Detection of rogue devices in Wireless Networks

by

Jeyanthi Hall

A thesis submitted to

the Faculty of Graduate Studies and Research

in partial fulfilment of

the requirements for the degree of

Doctor of Philosophy

Ottawa-Carleton Institute for Computer Science

School of Computer Science

Carleton University

Ottawa, Ontario

August 2006

Your file Votre référence
ISBN: 978-0-494-18221-5
Our file Notre référence
ISBN: 978-0-494-18221-5

Abstract

The need for robust access control mechanisms is paramount, especially in Wireless Local Area Network (WLAN)s and Wireless Wide Area Network (WWAN)s. Current authentication systems are vulnerable to device impersonation by rogue devices.

Within cellular mobile networks, this threat is actualized by cloning cell phones, and using the clones for obtaining free services. A well known example in Wireless Fidelity (WiFi)/802.11 networks is Media Access Control (MAC) address spoofing. In this case, an attacker captures the MAC address of an authorized device and programs it into his device, in order to obtain unauthorized access. The threat of address spoofing is equally applicable to Bluetooth (BT) ad-hoc networks.

The underlying problem is the continued use of Access Control List (ACL)s, based on a single malleable identifier, e.g. MAC addresses. Given the ease with which the aforementioned attacks are mounted, and the potential impact on these networks, there is a requirement for access control mechanisms that are capable of detecting impersonation attacks.

What would prove useful is to associate a malleable identifier with less malleable characteristics. Hence, we explore the feasibility of using Anomaly-based Intrusion Detection (ABID), which makes use of device-based and/or user-based profiles for addressing the aforementioned problem. For example, an ABID system would compare multiple instances of device/user characteristics, associated with a given identifier, to those in the corresponding profile. Deviations from pre-established thresholds would be indicative of cloning or address spoofing.

More specifically, we explore the use of Radio Frequency Fingerprinting (RFF) for characterizing transceivers in WiFi/802.11 and BT wireless cards, i.e. create device-

based profiles, and Hotelling's T^2 statistics for classification purposes. Similarly, we also investigate the adoption of User Mobility Pattern (UMP)s for user-based profiles and the Instance-Based Learning (IBL) technique for classification. Average detection rates of 93% (BT) and 94.5% (WiFi/802.11) support the feasibility of incorporating RFF, in ABID, for detecting address spoofing. On the other hand, the use of UMPs for similar purposes is also technically feasible. Thus, device-based and user-based characteristics can be exploited for detecting rogue devices in wireless networks.

To *my late father and sister,* **Edwin and Jessie Vethamuthu**, *my mother* **Vasantha**, *my husband* **Charles** and *my daughter* **Tasha**.

Acknowledgements

The undertaking and completion of the Ph.D program has proved to be an eventful journey, filled with tribulations and challenges, both personal and professional. Hence, it is with immense gratitude that I acknowledge those, who have provided me with inspiration, financial and moral support, and technical direction.

I start by thanking God for the strength he had provided, especially during the most difficult moments in my life. This work would not have been possible without him.

I will always be grateful to my father and sister for being two of my role models in life. Although they are no longer with me, their infinite love and kindness exemplified the unmeasurable power of the human spirit. I would also like to thank my mother for her unconditional love and support, especially when she assumed the role of the chauffeur. Likewise, I would like to acknowledge my husband, for his services as editor and chef. He not only prepared the meals but also served it with a smile, sometimes night after night. As my personal editor, he has been instrumental in improving the quality of all research documents. If there is one thing that I have learnt from my daughter, it is the attitude with which to face all adversities of life. I would like to thank her immensely for being a wonderful daughter and friend. Finally, I thank each member of my extended family (Juana, Jennifer, Mike, Vince and Marcin) and close friends (Sue and Florentina) for their continued support and prayers.

There are no words, which can adequately express my gratitude for the encouragement and technical direction, which I have been so fortunate to receive from my supervisors, Professors Evangelos Kranakis and Michel Barbeau. They have taught me so much more than wireless networks and security.

I would also like to thank the committee members for their invaluable comments and suggestions. Their input and guidance has proved beneficial in establishing the scope and objectives of this research initiative. Moreover, I am very grateful to Dr. Nur Serinken and Michel Paquette for their invaluable feedback on technical matters.

The fact that the quality of my life, as a student, had been exceptional, is due to the support that I had received from my fellow colleagues and the administrative team. I would like to personally thank Cindy, Miguel, Tao, Paul, Shao, Linda, Sandy, Jane and Sharmilla for their assistance in coping with all facets of the Ph.D program.

Finally, for providing me with financial assistance, I extend my sincere appreciation to the following organizations: Natural Sciences and Engineering Research Council of Canada, Mathematics of Information Technology and Complex Systems, and Alcatel.

Contents

List of Tables

List of Figures

List of Algorithms

Part I

Preliminaries

Chapter 1

Introduction

The new information age can be characterized by two significant trends: the unprecedented growth in the number of wireless users, applications and network technologies; and interoperability between these technologies, a necessary prerequisite for m-commerce, multi-media services, and other applications.

Significant technological advances in both short and long-range wireless communications, simplification of installation and maintenance procedures using built-in radio link analysis and menu-driven configuration [78] as well as reduction in costs, are but a few factors that have contributed to the accelerated deployment of wireless networks.

In fact, this level of growth is only expected to increase in the next few years. The standardization of WLANs around WiFi/802.11b has provided the impetus for the abrupt increase in wireless devices, equipped with WLAN access cards. According to a Gartner report [146], 60% of the population in the U.S. and Europe will be carrying these devices by the end of 2007, with this figure increasing to 75% by 2010. In terms of WWANs, the deployment of Code Division Multiple Access (CDMA)-based cellular mobile networks in areas, such as Asia, North/South America and Europe [66], is being fueled by the tremendous increase in the use of cellular phones, and the potential revenues that can be generated. Finally, the prevalence of short to moderate range communications, combined with very high data rates in excess of 100 Mbs, suggests a renewed interest in ad hoc operations [53]. In fact, the increased

use of BT radios, in Personal Area Network (PAN)s, sensor networks, cellular phones and embedded systems, provides a clear indication as to the type of applications that are starting to emerge.

In addition to the increased coverage and services provided by these networks, there has also been a concerted effort to promote interoperability between WLANs and WWANs. More specifically, the current trend, within the WWAN domain, is marked by the unification of technologies, including Advanced Mobile Phone System and Digital Cellular System (DCS) towards International Mobile Telecommunications (IMT)-2000, as indicated by Schiller [151], and ultimately towards full Internet Protocol (IP)-based multimedia networks. This strategic direction should prove beneficial to users and service providers. Not only would it fulfill users' needs for enhanced IP-based services, but it would also permit service providers to develop and to deploy these services, in a more cost effective and timely manner.

As a matter of fact, the convergence of WWAN and WLAN technologies can already be witnessed. Prompted by the need to improve cellular reception inside businesses and homes, LG Electronics has recently introduced a cellular-WLAN handset, e.g. LG CL400, which combines Global System for Mobile Communications (GSM) with WiFi [128]. A similar strategy has also been adopted by Qualcomm. Their Mobile Station Modem chipsets will provide support for a new WLAN module, from Philips Electronics, thus permitting manufacturers to WiFi enable cellular handsets. This new handset platform integration will enable connectivity to both WiFi/802.11b and 802.11g networks, as well as to existing CDMA2000 and Wideband CDMA (WCDMA) cellular networks. Regardless of the underlying technologies, with this hybrid model, users will benefit from higher WLAN speed, e.g. 11 to 54 Mbit/sec, when they are within the range of WLAN Access Point (AP)s. According to Bharat Sanchar Nigam Limited [138], the market for this new service is expected to reach $1.6 billion in the United States by 2010, with more than 26 million subscribers.

In addition to these trends, there are other initiatives being undertaken to support 4^{th} generation systems. For example, key developments in information and communication fields include inter-machine communication (e.g. cars, household and office

components equipped with a wireless interface at a cost of US$20), packet-oriented wireless systems and heterogeneous wireless infrastructures, as identified by Bria *et al.* [25].

1.1 Motivation

While these initiatives continue to support the vision of pervasive and ubiquitous computing, where users have access to network applications and resources anytime and from anywhere, security must also keep pace. Better yet, network operators and service providers should take a proactive role in implementing a secure infrastructure. Confidentiality of information is critical for supporting applications, including on-line banking and electronic payments, e.g. m-commerce. As infrastructure-based WLANs represent an extension to wired LANs, the need for effective access control to network resources is equally paramount. This requirement is also evident in WWAN, and to a lesser extent, in Ad-Hoc networks.

Whereas the need for encryption can be fulfilled with appropriate cryptographic mechanisms, e.g. shared secret keys, providing robust access control remains a challenge, for a number of reasons.

First and foremost, current authentication protocols, in all three areas of wireless networks, continue to exhibit vulnerabilities, which are being exploited by attackers. In response to these threats, a considerable level of effort has been expended by various standards bodies to address the key vulnerabilities. For example, the replacement of the COMP128 authentication algorithm in GSM networks by COMP128 version 2 or 3, has been successful in minimizing Subscriber Identity Module (SIM) cloning.

Therefore, until network operators/administrators implement similar protocols and algorithms, the resources of the network remain exposed. Of course, these enhancements may not necessarily address the actual problem. For example, the 802.11b and BT standards support **device** authentication at the ***link layer***. Hence, the adoption of link layer strategies, for addressing weaknesses at this layer, would seem logical. Instead, both the standards bodies and research teams have recommended the use of

higher layer security services, such as ***user*** authentication, to compensate for these weaknesses. While this approach is perfectly acceptable and appropriate for certain types of applications, it is rather technically challenged when it comes to severely constrained devices such as BT and those used in sensor networks. More specifically, the use of public-key cryptography not only demands more resources from these devices (assuming such resources are available), but based on the number of layers of authentication used, there could be a significant degradation in performance as well as increased latency.

Second, there are threats, which cannot be addressed effectively by authentication, an *intrusion prevention* technique. In particular, *device impersonation* represents one of the most significant threats in WLAN/WWANs. A well known actualization of this threat, in WiFi/802.11 networks, is MAC address spoofing. An instantiation of device cloning, this attack is carried out by obtaining the MAC address of a legitimate user, programming it into another device, and subsequently using it to obtain access to a WLAN.

In order to address these and other types of threats, various *intrusion detection* techniques have been proposed by research teams. While some of them are practical, they nevertheless support short-term strategies, and hence may fail to protect networks in the long-term. On the other hand, others demand a considerable level of effort and expenditure, but are designed to provide an enhanced level of access control.

In any event, one important characteristic, that is exhibited by a majority of these solutions, is the use of ABID. This detection mechanism is carried out as follows. First, the normal behavior (feature) of a device or user is captured and stored in a profile. Second, an Intrusion Detection System (IDS) classifies a newly observed behavior as normal or anomalous. Essentially, it compares the observed behavior to that in the profile. If the level of similarity is within pre-established thresholds, a verdict of normalcy is rendered. Otherwise, an intrusion is suspected and an appropriation alert mechanism is initiated.

1.2 Research Objective

Given that "WLANs will be the largest growing wireless security problem faced by enterprises through 2008", as predicted by Gartner [57], and the need for robust wireless access control, our primary objective is to detect device impersonation by rogue devices, as exemplified by MAC address spoofing.

Impersonation of devices

As aforementioned, MAC-address spoofing will continue to be problematic, as long as organizations use ACLs that are based exclusively on *single* and *malleable* identifiers. As a matter of fact, MAC-address filtering is not only being used to grant access to wireless users, it is also intended to be used for addressing Rogue Access Point (RAP)s. These unauthorized APs are either installed by an intruder or by internal staff, for the purpose of gaining access to the resources of the wired network. According to Ernst and Young [48] and other sources, it is currently the most pressing problem in WiFi/802.11 networks.

If left unresolved, the resulting damages could be significant. These include the consumption of valuable resources, e.g. bandwidth, theft of sensitive data and a free pass to initiate a number of attacks, including Denial of Service (DoS).

1.3 Approach

While intruders cannot be prevented from cloning MAC addresses of legitimate users, *additional* and *less malleable* identifiers can, nevertheless, be used for detecting device impersonation by rogue devices. The concept of using two or more identifiers, for corroborating the identity of an entity, has been applied to many facets of security. Two well known examples are the use of two keys, in public key cryptography, and two pieces of identification, for banking and other applications.

With that in mind, we explore the feasibility of using ABID, which makes use of device-based and/or user-based profiles.

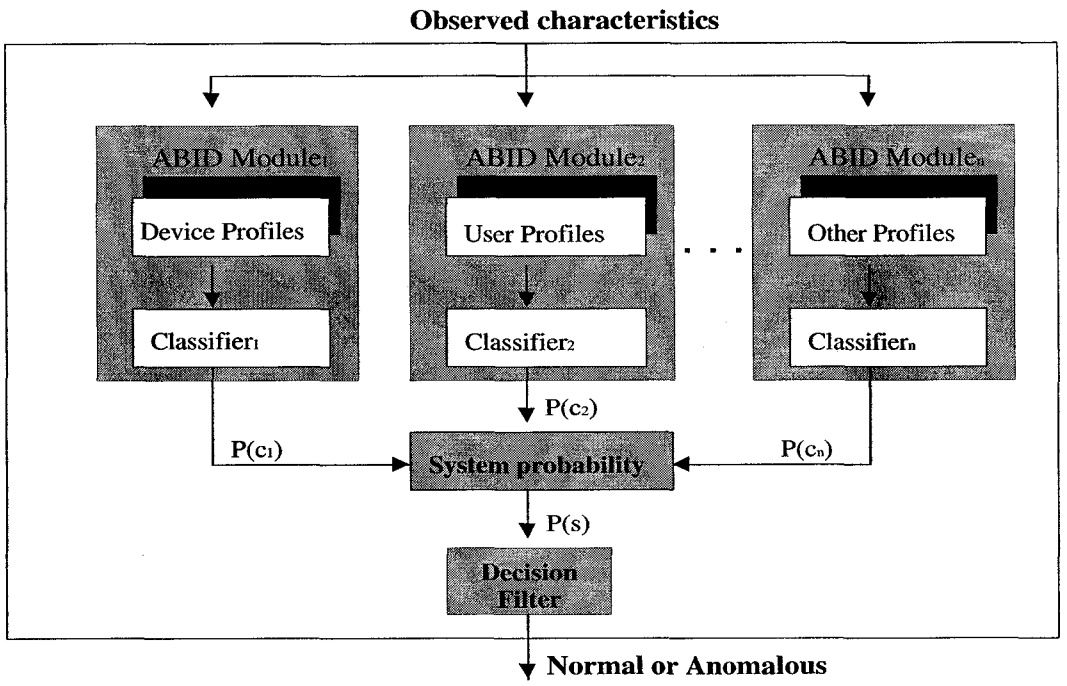

Figure 1.1: Solution framework: Multifactor WIDS

The novelty of this approach is illustrated in Fig. 1.1. The key underlying concepts, associated with the profiling and classification phases, are as follows:

1. Profiling: characteristics, which are difficult to replicate, are captured and stored in a profile; and

2. Classification: a final decision is rendered, by a Wireless Intrusion Detection System (WIDS), based on individual or multiple profiles, using a probabilistic model.

In light of the fact that our objective is to detect device impersonation, it is essential that a given set of features characterizes a device in an accurate manner. In the case of a device-based profile, this set would characterize the hardware, software or other components of a device.

Once the profiles of authorized devices are created, classification of an observed behavior or characteristic is carried out, in order to determine whether or not an impersonation attempt is in progress. Therefore, by associating a MAC address, in

the ACL, with the profile of the corresponding device, one should be able to detect MAC address spoofing. Moreover, it is anticipated that a generic IDS module can be adapted to other wireless networks with a minimum level of effort. This would prove particularly useful for addressing other forms of device impersonation, e.g. BT address spoofing and cloning of cellular phones in CDMA2000 networks.

While device-based profiles may prove sufficient for our purposes, it would be interesting to explore user-based profiles. The underlying premise is that the normal behavior of a legitimate user will be different, to some degree, from that of an intruder. Therefore, a user-based IDS module should be able to detect this type of intrusion. Those, based on mobility, speech, calling patterns and other characteristics, have been used in WWANs, as countermeasures against fraud. Perhaps, one could make use of both device-based and user-based profiles, as with multi-factor authentication, for ensuring that an authorized *user* is using his/her own *device*.

Although targeted as a future research initiative, our secondary objective is to explore the use of specific probabilistic models, e.g. Dempster-Shafer theory, for aggregating the results of these independent modules. This approach differs from the use of multi-sensor data, in that only the results from each module are amalgamated, and not the data. Taking these results into consideration should permit a WIDS to determine a system-wide probability of an intrusion, and to subsequently render a more accurate decision.

1.4 Summary of contributions

Before one embarks on a quest for a multi-factor WIDS, the feasibility of employing both types of profiles must be assessed. This section presents a summary of the contributions, which have originated from this research initiative. Table 1.1 highlights the evaluation results, associated with the use of device-based and user-based profiles. As far as device-based classification metrics are concerned, the False Alarm Rate (FAR) represents the percentage of transceiverprints (defined in the sequel), from transceiver X, which have been incorrectly classified as belonging to transceiver Y.

Device-based profile	Evaluation results
Transient detection	BT:85-90% WiFi:95%
Transceiver identification	WiFi:94-100%
Transceiver verification	WiFi:FAR(0%) and average DR(94.5%) BT:average FAR(0.05%) and DR(93%)
User-based profile	Evaluation results
Classification	class1 (user 19)-FAR(0%)DR(92.5%) class2 (user 23)-FAR(100%)DR(92.5%) class3 (user 41)-FAR(5%)DR(90%)
	95% confidence interval class1-highly consistent class2-consistent class3-chaotic

Table 1.1: Evaluation results

On the other hand, Detection Rate (DR) refers to the percentage of transceiverprints, which have been correctly classified as belonging to transceivers, other than X. These metrics have been used, in a similar manner, to assess the performance of user-based classification.

ABID using device-based profiles

A brief summary of the contributions, associated with the profiling and classification phases of ABID, is presented next.

Profiling Phase: Detection of start of transients

As the unique characteristics of a transceiver are manifested in the transient (between channel noise and data transmission) of a signal, the key task is to extract it for profiling purposes. But first, the starting point of transients must be determined. Existing algorithms, which make use of amplitude data or discriminator output, do not perform well with signals that exhibit a gradual shift in amplitude at the start of the transients, e.g. BT and WiFi/802.11 devices. Hence, we exploit the phase characteristics, associated with channel noise and transient, for this purpose, as demonstrated in [71]. Using phase data is advantageous since it is less susceptible to noise and interference.

Classification Phase: Transceiver identification

For addressing the issue of MAC address spoofing, we propose a novel approach, which makes use of RFF [72]. RFF is used to *identify* a transceiver based on its transceiverprint, i.e. *set* of features extracted from the transient. The use of multiple features reflects the strategic direction, currently being adopted by those in the area of biometrics. Moreover, the success rate of a wireless IDS is also improved by correlating several observations in time using Bayesian filtering. Evaluation results, based on a set of thirty transceivers from Lucent, are promising. Nevertheless, a larger sample is required in order to fully assess the feasibility of incorporating RFF and Bayesian filtering techniques into a WIDS.

Classification Phase: Transceiver verification

Unlike transceiver identification, the verification of a transceiver is carried out in order to determine if a given WiFi/802.11 transceiver had in fact generated an observed signal/transceiverprint. Although we use the same transceiver profiles for this purpose, there are a few nuances, associated with the underlying classification process.

Firstly, a Multivariate Statistical Process Control (MSPC) technique, namely Hotelling's T^2, and a threshold that has been established using the F distribution, are used to classify a transceiverprint. That is, the statistical classifier determines if a transceiverprint matches the profile of the transceiver, corresponding to the claimed MAC address.

Secondly, it is generally known that current IDS render a decision, as to whether an observed behavior is normal or anomalous, based on a *single* observation. In an environment that is characterized by interference and noise, delaying the decision until *multiple* observations have been processed, could reduce the level of uncertainty and result in a lower FAR. Therefore, as with the Bayesian filter, a decision filter is used to achieve this goal.

Lastly, the notion of concept drift, i.e. normal changes in behavior, is addressed by continuously updating the profile of a transceiver. The application of this strategy

not only results in reducing the FAR, but in increasing the DR also. Evaluation results for 95% confidence interval, see Table 1.1, are encouraging. Additional details are available in [74].

Within the category of transceiver verification, we have also experimented with BT devices for the purpose of developing a generic IDS module. Our initial objective is to determine the success rate resulting from the application of the aforementioned algorithms and processes. Preliminary results support the technical feasibility of using a device-based IDS module within BT networks [75]. A potential application is the detection of device impersonation, which is carried out by using the address of another BT device and its unit key.

ABID using user-based profiles

The utilization of profiles, based on hardware signatures, calling patterns, service usage, and mobility patterns, have not only been explored by various research teams, but have been implemented in commercial systems also, namely the Fraud Management System by Hewlett-Packard [80] and Compaq [34]. We examine the feasibility of using profiles, which are based on mobility patterns of public transportation users [73]. More specifically, a novel framework, which makes use of an IBL technique [104], for classification purposes, is proposed. In addition, an empirical analysis is conducted in order to assess the impact of two key parameters, the Sequence Length (SL) and Precision Level (PL), on the FAR and DR. Moreover, a strategy for enhancing the characterization of users is also identified. Current evaluation results reflect the need to accurately characterize the mobility behavior of users, in particular, those with quasi-consistent and chaotic mobility patterns. Details of experiments, using a smaller set of users, are available in [76].

ABID using both profiles

The complementary use of device-based and user-based profiles, for detecting impersonation attacks in future wireless and mobile networks, is discussed in the paper by Barbeau, Hall and Kranakis [20].

1.5 Outline of Thesis

The remainder of the thesis is structured as follows:

Part II The current state of intrusion prevention in PANs (e.g. BT), WLANs (e.g. WiFi/802.11b and 802.16), and WWANs (e.g. GSM and CDMA2000) is analyzed. This exercise is carried out by examining *front door* security, i.e. the standards-based protocols used for authentication purposes. In addition, key vulnerabilities, associated with these protocols, as well as proposed resolution strategies are presented.

Part III Specific attacks, related to the breach of *back door* or *peripheral* security, are presented, under the umbrella of intrusion detection. They exploit authentication-based vulnerabilities or have surfaced, as a result of advanced technology. Moreover, countermeasures, for overcoming the underlying weaknesses, are also identified.

Part IV Despite the availability of resolution strategies and countermeasures, robust access control in WLANs continues to be problematic. The outstanding problems or attacks, related to device impersonation, are analyzed using Attack Risk Analysis (ARA), a variant of Threat Risk Analysis (TRA). Finally, the detection of MAC address spoofing is targeted for further investigation.

Part V The use of a device-based profile in WIDS, for detecting MAC address spoofing, is explored. More specifically, the feasibility of using RFF and T^2 statistical classifier, for profiling and classification purposes respectively, is considered. Lastly, evaluation results of the classification component, including memory requirement and time complexity of classification algorithms, are discussed in detail.

Part VI In a similar vein, the incorporation of UMPs into WIDS, is investigated. Furthermore, results from an empirical analysis of the parameters, associated with IBL classification, are presented. Finally, evaluation results of classifica-

tion metrics, e.g. FAR and DR, as well as the memory requirement and time complexity of the classification algorithm are discussed.

Part VII The last part of the thesis highlights the conclusions and future research activities, associated with the use of device-based and user-based profiles. Other related research activities are also identified.

Appendix A In order to enhance the understanding of Part V, a brief introduction to digital signal processing is provided.

Appendix B A brief overview of the Dempster-Shafer theory and its application to WIDS are presented in this appendix.

Appendix C A list of acronyms is provided in this appendix.

Part II

Background: Access Control in Wireless Networks

Chapter 2

Intrusion prevention in Wireless Networks

In order to provide access control to the resources of the wired network via a wireless network, organizations rely on two key mechanisms: intrusion *prevention* and intrusion *detection*. These mechanisms can readily support the notion of a multi-layered approach to access control. Whereas intrusion prevention, a first line of defense, is carried out through encryption and authentication (e.g. using passwords or biometrics), intrusion detection (discussed in chapter 3) is performed, over a longer time period, using misuse and/or anomaly-based mechanisms. Of course, in the case of a successful intrusion, an appropriate response is initiated.

