NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

A ROUTING-BASED SOLUTION TO SIMULATING AN ACCESS POINT

by

Wayne E. Collins

September 1999

Thesis Advisor: Dennis Volpano

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1999	Master's Thesis

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
A Routing-Based Solution to Simulating an Access Point	
6. AUTHOR(S) Wayne E. Collins	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited.	

13. ABSTRACT *(maximum 200 words)*

Military environments require flexible network configurations that must adapt under dynamic and mobile operating conditions. Wireless data networks offer some solutions. A wireless network typically requires an access point to bridge network traffic between wireless and wired media. These devices though are often too inflexible for use in such dynamic conditions. One major problem is that they are dedicated to a single type of wired network, usually Ethernet, which prevents them from being used to bridge traffic to other kinds of networks, for example, ATM or even cellular.

This thesis shows how any device running a recent Linux kernel can be configured to route packets in way that simulates an access point. The advantages of such a configuration are described and its potential for military use is discussed.

14. SUBJECT TERMS wireless, networking, access points, bridges			15. NUMBER OF PAGES 68
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited.

A ROUTING-BASED SOLUTION TO SIMULATING AN ACCESS POINT

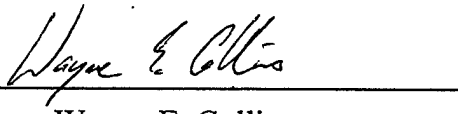
Wayne E. Collins
Captain, United States Marine Corps
B.S., San Diego State University, 1993

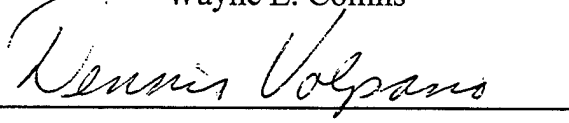
Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 1999

Author: 
Wayne E. Collins

Approved by: 
Dennis Volpano, Thesis Advisor


Xiaoping Yun, Second Reader


Dan Boger, Chairman
Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Military environments require flexible network configurations that must adapt under dynamic and mobile operating conditions. Wireless data networks offer some solutions. A wireless network typically requires an access point to bridge network traffic between wireless and wired media. These devices though are often too inflexible for use in such dynamic conditions. One major problem is that they are dedicated to a single type of wired network, usually Ethernet, which prevents them from being used to bridge traffic to other kinds of networks, for example, ATM or even cellular.

This thesis shows how any device running a recent Linux kernel can be configured to route packets in way that simulates an access point. The advantages of such a configuration are described and its potential for military use is discussed.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

The author thanks his family for their generous support during this endeavor.

The author additionally thanks Joe Wronkowski, a fellow student, whose familiarity with Linux and willingness to share his knowledge and equipment was instrumental in executing the tests contained herein; Professor Xioping Yun for providing office space, equipment, and an environment to succeed; Professor Dennis Volpano for his guidance, which cannot be underestimated.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Over the past ten years, the number of commercially available wireless data networking products has increased significantly. Some of the products use technology that has been in use in military radios for years (e.g. SINCGARS). Emerging standards, like IEEE 802.11-1997, should facilitate growth in the area as different products begin to operate with one another. Research at the Naval Postgraduate School and elsewhere suggests that commercial products can be used in a wide variety of applications on board ships and submarines without changing their electronic signatures [Debus, 1998].

The U.S. Marines Corps is also interested in the technology for its mobility. Lightweight local-area networks (LANS) can be assembled and disassembled easily and quickly, giving forces the ability to rapidly share information about their situation in combat. Mobility stems from the fact that there are no cables in a wireless LAN. Instead, hosts (mobile computing devices) are equipped with transceivers capable of transmitting and receiving Ethernet frames broadcast in an FCC-unregulated portion of the spectrum called the ISM band, or Industrial, Scientific and Medical band. There are no twisted-pair cables and devices can communicate directly with one another, sometimes called peer-to-peer communications.

But eventually we would like our mobile wireless networks to interface with an established wired network that may be installed at a command post or on a ship just off shore. For this, we need something that will bridge packets between the wired and wireless media. That bridge is commonly called an Access Point (AP). An AP has both wired and wireless interfaces and allows the mobile devices to appear as though they are

physically connected to the wired network. A mobile device may not be able to distinguish a host on the wired network from one with a wireless transceiver. Likewise, a host on the wired network may have no idea that it is transmitting packets to a mobile device, nor does it need to know.

Access points, however, can be a burden. Their placement often requires a fair amount of study to determine where they should be mounted to provide the best reachability without interfering with surrounding electronics. Typically, they are not relocated once they are placed. Further, an AP's hardware interfaces are fixed. One has to decide in advance on the type of bridge needed. For instance, if a ship or submarine is equipped with its own Ethernet, then an Ethernet bridge to the wireless network is needed. But one can imagine needing other kinds of bridges, say for instance, to an ATM network, a serial line, or even a cellular phone, depending on the environment.

Note that the interfaces of a host or mobile device are not fixed at all. Typically such a device has at least two Type II PCMCIA slots in which cards for many different interfaces can be inserted. Therefore, one slot could contain a transceiver while the other has a card for the interface of your choice. That means that, at least at the level of hardware, we have the ports necessary for an AP already on the mobile device. All that remains to be done is to configure the right software running on that device to route packets correctly. This gives us a completely symmetric wireless LAN configuration: every mobile device can also function as a bridge. There is no need for dedicated hardware, like an AP, to bridge wireless and wired networks. [Lewis and Volpano, 1998] Imagine needing a bridge in a portion of a ship or submarine where there is no access

point installed. As long as there is a network drop nearby, one of the mobile devices of the wireless LAN can be configured as a bridge, and later removed.

This thesis describes a way to configure a mobile computing device as a bridge to a wired network, thereby eliminating the need for an AP. We shall call this new bridge a *proxy AP*. In addition to the advantages described above, a proxy AP has other advantages as well over a traditional AP. They are truly mobile, as they can operate on batteries. Some of the devices tested have battery lifetimes of 6 hours or more. A proxy AP configuration may be cheaper. Older equipment, such as slower 386/486 Intel-based machines, can be reused to run the needed software to function as a proxy AP. The software is free, leaving the transceivers as the only expense. In many cases, one can completely avoid buying an access point, which currently costs between $1,300 and $1,500.

Security is a significant concern for wireless networks. Traditional access points treat security in a very ad hoc fashion. They provide address filtering at the physical layer to prevent some mobile devices from accessing the wired network. They also offer some weak encryption. Beyond this, however, security responsibilities rest with the administrator of the wired network. But this is clearly inadequate. We may wish to prevent a mobile device from receiving any packets on the wired network prior to being authenticated. Yet traditional access points broadcast all activity on the wired network, regardless of whether these packets are destined for wireless devices. That means any wireless device with the same kind of commercially available transceiver can sniff all activity on the wired network. Some vendors, like Lucent Technologies, deviate from the 802.11 standard and make this a bit more difficult by requiring all devices of a wireless

network to share a "secret" with the access point. But this secret (called a network name) may be shared with many users, each of whom is trusted to keep it a secret. Further, any "trusted" user can remove a wireless transceiver (especially if it is a PCMCIA card) and place it in another device where they have the privilege of putting it into unicast promiscuous mode and sniff all wired traffic. The same vulnerability exists with a wired network card, however, the key difference is that, with a wireless transceiver, this can be done without gaining access to a building where wired network drops are usually secured. One only has to be within range of the access point (up to 540m depending on the environment). The proxy AP configuration, presented in this thesis, broadcasts only those packets destined for wireless devices, a sort of selective broadcast. Though these devices are currently statically configured within the proxy AP, one can imagine adding an authentication protocol to the proxy AP to grant "wireless connections" dynamically to only those mobile devices for which the protocol succeeds.

