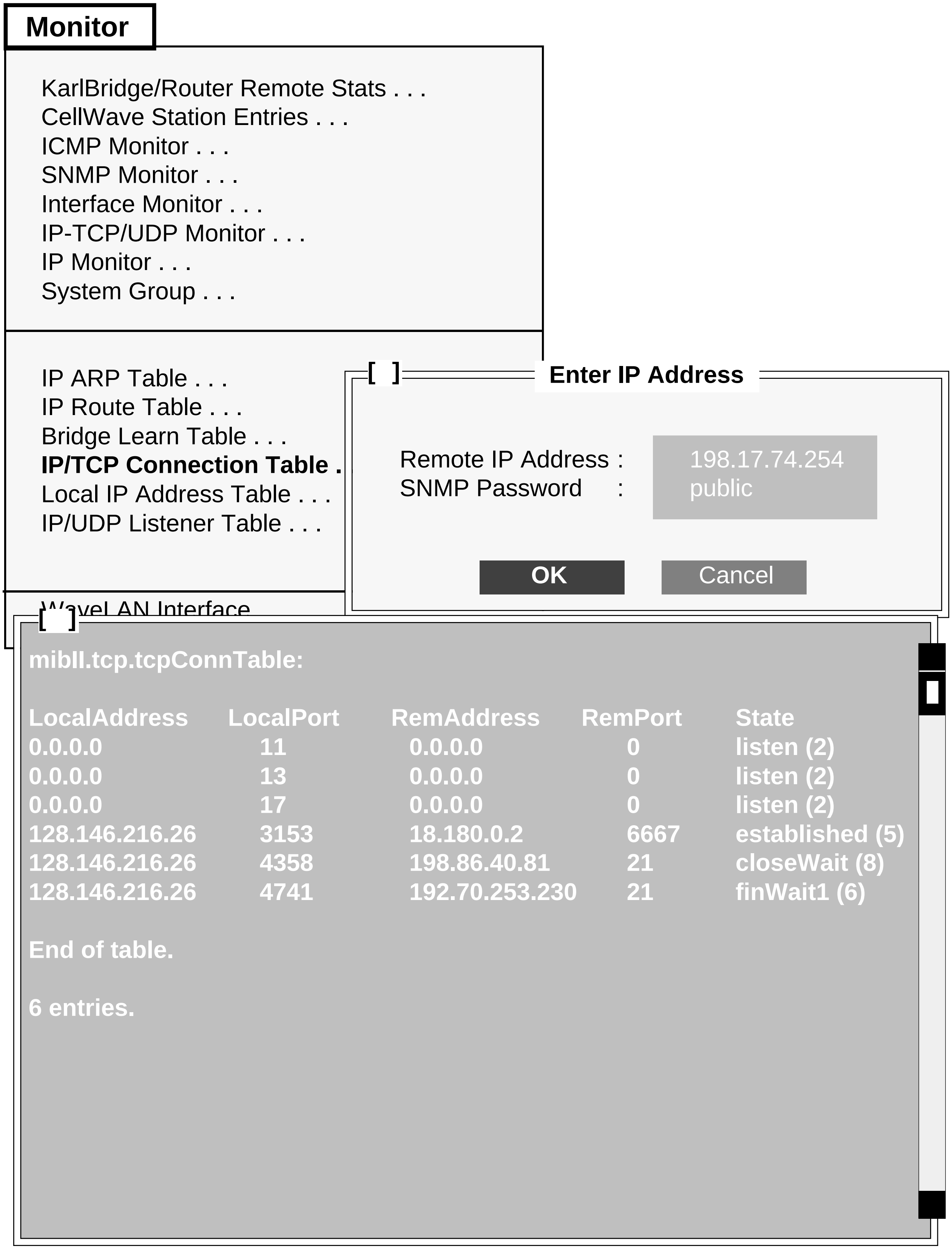
invalid(2) This entry is not longer valid (e.g., it was learned but has since aged-out), but has not yet been flushed from the table.

learned(3) This entry was learned, and is being used.

self(4) This entry represents one of the bridge's addresses. The Port value indicates which of the bridge's ports has this address.

mgmt(5) This entry is also the value of an existing instance in the static table.

IP/TCP CONNECTION TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.tcp.tcpConnTable:

LocalAddress      LocalPort    RemAddress       RemPort    State
0.0.0.0              11        0.0.0.0             0       listen (2)
0.0.0.0              13        0.0.0.0             0       listen (2)
0.0.0.0              17        0.0.0.0             0       listen (2)
128.146.216.26       3153      18.180.0.2          6667    established (5)
128.146.216.26       4358      198.86.40.81        21      closeWait (8)
128.146.216.26       4741      192.70.253.230      21      finWait1 (6)

End of table.

6 entries.
```

This table reports the state of this TCP connections and contains the following fields:

LocalAddress: The local IP address for this TCP connection. In the case of a connection in the listen state which is willing to accept connections for any IP interface associated with the node, the value 0.0.0.0 is used.

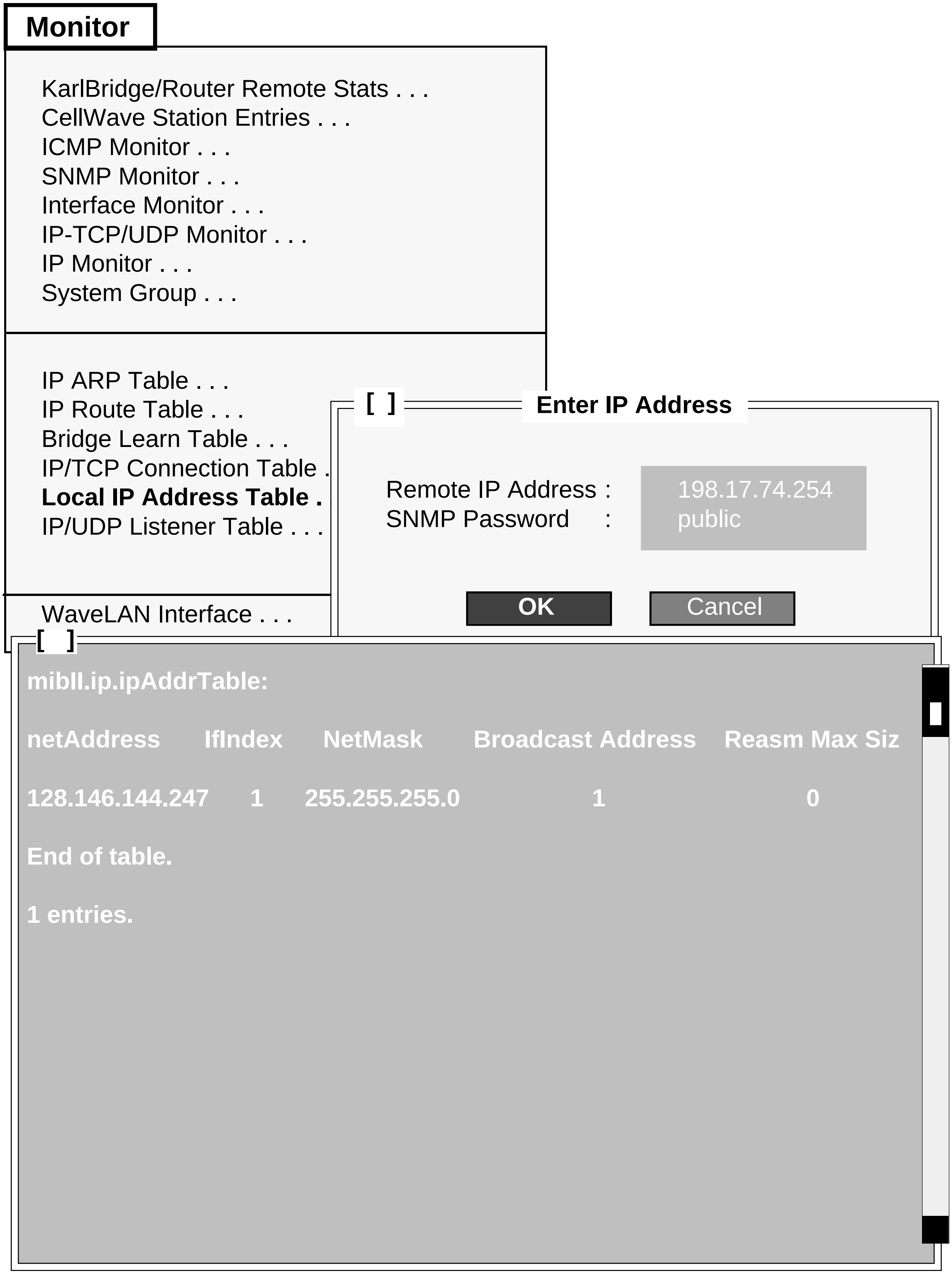
LocalPort: The local port number for this TCP connection.

RemAddress: The remote IP address for this TCP connection.

RemPort: The remote port number for this TCP connection.

State: The state of this TCP connection which can be one of the following: closed(1), listen(2), synSent(3), synReceived(4), established(5), finWait1(6), finWait2(7), closeWait(8), lastAck(9), closing(10), timeWait(11), deleteTCB(12).

LOCAL IP ADDRESS TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

```
mibII.ip.ipAddrTable:

netAddress       IfIndex    NetMask        Broadcast Address   Reasm Max Siz

128.146.144.247     1     255.255.255.0           1                   0

End of table.

1 entries.
```

The table of addressing information relevant to this entity's IP addresses.

netAddress: The IP address to which this entry's addressing information pertains.

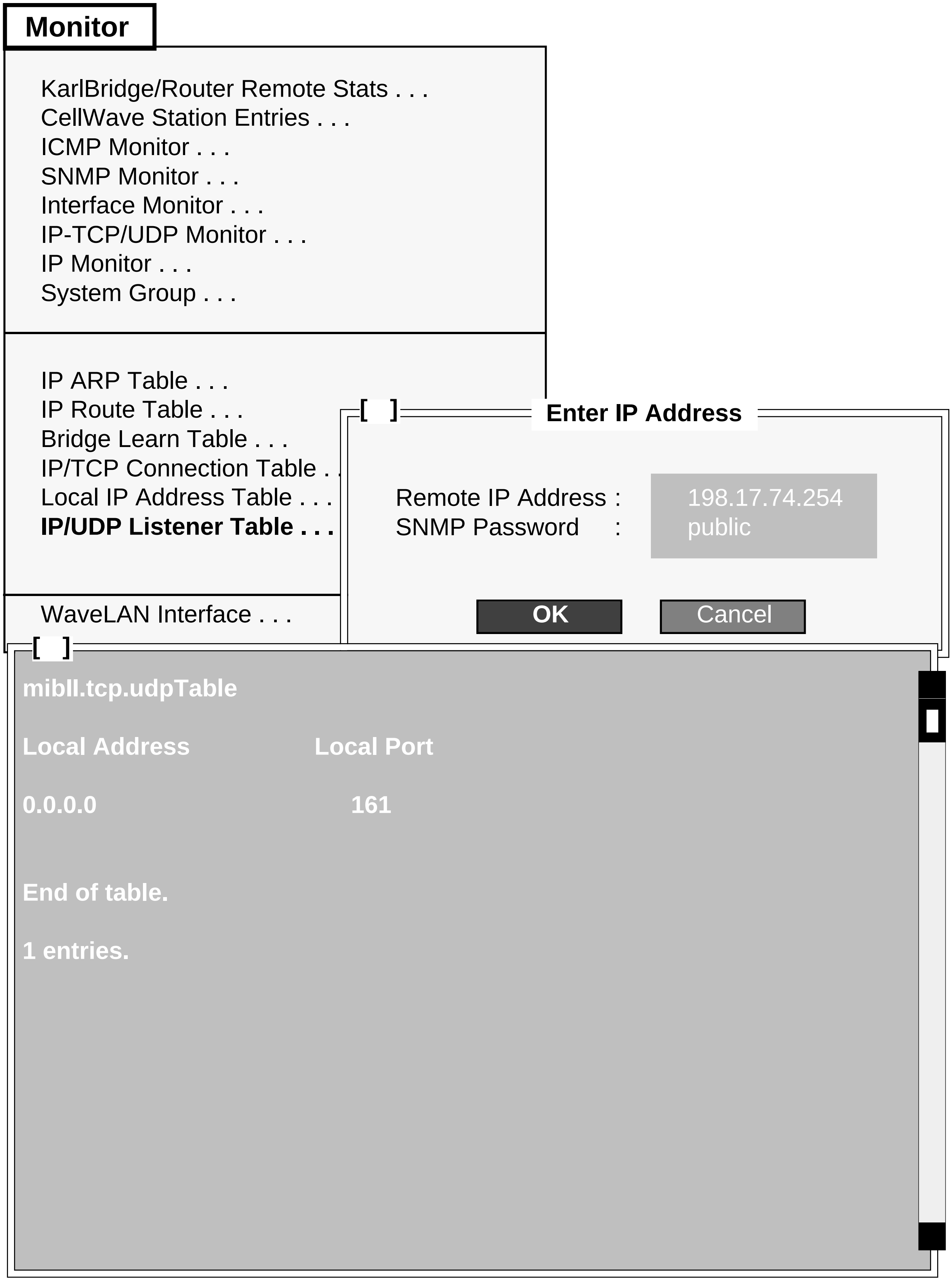
IfIndex: The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

NetMask: The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0.

Broadcast Address: The value of the least-significant bit in the IP broadcast address used for sending datagrams on the (logical) interface associated with the IP address of this entry. For example, when the Internet standard all-ones broadcast address is used, the value will be 1. This value applies to both the subnet and network broadcasts addresses used by the entity on this (logical) interface.

Reasm Max Size: The size of the largest IP datagram which this entity can re-assemble from incoming IP fragmented datagrams received on this interface.

IP/UDP LISTENER TABLE



Remote IP Address : 198.17.74.254
SNMP Password : public

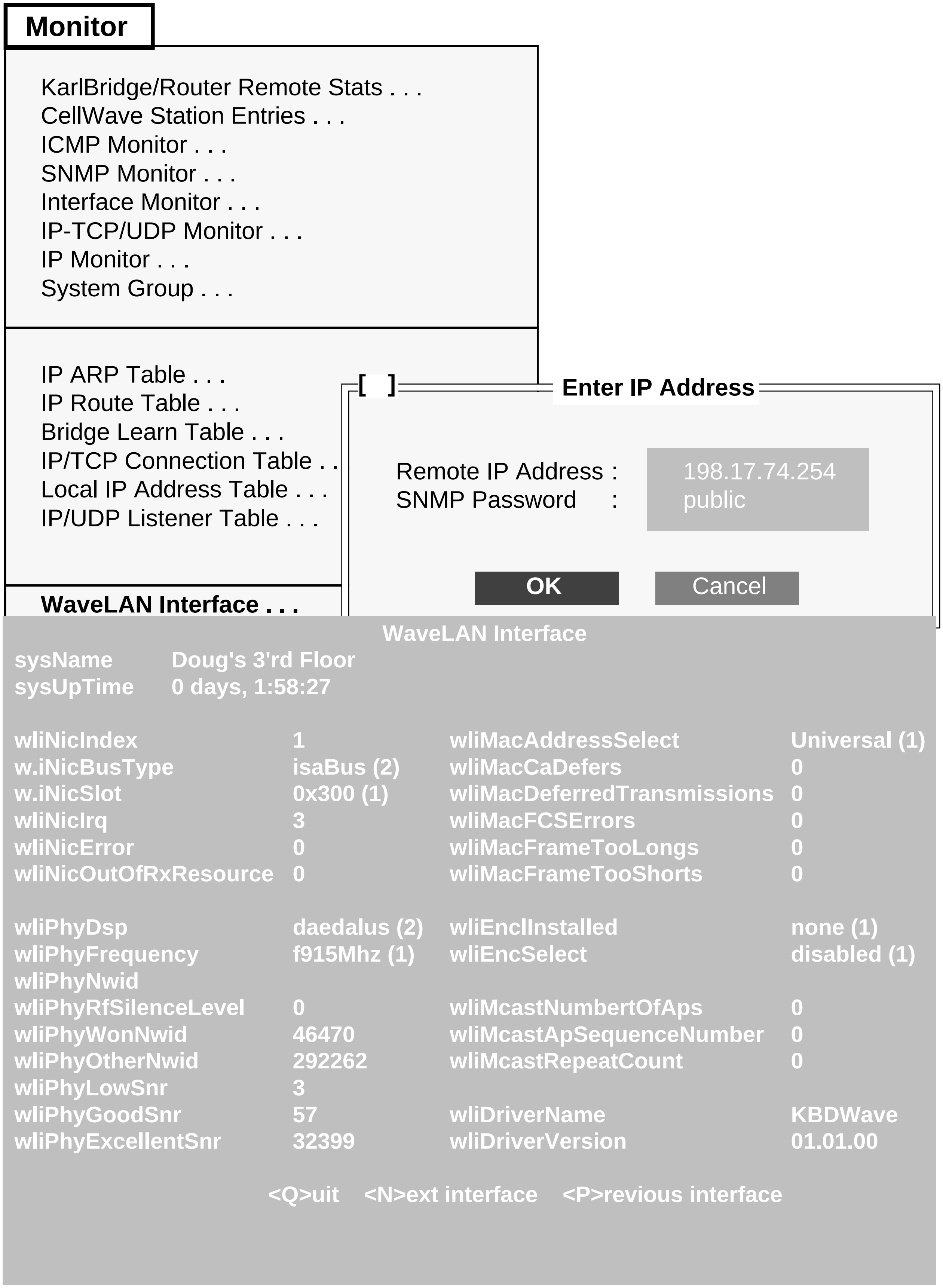
```
mibII.tcp.udpTable

