An Agent Based Intrusion Detection and Response System for Wireless LANs

By

Mohan K Chirumamilla

A master's project presented to the faculty of the graduate college in the University of Nebraska in partial fulfillment of requirements for the degree of Master of Science

Major: Computer Science & Engineering

Under the supervision of Prof. Byrav Ramamurthy

Lincoln, Nebraska
December 2002

Acknowledgement

I would like to thank Prof. Byrav Ramamurthy, my advisor, for his incessant encouragement, guidance and support. I extend my deepest gratitude to him for giving me the chance to work in the area of my choice.

I would like to thank Prof. David R. Swanson and Prof. Vinodchandran N. Variyam of the Computer Science and Engineering Department for obliging my request to be on my defense committee and for reviewing this work.

I acknowledge the support of the UNL Mobile Communications Research Project (MoCoRePro) and the NSF Secure Distributed Information (SDI) Infrastructure Project in funding this work.

Finally, I am greatly indebted to all my family members who have always been a source of moral support throughout my program.

Abstract

Wireless LAN technology, despite the numerous advantages it has over its competing technologies, has not seen widespread deployment. A major reason for markets not adopting this technology is its failure to provide adequate security. Data that is sent over wireless links can be compromised with utmost ease.

Security has always been a concern in communication networks as it is in many other areas. The important typical security issues that one should consider are threats to the physical network, unauthorized access to network resources and internal and external attacks. In the context of wired LANs, the solutions to the above issues are well defined and significantly reliable. The same approaches, however, cannot be adapted to the wireless LANs.

In this project, we propose a distributed agent based intrusion detection and response system for wireless LANs that can detect unauthorized wireless elements like access points, wireless clients that are in promiscuous mode, etc. The system reacts to intrusions by either notifying the concerned personnel, in case of rogue access points and promiscuous nodes, or blocking unauthorized users from accessing the network resources.

Contents

1 Introduction **9**

1.1 Wireless LANs…….10

1.1.1 Wireless LAN Components……..12

1.1.2 Wireless LAN Architecture……..12

1.2 Security in Wireless LANs……..14

1.3 Project Contribution…….14

2 Background **16**

2.1 WEP…….19

2.2 Related Work……...22

2.3 Our Project……...26

3 Design **28**

3.1 Central Administrator……..30

3.2 Agents……..32

3.3 Registration and Unregistration of APs……...34

3.4 Registration and Unregistration of Clients……..35

3.5 Registration and Unregistration of Agents……..36

3.6 Detection of Rogue Access Points and Promiscuous nodes……37

3.7 Scanning Agents and Access Points……....39

4 Implementation and Testbed **40**

4.1 Software Requirements……40

4.2 Hardware Requirements……...50

4.3 Test Bed……...50

4.4 Results……..51

4.5 Advantages and Disadvantages……....53

4.6 Future Work……54

5 Conclusion **56**

A Appendix **57**

References **58**

List of Figures

1.1 Wireless in OSI……11

1.2 CSMA in Wireless……...11

1.3 Ad-Hoc Mode……..13

1.4 Infrastructure Mode…….13

2.1 Open Systems Authentication……..17

2.2 Shared Key Authentication……..18

2.3 WEP at station A……..20

2.4 WEP at station B……..21

2.5 Netstumbler……..23

2.6a AirTraf – Scanning Access Points ……25

2.6b AirTraf - User based Scanning..……...25

3.1 An area divided into logical cells……29

3.2 Overview of the Architecture……..29

3.3 A VPN Tunnel between a mobile node and Central Administrator……....31

3.4 Access Point registration process…….…34

3.5 Client registration process……35

3.6 Agent registration…….…36

3.7 Detecting rogue access points……..37

3.8 Detecting promiscuous nodes……..38

3.9 Scanning access points or agents…….39

4.1 Access Point Registration Form……...41

4.2 Client Registration Form……..42

4.3 Agent Registration Form……..44

4.4 Scan Access Point GUI……45

4.5 Scan Agent GUI……...46

4.6 Pseudo Code for scanning rogue access points……47

4.7 Pseudo Code for detecting promiscuous nodes……...49

4.8 Test Bed……...50

4.9 Introducing a registered access point……...51

4.10 E-mail sent to the administrator when a rogue access point is installed……52

4.11 Introducing a rogue access point……52

4.12 E-mail sent to the administrator when a promiscuous node is detected……53

List of Tables

2.1 Comparison of Tools……....27

A Hardware used for the Test Bed……...56

Chapter 1

Introduction

An ever-growing demand for seamless computing power anytime, anywhere has ignited the widespread deployment of Wireless Lans (WLANs). Other advantages such as easy installation, reduced installation and maintenance costs have made WLANs an obvious alternative to wired networks. However, the IT sector is refraining from using this technology where data confidentiality is considered to be of high priority.

Security has always been a concern in communication networks as it is in many other areas. The important typical security issues that one should consider are threats to the physical network, unauthorized access to network resources, internal and external attacks and eavesdropping. In the context of wired LANs, the solutions to the above issues are well defined and are significantly reliable. The same approaches, however,

cannot be adapted to the wireless LANs due to the characteristics of the medium of communication, a radio signal. A radio signal, unlike the electrical signal in wired networks, can be received by any one with a receiver if present within the range of propagation of the signal. This makes Wireless LANs highly susceptible and vulnerable to attacks.

1.1 Wireless LANs

Academic interest in combining network technologies with radio communications first started in 1971 as a research project called ALOHANET at the University of Hawaii. In 1985, the Federal Communications Commission (FCC) made the commercial development of radio-based LAN components possible by authorizing the public use of the Industrial, Scientific and Medical (ISM) bands- frequencies between 902 MHz and 5.8 GHz. In June 1997, the IEEE 802.11 working group released the first version of Wireless LAN medium access control and Physical Layer specifications which prompted vendors to release 1Mbps and 2Mbps 802.11 - compliant radio cards and access points. In December 1999, the IEEE released supplements (802.11a and 802.11b) to the 802.11 standard in order to increase the performance of the wireless LANs up to 54 Mbps.

Figure 1.1 shows where the wireless technology comes into the picture in the overall Open Systems Interconnection (OSI) reference model. Medium Access Control (MAC) sublayer is a data link layer function similar to that in Ethernet, which enables multiple appliances in a wireless LAN to share a common transmission medium via a carrier sense protocol. Figure 1.2 shows a generic carrier sense protocol known as Carrier Sense Multiple Access (CSMA). Wireless networks handle error control by having each station check incoming data for altered bits. If the destination station does not detect errors, it

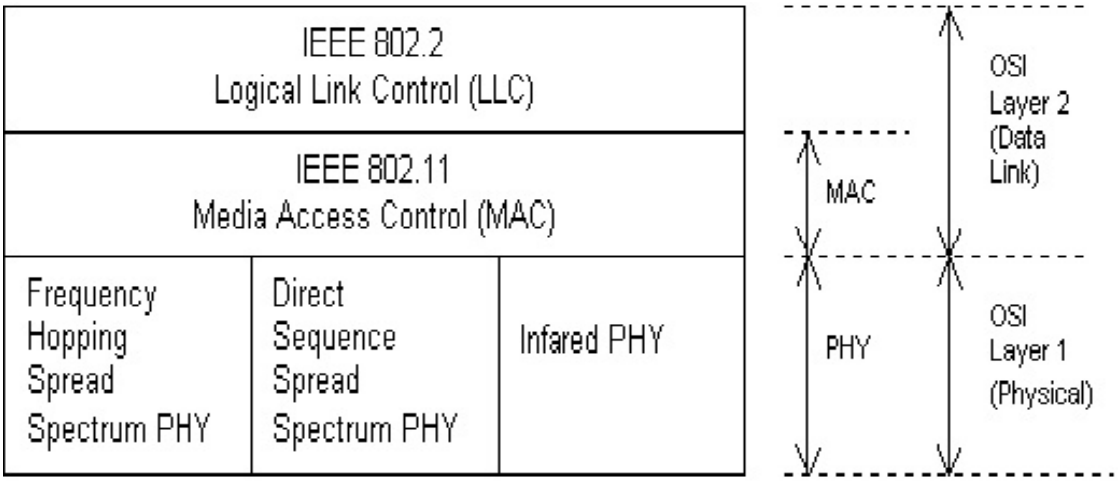

Figure 1.1 wireless in OSI.

sends an acknowledgement back to the source station. If the station detects errors, the data link protocol ensures that the source station resends the packet.

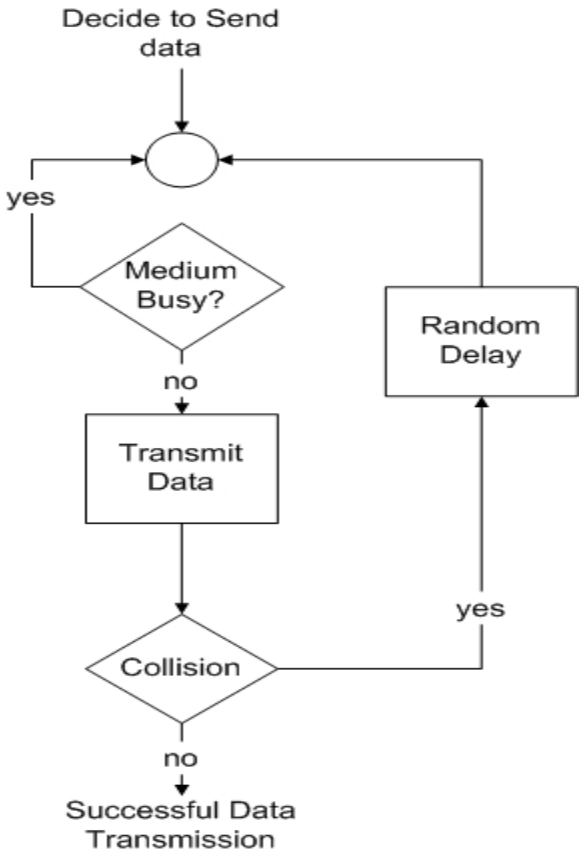

Figure 1.2 CSMA in wireless LANs.

1.1.1 Wireless LAN Components

The major components of a wireless LAN are a wireless Network Interface Card (NIC) and a wireless bridge which is often referred to as an access point. A wireless NIC, just as an Ethernet adapter connects a PC to a wired network, connects a mobile device to a wireless network. It takes the digital data from the host machine, converts it into signals that can be modulated into radio signals, and transmits these signals into the common medium, air. The receiving machine should use such a wireless NIC to receive these radio signals, demodulate them and convert them back to digital form. Wireless NICs come in various interfaces such as PCMCIA, USB, RS232 and parallel port, etc. Bridges, on the other hand, are devices that connect multiple LANs at the MAC layer to produce a single logical network. Bridges act as interfacing devices between Ethernet and Ethernet or between Ethernet and token ring, etc. A bridge that acts as an interface between a wired and a wireless network is referred to as a wireless bridge or as an access point (AP). The other common wireless component that one can find in a wireless LAN is an Antenna. Antennas can come in many shapes and sizes and have certain electrical characteristics like propagation pattern, gain, transmit power and bandwidth. These are generally used for connecting LAN segments across buildings separated by short range distances.

1.1.2 Wireless LAN Architecture

In this section let us explore the different types of wireless LAN architectures one can build with the components discussed in the above section. The two basic architectures (referred to as topologies from now on) are Peer-to-Peer Wireless LANs and Multiple Cell Wireless LANs. The components that are required to build a Peer-to-Peer wireless

LAN (Ad-Hoc Mode) are just wireless NICs. Wireless NICs are connected to host nodes which should be within a range of 75 to 300 feet [15]. This area which is covered by Peer-to-Peer wireless LAN is generally referred to as Basic Service Area (BSA). A single BSA can support between 6-25 users and still keep network access delays at an acceptable level.

On the other hand a Multiple Cell Wireless LAN (Infrastructure Mode) consists of both wireless NICs and access points. The access points are generally connected to a common wired backbone forming multiple cells, which allow a mobile node with a wireless NIC to roam within the coverage area with seamless network connection. Figures 1.2 and 1.3 show the Ad-hoc mode and the Infrastructure mode respectively.

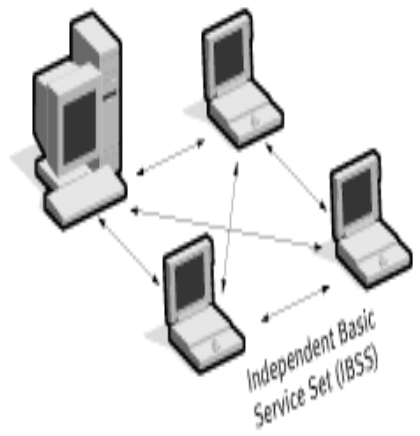

Figure 1.3 Ad-hoc mode

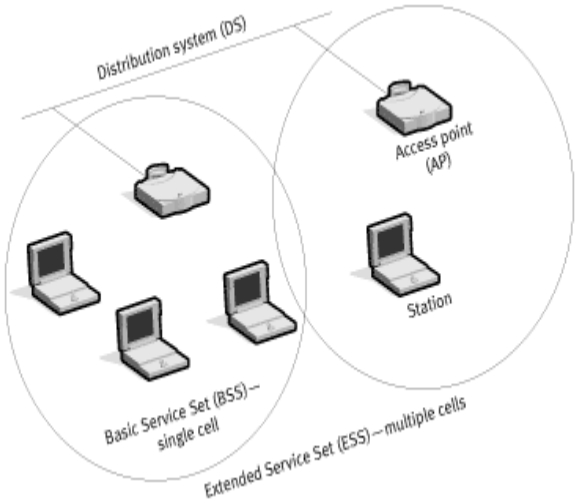

Figure 1.4 Infrastructure mode

An Infrastructure Mode where there is only one cell with one access point installed on the network is referred to as Basic Service Set (BSS) and where there are multiple cells with multiple access points installed is referred to as Extended Service Set (ESS). Similarly an

Ad-Hoc mode is also referred to Independent Basic Service Set (IBSS). There are other types of wireless networks, which do not fall into the 802.11 category like Wireless Metropolitan Area Networks, Radio-Based Wireless Point-to-Point Networks and Laser-Based Wireless Point-to-Point Networks, etc., which fall beyond the scope of this study.