It cannot be overstated that a proxy AP provides much more flexibility, and since it is a full-fledged computing device, it can be extended with new applications that give it more functionality. As mentioned above, its interfaces are not fixed, allowing bridges to be easily configured between the wireless and wired networks. It can be used to bridge an ATM network, a serial line or even a cellular phone. Unfortunately, these are all cases for which vendors today are making dedicated hardware solutions! Lastly, a proxy AP can actually outperform a traditional access point.

In Chapter II, we look at the steps one can take to configure a wireless device as a proxy AP. Chapter III contrasts this configuration with a traditional AP. Finally, Chapter IV discusses some future directions and applications of this work.

II. A PROXY AP CONFIGURATION

Here we describe the configuration of a proxy AP as a bridge to a wired subnet, specifically, 131.120.1.0. We also show how to configure a wireless device, or client, that uses the proxy AP. The proxy AP used in this study is an IBM ThinkPad 760XL, which we shall call Emily, running Linux (kernel 2.0.34). The client is another ThinkPad 760ED, which we call Megan, also running Linux (kernel 2.0.18). First we examine the client.

A. CLIENT CONFIGURATION

The client has a loopback interface and an Ethernet interface "eth0" with Internet Protocol (IP) address 131.120.1.84. Interface "eth0" is a wireless interface. The parameters of each interface are shown in Figure 1.

```
[root@megan /root]# ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Bcast:127.255.255.255  Mask:255.0.0.0
          UP BROADCAST LOOPBACK RUNNING MTU:3584 Metric:1
          RX packets:78 errors:0 dropped:0 overruns:0
          TX packets:78 errors:0 dropped:0 overruns:0

eth0      Link encap:10Mbps Ethernet HWaddr 08:00:6A:2A:DF:03
          inet addr:131.120.1.84 Bcast:131.120.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:2719 errors:0 dropped:0 overruns:0
          TX packets:2626 errors:0 dropped:0 overruns:0
          Interrupt:5 Base address:0x100 Memory:d1000-d2000
```

Figure 1. Wireless Client Interface Configuration

The HWaddr is the Medium Access Control (MAC) address of a Lucent WaveLAN wireless card. Notice that the Address Resolution Protocol (ARP) is running on interface "eth0" (if it were disabled, then we would see NOARP specified). It really does not have to run on wireless clients as these clients should not have to respond to ARP requests (see proxy ARP below). A MAC address could be negotiated as part of an authentication protocol. At that point, it could be stored in the proxy AP's ARP table and there should be no need to ever issue an ARP request to any wireless client thereafter. However, ARP must run on "eth0" if wireless clients can communicate directly with one another (see client's routing below).

The client's routing table is given in Figure 2. All network traffic from the client is sent to the gateway 131.120.1.60, the wireless interface on the proxy AP Emily.

```
[root@megan /root]# netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  Irtt Iface
127.0.0.0       0.0.0.0         255.0.0.0       U      3584 0         0 lo
0.0.0.0         131.120.1.60    0.0.0.0         UG     1500 0         0 eth0
```

Figure 2. Wireless Client Network Routing Table

This configuration means that wireless peers do not talk directly to each other. All network traffic will go through the gateway and peer-to-peer traffic will pass through the proxy AP. The Lucent WavePoint II AP handles peer-to-peer communication differently when transceivers are in ad hoc mode. It does not forward packets between clients; they communicate directly. Of course, we could configure the routing table to do the same if the wireless network were a subnet other than 131.120.1.0. It would require only one

more subnet-destination entry for the wireless subnet, and of course ARP would have to run on the interface for the client's IP address. This is necessary because other clients in the wireless subnet will issue ARP requests, and the proxy AP certainly cannot respond to them.

The client's ARP table is shown in Figure 3. The IP address shown is that of the wireless interface on Emily, and the HWaddress is the MAC address of that same interface, namely "eth0" (see Figure 4).

```
[root@megan /root]#arp -a -n
Address          HWtype     HWaddr              Flags   Mask    Iface
131.120.1.60     ether      08:00:6A:2A:DE:C8   C       *       eth0
```

Figure 3. Wireless Client ARP Table

It is important to note that the client's network interfaces and routing table must be configured manually, whereas its ARP table is created dynamically. Fortunately, under Linux, the manual configuration can be done via scripts, examples of which are given in Appendix C.

B. PROXY AP CONFIGURATION

The proxy AP, Emily, has three interfaces shown in Figure 4: "l0" is the local loopback, "eth1" is a wired card, and "eth0" is a wireless card. The IP address for "eth1" is Emily's IP address. Note that ARP is running on both "eth1" and "eth0".

```
[root@ emily pcmcia]# ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Bcast:127.255.255.255  Mask:255.0.0.0
          UP BROADCAST LOOPBACK RUNNING MTU:3584 Metric:1
          RX packets:22 errors:0 dropped:0 overruns:0
          TX packets:22 errors:0 dropped:0 overruns:0

eth1      Link encap:Ethernet HWaddr 00:60:97:8D:78:7B
          inet addr:131.120.1.48 Bcast:131.120.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:16597 errors:0 dropped:0 overruns:0
          TX packets:2728 errors:0 dropped:0 overruns:0
          Interrupt:9 Base address:0x300

eth0      Link encap:Ethernet HWaddr 08:00:6A:2A:DE:C8
          inet addr:131.120.1.60 Bcast:131.120.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:2668 errors:0 dropped:0 overruns:0
          TX packets:2774 errors:1 dropped:0 overruns:0
          Interrupt:5 Base address:0x100 Memory:d2000-d3000
```

Figure 4. Proxy AP Interface Configuration

The routing table for Emily is given in Figure 5. It routes network traffic either to the 131.120.1.0 subnet directly, to the gateway for that subnet, namely 131.120.1.1, or to Megan (131.120.1.84). Additional wireless clients could be added by adding their IP addresses as destination entries in this table, similar to that given for Megan.

```
[root@emily pcmcia]# netstat -nr
Kernel IP routing table
```

Destination	Gateway	Genmask	Flags	MSS	Window	irtt	Iface
131.120.1.84	0.0.0.0	255.255.255.255	UH	1500	0	0	eth0
131.120.1.0	0.0.0.0	255.255.255.0	U	1500	0	0	eth1
127.0.0.0	0.0.0.0	255.0.0.0	U	3584	0	0	lo
0.0.0.0	131.120.1.1	0.0.0.0	UG	1500	0	0	eth1

Figure 5. Proxy AP Routing Table

By putting wireless clients in the proxy AP's routing table in this fashion, we limit the network traffic on the wireless medium to only those clients. This is quite different from a traditional AP which broadcasts all network traffic on the wired medium.