Local Address              Local Port

0.0.0.0                       161


End of table.

1 entries.
```

The UDP listener table contains information about this entity's UDP end-points on which a local application is currently accepting datagrams.

Local Address: The local IP address for this UDP listener. In the case of a UDP listener which is willing to accept datagrams for any IP interface associated with the node, the value 0.0.0.0 is used.

Local Port: The local port number for this UDP listener.

WAVELAN INTERFACE



Enter IP Address

Remote IP Address : 198.17.74.254
SNMP Password : public

```
                            WaveLAN Interface
sysName       Doug's 3'rd Floor
sysUpTime     0 days, 1:58:27

wliNicIndex              1              wliMacAddressSelect            Universal (1)
w.iNicBusType            isaBus (2)     wliMacCaDefers                 0
w.iNicSlot               0x300 (1)      wliMacDeferredTransmissions    0
wliNicIrq                3              wliMacFCSErrors                0
wliNicError              0              wliMacFrameTooLongs            0
wliNicOutOfRxResource    0              wliMacFrameTooShorts           0

wliPhyDsp                daedalus (2)   wliEnclInstalled               none (1)
wliPhyFrequency          f915Mhz (1)    wliEncSelect                   disabled (1)
wliPhyNwid
wliPhyRfSilenceLevel     0              wliMcastNumbertOfAps           0
wliPhyWonNwid            46470          wliMcastApSequenceNumber       0
wliPhyOtherNwid          292262         wliMcastRepeatCount            0
wliPhyLowSnr             3
wliPhyGoodSnr            57             wliDriverName                  KBDWave
wliPhyExcellentSnr       32399          wliDriverVersion               01.01.00