1.2 Security in Wireless LANs

Security has always been a concern in communication networks as it is in many other areas. The important typical security issues that one should consider in the context of data networks are threats to the physical network, unauthorized access to network resources, and both internal and external attacks like Denial of Service (DoS). In the context of wired LANs, the solutions to the above issues are well defined and significantly reliable. The same approaches, however, cannot be adapted to wireless LANs due to the characteristics of the radio signal. The difference between a wired LAN and a wireless LAN is the absence of physical medium in the case of wireless LANs. In a wired LAN any node can receive or transmit data only if the node has physical access to the medium. But in the case of a wireless LAN any node can both transmit data to and receive data from another node as long as each node remains within a certain range where the signals of the other node can propagate. The absence of a physical medium and the fact that a node that wants to participate in a wireless network can be anywhere within a certain range leave wireless networks highly vulnerable to attacks like eavesdropping, unauthorized access, DoS and hacking.

1.3 Project Contribution

In this project we propose a new prototype to minimize the threats on wireless LANs from the vulnerabilities discussed above. The main objective of the project is to provide

the network administrators with an easily maintainable wireless network while at the same time making the wireless network as secure as possible.

The rest of the report is organized as follows. In Chapter 2 we will discuss the existing security model in wireless LANs and its weakness that render a wireless network vulnerable to attacks. In Chapter 3 we will discuss the design issues of our Agent Based Intrusion Detection and Response System and in Chapter 4 we will discuss how the system is implemented and the test bed on which it is implemented . In Chapter 5 we will discuss some of the advantages and disadvantages of the system and future extensions that can be made to this system.

Chapter 2

Background

Because of the limited nature of the physical security in wireless LANs, it is difficult to prevent unauthorized access, intrusion and eavesdropping. The IEEE 802.11 standard [8] defines authentication services to control LAN access to bring the security level in wireless LANs closer to the security level in wired LANs. Every node in a wireless LAN whether part of an IBSS or an ESS, must use the authentication service prior to establishing a connection with another station with which it will communicate. The 802.11 standard describes two authentication services, Open System Authentication and Shared Key Authentication both of which are meant to increase the security in wireless

networks. Open System Authentication is the simple and default authentication type that is set when a wireless LAN is setup. In Open System Authentication the station to be authenticated sends an authentication management frame with the SSID of the network to the authenticating stations. The authenticating station then sends back a frame indicating whether it recognizes the identity (SSID field) of the authenticating station. Figure 2.1 shows the authentication process in an Open System Authentication.

Shared Key Authentication on the other hand is an optional approach that provides a higher degree of security when compared to that of the Open System Authentication. In

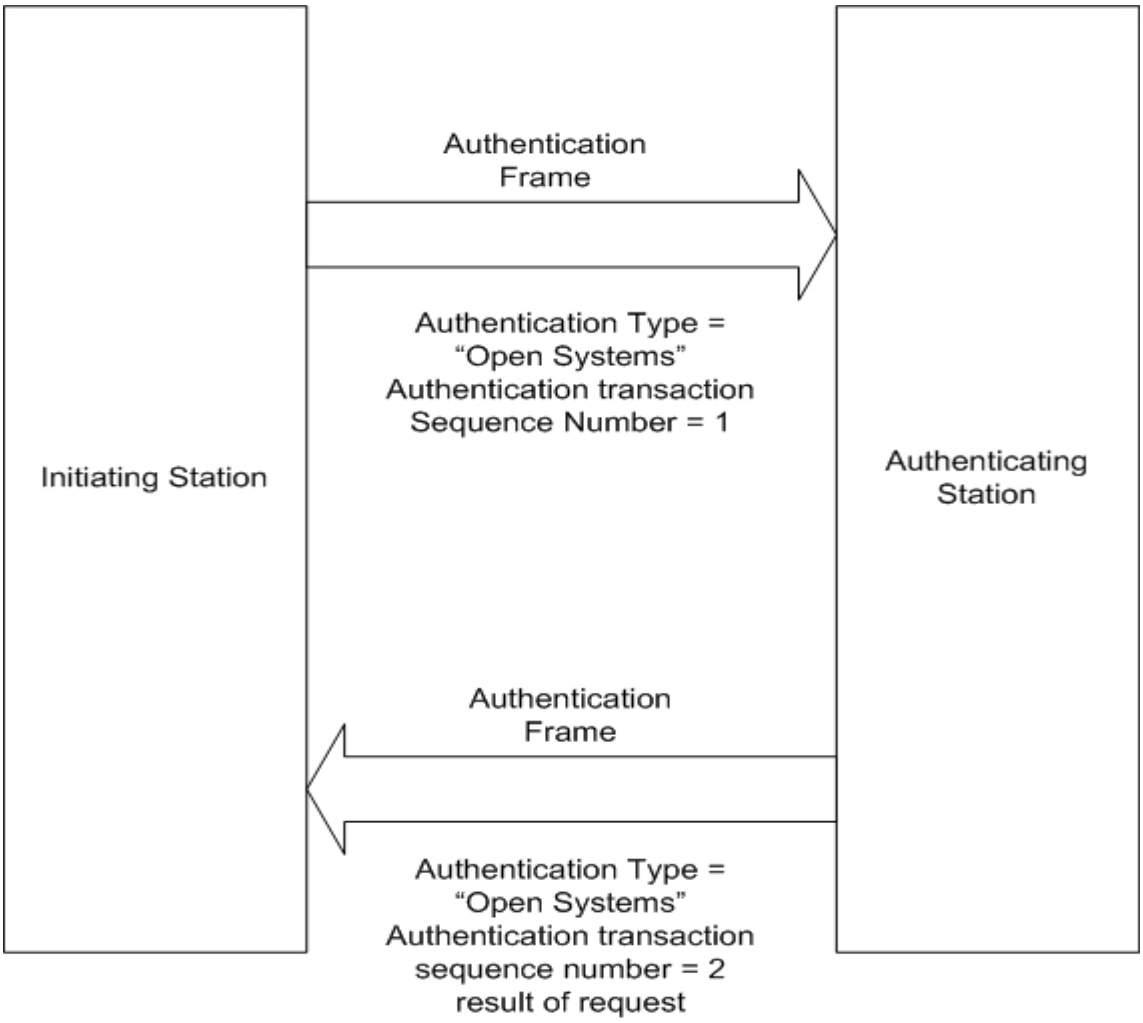

Figure 2.1 Open Systems Authentication.

Shared Key Authentication it is assumed that each station has a secret shared key acquired through a secure channel prior to participating in the wireless network. Both the requesting and the authenticating stations should use the shared key to communicate with each other. Figure 2.2 illustrates the shared key authentication process.

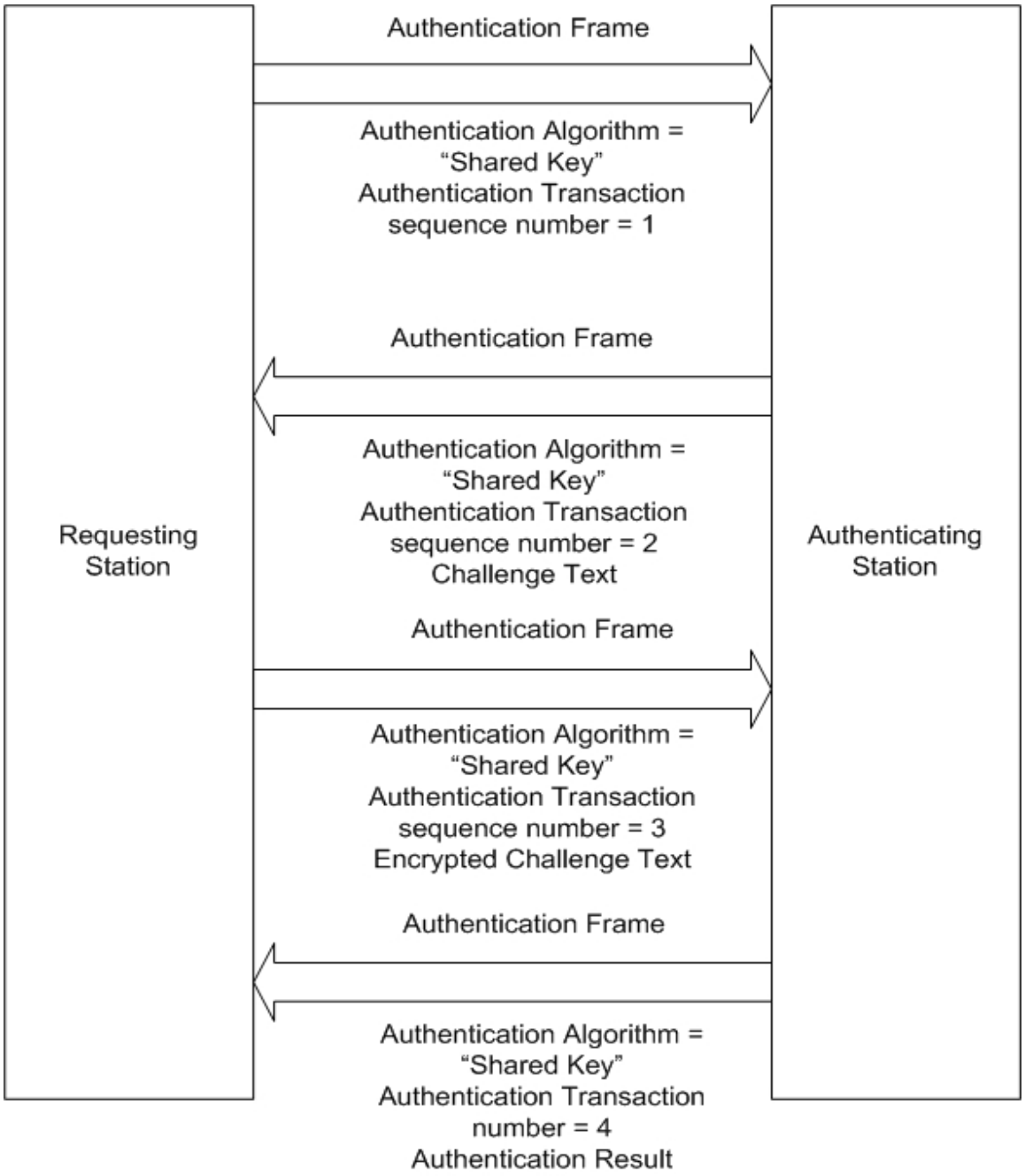

Figure 2.2 Shared Key Authentication.

The requesting station sends an authentication request frame to the authenticating station. When the authenticating station receives the initial authentication frame, the station will reply with an authentication frame containing a challenge text generated using an encryption algorithm – Wired Equivalent Privacy – and the shared key which is already know to both the communicating stations. The requesting station (station which is requesting the authentication) after receiving the challenge text, encrypts it with its shared key and sends back to the authenticating station. The authenticating station receives the encrypted challenge text, decrypts it and compares the decrypted challenge text with the original text it sent out to the requesting station. Depending on if a match occurred or not, the authenticating station sends the results back to the requesting station.

2.1 WEP

Wired Equivalent Privacy (WEP) is the security protocol provided in the 802.11 [8] wireless standard to bring the security in wireless networks comparable to the security in wired networks. Its main objective is to protect the data that is transmitted over wireless links from malicious eavesdroppers. WEP tries to accomplish its goal by encrypting the data that is being transmitted at the MAC layer. We provide a brief description on how WEP works in the following section.

WEP, as mentioned earlier, uses a secret shared key that all the communicating stations share prior to participating in the network. Let us consider a scenario where station A wants to communicate with station B. Let M be the message that A wants to send to B. The following steps describes the WEP algorithm.