Whereas the need for encryption is fulfilled by employing appropriate cryptographic mechanisms, e.g. shared secret keys, authentication functions and protocols are implemented at different layers of the Transmission Control Protocol/Internet Protocol (TCP/IP) stack.

Authentication Functions

There are two main classes of authentication functions, which produce an authenticator, a value that is used to verify the identity of an entity, such as a device or user [164]. It is assumed that the key, used in the following techniques, is known only to the claimant (entity being authenticated) and the verifier.

Message Authentication Code (MAC_2)

A Message Authentication Code (MAC_2), a small fixed-size block of data, is created as a function of a message and secret key. It is then attached to the message before the message is sent to the intended recipient, who also shares this key. The use of a MAC_2 is intended to support authentication and validation of data integrity. Although the MAC_2 function is similar to encryption, it is less vulnerable due to its mathematical properties. It is also particularly useful in situations where confidentiality is not required (e.g. the Simple Network Management Protocol (SNMP) Version 3) or when authentication and encryption services are provided at different layers.

Message Encryption

Unlike a MAC_2, a ciphertext of the message is used as the authenticator. Encryption can be carried out using a symmetric or asymmetric (public-private) keys. While symmetric encryption supports confidentiality and authentication, asymmetric encryption provides support for non-repudiation as well. This security service is realized by having the claimant create a digital signature, by encrypting the MAC_2 of a message with his/her private key.

Authentication Protocols

As aforementioned, authentication protocols have been implemented at different layers of the TCP/IP protocol stack, see Fig. 2.1. It is important to note that while the following protocols and applications have traditionally been used to secure wired networks, they are slowly being migrated to the wireless domain. Although the transfer of technology, from the wired to wireless environment can prove useful, it can be equally challenging. For one thing, the operational characteristics (e.g. directed vs. undirected form of communication) are significantly different. Furthermore, the underlying assumptions, upon which the protocols have been developed, may no longer be valid.

Layer of Protocol Stack	Authentication Protocols
Application	RADIUS / KERBEROS
Transport	SSL/TLS or WTLS
Internet	IPSec - Virtual Private Network (VPN)
Network Access / Data link (OSI)	802.11b, Bluetooth, GSM
Physical	Typically not used

Figure 2.1: Authentication protocols at different layers

As depicted in Fig. 2.1, the authentication protocols fall into one of two categories. On one hand, the protocols, which authenticate devices (e.g. BT), are typically implemented at the link layer using hardware or firmware. On the other hand, higher-layer (from network to application) protocols, which authenticate users and provide other security services, e.g. end-to-encryption and non-repudiation, are implemented using a combination of hardware, firmware and software, as indicated by Stallings [165].

A bottom-up approach is adopted for the presentation of the key user authentication protocols, starting from the Internet or IP layer. Device authentication protocols, implemented at the link-layer, are presented in the following sections. Although the implementation of an authentication mechanism at the physical layer is not that common, it is nevertheless possible. For example, the use of secret codes, with direct sequence spread spectrum, could provide a rudimentary form of authentication, given that only authorized devices, equipped with a secret code, would be capable of recovering the original signal.

Internet Layer - Virtual Private Network (VPN)

The implementation of security services, such as device authentication, data confidentiality and key management (e.g. IP Security (IPSec), can provide a secure form of networking [164]. Moreover, higher-level protocols and applications can benefit from these services. It is a strategy, proposed by Bria *et al.* [25], that can be adopted for securing wireless networks such as WLAN and General Packet Radio Service (GPRS)/Universal Mobile Telecommunications System (UMTS).

Authentication, using a MAC_2, is provided using the Authentication Header (AH) or Encapsulating Security Payload (ESP). In addition, both protocols can be implemented in transport mode, i.e. AH is inserted into original IP packet, or tunnel mode, i.e. original IP packet, including AH, is hidden behind a new IP header.

Using the AH protocol, in tunnel mode and between two wireless devices, e.g. wireless laptop and firewall, may provide a robust form of authentication, as a result of using MAC_2s. However in order to fulfill the additional requirement for data confidentiality and to mitigate the risk, associated with traffic analysis, the use of ESP is often preferred.

Whereas the ESP protocol provides data confidentiality through symmetric encryption of the IP payload (transport mode) or of the entire original packet (tunnel mode), protection from traffic analysis is only supported in tunnel mode. The implementation of ESP in either mode may prove useful in a wireless environment. Authentication at the transport-level can be achieved using ESP in transport mode (between end devices, such as a wireless device and AP/base station). However, it is susceptible to traffic analysis. On the other hand, ESP in tunnel mode (between a wireless device and firewall), often referred to as a VPN, can be used to authenticate devices prior to granting access to the resources of the wired network.

Transport Layer - SSL/TLS

Unlike IPSec, which can provide a *generic* authentication solution that is transparent to end users and applications, Secure Sockets Layer (SSL) and Transport Layer Security (TLS) - internet standard of SSL, have been carefully designed to mitigate

the increased risk, associated with Web-based applications, e.g. electronic commerce. It uses TCP to provide Point to Point Protocol (PPP) security services, namely authentication and confidentiality, between two entities (client and server). As a result, application-level protocols, such as the HyperText Transfer Protocol (HTTP) and File Transfer Protocol (FTP), can benefit from these security services.

As with IPSec, authentication and encryption are based on a shared secret key that is established using the Handshake Protocol (one of the SSL protocols). The 3-way handshake (client-server, server-client, client-server) also permits the negotiation of a ciphersuite, which identifies the key exchange methods (e.g. RSA and Diffie-Hellman) and specific cryptographic algorithms (e.g. MD5, 3DES and RC4). Moreover, digital certificates are exchanged for the purpose of initial authentication and the exchange of the shared secret key. Finally, the use of a digital signature, although optional, could also provide non-repudiation services.

By using an SSL-enabled web browser, a secure connection to a WLAN authentication server can be established by a wireless device, as proposed by Peikari and Fogie [132]. Once a connection has been established, user authentication, through username-password or challenge-response mechanism, can then be initiated. A more streamlined version of SSL, the Wireless Transport Layer Security (WTLS), has been incorporated into the Wireless Application Protocol (WAP). This protocol is used extensively for resource-constrained devices. An overview of WAP and WTLS is provided in section 2.1.4.

Application Layer - Remote Authentication Dial-In User Service (RADIUS)

Although there are no technical barriers, which prevent applications from using the security services at the TCP and IP layers, some security-oriented applications have been developed independent of these services. RADIUS is one such application, which authenticates remote connections, authorizes access to appropriate resources and logs pertinent information for tracking and billing purposes [132]. Initially designed to authenticate remote users, using modems, and to authorize access to network resources,

it has evolved to accommodate authentication requests from VPNs and WLANs.

One of the most popular brands of RADIUS servers is the Funk's Steel-Belted Radius Server (SBRS). As with other versions, it fulfills the need for a centrally managed authentication service and leverages the existing infrastructure (use of a single database) to accommodate WLAN users. When a user connects to the wired network, using an AP, firewall, VPN, Remote Access Server (RAS) or any other RADIUS-compliant network access device, his/her device, acting as a RADIUS client, submits a user-authentication request to the SBRS. The SBRS authenticates the user, according to one of several mechanisms, the default being username-password. It also authorizes access to the appropriate type of connection or service.

In addition to authenticating WLAN users, the SBRS also addresses the need for securing wireless connections. To this end, it supports the Extensible Authentication Protocol (EAP) [7]. It is a transport protocol that is used by the 802.1x authentication protocol [119], for the negotiation of connections between WLAN users and an AP. SBRS not only supports EAP's authentication methods (EAP-MD-5 and EAP-Cisco Wireless) but also requirements for key generation and exchange. Moreover, support for EAP permits the use of other authentication mechanisms, namely smart cards and digital certificates. In comparison to the user/password procedure, these mechanisms are less vulnerable to forgery. Finally, the use of time session limits and other attributes allow for a more granular and robust form of authentication in WLANs.

In the following section, the current state of intrusion prevention is assessed, by examining the authentication protocols that are analogous to *front door* security. They provide the means to authenticate users, processes, hosts and devices. In addition, the key weaknesses, exhibited by these protocols, as well as proposed resolution strategies are presented. Finally, specific attacks (i.e. breach of back door or *peripheral* security), which exploit these vulnerabilities or have surfaced, as a result of environmental characteristics, are presented in chapter 3, under the umbrella of intrusion detection.

Fig. 2.2 presents a categorization of the wireless networks, within the scope of this

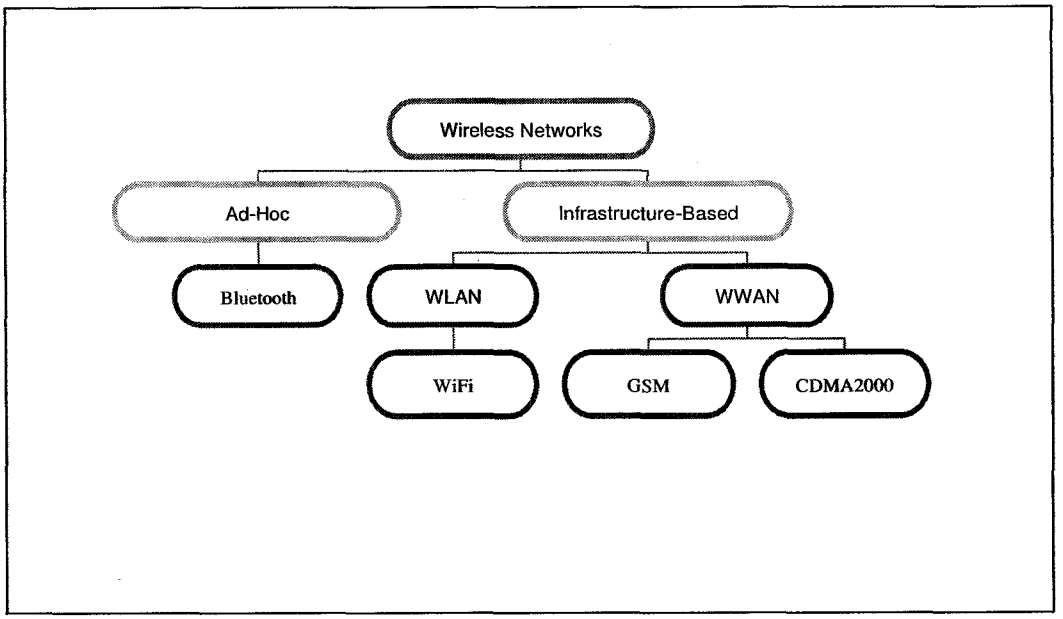

Figure 2.2: Wireless networks

research project, based on their primary architecture (e.g. ad-hoc, infrastructure). Nevertheless, many of them, e.g. BT and 802.11, can be deployed in either mode.

Whereas the term *connectionless* is used to refer to data-oriented networks, *connection-oriented* is typically associated with voice-oriented networks, although the distinction has become convoluted over the last few years. For example, Voice over Internet Protocol (VoIP) is implemented using a packet based mechanism.

2.1 Bluetooth

As depicted in Fig. 2.3, ad-hoc wireless networks are characterized by the lack of an infrastructure. Consequently, each Mobile Station (MS) or Mobile Node (MN) communicates with other nodes using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). However, in BT [64], MAC is provided by the *master*, which uses Time Division Multiple Access (TDMA) to poll each node. Although this form of control is also evident in infrastructure-based networks, the master node is identified in a dynamic manner, i.e. node that initiates communication, and hence is not technically a component of pre-established infrastructure. Nodes can communicate with

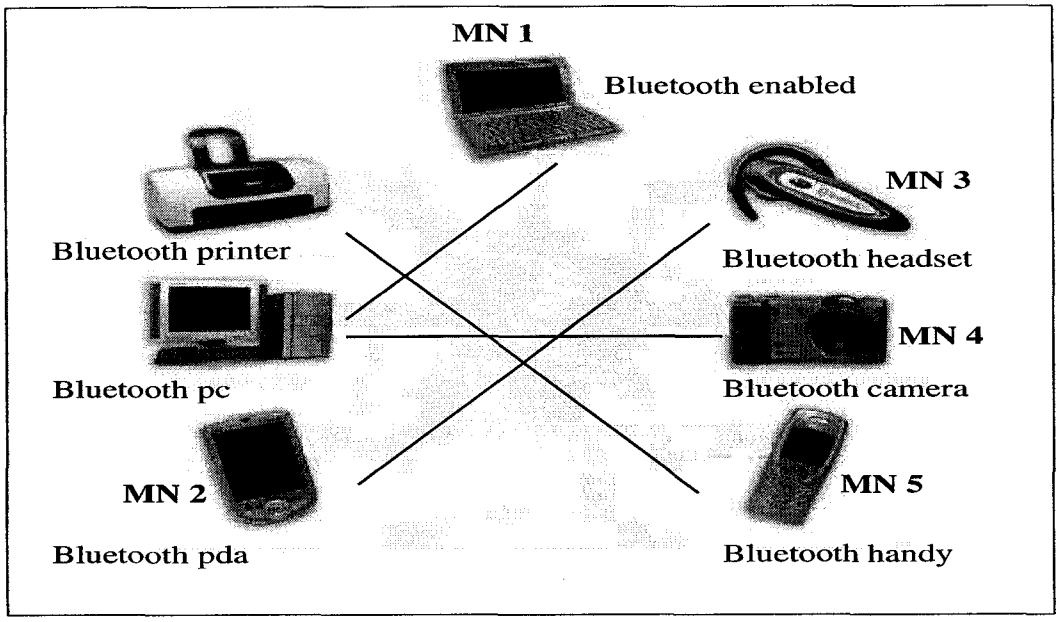

Figure 2.3: Ad-Hoc wireless network

one another as long as they are within each other's radio range (approximately 10-100 meters) or if other nodes can forward/route the messages.

One of the most well known examples of ad-hoc networking is BT. The term Bluetooth refers to an open specification, which enables short-range (10-100 meters) peer-to-peer wireless communications of voice and data, based on proximity networking. Specification 1.0B (Dec. 1999) was developed by the Bluetooth Special Interest Group (SIG), formed in May 1998. Some of the biggest players in the SIG include 3Com, Microsoft and Motorola. The key objectives of the SIG were to eliminate the need for cables associated with today's pervasive devices and to promote the formation of ad-hoc networks, using portable devices in the 2.4 GHz unlicensed band, as discussed by Miller [118].

While technologies such as 802.11 [54] and HomeRF [67], can fulfill the minimum requirements for WLANs (high data rates and a comprehensive set of networking features), BT is the preferred choice for PANs. It supports both voice and data communications, as identified by Shorey and Miller [158]. Initially designed as a cable replacement technology, it offers wireless connectivity between computers, PDAs, digital phones, printers, mobile phones and other devices.

Cable replacement usage models or potential applications, identified in the specification, include the cordless computer (i.e. wireless links used to connect peripherals to the computer) and the automatic synchronizer, which synchronizes data between two devices using proximity networking. One of the most interesting usage models, supported by ad-hoc networking, is the interactive conference. Using this model, business cards, files and objects can be quickly exchanged among the participants. As with 802.11, the BT specification does support infrastructure-based networking (using a Data Access Point (DAP)). It is particularly useful for realizing the direct network access usage model, which enables devices to access the Internet by first accessing the Local Area Network (LAN) via the DAP [118].

In order to provide an inexpensive alternative for ad-hoc networking and to support ubiquitous computing, this technology enables the design of low-cost, low-power and small-sized radios. These radios can be embedded into current and future devices that are equipped with small batteries. In addition, operating in the 2.4 GHz unlicensed band permits BT devices to be used anywhere in the world [63].

A brief analysis of the authentication protocol, defined in BT specification 1.1, reveals flaws, which must be addressed, if one hopes to instill user confidence and to expand the application of this technology. Of course, the 802.15 working group's strategic decision, to adopt a subset of the BT specification, should encourage a widespread adoption of BT [61], [60]. Currently, there are cell phones and other devices, such as printers and cameras, that are equipped with BT.

As depicted in Fig. 2.4, the BT protocol stack is divided into three main groups [118]. The transport protocol group (physical and data link layers of the Open Systems Interconnection model) is responsible for the transportation of audio and data between BT devices. It consists of the radio, baseband, link manager, logical link and adaptation protocols and the host controller interface. In terms of the protocols in the middleware protocol group, they make use of the transport protocols, while providing a standard interface to the application layer. Finally, the application group represents applications including those, which instantiate the BT profiles (discussed in section 2.1.4). These applications access the protocols in the middleware protocol

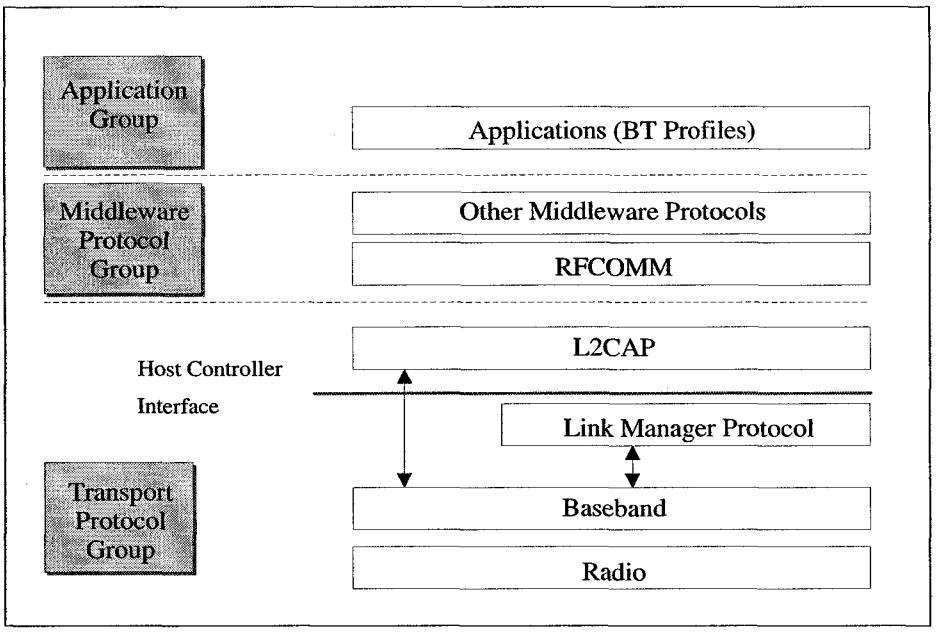

Figure 2.4: BT protocol stack

group, which in turn, exploit the protocols in the transport protocol group.

2.1.1 Security Services

The need for security services such as authentication and encryption is of paramount importance, especially in an RF-based ad-hoc setting. Having recognized this fundamental requirement early on, the BT SIG has taken appropriate measures to ensure the quality of the specification.

Whereas the baseband protocol defines the security algorithms and procedures for carrying out *device* authentication and link encryption (optional), it is the link manager that oversees the application of these security services.

Device authentication The unidirectional or mutual authentication of BT devices is carried out by the link manager, as described in the sequel.

Confidentiality Encryption of the payloads of packets, at the link layer, is carried

out using a stream cipher with four linear feedback shift registers. It is also based on a secret key (8-128 bits) that is shared by a pair of devices. The variable length of the key is intended to accommodate the security requirements of different applications. It is negotiated between the applications that reside on the participating devices (e.g. master and slave). As applications have a minimum key size, the negotiation process is terminated if this size cannot be supported by both devices.

2.1.2 Device Authentication Protocol

Authentication of a BT *device* (e.g. slave S) by another (e.g. master M) is based on a challenge-response mechanism, which requires the following parameters [61]. If there is a requirement to authenticate the user of a device also, it can only be fulfilled by using application level security, as stated by Traskback [179].

Parameters used in the authentication procedure

- Address of BT device (BT_ADDR) - 48 bit IEEE Address, unique to each BT device
- Shared secret key or link key - 128 bits
- Random Number (RAND) - 128 bits

Case I - Link key not available

If S and M have never communicated with one another, a shared link key must be established. It is not only used during the authentication procedure but also serves as input to the link encryption key generation process.

Pairing Process

In order to generate a link key, the devices undergo a *pairing* process. The end of this process is marked by the existence of the initialization key. This key is used subsequently by S and M to encrypt data during the link key generation process.

Link key generation process

Once the pairing process has been completed and K_{init} is available to both devices, the link key can then be generated using one of two options.

Link key generation using unit key

If one of the two devices, e.g. S, is memory-constrained, then the unit key (128-bit long-term private key) of S is used as the link key between S and M. The unit key is encrypted by S, using K_{init}, and transmitted to M. Finally, as with the pairing process, mutual authentication takes place to confirm the use of the correct link key. As a side note, once the unit key has been exchanged securely, the initialization key is discarded.

Link key generation using combination key

Option two permits the generation of a link key, based on the properties of both devices. Thus, the link key represents a stronger relationship between both devices.

The link key generation process is similar in principle to the pairing process, in that the combination key K_{SM} or K_{MS} is first computed, followed by mutual authentication of both devices.

Case 2 - Link key is available

In order to expedite the authentication process in the future, link keys can be stored in BT devices. Hence, the pairing process becomes unnecessary. The devices simply perform mutual authentication, as in Step 3 of the link key generation process.

2.1.3 Weaknesses and Resolution Strategies in Authentication

Unlike the mass publicity and ensuing research initiatives, which had been undertaken to address the vulnerabilities of 802.11 Wired Equivalent Privacy (WEP) protocol (discussed in the sequel), the weaknesses in the BT authentication protocol have not attracted as much attention. One possible explanation would be that the popularity and increased deployment of 802.11 WLANs would have had organizations seeking

ID	Weaknesses	Resolution strategies
BT-W01	Selection and distribution of PINs	Application-level encryption, Longer PINs and other strategies
BT-W02	Vulnerability of the unit key	Various strategies
	BT-W represents a weakness	

Table 2.1: BT: Authentication weaknesses and resolution strategies

immediate solutions to protect their investment. On the other hand, the BT specification had always been subjected to public scrutiny, and as a result, the remaining weaknesses could be considered relatively less significant.

Moreover, although developers of the BT specification (version 1.0b) had been aware of the weaknesses in the authentication protocol, the core specification (version 1.1) has remained unchanged. I suspect that the key decision-making factor was the need to maintain backward compatibility. Furthermore, the correct use and implementation of a given protocol is typically outside the mandate of the specification body. Table 2.1 lists the key weaknesses and the proposed strategies for addressing them. Please note the use of the term BT-W#-R#, for identifying resolution strategies (-R#), which are associated with a given weakness (-W#).

Weakness in the pairing process

Unlike 802.11, BT uses different keys to carry out authentication and encryption. Thus, practical studies on the security of BT have focussed on the discovery of the keys, and the most opportune moment to launch these attacks is during the initial pairing process.

BT-W01: Selection and distribution of PINs

The key problem, associated with the pairing process, is the selection and distribution of Personal Identity Number (PIN)s. One of the complaints, voiced by users, is the need to enter a PIN twice, every time two devices are required to communicate with one another [182]. Thus, there is a high tendency to use the shortest PINs possible.

BT-W01-R01: Application-level encryption

As the specification does not define a precise mechanism for the distribution of PINs, various strategies have been adopted. One option is to transmit a PIN (any length) in the clear. It is obviously a poor choice since it can be captured by an attacker. Even if it is encrypted, using application level encryption, before being sent (out of band), it can still be discovered. According to Jakobsson and Wetzel, off-line PIN crunching and using brute-force represent two mechanisms that are facilitated by eavesdropping and stealing by participation [92]. The discovery process is further simplified by the fact that most users often use 4-digit PINs and that 50% of PINs used are set to 0000 [182].

BT-W01-R02: Longer PINs

This weakness can be addressed by using relatively simple techniques. One way to minimize the discovery of PINs is to use longer PINs, as recommended in version 1.1 of the specification [65]. It has been suggested, by Jakobsson and Wetzel, that a PIN of 64-bits should be secure, providing that users select those, which are uniformly random [92]. While this suggestion is technically feasible, the usability factor may be of concern. Given that the PIN has to be entered twice on unpaired devices, users may find it unacceptable, unless both the selection and distribution of longer PINs are automated. In order to address this issue, the BT specification suggests the use of application-level key agreement software, e.g. Diffie-Hellman [166], which permits the generation of longer PIN codes (16 octets) and eliminates the need for manual entry. In fact, this solution would be suitable for devices that are equipped with limited data entry mechanisms.