Hosts on subnet 131.120.1.0 will need to send packets to Megan, however, Megan is not attached by cable to this subnet. Hence Megan is unable to respond to ARP requests from these hosts. Thus we must add an entry to Emily's ARP table so that Emily serves as a proxy ARP server for Megan. Any requests for the MAC address corresponding to Megan's IP address will cause Emily to reply with the HWaddr of its own wired interface. Figure 6 shows the ARP table for Emily. Emily will therefore receive packets destined for Megan and will forward them to Megan.

```
[root@emily pcmcia]# arp -a -n
?(131.120.1.1) at 00:00:0C:75:29:00 [ether] on eth1
?(131.120.1.84) at 08:00:6A:2A:DF:03 [ether] on eth0
?(131.120.1.13) at 08:00:20:71:27:19 [ether] on eth1
?(131.120.1.84) at 00:60:97:8D:78:7B [ether] PERM PUB on eth1
```

Figure 6. Proxy AP ARP Table

The last entry in the table specifies that any ARP requests on interface "eth1" (wired interface) for the MAC address correspoinding to Megan's IP address (131.120.1.84) will be replied to with MAC address 00:60:97:8D:78:7B.

What we have presented is basically a simple routing-based solution to replacing an access point with a machine, here called Emily, capable of forwarding packets. No other changes, for instance to the PC card drivers, are necessary with this approach.

C. NOTES ON CONFIGURING PROXY AP WITH WINDOWS

Configuring different laptops loaded with various versions of Windows proved problematic. Windows 95 and 98 do not support IP routing, so we were unable to route IP packets between the two PC card interfaces. Microsoft technicians attempted to provide a "workaround solution" that involved editing the registry and, in the end, it did not work. The reason for the failure is unclear, as the operating system and PC card vendors faulted each other. Microsoft technical support claims that Windows 98, Second Edition, released during summer of 1999, had the ability to forward IP packets as a supported feature. However, no testing was performed to confirm it.

The assignment of IP addresses became an issue while troubleshooting the difficulties configuring the Windows platform. Two distinct methods were used to assign IP addresses. Initially, wireless clients were given IP addresses in the same subnet as the wired clients. This is the method used in the successful tests conducted with the Linux AP proxy. The other method was to put the wireless clients on a different subnet; this was done at the suggestion of Microsoft technicians. In this latter case, the class C network of 192.168.0.0 (used for testing "off-line" IP hosts) was used for the wireless clients. The assignment of a different subnet to the wireless PC card had no apparent effect on the conflict in the Windows laptop.

Both NT Server and Workstation were tested to determine their capability to act as an AP proxy. Both operating systems were loaded with service patches 3, 4, and 5. The wired and wireless card provided network connectivity when used individually.

Because there is no "plug and play" functionality in NT, there is no "hot swapping" of cards; i.e., the machine must be powered off to safely remove or insert PC cards. The PC cards must be inserted into the laptop before bootup to ensure they are correctly recognized and configured. Booting with both cards inserted resulted in either a boot failure (referred to as the "blue screen") or a system lockup. Removing a card during a system lockup will restore use, but subsequent re-insertion does not initialize the card. The only recovery from a boot failure is to reboot after changing the environment (i.e., removing one of the PC cards). The apparent IRQ conflict could not be resolved using the PC card settings available in NT. It is not clear whether the conflict was due to operating system or card driver inadequacies.

IP forwarding, which is necessary to route IP traffic between interfaces (represented by PC cards), is enabled under the Control Panel in Network Properties. Doing this should enable a multi-homed host to service, or route between, different subnets. While we have successfully configured NT in this scenario using two wired cards in the past, NT's failure to boot with both a wired and wireless card did not allow us to configure a proxy AP using NT.

D. NOTES ON CONFIGURING PROXY AP WITH LINUX

The Linux proxy AP was configured using Red Hat 6.0. It contained two ethernet interfaces: one wired, one wireless. Each interface had an IP address in the same subnet.

Using Card Services for Linux (version 3.0.4) enables us to initialize the drivers for various cards. Scripts initialize the cards and configure the networking options. The /etc/pcmcia/config.opts file initializes the card driver for the WaveLAN card. The /etc/pcmcia/network.opts configures the network options, setting the IP address for each card. Finally, the /etc/rc.d/rc.local script uses the "route add" and "arp" commands to modify the kernel's IP routing table. Specifically, a route is added for each wireless client expected and a permanent ARP table entry is made so the proxy AP will serve as the proxy ARP for that client. See Appendix C for complete listings of the files that were changed on the proxy AP.

Configuring the wireless client is independent of the operating system used. The wireless PC card is loaded and initialized for the specific operating system. The interface is then configured with an IP address, which is on the same subnet as the wireless interface of the proxy AP. The default gateway IP address of the client is set to the wireless interface on the proxy AP. See Appendix D for complete listings of the files that were used on the Linux client.

III. CONTRASTING THE AP WITH THE PROXY AP

In this chapter, we compare our proxy AP with one of the market leaders, namely the Lucent Technologies WavePOINT II AP. It should be noted that it is the WavePOINT's design that allows one to build a simple routing-based proxy AP in the first place. Unlike all other vendors, Lucent requires its AP to use at least one wireless PC card which houses a transceiver. It is this card that can be placed in a mobile computer and allow that computer to function as an AP. Other vendors manufacture access points where transceivers are built in and cannot be removed. These AP designs cannot be simulated as easily which makes them very unattractive when one wants a single device which can be configured to support different kinds of bridges.

We have in mind some desirable properties of a wireless bridge, such as mobility, security, extensibility, and throughput. And we would like to know how well the proxy AP and WavePOINT II satisfy these properties. Also, the WavePOINT II has certain features and we need to know whether our proxy AP configuration has them too. Some of these features are implemented by the wireless card, while others are implemented by software (version 3.28) on the AP itself [Van der Moolen, 1998]. Those provided by the card carry over to the proxy AP since it uses one of the cards. For example, Lucent's Wired-Equivalent-Privacy (WEP) card supports symmetric encryption, so a proxy AP would inherit this feature. The others need to be implemented by code on the proxy AP, which in our case, is a Linux kernel.

A. MOBILITY

We mentioned that the lack of cables and the ability to use batteries afford wireless users a level of mobility that is unavailable with a traditional AP. Mobility is important in dynamic military operational environments. The WavePOINT II is "portable" in the sense that it can be relocated to another place with AC power, but it is not "mobile" in the sense that it still functions as a bridge while being moved. This is needed when the proxy AP is a bridge (more precisely a gateway) to another wireless network. The WavePOINT II is not battery powered, nor are any of the other access points currently on the market. The proxy AP, however, does have a battery, and depending on the device used for it, will have varying lifetimes.

B. SECURITY

Security is a major consideration when deploying a wireless LAN. We can address the security supported by the WavePOINT II and the proxy AP in layers.