1. A computes the integrity check i(M) and concatenates the result with the given message M obtaining P = <M, i(M)>

2. A chooses an Initialization Vector (IV)

3. Using RC4 algorithm the IV in step 2 and the shared key as the inputs, A generates a pseudorandom key stream

$$\text{Pseudorandom key stream} = \text{RC4(IV, shared key)}$$

4. We then perform an exclusive-or between the P in step 1 and the Pseudorandom key stream in step 3 to obtain the cipher text C.

$$C = P \oplus \text{Pseudorandom key stream}$$

5. Finally, A transmits the cipher text C along with the Initialization Vector (IV) in step 2 to B

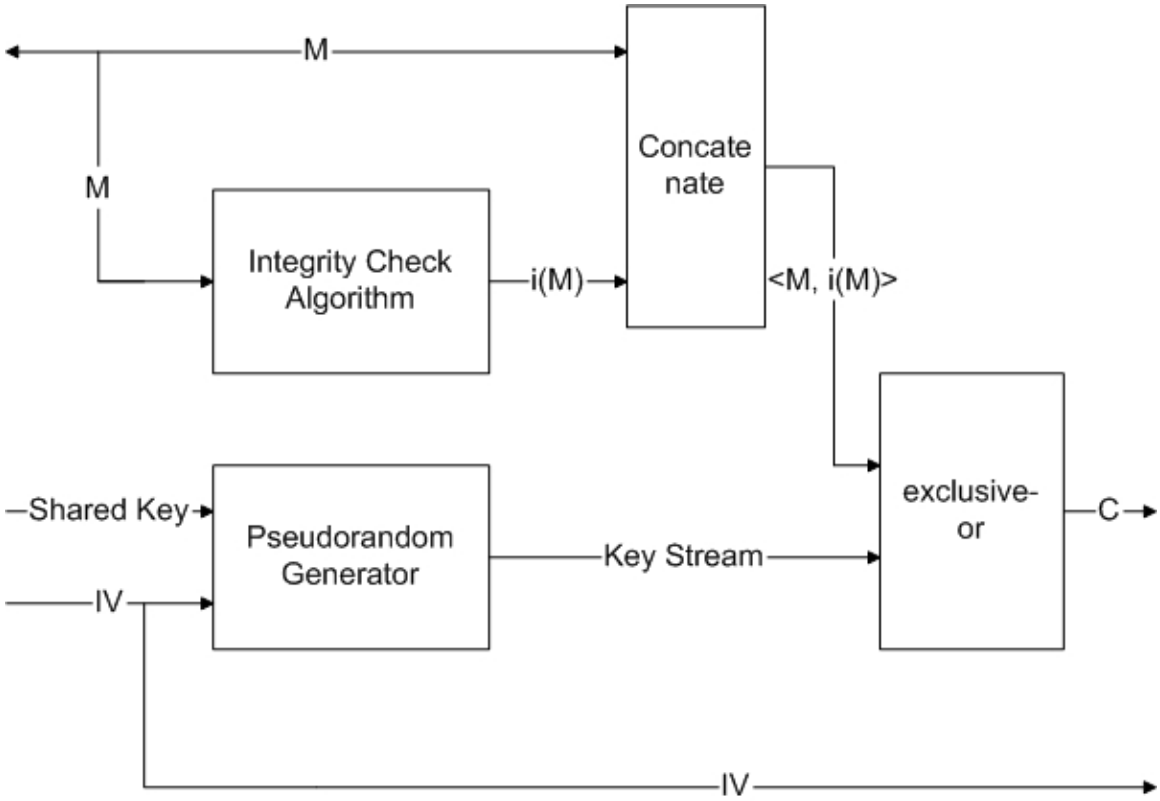

Figure 2.3 WEP at Station A.

B on receiving the cipher text C and the Initialization Vector (IV) goes through the following steps

1. Calculate the same pseudorandom key stream using the IV received from A and its shared key
2. Exclusive-or the key stream obtained in step 1 with the received cipher text to obtain the plain text P

   $$P' = C \oplus \text{pseudorandom key stream}$$

   $$= (P \oplus \text{Pseudorandom key stream}) \oplus \text{pseudorandom key stream}$$

   $$= P$$

3. B splits P' into <M' , i(M)>
4. Calculate the integrity check i(M') on M'
5. Accept the frame if the integrity check i(M) = i(M') otherwise reject the frame

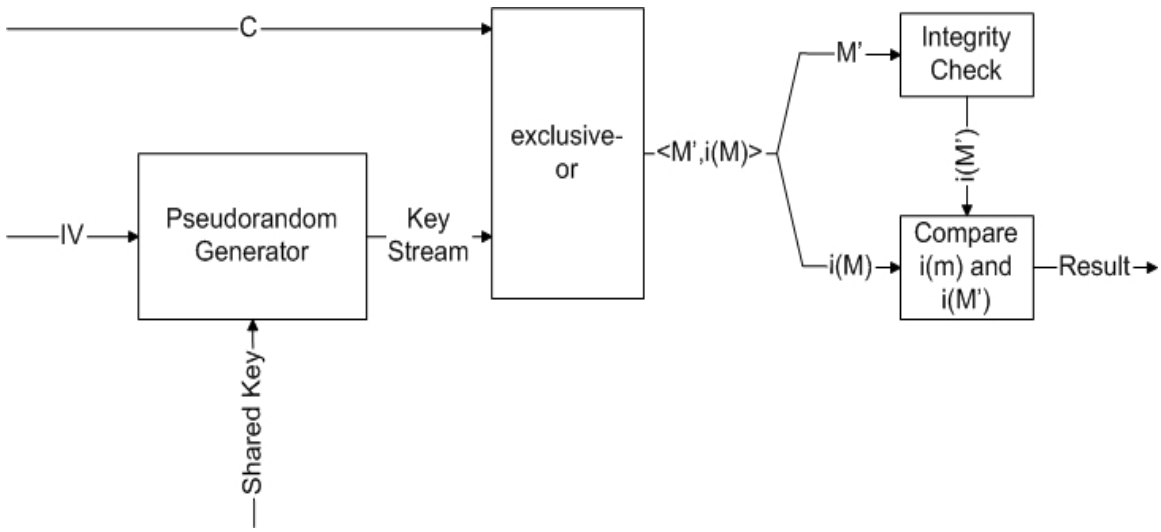

Figure 2.4 WEP at station B.

The main objective of the WEP protocol is to enforce data confidentiality, access control and data integrity [8]. Data confidentiality is aimed at preventing the network from eavesdropping. Access Control is needed to prevent unauthorized users from accessing the network resources. And Data integrity is meant to protect the data from being modified during transmission. It is claimed that WEP provides all the above three features making wireless networks as secured as their wired counterparts. However, recent studies [1,3,4] proved that none of the three security goals, data confidentiality, access control and data integrity, are delivered with WEP. In [2] the authors show how one could recover the shared keys in a wireless network with WEP enabled by passively sniffing the wireless network. To make things worse, there are freely available software (e.g., Airsnort [10]) to automatically recover the shared keys in a wireless network even if WEP is enabled.

In this work we propose and develop a prototype that detects intrusions made by unauthorized elements and prevents them up to a limited extent and provides Access Control and Data Integrity. Rather than proposing any changes to the WEP protocol, we try to utilize and combine already existing techniques which provide network security in our solution.

2.2 Related Work

In this section we will see some of the relevant tools that already exist that are being used to enhance the security in wireless networks. In order to determine whether a given wireless network is vulnerable or not, one should be able to spot such points where security can be easily compromised. Some of the undesirable practices that render a wireless network vulnerable to attacks are not detecting rogue access points (a rogue

access point is one that is installed on a network without the consent of the personnel in charge of the network) on the network, not enabling WEP, etc. Netstumbler [5], AirTraf [7], AP Tools [11], Kismet [6] and Dstumbler [12] are existing tools that aim at detecting the presence of access points and their configuration like the status of WEP, signal strength, etc.

Netstumbler is a windows utility for "wardriving" – a brute force attack performed by driving around with wireless gear, scanning for unsecured wireless networks to gain illicit network access[6]. Figure 2.5 shows a screenshot of this tool.

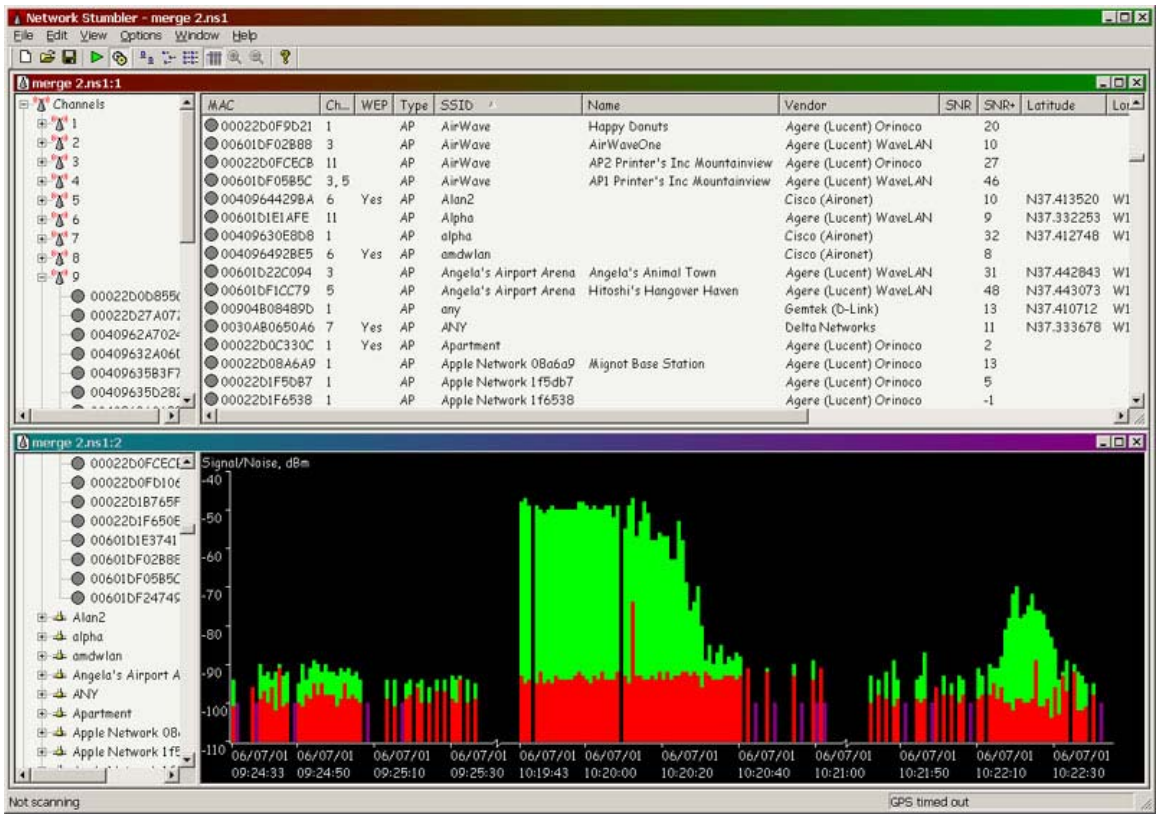

Figure 2.5 Netstumbler.

As shown above, Netstumbler is a very convenient tool that scans for the MAC address of the access point, SSID of the access point, signal strength, status of WEP, the channel

being used etc. It also plots the SNR(Signal/Noise Ratio) graph as shown. One of the disadvantages with this tool and some of the other tools mentioned above is that they are not flexible to use in a corporate environment since they do not keep scanning the wireless networks round the clock. To scan a network, one has to physically move around the network with a wireless gear. Another disadvantage is that, rogue access points are not distinguished from the legitimate access points. All the scanned access points will be listed and it is up to the user to identify the rouge access points.

APTools is a Unix utility that queries ARP Tables and Content-Addressable Memory (CAM) for MAC Address ranges associated with 802.11b Access Points in a Virtual LAN (VLAN). It utilizes Cisco Discovery Protocol (CDP) to propagate through a VLAN. This tool, however, is very focused and relies upon the Cisco Products for propagation.

AirTraf is a very sophisticated tool with a client-server architecture that does more than just scanning a network for access points. As in our model, it has two modules, sniffer server and polling server. The sniffer servers and the polling server can communicate to exchange information about a wireless cell. The best feature in this tool is that the information that is gathered from the sniffer servers can be stored in a database and viewed through web pages. It also performs host-based protocol analysis which is very helpful in recording the bandwidth utilization for individual nodes. Figures 2.6a and 2.6b show the GUI for this tool which are used of scanning a wireless cell and analyzing the bandwidth used by individual hosts. Figure 2.6a shows the sreenshot of the tool while it is analyzing for the available access points. In Figure 2.6b the statistics such as number of TCP connections, the amount of uplink and downlink traffic for each node, signal

strength at the node, etc, are shown. Any network administrator who is planning to maintain a wireless network should know about this tool.

Figure 2.6a AirTraf – Scanning Access Points.

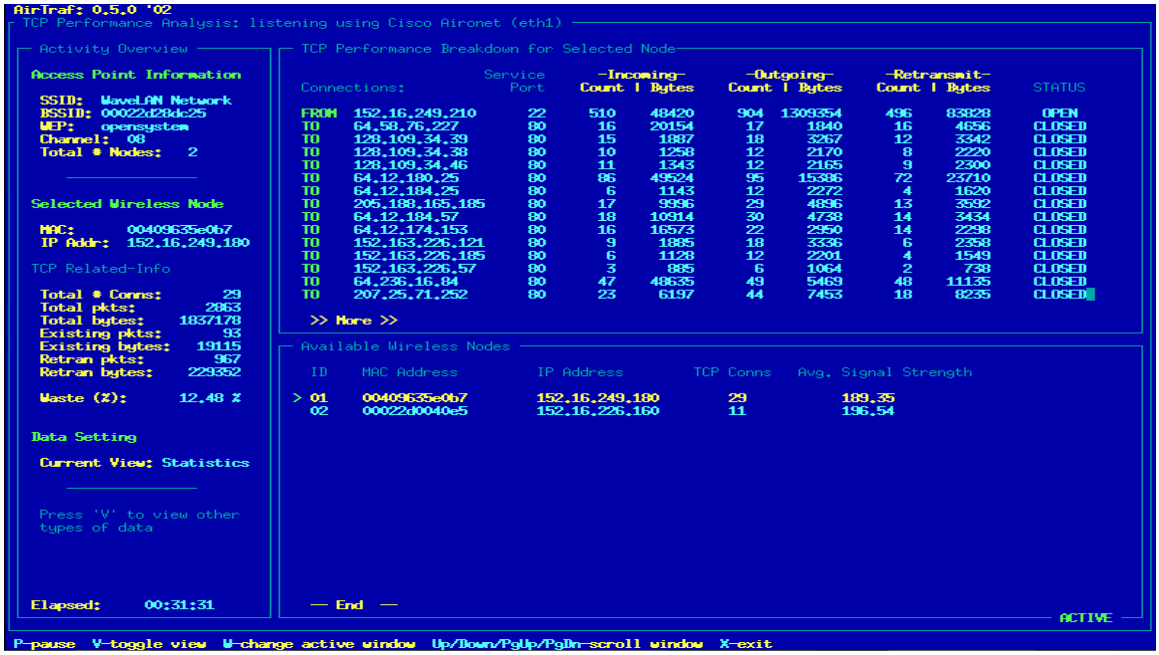

Figure 2.6b AirTraf – Analysing the traffic for each client.