BT-W01-R03: Other strategies

It has also been suggested that, in addition to the use of longer PINs, the pairing process be initiated only in a private place, a requirement that may be hard to fulfill under certain circumstances. Another option, proposed by Drabwell [41], is the use

of a shared secret PIN for generating the necessary keys for device authentication and encryption of links. Finally, the use of a Faraday's cage (a metal coated plastic bag), considered a physical-layer remedy, is proposed by Jacobsson and Wetzel [92]. According to the authors, it can be used to prevent an attacker from detecting signals that are transmitted by a victim's device.

BT-W02: Vulnerability of the unit key

Once a PIN has been discovered and used to generate the K_{init}, obtaining the unit key (also used as a link key) is straightforward. As you may recall, the unit key of a memory-constrained device is typically encrypted, using the K_{init}, before it is transmitted to another device. Another approach is to initiate communication with a memory-constrained device in order to obtain the unit key for nefarious purposes.

The following resolution strategies (BT-W02-R01 and BT-W02-R02) address the use of unit keys as link keys. On the other hand, the last two strategies (BT-W02-R03 and BT-W02-R04) can be adopted for mitigating the risk, associated with a man-in-the-middle attack.

BT-W02-R01: Use of pseudo random number generator

In terms of protecting unit keys and minimizing the impact, in case of a compromise, there are two options, which can be exercised. Option one makes use of a set of keys, a different one for each device (similar to the concept of a combination key but without the overhead). Unfortunately, it defeats the principle behind the use of the unit key, i.e. limited memory. However, the second option is more sensitive to the characteristics of BT devices. It requires that the unit key be used as an input parameter to a Pseudo Random Number Generator (PRNG). In particular, if a device uses the address of another to seed the PRNG, then a link key can be generated dynamically, thus minimizing the requirements for memory [92].

BT-W02-R02: Use of combination key

While it is rather obvious, the recommendation made by the standards body, and reflected in version 1.1 of the specification, is to refrain from using the unit key whenever possible. Instead, the use of a combination key is encouraged [65].

BT-W02-R03: Certificate-based solution

Although the use of application level security, e.g. a standard certificate-based authentication system, may prove beneficial as a defense against man-in-the middle attacks, it may not be practical. According to Miller [118], public-key and certificate-based schemes are not appropriate for ad-hoc networks. For example, these schemes are dependent on trusted authentication agencies, which may not be available.

BT-W02-R04: Unpredictable frequency hopping

Another approach for mitigating this attack is to increase the difficulty in locking onto the frequency used for communication. As suggested by Hager and Midkiff [69], it can be accomplished, using frequency hopping intervals and patterns that are unpredictable. A seed, which is a function of the clock and BT device address of a master, is used for generating the hopping sequence. This strategy also requires that it be changed periodically, and communicated to the slaves in a piconet.

2.1.4 Link Layer/Higher Layer Solutions

It is important to keep in mind that the two security services, to be provided by BT, are device authentication and link-level encryption. Consequently, the core specification does not address the need for user authentication, message authentication or end-to-end encryption. However, it does encourage the use of existing security, implemented at the transport, session or application layers [192].

In order to provide guidance to application developers, the BT Security Expert Group has developed security architecture models [65]. These models identify appropriate levels of security, based on the profiles, sensitivity of the information exchanged,

and user requirements. Profiles define a standard set of messages and procedures (based on BT SIG specifications) in order to promote interoperability between different implementations of the underlying protocol stack.

Profile-specific solutions

Given the large number of profiles, defined in volume 2 of the specification, two of the most common ones have been selected for discussion purposes. They demonstrate the need for authentication at multiple layers.

BT-S01: Automated Synchronization Profile

Synchronization is an application, which permits Personal Information Management (PIM) data to be synchronized between any two entities, including devices and networks [64]. This profile supports unidirectional pushing or pulling of objects. The synchronization procedure is carried out, using the following steps, as per Telecom/IrMC specification [17]:

- a client, which is typically a PC, pulls the data from a server, usually a phone or PDA;
- it then synchronizes the data against its local objects; and
- pushes the synchronized data back to the server.

What is interesting about this model is the use of a more powerful device to represent the client, instead of the server. Since the synchronization engine requires sufficient storage and processing capacity and it is the client, which synchronizes the data, it must be able to fulfill these requirements. The server, on the other hand, supports an object exchange service, which is based on the General Object Exchange Profile (GOEP).

The *automated* aspect of this profile permits client-initiated synchronization to proceed without user intervention. However, only bonded or paired devices can synchronize automatically when they are within the vicinity of one another (also referred to as proximity networking).

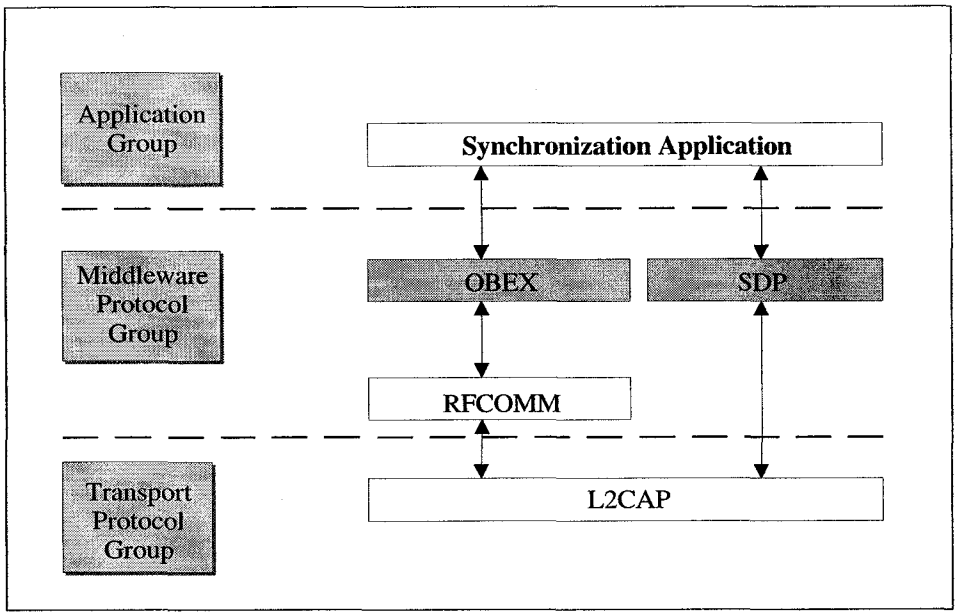

Figure 2.5: BT synchronization protocol stack

In terms of additional security services, associated with the synchronization protocol stack, see Fig. 2.5, they are provided by the Object Exchange (OBEX) protocol [62]. The authentication service, provided by OBEX, is based on a challenge-response scheme using MD5 and passwords, which are stored in both devices.

Thus, while device authentication and link encryption are provided by the BT baseband protocol, the OBEX protocol also supports authentication. This form of authentication prevents an attacker from assuming the role of a client or server, and ultimately, gaining unauthorized access to the synchronization service.

Advantages:

The key advantage of using a multi-layered (transport and middleware) authentication scheme is the increased difficulty in circumventing it. So, while the device authentication mechanism, at the baseband layer, could fail due to unforeseen circumstances, it would not necessarily result in unauthorized access to the service.

Disadvantages:

As far as disadvantages are concerned, employing two levels of authentication, using potentially two different algorithms, could translate to increased processing requirements and latency.

BT-S02: LAN access profile (to be replaced by the PAN profile)

As indicative of the name, the LAN access profile (LAP) permits wireless devices to access the resources of the wired LAN through a DAP, similar in principle to that of 802.11 networks [64]. However, it uses the IETF PPP over the BT link, which has been established between a device (slave) and a DAP (master). Traffic, related to IP and other network protocols (e.g. WAP), can then flow over PPP. Furthermore, PPP can be implemented over serial connections, that are managed by the RF Communications (RFCOMM) protocol, see Fig. 2.6.

Although a more general and IP-based networking solution would provide additional benefits, due to time constraints, the adoption of the PPP protocol would have represented a more practical strategy. Since PPP is a widely used Internet standard, and that many devices, including PDAs, support IP communications over PPP, the strategic direction taken by the SIG's networking group seems reasonable.

As with the previous profile, the device authentication service of the baseband protocol is used for providing access control to the wired LAN (via the DAP). Since authentication is based on PINs, network access could be restricted by divulging the PIN only to authorized devices (slaves). On the other hand, the use of default (zero length) PIN would prove useful for providing non-authenticated access in a public domain. You may have already recognized a similarity between this strategy and the use of shared-key authentication in 802.11.

In addition to baseband layer security, user authentication schemes [159], associated with PPP, can also be employed. One such configuration has the following components: a DAP and wireless device of a user, which also acts as a RADIUS client. The client module connects to a RADIUS server on the wired LAN. The

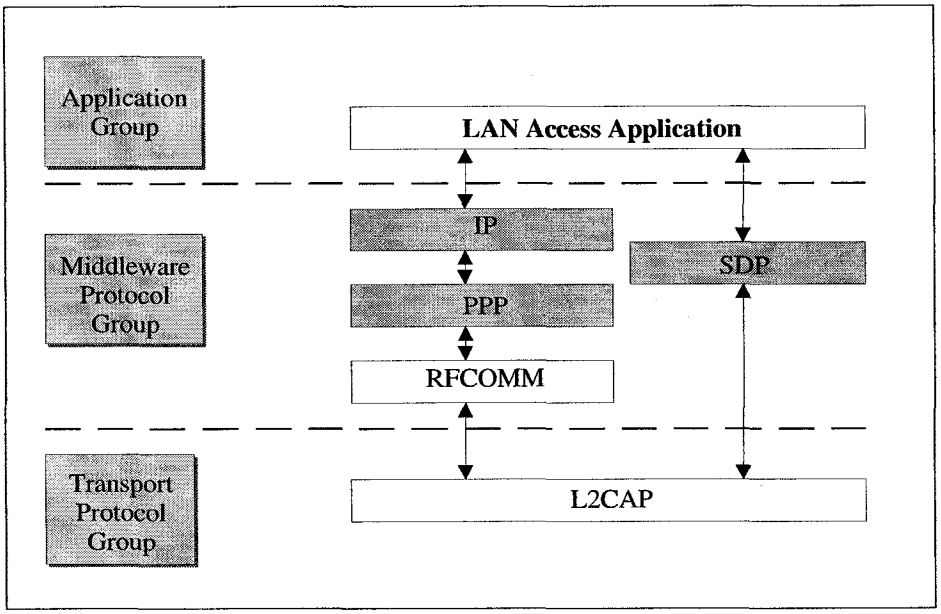

Figure 2.6: BT LAN access protocol stack

server, in turn, makes use of an authentication database in order to authenticate the user of the device, using the username-password mechanism.

Advantages:

Using a RADIUS server for user authentication is a common approach that has been readily adopted by many organizations. Moreover, with the increased deployment of 802.11 WLANs, there is a high probability that the required infrastructure may already be in place. Hence, the implementation cost, associated with this profile, is likely to be minimal.

Disadvantages:

If, on the other hand, such an infrastructure does not already exist, this solution could exact a higher price. It would also suffer from the same disadvantages as the previous profile. Of course, the degradation in performance will depend on the type

of application and resources being utilized.

Link layer solution

BT-S03: User and device Authentication

As aforementioned, when user authentication is required, it is typically carried out at the application layer. However, in order to initiate the authentication mechanism at this layer, a BT device must be in security mode two. This necessarily precludes the use of link-layer security services, such as device authentication.

In order to realize the benefits of device and user authentication, i.e. permit device A to determine if device B has been stolen and is currently under the control of an attacker, Nguyen *et al.* [125] propose a Password Mutual Authentication and Key Agreement Protocol (PMAKAP). Intended to leverage the BT device authentication mechanism, the security services provided by the protocol include secure user authentication and authorization, secure pairing process and message authentication.

In terms of user authentication, PMAKAP makes use of the Secure Remote Password (SRP) protocol, described by Wu [191]. It is a logical choice given that the cost of computation and communication is low, an essential factor for power-constrained BT devices. As far as service authorization is concerned, a Role-Based Access Control (RBAC) approach, defined by Sandhu and Samarati [150], is used to model different security policies and to simulate traditional methods. Employing RBAC, instead of an ACL, lowers the storage requirement from $O(cs)$ to $O(c+os)$, where the number of users is represented by c, services by s and roles by o. Lastly, message authentication is implemented using a 20-byte MAC_2, which is incorporated into each RFCOMM message. The MAC_2 itself is created using the session key K derived at the end of the PMAKAP.

Procedure

Before the details of the PMAKAP are disclosed, a brief description of SRP is warranted. The SRP operates in Z_n^*, where n represents a large prime number. All of the computations, associated with this protocol, are carried out modulo n.

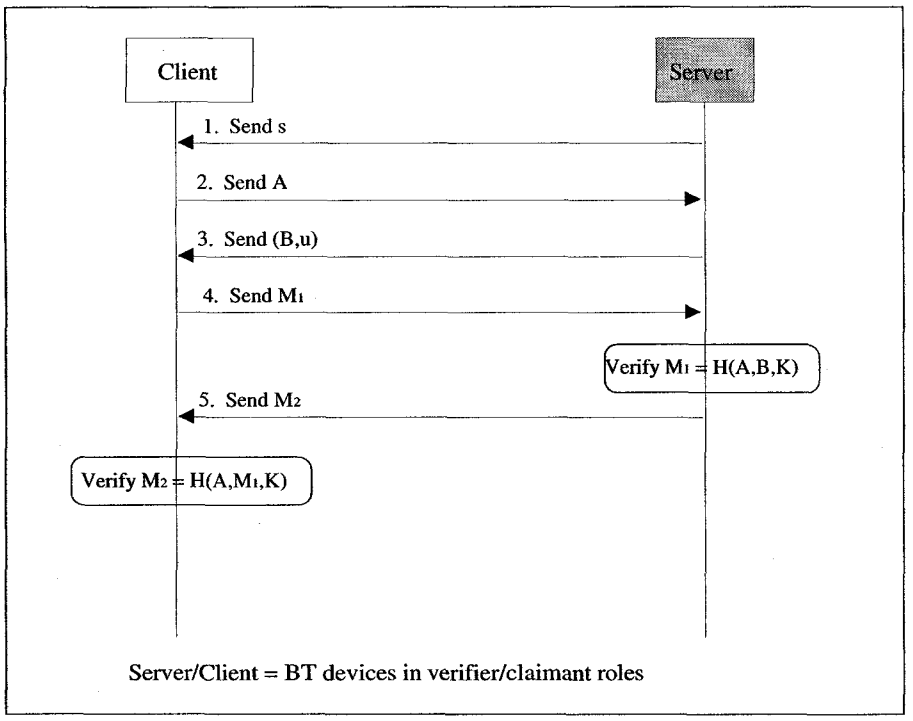

Figure 2.7: Password-based Mutual Authentication and Key Agreement Protocol

A server stores a client's public key v in its database, where $v = g^x$, x is the client's secret key and g represents a primitive root. In addition, x is obtained from $H(s, P)$, with s denoting a random string used as the client's salt, P the client's password and H a one-way hash function.

Fig. 2.7 illustrates the application of PMAKAP, where a client C (or claimant) wishes to be authenticated by a server S (or verifier).

1. Upon receiving an authentication request by C, S finds (s, v) and sends s to C (step 1). The salt is used by C to generate g^x prior to step 4.

2. C generates $x = H(s, P)$ and $A = g^a$, where a denotes an Ephemeral private key (generated randomly), and transmits A to S (step 2).

3. S, in turn, computes $B = v + g^b$, where b denotes an Ephemeral private key (generated randomly), and generates another random number μ. It subsequently sends (B, μ) to C (step 3). Both $B = (g^x + g^b))$ and μ are used by C to derive

the session key K. In particular, μ serves as a nonce for the prevention of replay attacks.

4. C computes $T = (B - g^x)^{a+\mu x}$ and $K = H(T)$. In fact, $T = (g^x + g^b - g^x)^{a+\mu x}$ which is equivalent to $T = (g^b)^{a+\mu x}$. The session key K, to be used for this session, is obtained by hashing the value of T.

5. In a similar manner, S computes $T = (Av^{\mu})^b$ and $K = H(T)$. Given that $T = (g^a * g^{x\mu})^b = (g^{a+\mu x})^b$, which is equivalent to $(g^b)^{a+\mu x}$ calculated by C, S also computes an identical session key K.

6. In order to be authenticated by S, C computes $M_1 = H(A, B, K)$ and transmits it to S (step 4).

7. S authenticates C by determining if M_1 is equivalent to the value of $H(A, B, K)$, which has been previously calculated by S.

8. To fulfill the need for mutual authentication, S computes $M_2 = H(A, M_1, K)$ and transmits it to C (step 5).

9. C, in turn, authenticates S by verifying if $M_2 = H(A, M_1, K)$.

10. At the completion of the protocol, both S and C share the session key K.

Advantages:

One of the key advantages of PMAKAP is that it is executed at the link layer (security mode three), instead of being associated with a given application (security mode two). Thus, as long as a BT device operates in mode three, both BT device authentication as well as the security services, provided by this protocol, will remain activated.

Disadvantages:

The drawback of using a link-layer protocol is that not all applications will necessarily require the security services offered by PMAKAP. Under these circumstances, the security overhead, associated with PMAKAP, would be unwarranted.

Another potential disadvantage is the storage requirement for clients' public keys. This would require that a BT device, with sufficient storage capacity, assume the role of the server/verifier. Thus, the application of PMAKAP is somewhat limited to scenarios/profiles, where a server (BT device) is readily available. Having said that, with the introduction of new technologies, this issue may no longer be significant.

2.2 WiFi/802.11

The key difference between an ad-hoc and infrastructure-based WLAN is the continued presence of an AP, which acts as a bridge between wireless or MNs and network hosts (NH). In terms of communication, MNs communicate with one another via an AP (see Fig. 2.8), which assumes the role of a coordinator for MAC. In fact, the AP is equipped with most of the networking functionalities, including the provision of Quality of Service (QoS), synchronization, support for security and bridging capabilities.

Although the radio coverage of a single network is approximately 100 meters, by placing several APs near one another and connecting them using a distribution system, a larger wireless network can be created. This feature provides users with roaming capabilities within the extended network, as indicated by Schiller [151].

The IEEE standard (1997) specifies the physical (PHY) and MAC layers, which have been customized based on the requirements of WLANs. The primary goal of the standard is to define a robust WLAN capable of supporting time-bounded and asynchronous services. While the MAC layer is primarily responsible for medium access control and fragmentation of user data, the MAC management component supports the association/re-association of a MN to an AP, roaming between APs, authentication, encryption and power management. The PHY layer, on the other hand, supports both infrared and radio-based transmission in the 2.4 GHz band [151]. Since the initial ratification of the standard, new PHY layers, supporting data rates of 11 Mbps (802.11b) and 54 Mbps (802.11a/g) have been defined.

The popularity of WLANs is clearly reflected in the ever-expanding customer base.

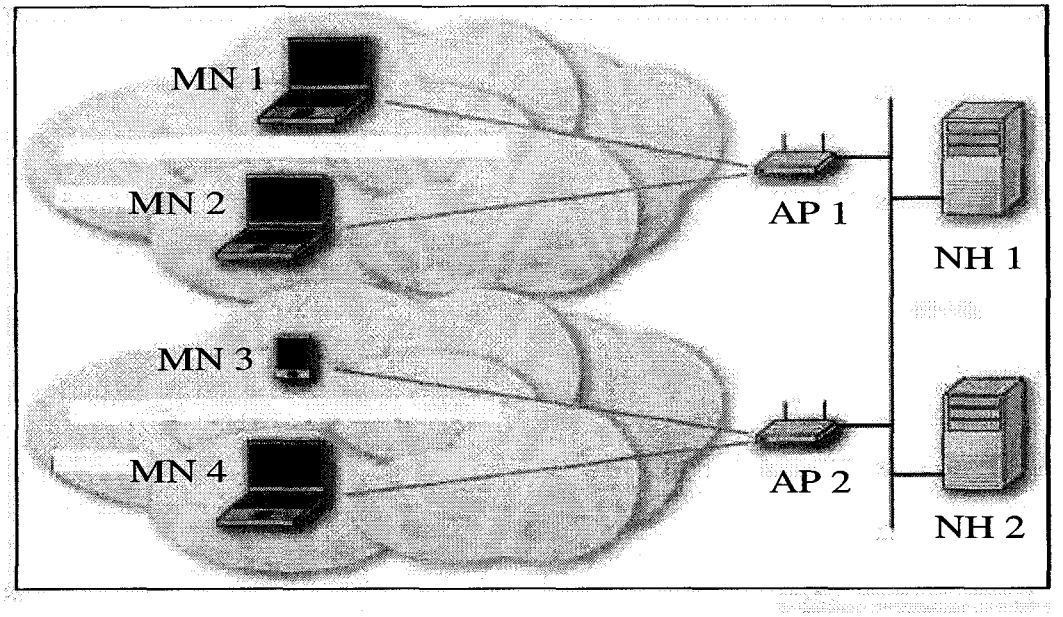

Figure 2.8: Infrastructure-based WLAN (802.11)

Jupiter Research predicts the deployment of 99 million corporate 802.11 devices, e.g. adapter cards and APs, by 2008. Meanwhile, International Data Corporation estimates a 57% annual growth of WLAN locations over the same period [154]. In line with this prediction, enterprises have deployed wireless networks in public domains, such as airport lounges, coffee shops, shopping malls and hotels. This lucrative undertaking keeps mobile employees connected to the home office [186]. Moreover, several ventures have implemented 802.11b portals in order to provide value-added services to customers. One example is the transmission of boarding calls to a wireless device, when a plane is ready for boarding.

Although the WLAN industry was expected to grow from $1.1 billion in 2000 to $5.2 billion in 2005, as predicted by Abramowitz [8], one may question the degree of growth, given the inherent security weaknesses in the standard. While reduced cost of ownership, scalability, installation speed and flexibility, and user mobility provide significant benefits to the organization, there is an expectation that the risks can be mitigated, using a layered approach to security.

2.2.1 Security services

As previously stated, it is the responsibility of the MAC management layer to provide the following services [129]:

Authentication Support for unilateral authentication of wireless devices is provided using the open authentication and shared key authentication schemes (discussed in the sequel).

Confidentiality The WEP specification also accommodates the need for user confidentiality. A pseudorandom generator, which takes a 40-bit secret key as input, is used to create a key sequence that is XOR-ed with the payload of each frame. The fundamental goal of WEP is to prevent casual eavesdropping.

Data integrity Through the use of the integrity checksum field, data integrity can be validated.

2.2.2 Authentication protocol

The WEP protocol, employed by the 802.11b standard, does support two techniques for authenticating devices [55], in addition to the encryption mechanism.

When a WLAN is deployed using the infrastructure mode, i.e. using APs that are connected to a distribution system, authentication is based on the *open system* (default) or *shared key* principle. As a side note, WEP is not supported in Ad-Hoc mode) [103].

The use of the null-authentication option (open system) permits all users to access the WLAN [136]. Although a typical reaction would be to question the motivation for selecting this option, it is an appropriate choice, under certain circumstances. For example, link-level authentication may not be required if higher-layer security mechanisms are in place [192]. This strategy is commonly adopted by publicly accessible WLANs, such as *hot-spots*. However, in a majority of cases, administrators usually prefer this option, when ease of administration is the primary factor.