                <Q>uit   <N>ext interface   <P>revious interface
```

The WaveLAN Interface table contains information about this entities ATT/NCR WaveLAN interface board. These numbers have been specified by ATT/NCR and are defined as follows:

WaveLAN Interface NIC Information

wliNicIndex: An index value that uniquely identifies a WaveLAN network interface this NIC information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of ifIndex."

wliNicBusType: The bus-type supported by this NIC. One of the following: xtBus(1), isaBus(2), mcBus(3), pcmcia2Bus(4), wavepointBus(5)

wliNicSlot: The I/O Base Address (ISA/AT) or Slot Number (MC) or Socket Number (PCMCIA) used by this NIC. For ISA/AT (and alike) Base Addresses, the following values are used: 1='0300'H, 2='0390'H, 3='03C0'H, 4='03E0'H.

wliNicIrq: The Interrupt Request Number (IRQ) used by this NIC.

wliNicError: A counter for miscellaneous board errors. It indicates (intermittent) NIC hardware problems.

wliNicOutOfRxResource: A counter for the number of times the NIC is out of resources for the receiver, causing the receiver to be switched off temporarily.

WaveLAN Interface PHY (physical) information

wliPhyDsp: The Digital Signal Processor on the board." The following are valid values: icarus(1), daedalus(2)

wliPhyFrequency: The mid-point of the frequency band this WaveLAN NIC operates in. The following values are valid: 915Mhz(1), 2425Mhz(2), 2460Mhz(3), 2484Mhz(4), 2430Mhz(5) -- actually 2430.5 MHz.

wliPhyNwid: The WaveLAN Network ID (NWID) this RF-modem is currently configured for.

wliPhyRfSilenceLevel: The RF Silence Level as currently read from the RF modem.

wliPhyOwnNwid: Own NWID counter; the number of frames received with the configured NWID.

wliPhyOtherNwid: Other NWID counter; the number of frames received with different NWID than configured.

wliPhyLowSnr: The count of the number of KarlBridge test frames received with a Low signal to noise ratio. (ATT/NCR does not have support for this object)

wliPhyGoodSnr: The count of the number of KarlBridge test frames received with a Good signal to noise ratio. (ATT/NCR does not have support for this object)

wliPhyExcellentSnr: The count of the number of KarlBridge test frames received with a Excellent signal to noise ratio. (ATT/NCR does not have support for this object)

WaveLAN Interface MAC information

MAC status information and control variables for a collection of WaveLAN interfaces attached to a particular system.

wliMacAddressSelect: MAC Address type select. As follows: universal(1), local(2)

wliMacCaDefers: CSMA/CA Defer counter.

wliMacDeferredTransmissions: A counter for the number of frames for which the transmission attempt is delayed because the medium is busy. (same as dot3StatsDeferredTransmissions).

wliMacFCSErrors: A counter for the number of frames received that do not pass the FCS check and/or that are not an integral number of octets in length. WaveLAN hardware does not distinguish between FCS errors and Alignment errors. (same as dot3StatsFCSErrors + dot3StatsAlignmentErrors)"

wliMacFrameTooLongs: A counter for the number of frames received that exceed the maximum permitted frame size for the medium (1518 bytes). (Same as dot3StatsFrameTooLongs)

wliMacFrameTooShorts: A counter for the number of frames received that are shorter than the minimum permitted frame size for the medium (64 bytes)"

WaveLAN Interface Driver information

Driver information for a collection of WaveLAN interfaces attached to a particular system.

wliDriverName: The name of the software driver for this WaveLAN network interface.

wliDriverVersion: The version number of the software driver. A text string as 'mm.nn.pp', where mm = major release number; nn = point release number; pp = optional patch number.

WaveLAN Interface Encryption information

Encryption status information and control variables for a collection of WaveLAN interfaces attached to a particular system.

wliEncInstalled: Which encryption option is installed as follows: none(1), des(2), aes(3)

wliEncSelect: Whether encryption is enabled or disabled as follows: disabled(1), enabled(2)

WaveLAN Interface Multicast Delay group

Information about the Multicast Delay feature for a collection of WaveLAN interfaces attached to a particular system. Implementation of this group is optional.

wliMcastNumberOfAps: The total number of Access Points in the coverage area. Together with wliMcastApSequenceNumber this is used to determine the delays before and after the transmission of each multicast frame. This results in a transmission slot per Access Point per multicast frame. 0 means: no multicast delay specified (use default mechanism).

wliMcastApSequenceNumber : The sequence number of this Access Point in the coverage area. Together with wliMcastNumberOfAps this is used to determine the delays before and after the transmission of each multicast frame. This results in a transmission slot per Access Point per multicast frame.

wliMcastRepeatCount: The number of times a multicast frame transmission is to be repeated.

KarlNet

APPENDIX

COMMON ETHERNET PROTOCOLS

This table contains the protocols that can be specified in the KarlBridge’s “Ethernet Protocol Menu”.

* 0600	Xerox NS IDP
0601	XNS Address Translation (3Mb only)
* 0800	DOD Internet Protocol (IP)
0801	X.75 Internet
0802	NBS Internet
0803	ECMA Internet
* 0804	CHAOSnet
0805	X.25 Level 3
* 0806	Address Resolution Protocol (ARP) (for IP and for CHAOS)
0807	XNS Compatibility
081C	Symbolics Private
0888-088A	Xyplex
0900	Ungermann-Bass network debugger
0A00	Xerox IEEE802.3 PUP
0A01	Xerox IEEE802.3 PUP Address Translation
* 0BAD	Banyan Systems
0BAF	Banyon VINES Echo
1000	Berkeley Trailer negotiation
1001-100F	Berkeley Trailer encapsulation for IP
1234	DCA - Multicast
* 1600	VALID system protocol
1989	Artificial Horizons Aviator dogfight simulator on Sun
3C00	3Com NBP virtual circuit datagram (like XNS SPP) not registered
3C01	3Com NBP System control datagram not registered
3C02	3Com NBP Connect request (virtual cct) not registered
3C03	3Com NBP Connect repsonse not registered
3C04	3Com NBP Connect complete not registered
3C05	3Com NBP Close request (virtual circuit) not registered
3C06	3Com NBP Close response not registered
3C07	3Com NBP Datagram (like XNS IDP) not registered
3C08	3Com NBP Datagram broadcast not registered
3C09	3Com NBP Claim NetBIOS name not registered
3C0A	3Com NBP Delete NetBIOS name not registered
3C0B	3Com NBP Remote adapter status request not registered
3C0C	3Com NBP Remote adapter response not registered
3C0D	3Com NBP Reset not registered
4242	PCS Basic Block Protocol
4321	THD - Diddle
6000	DEC unassigned, experimental
6001	DEC MOP Dump/Load Assistance
6002	DEC MOP Remote Console
6003	DECNET Phase IV, DNA Routing
6004	DEC Local Area Transport (LAT)
6005	DEC diagnostic protocol (at interface initialisation?)
6006	DEC customer protocol
6007	DEC Local Area VAX Cluster (LAVC SCA)
6008 & 6009	DEC unassigned

6010-6014	3Com Corporation
7000	Ungermann-Bass download
7001	Ungermann-Bass NIUs
7002	Ungermann-Bass diagnostic/loopback
7003	Ungermann-Bass ??? (NMC to/from UB Bridge)
7005	Ungermann-Bass Bridge Spanning Tree
7007	OS/9 Microware
7009	OS/9 Net?
7020-7029	LRT (England) (now Sintrom)
7030	Racal-Interlan
7034	Cabletron
8003	Cronus VLN
8004	Cronus Direct
8005	HP Probe protocol
8006	Nestar
8008	AT&T/Stanford University local use
8010	Excelan
8013	Silicon Graphics diagnostic
8014	Silicon Graphics network games
8015	Silicon Graphics reserved
8016	Silicon Graphics XNS NameServer, bounce server
8019	Apollo DOMAIN
802E	Tymshare
802F	Tigan, Inc.
* 8035	Reverse Address Resolution Protocol (RARP)
8036	Aeonic Systems
8037	IPX - Novell Netware
8038	DEC LanBridge Management
8039	DEC unassigned (DSM/DTP?)
803A	DEC unassigned (Argonaut Console?)
803B	DEC unassigned (VAXELN?)
803C	DEC unassigned (NMSV? DNA Naming Service?)
803D	DEC Ethernet CSMA/CD Encryption Protocol
803E	DEC unassigned (DNA Time Service?)
803F	DEC LAN Traffic Monitor Protocol
8040	DEC unassigned (NetBIOS Emulator?)
8041	DEC unassigned (MS/DOS?, Local Area System Transport?)
8042	DEC unassigned
8044	Planning Research Corp.
8046 & 8047	AT&T
8049	ExperData
805B	VMTP (Versatile Message Transaction Protocol, RFC-1045)
805C	Stanford V Kernel, version 6.0
805D	Evans & Sutherland
8060	Little Machines
8062	Counterpoint Computers
8065 & 8066	University of Mass. at Amherst
8067	Veeco Integrated Automation
8068	General Dynamics
8069	AT&T
806A	Autophon

806C	ComDesign
806D	Compugraphic Corporation
806E-8077	Landmark Graphics Corporation
807A	Matra
807B	Dansk Data Elektronik
* 807C	Merit Internodal (or University of Michigan?)
807D-807F	Vitalink Communications
8080	Vitalink TransLAN III Management
8081-8083	Counterpoint Computers
8088-808A	Xyplex
* 809B	EtherTalk (AppleTalk Phase I over Ethernet)
809C-809E	Datability
809F	Spider Systems Ltd.
80A3	Nixdorf Computers
80A4-80B3	Siemens Gammasonics Inc.
80C0-80C3	DCA (Digital Comm. Assoc.) Data Exchange Cluster
80C6	Pacer Software
80C7	Applitek Corporation
80C8-80CC	Intergraph Corporation
80CD-80CE	Harris Corporation
80CF-80D2	Taylor Instrument
80D3-80D4	Rosemount Corporation
80D5	IBM SNA Services over Ethernet
80DD	Varian Associates
80DE-80DF	TRFS (Integrated Solutions)
80E0-80E3	Allen-Bradley
80E4-80F0	Datability
80F2	Retix
80F3	AppleTalk Address Resolution Protocol (AARP)
80F4-80F5	Kinetics
80F7	Apollo Computer
80FF-8103	Wellfleet Communications
8107-8109	Symbolics Private
812B	Talaris
8130	Waterloo Microsystems Inc.
8131	VG Laboratory Systems
8137	Novell (old) NetWare IPX (ECONFIG E option)
8138	Novell, Inc.
8139-813D	KTI
814C	SNMP over Ethernet (see RFC1089)
817D	XTP
81D6	Lantastic
8888	HP LanProbe test?
9000	Loopback (Configuration Test Protocol)
* 9001	3Com XNS Systems Management
* 9002	3Com TCP/IP Systems Management
9003	3Com loopback detection
AAAA	DECNET? (Used by VAX 6220 DEBNI)
FF00	BBN VITAL-LanBridge cache wakeups

* These protocols use Ethernet broadcast

COMMON ETHERNET VENDOR ADDRESSES

This table contains the Vendor portion of the assigned Ethernet Addresses. They may be specified in the KarlBridge's "Ethernet Address Menu".

000002	BBN (internal usage only)
00000C	Cisco
00000E	Fujitsu
00000F	NeXT
000010	Hughes LAN Systems (formerly Sytek)
000011	Tektronix
000015	Datapoint Corporation
000018	Webster (?)
00001B	Novell
00001D	Cabletron
000020	DIAB (Data Intdustrier AB)
000021	SC&C
000022	Visual Technology
000029	IMC
00002A	TRW
0000037	Oxford Metrics Limited
00003C	Auspex
00003D	AT&T
00003F	Syntrex Inc.
000044	Castelle
000046	ISC-Bunker Ramo, An Olivetti Company
000049	Apricot Ltd.
00004B	A.P.T. Appletalk WAN router
00004C	NEC Corporation
00004F	Logicraft 386-Ware P.C. Emulator
000050	Radisys Corporation
000051	HOB Electronic GMGH & Co.
000052	ODS
000055	AT&T
000058	Racore Computer Products Inc.
00005A	(Schneider & Koch in Europe and Syskonnect)
00005A	Xerox 806 (unregistered)
00005D	RCE
00005E	U.S. Department of Defence (IANA)
000061	Gateway Communications
000062	Honeywell
000064	Yokogawa Digital Computer Corp.
000065	Network General
000068	Rosemount Controls