Kismet and Dstumbler are 802.11b network sniffers and do almost the same job of sniffing for available wireless networks (or access points). Like APTools, Kismet can use CDP in a Cisco environment to propagate through a VLAN. Dstumbler is a Unix version of Netstumbler without the plotting facility.

2.3 Our Project

In our project we developed a tool, Agent Based Intrusion Detection and Response System (ABIDERS), that not only scans for access points installed on a network but also acts as an intrusion detector. Unlike other tools, it stops unauthorized access by preventing unknown users from accessing the network resources. Another feature that is not present in any of the other tools except AirTraf is its flexibility to gather data from a central location. With this option, one need not physically move around with a wireless node scanning the networks. Keeping in view the observation made in [13], that the performance when employing Virtual Private Network (VPN) is more than that when employing WEP, the support for IPSec tunneling (for building VPNs) has been built in to the system.

The main focus is on intrusion detection in real time rather than monitoring the network first and then analyzing manually for possible intrusions. The agents in this system keep scanning their respective cells all the time for rouge access points, unauthorized elements and promiscuous nodes. Each time a rogue access point or a promiscuous node is seen, the corresponding agent will send an email to the concerned network administrator with information such as the geographic location, the name of the cell, time of intrusion etc. Table 2.1 compares all the tools discussed above to clearly identify the differences between each tool.

	N	APT	AT	K	D	ABIDERS
Client/Server Model			X			X
Scan Access Points	X	X	X	X	X	X
Identify Rogue Access Points						X
Prevent Intrusions by unauthorized wireless nodes						X
Detect Promiscuous nodes						X
Host-Based Traffic Analysis			X			
VPN Support						X
Database Support			X			

Table 2.1 Comparison of Tools

N - Netstumbler

APT - APTools

AT - AirTraf

K - Kismet

D - DStumbler

ABIDERS - Agent Based Intrusion Detection and Response System

Chapter 3

Design

In this chapter we discuss the design and architecture of the system. The system can be deployed in both BSS and ESS (see chapter 1) without any modifications. In a corporate environment one has to deploy an ESS with multiple access points located at various locations to have a reasonable amount of wireless coverage. So given an area where a wireless network has to be deployed, it should first be logically divided into cells as shown in Fig 3.1. To have a wireless connectivity in a cell, an access point should be installed in that particular cell. So in a conventional wireless network we have two backbones, wired and wireless. The wired backbone is the already existing wired network and the wireless backbone is the backbone that connects all the access points in all the

logical cells. With this setup, wireless connectivity can be provided throughout a campus where wireless users can walk from one cell to another.

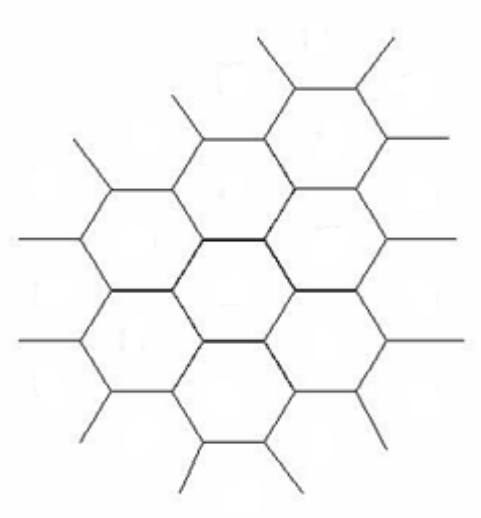

Figure 3.1 An area divided into logical cells.

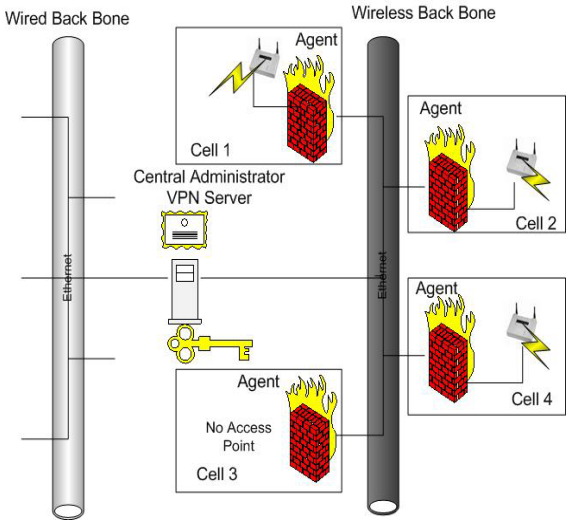

Figure 3.2 Overview of the Architecture.

In the proposed solution we have two major entities namely an Agent and a Central Administrator (CA). An agent is a desktop and is placed along with an access point in a given cell. Unlike a conventional installation of an access point on a wired backbone, in our solution an access point is connected to the wireless backbone via the agent and this wireless backbone is bridged to the existing wired backbone via the Central Administrator as shown in Figure 3.2. The Central Administrator is the main entity that communicates and controls the agents. It can also act as a VPN server to build VPN tunnels between the CA and the mobile nodes thus making the wireless network more secured.

The agents, in the absence of access points, act as sniffers, scanning for rogue access points and promiscuous nodes in their respective cells. In the case of an access point

being present, the agents act as both sniffers to detect unauthorized wireless elements and firewalls to stop unauthorized users from accessing the network resources. In the following sections we explore the details of the system by closely examining the roles played by the agents and the Central Administrator.

3.1 Central Administrator

The Central Administrator is the main central server that acts as the centralized entity to monitor all the wireless cells in the network. It maintains the information of all the access points, agents and users of the wireless network. Should a new access point or wireless client be installed on the network, it has to be registered with the central administrator; otherwise, it would be considered as an unauthorized element. The Central Administrator maintains a record for each access point consisting of information about the geographical location it is serving, its MAC address, the name and IP address of the agent present in that cell, status of WEP etc. This information can help network administrators in making strategic decisions in strengthening the security of the network. In addition to this information it also maintains critical information about the client card like its MAC address and the public key. The following are the functions offered by the central administrator.

- Registering and unregistering access points (AP), client cards, and agents;
- Scanning the network;
- Generating x.509 certificates for client cards; and
- Acting as a VPN server for the whole wireless network.

When a wireless element like an access point or a wireless client card is registered, the central administrator will forward the element's information to all the agents. The agents upon receiving this information act accordingly which is discussed in detail later in section 3.3. But, registering a client card involves an extra action other than just sending the information to the agents. When a client card is registrered, an x.509 certificate is generated using the information like the MAC address, user name, email address, department etc. This certificate can be used by the wireless client to build a secured tunnel (using IPSec [14]) between the central administrator and the wireless node as shown in Figure 3.3.

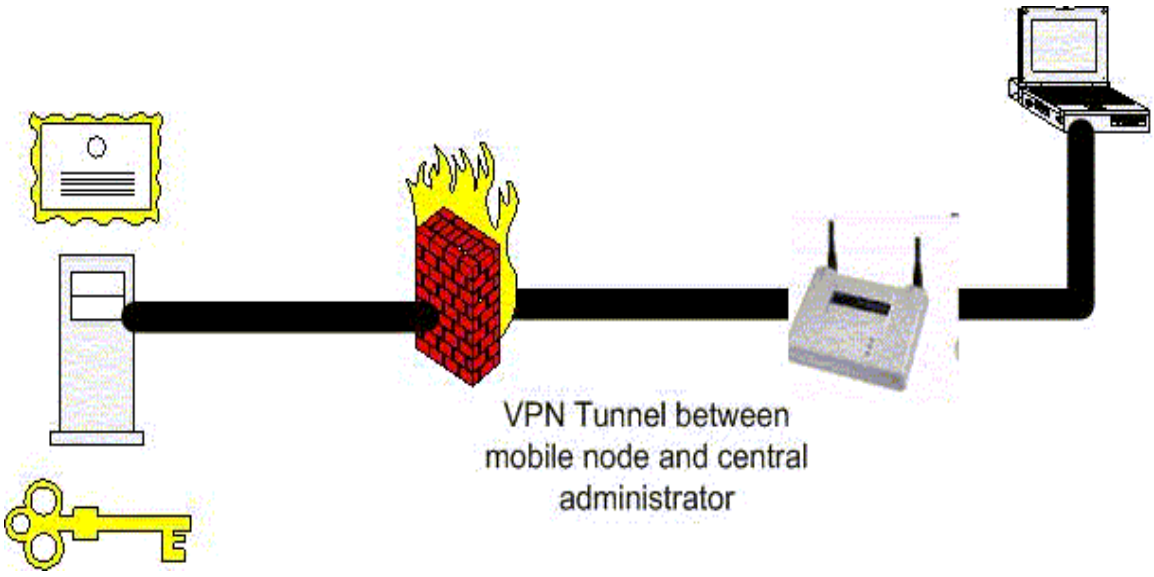

Figure 3.3 A VPN Tunnel between a mobile node and the Central Administrator.

When an agent is registered, the central administrator will add the agent's information, name, IP address and geographical location to its list of other agents' records. This information is used for communicating with the agents. A central administrator can unregister an agent by removing the agent's record from its list of agents.

3.2 Agents

Agents are the main building blocks on which the whole system relies. It is the job of the agents to detect unauthorized wireless elements like rogue access points and promiscuous wireless nodes. These agents also act as the first line of defense to protect the network from unauthorized use of the network resources.

Each agent will be equipped with three network interface cards, one of which is a wireless interface. This interface is necessary in order to sniff and detect rogue access points. The other two are normal ethernet cards which are used to connect the access point to the backbone (Fig 3.2). The following are the functions of agents

- Accept registration and unregistration messages for APs and client cards from the Central Administrator;
- Keep scanning the cell for rogue access points (Those access points that are not registered with the central administrator) and notify the concerned personnel if found;
- Detect promiscuous wireless nodes; and
- Block unauthorized users (those who did not get their card registered with the central administrator).

Each agent maintains a list of registered APs. When a new AP is registered with the central administrator, the central administrator will send the MAC address of the AP to all the agents. The agents, upon receiving the registration, will update their list of authorized APs by adding the new MAC address to their list. Similarly when a message to unregister an access point is received from the Central Administrator, the corresponding MAC address would be removed from the list of trusted access points. In

case of a client card, upon receiving a registration or unregistration message from the Central Administrator, the agents update their firewall policy to allow or block the user depending on the MAC address of the client card. Some might argue that vendors incorporate MAC filtering into their APs. But the main challenge lies in maintaining the list of such allowable MAC addresses in all the APs present on the backbone. There is no centralized solution that can update the list efficiently on all the APs in a heterogeneous environment with cross-vendor products. Using our solution, we register the client card once at the central administrator thus making it easy to maintain the list.

The other two important activities of the agents are (1) scanning their respective cells for rogue APs and (2) scanning for wireless promiscuous nodes. The agents scan for rogue APs by sniffing for beacon frames emitted by the APs. Each time a beacon is sniffed, the agent compares the source MAC address of the frame with the list of registered access points. If it is not found in the list of registered users, the agent considers the AP to be a rogue one and sends an email notification to the concerned personnel. The agents also maintain the list of unregistered APs they find in order to make sure that they do not start a Denial of Service (DoS) attack by continuously sending out emails for each beacon found.

For detecting promiscuous wireless nodes, the agents use ARP Spoofing [9] to fool the promiscuous nodes. The Agents broadcast fake ARP messages, for all the IP addresses in the subnet that the agent operates in, with a fake broadcast address (address that is not a true broadcast address but will be considered as broadcast address [9]). Should a promiscuous node be present in their respective cells, it will respond back with an ARP response (refer to [9] for details). Once the node is detected by its ARP response, the

information, like the geographical location, the IP address, and time when the node was seen, will be passed on to the administrator. In the section 3.6 we will see the design details of the above described functionalities.

3.3 Registration and Unregistration of APs

Figure 3.4 shows the registration process of an access point. In order to register an access point, information such as MAC address of the access point is necessary. Apart from the MAC information, other information such as the cell in which the access point would be installed has to be choosen.