1. Physical Layer

The first layer is the physical layer (PHY), which in the case of the WavePOINT II is Direct Sequence Spread Spectrum (DSSS) modulation. Spread spectrum technology takes a narrow band signal and spreads it over a wide-band spectrum, effectively lowering its signal strength for any given channel. When scanning the entire spectrum the signals fall under a pseudo-noise level, making the network traffic harder to find. This spread spectrum technology has been used for years by the military as a more

reliable and secure method of transmitting signals. Spreading the signal increases the opportunity for a receiver to correctly interpret transmissions, even in the presence of noise. For a more detailed explanation of DSSS and the other PHY layer specified in the IEEE 802.11 Standard (namely Frequency Hopping and Infrared), see Appendix A. DSSS is a property of the transceiver housed within the wireless PC card, so its advantages carry over to the proxy AP.

2. Selective broadcasting

The WavePOINT II and other traditional access points function as transparent bridges. All traffic on the wired medium is broadcast directly to the wireless medium, and is available to any station that possesses the same type of commercially-available card. Our simple routing-based solution, on the other hand, filters wired network traffic. All wired traffic is blocked from wireless devices unless addressed to an authenticated wireless client. Only packets destined for authenticated wireless clients pass through the proxy AP and get broadcast over the air. Notice that this implies wireless clients do not receive IP broadcast packets, and, as a result, no client could be a DHCP or BOOTP server for example.

3. Filtering

MAC address filtering is possible with the WavePOINT II. Each wireless interface can have a list of stored MAC addresses. Only frames whose source Ethernet

addresses are in the list are allowed to pass; all others are dropped. MAC address groups can be specified using wild-card characters.

MAC address filtering is a fairly weak security mechanism. First of all, anyone who merely gains possession of a card with an acceptable MAC address has access to the wired network (assuming the WavePOINT's network name is known to the card user). But in some cases, a person doesn't even need to possess the card but only needs to know an address in the list. The Linux driver for the WaveLAN/IEEE card (version 3.10) is distributed in source form under the GNU public license. Only one line of code in the driver needs to change in order to allow a card's factory universal MAC address to be temporarily reset to any universal address. Once this is done, any WaveLAN/IEEE card can bypass filtering regardless of its factory MAC address.

The WavePOINT II can also be configured to screen certain protocols. For example, an AP can refuse bridging to Banyan VINES, NetBUI, and Apple Talk traffic. Up to 20 protocols can be filtered at a time. IP firewalling in Linux 2.0 kernels, and IP chains in 2.2 kernels, provide this sort of flexibility and more.

4. Encryption

The IEEE 802.11 standard describes Wired Equivalent Privacy (WEP) as an optional method to achieve privacy on IEEE 802.11 compliant wireless networks. It is designed to be reasonably strong and efficient, and can be implemented in either hardware or software. Lucent Technologies has implemented WEP in hardware in the form of a chip installed on their "Silver" label cards. Encryption on the WavePOINT II is

possible only when using Silver label cards with the security chip installed at the factory. ("White" label and "Bronze" label cards do not support WEP, and cannot be upgraded to support the encryption.) Administrators then select an encryption key that must be shared by all wireless clients on that network. The management of this key is not specified by the 802.11 standard. Since encryption at this level occurs in the card, it carries over to the proxy AP. However, a proxy AP and the mobile client can be loaded with alternative application-based encryption to protect privacy.

5. Authentication

The 802.11 Standard requires mutually acceptable, successful, authentication. [IEEE 802.11-1997, p. 20] The standard specifies two forms of authentication: Open System and Shared Key. Both forms simply authenticate a host to another host or to an AP at the link level. Open System is the default method, and allows any station to be authenticated. Shared Key assumes the hosts know a shared key. WEP must be in use to invoke the Shared Key authentication scheme. For more information on IEEE 802.11 authentication, see Appendix A.

The WavePOINT II AP can operate in an "open" or "closed" mode. The open mode is IEEE 802.11 compliant, and will accept network traffic from compliant implementations from other vendors. However, it also has an additional layer of authentication that presumes knowledge of a network name. When a WavePOINT II AP requires the use of a network name, it is operating in a closed mode and is no longer

IEEE 802.11 compliant; in this mode, it will no longer accept traffic from other vendor implementations.

A proxy AP can implement any authentication protocol and might require successful authentication before forwarding any packets to a wireless client. The important point is that the proxy AP, especially a Linux implementation, can make use of a variety of protocols for this purpose. This is in addition to any shared-key based authentication scheme based on Lucent's WEP architecture since the proxy AP can also use Lucent's WEP card.

C. EXTENSIBILITY

We mentioned the ability of a proxy AP to accommodate many types of interfaces. Traditional access points do not have the same level of flexibility, and require a separate hardware device for each pair of media to be bridged.

A proxy AP has the interesting capacity of bridging two IEEE 802.11 compliant LANs using different physical media. Several types of wireless systems have been introduced into the military, both for testing purposes and the fielding of small systems. As research continues in this fast-growing area, the number of legacy systems will increase until a standard implementation is chosen. However, it is not clear whether this choice of standards will, in fact, be made. The IEEE 802.11 standard's recognition of three incompatible PHY specifications (DSSS, FHSS, Infrared) suggests that these fundamentally different technologies will be present on the market for a long time.

One can envision wireless networks on the battlefield employing several different media. Network traffic is being passed via radio, cellular, coaxial cable, twisted pair, fiber optic, and serial lines. A proxy AP can link all of these media at the hardware level. Devices and drivers exist for the interfaces to these media; all that is required is to configure the appropriate routing of packets.

D. THROUGHPUT

On average, our proxy AP showed that it could outperform the WavePOINT II AP in our experiments. Table 1 shows throughput speeds using the Linux client; Table 2 shows throughput speeds using the Windows 98 client. Each cell in the table represents the average speed achieved after four file transfers under the specified conditions. These are summary results; see Appendix B for a thorough description of the experiment and the data it generated.

Using the Windows 98 client generated similar speeds for both the proxy AP and the AP. The reason for the difference in speeds between the use of the different clients is unclear. Potential reasons could include different versions of FTP and background contention for the Ethernet connection. The Linux client starts daemons that use the network connection and may slow overall throughput. The system administrator can disable these daemons.

	FILE SIZE	
	4.5 MB	6.5 MB
Proxy AP	95.5 Kbps	98.25 Kbps
AP	88.63 Kbps	91 Kbps

Table 1. Average Throughput Speeds Using Linux Client

	FILE SIZE	
	4.5 MB	6.5 MB
Proxy AP	100.56 Kbps	104.37 Kbps
AP	101.83 Kbps	104.76 Kbps

Table 2. Average Throughput Speeds Using Windows 98 Client

IV. CONCLUSIONS

We have shown that a mobile client can effectively function as an AP, routing network traffic between wired and wireless media. This capability allows us to take advantage of applications running on a mobile client. The proxy AP can continue to act as a bridge in a mobile environment for a substantial length of time due to batteries. It can handle a variety of network interfaces through built-in ports or available PC cards. Its software can lend additional security to hardware solutions inherent to the wireless card through encryption, firewalls, and authentication protocols. Filtering minimizes the wireless network traffic and thus lessens susceptibility to network sniffers. The throughput is comparable or better than an AP, and the hardware can be cheaper.