000069	Silicon Graphics(?)
00006B	MIPS

00006D	Cray Communications, Ltd.
00006E	Artisoft, Inc.
00006F	Madge Networks Ltd.
000074	Ricoh Company Ltd.
000077	MIPS(?), Interphase(?)
000079	Networth Inc.
00007A	Ardent
00007B	Research Machines
00007D	Cray Research Superservices Inc.
00007F	Linotype
000080	Imagen(?) Also shows as "Harris (3M) (new)"
000081	Synoptics
000084	Aquila (?), ADI Systems Inc.(?)
000086	Gateway (?), Megahertz Corporation(?)
000089	Cayman Systems Gatorbox
00008A	Datahouse Information Systems
00008E	Jupiter(?), Solbourne(?)
000093	Proteon
000094	Asante
000095	Sony/Tektronix
000097	Epoch
000098	Crossomm Corporation
000099	Memorex Telex Corporation
00009F	Ameristar Technology
0000A0	Sanyo Electronics
0000A2	Wellfleet
0000A3	Network Application Technology (NAT)
0000A4	Acorn Computers Ltd.
0000A5	Compatible Systems Corporation
0000A6	Network General (internal assignment)
0000A7	Network Computing Devices (NCD) X-terminals
0000A8	Stratus Computer, Inc.
0000A9	Network Systems
0000AA	Xerox machines
0000AC	Apollo
0000AE	Dassault Automatismes
0000AF	Nuclear Data Acquisition Interface Modules (AIM)
0000B0	RND (RAD Network Devices)
0000B1	Alpha Microsystems Inc.
0000B3	CIMLinc
0000B5	Datability Terminal Servers
0000B6	Micro-Matic Research
0000B7	Dove Computer Corporation
0000BC	Allen-Bradley Co. Inc.
0000C0	Western Digital now SMC
0000C1	Olicom A/S

0000C6	HP Intelligent Networks Operation
0000C8	Altos
0000C9	Emulex Terminal Servers
0000CC	Densan Co. Ltd.
0000CD	Industrial Research Ltd.
0000D0	Develcon Electronics, Ltd.
0000D1	Adaptec, Inc. "Nodem" product
0000D2	SBE Inc.
0000D7	Dartmouth College (NED Router)
0000D8	3Com? Novell? PS/2
0000DD	Gould
0000DE	Unigraph
0000E2	Acer Counterpoint
0000E3	Integrated Micro Products Ltd.
0000E6	Aptor Produits de Comm. Indust.
0000E7	Star Gate Technologies
0000E8	Accton Technology Corporation
0000E9	Isicad Inc.
0000ED	April
0000EE	Network Designers Limited(?)
0000EF	Alantec
0000F0	Samsung
0000F2	Spider Communications
0000F3	Gandalf
0000F4	Allied Telesis, Inc.
0000F6	A.M.C. (Applied Microsystems Corp.)
0000F8	Digital Equipment Corp.
0000FB	Rechner Zur Kommunikation
0000FD	High Level Hardware (Orion, UK)
000102	BBN internal usage (not registered)
000143	IEEE 802
000163	NDC (National Datacomm Corporation)
000168	W&G (Wandel & Goltermann)
0001C8	Thomas Conrad Corp.
000267	Node Runner Inc.
000701	Racal-Datacom
001700	Kabel
002002	Seritech Enterprise Co. Ltd.
002006	Garrett Communications Inc.
002008	Cable & Computer Technology
002009	Packard Bell Elec. Inc.
00200C	Adastra Systems Corp.
00200E	Satelite Technology Mgmt, Inc.
002011	Canopus Co. Ltd.
002014	Global View Co. Ltd.
002015	Actis Computer SA.

002016	Showa Electric Wire and Cable Co.
002017	Orbotech
00201C	Excel Inc.
00201E	Netquest Corporation
00201F	Best Power Technology Inc.
002021	Algorithms Software Pvt. Ltd.
002022	Teknique, Inc.
002024	Pacific Communications Sciences
002025	Control Technology Inc.
002027	Ming Fortune Industry Co. Ltd.
002028	West Egg Systems Inc.
002029	Teleprocessing Products Inc.
00202C	Welltronix Co. Ltd.
00202E	Daystar Digital
002030	Analog & Digital Systems
002032	Alcatel Taisel
002033	Synapse Technologies Inc.
002036	BMC Software
00203A	Digital Biometrics Inc.
00203B	Wisdm Ltd.
00203C	Eurotime AB
00203F	Juki Corporation
002042	Datametrics Corp
002044	Genitech Pty. Ltd.
002045	Solcom Systems Ltd.
002048	Fore Systems Inc.
002049	Comtron Inc.
00204A	Pronet GMBH
00204B	Autocomputer Co. Ltd.
00204C	Mitron Computer Pte. Ltd.
00204D	Inovis GMBH
00204E	Network Security Systems Inc.
00204F	Deutsche Aerospace AG.
002050	Korea Computer Inc.
002051	Phoenix Data Communications Corp.
002053	Huntsville Microsystems Inc.
002056	Neoproducts
00205B	Skyline Technology
00205D	Nanomatic OY.
00205F	Gammadata Computer GMBH
002061	Dynatech Communications Inc.
002063	Wipro Infotech Ltd.
002064	Protec Microsystems Inc.
002066	General Magic Inc.
002068	Isdyne
002069	ISDN Systems Corporation

00206A	Osaka Computer Corporation
00206D	Data Race Inc.
00206E	Xact Inc.
002074	Sungwoon Systems
002076	Reudo Corporation
002077	Kardios Systems Corporation
002078	Runtop Inc.
00207F	Kyoelsangyo Co. Ltd.
002082	Oneac Corporation
002083	Presticom Inc.
002084	OCE Graphics USA Inc.
002088	Global Village Communication
002089	T3Plus Networking Inc.
00208A	Sonix Communications Ltd.
00208B	Lapis Technologies Inc.
00208C	Galaxy Networks Inc.
00208E	Chevin Software Eng Ltd.
002095	Riva Electronics
002096	Siebe Environmental Controls
002099	Bon Electric Co. Ltd.
00209B	Ersat Electronic GMBH
00209C	Primary Access Corp.
00209D	Lippert Automationstechnik
0020A1	Dovatron
0020A4	Multipoint Networks
0020A6	Proxim Inc.
0020A9	White Horse Industrial
0020AA	NTL Advanced Products
0020AC	Interflex Datensysteme GMBH
0020AE	Ornet Data Communication Tech.
0020AF	3COM Corporation
0020B0	Gateway Devices Inc.
0020B1	Comtech Research Inc.
0020B3	Scltec Communications Systems
0020B6	Agile Networks Inc.
0020BA	Center for High Performance
0020BB	Zax Corporation
0020BE	LAN Access Corporation
0020BF	Aehr Test Systems
0020C2	Texas Memory Systems Inc.
0020C5	Eagle Technology
0020C6	Nectec
0020C8	Larscom Inc.
0020C9	Victron BV
0020CA	Digital Ocean
0020CC	Digital Services Ltd.

0020CD	Hybrid Networks Inc.
0020CE	Logical Design Group Inc.
0020D1	Microcomputer Systems (M) SDN
0020D2	Rad Data Communicatiosn Ltd.
0020D3	OST (Quest Standard Telematiqu)
0020D6	Lannair Ltd.
0020DB	XNET Technology Inc.
0020DC	Densitron Taiwan Ltd.
0020E1	Alamar Electornics
0020E7	B & W Nuclear Service Company
0020E8	Datatrek Corporation
0020E9	Dantel
0020EA	Efficient Networks Inc.
0020EC	Techware Systems Corp.
0020ED	Giga-Byte Technology Co. Ltd.
0020EE	Gtech Corporation
0020EF	U S C Corporation
0020F1	Altos India Ltd.
0020F2	Spectrix Corp
0020F5	Pan Dacom Telecomcations GMBH
0020F6	NetTek & KarlNet Inc.
0020F8	Carrera Computers Inc.
0020FF	Symmetrical Technologies
004001	Zero One Technology Co. Ltd.
004009	Tachibana Tectron Co Ltd.
00400C	General Micor Systems Inc.
00400D	Lannet Data Communicatiosn Ltd.
004010	Sonic Systems
004013	NTT Data Comm. Systems Corp.
004014	Comsoft GMBH
004015	Ascom Infrasys AG
00401F	Colorgraph Ltd.
004020	Pinacl Communications
004023	Logic Corporation
004025	Molecular Dynamics
004026	Melco Inc.
004027	SMC Massachusetts Inc.
00402A	Canoga-Perkins
00402B	TriGem
00402F	XLNT Designs Inc.
004030	GK Computer
004032	Digital Communications
004033	Addtron Technology Co. Ltd.
004039	Optec Daiichi Denko Co. Ltd.
00403C	Forks Inc.
004041	Fujikura Ltd.

004043	Nokia Data Communications
004048	SMD Informatica S.A.
00404C	Hypertec Pty Ltd.
00404D	Telecommunications Techniques
00404F	Space & Naval Warfare Systems
004050	Ironics Inc.
004052	Star Technologies Inc.
004054	Thinking Machines Corp.
004057	Lockheed Sanders
004059	Yoshida Kogyo K K
00405B	Funasset Limited
00405D	Star-Tek Inc.
004066	Hitachi Cable Ltd.
004067	Omnibyte Corporation
004068	Extended Systems
004069	Lemcom Systems Inc.
00406A	Kentek Information Systems Inc.
00406E	Corollary Inc.
00406F	Sync Research Inc.
004074	Cable and Wireless Communications Inc.
004076	AMP Incorporated
004078	Wearnes Automation Pte Ltd.
00407F	Agema Infrared Systems AB
004082	Laboratory Equipment Corp
004085	SAAB Instruments AB
004086	Michels & Kleberhoff Computer
004087	Ubitrex Corporation
00408A	TPS Teleprocessing Sys GMBH
00408C	Axis Communications AB
00408E	CXR/Digilog
00408F	WM-Data Minfo AB
004091	Procomp Industria Electronca
004092	ASP Computer Products Inc.
004094	Shographics Inc
004095	R.P.T. Intergroups Intl. Ltd.
004096	Telesystems SLW Inc.
00409A	Network Express Inc.
00409C	Transware
00409D	Digiboard Inc.
00409E	Concurrent Technologies Ltd.
00409F	Lancast/Casat Technology Inc.
0040A4	Rose Electronics
0040A6	Cray Research Inc.
0040AA	Valmet Automation Inc.
0040AD	SMA Regelsysteme GMBH
0040AE	Delta Controls Inc.

0040B4	3Com K.K.
0040B5	Video Technology Computers Ltd.
0040B6	Computerm Corporation
0040B9	MACQ Electronique SA.
0040BD	Starlight Networks Inc.
0040C0	Vista Controls Corporation
0040C1	Bizerba-Werke Wilheim Kraut
0040C2	Applied Computing Devices
0040C3	Fischer and Proter Co.
0040C5	Micom Communications Corp.
0040C6	Fibernet Research Inc.
0040C8	Milan Technology Corp.
0040CC	Silcom Manuf'g Technology Inc.
0040CF	Strawberry Tree Inc.
0040D2	Pagine Corporation
0040D4	Gage Talker Corp.
0040D7	Studio Gen Inc.
0040D8	Ocean Office Automation Ltd.
0040DC	Tritec Electronic GMBH
0040DF	Digalog Systems Inc.
0040E1	Marner International Inc.
0040E2	Mesa Ridge Technologies Inc.
0040E3	Quin Systems Ltd.
0040E4	E-M Technology Inc.
0040E5	Sysbus Corporation
0040E7	Arnos Instruments & Computer Systems
0040E9	Accord Systems Inc.
0040EA	Plain Tree Systems Inc.
0040ED	Network Controls Int'natl Inc.
0040F0	Micro Systems Inc.
0040F1	Chuo Electronics Co. Ltd.
0040F4	Cameo Communications Inc.
0040F5	OEM Engines
0040F6	Katron Computers Inc.
0040F9	Combinet
0040FA	Microboards Inc.
0040FD	LXE
0040FF	Telebit Corporation
00608C	3Com Corporation
008000	Multitech Systems Inc.
008004	Antlow Computers Ltd.
008005	Cactus Computers Inc.
008006	Compuadd Corporation
008007	DLOG NC Systeme
00800D	Vosswinkel F.U.
00800F	SMC (Standard Microsystem Corp.)

008010	Commodore
008015	Seiko Systems Inc.
008017	PFU
008016	Wandel and Goltermann
008018	Kobe Steel Ltd.
008019	Dayna Communications Inc.
00801A	Bell Atlantic
00801B	Kodiak Technology
008021	Newbridge Research Corp.
008023	Integrated Business Networks
008024	Kalpana Inc.
008026	Network Products Corporation
008029	Microdyne Corporation
00802A	Test Systems & Simulations Inc.
00802C	The Sage Group PLC
00802D	XYLogics Inc.
00802E	Plexcom, Inc.
008034	SMT-Goupil
008035	Technology Works
008037	Telefon AB LM Ericsson Crop.
008038	Data Research & Applications
00803B	APT Communications Inc.
00803D	Surigiken Co. Ltd.
00803E	Synernetics
008042	Force Computers
008043	Networld Inc.
008044	Systech Computer Corp.
008045	Matsushita Electric Ind. Co.
008046	University of Toronto
008049	Nissin Electric Co. Ltd.
00804C	Contec Co. Ltd.
00804D	Cyclone Microsystems Inc.
008051	Fibermux
008052	Network Professor
008057	Adsoft Ltd.
00805A	Tulip Computers Internat'l B.V.
00805B	Condor Systems Inc.
008062	Interface Co.
008063	Richard Hirschmann GBMH & Co.
008067	Square D Company
008069	Computone Systems
00806A	ERI (Empac Research Inc.)
00806B	Schmid Telecommunication
00806C	Cegelec Projects Ltd.
00806D	Centrury Systems Corp.
00806E	Nippon Steel Corporation

00806F	Onelan Ltd.
008071	SAI Technology
008072	Microplex Systems Ltd.
008074	Fisher Controls
008079	Microbus Designs Ltd.
00807B	Artel Communications Corp.
00807C	FiberCom
00807E	Southern Pacific Ltd.
008082	PEP Modular Computers GMBH
008086	Computer Generations Inc.
008087	Okidata
008088	Victor Company of Japan Ltd
008089	Tecnetics (Pty) Ltd.
00808A	Summit Microsystems Corp.
00808B	Dacoll Limited
00808C	Frontier Software Development
00808D	Westcoast Technology B.V.
00808E	Radstone Technology
008090	Microtek International Inc.
008092	Japan Computer Industry Inc.
008093	Xyron Corporation
008094	Sattcontrol AB
008096	HDS (Human Designed Systems) X terminals
008098	TDK Corporation