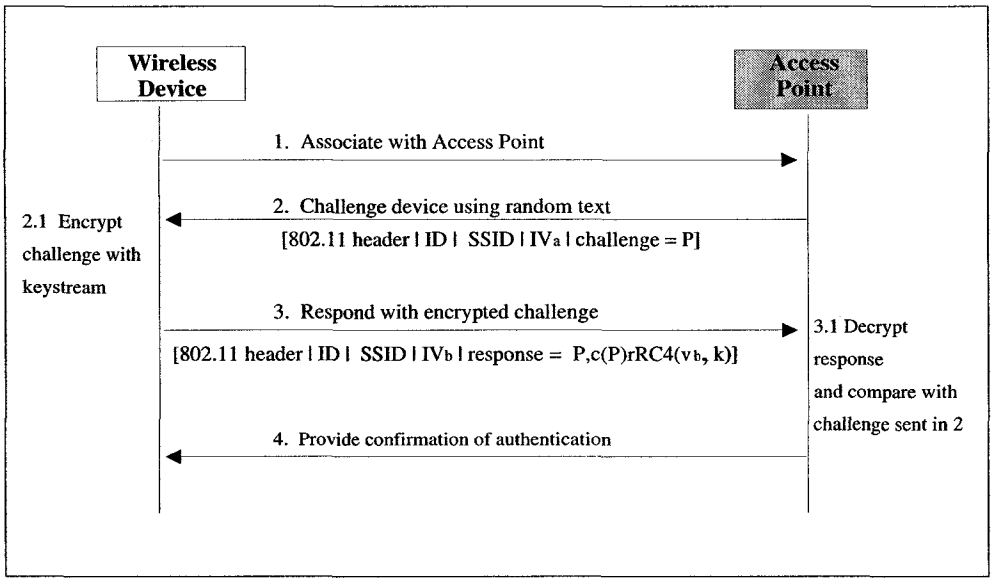

Figure 2.9: IEEE 802.11b Shared key authentication

Nevertheless, in order to provide some form of authentication/access control to the wireless network, Access Control Lists (ACLs), using MAC addresses, are often implemented at the APs. A more stringent form of access control is provided by *closed network* authentication, a proprietary solution by Lucent [67]. With this solution, the identifier of a Service Set Identifier (SSID) is configured into an AP. Therefore, in order to associate with a given AP and to ultimately gain access to the network, a MN must have knowledge of this identifier (shared secret).

The more secure version of the protocol (shared key authentication) is based on a secret key, which has been programmed into all authorized MNs and APs of a WLAN. Authentication of devices proceeds according to Fig. 2.9, and it is based on the challenge-response mechanism and symmetric encryption.

When a MN or wireless device attempts to associate with an AP, it sends an authentication request management frame, specifying the use of *shared-key* authentication (step 1). The AP responds by returning an authentication management frame

that contains a challenge (step 2). The components of the WEP frame consist of: [802.11 header | ID | SSID | IV_a | *challenge* = P], where ID and SSID represent the identifiers of the wireless device and AP respectively, and IV is the initialization vector. In most cases, it is the MAC address that is used as an identifier for both. As far as the IV_a is concerned, it is a 24-bit value used primarily to achieve and maintain synchronization between devices and an AP. This value is typically changed with each frame. The subscript has been used to highlight the fact that the Initialization Vector (IV) values, associated with the challenge and response frames, are different. Finally, the challenge or plaintext P is a 128-bit random text. It is generated using the WEP pseudo-random generator, which takes the IV and shared-key as inputs. It is important to note that all components of the frame are transmitted in the clear.

In response to the challenge, the device encrypts P using the WEP encryption mechanism (step 2.1). It essentially involves the bit-wise modulo-2 addition (bit-wise exclusive-OR) of the keystream (long sequence of pseudorandom bytes) with the data to be encrypted.

The response (ciphertext) is subsequently transmitted to the AP (step 3) using a similar frame structure: [802.11 header | ID | SSID | IV_b | *response* = $(< P, c(P) > \oplus RC4(v_b, k))$]. You may have already noticed that the key difference between the challenge and response frames are the *challenge/response* text and IV value.

In order to authenticate the device, the AP decrypts the response first and then calculates the ICV of the resulting plaintext, P (step 3.1). If the $c(P)'$ is valid, ie. data integrity has not been violated, the AP then compares P' to the challenge text P sent in step 2. If there is a match, the AP sends an authentication confirmation message to the device and provides access to the network [184]. On the other hand, if an incorrect response is sent to the AP, the device is prohibited from accessing the network.

An important element of this authentication protocol, the secret key, deserves more attention. It is primarily used as input to the WEP encryption mechanism. Therefore, while shared key authentication is preferable to the open system authentication mode, the benefits cannot be realized unless WEP encryption is enabled on

ID	Weaknesses	Resolution strategies
802.11-W01	Secret key	Public-key cryptography
802.11-W02	Initialization Vector	Non-shared secret keys
802.11-W03	Integrity Check Algorithm	SHA1-HMAC
802.11-W04	Use of malleable identifiers	Public-key cryptography
802.11-W05	Use of SSID	Configuration of AP, ACL
802.11-W06	Lack of network authentication	RADIUS
802.11-W07	Lack of mutual authentication	Link/Higher layer solutions

Table 2.2: WiFi/802.11: Authentication weaknesses and resolution strategies

the AP. One of the biggest problems, in securing WLANs, is the failure to activate an inherent security measure, such as WEP. Although not directly stated in the literature, I speculate that the degradation of throughput of approximately 16%, as identified by Janowski and Chang [93], may be a contributing factor.

From a security perspective, the use of a secret key, for both encryption and authentication purposes, certainly deviates from one of the primary tenets of cryptography. Consequently, the possession of this key could permit an attacker to not only access the network but to decrypt messages also.

2.2.3 Weaknesses and Resolution Strategies in Authentication

While the strength of the WEP algorithm is based on the difficulty of discovering the secret key, generally through brute-force attacks, as confirmed by Mishra and Arbaugh [55], this level of comfort is misplaced. In this section, the weaknesses of the shared key authentication protocol are discussed in detail. The vulnerabilities, associated with the open system authentication option, are also presented.

The flaws, associated with WEP encryption, have been widely publicized to say the least. Well known research papers, such as the Berkely report [24] and others [117] [79], provide a comprehensive treatment of the flaws and the attacks which exploit them. Please note that while issues, related to key management, are beyond the scope of this document, they are nevertheless addressed, by some of the solutions presented in this section.

According to Mishra and Arbaugh, one of the most significant flaws with the

authentication protocol is the use of the RC4 stream cipher for symmetric encryption [79]. As mentioned earlier, the RC4 algorithm XORs the keystream with the plaintext to generate the ciphertext. Given that the values of the fields of the WEP frames (challenge and response) are identical, with the exception of the IV, challenge (plaintext) and response (encrypted text) fields, it is straightforward to derive the keystream [16].

Within the context of shared key authentication, the possession of the keystream eliminates the need for an attacker to derive the shared key. Once the keystream, for a given IV, is known, the attacker can respond to future challenges until the IV is changed. At that point, a new keystream must be extracted.

Before proceeding any further, one should keep in mind that some of the weaknesses have been introduced, as a result of ambiguities in the standard. I concur with the opinion of Borisov, Goldberg and Wagner that the standards body should have expended more effort and time to clearly understand the ramifications of certain specifications. Furthermore, it would have proved beneficial had the standard been subjected to public scrutiny in a proactive manner, i.e. not expecting the research community to pay for it [24].

802.11-W01: Secret key

As the standard does not specify a mechanism (automated or manual) for the distribution of keys to different devices, in practice, most WLAN installations not only use a single shared key but fail to change it periodically. The consequences of this action are quite severe:

- The more users in a WLAN, the higher the probability of a compromise (shared key) even if administrators do not reveal the key, which has been used to configure the devices. Having to reconfigure all devices with a new key, as a result of a compromise, can quickly become a nightmare; and
- Attacks, associated with keystream reuse (IV is changed but secret key remains the same), become more feasible since the IV space (2^{24}) is often exhausted in

less than a day, as indicated by Harris [79]. According to Mehta, with many users sharing the secret key, this time interval can be as short as 1 hour before IV collisions occur [117]. This situation is further exasperated by the fact that the key is infrequently changed, if at all, thus permitting an attacker to analyze traffic for an extended period of time.

802.11-W01-R01: Public-key cryptography

A potential solution, which I believe may prove useful, is the use of public-key cryptography. It would not only accommodate authentication but also provide a mechanism for the exchange of secret keys. These keys could be regenerated on a per-session or per-frame basis.

802.11-W02: IV

The second input to the RC4 cipher is the IV. It is used to randomize the keystream. Not only is the number of bits allocated to this field (2^{24}) too short, but the standard does not require that a different value be used in each frame either. The fact that the standard does not specify a precise mechanism for the generation of the IVs, however, is understandable. It is generally accepted that standards specify the goals/objectives to be met and not necessarily the mechanism with which they are achieved.

In any event, this void in the specification has resulted in the following:

- It has encouraged vendors to implement elementary and predictable IV calculations that reduce the number of unique keystreams. For example, if the value of an IV is initiated to 0 when a device reboots, and is henceforth incremented by 1 each time the Personal Computer Memory Card International Association card initializes, then the device may be limited to the values in the range of 0-4. Hence, this strategy substantially decreases the number of distinct IV values to 32 (2^5) and reduces the time interval prior to the first IV collision; and

- The small space of IV values, combined with a static shared key, increase the feasibility of constructing a decryption dictionary/table [24] of IV values and corresponding keystreams. While this undertaking is quite laborious, such a table would permit an attacker to decrypt all ciphertext, associated with a given IV.

802.11-W02-R01: Non-shared secret keys

In order to minimize the reuse of keystreams, one can either use non-shared secret keys and/or increase the size of the IV field. Increasing the size of the key, from 40-bits to 128-bits (actually 104-bit key + 24-bit IV) would increase the level of complexity of brute-force attacks. However, it would not prove useful in minimizing the reuse of key streams since the 24-bit IV remains unchanged, as indicated by Walker [183].

802.11-W03: Integrity check algorithm

In order for a receiving device to validate the integrity of the transmitted plaintext, a checksum of the plaintext is created using a 32-bit Cyclical Redundancy Check (CRC)-32. Since CRC-32 is not a cryptographically secure MAC_2, it is less resilient to malicious attacks, such as message alteration (described in the sequel). This vulnerability is intensified by the fact that the plaintext is encrypted using a stream cipher [24].

Attack: Message alteration

Property of all CRC checksums: WEP checksum is a linear function of the message.

Thus, the checksum distributes over the XOR operation i.e. $c(x \oplus y) = c(x) \oplus c(y)$ for all x, y. As a result, it is possible to make controlled modifications to the ciphertext (by flipping specific bits) such that the decryption of the modified ciphertext will result in an altered message having the correct checksum.

Given that C corresponds to an unknown plaintext P,
$C = RC4(v,k) \oplus < P, c(P) >$ where c(P) is the checksum of P. Hence, in order

to calculate the new ciphertext C', which will ultimately be decrypted to P', an attacker only requires the original ciphertext C and the difference (Δ) between P and P'. As stream ciphers, such as RC4, are also linear, one can reorder many terms in the following equation. In particular, both sides of the equation are XORed with $< \Delta, c(\Delta) >$.

$$\begin{aligned} C' &= C \oplus < \Delta, c(\Delta) > \\ &= RC4(v,k) \oplus < P, c(P) > \oplus < \Delta, c(\Delta) > \\ &= RC4(v,k) \oplus < P \oplus \Delta, c(P) \oplus c(\Delta) > \\ &= RC4(v,k) \oplus < P', c(P \oplus \Delta) > \\ &= RC4(v,k) \oplus < P', c(P') > \end{aligned}$$

While it is possible to apply carefully formulated modifications to the ciphertext, without fear of detection, it is assumed that an attacker has knowledge of the keystream ($RC4(v,k)$), and therefore, capable of deriving the plaintext. This would permit her to determine Δ, by XORing the original and altered plaintexts, and to subsequently determine C'.

In the opinion of Harris, the ability to alter data, in transit, also opens the door for other types of attacks, namely IP redirection and message injection [79]. As indicative of the name, redirection of IP packets can be achieved through the modification of the source IP address. Hence, replies sent to the altered source address would be received by an attacker and not the original sender. Message injection is more or less an extension to the message alteration attack. The original plaintext, in this case, is replaced in entirety with a different message.

802.11-W03-R01: SHA1-HMAC

One possible solution, proposed by Borisov, Goldberg and Wagner [24], is to replace CRC-32 with a keyed MAC_2 such as the Hash Message Authentication Code (HMAC). Another possibility is to continue to use CRC-32 and RC4, which have

been selected for speed and implementation simplicity, but mandate the encryption of challenge texts by APs. A device or claimant would then decrypt the challenge, assuming that it has the key, and modify it according to predefined specifications. Finally, the CRC-32 would be applied, followed by the use of a keystream for the purpose of encryption. The benefit of encrypting the challenge text would be to increase the complexity of the attacks. Of course, the protocol would have to be changed accordingly.

802.11-W04: Use of malleable identifiers

The use of MAC address-based ACL, in open-system authentication, renders the access control mechanism vulnerable to MAC address spoofing. Please refer to section 3.2.1 (802.11-A03) for a description of this attack.

802.11-W04-R01: Use of public-key cryptography

In order to accommodate the use of MAC address-based ACL, yet remain invulnerable to MAC address spoofing, vendors have started to produce Network Interface Cards (NICs), which not only hold unique MAC addresses but unique public/private key pairs also [126]. Hence, as suggested by Nichols and Lekkas, this strategy would permit APs to authenticate devices, based on the combination of a MAC address and public/private key pair.

802.11-W05: Use of SSID

Even though the use of SSID (shared secret) was intended to provide enhanced access control, in comparison to the use of MAC address-based ACL, this approach also exhibits a key weakness. During normal operation, the SSID, which has been incorporated into management frames, e.g. beacon frame, is broadcast in the clear. Thus, an intruder can obtain this secret by passively eavesdropping on a given channel.

Configuring an AP to operate in stealth mode, i.e. instructing it to remove the SSID from beacon frames, will not prevent an adversary from obtaining the SSID. Unfortunately, it is also used in probe requests and probe responses, which are sent in

the clear. Furthermore, wireless network analyzers or sniffers, namely AiroPeek [122] and Kismet [152], can be used to detect the presence of WLANs.

802.11-W05-R01: Configuration of AP

One option, for minimizing the discovery of APs, is to install them in a central location and away from walls and windows. In addition to the physical placement of APs, one can also adjust the power and direction of the signal. The direction can be altered, by positioning one of the antennae and disabling the other. For example, with APs from Linksys, e.g. BEFW1154, the signal from either side of an AP, can be turned off.

802.11-W05-R02: Use of ACL

In order to accommodate the use of SSID, but nevertheless, provide a minimum level of authentication, some vendors have developed sophisticated APs that are equipped with more advanced security features. The use of a MAC address-based ACL, for providing authentication and access control, is one such feature. Unfortunately, MAC addresses can be spoofed by a determined hacker, as previously discussed, thus rendering this strategy ineffective.

802.11-W06: Lack of network authentication

Within the domain of wired networks, the authentication of network components (e.g. servers, switches, hubs, and routers) is typically carried out using various strategies, including the use of SNMP. In addition, physical security is also in place to secure these components.

However, vulnerabilities in physical security or the absence of a network auditing system can permit an individual to attach unauthorized devices, for the purpose of gaining access to the wired network. A well known scenario, associated with 802.11 networks, is the use of , see section 3.2.1 (802.11-A04).

802.11-W06-R01: RADIUS

Fortunately, administrators can make use of RADIUS to mitigate the security risk posed by RAPs, as suggested by Peikari and Fogie [132]. The RADIUS protocol, in particular, the Funk's Steel-Belted version (discussed in the sequel), can not only be used to authenticate WLAN users (mutual authentication) but APs also. In fact, all APs are required to *log in* before they become part of the network. The primary goal is to prevent an attacker from simply installing an illegal AP, into a remote hub or switch, and gaining access to the network.

While this approach specifically targets the authentication of APs, the use of RADIUS, for other types of authentication, is presented in the following section.

802.11-W07: Lack of mutual authentication

When shared key authentication has been implemented, users are authenticated by APs. However, the protocol is unidirectional, i.e. it does not accommodate the corroboration of an AP by users. This limitation renders wireless devices vulnerable to masquerading or impersonation attacks using RAPs.

However, it is encouraging to note that the link layer solutions, e.g. 802.11-S04, proposed by independent research teams, do support mutual authentication. These as well as higher layer solutions are presented in the next section.

2.2.4 Link Layer/Higher Layer Solutions

In the opinion of Taschek, the use of security measures, e.g. SSID, MAC address-based ACLs and WEP, may prove sufficient for networks, installed at home or small businesses, but it is seldom adequate for enterprises [171]. In this environment, there is a significant requirement to secure all direct access to corporate networks. What would prove useful are solutions that are robust (less vulnerable to attacks) and effective (fulfills predefined goals), especially when hacking tools, such as Air Snort [39] and WebCrack [105], used for exploiting weaknesses in WEP, are readily available.

As the following solutions are explored, one should keep in mind that the primary goal of the 802.11 standard is to provide an equivalent level of security to that of an

unsecured wired LAN [15]. Therefore, it was never intended to provide an end-to-end security solution. In addition, a distinction can be made between solutions, which address ***device authentication*** using a link layer protocol, and those that provide ***user authentication*** and other services using higher layer protocols.

Higher layer solution

Unlike the approach adopted by 802.11b (device authentication), the following solutions make use of authentication mechanisms, which are not only used in wired networks but are also implemented from the network to the application layer.

802.11-S01: Authentication without WEP

On August 20, 2001, the US-based National Aeronautics and Space Administration (NASA) proposed a solution for securing WLANs without the use of WEP [135]. The key objective of the authors was to develop a cost-effective solution with increased security, by using a *wireless firewall gateway.* The proposed architecture is comprised of a Dynamic Host Configuration Protocol (DHCP) server, an IP filtering mechanism and a Web authentication system:

1. The beta DHCPv3 open source server not only provides IP addresses to users but dynamically removes hosts, from the firewall access list, when the lease has expired;

2. Using OpenBSD's IPF software, IP filtering is enabled in the kernel state permitting packet filtering to be realized at the transport layer; and

3. Authentication is provided using scripts, which are installed on an Apache web server. This server can support multiple platforms including Unix, MAC OS, Windows and Linux. Devices, on the other hand, only require a Web browser and DHCP client software. In addition, the PHP and Perl scripts communicate with a RADIUS server in order to authenticate users. Once a user has been authenticated, his/her IP address is added to the IPF access rules. Finally, support for confidentiality is achieved by using SSL and digital certificates (public-and-private key RSA encryption).

One distinctive feature of this solution is the classification of users as authenticated and non-authenticated, followed by an appropriate level of access to various services. For example, non-authenticated users are granted access to services such as e-mail, VPN and the Web.

Advantages:

The cost-effective aspect of this solution is illustrated through the use of existing Unix stations as base stations. In addition, there appears to be a trend in the increased use of digital certificates, especially for supporting e-commerce applications (e.g. support for digital signatures). Thus, the use of public-and-private key RSA encryption is a reasonable choice.

Disadvantages:

While the ability to leverage existing Unix stations is advantageous, the acquisition and deployment of a RADIUS server may represent a sizeable investment. The cost factor alone may hinder the adoption of this solution, by most residential and small businesses, as indicated by Williams [187]. Also, the feasibility of employing RSA encryption (required for SSL) with existing wireless devices may be questionable.

802.11-S02: Authentication using DHCP

As with 802.11-S01, this solution, proposed by Arbaugh, Shankar and Wan, also makes use of DHCP, as a transport mechanism, in order to provide various security services including authentication and key management [16]. While DHCP is used for authentication at a higher layer [42], the rekey option, in DHCP, provides a mechanism for the transport of WEP keys. They are encrypted using one of three implementation options. The implementation strategy is based on the principle that a long term key be used for authentication (e.g. public-key cryptography) while a short-term key, e.g. WEP key (link layer key), which is changed frequently, is appropriate for encryption purposes.

As far as mutual authentication, between users and a DHCP server, is concerned,

it can be implemented using the public key, shared key or the session key options. The public key variant uses digital certificates, whereby authentication information, such as X.509 certificates, signature and appropriate nonces are exchanged, along with DHCPREQUEST and DHCPACK messages. This variant solves the problem of scalability when a static shared key is used.

On the other hand, the use of shared key (user and DHCP server) is similar in principle to the public key approach. However, the shared key, which is based on userID and generated by the server (not stored by server), is sent to the user in an out-of-band manner. In this case, the exchanged authentication information consists of the user identifier (userID) and MAC_2 of the shared key.

With the third option, a constraint is incorporated into the shared key strategy, i.e. the key is only valid for a session. Thus, the server is required to store the state for each user (i.e. a session key). This requirement can, nevertheless, be eliminated by providing valid users with a master secret S_m, and directing the server to include a nonce N with each lease. As a result, wireless devices derive the WEP key according to: $K = hash(S_m, N)$.

Advantages:

This solution neither requires modifications nor enhancements to the deployed wireless equipment. It is also transparent to end users. Moreover, it is non-proprietary, unlike those proposed by many vendors (e.g. use of 128-bit encryption). Finally, support for timed key management and disconnection of users is also provided.

Disadvantages:

Having to change the source code of a DHCP server and client, in order to implement the wireless rekeying option and to support large options (greater than 256 bytes), makes this solution more proprietary than generic. However, this approach is rather common, and has been adopted by other research initiatives also.

802.11-S03: Layered approach to authentication

A number of solutions, e.g. Borisov, Goldberg and Wagner [24], Mehta [117] and Laing [103], have been conceived, based on the layered approach to authentication and the desire to leverage existing technologies of the wired networks. What is interesting, however, is that some of the solutions only employ VPNs or RADIUS servers, and not necessarily both.

The key components of a fully layered architecture, which work in concert to provide a supplementary authentication mechanism to WEP, are the firewall, VPN and RADIUS server, see Fig. 2.10.

In a typical scenario, a user, using the VPN client software installed on a wireless device (or MN), is authenticated by a VPN Server. The encrypted tunnel, established by the VPN, protects the communication channel from nearby attackers. An enhanced form of authentication is also provided, through the use of certificates and mutual authentication.

Should security requirements warrant an additional level of authentication, the capabilities of a RADIUS server can be utilized. In this case, the corporate firewall forwards a user authentication request to a RADIUS server. Once the server has authenticated the user, using the central database, it returns access control-related information, such as the list of authorized services, to the firewall. Access to the resources of the wired network is henceforth managed and monitored by the firewall and RADIUS server.

This architecture can be further enhanced through the incorporation of an IDS. This component is typically installed between the AP and firewall, as suggested by Laing [103]. Hence, multiple layers of security can be implemented to render WLANs virtually impenetrable. The key issue is, and always will be, the need to achieve a balance between the benefits and costs.

Advantages:

VPN

The deployment of a VPN should not be overly challenging for most enterprises,

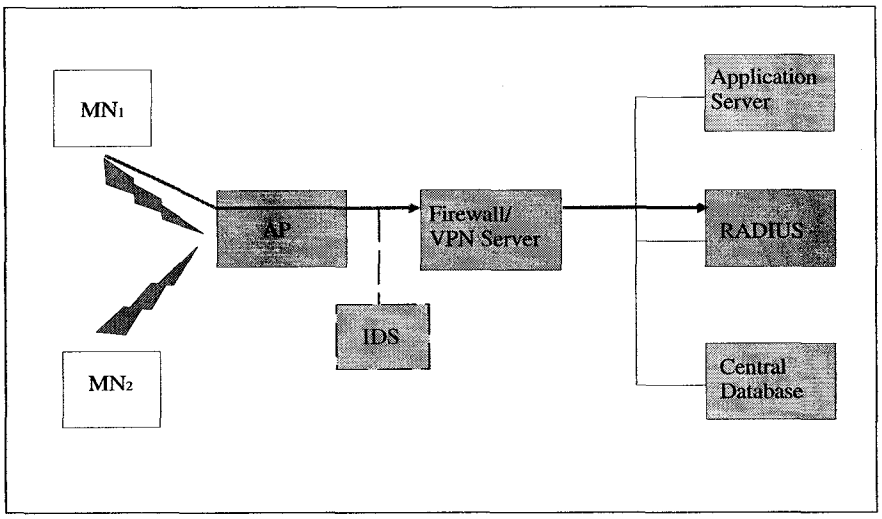

Figure 2.10: Authentication using a layered architecture

given that most corporate users would already have the VPN client software installed on their laptops. In addition, the use of digital certificates and the mutual authentication model renders this implementation of VPN superior to those based on the 802.1x standard [187]. However, these characteristics may be problematic with resource-constrained devices. More specifically, the memory requirements for the RSA public-key algorithm and VPN client may exceed the current capabilities of wireless devices.