The idea of a wireless client also acting as a proxy access point offers some exciting options. The additional flexibility in network configurations provides the Marines with the potential to deal with more dynamic and ever changing environments and scenarios. A proxy AP is inherently more flexible than a traditional AP. An AP's placement is typically based upon a study of the environment, taking into consideration the number of users, radio signal propagation characteristics, range, and so on. Dynamic environments do not maintain the same characteristics over time. Unforeseen or changing requirements demand solutions that a more traditional static AP cannot deliver.

For example, Navy ships are notoriously difficult environments to arrange hardware installations, especially those requiring changes to decking (as in a permanent AP installation), and this has proven difficult to Marines attempting to deploy with their

own LAN equipment. A proxy AP can reside in a desktop device that already has room designated for it.

As another example, consider the following. A unit of Marines maintains a network of wired and wireless hosts while in garrison. Wireless hosts are used for tracking local exercises, attending staff meetings, and warehousing duties. Proxy access points provide efficient and secure wireless connectivity to mobile hosts while simultaneously providing client functionality for the user of that device. With each proxy AP, is a package of adapters for known military network interfaces.

Now suppose this same unit of Marines deploys on board naval vessels. The Navy uses a wired network specific to its ships. The Marines establish their proxy access points throughout the ship to accommodate their wireless clients. Note that the type of network the Navy has installed is irrelevant; the Marines require only the adapter for the proxy access points to route their network traffic to the wired network, whatever that might be (Ethernet, ATM, etc.).

When the order comes to attack the beach in an amphibious landing, the Marines unplug their wireless-capable clients from the ship's AC power and board the various types of landing craft (e.g., hovercraft, helicopter, and amphibious assault vehicle). The transit time for landing craft between naval shipping and the shore can vary greatly, and can be extensive. Currently communications are limited to little more than voice. Significant changes to necessarily dynamic operational plans are extremely difficult to coordinate effectively, and maintaining network connectivity is crucial.

Network connectivity is maintained while the landing craft are on or in the ship because the Marines are running on battery power and communicating using the wireless

medium. As the landing craft depart naval shipping, they spread out geographically; however, they maintain network connectivity themselves using their own network. Again the type of medium used for the landing craft network is irrelevant as the Marines merely swap the PC cards (or otherwise change network interfaces) in their proxy access points. With a single device configured as a proxy AP, all other Marine devices in the landing craft also maintain full network connectivity.

When the Marines land on shore, connectivity to the network is maintained through the use of a wireless infrastructure established with relay stations in landing craft or airborne vehicles. It is important to note that continuous network connectivity is maintained, not only because of the proxy AP's ability to work off battery or AC power, but that the transition between network interfaces occurs without the need for rebooting the proxy AP. Network administrators could insert a new network interface card (NIC) for the new medium, bring up its interface and then modify the kernel's routing table, to make for a seamless changeover.

If the infrastructure of the host country can support it, Marines can immediately tap into the land-based network and use it to their advantage. Note that virtually any medium for the land-based infrastructure can be accommodated for the Marine proxy access points can host any type of network for which there is a NIC. No commercially available AP comes close to this functionality.

Communication equipment from the ships is always among the first to land, so in addition to any existing infrastructure the Marines should enjoy stable network access using organic assets soon after landing. This enhanced capability is a force multiplier.

One can envision the network coverage seeping over the beachhead even as the Marines first walk ashore.

Configuration issues can be minimized with adequate planning. Client IP addresses need not be changed as the deployment progresses. Knowledge expectations of typical wireless users are little more than current expectations of wired users; those who carry a proxy AP need only be trained in the flexibility of using multiple interface types. Security is maintained through the use of spread spectrum technology, the selective broadcasting available through the proxy AP, hardware-based encryption (as well as any additional software-based encryption that may be used, either in addition or as an alternative), firewalls configured according to the threat, and authentication on the devices.

The selective broadcasting available by using a proxy AP even lends itself to the organizational structure of the Marine Corps. For example, squads generally stay in platoon formations that roughly approximate the range capabilities of a wireless cell, so the platoon commander may serve as the proxy AP for the squad leaders. Autonomous operations by squads for small unit taskings are still possible.

Recent Marine Corps Advanced Warfighting Experiments (AWEs) demonstrated the effective use of wireless clients at the small unit level. However, the demonstration had to simulate certain conditions, and thus did not always portray realistic scenarios. For example, access points were installed atop tall buildings prior to, and for the duration of, the demonstration. If this were attempted, in a real battle situation, the enemy would immediately have a fixed target to attack. It also means there is a fixed asset that must be eliminated if it should fall into enemy hands. Also issues such as set-up time and reliance

on continuous AC power cannot be ignored. Further, spare access points may be needed if they become damaged. In the case of a damaged proxy AP, we can turn to another client device to take over the role of proxy AP.

The preceding applications of the proxy AP are speculative. Although these directions seem plausible, more experience is needed. So far, experience with the simple routing-based proxy AP has been limited to the classroom and laboratory where it has performed flawlessly. Future work involves actually deploying it in the scenarios above.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. IEEE 802.11 STANDARD

Here we provide background information on the current wireless technology, along with a brief description of the Institute of Electrical and Electronic Engineers (IEEE) 802.11 standard. The discussion of 802.11 is not meant to be a regurgitation of the standard or even serve as a comprehensive tutorial; it is intended to provide the reader an overview of the important considerations the standard addresses. More importantly, the discussion highlights what is NOT addressed in the standard.

Wireless technologies exist in several forms, and are distinguished primarily by their characteristics that are a direct result of their use of the electromagnetic spectrum. Physical properties of the electromagnetic spectrum used and regulations placed by various countries determine in the characteristics of wireless devices.

Wireless LAN solutions exist in the 400 megahertz (MHz) range that provide large coverage area with the ability to support a limited number of users. Other solutions exist in the 900 MHz range, which provides a balance between range and the capacity to support a number of users. Several groups are examining solutions in the 5 GHz range. The 802.11 standard concentrates on the 2.4 GHz range, encompassing one of the Industrial Science and Medical (ISM) bands. This range of frequencies is open for use in the United States by the Federal Communications Committee (FCC), in Europe by the CEPT (Conference of European Postal and Telecommunications), and in Japan by the MPT (Ministry of Post and Telecommunications).

The 802.11 standard was approved June 26, 1997 after a seven year effort. It effectively combines the efforts of a consortium of vendors to give some structure to the wireless market, sufficient enough to allow interoperation.

The IEEE Working Group is composed of two task groups, A and B. Working Group A is working on standards for the 5 GHz range. Working Group B is working on the 2.4 GHz range (ISM). Both groups are focusing on DSSS to achieve higher data rates than those currently outlined in the 802.11 standard.