00809A	Novus Networks Ltd.
00809B	Justsystem Corporation
00809D	Datacraft Manufactur'g Pty. Ltd.
00809F	Alcatel Business Systems
0080A1	Microtest
0080A3	Lantronix
0080A6	Republic Technology Inc.
0080A7	Measurex Corp.
0080AC	Imlogix, Division of Genesys
0080AD	Cnet Technology Inc.
0080AE	Hughes Network Systems
0080AF	Allumer Co. Ltd.
0080B1	Softcom A/S
0080B2	NET (Network Equipment Technologies)
0080BA	Specialix (Asia) Pte. Ltd.
0080C2	IEE 802 Committe, Fermi Nat'l Lab
0080C7	Xircom, Inc.
0080C8	D-Link (also Solectek Pocket Adapters)
0080C9	Alberta Microelectronic Centre
0080CE	Broadcast Television Systems
0080D0	Computer Products International
0080D3	Shiva - Appletalk-Ethernet interface

0080D4	Chase Limited
0080D7	Fantum Engineering Inc.
0080D8	Network Peripherals
0080DA	Bruel & Kjaer
0080DD	GMX Inc. / GIMIX
0080E0	XTP Systems Inc.
0080E7	Lynwood Scientific Dev Ltd.
0080EA	The Fiber Company
0080F0	Kyushu Matsushita Electric Co.
0080F1	Opus
0080F3	Sun Electronics Corp.
0080F4	Telemecanique Electrique
0080F5	Quantel Ltd.
0080FB	BVM Limited
0080FE	Azure Technologies Inc.
00AA00	Intel
00B0D0	Computer Products International
00C000	Lanoptics Ltd.
00C001	Diatek Patient Managment
00C002	Sercomm Corporation
00C003	Globalnet Communications
00C004	Japan Business Computer Co. Ltd.
00C005	Livingston Enterprise Inc.
00C006	Nippon Avionics Co. Ltd.
00C007	Pinnacle Data Systems Inc.
00C008	Seco SRL
00C009	KT Technology (S) Pte Ltd.
00C00A	Micro Craft
00C00B	Norcontrol A.S.
00C00D	Advanced Logic Research Inc.
00C00E	Psitech Inc.
00C00F	Quantum Software Systems Ltd.
00C011	Interactive Computing Devices
00C012	Netspan Corporation
00C013	Netrix
00C014	Telematics Calabasas Int'l Inc.
00C015	New Media Corporation
00C016	Electronic Theatre Controls
00C018	Lanart Corporation
00C019	Leap Technology Inc.
00C01A	Corometrics Medical Systems
00C01B	Socket Communications Inc.
00C01C	Systems Information
00C01D	Grand Junction Networks Inc.
00C01F	S.E.R.C.E.L.
00C020	Arco Electronic Control Ltd.

00C021	Netexpress
00C023	Tutankhamon Electronics
00C024	Eden Sistemas de Computacao SA
00C025	Dataproducts Corporation
00C027	Cipher Systems Inc.
00C028	Jasco Corporation
00C029	Kabel Rheydt AG
00C02A	Ohkura Electric Co. Ltd.
00C02B	Gerloff Gesellschaft
00C02C	Centrum Communications Inc.
00C02D	Fuji Photo Film Co. Ltd.
00C02E	Netwiz
00C02F	Okuma Corporation
00C030	Integrated Engineering B.V.
00C031	Design Research Systems Inc.
00C032	I-Cubed Limited
00C033	Telebit Communications APS
00C034	Dale Computer Corporation
00C035	Quintar Company
00C036	Raytech Electronic Corp.
00C039	Silicon Systems
00C03B	Multiaccess Computing Corp.
00C03C	Tower Tech S.R.L
00C03D	Wiesemann & Theis GMBH
00C03E	FA. Gebr. Heller GMBH
00C03F	Stores Automated Systems Inc.
00C040	ECCI
00C041	Digital Transmission Systems
00C042	Datalux Crop.
00C043	Stratacom
00C044	Emcom Corporation
00C045	Isolation Systems Ltd.
00C046	Kemitron Ltd
00C047	Unimicro Systems Inc.
00C048	Bay Technical Associates
00C04B	Creative Microsystems
00C04D	Mitec Inc.
00C04E	Comtrol Corporation
00C050	Toyo Denki Seizo K.K.
00C051	Advanced Integration Research
00C055	Modular Computing Technologies
00C056	Somelec
00C057	Myco Electronics
00C058	Data Expert Corp.
00C059	Nippondenso Co. Ltd.
00C05B	Networks Northwest Inc.

00C05C	Elonex PLC
00C05D	L&N Technologies
00C05E	Vari-Lite Inc.
00C060	ID Scandinavia AS
00C061	Solectek Corporation
00C063	Morning Star Technologies Inc.
00C064	General Datacomm Ind Inc.
00C065	Scope Communications Inc.
00C066	Docupoint Inc.
00C067	United Barcode Industries
00C068	Philip Drake Electronics Ltd.
00C069	California Microwave Inc.
00C06A	Zahner-Elektrik GMBH & Co. KG
00C06B	OSI Plus Corporation
00C06C	Svec Computer Corp.
00C06D	Boca Research Inc.
00C06F	Komatsu Ltd.
00C070	Sectra Secure Transmission AB
00C071	Areanex Communications Inc.
00C072	KNX Ltd.
00C073	Xedia Corporation
00C074	Toyoda Automatic Loom
00C075	Xante Corporation
00C076	I-Data International A S
00C077	Daewod Telecom Ltd
00C078	Computer Systems Engineering
00C079	Fonsys Co. Ltd.
00C07A	Priva B.V.
00C07D	Risc Developments Ltd.
00C07F	Nupon Computing Corp.
00C080	Netstar Inc.
00C081	Metrodata Ltd.
00C082	Moore Products Co.
00C084	Datalink Corp. Ltd.
00C086	The Lynk Corporation
00C087	UUNET Technologies Inc.
00C089	Telindus Distribution
00C08A	Lauterbach Datentechnik GMBH
00C08B	Risq Modular Systems Inc.
00C08C	Performance Technologies Inc.
00C08D	Tronix Product Development
00C08E	Network Information Technology
00C08F	Matsushita Electric Works Ltd
00C090	Praim S.R.L.
00C091	Jabil Circuit Inc.
00C092	Mennen Medical Inc.

00C093	Alta Research Corp.
00C096	Tamura Corporation
00C097	Archipsel SA
00C098	Chuntex Electronic Co. Ltd.
00C099	Yoshiki Industrial Co. Ltd.
00C09B	Reliance Comm/Tec R-Tec
00C09C	TOA Electronic Ltd.
00C09D	Distributed Systems Int'l Inc.
00C09F	Quanta Computer Inc.
00C0A0	Advanced Micro Research Inc.
00C0A1	Tokyo Denshi Sekei Co.
00C0A2	Intermedium A/S
00C0A3	Dual Enterprises Corporation
00C0A4	Unigraf OY
00C0A7	Seel Ltd.
00C0A8	GVC Corporation
00C0A9	Barron McCann Ltd.
00C0AA	Silicon Valley Computer
00C0AB	Jupiter Technology Inc.
00C0AC	Gambit Computer Communications
00C0AD	Marben Communication Systems
00C0AE	Towercom Co. Inc. (PC House)
00C0AF	Teklogix Inc.
00C0B0	GCC Technologies Inc.
00C0B2	Norand Corporation
00C0B3	Comstat Datacomm Corporation
00C0B4	Myson Technology Inc.
00C0B5	Corporate Network Systems Inc.
00C0B6	Meridian Data Inc.
00C0B7	American Power Conversion Corp.
00C0B8	Fraser's Hill Ltd.
00C0B9	Funk Software Inc.
00C0BA	Netvantage
00C0BB	Forval Creative Inc.
00C0BD	Inex Technologies Inc.
00C0BE	Alcatel - Sel
00C0BF	Technology Concepts Ltd.
00C0C0	Shore Microsystems Inc.
00C0C1	Quad/Graphics Inc.
00C0C2	Infinite Networks Ltd.
00C0C3	Acuson Computed Sonography
00C0C4	Computer Operational
00C0C5	SID Informatica
00C0C6	Personal Media Corp.
00C0C8	Micro Byte Pty Ltd.
00C0C9	Bailey Controls Co.

00C0CA	Alfa Inc.
00C0CB	Control Technology Corporation
00C0CD	Comelta S.A.
00C0D0	Ratoc System Inc.
00C0D1	Comtree Technology Corporation
00C0D2	Syntellect Inc.
00C0D4	Axon Networks Inc.
00C0D5	Quancom Electronic GMBH
00C0D6	J1 Systems Inc.
00C0D9	Quinte Network Confidentiality
00C0DB	IPC Corporation (PTE) Ltd.
00C0DC	EOS Technologies Inc.
00C0DE	Zcomm Inc.
00C0DF	KYE Systems Corp.
00C0E1	Sonic Solutions
00C0E2	Calcomp Inc.
00C0E3	Ositech Communications Inc.
00C0E4	Landis & GYR Powers Inc.
00C0E5	Gespac S.A.
00C0E6	Txport
00C0E7	Fiberdata
00C0E8	Plexcom Inc.
00C0E9	Oak Solutions Ltd
00C0EA	Array Technology Ltd.
00C0EB	SEH Comutertechnik GMBH
00C0EC	Dauphin Technology
00C0ED	US Army Electronic
00C0EE	Kyocera Corporation
00C0EF	Abit Corporation
00C0F0	Kingston Technology Corp.
00C0F1	Shinko Electric Co. Ltd.
00C0F2	Transition Engineering Inc.
00C0F3	Network Communications Corp.
00C0F4	Interlink System Co. Ltd.
00C0F5	Metacomp Inc.
00C0F6	Celan Technology Inc.
00C0F7	Engage Communication Inc.
00C0F8	About Computing Inc.
00C0F9	Harris and Jeffries Inc.
00C0FA	Canary Communications Inc.
00C0FB	Advanced Technology Labs.
00C0FC	ASDG Inc.
00C0FD	Prosum
00C0FF	Box Hill Systems Corporation
00DD00	Ungermann-Bass - IBM RT
00DD01	Ungermann-Bass

00EFE5	IBM (3Com card) Micro channel interface
020406	BBN internal usage (not registered)
020701	Racal Datacom (Micom/Interlan)
026060	3Com
026086	Satelcom MegaPac (UK)
02608C	3Com IBM PC; Imagen; Valid; Cisco; Macintosh
02CF1F	CMC Masscomp; Silicon Graphics; Prime EXL
02E6D3	BTI (Bus-Tech, Inc.) IBM Mainframes
080001	Computer Vision
080002	3Com (formerly Bridge)
080003	ACC (Advanced Computer Communications)
080005	Symbolics LISP machines
080007	Apple Computer Inc.
080008	BBN
080009	Hewlett-Packard
08000A	Nestar Systems
08000B	Unisys Corporation
08000D	IInternational Computers Ltd.
08000E	NCR/AT&T
08000F	SMC (Standard Microsystems Corp.)
080010	AT&T [misrepresentation of 800010?]
080011	Tektronix, Inc.
080014	Excelan BBN Butterfly, Masscomp, Silicon Graphics
080017	NSC (National Semiconductor Corp.)
08001A	Data General
08001B	Data General
08001E	Apollo
08001F	Sharp Corporation
080020	Sun
080022	NBI (Nothing But Initials)
080023	Matsushita Denso
080025	CDC
080026	Norsk Data (Nord)
080027	PCS Computer Systems GmbH
080028	Texas Instruments
08002B	DEC
08002E	Metaphor
08002F	Prime 50-Series LHC300
080030	CERN
080036	Intergraph CAE stations
080037	Fujitsu-Xerox
080038	Bull
080039	Spider Systems Ltd.
08003B	Torus Systems
08003E	Motorola VME bus processor modules
080041	DCA (Digital Comm. Assoc.)

080044	DSI (DAVID Systems, Inc.)
080046	Sony
080047	Sequent
080048	Eurotherm Gauging Systems
080049	Univation
08004C	Encore
08004E	BICC
080051	Experdata
080056	Stanford University
080057	Evans & Sutherland (?)
080058	DECsystem-20
08005A	IBM
080067	Comdesign
080068	Ridge
080069	Silicon Graphics
08006A	ATTst (?)
08006E	Excelan
080070	Mitsubishi
080074	Casio Computer Co. Ltd.
080075	DDE (Danish Data Elektronik A/S)
080077	TSL (now Retix)
080079	Silicon Graphics
08007C	Vitalink TransLAN III
080080	XIOS
080081	Crossfield Electronics
080083	Seiko Denshi
080086	Imagen/QMS
080087	Xyplex terminal servers
080089	Kinetics AppleTalk-Ethernet interface
08008B	Pyramid
08008D	XyVision machines
08008E	Tandem
08008F	Chipcom Corporation
080090	Retix Inc. Bridges
10005A	IBM
1000D4	DEC
1000E0	Apple A/UX (modified addresses for licensing)
400003	NetWare (?)
475443	GTC (Not registered!) (This number is a multicast!)
484453	HDS ???
800010	AT&T (misrepresented as 080010?)
AA0000	DEC obsolete
AA0001	DEC obsolete
AA0002	DEC obsolete
AA0003	DEC Global physical address for some DEC machines
AA0004	DEC Local logical address for systems running DECNET

COMMON ETHERNET MULTICAST ADDRESSES

This table contains commonly used Ethernet Multicast Addresses and the Ethernet Protocols they use. They may be specified in the KarlBridge's "Ethernet Address Menu".

01-00-1D-00-00-00	-802-	Cabletron PC-OV PC discover
01-00-1D-42-00-00	-802-	Cabletron PC-OV Bridge discover
01-00-1D-52-00-00	-802-	Cabletron PC-OV MMAC discover
01-00-5E-00-00-00	0800	DoD Internet Multicast (RFC-1112)
through		
01-00-5E-7F-FF-FF		
01-00-5E-80-00-00		DoD Internet reserved by IANA
through		
01-00-5E-FF-FF-FF		
01-00-81-00-00-02		Synoptics Network Management
01-80-C2-00-00-00	-802-	Spanning tree (for bridges)
01-80-C2-00-00-01	-802-	802.1 alternate Spanning multicast
through		
01-80-C2-00-00-0F		
01-80-C2-00-00-14	-802-	OSI Route level 1 (within area) IS hello?
01-80-C2-00-00-15	-802-	OSI Route level 2 (between area) IS hello?
01-DD-00-FF-FF-FF	7002	Ungermann-Bass boot-me requests
01-DD-01-00-00-00	7005	Ungermann-Bass Spanning Tree
03-00-00-00-00-10	80D5	(OS/2 1.3 EE + Communications Manager)
03-00-00-00-00-40	80D5	(OS/2 1.3 EE + Communications Manager)
09-00-02-04-00-01?	8080?	Vitalink printer messages
09-00-02-04-00-02?	8080?	Vitalink bridge management
09-00-07-00-00-00	-802-	AppleTalk Zone multicast addresses
through		
09-00-07-00-00-FC		
09-00-07-FF-FF-FF	-802-	AppleTalk broadcast address
09-00-09-00-00-01	8005	HP Probe
09-00-09-00-00-01	-802-	HP Probe
09-00-09-00-00-04	8005?	HP DTC
09-00-0D-xx-xx-xx	-802-	ICL Oslan Multicast
09-00-0D-02-00-00		ICL Oslan Service discover on boot