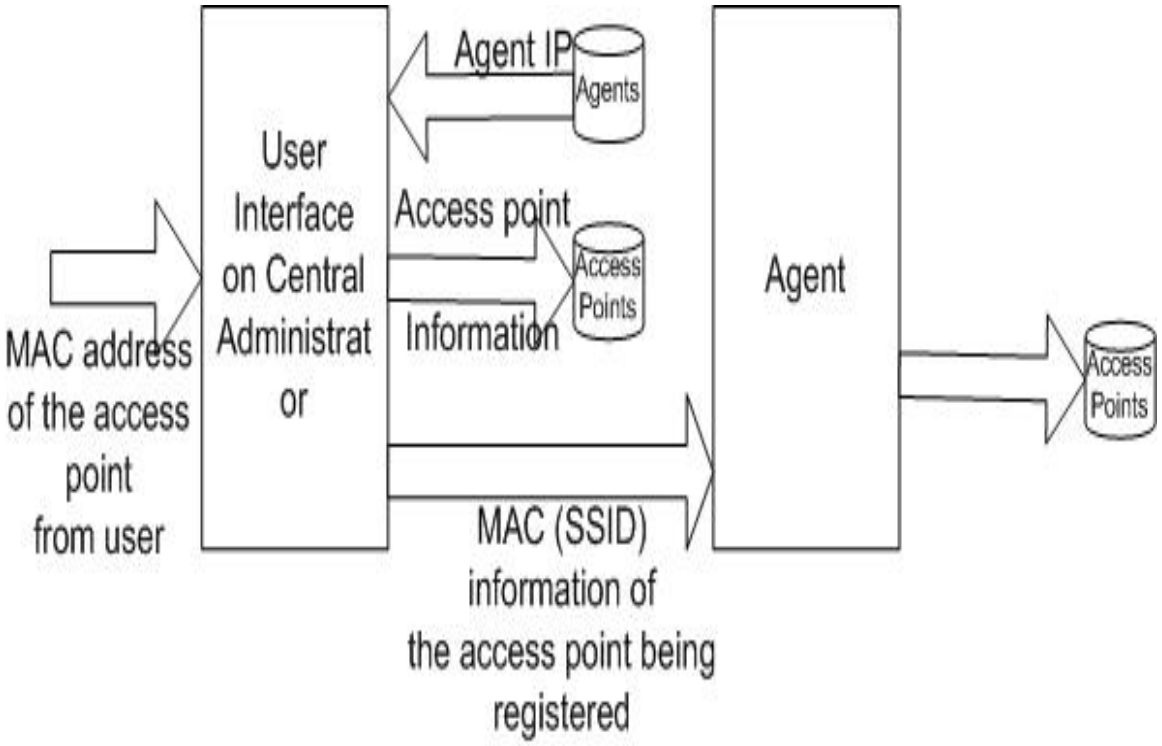

Figure 3.4 Access Point registration process.

The CA forwards the MAC address to all the agents on the wireless backbone. Once the agents receive this information they will store the MAC address of the AP sent by the CA

in their list of registered access points as shown in Fig 3.4. Likewise, when an access point is unregistered, the central administrator will instruct all the agents to the delete the corresponding information from their lists. The reason that the CA sends the information to all the agents instead of just to the concerned agent is that, the cells might overlap and should there be an agent located in the overlapped area, that agent would consider the access point in the other cell as an unregistered one even though it is a registered access point. To avoid this ambiguity, the MAC address of any access point that is being registered is sent to all the agents.

3.4 Registration and Unregistration of Clients

Figure 3.5 illustrates how the client cards are registered. In order to register a client, information such as user name, the user's department, user's e-mail address, and geographical information such as city and state and the MAC address of the user's wireless card is required.

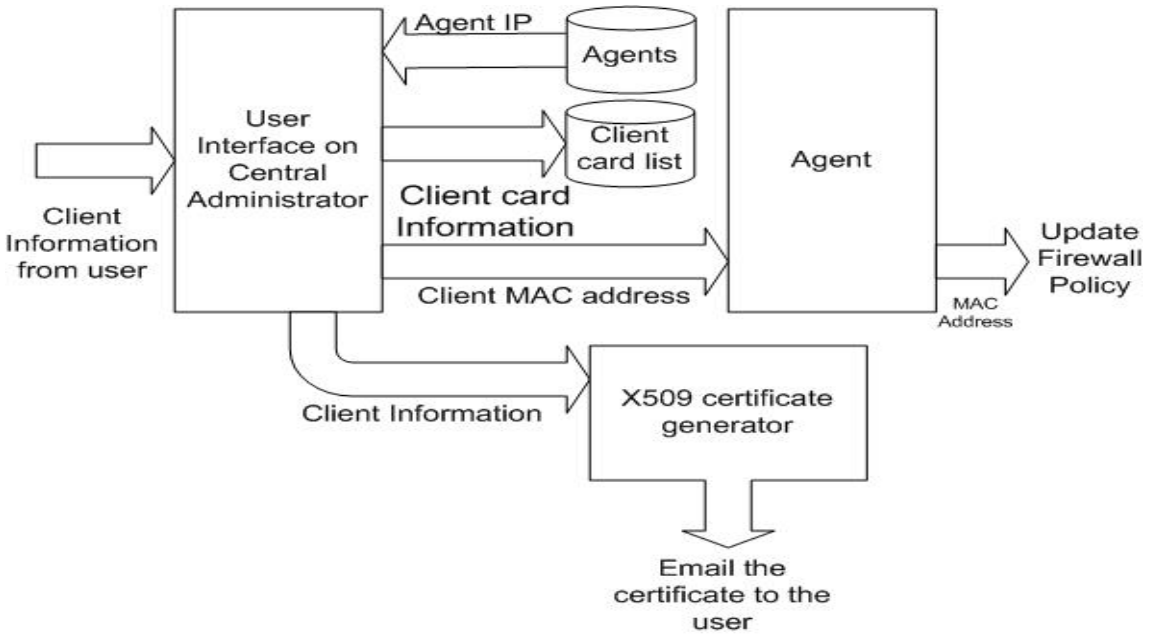

Figure 3.5 Client registration process.

Once the information is submitted, an X.509 [17] certificate will be emailed to the user of the card. This certificate can be used in implementing a secured tunnel between the user's mobile node and the central administrator. A copy of the user information is stored in a local flat file and sent to the agents. The agents upon receiving this information update their firewall policies, thus allowing the user to use the network resources. For unregistering a card, the central administrator instructs the agents to update their firewall policies to block the traffic from the user's wireless card.

3.5 Registration and Unregistration of Agents

Figure 3.6 shows the agent registration process for registering agents. For registering an agent information such as the name of the agent, the IP address of the agent, the cell in which the agent would be serving and the location information. Once the information corresponding to a particular agent is submitted, it will be stored in a local flat file and used when registering access points and client cards. Similarly when an agent is unregistered, the corresponding entry will be deleted from the list of agents.

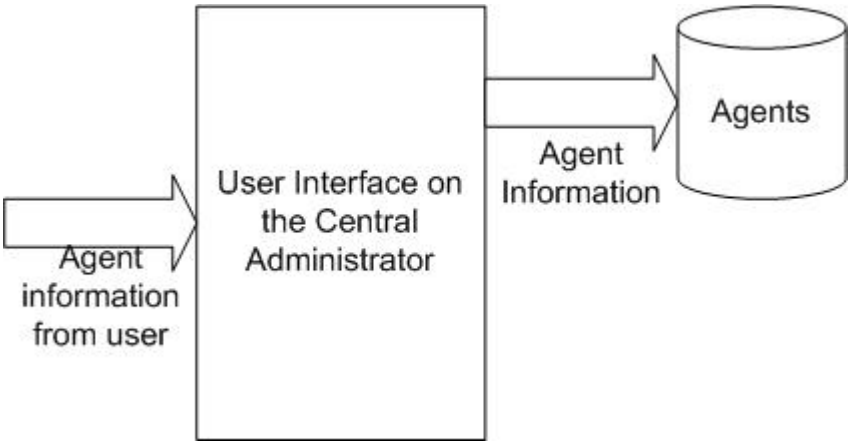

Figure 3.6 Agent registration process.

3.6 Detection of Rogue Access points and Promiscuous nodes

The following two figures (Figures 3.7 and 3.8) show the flow charts for detecting the rogue access points and the promiscuous nodes respectively.

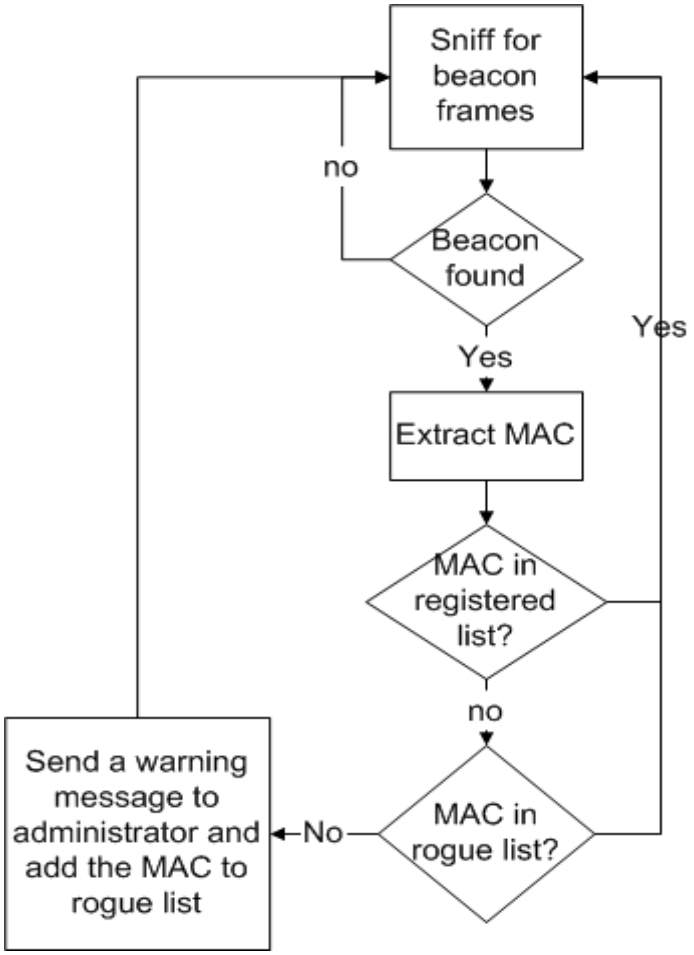

Figure 3.7 Detecting rogue access points.

Figure 3.7 shows how agents scan for rogue access points. The agents sniff for beacon frames using their wireless cards. When a beacon frame is found, they will check the SSID field (MAC address of the access point) of the frame with the existing MAC addresses in the list of registered access points and decide whether the access point in question is a legitimate or rogue one.

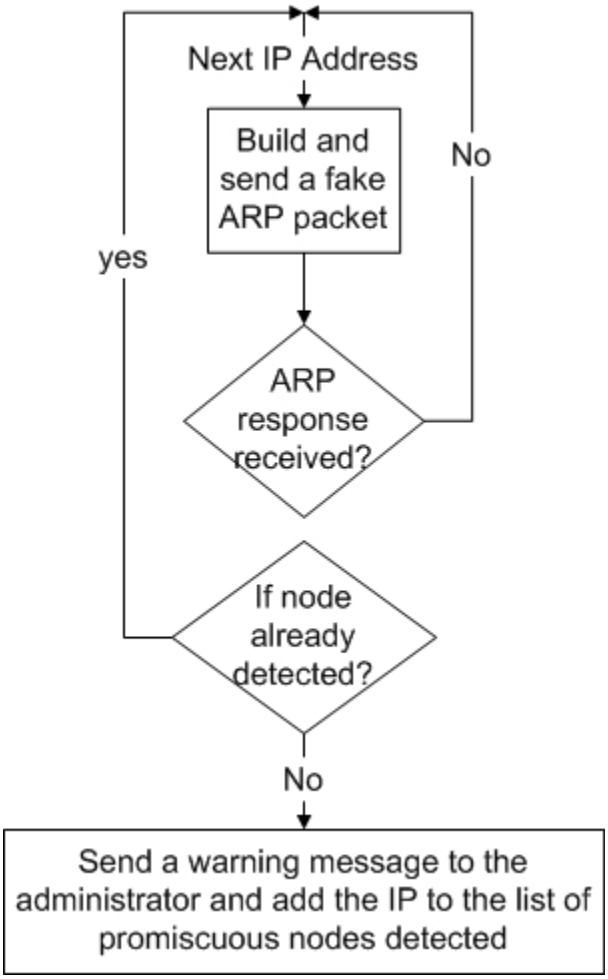

Figure 3.8 Detecting promiscuous nodes.

Figure 3.8 shows how agents would detect promiscuous nodes in their respective cells. For each IP address in the subnet the agent is serving, the agent sends out a fake ARP request. Should the node with that IP address be in promiscuous node, the node's NIC would pass the ARP request to the kernel which in turn would send back an ARP response. The agent upon detecting this ARP response decides whether a given node is in promiscuous mode or not. If the agent already detected the node before it would go to the next IP in the subnet. On the other hand if the node is detected for the first time, the agent

sends a warning message to the administrator and adds the node to its list of detected nodes.

3.7 Scanning Agents and Access Points

One of the most striking features that is available in this system is the ability to monitor the whole wireless network from a single location. The system lists all the access points and all the agents installed on the wireless backbone and the user can choose the access point or the agents he wants to scan. If the user is scanning an access point, the corresponding agent would send back information such as the SSID (MAC address), status of WEP and the channel of communication. If the user is scanning an agent, the agent would send back a list of access points it could see along with the information whether those access points are registered or not. Figure 3.9 shows the block diagram of this process.