RADIUS authentication server

In addition to the various authentication mechanisms supported by this server, it is capable of generating a WEP encryption key for each session, as indicated by Williams [187]. Consequently, the key would not have to be stored on client devices and APs, unlike WEP.

Disadvantages:

VPN

While anecdotal evidence suggests that larger enterprises favor a VPN-based Demilitarized Zone deployment model for WLANs, the installation and configuration of a new VPN environment can prove challenging. The key factors include complex configuration and interoperability problems, resulting from the use of different vendors and platforms. If, on the other hand, the deployment of a WLAN leverages an existing VPN environment, the transition can be relatively seamless, as stated by Williams [187].

According to Fisher, although VPNs are gradually becoming the norm, they may not scale well from management and cost perspectives [52]. Moreover, Al-hajeri, Merabti and Askwith indicate that in addition to being vulnerable to some attacks, the degradation in performance can be substantial, up to 50% of normal throughput [11].

RADIUS authentication server

As stated earlier, the acquisition and deployment of a RADIUS server represents an expensive alternative. Nevertheless, according to Williams, a cost-benefit analysis may justify its use in medium-large organizations [187].

Link layer solutions

A common characteristic, that is exhibited by higher layer solutions, is the use of other authentication mechanisms for overcoming the vulnerabilities of WEP. On the other hand, the following initiatives attempt to address these vulnerabilities, by exploiting the 802.11b functional specification or introducing enhancements that can be integrated into the WLAN environment.

802.11-S04: Use of 802.11 Point Coordination Function

The primary objective of authors, Park, Ganz and Ganz, is to develop a security protocol, which can be integrated into the 802.11 Point Coordination Function mode of operation. A secondary requirement is to take into consideration, the characteristics

of the WLAN environment, e.g. limited bandwidth, limited computational power and noisy wireless channel [131].

Key design principles include the use of:

- public-key cryptography to update the unique and current session key, which provides support for mutual authentication and confidentiality;
- 3-way handshake (initiated by an AP at the start of the contention-free period) to replace the initial polling frame, ordinarily sent by an AP; and
- various timers, which are used by devices and APs, and are initialized based on channel quality and application requirements.

The 3-way handshake provides mutual authentication through the exchange of a session key. During the first step, an AP authenticates a device by encrypting *message1* with a known session key K_{BSS} (shared within the Basic Service Set (BSS). If the device has been programmed with K_{BSS}, it should be able to decrypt *message1* (implicit form of authentication) and to respond to the AP. The device, in turn, creates a new session key K_{NEW}. It is encrypted using K_{BSS} and sent, via *message2*, to the AP. At this point, the AP generates the session key K_c, using K_{NEW}. It then makes use of K_c to encrypt *message3*, which is subsequently sent to the device. If the device can successfully decrypt *message3*, then the authentication of the AP is considered to be successful. Public-key cryptography is also used to encrypt authentication information, e.g. nonces and message digests in a message, before the message is encrypted with the session key.

This solution differs, from recently published techniques for WLANs by Chiasserini and Ganz [29] and Cellular Digital Packet Data networks by Sklar [160], in three key areas: the confidentiality provided in the first step, non-exposure of nonces and reduced number of computations.

Advantages:

Not only does this approach work independently of any infrastructure-based components, e.g. an authentication server, it also accommodates the characteristics of

the WLAN environment. In particular, it minimizes the number of messages (limited bandwidth), makes use of XOR operation for encryption (limited computational power), and provides support for message retransmissions (noisy channel).

Disadvantages:

While the use of public-key cryptography is advantageous, the increased requirements, for memory and computing power, may exceed the current level of resources in most wireless devices. However, this situation is expected to change in the near future. Also, modifications to the 802.11 protocol itself would more than likely require updates to the firmware in all clients and APs. Nevertheless, a cost-benefit analysis may support this strategic direction.

802.11-S05: Synchronized Random Numbers for Wireless Security (SPRiNG)

Unlike the previous solution, which has been specifically designed to work with 802.11, the SPRiNG protocol, proposed by Pepyne, Ho and Zheng [133], is not only compatible with WEP but it is also a generic solution, i.e. it can be used to secure PPP communication.

In brief, this protocol makes use of synchronized pseudo-random number generation, e.g. the linear congruential generator, to generate authentication data and new encryption keys, on a per-frame basis. It is similar to WEP in that it is also a symmetric key protocol, which provides support for authentication, confidentiality and data integrity. However, unlike WEP, it prevents replay attacks, eliminates key reuse and masks the secret keys behind the RC4 cipher.

Mutual authentication, between device A and access point B, is carried out using a shared PRNG and a seed R_0, which is generated at the start of each session, and is unique to each device. This strategy permits both A and B to locally generate a long non-repeating sequence $R_0, R_1, ..., R_n$ that is difficult to predict. Moreover, a counter K is used by both entities to keep track of the total number of data frames exchanged during the current session. Thus, when a data frame is sent by A to B, the

appropriate R_{ka} value is attached to the frame. In turn, B locates the R_{kb} value in its sequence, based on the value of its counter K_b. If the R_{kb} and R_{ka} values are identical, then the data frame is accepted. As with WEP, an implicit form of authentication is used, i.e. based on the knowledge of R_0.

Advantages:

Whereas some of the proposed alternatives to WEP authentication are computationally intensive, and thus would require the next generation of wireless hardware, the SPRiNG protocol can theoretically be implemented, using software, on almost any device (not fully tested yet). Of course, it stands to reason that a higher level of complexity would demand a higher price, i.e. more resources.

Disadvantages:

Having to maintain state information, for each device (e.g. seed) may not be feasible with current APs. Once again, this situation is likely to change with the introduction of more sophisticated APs, e.g. from 3Com Corp. [127]. These APs support 802.11a, 802.11b or 802.11g exclusively, but can be upgraded, using a second transceiver, to support multiple protocols simultaneously.

Solutions proposed by the standards body

In response to the numerous research papers (e.g. Intercepting Mobile Communications: The Insecurity of 802.11), which have highlighted the vulnerabilities of WEP, the 802.11 standards body has expended significant energies to address these vulnerabilities.

Although various initiatives, undertaken by different task forces, have resulted in the formulation of different security-related standards, e.g. 802.11i [187] also referred to as Robust Security Network, 802.11e [67] and 802.1x [79], the following discussion primarily focuses on WEP2 and the 802.1x authentication standard.

It is important to note, nevertheless, that 802.11i has been ratified since July 2004 [13]. This standard is intended to provide a more robust form of security, by

revamping the entire architecture of the secure-key environment. In order to address the vulnerabilities of WEP, prior to the completion of 802.11i, three interim solutions were made available: the Temporal Key Integrity Protocol (TKIP), Counter-Mode-CBC-MAC Protocol (CCMP) [27] and WiFi Protected Access (WPA).

TKIP is a set of algorithms, which not only addresses the known flaws of WEP, but also takes into consideration the constraints, imposed by hardware implementations of WEP. On the other hand, the CCMP provides similar functionality but without the need to accommodate existing implementations. Furthermore, the replacement of the weaker RC4 encryption standard, with the Advanced Encryption Standard (AES), represents a key factor in hardening the security elements of the protocol.

The WPA release provides a subset of the security services identified in 802.11i. It makes use of either 802.1x or pre-shared key technology for mutual authentication, and TKIP for data encryption. Finally, WPA2, an interoperable technology based on the full 802.11i standard [12], is intended to fulfill the most demanding security requirements of enterprises. The key difference, between WPA and WPA2, is the use of AES, which is based on CCMP. Consequently, it is eligible for compliance with Federal Information Processing Standards (FIPS)140-2 government security requirements. Currently, there are products from Cisco Systems, Atheros Communications Inc. and others, which have already been WPA2 certified.

802.11-S06: WEP2

In May 2001, Bernard Aboba [5], from Microsoft, presented an analysis of the security bandages implemented by the successor to WEP. Some of the key points are as follows:

- Although WEP2 does increase the IV key space to 128 bits, it fails to prevent IV replay exploits. Moreover, it still permits IV key reuse. Furthermore, a replay of the IV, in combination with a fabricated MAC address, also permits an attacker to forge authentication;
- Known plaintext exploits work as well with WEP2 as they did with WEP. These exploits are initiated by an intruder, who is aware of or can guess part or all of

the data payload or encrypted header contents. He/She subsequently makes use of this information, the IV key and CRC32 algorithm to crack the encryption itself;

- The mandatory support for KerberosV merely opens WEP2 to new dictionary-based attacks. It is estimated that up to 10% of *Kerberos-protected* user passwords can be cracked within 24 hours, using an inexpensive network of PCs, running parallel DES cracking techniques;
- Since re-associate and disassociate messages are not secured, WEP2 remains vulnerable to DoS attacks. These attacks severely deplete the resources of a server until it can no longer fulfill requests for services; and
- As beacon messages are not authenticated, client nodes, which roam to a RAP, can fall prey to its nefarious intentions.

Despite these vulnerabilities and limitations, there is one key advantage. The real-time decryption of WEP2 data streams, a prerequisite for eavesdropping, will be significantly more difficult, due to the increased size of the IV key space.

802.11-S07: 802.1x authentication standard

Recognizing the need for an enhanced and secure authentication/encryption mechanism, the IEEE has defined a new port-based standard to address most of the weaknesses of WEP. Some of the key features of the standard are increased key length, use of higher layer protocols, e.g. EAP, RADIUS and TLS, improved access control to networks and interoperability [79].

Given the increased level of security, many corporations, including Microsoft, have already begun to implement this standard in their operating systems (e.g. Windows XP) as early as December 2001. However, despite the enthusiasm demonstrated by Microsoft, other vendors have adopted a more cautious approach. Their caution has been justified by the discovery of two security problems, e.g. hijacking of user sessions and man-in-the-middle attacks, at the University of Maryland [119]. The IEEE

working group is in the process of addressing these flaws. Nevertheless, software development firms, such as Funk Software Inc., have incorporated this technology into their products, e.g. SBRS 4.0.

Advantages:

A standard-based solution certainly has its merits including interoperability, an acceptable level of security and the lack of significant vulnerabilities, assuming that it had been subjected to a comprehensive peer review process. As aforementioned, the use of higher layer protocols is not only beneficial, but it is also consistent with the strategic direction, adopted by vendors, e.g. Cisco Systems, and other research teams.

Disadvantages:

It is more than likely that the proposed changes will require upgrades to the firmware of all clients and APs, as indicated by Droms and Arbauth [16]. Furthermore, the processing power or other hardware limitations may prevent vendors from fully implementing these changes. As far as EAP is concerned, one must take into consideration, the vulnerabilities of this framework and the attacks that exploit them. A detailed treatment of both elements is available on the web site, maintained by Aboba [6].

2.2.5 Other Enhancements

The adaptation of user authentication solutions, from a wired to a wireless environment, is an iterative process, which brings forth alternatives that are more suited to the characteristics of the wireless domain. Some of the key initiatives are presented next.

Light Extensible Authentication Protocol

Although the use of EAP-TLS and RADIUS protocols does constitute a viable solution, it may not be optimal. TLS, as you may recall, is a certificate-based scheme. So, while it is feasible to use Public Key Infrastructure (PKI) with EAP-TLS, this

scenario has several drawbacks, according to Nichols and Lekkas [126]. Firstly, PKI schemes tend to be CPU-intensive, and therefore, not wireless device-friendly. Secondly, the administrative overhead may be prohibitive. Finally, the cost factor must also be taken into account.

In order to compensate for the disadvantages, associated with PKI schemes, Cisco Systems has developed the Light Extensible Authentication Protocol (LEAP). The most significant characteristics are: a) it can be implemented using variants of EAP; b) it makes use of the username-password authentication mechanism; and c) it permits the client to derive the encryption key, in a dynamic manner.

Key benefits of LEAP are as follows:

- Reduced CPU load on wireless devices;
- Support for operating systems, which do not natively support EAP; and
- Support for mutual authentication, unlike EAP-TLS, i.e. mutual authentication is NOT mandatory.

Wireless Public Key Infrastructure (WPKI)

The development and deployment of secure Web portals have received a lot of attention, over the past few years, in light of the increased demand for secure e-commerce transactions. While the PKI continues to support this initiative, within the wired domain, its scope is slowly being extended to accommodate wireless users.

A good example of a wireless PKI implementation is provided by Entrust's Secure Web Portal Solution [132]. In addition, Entrust's implementation also supports users' need for single sign-on to multiple applications. This requirement is fulfilled through the use of a secure token, which holds a user's identity and privileges. This token can then be forwarded to the appropriate application/service. Another advantage of using digital certificates is the built-in support for digital signatures. For example, when accessing e-commerce applications, a user is required to digitally sign the transaction contract, in order to prevent impersonation attacks.

WAP 2.0

While the previous versions of the WAP standard were developed specifically for the wireless domain, WAP 2.0 makes use of the standard IP protocol stack for providing Internet security services to 802.11 enabled devices [145].

2.3 Global System for Mobile Communication

Unlike WLANs, which have limited coverage (e.g. 100 meter radius) and are used primarily to extend data-oriented wired networks, WWANs are best represented by analog and digital cellular networks. These networks also extend the voice-oriented Public Switched Telephone Network (PSTN) or Integrated Services Digital Network (ISDN), as indicated by Pahlavan and Krishnamurthy [129]. Furthermore, they provide nationwide or worldwide roaming support to users, equipped with mobile cellular phones.

Although standardized in 1991, under the management of the European Telecommunications Standards Institute (ETSI), GSM remains one of the most deployed digital cellular networks today, especially in Europe. Operating within the 900, 1800 (DCS) or 1900 (Personal Communications Service in US) MHz band, its primary goal is to establish a mobile phone infrastructure, which has been specifically designed to accommodate voice services, comparable to PSTN and ISDN. According to Schiller [151], it also offers automatic location services, authentication, and encrypted wireless links.

As illustrated in Fig. 2.11, some of the key components of the architecture provide similar services to those in WLANs. In a typical scenario, a Mobile Station (MS) communicates with a Base Transceiver Station (BTS), which acts as an AP. Two or more BTSs, which form a Base Station Subsystem (BSS_2), are controlled by a Base Station Controller (BSC). A BSC manages the handover between BTSs, within a given BSS. As a component of the wired backbone, a Mobile Switching Center (MSC), which is a high-performance digital ISDN switch, supports handover of connections

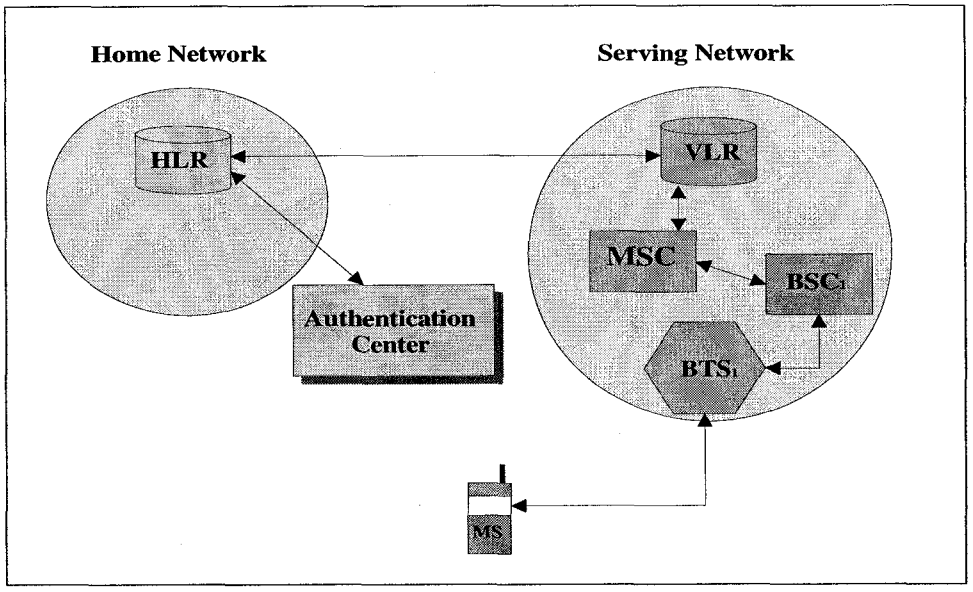

Figure 2.11: Architecture of GSM

between BSS_2. A detailed explanation of the protocol stack, by Mouly and Pautetare, is available at [123].

2.3.1 Security Services

GSM specification 02.09 [90] supports three security features, namely, user authentication, confidentiality of data, and user anonymity.

User Authentication Prior to rendering services to users/subscribers, the network validates the identity (stored in the SIM) of users, using a challenge-response scheme (discussed in the sequel) and the A3 algorithm.

Confidentiality of data Once users have been authenticated, encryption, using the A5 algorithm, is initiated by a BTS and MS, in order to protect signalling and user data. However, this feature is not supported within the remaining subsystems, e.g. the BSS_2 and network.

User anonymity Not only is user data protected, but measures are also in place

to hide the real identity of a MS. To this end, GSM minimizes the use of the subscriber's unique identifier, e.g. International Mobile Subscriber Identity (IMSI), over the air interface. Instead, a Temporary Mobile Subscriber Identity (TMSI), issued by the Visitor Location Register (VLR) of the current Location Area (LA), is used. Furthermore, this identifier is changed, as a subscriber moves from one LA to another. Finally, in most cases, the TMSI is not transmitted in plaintext. The use of a TMSI prevents attackers from determining the presence of a subscriber, in a given area (location confidentiality), and the types of services that are being rendered to him/her (user untraceability).

2.3.2 Authentication Protocol

As with most systems, authentication of subscribers, by the network, is based on a challenge-response mechanism. If an authentication attempt is unsuccessful, the network releases the radio connection. Additional details are available in the specification.

Prerequisite

There are three key entities, which participate in the authentication protocol, see Fig. 2.11: an Authentication Center (AuC) in the home network of the subscriber, a VLR in the serving network, and a subscriber, who is represented by his/her SIM, i.e. a smart card inserted into a GSM phone.

As the SIM is responsible for carrying out all security-related functions, on behalf of a subscriber, it is equipped with many identifiers and tables. This list includes the following: PIN, PIN unblocking key (PUK), IMSI, Authentication Key K_i [141], and all of the necessary algorithms.

The PIN (optional) is used for unlocking the SIM, a built-in security measure against theft. However, if an incorrect PIN is used three times in sequence, it will lock the SIM. Under these circumstances, a PUK, which is obtained from the network operator, is required to unlock it. The continued use of an invalid PUK (normally 10) results in disabling the SIM, thus rendering the phone inoperable.

ID	Weaknesses	Resolution strategies
GSM-W01	Lack of mutual authentication	Addressed in UMTS
GSM-W02	A3/A8 algorithm	COMP128-2,-3/MILENAGE, strong keys
GSM-W03	MS-VLR and VLR-HLR links	Public-key cryptography

Table 2.3: GSM: Authentication weaknesses and resolution strategies

Authentication of subscribers is based on IMSI and K_i. An IMSI is a number that uniquely identifies every subscriber in the world. It is a sequence of up to 15 decimal digits, which is used to represent, among other information, the home network of a subscriber and the country of issue. A K_i is a randomly generated 128-bit number that is assigned to a subscriber. This secret key is also made available to the network's AuC. Moreover, it acts as a seed, for the generation of all keys and challenges/responses, in the GSM system.

Other information is also stored in the SIM, while a subscriber is logged into the GSM network. It includes a 64-bit cipher key K_c, and location information, such as the TMSI and Location Area Identification (LAI).

2.3.3 Weaknesses and Resolution Strategies in Authentication

As indicated in the previous section, access control decisions are rendered, based on two elements of subscriber data: the identity of a subscriber (IMSI or TMSI) and a secret key (K_i). Thus, the acquisition of this information permits an intruder to impersonate a legitimate subscriber. Unfortunately, it is the victim, who is forced to bear the financial and other related costs, resulting from an impersonation attack.

In this section, specific flaws, which could permit an intruder to obtain an IMSI and the corresponding K_i, are identified. Table 2.3 lists the key weaknesses and the strategies, which have been proposed for addressing them.

GSM-W01: Lack of mutual authentication

The lack of mutual authentication represents the most serious flaw in the GSM authentication system. That is, there are no mechanisms in place, which permit a

subscriber/SIM to verify the credentials of the network, either implicitly (e.g. knowledge of K_i) or explicitly. This weakness has yet to be addressed.

GSM-W01-R01: UMTS

While GSM is considered a 2^{nd} generation system, the UMTS represents an evolution from GSM to a 3^{rd} generation system [151]. Please note that the term *2000* refers to both the starting year and the spectrum used (approximately 2000 MHz). The goal of IMT-2000 is to ultimately establish a world wide communication system, capable of supporting terminal and subscriber mobility, and thus permitting the realization of universal personal telecommunications.

As far as security services are concerned, they are implemented over the base security architecture of GSM. Defined in the first major release of the 3rd Generation Partnership Project (3GPP) specification (Release 99) [2], these services represent a superset of those provided in GSM and include mutual authentication, integrity and authentication of signalling data, partial integrity protection of user data and visibility of security.

GSM-W02: Flawed Implementation of the A3/A8 algorithm

Due to the fact that a standardized algorithm was not available, almost every GSM operator worldwide has adopted the COMP128 algorithm. It implements the A3/A8 specifications, defined by the GSM Consortium. This algorithm is seriously flawed, i.e. the K_i can be determined, using specific values for the parameter $RAND$. Furthermore, the details of the algorithm, which were once hidden from the public, are now available on the Internet.

Authors Briceno and Goldberg [26] present the technical details of the procedure, used to break the COMP128 algorithm. The attack, which requires physical access to the target SIM, is carried out by submitting specifically-chosen challenges (approximately 150,000) to the SIM, and analyzing the resulting responses.

In terms of the infrastructure, only an off-the-shelf smartcard reader and computer are required. Given that the smartcard reader, used by the authors, has a throughput

of 6.25 queries per second, the attack takes approximately 8 hours, with very little time being consumed for the analysis of the responses. However, depending on the speed of a SIM, which can be overclocked by many smartcard readers, and the utility being used, e.g. Sim Scan by Dejan Kaljevic, a K_i can be extracted within one hour.

It is no wonder that attackers were able to launch one of the most publicized attacks in GSM, i.e. SIM cloning (GSM-A02 in section 3.3.1).

GSM-W02-R01: COMP128-2,3

Since neither the cryptographic functions nor the storage of long-term secret keys are required by a serving network, a standardized authentication algorithm was deemed unnecessary. However, upon discovery of the aforementioned vulnerability, the GSM association quickly responded with a replacement algorithm, COMP128-2 or COMP128-3 (GSM-W02-R01).

Whereas COMP128-2 still weakens the ciphering key K_c, by using only 54 of the 64-bit key, COMP128-3 makes use of all 64 bits, thus preserving the intended level of security.

Although neither one of these two algorithms has exhibited any vulnerabilities to date, the fact that they have also been developed, using the *secrecy through obscurity* approach, provides little comfort. As everyone knows, it is only a matter of time before potential vulnerabilities are exposed.

In any event, the 3GPP standards body has pro-actively included the MILENAGE algorithm set [1] in the standards. With the exception of a random number generator, the set of algorithms includes all the necessary functions to support authentication and key generation requirements in UMTS.

GSM-W02-R02: Strong keys

Another strategy, for addressing this vulnerability, was introduced in February 2003 by the cryptographers of Prism Holdings [107]. They had discovered a set of keys, known as *strong keys*, that is more resilient to various attacks. They have subsequently developed tools for generating strong keys only. This technology has

not only been incorporated into the Prism SIM family, but it is also available, as a separate module, to other SIM manufacturers.

GSM-W03: MS-VLR and VLR-HLR links

According to GSM specifications, neither authentication nor data encryption is carried out between the VLR and Home Location Register (HLR). These two components exchange data, under the auspices of a mutual trust relationship. Of course, within the Pubic Land Mobile Network (PLMN) setting, this level of trust can be justified. However, when international roaming is taken into consideration, there is a need for enhanced security, i.e. based on strong cryptographic methods.