A. ARCHITECTURE

Here we discuss the different types of wireless architectures specified by the 802.11 standard. The basic service set (BSS) represents a cell in which member stations (hosts) can communicate with each other; it must have at least two stations to be a "minimally conformant network." A distribution system (DS) connects basic service sets. An extended service set (ESS) is a set of basic service sets connected via the DS. Any station that connects a BSS with the DS is an access point.

The stations that connect the DS with other non-802.11 LANs are termed portals. For the purposes of this thesis, the terms AP and portal are considered synonymous, since the traditional AP provides both services in a single hardware device.

The standard makes no assumptions about the physical location of the basic service sets, and thus the ESS may be of arbitrarily large and complex.

1. Independent (ad hoc)

An independent BSS (IBSS) consists only of stations; it does not have any connection to a DS. Stations communicate directly with each other. Only station services apply to an IBSS (see MAC services below for a description of the station services authentication, deauthentication, and privacy).

2. Infrastructure

An infrastructure network is one that possesses an AP, which is a station that provides DS services.

B. MAC SERVICES

The services described below reside at the MAC layer of the standard. Most of the services are provided as part of the DS, while the station provides the others. All MAC services are PHY independent; that is, the services are provided regardless of the PHY (DSSS, FHSS, Infrared) used.

1. Distribution

This service is provided by the distribution system, and delivers the message from its originating point within the DS to its destination point within the DS. Because the distribution system implementation is beyond the scope of the 802.11 standard, details (e.g., such as what the AP does when the communicating stations are within the same BSS) are not covered.

2. Integration

Integration is a distribution system service that is invoked when a message passes between a non-IEEE 802.11 LAN and the DS. Its details are also DS implementation-specific.

3. Association

A station must be associated with one and only one AP to pass messages on the distribution system. The station chooses an AP with which to associate (assuming there is a choice) through a scanning process where the station normally chooses that AP that provides the strongest signal. The station then initiates the association process and can pass messages on the DS once that process is complete.

4. Reassociation

Reassociation allows a station to roam within a BSS or between basic service sets in the same extended service set by allowing that station to change its association with an AP while informing the distribution system of the change. Roaming outside of an ESS is not supported.

5. Disassociation

Either an AP or a station may break the association by notifying the other device. A disassociated device will no longer be able to receive messages from the DS.

6. Authentication

Wireless LANs cannot be practically secured in the same fashion as wired LANs. The authentication service is provided by each station to identify itself to any other station with which it wants to communicate.

IEEE 802.11 does not mandate the use of any particular authentication; however, "The IEEE standards committee specifically recommends against running an IEEE 802.11 LAN with privacy but without authentication. While this combination is possible, it leaves the system open to significant security threats." [IEEE 802.11-1997, p. 62]

The standard specifies two forms of authentication: Open System and Shared Key. Both forms simply authenticate a host to another host or to an AP at the link level. Other forms of authentication are acceptable, as is expansion to those in the standard.

Open System authentication allows any station to be authenticated, whereas Shared Key authentication involves knowing a WEP key. WEP must be in use to invoke the Shared Key authentication scheme. The WEP key is distributed to stations via a secure channel not directly associated with the 802.11 standard (e.g., key management is beyond the scope of the standard).

Authentication messages can only be unicast as only pairs can be authenticated. There is no limit on the number of pairs a host may be authenticated with at any given time, however.

Authentication is required prior to a successful association. The goal is to provide the same level of authentication provided to a station that has access to a wired LAN, and does not extend to user or message authentication. In other words, the authentication is

only for the link, not the messages that pass along the link. Also, there is no user-to-user authentication. Note that this link-level authentication occurs regardless of any higher level authentication in use.

7. Deauthentication

Deauthentication is the notification of stopping an existing authentication. Because authentication must be performed to be associated, deauthentication also breaks an association. Any station can make such a notification, and like a disassociation the receiving station cannot refuse such notification.

8. Privacy

Privacy is protecting message content from being read by unauthorized entities. The standard provides for the optional use of WEP to encrypt wireless traffic with protection strong enough to be "at least as secure as a wire." [IEEE 802.11-1997, p. 21]

WEP can only act on data and some management messages. Those messages passed to initially establish authentication and privacy are passed unencrypted. The default state for 802.11 stations is to pass traffic unencrypted. Attempts to communicate between stations configured with different privacy standards are acknowledged (to minimize retransmissions) but discarded.

C. MAC ARCHITECTURE

The distributed coordination function (DCF) implemented on all stations is carrier sense multiple access with collision avoidance, or CSMA/CA. It is a wireless LAN's inherent lack of guarantee that stations can hear all other stations that precludes the use of carrier sense multiple access with collision detection (CSMA/CD), which is the basis of wired LANs. Any station desiring to transmit senses the medium using clear channel assessment (CCA, a detection scheme for RF energy). If the medium is idle a certain period of time, the station transmits. If the medium is busy, the station waits. Once the medium is idle, a random backoff interval timer commences, and when it reaches zero the process starts again.

If the data frame to be sent exceeds a configurable threshold, then a medium reservation method is invoked. A request-to-send (RTS) frame is sent to the destination station that includes the length of time that the medium is required. The destination station responds with a clear-to-send (CTS) frame that acknowledges the RTS and effectively advises all other stations to defer transmitting until the requesting station is done. The administrator sets the threshold; it is also possible to both refuse the use or mandate the use of RTS/CTS frames.

The use of RTS/CTS frames precludes a hidden node from causing interference or collisions at the AP. A hidden node is one that cannot be heard by a distant station while the AP can hear both stations.

The point coordination function (PCF) uses the AP as a point coordinator, effectively polling stations to determine who can transmit. Because a point coordinator

can allocate transmit time and because traffic using PCF may take priority over traffic using DCF, it is possible for a contention free access method to be implemented for periods of time. This specific implementation is not covered in the 802.11 standard, though it is mandated that PCF and DCF must coexist. For additional information, consult the IEEE 802.11 standard, Chapter 9.

D. PHY ARCHITECTURE

Along with the description of a single media access control (MAC) layer, the 802.11 standard describes three physical (PHY) layers. The PHY layers include infrared, direct sequence spread spectrum (DSSS), and frequency hopping spread spectrum (FHSS). Each of the PHY layers supports the MAC; however, devices using different PHY layers cannot communicate directly. The MAC supports the use of access points (an infrastructure network) and independent stations (an ad hoc network).

1. Spread Spectrum Overview

The spread spectrum technology used in DSSS and FHSS generally implies a radio transmitter that spreads a relatively narrow-band signal over a wide-band spectrum. This spreading is also known as whitening. Using a wider band provides greater opportunity for a receiver to correctly interpret the transmission even in the presence of noise. Also, the techniques used to spread the signal inherently add security to the system.

Transmitting power is limited by government regulations in various parts of the world, but the PC card slot limits the transmit power from a wireless client to 50-100 milliwatts (mW). This limitation of the PCMCIA form factor thus becomes a determinant for transmit power. Range potential between PHY thus becomes a function of minimum required signal-to-noise ratio (SNR). [Lucent Technologies, 1998]

The infrared PHY and FHSS PHY support 1 megabit per second (Mbps) data rate with an optional 2 Mbps data rate, while the DSSS support both 1 and 2 Mbps. FHSS vendors (specifically, Breezecom) claim 3 Mbps by modulating the signal using a patented algorithm.