09-00-0D-02-0A-38		ICL Oslan Service discover on boot
09-00-0D-02-0A-39		ICL Oslan Service discover on boot
09-00-0D-02-0A-3C		ICL Oslan Service discover on boot

09-00-0D-02-FF-FF		ICL Oslan Service discover on boot
09-00-0D-09-00-00		ICL Oslan Service discover as required
09-00-1E-00-00-00	8019?	Apollo DOMAIN
09-00-26-01-00-01?	8038	Vitalink TransLAN bridge management
09-00-2B-00-00-00	6009?	DEC MUMPS?
09-00-2B-00-00-01	8039	DEC DSM/DTP?
09-00-2B-00-00-02	803B?	DEC VAXELN?
09-00-2B-00-00-03	8038	DEC Lanbridge Traffic Monitor (LTM)
09-00-2B-00-00-04		DEC MAP End System Hello?
09-00-2B-00-00-05		DEC MAP Intermediate System Hello?
09-00-2B-00-00-06	803D?	DEC CSMA/CD Encryption?
09-00-2B-00-00-07	8040?	DEC NetBios Emulator?
09-00-2B-00-00-0F	6004	DEC Local Area Transport (LAT)
9-00-2B-00-00-1x		DEC Experimental
09-00-2B-01-00-00	8038	DEC LanBridge Copy packets
09-00-2B-01-00-01	8038	DEC LanBridge Hello packets (All local bridges) 1 packet per second, sent by the designated LanBridge
09-00-2B-02-00-00		DEC DNA Level 2 Routing Layer ?
09-00-2B-02-01-00	803C?	DEC DNA Naming Service Advertise?
09-00-2B-02-01-01	803C?	DEC DNA Naming Service Solicitation?
09-00-2B-02-01-02	803E?	DEC DNA Time Service
09-00-2B-03-xx-xx		DEC default filtering by bridges?
09-00-2B-04-00-00	8041?	DEC Local Area Sys Transport LAST?
09-00-2B-23-00-00	803A?	DEC Argonaut Console?
09-00-39-00-70-00?		Spider Systems Bridge Hello packet?
09-00-4C-00-00-00	-802-	BICC 802.1 management
09-00-4C-00-00-02	-802-	BICC 802.1 management
09-00-4C-00-00-06	-802-	BICC Local bridge STA 802.1(D) Rev6
09-00-4C-00-00-0C	-802-	BICC Rem bridge STA 802.1(D) Rev8
09-00-4C-00-00-0F	-802-	BICC Remote bridge Adaptive Routing (e.g. to Retix)
09-00-4E-00-00-02?	8137?	Novell IPX (BICC?)
09-00-56-00-00-00 through 09-00-56-FE-FF-FF	Stanford reserved	
09-00-56-FF-00-00 through 09-00-56-FF-FF-FF	805C	Stanford V Kernel, version 6.0
09-00-77-00-00-00	-802-	Retix Bridge Local Management System
09-00-77-00-00-01	-802-	Retix spanning tree bridges
09-00-77-00-00-02	-802-	Retix Bridge Adaptive routing
09-00-7C-01-00-01		Vitalink DLS Multicast 09-00-7C-01-00-03 Vitalink DLS Inlink

09-00-7C-01-00-04		Vitalink DLS and non DLS Multicast
09-00-7C-02-00-05	8080?	Vitalink diagnostics
09-00-7C-05-00-01	8080?	Vitalink gateway?
09-00-7C-05-00-02		Vitalink Network Validation Message
09-00-87-80-FF-FF	0889	Xyplex Terminal Servers
09-00-87-90-FF-FF	0889	Xyplex Terminal Servers
0D-1E-15-BA-DD-06		HP
80-01-43-00-00-00	-802-	Bridge
80-01-43-00-00-08	-802-	Bridge Management
80-01-43-00-00-28	-802-	ISO 10589 level-1 Intermediate Stations
80-01-43-00-00-48	-802-	Loadable Device
80-01-43-00-00-88	-802-	Load Server
80-01-43-00-00-A8	-802-	ISO 10589 level-2 Intermediate Stations
80-01-43-00-80-00	-802-	FDDI RMT Directed Beacon
80-01-43-00-80-08	-802-	FDDI status report frame
90-00-D4-00-00-20	-802-	OSI Network Layer Intermediate Stations
90-00-D4-00-00-A0	-802-	OSI Network Layer End Stations
AB-00-00-01-00-00	6001	DEC Maintenance Operation Protocol (MOP) Dump/Load Assistance
AB-00-00-02-00-00	6002	DEC Maintenance Operation Protocol (MOP) Remote Console 1 System ID packet every 8-10 minutes, by every: DEC DEUNA interface, DEC DELUA interface, and DEC DEQNA interface
AB-00-00-03-00-00	6003	DECNET Phase IV end node Hello packets 1 packet every 15 seconds, sent by each DECNET host
AB-00-00-04-00-00	6003	DECNET Phase IV Router Hello packets, 1 packet every 15 seconds, sent by the DECNET router
AB-00-00-05-00-00 through AB-00-03-FF-FF-FF		Reserved DEC
AB-00-03-00-00-00	6004	DEC Local Area Transport (LAT) - old
AB-00-04-00-xx-xx		Reserved DEC customer private use
AB-00-04-01-xx-yy	6007	DEC Local Area VAX Cluster groups System Communication Architecture
C0-00-00-00-00-01	-802-	Active Monitor
C0-00-00-00-00-02	-802-	Ring Parameter Monitor
C0-00-00-00-00-04	-802-	Network Server Heartbeat
C0-00-00-00-00-08	-802-	Ring Error Monitor
C0-00-00-00-00-10	-802-	Configuration Report Server
C0-00-00-00-00-20	-802-	Synchronous Bandwidth Manager
C0-00-00-00-00-40	-802-	Locate - Directory Server
C0-00-00-00-00-80	-802-	NETBIOS
C0-00-00-00-01-00	-802-	Bridge
C0-00-00-00-02-00	-802-	IMPL Server

C0-00-00-00-04-00	-802-	Ring Authorization Server
C0-00-00-00-08-00	-802-	LAN Gateway
C0-00-00-00-10-00	-802-	Ring Wiring Concentrator
C0-00-00-00-20-00	-802-	LAN Manager
C0-00-00-00-80-00	-802-	user-defined
through		
C0-00-40-00-00-00	-802-	
CF-00-00-00-00-00	9000	Ethernet Configuration Test protocol (Loopback)
FF-FF-00-60-00-04	81D6	Lantastic
FF-FF-00-40-00-01	81D6	Lantastic
FF-FF-01-E0-00-04	81D6	Lantastic

COMMON ETHERNET BROADCAST ADDRESSES

This table contains common uses for the Ethernet Broadcast Address and the Ethernet Protocols that use it. This table is for reference only.

FF-FF-FF-FF-FF-FF	0600	XNS packets, Hello or gateway search? 6 packets every 15 seconds, per XNS station
FF-FF-FF-FF-FF-FF	0800	IP (e.g. RWHOD via UDP) as needed
FF-FF-FF-FF-FF-FF	0804	CHAOS
FF-FF-FF-FF-FF-FF	0806	ARP (for IP and CHAOS) as needed
FF-FF-FF-FF-FF-FF	0BAD	Banyan
FF-FF-FF-FF-FF-FF	1600	VALID packets, Hello or gateway search? 1 packets every 30 seconds, per VALID station
FF-FF-FF-FF-FF-FF	8035	Reverse ARP
FF-FF-FF-FF-FF-FF	807C	Merit Internodal (INP)
FF-FF-FF-FF-FF-FF	809B	EtherTalk Phase I
FF-FF-FF-FF-FF-FF	9001	3Com (ex Bridge) Name Service
FF-FF-FF-FF-FF-FF	9002	3Com PCS/TCP Hello,Approximately 1 per minute per workstation

ASSIGNED IP - TCP/UDP SOCKETS

(from RFC1060)

0		Reserved
1	TCPMUX	TCP Port Service Multiplexer
2-4		Unassigned
5	RJE	Remote Job Entry
7	ECHO	Echo
9	DISCARD	Discard
11	USERS	Active Users

13	DAYTIME	Daytime
15		Unassigned
17	QUOTE	Quote of the Day
19	CHARGEN	Character Generator
20	FTP-DATA	File Transfer [Default Data]
21	FTP	File Transfer [Control]
23	TELNET	Telnet
25	SMTP	Simple Mail Transfer
27	NSW-FE	NSW User System
29	MSG-ICP	MSG ICP
31	MSG-AUTH	MSG Authentication
33	DSP	Display Support Protocol
35		Any private printer server
37	TIME	Time
39	RLP	Resource Location Protocol
41	GRAPHICS	Graphics
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
44	MPM-FLAGS	MPM FLAGS Protocol
45	MPM	Message Processing Module [receive]
46	MPM-SND	MPM [default send]
47	NI-FTP	NI FTP
49	LOGIN	Login Host Protocol
51	LA-MAINT	IMP Logical Address Maintenance
53	DOMAIN	Domain Name Server
55	ISI-GL	SI Graphics Language
57		Any private terminal access
59		Any private file service
61	NI-MAIL	NI MAIL
63	VIA-FTP	VIA Systems - FTP
65	TACACS-DS	TACACS-Database Service
67	BOOTPS	Bootstrap Protocol Server
68	BOOTPC	Bootstrap Protocol Client
69	TFTP	Trivial File Transfer
71	NETRJS-1	Remote Job Service
72	NETRJS-2	Remote Job Service
73	NETRJS-3	Remote Job Service
74	NETRJS-4	Remote Job Service
75		Any private dial out service
77		Any private RJE service
79	FINGER	Finger
81	HOSTS2-NS	HOSTS2 Name Server
83	MIT-ML-DEV	MIT ML Device
85	MIT-ML-DEV	MIT ML Device
87		Any private terminal link
89	SU-MIT-TG	SU/MIT Telnet Gateway

91	MIT-DOV	MIT Dover Spooler
93	DCP	Device Control Protocol
95	SUPDUP	SUPDUP
97	SWIFT-RVF	Swift Remote Vitural File Protocol
98	TACNEWS	TAC News
99	METAGRAM	Metagram Relay
101	HOSTNAME	NIC Host Name Server
102	ISO-TSAP	ISO-TSAP
103	X400	X400
104	X400-SND	X400-SND
105	CSNET-NS	Mailbox Name Nameserver
107	RTELNET	Remote Telnet Service
109	POP2	Post Office Protocol - Version 2
110	POP3	Post Office Protocol - Version 3
111	SUNRPC	SUN Remote Procedure Call
113	AUTH	Authentication Service
115	SFTP	Simple File Transfer Protocol
117	UUCP-PATH	UUCP Path Service
119	NNTP	Network News Transfer Protocol
121	ERPC	Encore Expedited Remote Proc. Call
123	NTP	Network Time Protocol
125	LOCUS-MAP	Locus PC-Interface Net Map Server
127	LOCUS-CON	Locus PC-Interface Conn Server
129	PWDGEN	Password Generator Protocol
130	CISCO-FNA	CISCO FNATIVE
131	CISCO-TNA	CISCO TNATIVE
132	CISCO-SYS	CISCO SYSMAINT
133	STATSRV	Statistics Service
134	INGRES-NET	NGRES-NET Service
135	LOC-SRV	Location Service
136	PROFILE	PROFILE Naming System
137	NETBIOS-NS	NetBIOS Name Service
138	NETBIOS-DGM	NetBIOS Datagram Service
139	NETBIOS-SSN	NetBIOS Session Service
140	EMFIS-DATA	EMFIS Data Service
141	EMFIS-CNTL	EMFIS Control Service
142	BL-IDM	Britton-Lee IDM
143	IMAP2	Interim Mail Access Protocol v2
144	NEWS	NewS
145	UAAC	UAAC Protocol
146	ISO-TP0	ISO-IP0
147	ISO-IP	ISO-IP
148	CRONUS	CRONUS-SUPPORT
149	AED-512	AED 512 Emulation Service
150	SQL-NET	SQL-NET
151	HEMS	HEMS

152	BFTP	Background File Transfer Program
153	SGMP	SGMP
154	NETSC-PROD	NETSC
155	NETSC-DEV	NETSC
156	SQLSRV	SQL Service
157	KNET-CMP	KNET/VM Command/Message
158	PCMail-SRV	PCMail Server
159	NSS-Routing	NSS-Routing
160	SGMP-TRAPS	SGMP-TRAPS
161	SNMP	SNMP
162	SNMPTRAP	SNMPTRAP
163	CMIP-Manage	CMIP/TCP Manager
164	CMIP-Agent	CMIP/TCP Agent
165	XNS-Courier	Xerox
166	S-Net	Sirius Systems
167	NAMP	NAMP
168	RSVD	SVD
169	SEND	SEND
170	Print-SRV	Network PostScript
171	Multiplx	Network Innovations Multiplex
172	CL/1	Network Innovations CL/1
173	Xyplex-MUX	Xyplex
174	MAILQ	MAILQ
175	VMNET	VMNET
176	GENRAD-MUX	GENRAD-MUX
177	XDMCP	X Display Manager Control Protocol
178	NextStep	NextStep Window Server
179	BGP	Border Gateway Protocol
180	RIS	ntergraph
181	Unify	Unify
182	Unisys-Cam	Unisys-Cam
183	OCBinder	OCBinder
184	OCServer	OCServer
185	Remote-KIS	Remote-KIS
186	KIS	KIS Protocol
187	ACI	Application Communication Interface
188	MUMPS	MUMPS
189	QFT	Queued File Transport
190	GACP	Gateway Access Control Protocol
191	Prospero	Prospero
192	OSU-NMS	OSU Network Monitoring System
193	SRMP	Spider Remote Monitoring Protocol
194	IRC	Internet Relay Chat Protocol
195	DN6-NLM-AUD	DNSIX Network Level Module Audit
196	DN6-SMM-RED	DNSIX Session Mgt Module Audit
197	DLS	Directory Location Service

198	DLS-Mon	Directory Location Service Monitor
199		Unassigned
200		Unassigned
201	AT-RMTP	AppleTalk Routing Maintenance
202	AT-NBP	AppleTalk Name Binding
203	AT-3	AppleTalk Unused
204	AT-ECHO	AppleTalk Echo
205	AT-5	AppleTalk Unused
206	AT-ZIS	AppleTalk Zone Information
207	AT-7	AppleTalk Unused
208	AT-8	AppleTalk Unused
209-223		Unassigned
224-241		Reserved
243	SUR-MEAS	Survey Measurement
245	LINK	LINK
246	DSP3270	Display Systems Protocol
247-255		Reserved

COMMON UNIX (TCP/IP SERVER) SOCKETS

(from RFC1060)

By convention, ports in the range 256 to 1024 are used for "Unix Standard" services. Listed here are some of the normal uses of these port numbers.

echo	7/tcp	
discard	9/tcp	sink null
systat	11/tcp	users
daytime	13/tcp	
netstat	15/tcp	
qotd	17/tcp	quote
chargen	19/tcp	ttytst source
ftp-data	20/tcp	
ftp	21/tcp	
telnet	23/tcp	
smtp	25/tcp	mail
time	37/tcp	timserver
name	42/tcp	nameserver
whois	43/tcp	nicname
nameserver	53/tcp	domain
apts	57/tcp	any private terminal service
apfs	59/tcp	any private file service
rje	77/tcp	netrjs
finger	79/tcp	
link	87/tcp	ttylink
supdu	95/tcp	

newacct	100/tcp	[unauthorized use]
hostnames	101/tcp	hostname
iso-tsap	102/tcp	tsap
x400	103/tcp	
x400-snd	104/tcp	
csnet-ns	105/tcp	CSNET Name Service
pop-2	109/tcp	
pop		postoffice
sunrpc	111/tcp	
auth	113/tcp	authentication
sftp	115/tcp	
uucp-path	117/tcp	
nntp	119/tcp	usenet readnews untp
ntp	123/tcp	network time protocol
statsrv	133/tcp	
profile	136/tcp	