In any event, the VLR-HLR link remains vulnerable to third-party attacks. Moreover, the MS-VLR link is equally vulnerable. The lack of mutual authentication (GSM-W01) and the absence of data encryption, during the initial phase of the authentication process, renders this link insecure. As a matter of fact, data encryption cannot be initiated until the authentication process has been completed, and the CK_i and IK_i keys have been generated, by the MS.

GSM-W03-R01: Public-key cryptography

In order to secure both links and to eliminate the need for TMSIs, Grecas *et al.* [59] combine the security parameters of PLMN and PKI techniques. This approach requires that the PLMN operator, which is responsible for PKI deployment, execute the following procedure: create a public-private key pair for each MS, bind it to a digital certificate (a PKI parameter), and store the certificate in a SIM. In addition, the generation and dissemination of either a *single* PLMN-based public/private key pair or *multiple* key pairs, one for each location register, are also essential for securing inter-PLMN links. Naturally, the use of location register-based key pairs would provide a higher level of security, than is afforded by a single network-based key pair. However, the administrative cost, e.g. key management, would also have to be considered, especially, in the case of international or inter-PLMN roaming.

The incorporation of asymmetric cryptography into the GSM authentication protocol, is illustrated in Fig. 2.12.

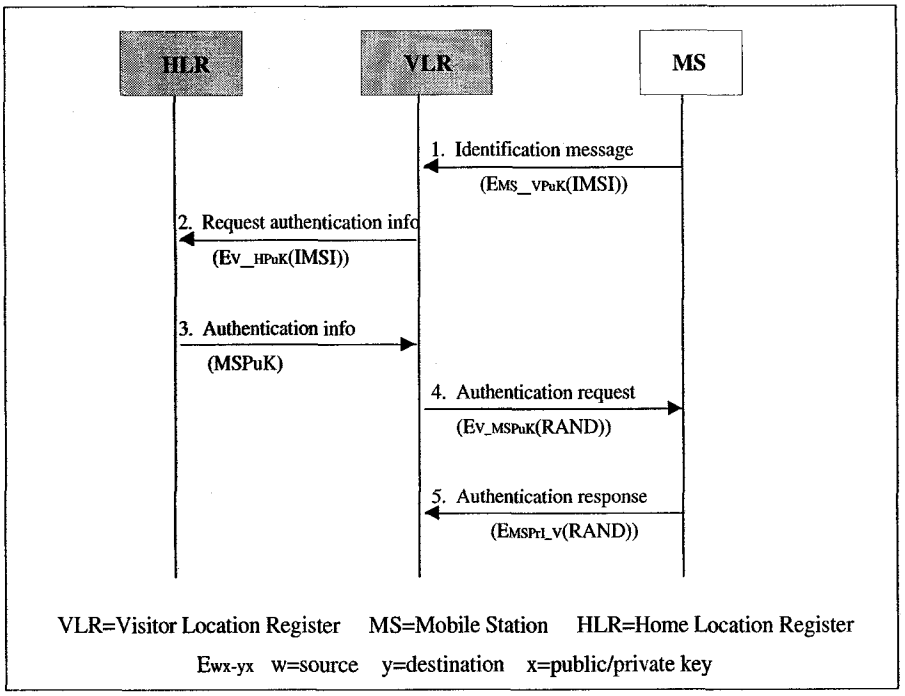

Figure 2.12: Incorporation of public-key cryptography into GSM authentication

1. At the start of the authentication process, a MS sends its IMSI, which has been encrypted using the public key of a VLR. This key would have to have been disseminated possibly via the Broadcast Channel.

2. The VLR requests authentication information, for the MS, from the HLR. This message is encrypted using the public key of the HLR.

3. An authentication response, which includes the public key of the MS, is returned to the VLR. Encryption is not used, nor is it required, for this message.

4. Once the VLR receives the public key of the MS, it uses the key to encrypt an authentication challenge (RAND), before forwarding it to the MS.

5. Lastly, the MS encrypts the authentication challenge, using its private key, and returns the encrypted response to the VLR, to be validated. In fact, the MS actually provides the VLR with a variant of its digital signature, which can be verified using the public key obtained in step 3.

Although the incorporation of asymmetric cryptography into the authentication protocol, has been demonstrated, using GSM, this mechanism can also be applied to GPRS and UMTS networks. Furthermore, if network operators assume the role and responsibilities, associated with the deployment of fully or partially-fledged PKIs, then PKI services can be delivered to subscribers. These services would permit them to participate in e/m-transactions that require the use of enhanced security services, such as digital signatures and non-repudiation.

2.4 CDMA 2000

Within the framework of Global Multimedia Mobility, ESTI has developed the basic requirements for UMTS and UMTS Terrestrial Radio Access (UTRA), the radio interface. Along with wideband CDMA (W-CDMA) and UTRA with time division duplexing, the third of the five radio interfaces, which was approved by the ITU in 1999, is CDMA2000. Also known as IMT-CDMA Multi Carrier (it's ITU name), and specified by 3GPP2 (www.3gpp2.org), this interface has evolved from its predecessor, the IS-95 system (cdmaOne), used in the US [66]. Thus, the need to satisfy specific requirements, for backward compatibility, has resulted in CDMA2000 inheriting many features and traits from its predecessor.

Not surprisingly, there are a number of variants, associated with this technology, e.g. CDMA2000 1X and CDMA2000 1xEV. The CDMA2000 family of radio interfaces collectively offers significant advantages, including increased voice capacity, higher data throughput, increased battery life and connectivity to other networks.

As far as deployments are concerned, the first 3G (CDMA2000 1X) commercial system was launched, by SK Telecom (Korea), in October 2000. It has subsequently been deployed in Asia, North and South America, and Europe. The CDMA-2000 1xEV-DO version was launched shortly afterwards, in 2002, by SK Telecom and KT freetel [66].

In terms of the security architecture, the role and responsibilities of the four participating entities, associated with GSM and UMTS, remain unchanged. The

only exception is the use of the term *UIM* instead of SIM/USIM. In this section, we focus on the 2G version of CDMA2000 security, and make appropriate references to the 3G version, as required.

2.4.1 Security Services

The security requirements, for CDMA2000, were initially specified in the Telecommunications Industry Association document *Enhanced Subscriber Authentication and Privacy*, but is now referenced as *Introduction to cdma2000 Spread Spectrum Systems* [18].

Although it may not be immediately obvious, the use of CDMA, as a MAC mechanism, does provide a rudimentary level of security. According to Wingert and Naidu [188], the use of *Long Code*, a 42-bit pseudo-random noise sequence, used for scrambling voice and data, makes eavesdropping very difficult, whether intentional or accidental.

The main security features, which have been implemented in CDMA2000, are discussed next.

Authentication Subscriber authentication, as described in the sequel, is supported by CDMA2000.

Confidentiality Confidentiality of voice, signaling and subscriber data is provided, by using the last 64-bits of the Secret Shared Data (SSD) and Cellular Authentication and Voice Encryption (CAVE) algorithm. The outputs of this algorithm are: Private Long Code Mask (PLCM); Cellular Message Encryption Algorithm (CMEA) key (64-bits); and data key (32-bits). As previously stated, the PLCM is used for voice scrambling, thus providing an additional layer of privacy, over the air interface. Likewise, both the CMEA and corresponding key are employed to encrypt signalling messages. Finally, the combined use of the data key and ORYX encryption algorithm, fulfills the need for confidentiality of subscriber data traffic.

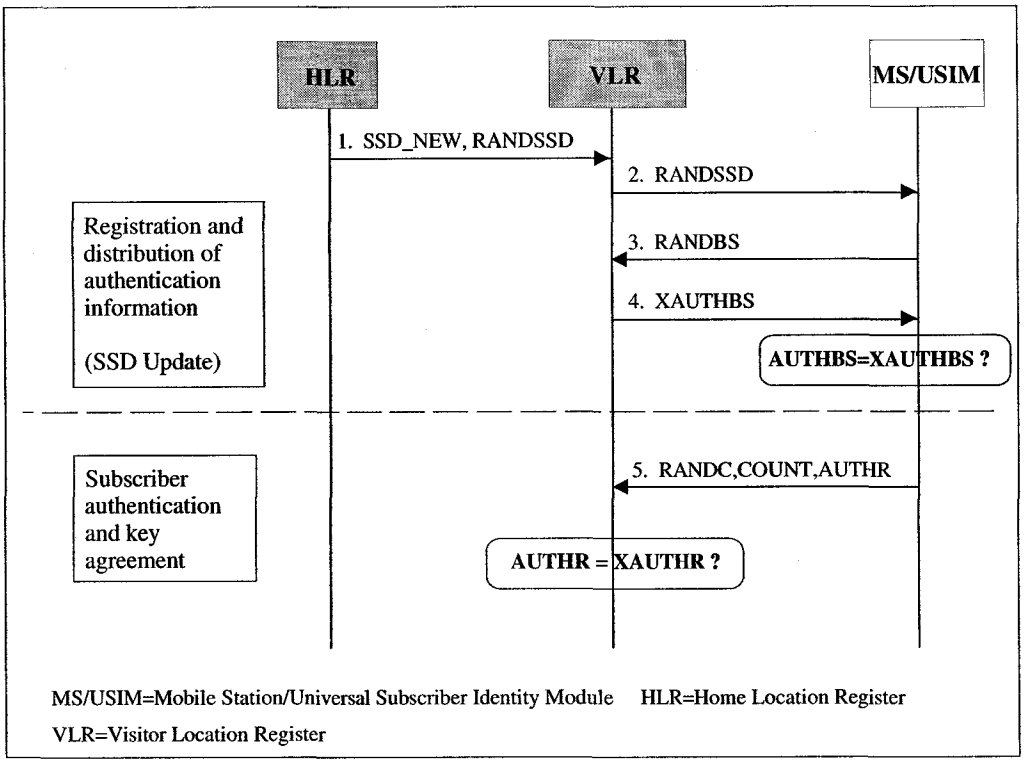

Figure 2.13: CDMA 2000: Authentication

User anonymity As with GSM and UMTS, CDMA2000 supports the allocation of a TMSI to a MS. As aforementioned, this feature makes it more difficult to establish an association, between transmissions of a MS and a mobile subscriber.

2.4.2 Subscriber Authentication and Key Agreement protocol

A simplified version of the registration and Subscriber Authentication and Key Agreement (SAKA) protocol (2G), is presented in Fig. 2.13. Additional details are available in [3] and [4]. These standards make use of the same authentication procedures and algorithms, which have been implemented in cdmaOne, a 2G system.

Prerequisite

Before proceeding to the description of the protocol, there are a number of details that are significant. First, the security protocols, i.e. registration and authentication, are dependent on a 64-bit authentication key $A-Key$, a randomly selected number

$RANDSSD$ and the factory coded Electronic Serial Number (ESN) of a MS.

Second, an $A-Key$ is programmed into a MS. It is also stored at the AuC, for the purpose of subscriber authentication. The $A-Keys$ are programmed, using one of the following methods: a) factory; b) dealer at point of sale; c) subscriber via telephone; and d) Over The Air Service Provisioning (OTASP), based on a 512-bit Diffie-Hellman key agreement algorithm. Using OTASP permits network providers to terminate services to a cloned MS, and to initiate new services to a legitimate subscriber, in a timely manner.

Finally, CDMA2000 employs the CAVE algorithm for generating a 128-bit sub-key SSD. Input to the CAVE algorithm consists of an $A-Key$, ESN and $RANDSSD$. Whereas the first 64-bits (most significant), of an SSD, are used for creating authentication signatures, the remaining bits are used to generate voice and data encryption keys.

Procedure

1. At the onset of the protocol, which is initiated by the SSD update procedure, a HLR selects a random number $RANDSSD$ and calculates a new SSD_{NEW}. It shares these values with a VLR (step 1).

2. The VLR forwards the $RANDSSD$ to a MS, so that it can derive the SSD_{NEW} (step 2).

3. In order to authenticate the VLR, the MS sends a base station challenge $RANDBS$ (step 3).

4. Based on the response $XAUTHBS$, from the VLR, the MS either updates the current SSD with SSD_{NEW}, if VLR was successfully authenticated, or simply discards SSD_{NEW} (step 4).

5. Next, the MS calculates a response $AUTHR$, using the CAVE algorithm and parameters $RAND$ and SSD_A. These parameters would have been disseminated via a global broadcast. It subsequently sends $AUTHR$, $RANDC$ (first 8

ID	Weaknesses	Resolution strategies
CDMA-W01	Lack of mutual authentication	HMAC, hash-chaining, AKA protocol

Table 2.4: CDMA2000: Authentication weaknesses and resolution strategies

bits of RAND) and *COUNT* (call history count used for clone prevention) to the VLR (step 5). This action initiates the SAKA protocol.

6. Finally, the VLR authenticates the MS. If the first authentication attempt is not successful, the VLR initiates a secondary challenge procedure, using a unique random number.

2.4.3 Weaknesses and Resolution Strategies in Authentication

The SAKA protocol suffers from one key weakness, i.e. the lack of mutual authentication, see Table 2.4.

CDMA-W01: Lack of mutual authentication

The SAKA protocol (2G) does not support the authentication of the network/VLR, by a MS. This weakness, as previously discussed, renders a MS vulnerable to impersonation attacks.

CDMA-W01-R01: HMAC and hash-chaining

In order to enhance the SAKA protocol, i.e. introduce mutual authentication and non-repudiation, Harn and Hsin [77] propose the use of two key techniques: HMAC and hash chaining.

Techniques

HMAC has been used extensively, by the Internet community, for message authentication. A HMAC is generated, using a cryptographic hash function and shared secret key. In principle, it is equivalent to a digital signature. The most common

form of HMAC is $hash(key, hash(key, message))$. HMAC-MD5 [111] and HMAC-SHA [112] are two of the most popular variants.

Lamport's one-time password/hash-chaining was originally proposed in 1981. Since then, it has been utilized in many applications such as the signing of digital streams, as proposed by Gennaro and Robatgi [56]. Its standard form is defined as:

$$f^M(x) = f \ldots (f(f(x)))$$

where $f(x)$ is a one-way function, and $f^M(x)$ represents M iterations of $f(x)$.

During the registration process, a claimant randomly selects an integer seed b. It then computes $f^M(b)$ and a HMAC of $f^M(b)$, using a shared secret key. Next, it sends both to a verifier. The key concept is that a claimant can prove its identity M times, by using each one of the hash chains. Thus, at the start of the first authentication attempt, a claimant submits $f^{M-1}(b)$ to a verifier, which determines if $f(f^{M-1}(b)) = f^M(b)$. If the claimant is successfully authenticated, the verifier replaces $f^M(b)$ with $f^{M-1}(b)$, in preparation for the next authentication attempt. When deemed necessary, the claimant submits $f^{M-2}(b)$, $f^{M-3}(b)$, ..., $f(b)$, where $(b = f^0(b))$, in sequence to authenticate itself repeatedly. The main advantage, of using a one-way hash chaining technique, is that it permits only the legitimate subscriber to generate a sequence of values that terminate with $f^M(b)$. As with most protocols, it is assumed that only the legitimate subscriber has knowledge of the secret key and b.

As far as non-repudiation is concerned, the combination of $f^{M-m}(b)$ and HMAC of $f^M(b)$, initially provided by the claimant, can be used by the verifier, as evidence for all m visits made by the claimant. In particular, the verifier can generate a proof of the claimant's j^{th} visit, where $1 \leq j \leq m-1$, by computing $f^{m-j}(f^{M-m}(b))$. The fact that the verifier stores only one value of the chain is very useful, especially for applications running on resource-constrained handsets.

An additional dimension can be incorporated into this scheme, in order to prolong the life time of a hash chain. More specifically, a claimant randomly selects I seeds, and computes $f^M(b_1), f^M(b_2), \ldots, f^M(b_I)$. It subsequently generates a HMAC of the concatenated message $f^M(b_1)||f^M(b_2)|| \ldots ||f^M(b_I)$, and sends both the message and

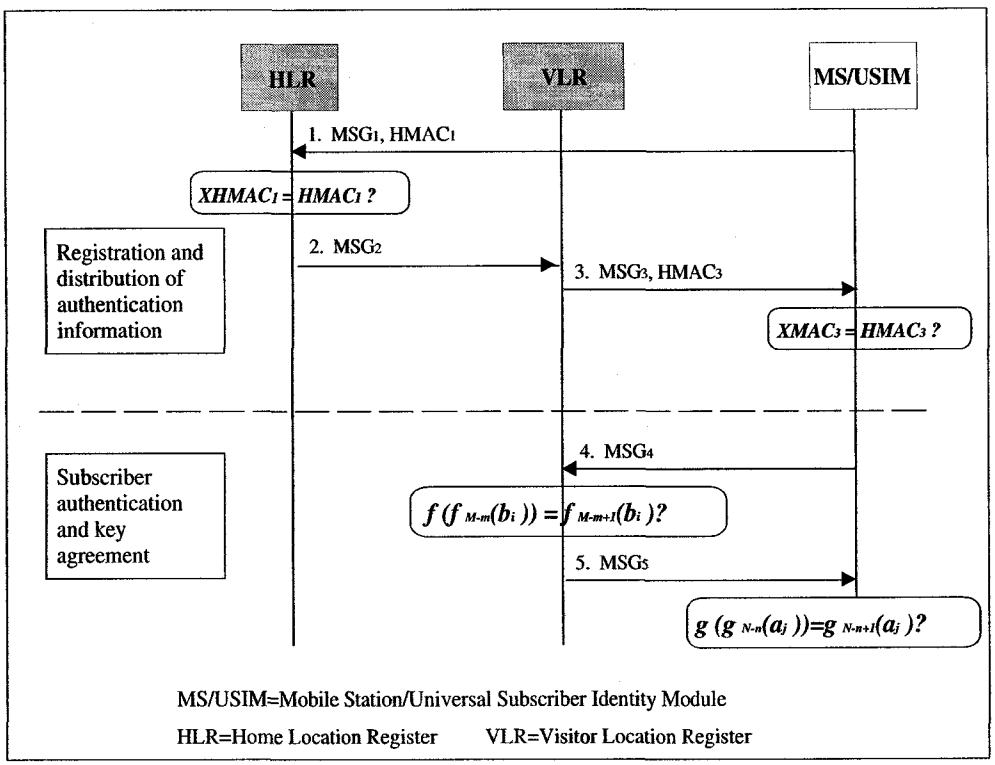

Figure 2.14: Enhanced SAKA Protocol

HMAC to a verifier.

Protocol Enhancement

The incorporation of HMAC and the hash-chaining technique into the SAKA protocol (3G) is depicted in Fig. 2.14. Given the high degree of similarity, between the SAKA protocol (3G) in CDMA2000 and the AKA protocol in UMTS, this proposed enhancement is equally applicable to both.

Furthermore, Table 2.5 lists the definition of all messages that are exchanged, during the registration and authentication processes. Finally, the assumptions, made by the authors, are that (1) a communication link between HLR and VLR is secure, and (2) a MS shares a secret key K with its HLR.

1. When a MS first roams into a new visitor domain, it sends MSG_1 and $HMAC_1$ to its HLR (step 1).

Acronyms	Definition
p, q, r, t	Various pre-defined functions
MSG_1	$IMSI \parallel f^M(b_1)\|\| \ldots \|\|f^M(b_I) \parallel timestamp$
$HMAC_1$	$t(K, MSG_1)$
$RAND_H$	Random number selected by HLR
CK_{II}	$p(K, RAND_H)$
IK_{II}	$q(K, RAND_H)$
AK	$r(K, RAND_H)$
MSG_2	$IMSI \parallel f^M(b_1)\|\| \ldots \|\|f^M(b_I) \parallel RAND_H \parallel AK \parallel CK_H \parallel IK_H$
MSG_3	$RAND_H \parallel g^N(a_1)\|\| \ldots \|\|g^N(a_J)$
$HMAC_3$	$t(AK, MSG_3)$
MSG_4	$f^{M-m}(b_i)$
MSG_5	$g^{N-n}(a_j)$

Table 2.5: Definition of messages

2. Once the HLR verifies the authenticity of MSG_1 and freshness of the timestamp, it selects a random number, prepares MSG_2 and sends it to the VLR (step 2).

3. Upon receiving MSG_2, the VLR prepares MSG_3 and $HMAC_3$, and sends both to the MS (step 3). This data is used, by the MS, to authenticate the VLR, during the registration procedure.

4. Following the registration procedure, the MS and VLR mutually authenticate one another, by exchanging an independently generated set of hash chains (steps 4 and 5). Since each authentication attempt makes use of one chain position, the MS can prove its identity, to the VLR, at most $I \times M$ times. Likewise, the VLR can also initiate $J \times N$ authentication requests. If a problem is encountered, while corroborating the identity of the claimant, the verifier sends an error message to the claimant. Upon receiving this message, the claimant attempts to re-authenticate itself, using the next fresh chain. For example, if the corrupted chain ID in the f series is 8, then the MS makes use of $f^{M-1}(b_9)$ for the next authentication request.

5. Although not explicitly illustrated in Fig. 2.14, the session keys for encryption and integrity, namely $CK_{i,m} = p(CK_{II}, g^{N-n}(a_j) \parallel f^{M-m}(b_i))$ and $IK_{i,m} = q(IK_{II}, g^{N-n}(a_j) \parallel f^{M-m}(b_i))$, are derived, by the MS and VLR, after a suc-

cessful authentication event.

The key advantages of this scheme are as follows:

Non-repudiation As aforementioned, the combination of $HMAC_1$ and $f^{M-m}(b_i)$ as well as $HMAC_3$ and $g^{N-n}(a_j)$, can be used for billing and dispute resolution.

Mutual authentication This requirement is fulfilled, through the exchange of an independently generated hash chain, between the MS and VLR.

Simplicity The SAKA protocol is simplified, to some extent, by eliminating the need to store and manage the call history counter $COUNT$.

Although the authors do not specifically comment on the disadvantages, they do exist and must be taken into consideration.

Higher overhead In comparison to the original SAKA protocol (2G), the overhead appears to be higher, i.e. transmission of additional data in some of the messages.

Overall system performance The storage requirements for an N element hash chain, as well as the generation of the hash chain itself, have an upper bound of $O(N)$. Nevertheless, the overall time complexity of the SAKA protocol is increased, as a result of employing the hash chaining algorithm. However, both Jakobsson [91] as well as Coppersmith and Jakobsson [35] have proposed the use of other algorithms, with space and time complexity of $log_2(N)$. These algorithms have been specifically designed to accommodate applications, such as micro-payments, authentication and digital signatures.

Finally, until such time as the SAKA (3G) protocol is widely deployed, interoperability between the 2G and 3G versions is likely to prove challenging. It remains to be seen as to how the two authentication styles, i.e. total control exercised by the 3G version vs. local control by 2G, will be integrated.

CDMA-W01-R02: Use of 3GPP AKA protocol

It is recommended that the 2G version of SAKA protocol and underlying algorithms be replaced with that of the 3GPP Authentication and Key Agreement (AKA) protocol (used by UMTS) and SHA-1/HMAC. The deployment of the AKA protocol should also address the lack of network authentication. This strategy, adopted by 3GPP2, is intended to not only fulfill the requirements for interoperability, but also to facilitate global roaming, in the future.

Chapter 3

Intrusion detection in wireless networks

The very nature of wireless communication, which has inspired organizations to adopt and deploy wireless networks, also has a vulnerable side, one that can be exploited at different layers of the TCP/IP protocol stack.

At the physical layer, the transmission of data, using undirected links, renders the networks susceptible to malicious attacks. These attacks range from passive eavesdropping to active forms including jamming and interference. Furthermore, unlike the wired environment, where an adversary is required to gain physical access to the network wires, attacks against wireless networks can be carried out from a distance of several kilometers.

Whereas attacks, at the MAC layer, include monopoly of the air interface, malicious routing represents the most significant attack, at the network layer. From this point on, the differentiation between the types of attacks, associated with wired and wireless networks, starts to diminish.

In addition to these types of attacks, there are others, which exploit specific weaknesses, e.g. definition, implementation, and/or appropriate use of security protocols, configuration of network components, and administration, in authentication systems. The remaining set of attacks, which we refer to as *peripheral - outside the scope of authentication*, have materialized, as a result of the increased availability of advanced

techniques and technological tools, e.g. wireless scanners and smart card readers.