2. Frequency Hopping Spread Spectrum (FHSS)

FHSS systems hop between narrow band transmissions in a pseudo-random order for a specific time period known to both receiver and transmitter. With 78 hopping patterns defined, fifteen FHSS systems can reasonably be colocated. Standardized hop sequences are published in Appendix B of the 802.11 standard. At least one vendor, Breezecom, allows the customization of hopping patterns. However, this mandates that all systems on a given LAN use the same hopping pattern. Customized hopping patterns are not fully 802.11 compliant.

Local authority governs the hop rate (the length of time spent on a given frequency); 79 channels are defined for use in the United States and most of Europe. The numbers are different for Japan, Spain, and France (see the 802.11 standard, Chapter 14, for more information). Data rates of 1 and 2 Mbps are supported, using 2- and 4-level

GFSK respectively (GFSK is Gaussian frequency shift keying, a modulation technique). The standard outlines data rates of 1 to 4 Mbps in steps of 500 Kbps for possible future use.

3. Direct Sequence Spread Spectrum (DSSS)

With DSSS a spreading code (called a chip sequence, specifically the "Barker sequence spreading"), an 11-chip pseudo-noise (PN) code increases the modulation rate and broadens the signaling band. This uses more of the frequency band to get the signal across; the 11-chip Barker sequence used requires 11 MHz to yield a raw bit rate of 2 Mbps. WaveLAN reserves 22 MHz to provide a clear channel; since the 802.11 standard mandates a 30 MHz distance between carrier frequencies, three DSSS systems may be collocated within the 83.5 MHz available in the 2.4 GHz ISM band.

DSSS uses differential binary phase shift keying (DBPSK) for the 1 Mbps ("basic") data rate and quadrature phase shift keying (DQPSK) for the 2 Mbps ("enhanced") data rate. All headers are transmitted at the 1 Mbps data rate.

4. Diffuse Infrared (IR)

IR uses near-visible light (850-950 nm) similar to devices in typical remote controls. Transmissions are termed diffuse IR because the signal is not directed and the receiver relies on reflected signals. Direct sunlight severely hinders IR signals.

Distance and obstacle limitations serve to significantly limit the range of IR devices, making them appropriate for indoor and close-range use. The ability to segment

and isolate portions of the LAN can also be viewed as inherently more secure from eavesdroppers. This characteristic also aids in minimizing interference from other IR sources. There are no frequency or bandwidth restrictions placed on IR emissions worldwide.

Infrared's 1 Mbps basic data rate uses 16-PPM (pulse position modulation) and the 2 Mbps enhanced data rate uses 4-PPM. [IEEE 802.11-1997]

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. THROUGHPUT DATA

To test the performance of the proxy AP we conducted experiments that measured file transfer times. A single server on the wired network functioned as the FTP server. We connected to the server from one of two clients via an AP or a proxy AP.

One client consisted of a laptop with a Pentium II 233 MHz Intel chip and 64 MB of RAM running Windows 98. It was equipped with a WaveLAN White label card. The versions of software and firmware are summarized in Table 3. The second client was a Toshiba Libretto with 32 MB RAM and running Red Hat Linux version 5.2. This client also used a WaveLAN White label card.

		Variant	Version
Network card	WaveLAN-II PC Card Type II Extended NIC	1	1.256
Primary Card Firmware	WaveLAN-II Privary Functions firmware	1	1.01
Secondary Card Firmware	WaveLAN/IEEE STA Function firmware	1	2.00
Driver	WaveLAN/IEEE Miniport driver	1	1.38
Utility	WaveMANAGER	1	1.3

Table 3. Client Software/Firmware Versions

The access point was a WavePOINT II with a WaveLAN White label card and an RJ45 jack connection to twisted-pair cabling. The proxy AP was a laptop with a Quantex 266 MHz with 64 MB RAM running Red Hat Linux version 6.0. The proxy AP contained a WaveLAN White label card and a 3Com3C589 Ethernet card.

The experiment consisted of transferring two files using File Transfer Protocol (FTP) from the FTP server to the clients. The two clients were collocated on the same desk, separated approximately two meters from the proxy AP and AP. The files used

were approximately 4.5 MB and 6.5 MB in size. Each transfer was conducted individually, and typical campus traffic was minimal because the experiment was performed after normal working hours.

Because times were consistently better using the binary mode over the ASCII transfer mode, separate data points were kept for each. The reason for the difference is unclear.

Each transfer was performed four times. Individual data points are shown in the following tables.

Linux client through Proxy AP

File size (bytes)	type of transfer	time (seconds)	speed (Kbps)
4513280	ascii	49	90
		46.1	96
		46.8	95
		47	94
6827520	ascii	69.6	96
		70	96
		70.3	95
		69.4	97
4513280	binary	47.7	92
		45.1	98
		44.8	98
		43.8	101
6827520	binary	67.1	99
		66.4	100
		65.7	101
		65.3	102

Linux client through WavePOINT II Access Point

File size (bytes)	type of transfer	time (seconds)	speed (Kbps)
4513280	ascii	49.3	90
		51.2	86
		49.5	89
		51.4	86
6827520	ascii	77.9	86
		77.3	87
		73.4	91
		77.5	86
4513280	binary	48.7	91
		47.5	93
		52.5	84
		48.7	90
6827520	binary	68.8	97
		70.4	95
		70.7	94
		72.1	92

Win98 client through Proxy AP

File size (bytes)	type of transfer	time (seconds)	speed (Kbps)
4513280	ascii	46.74	97.04
		44.38	102.2
		51.85	87.48
		45.21	100.33
6827520	ascii	67.23	102.11
		68.39	100.38
		67.71	102.2
		66.79	102.78
4513280	binary	42.12	107.15
		44.05	102.46
		43.55	103.63
		43.33	104.16
6827520	binary	63.82	106.98
		63.49	107.54
		64.7	105.53
		63.55	107.44

Win98 client through WavePOINT II Access Point

File size (bytes)	type of transfer	time (seconds)	speed (Kbps)
4513280	ascii	44.93	100.95
		46.47	97.61
		49.71	91.25
		45.53	99.62
6827520	ascii	65.53	104.76
		66.95	102.53
		66.46	103.29
		67.77	101.29
4513280	binary	41.74	108.13
		41.52	108.7
		42.67	105.77
		43.99	102.6
6827520	binary	68.21	100.1
		63.38	107.72
		62.89	108.56
		62.18	109.8

APPENDIX C. CONFIGURATION FILES FOR THE PROXY AP

/etc/pcmcia/config.opts

```
#
# Local PCMCIA Configuration File
#
# System resources available for PCMCIA devices
#
include port 0x100-0x4ff, port 0x1000-0x17ff
include memory 0xc0000-0xfffff, memory 0xa0000000-0xa0ffffff
#
# Extra port range for IBM Token Ring
#
include port 0xa00-0xaff
#
# Resources we should not use, even if they appear to be available
#
# First built-in serial port
exclude irq 4
# Second built-in serial port
#exclude irq 3
# First built-in parallel port
exclude irq 7
#
# Options for loadable modules
#
# To fix sluggish network with IBM ethernet adapter...
#module "pcnet_cs" opts "mem_speed=600"
#
# Options for Xircom Netwave driver...
#module "netwave_cs" opts "domain=0x100 scramble_key=0x0"
#
# WaveLan2 Network Driver
module "wavelan2_cs" opts "port_type=3"
```