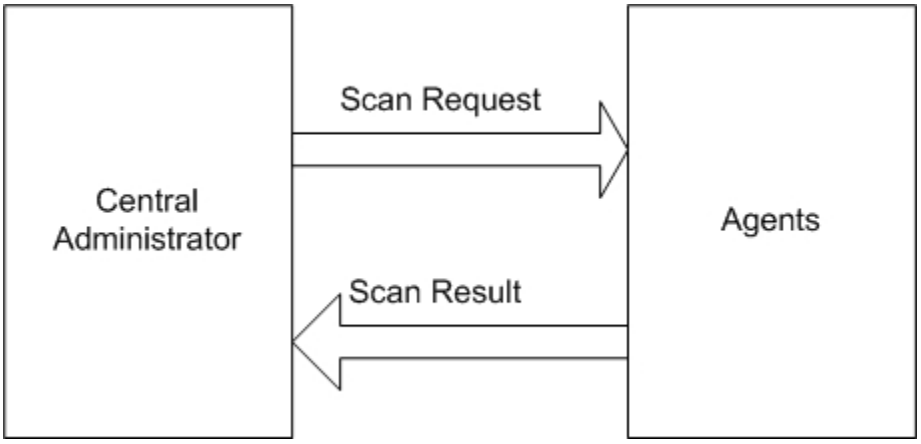

Figure 3.9 Scanning Access Points or Agents.

Chapter 4

Implementation and Test Bed

In this chapter we discuss the hardware and software requirements, implementation details and the test bed on which the system has been set up. Apart from these, we present the Graphical User Interfaces that have been designed to input the required information during the registering or unregistering of the respective wireless elements.

4.1 SOFTWARE REQUIREMENTS

The following are the software requirements for our system.

Operating Systems : Linux

Programming Languages and scripting tools : C, Gtk 2.0, Perl, Expect

Miscellaneous : FreeSwan 1.96 with X.509 Patch, tethereal, libnet, IPtables, Open SSL

The system is a client server model, where the central administrator is the client and the agents are the servers which are distributed across the wireless bank bone. The central administrator provides the Graphical User Interfaces where the network administrator could enter the required information. The central administrator then processes the data as described in the previous chapter and sends the relevant information to the agents. Figures 4.1, 4.2 and 4.3 show the GUIs that are mentioned above and which are used to input the relevant information and communicate with the agents.

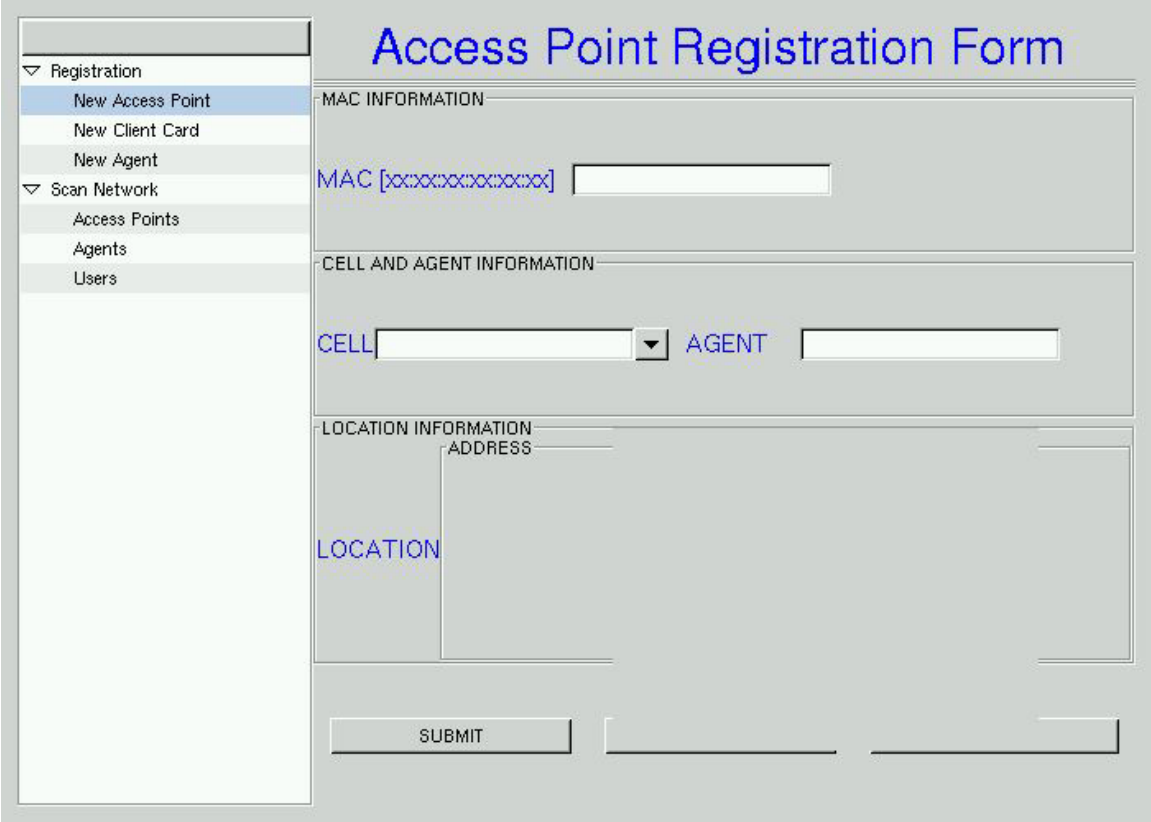

Figure 4.1 Access Point Registration Form.

Figure 4.1 shows the form that is used to register an access point. The MAC field should be filled in with the MAC address of the access point that is being registered. The user can select a cell from the dropdown list where the access point is going to be

installed. Once a cell is selected the corresponding agent serving that cell is shown in the agent field (The user need not type in an agent's name). The location information will also be filled in automatically depending on the cell selected. Once the user has all the information in the form, he/she can click the submit button. When this button is clicked, the MAC address of the information will be sent to all the agents in the following format "AP → xx:xx:xx:xx:xx:xx". The prefix AP in the message indicates to the agents that this is a registration of an AP. The agents upon receiving this information, extracts the MAC address and append it to the list of registered access points.

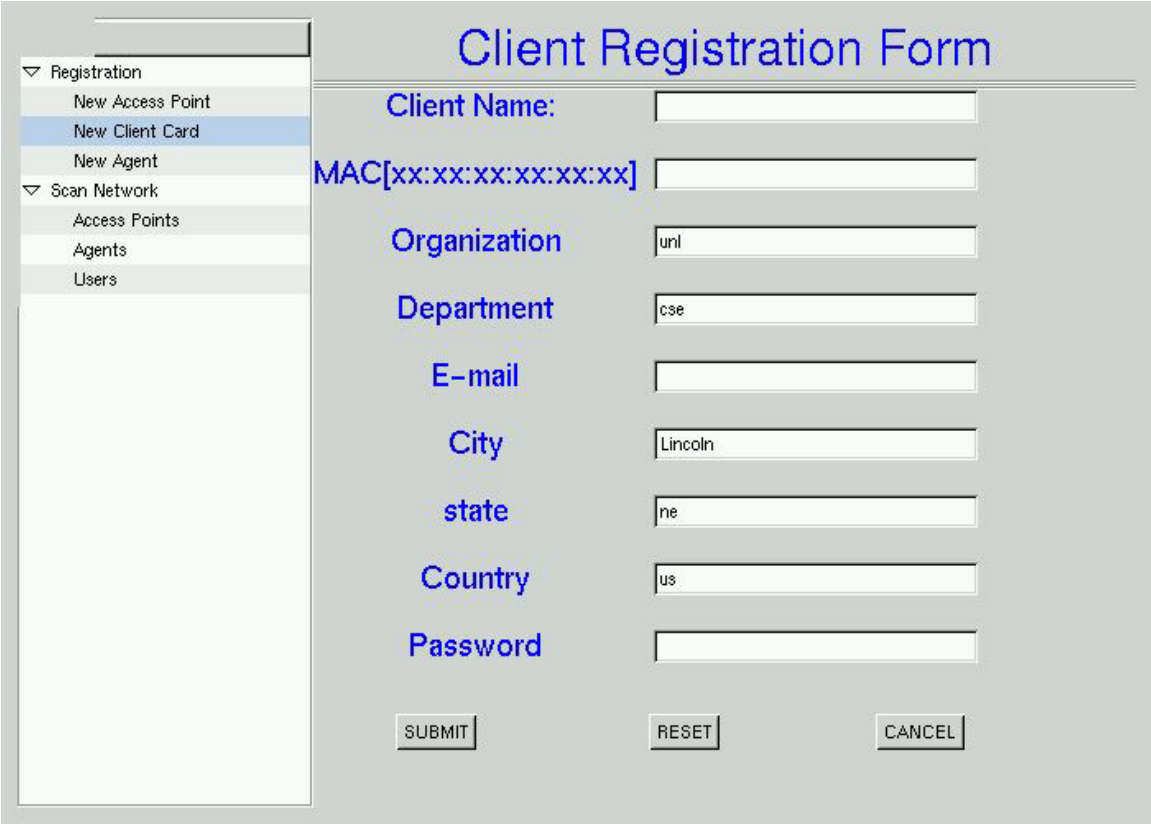

Figure 4.2 Client Registration Form.

Figure 4.2 shows the GUI used to register a client who wishes to use the wireless network. If a user is not registered, he/she would not be considered as a legitimate user

and would be blocked by the agents, which not only act as sniffers but also as firewalls. The agents use IPTables to write the firewall rules. For registering a user, all the fields above should be filled before clicking the submit button. When the submit button is clicked, the MAC address of the client card will be forwarded to all the agents and all the information that is entered will be passed on to an expect script that uses OpenSSL to generate an x.509 certificate (It is necessary to have all the information to generate an x.509 certificate). This certificate will be e-mailed to the user using the e-mail address specified in the form and can be used by the user to setup secure VPN tunnels between the wireless node and the CA as shown in figure 3.3. The second function that triggers when the submit button is clicked is that the MAC address of the client card in the form will be sent to all the agents in the following format "CLIENT → xx:xx:xx:xx:xx:xx". The prefix CLIENT in the format indicates that this is a registration of a client card. The agents upon receiving this information write a new firewall rule as follows

```
iptables -A INPUT -m mac --mac-source xx:xx:xx:xx:xx:xx -j ACCEPT
```

The rule above tells `iptables` [18] to accept any traffic that has xx:xx:xx:xx:xx:xx as the source. When a packet arrives, iptables checks all the rules until a match of the source MAC address is found and then accepts the packet. If a match is not found, the policy of the INPUT table, which is set to DENY, will be executed and the packet would be dropped without allowing it to pass through the agent.

Figure 4.3 shows the GUI used to register an agent. In order to install a new agent in a cell, the first thing to do is to physically install the agent. This means that the agent should be connected to the wireless back bone. Then the form in Fig. 4.3 should be used to register the installed agent.

Figure 4.3 Agent Registration Form.

Out of all the fields only the location information will be passed on to the new agent whose IP address is the one specified in the Agent IP field. The format of the message that is sent to the agent is as follows "AGENT → <address>". The agent upon receiving this information, stores it in a file called "Address" and uses it whenever it sends an email to the network administrator.

The interface in figure 4.4 is used for both unregistering and scanning an access point. The column labeled "access points" lists all the access points that are installed and registered. When an access point is selected from the list, the corresponding information such as the name of the cell where it is installed, the name of the agent in that cell, the IP address of the agent and the geographical information is listed in the column labeled

"Information". The form also has two buttons, namely, "Get Info" and "Delete". The "Get Info" button is used to get the parameters such as SSID of the access point and Status of WEP on that access point. When this button is clicked the Central Administrator requests the corresponding agent to send the above information. The agent upon receiving the request collects the data from the beacon frames emitted by the access point (the module that actually collects the data is implemented in PERL and using tethereal), extracts the relevant fields and sends the information back to the CA.

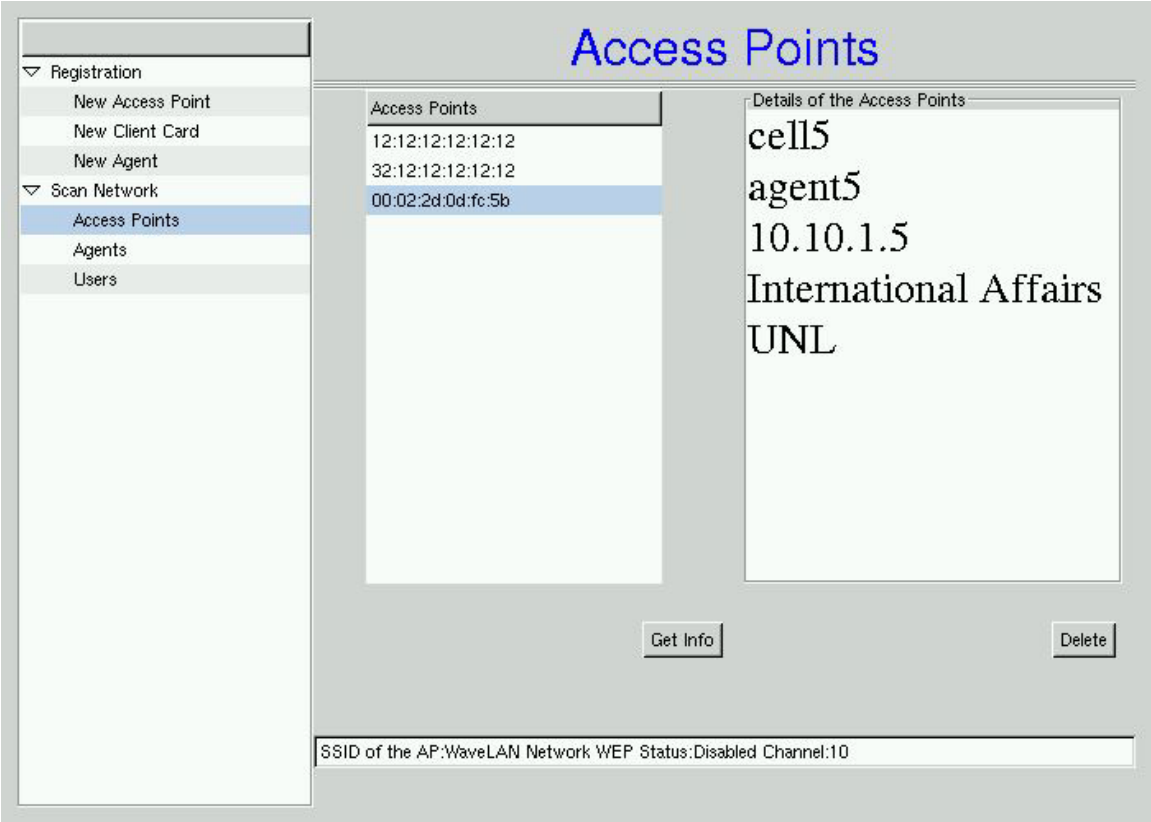

Figure 4.4 Scan Access Points GUI.