Irrespective of the type of attacks, successful intrusions can result in alterations to messages, device/user impersonation and, more importantly, unauthorized access to the resources and services of wired/wireless networks.

Notwithstanding the benefits of intrusion prevention, current authentication mechanisms are ill suited to *prevent* peripheral attacks. Hence, a complementary security mechanism is required in order to *detect* them and to respond to them, before significant damages are sustained. It is this requirement, which has served as an impetus for the introduction of IDSs.

IDSs are categorized based on the source of data and approach used, for detection purposes. Those that detect malicious activity, e.g. draining the battery of a single host, are referred to as *host-based* IDSs. They also make use of operating system (OS) audit data, for monitoring and analyzing events generated by programs and users.

On the other hand, *network-based* IDSs rely on network traffic and/or data flows, for detecting intrusions. These systems are typically implemented, using a centralized management system and sensors. The sensors, which vary in cost and functionality, are deployed at key points of entry, eg. base stations and APs. Although some research teams do make a distinction between *intrusion*, i.e. attacks from the outside, and *misuse*, i.e. attacks from within the network, it is not a significant factor amongst the research community at large, as noted by Yang *et al.* [193].

Today's IDSs are implemented, using a *rule-based* or an *anomaly-based* detection approach, as indicated by Bass [23]. Rule-based detection attempts to recognize known forms of attacks, which have been characterized as templates (signatures) and subsequently stored in a list. The primary disadvantage of this approach is that it fails to recognize new forms of attacks.

In contrast, ABID is carried out, using two primary components: profiles and a classification system. In ABID, the *normal* behavior of an entity, e.g. device [101], user [167], operating system [106] or system process [84], is defined in a profile. The primary functionality, of the classification system, is to determine if the current behavior of an entity, in comparison to its profile, is normal or anomalous/intrusion.

Since an Intrusion Detection (ID) component also serves as an alert mechanism, its secondary function is to raise an alarm if an intrusion is suspected.

Unless a profile accurately represents the behavioral characteristics of an entity, and is updated periodically, to reflect changes in behavior, a high FAR (legitimate entity classified as an intruder) can result. Nevertheless, the FAR can be minimized, by combining observations across time and from different sources, e.g. MAC and network layers. When ID is carried out, classification results, of multiple observations or events, are correlated in time, using a state-probabilistic model, such as Bayesian filters [148]. This strategy accommodates a moderate degree of variability in normal behavior, as indicated by Morin and Debar [121], and thus minimizes the FAR. Moreover, it can be further reduced, through the application of a statistical technique, namely Multivariate Analysis (MVA) [95]. With MVA, an observation, which is defined using statistical data from multiple sources, is classified using appropriate thresholds. As a result, classification uncertainty is reduced. Examples of IDSs, which make use of multi-sensor data, for enhanced detection, include AAFID by Balasubramaniyan [19] and EMERALD by Porras and Neumann [137].

Another variant of ABID is specification-based detection, as presented by Ko, Ruschitzka and Levitt [100]. It has been applied to privileged programs, applications and several network protocols. With this approach, the *correct* behavior of critical entities is manually abstracted and formulated as security specifications, which are compared to their *actual* behavior. Intrusions, that cause an entity to behave in an incorrect manner, are detected, without exact knowledge of their specificities.

As aforementioned, attacks can target different layers of a protocol stack. Hence, anomaly detection, in a communication system, can be carried out, from the application to the physical layer. For example, research, in the detection of network routing misbehavior, has been conducted by Just, Kranakis and Wan [96], as well as Zhang and Lee [196]. Furthermore, malicious activities, related to MAC (link layer), have also been investigated by Kyasanur and Vaidya [102].

In this chapter, various attacks that are mounted against BT, WiFi, GSM and CDMA2000, for the purpose of gaining unauthorized access to these networks, are

examined. In order to guide the discussion of these attacks, a summary of the most significant attacks, corresponding weaknesses in the authentication protocol, and proposed countermeasures, is presented in a tabular format. As discussed in the previous chapter, when an attack specifically exploits a known weakness, the most logical solution would be to eliminate the weakness, by implementing one or more of the corresponding resolution strategies. Should this course of action be infeasible, due to economical and/or other factors, a different strategy would then be required to detect these attacks.

3.1 ABID in Bluetooth Network

While the primary objective is to identify attacks, against a BT network, there are, nevertheless, some attacks that target BT devices. It would prove interesting to briefly explore these before continuing with the stated objective.

The family of attacks, presented by McFedries [116], can be easily distinguished from the rest, by the simple fact that all of the names begin with the term *blue*. For example, *bluespamming* refers to the activity of *bluejacking* BT devices, and forwarding unsolicited commercial messages to them. On the other hand, *bluebrowsing* is carried out, with the intention of acquiring an inventory, of the services, available on a BT device. Finally, *bluesnarfing* is mounted by perpetrators, who surreptitiously access information, such as e-mail messages and calendar entries, from a victim's BT device. One specific attack model exploits the vulnerabilities in the Object Exchange (OBEX) protocol, used for synchronizing files between two BT devices. According to the proponents of the BT standard and McFedries, the solution is as simple as configuring a BT device, such that it is non-discoverable.

3.1.1 Attacks and Countermeasures

As previously mentioned, due to the limited coverage of ad-hoc networks, and transitive aspect of their formation, BT networks are less susceptible to attacks, although not immune to them. Table 3.1 presents the key attacks (not an exhaustive list) and

Weakness	ID	Attacks	Countermeasures
BT-W01	BT-A01	Obtaining PINs	Not identified
BT-W02	BT-A02	Man-in-the-middle	Not identified
	BT-A03	Network penetration	ACL, Limited Discoverable Mode
	BT-A04	BT_ADDR Spoofing	Not identified

Table 3.1: BT: Attacks and Countermeasures

countermeasures.

BT-A01: Obtaining PINs

The tendency of users to use weak PIN can render the key generation protocol vulnerable to various attacks. If a PIN, which is the only secret element in the derivation of the initialization key K_{init}, is discovered, an attacker can generate the K_{init}. Equipped with K_{init}, he can subsequently obtain both the link key, used for authentication, and encryption key.

BT-A02: Man-in-the-middle attack

Possession of device A's unit key permits an attacker to impersonate A. This form of attack can be carried out against any and all other devices, which currently use the unit key of device A, as their link key. Moreover, since the encryption key is derived from the link key (unit key), an attacker can also eavesdrop on all communication between A and any other device that uses the same link key.

Furthermore, a man-in-the-middle attack can also be mounted, using a similar attack model, as suggested by Jakobsson and Wetzel [92]. We assume that an attacker, using device B, has already obtained the link key, either by eavesdropping or by having communicated with device A in the past. Now, device B can obtain the address of device C, which is sent in the clear, and through successful authentication, impersonate device C to device A and vise versa.

BT-A03: Network Penetration

Currently, there are no authentication mechanisms, which will prevent a malicious user from promiscuously interfacing with another BT device, or joining a BT network,

as stated by Barber [22]. Although this problem has yet to be resolved, possibly using application-level monitors, the following options may prove useful, in the meantime.

BT-A03-C01: Use of ACL

One possible strategy is for a master device to maintain a list of both authorized devices and their distinguishing features, e.g. PINs. This ID model would be appropriate in an environment, where a list of devices, to be used by meeting and conference attendees, is known in advance. In addition, a master device, with sufficient resources, would also be required.

BT-A03-C02: Limited Discoverable Mode

Another option is to make use of communication establishment policies, which have been defined in the generic access profile [118]. These policies can either be established, by manufacturers, or configured by users. Although the policies define different modes of operation, namely discoverability, connectability and pairing, it may prove sufficient to set the mode of participating devices to *limited discoverable.* In this mode, a device only responds to inquiries that contain the limited Inquiry Access Code (IAC). In fact, the correlators of its receiver are specifically tuned to a particular IAC [63]. Therefore, this technique should prevent all devices, in a pre-established network, from responding to inquiry messages from an intruder (device).

BT-A04: BT_ADDR Spoofing

As aforementioned, each BT device is identified using a *unique* address. This feature permits BT devices to establish relationships with one another, based on their addresses. Unfortunately, an intruding device can alter its address to match that of a legitimate device, as demonstrated by Hager and Midkiff [70]. Moreover, as you may have already guessed by now, it can certainly impersonate the legitimate device. In addition to address spoofing, the authors conclude that two devices, with identical addresses, are able to not only form a piconet, but to make the master/slave switch as well.

Weakness	ID	Attacks	Countermeasures
802.11-W05	802.11-A01	War driving	Software signature
802.11-W05	802.11-A02	Parking lot attack	OUI filtering
802.11-W04	802.11-A03	MAC address spoofing	Intruder location, AirDefense User mobility profiles
802.11-W06	802.11-A04	RAP-Unauthorized access	MAC filtering, OUI filtering, Embedded IDS, AirMagnet
802.11-W07	802.11-A05	RAP-AP spoofing	Location-based approach
		RAP=Rogue access point	

Table 3.2: WiFi/802.11: Attacks and Countermeasures

3.2 ABID in WiFi/802.11 Networks

The popularity of 802.11 wireless networks has been further reinforced, by device manufacturers. Most laptops today are equipped with a built-in 802.11 networking device. While this level of popularity is indicative of consumer acceptance, i.e. a large installed base, these networks have also become a target of numerous attacks. Ellison's research initiatives [46] confirm that a majority of 802.11b wireless LANs are vulnerable.

Thus, unlike wired IDSs, their wireless counterparts must also defend against attacks, which exploit the characteristics of wireless communication. According to Potter [139], this set includes war driving, RAPs, MAC address spoofing, and session hijacking. While this list of attacks is not exhaustive, it does demonstrate the diversity of potential threats, which should be addressed, using suitable defense mechanisms.

3.2.1 Attacks and Countermeasures

Table 3.2 presents the list of attacks and countermeasures, associated with 802.11 infrastructure-based networks.

802.11-A01: War driving

In order to obtain access to the electronic resources of an organization, an intruder must become a member of its wireless network. But first, he must locate one. War

driving refers to the activity of driving in a vehicle, for the purpose of detecting wireless networks. This activity is more technically feasible with infrastructure-based networks, since ad-hoc networks are usually temporary in nature, and thus, are usually more difficult to locate. However, it is possible to locate them, by conducting a surveillance of public areas, such as coffee shops and conventions. Equipped with a Global Positioning System (GPS) device, the coordinates of APs can also be identified.

As far as the detection of APs is concerned, it is carried out by capturing beacon frames, which are broadcast by APs approximately every 10 milliseconds. A software package, such as NetStumbler [132], is used for this purpose. Although the software extracts information, e.g. SSID, type of device (AP or peer) and MAC address of AP, from the header of beacon frames, it does not capture actual data or management frames. Once the SSID of an AP has been obtained, it is used for configuring a Wireless Network Interface Card (WNIC). Henceforth, an association can be established, with the corresponding AP, providing that only a SSID is being used for access control.

Other software packages, such as ORiNOCO client software, simplify this process even further. They permit users/attackers to specify the value *any*, as the network name (equivalent to SSID), in the configuration profile. This value instructs a WNIC to automatically associate with any of the APs, discovered by the software.

802.11-A01-C01: Signature of software

An interesting approach, for detecting war driving, makes use of software signatures. As aforementioned, NetStumbler is one of the most popular tools used for war driving. Moreover, as noted by Hsieh *et al.* [85], it immediately transmits a special packet, after the detection of an AP. This packet contains a unique value (signature) that can be used to identify this software. Hence, upon recognition of this signature, the WIDS alerts network administrators to the likelihood of an impending attack.

When an intrusion attempt has been confirmed, the intrusion reaction module automatically begins to generate false probe response frames, containing faked AP information. This strategy serves to confuse an attacker. In addition, a message is

sent, by the GSM alarm module, to inform network administrators of the change in security status.

802.11-A02: Parking lot attack

While the need for war driving, is justified in suburban districts, it is often unnecessary in areas, such as business parks and commercial zones. In this environment, an attacker can remain stationary and eavesdrop directly from a parking lot.

802.11-A02-C01: Organizationally Unique Identifier filtering

One relatively simple strategy, for detecting parking lot attacks, requires the verification of Organizationally Unique Identifiers (OUIs). An OUI, which is represented by the first three bytes of a MAC address, specifies the manufacturer of a wireless network card. These identifiers are assigned, by the IEEE, to hardware vendors. For example, as indicated by Hsieh *et al.* [85], 00-02-9C and 00-08-0D identify wireless cards, which have been manufactured by 3COM and Toshiba respectively.

As suggested by Sharma [156], an attacker typically generates a random MAC address, for the purpose of sending probe messages. Therefore, it is highly unlikely that the OUI, of this MAC address, would be associated with one of the wireless card manufacturers, on the publicly available list.

Although this strategy does have merits, its application is limited to a specific attack model. Moreover, given that MAC addresses of legitimate users can be obtained with relative ease, an attacker could simply use one of these addresses instead. Under these circumstances, the verification of the OUI would fail to detect this type of intrusion.

802.11-A03: MAC address spoofing

Obtaining a list of MAC addresses is relatively straightforward when a wireless packet sniffer, such as CommView [169], is used. A screen shot of this tool is presented in Fig. 3.1. Given that MAC addresses are typically used, e.g. in a WEP frame, to identify a wireless device, and are sent in the clear [97], there is very little challenge in

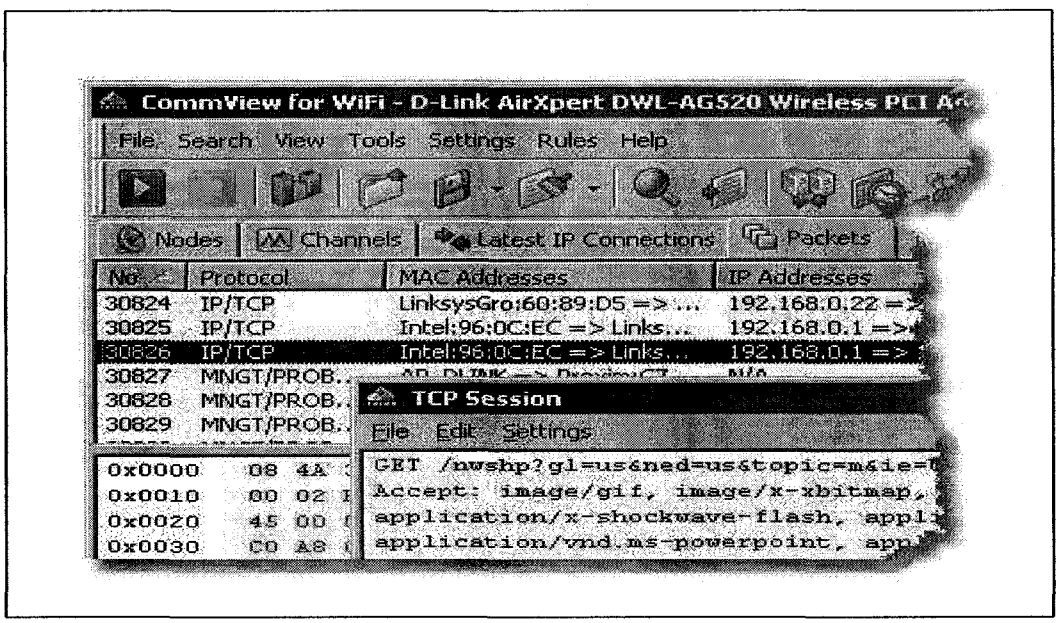

Figure 3.1: Tool: CommView for WiFi

obtaining these addresses. Once a WNIC has been configured with the MAC address of a legitimate user, an intruder becomes a member of the wireless network.

802.11-A03-C01: Location of Intruders

An interesting approach, adopted by Adelstein *et al.* [9], exploits the location of users for detecting MAC address spoofing. It is very similar, in principle, to the *where you are* form of authentication, advocated by Newbury networks [58].

The key objective of the authors is to implement an early intrusion detection and response system in 802.11 networks. In order to fulfill this objective, several APs, with directional antennas, are strategically positioned outside the estimated perimeter of a wireless network, see Fig. 3.2. The authors make the assumption that authorized users typically connect to the omnidirectional APs, which are located within the perimeter. In addition, they suspect that an intruder is likely to associate with a WIDS-AP, before coming within range of a network-AP.

In terms of functionality, a WIDS-AP performs three key tasks. First, it detects a signal from a remote wireless device. Second, an anomaly-based detection technique, which makes use of a device profile (associated with a given MAC address), is

Components	Details
Features in device profile	Packet and behavioral data
Classification	Statistical (inferred)

Table 3.3: Components of location-based WIDS

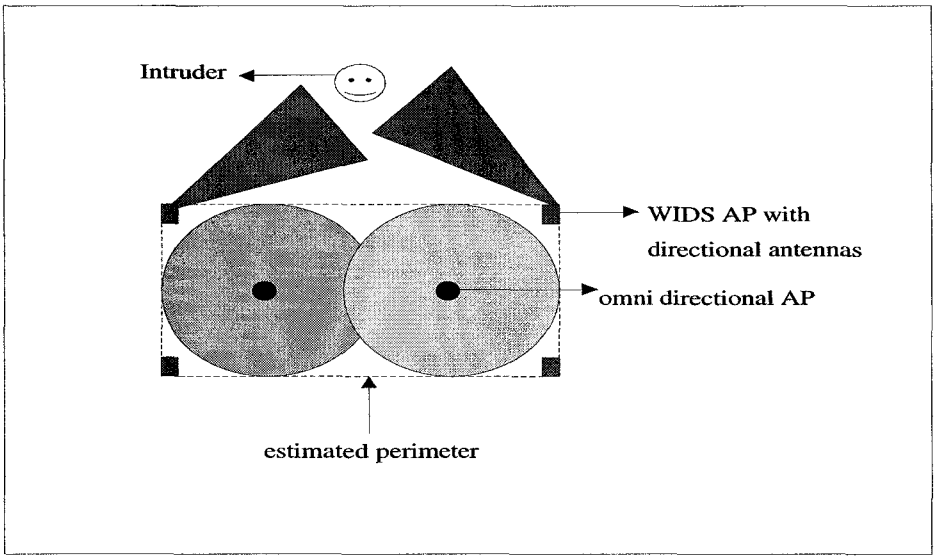

Figure 3.2: WIDS using location of users

employed to detect MAC address spoofing. Finally, the location of an intruder is determined, using two or more WIDS-APs. Table 3.3 highlights the key characteristics of this WIDS.

As far as the profiles are concerned, they are created using packet/frame data, as well as behavioral data. Specific features, used for profiling purposes, include signal strength levels, locations, Time-To-Live (TTL) values, IP identifiers, and OS characteristics. Thus, for example, an internal (within the perimeter) MAC address, which suddenly appears outside the perimeter, and is associated with a system that exhibits different OS characteristics (in comparison to the profile), would raise an alarm. In addition, device-specific triggers, e.g. packets from MAC address 00:20:E0:8C:92:88 must have a TTL of 30 and a signal strength greater than 10, can be defined. This concept is similar to a honeypot or honeynet [86], but permits a more precise form of control.

Once an intrusion has been detected, the final goal is to locate the intruder. This exercise is carried out by using beam patterns from directional antennas of two or more WIDS-APs, a mechanism that is similar to triangulation or trilateration. For each individual antenna, a training data set is created by rotating the directional antenna, and capturing the signal strength values of an arbitrary transmitter. In particular, the rotation is executed in fixed increments (e.g. 1°) over the entire 360°. In order to identify the location (angle) of the intruder, a set of signal strength measurements, i.e. obtained at different angles of the WIDS-APs, is obtained from his/her transmissions. The next step is to determine the degree of correlation, between the training data, associated with each antenna, and observed measurements. The most probable angle is defined as having the highest correlation. The resulting set of angles (one per antenna) is used to pin point the location of the intruder.

802.11-A03-C02: User mobility profiles

Instead of using single Location Coordinate (LC)s, for detection purposes, Jared Spencer leverages the use of mobility patterns [163], initially proposed by Tao *et al.* [170]. However, there are two key distinguishing characteristics. First, they use different classifiers. Whereas Tao and his colleagues make use of the Bayesian belief network, Spencer experiments with the use of Artificial Neural Network (ANN). Second, and most importantly, Spencer also proposes the use of mobility patterns, which are associated with a given user and MAC address, for detecting MAC address spoofing. This concept is based on the premise that the mobility behavior of users is sufficiently different, and thus can be used to uniquely characterize them. Hence, although a MAC address can still be spoofed, the mobility behavior of an intruder will most likely be different from that of a legitimate user. Table 3.4 highlights the unique characteristics of the proposed IDS.

As far as the ANN is concerned, it has previously been shown, by Cannady and Harrell [28], to be effective in the area of intrusion detection. It is used as follows. First, a training set is developed. It consists of normal (to a user being profiled) mobility patterns, i.e. sequences of LCs that are identified by location sensing mech-

Components	Details
Features in device profile	UMPs
Detection/Classification	ANN

Table 3.4: Components of mobility-based WIDS

anisms. Second, a test set is created, using normal patterns, in order to determine the accuracy of the ANN. Based on the results, the parameters of the ANN are adjusted accordingly. Finally, the detection phase is initiated. During this phase, observed mobility patterns of users, which do not match those (training set), associated with a given MAC address, are flagged as being abnormal, i.e. symptoms of a potential intrusion.

Although the use of mobility profiles can prove beneficial, for detecting MAC address spoofing, the true performance of the proposed system will only be revealed once the system has been implemented.

802.11-A04: RAPs - Unauthorized access

Unauthorized association with wireless networks, via administered APs, continues to be a source of concern for most organizations. As indicated by Peikari and Fogie [132], this situation is further exasperated by the installation of unauthorized APs, by internal employees. Referred to as internal RAP, they not only provide authorized users with much coveted mobility, but also conveniently open the door for denial-of-service attacks [32].

Likewise, an external RAP, one that is within the control of an attacker, can also be used for gaining access to a network. Technically, it can be as simple as plugging it into a remote hub or switch, which belongs to an organization.

802.11-A04-C01: MAC filtering

In order to detect RAPs, the use of MAC address filtering has been adopted by Chirumamilla and Ramamurthy [30]. In an effort to develop an intrusion detection and response system, which fulfills the three A's (authentication, authorization and

accounting) of security, the authors have adopted a centralized and agent-based architecture. The two key entities, which participate in the detection of RAPs, are the agents and central administrator. Deployed in all cells, i.e. coverage area of APs, and connected to the wired back bone, the agents continuously scan for RAPs. In other words, they sniff for beacon frames that are transmitted by all APs, within their respective coverage areas. If the source MAC address of a frame does not appear on the list of registered APs, the presence of a RAP is suspected, and an email notification is sent to the appropriate personnel. The list itself is created and maintained by a central administrator, which distributes it to all participating agents.

While the overall approach is technically feasible, it incurs an administrative overhead. For example, all APs and clients must register in order to obtain membership to a wireless network. Nevertheless, the primary weakness is the use of MAC addresses for the detection of RAPs.

802.11-A04-C02: OUI-based ACL

As aforementioned, the use of OUIs, for the detection of parking lot attacks, was proposed by Sharma [156]. In order to identify RAPs, the author suggests the use of an ACL, which is based on the OUIs of wireless cards deployed by an organization. However, this variant of MAC filtering is also vulnerable to MAC address spoofing.

802.11-A04-C03: Embedded IDS

The utilization of UMPs, for addressing MAC address spoofing, was presented earlier. In a similar vein, Raja and Suganthi [144] explore the concept of a low-level IDS, e.g. an ID chip or embedded IDS, which monitors the behavior of APs and wireless nodes/devices of users. It determines their *normal* characteristics, in an adaptive manner. Furthermore, based on these characteristics, it identifies deviations, which are indicative of potential intrusions. Although the authors do advocate the need for low-level intrusion detection capabilities, using hardware or firmware, their rationale is based on performance gains that can be achieved, via an Application-Specific Integrated Circuit design. While HW-based solutions are typically faster than those

Components	Details
Features in user profile	User session data
Features in device profile	Routing and position data
Classification	Genetic Algorithm

Table 3.5: Components of Embedded IDS

based on SW, in this case, it appears to be a requirement for accommodating the use of large volumes of audit data and the genetic algorithm. Nevertheless, the performance benefits and increased security, especially if a trusted platform module [68] is used, represent the two primary advantages of a low-level solution. Table 3.5 presents a brief summary of the key characteristics, associated with this IDS.