/etc/pcmcia/network.opts

```
# Network adapter configuration
#
# The address format is "scheme,socket,instance,hwaddr".
#
# Note: the "network address" here is NOT the same as the IP address.
# See the Networking HOWTO.  In short, the network address is the IP
# address masked by the netmask.
#
case "$ADDRESS" in
*,0,*,*)
    # Transceiver selection, for some cards -- see 'man ifport'
    IF_PORT=""
    # Use BOOTP? [y/n]
    BOOTP="n"
    # Use DHCP? [y/n]
    DHCP="n"
    # Host's IP address, netmask, network address, broadcast address
    IPADDR="131.120.27.62"
    NETMASK="255.255.252.0"
    NETWORK="131.120.27.0"
```

```
    BROADCAST="131.120.27.255"
    Gateway address for static routing
    GATEWAY="131.120.24.1"
    Things to add to /etc/resolv.conf for this interface
    DOMAIN="ece.nps.navy.mil"
    SEARCH=""
    DNS_1="131.120.27.23"
    DNS_2=""
    DNS_3=""
    # NFS mounts, should be listed in /etc/fstab
    MOUNTS=""
    # For IPX interfaces, the frame type and network number
    IPX_FRAME=""
    IPX_NETNUM=""
    # Extra stuff to do after setting up the interface
    start_fn () { return; }
    # Extra stuff to do before shutting down the interface
    stop_fn () { return; }
    ;;

*,1,*,*)
    IF_PORT=""
    BOOTP="n"
    DHCP="n"
    IPADDR="131.120.27.68"
    NETMASK="255.255.252.0"
    IPX_FRAME=""
    IPX_NETNUM=""
    start_fn () { return; }
    stop_fn () { return; }
    ;;
esac
```

/etc/rc.d/rc.local

```
#!/bin/sh

# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

if [ -f /etc/redhat-release ]; then
    R=$(cat /etc/redhat-release)

    arch=$(uname -m)
    a="a"
    case "_$arch" in
          _a*) a="an";;
          _i*) a="an";;
    esac

    # This will overwrite /etc/issue at every boot.  So, make any
changes you
    # want to make to /etc/issue here or you will lose them when you
reboot.
    echo "" > /etc/issue
    echo "$R" >> /etc/issue
    echo "Kernel $(uname -r) on $a $(uname -m)" >> /etc/issue

    cp -f /etc/issue /etc/issue.net
```

```
    echo >> /etc/issue
fi

# This is to Establish the network of known users in the area.
echo "Establish local wireless clients "

/sbin/route add 131.120.27.69 dev eth1
/sbin/arp  -Ds 131.120.27.69 eth0 pub
/sbin/route add 131.120.27.70 dev eth1
/sbin/arp  -Ds 131.120.27.70 eth0 pub
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. CONFIGURATION FILES FOR THE CLIENT

/etc/pcmcia/config.opts

```
#
# Local PCMCIA Configuration File
#
# System resources available for PCMCIA devices
#
include port 0x100-0x4ff, port 0x1000-0x17ff
include memory 0xc0000-0xfffff, memory 0xa0000000-0xa0ffffff
#
# Extra port range for IBM Token Ring
#
include port 0xa00-0xaff
#
# Resources we should not use, even if they appear to be available
#
# First built-in serial port
exclude irq 4
# Second built-in serial port
#exclude irq 3
# First built-in parallel port
exclude irq 7
#
# Options for loadable modules
#
# To fix sluggish network with IBM ethernet adapter...
#module "pcnet_cs" opts "mem_speed=600"
#
# Options for Xircom Netwave driver...
#module "netwave_cs" opts "domain=0x100 scramble_key=0x0"
#
module "wavelan2_cs" opts "port_type=3"
#
```

/etc/pcmcia/network.opts

```
# Network adapter configuration
#
# The address format is "scheme,socket,instance,hwaddr".
#
# Note: the "network address" here is NOT the same as the IP address.
# See the Networking HOWTO.  In short, the network address is the IP
# address masked by the netmask.
#
case "$ADDRESS" in
*,1,*,*)
    IF_PORT=""
    BOOTP="n"
    DHCP="n"
    IPADDR="131.120.27.69"
    NETMASK="255.255.252.0"
    NETWORK="131.120.24.0"
    BROADCAST="131.120.24.255"
    GATEWAY="131.120.27.68"
    DOMAIN="ece.nps.navy.mil"
    DNS_1="131.120.27.23"
```

```
;;
esac
```

REFERENCES

Debus, S., *Feasibility Analysis for Wireless Computer Networks on Submarines Using Commercial Off The Shelf Components*, Master's Thesis, Naval Postgraduate School, Monterey, California, September 1998.

Institute of Electrical and Electronics Engineers Std 802.11-1997, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, 26 June 1997.

Lewis, T. and Volpano, D., Wired and Wired-er, IEEE Internet Computing, Vol. 2, No. 4, Jul/August 1998, pp. 97-99].

Lucent Technologies Nederland BD, WCND, *WaveLAN Wireless LAN Technology and Market Backgrounder*, 1998.

Van der Moolen, William, WavePOINT II Getting Started Guide, July 1998.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Road, STE 0944
   Fort Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, CA 93943-5101

3. Director, Training and Education 1
   MCCDC, Code C46
   1019 Elliot Road
   Quantico, VA 22134-5027

4. Director, Marine Corps Research Center 2
   MCCDC, Code C40RC
   2040 Broadway Street
   Quantico, VA 22134-5107

5. Director, Studies and Analysis Division 1
   MCCDC, Code C45
   3300 Russell Road
   Quantico, CA 22134-5130

6. Marine Corps Representative..... 1
   Naval Postgraduate School
   Code 037, Bldg. 330 In-116
   555 Dyer Road
   Monterey, CA 93940

7. Marine Corps Tactical Systems Support Activity..... 1
   Technical Advisory Branch
   Attn: Maj J.C. Cummiskey
   Box 55171
   Camp Pendleton, CA 92055-5080

8. Dennis Volpano 2
   Code CS/VO
   Naval Postgraduate School
   Monterey, CA 93943

9. Xiaoping Yun 2
Code EC/YX
Naval Postgraduate School
Monterey, CA 93943

10. Chairman, Code CS..... 1
Naval Postgraduate School
Monterey, CA 93943

11. Gordon Bradley..... 1
Code OR/BZ
Naval Postgraduate School
Monterey, CA 93943

12. Captain James Powell..... 1
IW/Powell Root Hall 200A
Naval Postgraduate School
Monterey, CA 93943

13. Wayne Collins 4
1255 Piedmont Drive
Upland, CA 91784