NeWS	144/tcp	news
print-srv	170/tcp	
exec	512/tcp	remote process execution;
login	513/tcp	remote login a la telnet;
cmd	514/tcp	like exec, but automatic
printer	515/tcp	spooler
efs	520/tcp	extended file name server
tempo	526/tcp	newdate
courier	530/tcp	rpc
conference	531/tcp	chat
netnews	532/tcp	readnews
uucp	540/tcp	uucpd
klogin	543/tcp	
kshell	544/tcp	krcmd
dsf	555/tcp	
remotefs	556/tcp	rfs server
chshell	562/tcp	chcmd
meter	570/tcp	daemon
pcserver	600/tcp	Sun IPC server
nqs	607/tcp	nqs
mdqs	666/tcp	
rfile	750/tcp	
pump	751/tcp	
qrh	752/tcp	
rrh	753/tcp	
tell	754/tcp	send
nlogin	758/tcp	
con	759/tcp	
ns	760/tcp	
rxe	761/tcp	

quotad	762/tcp	
cycleserv	763/tcp	
omserv	764/tcp	
webster	765/tcp	
phonebook	767/tcp	phone
vid	769/tcp	
rtip	771/tcp	
cycleserv2	772/tcp	
submit	773/tcp	
rpasswd	774/tcp	
entomb	775/tcp	
wpages	776/tcp	
wpgs	780/tcp	
mdbs_daemon	800/tcp	
device	801/tcp	
maitrd	997/tcp	
busboy	998/tcp	
garcon	999/tcp	

COMMON UNIX SOCKETS

(from RFC1060)

(UDP/IP "Server" Sockets)

echo	7/udp	
discard	9/udp	sink null
systat	11/udp	users
daytime	13/udp	
netstat	15/udp	
qotd	17/udp	quote
chargen	19/udp	ttytst source
time	37/udp	timserver
rlp	39/udp	resource
name	42/udp	nameserver
whois	43/udp	nicname
nameserver	53/udp	domain
bootps	67/udp	bootp
bootpc	68/udp	
tftp	69/udp	
sunrpc	111/udp	
erpc	121/udp	
ntp	123/udp	
statsrv	133/udp	
profile	36/udp	
snmp	161/udp	

snmp-trap	162/udp	
at-rtmp	201/udp	
at-nbp	202/udp	
at-3	203/udp	
at-echo	204/udp	
at-5	205/udp	
at-zis	206/udp	
at-7	207/udp	
at-8	208/ud	
biff	512/udp	used by mail system to notify users of new mail received; currently receives messages only from processes on the same machine.
who	513/udp	maintains data bases showing who's logged in to machines on a local net and the load average of the machine.
syslog	514/udp	
talk	517/udp	like tenex link, but across machine - unfortunately,doesn'tuse link protocol (this is actually just a rendezvous portfrom which a tcp connection is established)
ntalk	518/udp	
utime	519/udp	unixtime
router	520/udp	local routing process (RIP);
timed	525/udp	timeserver
netwall	533/udp	for emergency broadcasts
new-rwho	550/udp	new-who
rmonitor	560/udp	rmonitord
monitor	561/udp	
meter	571/udp	udaemon
elcsd	704/udp	errlog copy/server daemon
loadav	750/udp	
vid	769/udp	
cadlock	770/udp	
notify	773/udp	
acmaint_dbd	774/udp	
acmaint_transd	775/udp	
wpages	776/udp	
puparp	998/udp	
applix	999/udp	Applix ac
puprouter	999/udp	
cadlock	1000/udp	

COMMON UNIX SOCKETS (TCP/IP Sockets > 1023)
(from RFC1060)

blackjack	1025/tcp	network blackjack
bbn-mmc	1347/tcp	multi media conferencing
bbn-mmx	1348/tcp	multi media conferencing
orasrv	1525/tcp	oracle
ingreslock	1524/tcp	
issd	1600/tcp	
nkd	1650/tcp	
dc	2001/tcp	
mailbox	2004/tcp	
berknet	2005/tcp	
invokator	2006/tcp	
dectalk	2007/tcp	
conf	2008/tcp	
news	2009/tcp	
search	2010/tcp	
raid-cc	2011/tcp	raid
ttyinfo	2012/tcp	
raid-am	2013/tcp	
troff	2014/tcp	
cypress	2015/tcp	
cypress-stat	2017/tcp	
terminaldb	2018/tcp	
whosockami	2019/tcp	
servexec	2021/tcp	
down	2022/tcp	
ellpack	2025/tcp	
shadowserver	2027/tcp	
submitserver	2028/tcp	
device2	2030/tcp	
blackboard	2032/tcp	
glogger	2033/tcp	
scoremgr	2034/tcp	
imsldoc	2035/tcp	
objectmanager	2038/tcp	
lam	2040/tcp	
interbase	2041/tcp	
isis	2042/tcp	
rimsl	2044/tcp	
dls	2047/tcp	
dls-monitor	2048/tcp	
shilp	2049/tcp	
NSWS	3049/tcp	
rfa	4672/tcp	remote file access server

commplex-main	5000/tcp
commplex-link	5001/tcp
padl2sim	5236/tcp
man	9535/tcp

COMMON UNIX SOCKETS (UDP SOCKETS > 1023)

(from RFC1060)

hermes	1248/udp	
wizard	2001/udp	curry
globe	2002/udp	
emce	2004/udp	CCWS mm conf
oracle	2005/udp	
raid-cc	2006/udp	raid
raid-am	2007/udp	
terminaldb	2008/udp	whosockami
news	2009/udp	
pipe_server	2010/udp	
servserv	2011/udp	
raid-ac	2012/udp	
raid-cd	2013/udp	
raid-sf	2014/udp	
raid-cs	2015/udp	
bootserver	2016/udp	
bootclient	2017/udp	
rellpack	2018/udp	
about	2019/udp	
xinupageserver	2020/udp	
xinuexpansion1	2021/udp	
xinuexpansion 2	2022/udp	
xinuexpansion 3	2023/udp	
xinuexpansion 4	2024/udp	
xribs	2025/udp	
scrabble	2026/udp	
isis	2042/udp	
isis-bcast	2043/udp	
rimsl	2044/udp	
cdfunc	2045/udp	
sdfunc	2046/udp	
dls	2047/udp	
shilp	2049/udp	
rmonitor_secure	5145/udp	
xdsxdm	6558/udp	
isode-dua	17007/udp	

KARLBRIDGE & KARLBROUTER SNMP OBJECTS

The KarlBridge and KarlBrouter supports several standard SNMP MIB's. It supports MIB-II, Ethernet-like Interface MIB, Bridge MIB, SNMP MIB and the WaveLAN MIB. The following table documents exactly which MIB variables are used by the KarlBridge and KarlBrouter.

The following key is used throughout these tables:

I	Implemented as described in related RFCs.
N	Not Implemented (to be implemented).
R	Implemented as read-only. (Applies to objects described as read-write in RFC) These objects can only be changed by use of the KBCONFIG program.
Z	The object implemented such that it always reads zero.

THE SYSTEM GROUP

(RFC 1213)

sysDescr	1.3.6.1.2.1.1.1.0	I
sysObjectID	1.3.6.1.2.1.1.2.0	I
sysUpTime	1.3.6.1.2.1.1.3.0	I
sysContact	1.3.6.1.2.1.1.4.0	R
sysName	1.3.6.1.2.1.1.5.0	R
sysLocation	1.3.6.1.2.1.1.6.0	R
sysServices	1.3.6.1.2.1.1.7.0	I

THE INTERFACES GROUP

(RFC 1213)

ifNumber	1.3.6.1.2.1.2.1.0	I

THE INTERFACE TABLE

ifIndex	1.3.6.1.2.1.2.2.1.1.ifIndex	I
ifDescr	1.3.6.1.2.1.2.2.1.2.ifIndex	I
ifType	1.3.6.1.2.1.2.2.1.3.ifIndex	I
ifMtu	1.3.6.1.2.1.2.2.1.4.ifIndex	I
ifSpeed	1.3.6.1.2.1.2.2.1.5.ifIndex	I
ifPhysAddress	1.3.6.1.2.1.2.2.1.6.ifIndex	I
ifAdminStatus	1.3.6.1.2.1.2.2.1.7.ifIndex	R
ifOperStatus	1.3.6.1.2.1.2.2.1.8.ifIndex	I
ifLastChange	1.3.6.1.2.1.2.2.1.9.ifIndex	I
ifInOctets	1.3.6.1.2.1.2.2.1.10.ifIndex	I
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11.ifIndex	I

ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12.ifIndex	I
ifInDiscards	1.3.6.1.2.1.2.2.1.13.ifIndex	I
ifInErrors	1.3.6.1.2.1.2.2.1.14.ifIndex	I
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15.ifIndex	Z
ifOutOctets	1.3.6.1.2.1.2.2.1.16.ifIndex	I
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17.ifIndex	I
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18.ifIndex	I
ifOutDiscards	1.3.6.1.2.1.2.2.1.19.ifIndex	Z
ifOutErrors	1.3.6.1.2.1.2.2.1.20.ifIndex	I
ifOutQLen	1.3.6.1.2.1.2.2.1.21.ifIndex	Z
ifSpecific	1.3.6.1.2.1.2.2.1.22.ifIndex	I

THE IP GROUP
(RFC 1213)

ipForwarding	1.3.6.1.2.1.4.1.0	I
ipDefaultTTL	1.3.6.1.2.1.4.2.0	R
ipInReceives	1.3.6.1.2.1.4.3.0	I
ipInHdrErrors	1.3.6.1.2.1.4.4.0	I
ipInAddrErrors	1.3.6.1.2.1.4.5.0	I
ipForwDatagrams	1.3.6.1.2.1.4.6.0	I
ipInUnknownProtos	1.3.6.1.2.1.4.7.0	I
ipInDiscards	1.3.6.1.2.1.4.8.0	I
ipInDelivers	1.3.6.1.2.1.4.9.0	I
ipOutRequests	1.3.6.1.2.1.4.10.0	I
ipOutDiscards	1.3.6.1.2.1.4.11.0	I
ipOutNoRoutes	1.3.6.1.2.1.4.12.0	I
ipReasmTimeout	1.3.6.1.2.1.4.13.0	I
ipReasmReqds	1.3.6.1.2.1.4.14.0	I
ipReasmOKs	1.3.6.1.2.1.4.15.0	I
ipReasmFails	1.3.6.1.2.1.4.16.0	I
ipFragOKs	1.3.6.1.2.1.4.17.0	I
ipFragFails	1.3.6.1.2.1.4.18.0	I
ipFragCreates	1.3.6.1.2.1.4.19.0	I

THE ICMP GROUP
(RFC 1213)

icmpInMsgs	1.3.6.1.2.1.5.1.0	I
icmpInErrors	1.3.6.1.2.1.5.2.0	I
icmpInDestUnreachs	1.3.6.1.2.1.5.3.0	I
icmpInTimeExcds	1.3.6.1.2.1.5.4.0	I
icmpInParmProbs	1.3.6.1.2.1.5.5.0	I
icmpInSrcQuenchs	1.3.6.1.2.1.5.6.0	I
icmpInRedirects	1.3.6.1.2.1.5.7.0	I
icmpInEchos	1.3.6.1.2.1.5.8.0	I

icmpInEchoReps	1.3.6.1.2.1.5.9.0	I
icmpInTimestamps	1.3.6.1.2.1.5.10.0	I
icmpInTimestampReps	1.3.6.1.2.1.5.11.0	I
icmpInAddrMasks	1.3.6.1.2.1.5.12.0	I
icmpInAddrMaskReps	1.3.6.1.2.1.5.13.0	I
icmpOutMsgs	1.3.6.1.2.1.5.14.0	I
icmpOutErrors	1.3.6.1.2.1.5.15.0	I
icmpOutDestUnreachs	1.3.6.1.2.1.5.16.0	I
icmpOutTimeExcds	1.3.6.1.2.1.5.17.0	I
icmpOutParmProbs	1.3.6.1.2.1.5.18.0	I
icmpOutSrcQuenchs	1.3.6.1.2.1.5.19.0	I
icmpOutRedirects	1.3.6.1.2.1.5.20.0	I
icmpOutEchos	1.3.6.1.2.1.5.21.0	I
icmpOutEchoReps	1.3.6.1.2.1.5.22.0	I
icmpOutTimestamps	1.3.6.1.2.1.5.23.0	I
icmpOutTimestampReps	1.3.6.1.2.1.5.24.0	I
icmpOutAddrMasks	1.3.6.1.2.1.5.25.0	I
icmpOutAddrMaskReps	1.3.6.1.2.1.5.26.0	I

THE UDP GROUP
(RFC 1213)

udpInDatagrams	1.3.6.1.2.1.7.1.0	I
udpNoPorts	1.3.6.1.2.1.7.2.0	I
udpInErrors	1.3.6.1.2.1.7.3.0	I
udpOutDatagrams	1.3.6.1.2.1.7.4.0	I

THE UDP TABLE

udpLocalAddress	1.3.6.1.2.1.7.5.1.1.1.IPAdd.UDPPort	I
udpLocalPort	1.3.6.1.2.1.7.5.1.2.1.IPAdd.UDPPort	I

THE TRANSMISSION GROUP (Ethernet/WaveLAN)
(Ethernet Interface MIB RFC 1398)

Note: dot3Index = ifIndex

dot3Index	1.3.6.1.2.1.10.7.1.1.1.dot3Index	I
dot3InitializeMac	1.3.6.1.2.1.10.7.1.1.2.dot3Index	R
dot3MacSubLayerStatus	1.3.6.1.2.1.10.7.1.1.3.dot3index	R
dot3MulticastReceiveStatus	1.3.6.1.2.1.10.7.1.1.4.dot3Index	R
dot3TxEnabled	1.3.6.1.2.1.10.7.1.1.5.dot3Index	I
dot3TestTdrValue	1.3.6.1.2.1.10.7.1.1.6.dot3Index	Z

THE TRANSMISSION GROUP STATUS TABLE (Ethernet/WaveLAN)

Note: dot3StatsIndex = ifIndex

dot3StatusIndex	1.3.6.1.2.1.10.7.2.1.1.dot3StatsIndex	I
dot3StatsAlignmentErrors	1.3.6.1.2.1.10.7.2.1.2.dot3StatsIndex	I
dot3StatsFCSErrors	1.3.6.1.2.1.10.7.2.1.3.dot3StatsIndex	I
dot3StatsSingleCollisionFrames	1.3.6.1.2.1.10.7.2.1.4.dot3StatsIndex	I
dot3StatsMultipleCollisionFrames	1.3.6.1.2.1.10.7.2.1.5.dot3StatsIndex	I
dot3StatsSQETestErrors	1.3.6.1.2.1.10.7.2.1.6.dot3StatsIndex	Z
dot3StatsDeferredTransmissions	1.3.6.1.2.1.10.7.2.1.7.dot3StatsIndex	I
dot3StatsLateCollisions	1.3.6.1.2.1.10.7.2.1.8.dot3StatsIndex	I
dot3StatsExcessiveCollisions	1.3.6.1.2.1.10.7.2.1.9.dot3StatsIndex	I
dot3StatsInternalMacTransmitError	1.3.6.1.2.1.10.7.2.1.10.dot3StatsIndex	I
dot3StatsCarrierSenseErrors	1.3.6.1.2.1.10.7.2.1.11.dot3StatsIndex	I
dot3StatsFrameTooLongs	1.3.6.1.2.1.10.7.2.1.13.dot3StatsIndex	I
dot3StatsInternalMacReceiveErr	1.3.6.1.2.1.10.7.2.1.16.dot3StatsIndex	I

THE SNMP GROUP

(RFC 1213)

snmpInPkts	1.3.6.1.2.1.11.1.0	I
snmpOutPkts	1.3.6.1.2.1.11.2.0	I
snmpInBadVersions	1.3.6.1.2.1.11.3.0	I
snmpInBadCommunityNames	1.3.6.1.2.1.11.4.0	I
snmpInBadCommunityUses	1.3.6.1.2.1.11.5.0	I
snmpInASNParseErrs	1.3.6.1.2.1.11.6.0	I
snmpInTooBigs	1.3.6.1.2.1.11.8.0	I
snmpInNoSuchNames	1.3.6.1.2.1.11.9.0	I
snmpInBadValues	1.3.6.1.2.1.11.10.0	I
snmpInReadOnlys	1.3.6.1.2.1.11.11.0	I
snmpInGenErrs	1.3.6.1.2.1.11.12.0	I
snmpInTotalReqVars	1.3.6.1.2.1.11.13.0	I
snmpInTotalSetVars	1.3.6.1.2.1.11.14.0	I
snmpInGetRequests	1.3.6.1.2.1.11.15.0	I
snmpInGetNexts	1.3.6.1.2.1.11.16.0	I
snmpInSetRequests	1.3.6.1.2.1.11.17.0	I
snmpInGetResponses	1.3.6.1.2.1.11.18.0	I
snmpInTraps	1.3.6.1.2.1.11.19.0	I
snmpOutTooBigs	1.3.6.1.2.1.11.20.0	I
snmpOutNoSuchNames	1.3.6.1.2.1.11.21.0	I
snmpOutBadValues	1.3.6.1.2.1.11.22.0	I
snmpOutGenErrs	1.3.6.1.2.1.11.24.0	I
snmpOutGetRequests	1.3.6.1.2.1.11.25.0	I
snmpOutGetNexts	1.3.6.1.2.1.11.26.0	I
snmpOutSetRequests	1.3.6.1.2.1.11.27.0	I

snmpOutGetResponses	1.3.6.1.2.1.11.28.0	I
snmpOutTraps	1.3.6.1.2.1.11.29.0	I
snmpEnableAuthTraps	1.3.6.1.2.1.11.30.0	R

THE BRIDGE GROUP

(RFC 1286)

dot1dBaseBridgeAddress	1.3.6.1.2.1.17.1.1.0	I
dot1dBaseNumPorts	1.3.6.1.2.1.17.1.2.0	I
dot1dBaseType	1.3.6.1.2.1.17.1.3.0	I

THE dot1bBasePORT TABLE

Note: dot1dBasePort = dot1dBasePort

dot1dBasePort	1.3.6.1.2.1.17.1.4.1.1.dot1dBasePort	I
dot1dBasePortIfIndex	1.3.6.1.2.1.17.1.4.1.2.dot1dBasePor	I
dot1dBasePortCircuit	1.3.6.1.2.1.17.1.4.1.3.dot1dBasePort	I