The "Delete" button shown in the figure above is used to unregister the access point. When this button is clicked the CA sends a message, " unregAP →xx:xx:xx:xx:xx:xx", to

the agents. The agents, upon receiving this unregistration order, delete the access point from their list of registered access points.

Figure 4.5 shows the interface that is used to scan the agents. The left column labeled Agents, consists of all the agents that are deployed across the network. When an agent is selected in this column the corresponding agent's information such as the access point it is serving, its name, its IP address and its location is displayed in the right side column labeled "Details of the Agent". The "Scan this Cell" button is used to scan the agents cell for all the available access points.

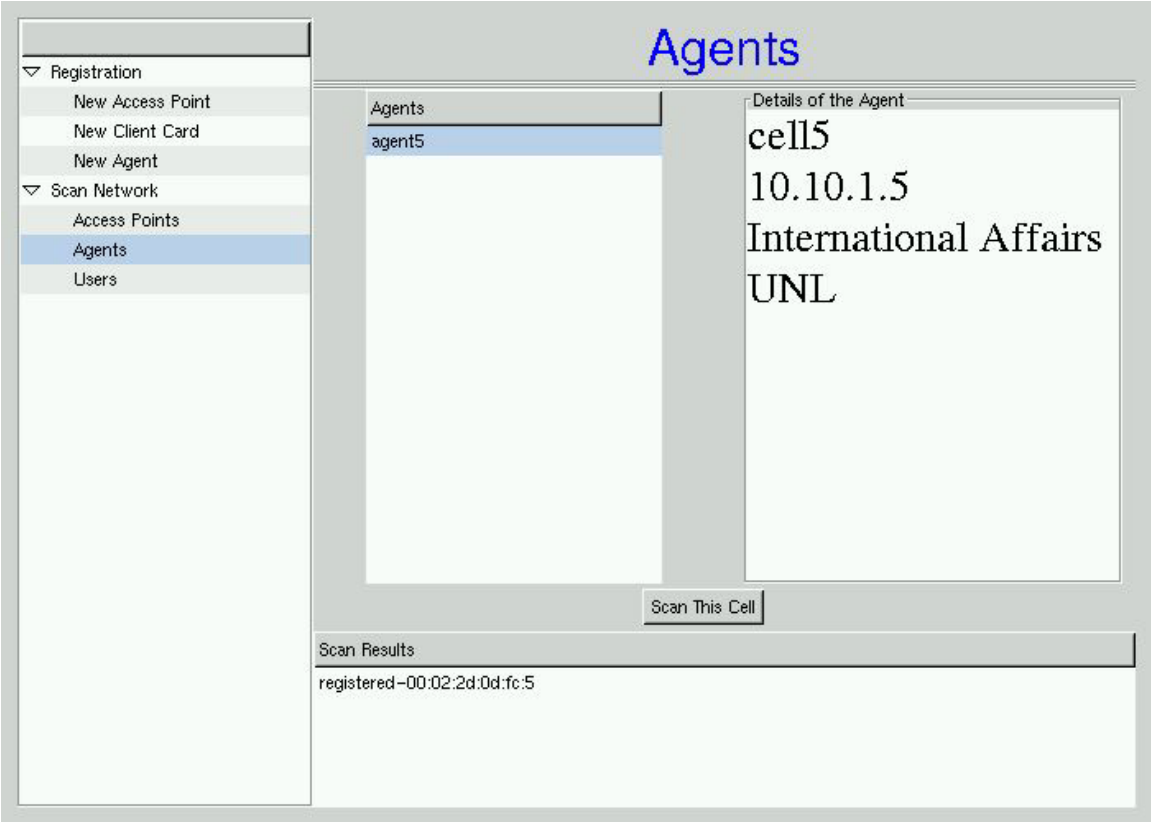

Figure 4.5 Scan Agents GUI.

When this button is clicked, the Central Administrator sends a scan request to that agent and the agent scans the network for 3 seconds (the scanner is a PERL module using

tethereal) and then sends the list of the access points along with the information whether it is registered or not back to the Central Administrator. The Central Administrator will then display the list it received from the agent as shown in Figure 4.5.

Figure 4.6 Pseudo Code for scanning rogue access points.

Figure 4.6 shows the pseudo code for scanning the rogue access points in a given cell. The module is a PERL (`sniff.pl`) script, which invokes tethereal to sniff the beacon frames as shown below

```
tethereal -i eth2 -a duration:10 -f 'ether[0]&0x0080==0x0080' -V|
```

The first part in the above statement, "`tethreal –i eth2`", tells tethereal to sniff for packets on interface eth2 (which is the wireless interface in our test bed setup). The second part of the statement, "-a duration:10" tells tethereal to sniff only for 10 seconds. The main reason to keep a 10 second duration is to reinitialize the counters in the PERL script before they go out of bounds due to continuous sniffing. We provide a wrapper with an infinite loop that calls the PERL script in each of its iterations. With this approach we will be able to run our sniffer continuously without running into any out of bound problems with the counters in the PERL script. The last part of the statement, "`-f 'ether[0]&0x0080==0x0080' -V|`", is a capture filter which tells tethereal only to capture beacon frames and dump the verbose description of the content into a pipe. The meaning of the above expression is to capture the packet only if the word at offset 0 in the Ethernet header is 0x0080. This word 0x0080 identifies that the packet sniffed is a beacon frame [8]. For each such packet dumped, the pseudo code shown in Fig. 4.6 is executed.

Figure 4.7 shows the pseudo code of the promiscuous-node-detecting module. This module consists of two source files, a C file (`arp.c`) and a PERL script (`arpresp_sniff.pl`). The `arp.c` file is used to send out fake ARP packets and the `arpreps_sniff.pl` script is used to sniff the ARP responses. In order to detect promiscuous nodes we would first send out a fake ARP packet for each of the IP address

in the given subnet (the `arp.c` file reads the subnet details from the subnet-range file) with the destination address ff:ff:ff:ff:ff:fe instead of ff:ff:ff:ff:ff:ff. Should there be a promiscuous node in the given subnet, the node's NIC would pass the ARP request to the TCP/IP stack on that node. The stack upon receiving this request just responds back with an ARP response. Now by detecting this ARP response we could conclusively determine whether a node with a given IP address is in promiscuous node or not.

```
1. Begin
2. For each IP in the subnet
3.      send a fake ARP and wait for reponse
4.      if response seen
5.          check log if entry exists;
6.          if entry exits then
7.              go to step 3 with the next IP
8.          else
9.              send a warning message and log the entry
10.             go to step 3 with the next IP
11.         End if
12.     else
13.         go to step 3 with the next IP
14.     End if
15.End
```

Figure 4.7 Pseudo code for detecting promiscuous nodes.

4.2 HARDWARE REQUIREMENTS

In order to deploy the system discussed above we need to have the following hardware resources (see Appendix A for a detailed list including pricing information).

- Access Points (could be from different vendors)
- Agents (One agent should be placed in each cell)
- 2 Ethernet cards and 1 wireless card for each agent. The wireless card that is supported by this system is Cisco Aironet 350 [The reason for supporting Cisco is because of the ease to put the wireless card into promiscuous mode].

4.3 TEST BED

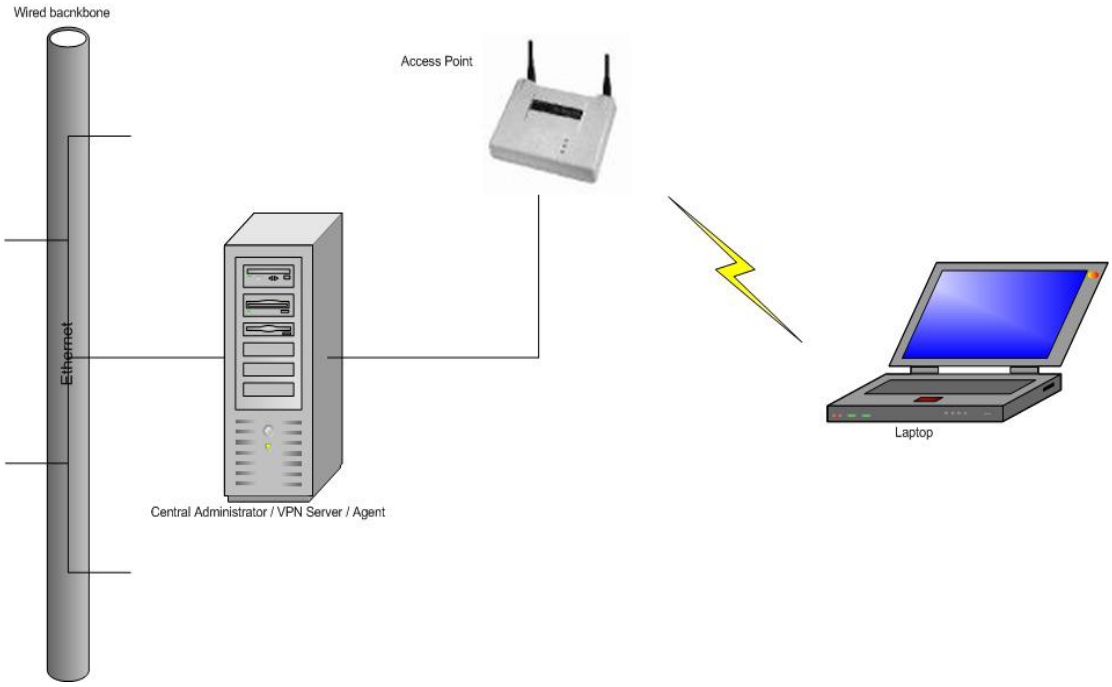

Figure 4.8 Test Bed.

Figure 4.8 shows the test bed on which the system has been implemented. Due to limited resources we implemented the system with one Lucent Orinoco AP 1000 access

point, one desktop, which acts as a VPN server, the Central Administrator and also as an agent. The desktop runs Redhat Linux 7.0 with 2.4.18 kernel and is equipped with three NICs, two Ethernet adapters and one wireless PCI-to-PCMCIA bridge to plug in the wireless card. We used two Orinoco Silver wireless cards, one for the laptop to communicate with the access point and one for the agent.

4.4 Results

We ran the following experiments on the test bed to test the accuracy of the software.

Experiment 1: Introducing a registered access point (00:02:2D:0D:FC:5B)

Figure 4.9 shows the screenshot of the result of scanning the cell in which the access point is installed. Please note the information listed in the scan results section of the screenshot.

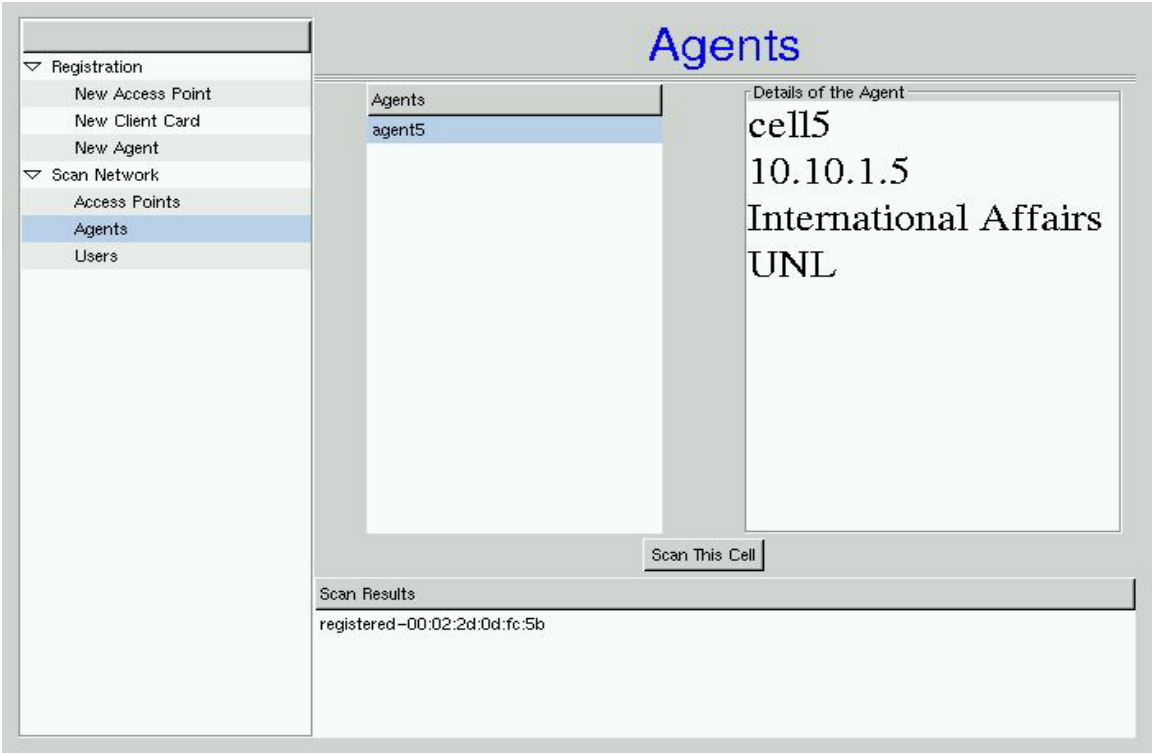

Figure 4.9 Introducing a registered Access Point

Experiment 2: Introducing an unauthorized access point (00:02:2D:0D:FC:5B)

The following two figures, Figure 4.10 and Figure 4.11, show the e-mail sent to the administrator and the result of scanning a cell where an unauthorized access point is introduced respectively.