dot1dBasePortDelayExceededDisc	1.3.6.1.2.1.17.1.4.1.4.dot1dBasePort	I
dot1dBasePortMtuExceededDiscards	1.3.6.1.2.1.17.1.4.1.5.dot1dBasePort	I

THE dot1dTp GROUP

dot1dTpLearnedEntryDiscards	1.3.6.1.2.1.17.4.1.0	I
dot1dTpAgingTime	1.3.6.1.2.1.17.4.2.0	R

THE dot1dTpFdb TABLE

dot1dTpFdbAddress	1.3.6.1.2.1.17.4.3.1.1.dot1dTpFdbAddress	I
dot1dTpFdbPort	1.3.6.1.2.1.17.4.3.1.2.dot1dTpFdbAddress	I
dot1dTpFdbStatus	1.3.6.1.2.1.17.4.3.1.3.dot1dTfFdbAddress	I

THE dot 1dTp PORT TABLE

Note: dot1dTpPort = ifIndex

dot1dTpPort	1.3.6.1.2.1.17.4.4.1.1.dot1dTpPort	I
dot1dTpPortMaxInfo	1.3.6.1.2.1.17.4.4.1.2.dot1dTpPort	I
dot1dTpPortInFrames	1.3.6.1.2.1.17.4.4.1.3.dot1dTpPort	I
dot1dTpPortOutFrames	1.3.6.1.2.1.17.4.4.1.4.dot1dTpPort	I
dot1dTpPortInDiscards	1.3.6.1.2.1.17.4.4.1.5.dot1dTpPort	I

THE dot 1d STATIC TABLE

Note 1: dot1dStaticReceivePort = ifIndex
Note 2: index = dot1dStaticAddress.dot1dStaticReceivePort

dot1dStaticAddress	1.3.6.1.2.1.17.5.1.1.index	N
dot1dStaticReceivePort	1.3.6.1.2.1.17.5.1.2.index	N
dot1dStaticAllowedToGoTo	1.3.6.1.2.1.17.5.1.3.index	N
dot1dStaticStatus	1.3.6.1.2.1.17.5.1.4.index	N

THE NCR WaveLAN GROUP

(unpublished NCR information)

WaveLAN INTERFACE NIC INFORMATION

wliNicIndex 1.3.6.1.4.1.74.2.21.1.1.1.1.wliNicIndex
An index value that uniquely identifies a WaveLAN network interface this NIC information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of ifIndex.

wliNicBusType 1.3.6.1.4.1.74.2.21.1.1.1.2.wliNicIndex
The bus-type supported by this NIC. One of the following: xtBus(1), isaBus(2), mcBus(3), pcmcia2Bus(4), wavepointBus(5)

wliNicSlot 1.3.6.1.4.1.74.2.21.1.1.1.3.wliNicIndex
The I/O Base Address (ISA/AT) or Slot Number (MC) or Socket Number (PCMCIA) used by this NIC. For ISA/AT (and alike) Base Addresses, the following values are used: 1='0300'H, 2='0390'H, 3='03C0'H, 4='03E0'H.

wliNicIrq 1.3.6.1.4.1.74.2.21.1.1.1.4.wliNicIndex
The Interrupt Request Number (IRQ) used by this NIC.

wliNicError 1.3.6.1.4.1.74.2.21.1.1.1.5.wliNicIndex
A counter for miscellaneous board errors. It indicates (intermittent) NIC hardware problems.

wliNicOutOfRxResource 1.3.6.1.4.1.74.2.21.1.1.1.6.wliNicIndex

A counter for the number of times the NIC is out of resources for the receiver, causing the receiver to be switched off temporarily.

WaveLAN INTERFACE PHY INFORMATION

wliPhyIndex
1.3.6.1.4.1.74.2.21.1.2.1.1.wliPhyIndex
An index value that uniquely identifies a WaveLAN network interface this PHY information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of ifIndex.

wliPhyDsp
1.3.6.1.4.1.74.2.21.1.2.1.2.wliPhyIndex
The Digital Signal Processor on the board." The following are valid values: icarus(1), daedalus(2)

wliPhyFrequency
1.3.6.1.4.1.74.2.21.1.2.1.3.wliPhyIndex
The mid-point of the frequency band this WaveLAN NIC operates in. The following values are valid: 915Mhz(1), 2425Mhz(2), 2460Mhz(3), 2484Mhz(4), 2430Mhz(5) -- actually 2430.5 MHz.

wliPhyNwid
1.3.6.1.4.1.74.2.21.1.2.1.4.wliPhyIndex
The WaveLAN Network ID (NWID) this RF-modem is currently configured for.

wliPhyRfSilenceLevel
1.3.6.1.4.1.74.2.21.1.2.1.5.wliPhyIndex
The RF Silence Level as currently read from the RF modem.

wliPhyOwnNwid
1.3.6.1.4.1.74.2.21.1.2.1.6.wliPhyIndex
Own NWID counter; the number of frames received with theconfigured NWID.

wliPhyOtherNwid
1.3.6.1.4.1.74.2.21.1.2.1.7.wliPhyIndex
Other NWID counter; the number of frames received with different NWID than configured.

wliPhyLowSnr
1.3.6.1.4.1.74.2.21.1.2.1.8.wliPhyIndex
The count of the number of KarlBridge test frames received with a Low signal to noise ratio. (ATT/NCR does not have support for this object)

wliPhyGoodSnr
1.3.6.1.4.1.74.2.21.1.2.1.9.wliPhyIndex

The count of the number of KarlBridge test frames received with a Good signal to noise ratio. (ATT/NCR does not have support for this object)

wliPhyExcellentSnr
1.3.6.1.4.1.74.2.21.1.2.1.10.wliPhyIndex
The count of the number of KarlBridge test frames received with a Excellent signal to noise ratio. (ATT/NCR does not have support for this object)

WaveLAN INTERFACE MAC INFORMATION

MAC status information and control variables for a collection of WaveLAN interfaces attached to a particular system.

wliMacIndex
1.3.6.1.4.1.74.2.21.1.3.1.1.wliMacIndex
An index value that uniquely identifies a WaveLAN network interface this MAC information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of ifIndex.

wliMacAddressSelect
1.3.6.1.4.1.74.2.21.1.3.1.2.wliMacIndex
"MAC Address type select." As follows: universal(1), local(2)

wliMacCaDefers
1.3.6.1.4.1.74.2.21.1.3.1.3.wliMacIndex
CSMA/CA Defer counter."

wliMacDeferredTransmissions
1.3.6.1.4.1.74.2.21.1.3.1.4.wliMacIndex
A counter for the number of frames for which the transmission attempt is delayed because the medium is busy. (same as dot3StatsDeferredTransmissions)"

wliMacFCSErrors
1.3.6.1.4.1.74.2.21.1.3.1.5.wliMacIndex
A counter for the number of frames received that do not pass the FCS check and/or that are not an integral number of octets in length. WaveLAN hardware does not distinguish between FCS errors and Alignment errors. (same as dot3StatsFCSErrors + dot3StatsAlignmentErrors)"

wliMacFrameTooLongs
1.3.6.1.4.1.74.2.21.1.3.1.6.wliMacIndex
A counter for the number of frames received that exceed themaximum permitted frame size for the medium (1518 bytes). (same as dot3StatsFrameTooLongs)"

wliMacFrameTooShorts
1.3.6.1.4.1.74.2.21.1.3.1.7.wliMacIndex
A counter for the number of frames received that are shorter than the minimum permitted frame size for the medium (64 bytes)"

WaveLAN INTERFACE DRIVER INFORMATION

Driver information for a collection of WaveLAN interfaces attached to a particular system.

wliDriverIndex
1.3.6.1.4.1.74.2.21.1.4.1.1.wliDriverIndex
An index value that uniquely identifies a WaveLAN network interface this Driver information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of ifIndex.

wliDriverName
1.3.6.1.4.1.74.2.21.1.4.1.2.wliDriverIndex
The name of the software driver for this WaveLAN network interface.

wliDriverVersion
1.3.6.1.4.1.74.2.21.1.4.1.3.wliDriverIndex
The version number of the software driver. A text string as 'mm.nn.pp', where mm = major release number; nn = point release number; pp = optional patch number.

WaveLAN INTERFACE ENCRYPTION INFORMATION

Encryption status information and control variables for a collection of WaveLAN interfaces attached to a particular system.

wliEncIndex
1.3.6.1.4.1.74.2.21.1.5.1.1.ifIndex
An index value that uniquely identifies a WaveLAN network interface that this encryption information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of IfIndex.

wliEncInstalled
1.3.6.1.4.1.74.2.21.1.5.1.2.wliEncIndex
Which encryption option is installed as follows: none(1), des(2), aes(3)

wliEncSelect
1.3.6.1.4.1.74.2.21.1.5.1.3.wliEncIndex
Whether encryption is enabled or disabled as follows: disabled(1), enabled(2)

wliEncKey 1.3.6.1.4.1.74.2.21.1.5.1.4.wliEncIndex

The encryption key. (This variable is not implemented in the KarlBridge it will always return a value of 0).

WaveLAN INTERFACE MULTICAST DELAY GROUP

Information about the Multicast Delay feature for a collection of WaveLAN interfaces attached to a particular system. Implementation of this group is optional.

wliMcastDelayIndex 1.3.6.1.4.1.74.2.21.1.6.1.1.ifIndex

An index value that uniquely identifies a WaveLAN network interface this Multicast Delay information applies to. The interface associated with a particular value of this index is the same interface as identified by the same value of wliIndex (and ifIndex).

wliMcastNumberOfAps 1.3.6.1.4.1.74.2.21.1.6.1.1.ifMcastDelayIndex

The total number of Access Points in the coverage area. Together with wliMcastApSequenceNumber this is used to determine the delays before and after the transmission of each multicast frame. This results in a transmission slot per Access Point per multicast frame. 0 means: no multicast delay specified (use default mechanism).

wliMcastApSequenceNumber 1.3.6.1.4.1.74.2.21.1.6.1.2.ifMcastDelayIndex

The sequence number of this Access Point in the coverage area. Together with wliMcastNumberOfAps this is used to determine the delays before and after the transmission of each multicast frame. This results in a transmission slot per Access Point per multicast frame."

wliMcastRepeatCount 1.3.6.1.4.1.74.2.21.1.6.1.3.ifMcastDelayIndex

The number of times a multicast frame transmission is to be repeated.

THE KARLNET EXTENDED WaveLAN GROUP

(Valid only if the Bridge is in CellWave Mode)

kbWaveLANStationNumber 1.3.6.1.4.1.762.2.5.1.0

The Number of Valid WaveLAN (CellWave) remote stations.

kbWaveLanStationIndex 1.3.6.1.4.1.762.2.5.2.1.index Integer

The Index for this receive station entry

kbWaveLanPortIndex	1.3.6.1.4.1.762.2.5.2.2.index	Integer	
kbWaveLanStationName	1.3.6.1.4.1.762.2.5.2.3.index	Integer	The name of the remote CellWave Bridge for this index.
kbWaveLanExclHellos	1.3.6.1.4.1.762.2.5.2.4.index	Counter	Number of times a Hello packet was received from this station with Excelenet Signal to Noiseratio.
kbWaveLanGoodHellos	1.3.6.1.4.1.762.2.5.2.5.index	Counter	Number of times a Hello packet was received rom this station with Good Signal to Noise ratio.
kbWaveLanLowHellos	1.3.6.1.4.1.762.2.5.2.6.index	Counter	Number of times a Hello packet was received from this station with Low Signal to Noise ratio.
kbWaveLanSignalLevel	1.3.6.1.4.1.762.2.5.2.7.index	Integer	Current Signal Level from 0 to a nominal 100%; you could see values slightly above 100%.
kbWaveLanSilenceLevel	1.3.6.1.4.1.762.2.5.2.8.index	Integer	Current Silence (Noise) Level from 0 to 100%.
kbWaveLanSignalQuality	1.3.6.1.4.1.762.2.5.2.9.index	Integer	Current Signal Quality from 0 to 100% the signal quality will be low if there are any multipath problems, reflection etc.
kbWaveLanThresholdLevel	1.3.6.1.4.1.762.2.5.2.10.index	Integer	Not yet implemented; always read as 0.
kbWaveLanInSeqNumber	1.3.6.1.4.1.762.2.5.2.11.index	Integer	Not yet implemented; always read as 0.
kbWaveLanTransmits	1.3.6.1.4.1.762.2.5.2.12.index	Counter	The number of times a packet that needs to be delivered reliably has been offered for transmit.
kbWaveLanBadTransmits	1.3.6.1.4.1.762.2.5.2.13.index	Counter	Number of times a packet that needs to be delivered reliably has not been transmitted after 16 retransmit attempts.

kbWaveLanRetransmits 1.3.6.1.4.1.762.2.5.2.14.index Counter
Number of times a packet that needs to be delivered reliably has been retransmitted.

kbWaveLanTimeOut 1.3.6.1.4.1.762.2.5.2.15.index Integer
Not implemented always read as 0.

kbWaveLanType 1.3.6.1.4.1.762.2.5.2.16.index Integer
How remote station is configured.
1 = Remote station is in Compatability Mode
2 = Remote station is in CellWave No Base Station Mode
3 = Remote station is in CellWave Mode and is a Base Station
4 = Remote station is in CellWave Mode and is a Satalite Station

kbWaveLanSNR
1.3.6.1.4.1.762.2.5.2.17.index Integer
Current Signal to Noise Ratio Value.
1 = Unknown
2 = Low Signal to Noise Ratio
3 = Good Signal to Noise Ratio
4 = Excelent Signal to Noise Ratio

kbWaveLanState
1.3.6.1.4.1.762.2.5.2.18.index Integer
1 = On line
2 = Off line