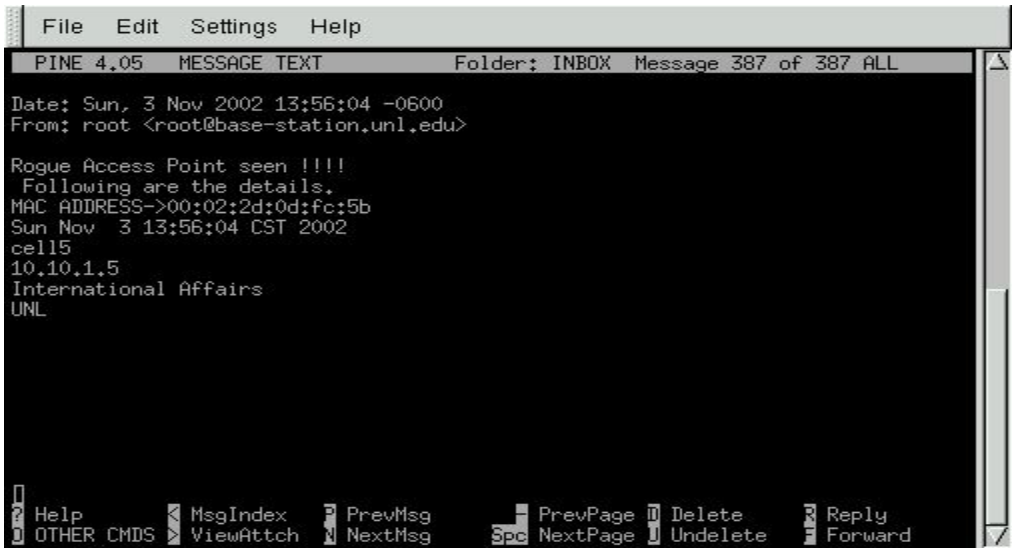

Figure 4.10 E-mail sent to the administrator when a rogue access point is installed.

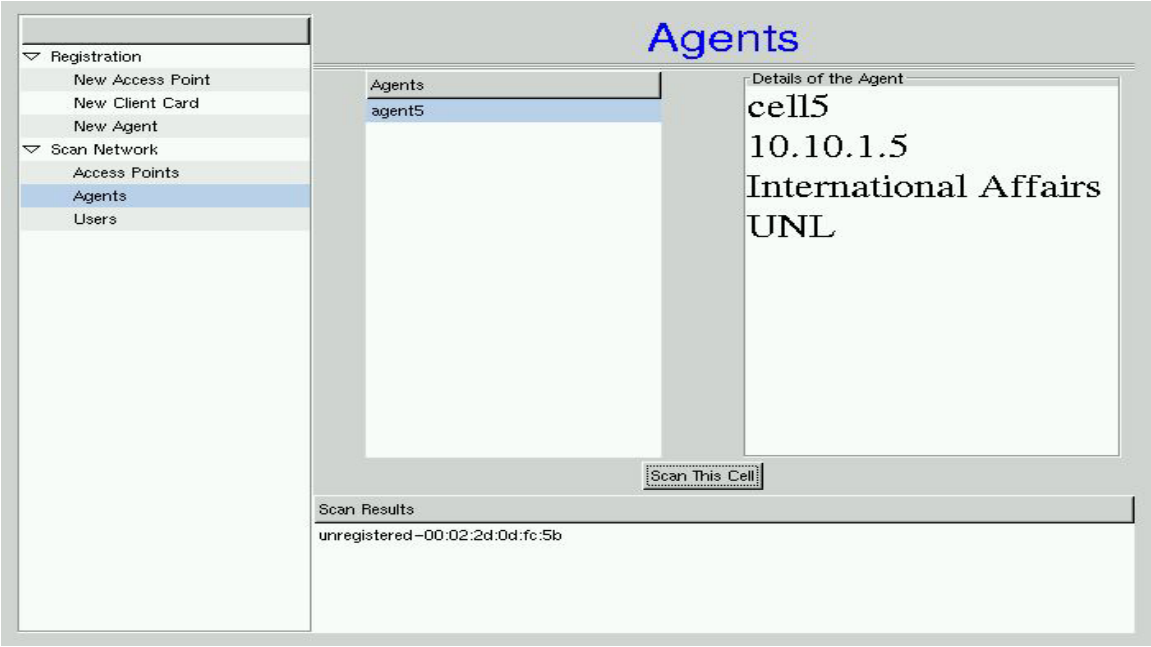

Figure 4.11 Introducing a rogue access point.

Experiment 3: Introducing a promiscuous node in a cell

Figure 4.12 is screenshot of the e-mail sent to the network administrator as soon as a promiscuous node is detected in a given cell.

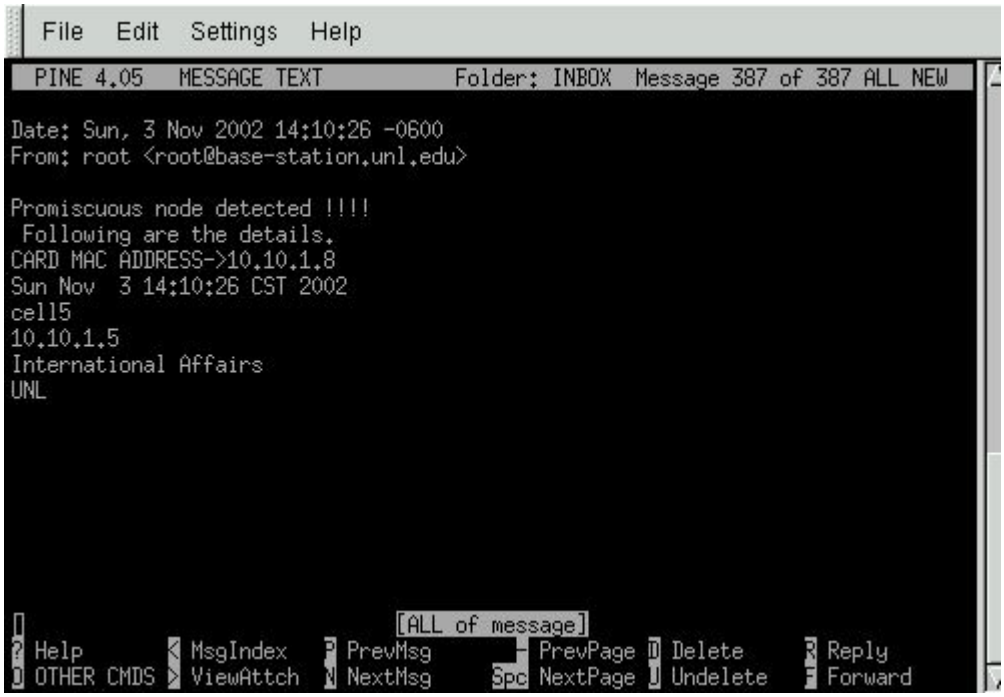

Figure 4.12 E-mail sent to the administrator when a promiscuous node is detected.

4.5 Advantages and Disadvantages

The main goals of the project are to make the wireless network as secure as possible – using VPN, and to detect intrusions as early as possible. The system has been fully tested for its accuracy and promptness in identifying intrusions. One of the greatest advantages with our approach is that the wireless cell to be monitored has no geographical restrictions. As long as the agent is connected to the Internet, the wireless cell can be at any distance from the central administrator. However, there are a few complications and restrictions involved. Every wireless element should be registered with the central administrator in order to be labeled as legitimate. Another limitation on the functionality

of agents is its inability to detect a promiscuous node which is not IP based. Since we use the technique of ARP spoofing, the agent can detect a promiscuous node if and only if that node is associated with a TCP/IP stack. Lastly, the system is meant for IEEE 802.11b networks. Should there be an access point which does not comply with the IEEE standard, the system will fail in detecting it. Another issue is involved is the fault tolerance of the system. If the CA dies for some reason, the whole wireless network would not be able to communicate with the wired network and all the current VPN connections would go down.

4.6 Future Work

The architecture employed in this project can be used to solve some of the existing problems in IEEE 802.11 networks. One of the most important problems besides security is interoperability between different vendor products. For issues like the handoff mechanism, which does not have any standard defined, different vendors have their own proprietary solutions making it impossible to achieve interoperability. With our proposed architecture, agents can be allowed to transfer the clients between the cells instead of the access points performing the handoff. With this kind of approach we can have access points from different vendors cooperating with one other.

Another interesting improvement that can be made to the existing system is to allow restricted access to unregistered users instead of simply blocking them. Traffic can be analyzed at a corresponding agent, each time an unregistered user accesses the wireless network. The results of the analysis can be communicated back to the central administrator, which decides the legitimacy of the user cumulatively using similar kinds

of information from other agents. Using such a solution, even unregistered users can enjoy the convenience of wireless connectivity with, of course, limited access.

Chapter 5

Conclusion

Our agent-based intrusion detection and response system has been developed to provide a secured wireless LAN solution. Our main focus is on detecting unauthorized elements as soon as possible. By incorporating the feature to generate x.509 certificates, a VPN solution can be easily deployed to overcome the shortcomings of WEP. The architecture proposed here can stand as a building block for finding efficient solutions for various other problems discussed earlier. Since the whole project involves open source tools, the cost for deploying such a system is very minimal.

Appendix A

Equipment

Hardware	Manufacturer	Quantity	Cost in USD
ORINOCO AP-1000 Access Point	Lucent (Now Proxim)	1	699.05
ORINOCO PCI Adapter	Lucent (Now Proxim)	1	72.00
ORINOCO PC Silver Cards	Lucent (Now Proxim)	4	518.00
CISCO Aironet 350 Wireless cards	Cisco	1	135.00
Winbook Notebook Computer	Winbook	1	1498.00

Table A. Hardware used for the Test Bed.

Total Cost: 2,922.00 USD

Vendor Websites

1. Lucent http://www.orinocowireless.com/template.html?section=m58&page=103&envelope=94
2. Cisco http://www.cisco.com/en/US/products/hw/wireless/ps458/index.html
3. WinBook http://www.winbook.com

References

[1] I.N. Borisov, I. Goldberg, D. Wagner. Intercepting Mobile Communications, In Mobicom 2001, Rome, Italy, Jul. 2001.

[2] A. Stubblefield, J. Ioannidis, A.D. Rubin. Using the Fluhrer, Mantin, and Shamir. Attack to break WEP, In Network and Distributed System Security Symposium. San Diego, California Feb. 2002.

[3] S. Fluhrer, I. Mantin, and A. Shamir. Weakness in the key scheduling algorithm of RC4. In Eigth Annual Workshop on selected Areas in Cryptography, Toronto, Canada, Aug. 2001.

[4] A. Mishra, W.A. Arbaugh. An Initial Security Analysis of the IEEE 802.1X Standard CS-TR-4328, University of Maryland, Feb 2002 http://www.cs.umd.edu/~waa/wireless.html Accessed in Aug. 2002.

[5] Netstumbler. A windows tool to scan for available wireless networks. www.netstumbler.org Accessed in Aug. 2002.

[6] Kismet. A linux tool used for scanning wireless networks. www.kismetwireless.net Accessed in Aug. 2002.

[7] AirTraf. A linux tool for gathering information about the available wireless networks. http://airtraf.sourceforge.net/ Accessed in Aug. 2002.

[8] 802.11 Standard http://grouper.ieee.org/groups/802/11.

[9] D. Sanai. Detection of promiscuous Nodes Using ARP packets. A white paper from http://www.securityfriday.com Accessed in Aug. 2002.

[10] Airsnort. http://airsnort.shmoo.com/ Accessed in Oct. 2002

[11] APTools. http://aptools.sourceforge.net Accessed in Oct. 2002

[12] Dstumbler. http://www.bawug.org/howto/reviews/dstumbler.html Accessed in Oct. 2002

[13] Mohan Chirumamilla. Interoperability and Performance between cross-vendor 802.11b products. A white paper. http://cse.unl.edu/~mohankc Accessed in Oct. 2002

[14] IPSec Security Architecture for the Internet Protocol. http://www.ietf.org/html.charters/ipsec-charter.html Accessed in Oct.2002

[15] James T. Geier, Jim Geier. Wireless LANs. ISBN: 0-672-32058-4. Purchased in Jul. 2001

[16] http://www.personaltelco.net/index.cgi/WarDriving

[17] X.509 Certificates. http://www.ietf.org/html.charters/pkix-charter.html Accessed in Oct. 2002

[18] IPTables http://www.netfilter.org/ Accessed in Oct 2002