The three primary components of the ABID architecture are: audit collection (sensor), audit pre-processing and anomaly detection (classifier). The audit collection mechanism monitors all communications between wireless nodes and APs. In addition, it generates audit records that are logged in a file. On the other hand, the audit pre-processing component formats the audit data, based on the type (AP or user) of detection to be performed, by the anomaly detector. This component fulfills the profiling and detection requirements of the AIDS.

In terms of profiling, local routing information, including traffic statistics and geographical co-ordinates, represent the features used for APs. User profiles are generated using patterns of system usage, e.g. commands. One of the distinguishing characteristics of this approach is the use of *dynamic* profiles. They are continuously updated, using current behavior, which has been observed for a given period of time. This strategy permits the anomaly detector to take this information into consideration. Finally, anomaly detection is carried out using the genetic algorithm and threshold values.

Genetic algorithms [120], which mimic the natural evolution of biological species, have been used to obtain a near optimal solution to a problem. The algorithm's three main components: crossover; selection of the fittest using a fitness function; and mutation are applied recursively, to the remaining set of solutions, until the stop criterion is met.

The fitness function, in this case, is based on the entropy (trend, in particular), associated with user behavior, e.g. system usage. Behavior entropy represents the randomness of a set of commands, used in a user session. More specifically, it indicates the frequency with which the various commands have been issued, in a given session. Thus, frequent changes in a user's (legitimate or attacker) behavior result in a large entropy value. The fitness function is defined by $I = [\sigma_x - \phi]$, where the first term indicates the entropy of a predicted set of commands and the second term represents the average entropy of previous n sets of commands.

Hence, after having monitored $n+1$ sessions, sessions $1, 2, \ldots, n$ are used as input to the genetic algorithm. The output of the algorithm, i.e. a predicted session, describes the normal or expected behavior of a user. Next, a three-tuple value, namely the match index, entropy index and newness index, is calculated by the comparator. It uses the commands in the predicted session as well as the most recent session $n+1$ for this purpose. Finally, potential intrusions are identified using threshold values.

As the primary objective of the authors is to develop a low-level IDS, the adoption of Field Programmable Gate Array (FPGA) technology is appropriate. This technology can be used to create specific hardware circuits, which are tailored to the computational requirements of an application. In particular, FPGAs are composed of a matrix of logic cells that are overlaid with a network of wires. In addition, both the computation, performed by the logic cells, and connections between the wires, can be reconfigured quickly and on-demand (using static random access memory), based on the requirements of a computation. The evolution of the fabrication technology has also resulted in the development of FPGAs, with increased computational density, i.e. number of gates, and power, at a lower cost. Thus, the use of FPGAs represents a cost-effective solution.

In order to leverage this technology, the comparator is designed using a public tool, namely the *Automated synthesis of efficient IDS on FPGAs.* This tool automates the development of FPGA architectures using system-level optimizations. Moreover, given the need to match large strings against very large pattern databases, the optimized string matching feature becomes an invaluable asset.

Although the authors demonstrate the feasibility of using genetic algorithms and FPGA architecture, with respect to user behavioral profiles, it stands to reason that these components can also be used, in a similar manner, to detect RAPs.

802.11-A05: RAPs - AP spoofing

Another incentive for using external RAPs is to mount an AP spoofing attack, also known as *phishing* [58], connection hijacking and evil twin [124]. A RAP, which is programmed with the SSID of an authorized AP, is deployed. In addition, its signal strength is increased until it surpasses that of the authorized AP. This action causes client devices to reduce power and to associate with the RAP. Furthermore, an attacker is capable of obtaining confidential information, such as passwords and credit card numbers, via spoofed web pages and sign-on screens that are served to unsuspecting victims. This information is subsequently used to impersonate users and to initiate various attacks on the network. According to Karygiannis and Owens [97], it is the lack of mutual authentication, e.g. lack of authentication of beacon frames in the shared-key authentication scheme, which permits AP spoofing to be carried out.

Finally, to make matters worse, some APs have features or settings, which make their presence difficult to detect by network administrators. Consequently, this attack can persist for a long period of time. Not surprisingly, the AP masquerading threat, which is realized by this attack, is one of the top security threats in WiFi/802.11 networks, as noted by Ernst and Young [48].

802.11-A05-C01: Location-based approach

As discussed earlier, the notion of using location-based information for detecting MAC address spoofing (802.11-A03) was proposed by Adelstein *et al.* [9]. In fact, the authors of the white paper, from Newbury Networks [124], are also advocates of the multi-factor, e.g. *device* and *location*, access control schemes. Consequently, they briefly analyze three of the generally available methods for locating wireless devices, including RAPs, in a facility/building. Additionally, their primary evaluation criteria is the accuracy of these methods, i.e. the radius within which the device can be located.

Nearest AP This method, being the most simple and least accurate, identifies the authorized (configured by network administrators) AP that is closest to a device, by monitoring its signal strength. Given that these APs can provide coverage, that is well over a radius of 100 feet, the accuracy of this method is limited. That is, the precise location of devices is expected to be within an area of 30,000 square feet on multiple floors.

Triangulation As aforementioned, triangulation or trilateration is a common approach, which has been adopted and subsequently implemented by various research teams and service providers. With an error rate of plus or minus 20 feet in any direction, the underlying algorithm can effectively pinpoint the location of devices to within two floors. Hence, the detection of RAPs is feasible. Although it represents an improvement over the previous method, triangulation is seriously prone to a number of environmental factors, which influences its accuracy. These include multi-path, attenuation, occlusion and reflection of RF signals.

Pattern Matching According to the authors, the most accurate method to date is RF pattern matching. Unlike the previous methods, which are negatively impacted by environmental factors, this method takes them into consideration when determining the location of devices. These systems are implemented as follows. First, RF signatures, which represent RF signals at different points in physical space, are defined. Then, these signatures are associated with predefined locations, such as conference rooms, hallways and parking lot. Next, sensors listen for RF signals from a wireless device and transmit them to a pattern matching system, which matches the collection of sensor data to a library of RF signatures and identifies the physical location of the device. With an accuracy within ten feet, this method not only permits the detection of RAPs, but also the differentiation between internal (inside facility) and external ones.

While the primary objective is to detect RAPs using location-based information, appropriate response strategies can also be initiated depending on whether their lo-

cation is *inside* or *outside* the virtual WLAN perimeter of an organization.

3.2.2 Commercial and non-commercial WIDS

Given the popularity of 802.11 wireless networks and the severity of the aforementioned attacks, it is not surprising that a number of commercial WIDS have recently been deployed. In order to limit our discussion of this topic, only a subset of these systems is presented in this section. Although all of them do address one or more threats, according to Lim *et al.* [106], none provide adequate protection, especially for larger deployments.

AirDefense

One of the well known products in the market is AirDefense [89]. It is a complete IDS that comes equipped with hardware and software. The sensors, which capture all relevant information, are deployed throughout the network. They interface with a management console that carries out intrusion detection. In addition, this product also diagnoses potential vulnerabilities in the network, eg. misconfigurations. Moreover, it offers other management functions including fault tracking and inventory auditing. With the introduction of ActiveDefense, a complementary product, intruders will be forced to dissociate from valid networks, and optionally re-associate with a *honey pot* AP(HAP). A HAP is an AP that is expected to be probed and compromised. The use of HAPs is intended to divert the nefarious activities of intruders, away from the essential APs of a network.

Unlike other products, AirDefense does address the issue of MAC address spoofing to some extent. Their strategy is to fingerprint wireless cards based on the address prefix. This information uniquely identifies the manufacturer of a card. Henceforth, each time a user communicates with an AP, the IDS compares the address prefix to that stored in the IDS database. An intrusion is detected if there is a discrepancy.

AirMagnet

Another commercial product, which runs on laptops and handheld devices, is AirMagnet [113]. It comes bundled with a Cisco wireless card. Like AirDefense, it is capable of identifying vulnerabilities and detecting intrusions. More specifically, it detects RAPs and unauthorized clients, as well as DoS attacks.

While the full set of features is noteworthy, the use of flooding, as a response mechanism, does raise some concern. If this mechanism is not properly implemented, it could have an adverse effect on the overall performance of the network. Even if the channel, being flooded, is typically different from that used for normal operations, interference between adjacent channels can still occur. A more suitable alternative has been proposed by Lim *et al.* [106]. They recommend the transmission of specially crafted frames targeted at an intruder. These malformed frames could exploit software vulnerabilities, associated with the implementation of 802.11b specification, and force the software to crash.

A similar product is Surveyor Wireless [43]. This product requires that a technician perform a perimeter check, i.e. visit different parts of a network, in order to detect potential security threats. Although it can be used by intruders also, the cost is prohibitive at this time.

Non-commercial WIDS

In terms of non-commercial products, AirSnare [38] is a Windows program that detects DHCP requests from unauthorized devices (MAC addresses). In the event of an intrusion, an alert is sent to an administrator. Furthermore, an optional message is sent to the intruder, via Windows net messaging.

A non-commercial product, which is customized for the Linux platform, is Fake AP [47]. It simulates a user-defined list of APs, through the use of 802.11b beacon frames. This mitigation strategy results in confusing an intruder, who is passively sniffing the network.

Weakness	ID	Attacks	Countermeasures
GSM-W01	GSM-A01	BTS spoofing	Not identified
GSM-W02	GSM-A02	SIM cloning	Mobility profiles, Usage profiles
	GSM-A03	Phone theft	Use of IMEI, Usage profiles User mobility profiles
	CDMA-A01	Cloning (CDMA2000)	Count parameter, Duplicate detection, Velocity trap, Voice recognition, Usage profiles

Table 3.6: GSM/CDMA2000: Attacks and Countermeasures

3.3 ABID in GSM/CDMA2000 Networks

As we all know, the popularity of wireless communications is not limited to WLANs. The tremendous increase in the use of cellular phones world-wide is not only taxing the resources of the underlying infrastructure, but is also exposing security issues, associated with interoperability between cellular networks, backward compatibility and technology-inspired attacks, such as cloning.

3.3.1 Attacks and Countermeasures

While the authentication protocols, employed in the three cellular networks, continue to serve their intended purposes, they are nevertheless susceptible to the attacks, identified in Table 3.6. Furthermore, in the case of GSM and CDMA2000, they were neither designed to detect the cloning of identity modules nor the use of stolen phones. However, countermeasures have been proposed for addressing these problems.

GSM-A01: BTS spoofing

If mutual authentication is not supported, it is possible for an attacker to install a false/Rogue Base Station (RBS), with the same Mobile Network Code of users, as with RAPs. In order to attract potential victims to a RBS or false network, an attacker sets the cell reselection parameters, e.g. $CELL\_RESELECT\_OFFSET$, to high values, as indicated by Quirke [141].

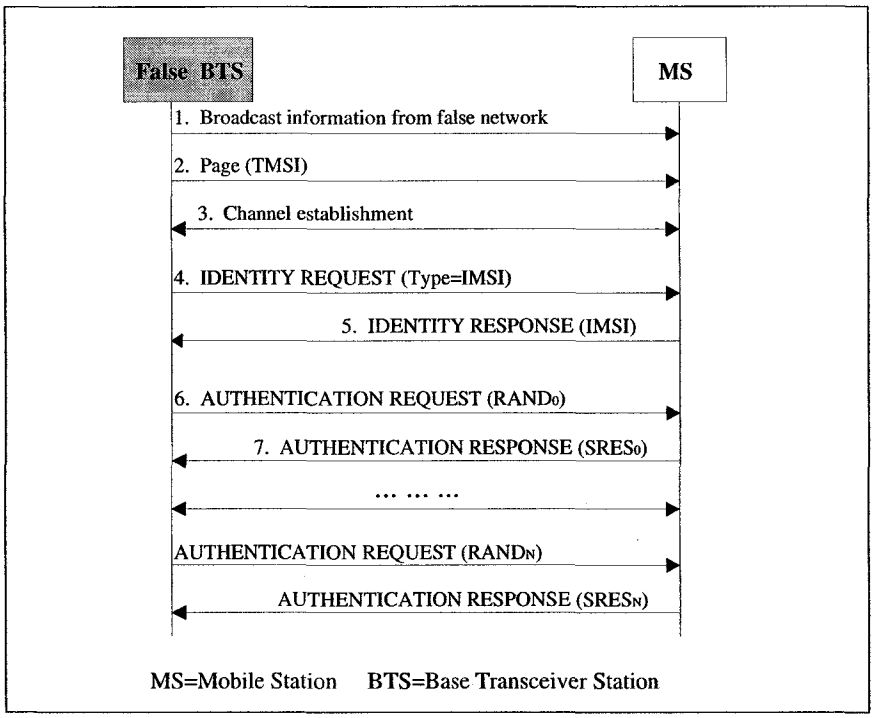

Figure 3.3: Obtaining IMSI and secret key over the air

When authentication is initiated by a MS, a false network may send authentication challenges and simply ignore the responses, or it may terminate the authentication procedure itself. Moreover, it may choose not to initiate ciphering, thus forcing all user data to be transmitted in the clear. This scheme could permit a RBS to mount a man-in-the-middle attack, whereby all calls of a user are eventually routed to the public telephone network. Finally, the impersonation of a legitimate network or BTS, can be carried out by an intruder with the goal of obtaining both the IMSI and K_i.

This attack proceeds according to Fig. 3.3.

1. An intruder impersonates a legitimate GSM network by installing a RBS or false BTS (FBTS).

2. The false BTS pages a MS using the TMSI of the corresponding user. A random user is typically selected by monitoring radio traffic.

3. A connection is established with the MS that responds to the page.

4. The IMSI is subsequently acquired by the FBTS, which sends the IDENTITY REQUEST message to the MS. According to the specifications, the MS must respond to this message at all times.

5. In response to the previous request, the MS provides the IMSI of its user. The attacker has now obtained half of the information required to mount an impersonation attack.

6. In order to discover the K_i, the attacker carefully selects a $RAND$ value, which exploits the weakness of the COMP128 algorithm, and transmits it to the MS, using the AUTHENTICATION REQUEST message.

7. The MS, unable to authenticate the FBTS, sends a request, to the SIM, to carry out the COMP128 algorithm. It then forwards the resulting SRES to the FBTS, using the AUTHENTICATION RESPONSE message. By sending a series of authentication requests to the MS, the attacker *eventually* discovers the secret key, K_i.

One may wonder as to how long it would take to discover K_i. Given that approximately four challenges can be issued per second [141], the expected time is equivalent to $2^{17} RANDs$ or 9 hours. Since most battery-powered terminals are unable to sustain constant transmission for this period of time, an attacker would most likely execute the attack, using multiple sessions. Of course, she would be required to maintain the context of the challenges and responses, which have been sent and received. However, should she opt to use a more sophisticated attack, which makes use of 3 Rounds (3R), 4R or 5R, the 9 hour timeframe can be reduced to one hour, thus rending this type of attack more realistic.

Equipped with the IMSI and K_i, she can now impersonate that user, i.e. make/receive calls and SMS messages using their account. What is interesting, but alarming, is that she can be positioned many kilometers away and yet successfully obtain both pieces of information. Finally, the fact that the two largest operators (Telstra and Optus) in Australia, which have approximately 80% of the mobile market share [141], are vulnerable to this attack is rather unsettling.

GSM-A02: SIM Cloning

According to Prism Group marking director, Steven Sidley, there had been an alarming rise in the cloning of SIMs during 2000-2002 [107]. Although the frequency has diminished over the years, a more recent incident of SIM cloning has been reported by Financial Times in February 2005 [176]. Apparently, the perpetrator was able to clone two cards within 30 minutes.

A SIM can be cloned by extracting the IMSI and K_i from its memory, using one of two methods, and transferring them to another SIM. On one hand, an attacker can obtain these values by mounting a BTS spoofing attack (GSM-A01). On the other hand, when physical access to SIMs is possible, he can extract the K_i using a PC and a smart card reader, as illustrated by authors Briceno and Goldberg [26]. Obtaining the IMSI value is considerably less time consuming since it is transmitted in the clear at the start of the authentication process.

While these methods of attack may differ, they exploit the same vulnerability in the COM128-1 authentication algorithm. In fact, other documented attacks on COMP128-1 and step-by-step instructions e.g. GSM SIM Cloning for Dummies, are readily available on the Internet. This enhanced set of tools has permitted perpetrators to build systems for the purpose of cloning SIMs.

GSM-A03: Impersonation attack using stolen phones

The use of stolen phones, to impersonate a legitimate user and to obtain unauthorized access to services, continues to be problematic, and for a good reason. This type of attack cannot be detected using existing authentication mechanisms.

GSM-A03-C01: Use of international mobile equipment identity

In order to address the issue of stolen phones, some service providers make use of a black list, which includes the International Mobile Equipment Identity (IMEI) value that is factory-fitted into GSM phones. When a stolen phone is reported by a legitimate user, the corresponding IMEI is added to this list. Henceforth, the use

of this phone, by an attacker, can be verified by requesting the MS to transmit its IMEI.

Other countermeasures include the use of user mobility profiles and usage profiles, which are discussed next.

GSM-A02/A03-C01: Mobility and Activity Profiles

One of the most cited research initiatives, in the area of cellular network security, is the Intrusion Detection Architecture for Mobile Networks (IDAMN), by Samfat and Molva [149]. In order to overcome the aforementioned drawbacks, associated with previous approaches, and to satisfy the need for a *real-time* IDS, the authors have defined a *distributed* architecture. It permits the detection of intruders in the visitor location, as opposed to the home location, and within the duration of a call. Furthermore, two different algorithms are used to model usage patterns and mobility behavior of users. These algorithms and other components of IDAMN have been designed to minimize overhead, which is incurred in the wired segments of the cellular network, and to fulfill other requirements for real-time detection.

Architecture of IDS

In terms of the underlying architecture, there are two key entities, which support the intrusion detection mechanisms. The Global Monitor (GM) is primarily responsible for the management of user profiles. As a component of the IDS, it is connected directly to each intrusion detector (ID) and to the HLR.

An ID, in turn, is directly connected to a MSC, the GM and a VLR. It obtains a copy of the relevant signalling messages, generated by a user, from a MSC that manages the corresponding MS. Based on the nature of these messages, it initializes a set of statistical variables. Furthermore, it obtains the profiles, associated with that user, from the GM. It subsequently detects potential intrusions by comparing the statistical variables to the corresponding data in the profile, and by applying appropriate thresholds.

Components	Details
Features in user profiles	Activity and roaming patterns
Classification/Detection	Statistical measure and threshold values

Table 3.7: Components of IDS for GSM

As previously stated, the two key phases, associated with ABID systems, are profiling and classification/detection. Table 3.7 provides a high-level overview of the key components of this IDS. In the case of GSM, two user profiles are used for the purpose of intrusion detection. Whereas mobility/roaming patterns are used for one of the profiles, the other is based on activity patterns, e.g. speech and dialed number analysis (forthcoming). In addition, the two detection mechanisms, associated with the profiles, raise different alarms. These alarms are analyzed by a rule based classification system in order to render the final decision.

Activity Behavior: Profiling Phase

The usage behavior of users is defined by two statistical vectors: the call vector and session vector. As aforementioned, the values of these vectors are initialized by the ID. The call vector $\vec{V}_i^c$, which represents a local magnitude, is represented by:

$$\vec{V}_i^c = [td_i - tf_{i-1}, tf_i - td_i, nh_i]$$

where $\vec{V}_i^c$ models the i^{th} mobile originated call, $td_i - tf_{i-1}$ denotes the time period between two calls, $tf_i - td_i$ represents the duration of an outgoing call, and nh_i is the number of handovers performed.

On the other hand, the session (one day of connection to the network) vector $\vec{V}_i^s$, a global magnitude, is defined as:

$$\vec{V}_j^s = [D_j, Dc_j, Nh_j, Nc_j]$$

where $\vec{V}_j^s$ models the j^{th} session of the user, in terms of the duration of the network connection D_j, total number of calls Nc_j, total duration of these calls Dc_j, and total number of handovers Nh_j.

During the profiling phase, a subset of vectors, which represents the behavioral history of a user, is typically selected and stored in a profile. However, in this case, an activity profile P_A is composed of a mean vector $\vec{M}_n$, covariance matrix C_n and a forgetting factor $0 < \alpha > 1$. The use of these elements, instead of actual vectors, provides the following benefits. First, and most importantly, the size of the profile is reduced from $(m * n + 1)$ to $(m^2 + m + 1)$, where m denotes the dimension of a statistical vector (e.g. m=3 and m=4 for the call and session vector respectively) and $n < 30$ is the number of vectors used. For example, setting n to 20 requires 61 bytes of memory for the call vectors versus 13 bytes for the P_A. Therefore, this strategy minimizes both memory and transmission (between GM and ID) requirements. Second, unlike many other research initiatives undertaken to date, it also permits the IDS to recursively update the profile of a user, in order to capture his/her current behavior (activity).

Activity Behavior: Classification Phase

Once the initial P_A has been created, the next step is to determine whether $\vec{V}_{n+1}$ is normal or anomalous. This decision is made based on the distance measure defined in Eq. 3.1, where C_n^{-1} denotes the inverse of the covariance matrix, and S^2_{max} represents the threshold. It is computed, using Eq. 3.2, and is based on the first n vectors that have been observed. If the distance measure is less than the threshold, then $\vec{V}_{n+1}$ is considered normal.

$$(\vec{V}_{n+1} - \vec{M}_n)^t C_n^{-1} (\vec{V}_{n+1} - \vec{M}_n) \geq S^2_{max} \tag{3.1}$$

$$S_{max} = max\{(\vec{V}_i - \vec{M}_n)^t C_n^{-1} (\vec{V}_i - \vec{M}_n)\} \ \ (for \ \ i \ \ \varepsilon[1 \ldots n]) \tag{3.2}$$

Vector	Category	FAR	DR	m=test size
Call	Business (during business hours)	1%	88-100%	600-2000 vectors
Session	Roamer (any time)	1%	95-100%	300 vectors

Table 3.8: Activity Behavior: Simulation Results

Activity Behavior: Simulation and Results

The simulation platform, which encompasses the components of IDAMN and a Wireless Network Simulator (WINES), is used to simulate the functionality of a GSM network and MSs at the protocol level. Moreover, IDAMN can be customized to support other cellular networks.

Table 3.8 presents the optimal results of the simulation exercise. In order to test the performance of the IDS, two performance metrics, e.g. the FAR and DR, were used. Whereas FAR is defined as $\frac{n}{m}$, where m denotes the total number of vectors and n being classified as anomalous, DR is based on the detection of n of m intrusive vectors. Finally, the criteria, used for the categorization of real GSM users (400 users used in simulation), had been obtained from a French operator.

For both categories of users, the low FAR is an indication that the activity characteristics of users have been captured in an accurate manner. It also symbolizes the consistent behavior of these users over time. While this outcome is reasonable for business-class users, who typically place local or national calls, it seems somewhat unusual for those, considered as roamers. Even if session vectors had been used for classification purposes, this would imply that each individual session, i.e. the total number and duration of calls, initiated by this group of users, has remained consistent during the entire simulation period.

On the other hand, the difference in the DR can be attributed to the *uniqueness* of a user's behavior, in comparison to others in the same category. So, for example, if roamers A and B frequently make international calls during a session, the individual attributes of their sessions, eg. the number handovers, would very likely be different. Hence, the probability of detecting an intrusion, i.e. user B masquerading as user A, would also be higher. However, the difference in behavior is not as pronounced with users in the business category, whether session vectors or call vectors had been used.

Roaming Behavior: Profiling Phase

In addition to the use of activity patterns, roaming or mobility behavior of users is also employed for intrusion detection. A user's roaming profile P_R is defined by the LAs visited by a user, as well as the frequency with which these itineraries are followed. It is also modeled as a graph, where each LA is represented by a state and each transition probability is established, based on the frequency of LA crossings. Finally, when a location update signalling message is received from a MS, the profile of that user is updated. This procedure is typically carried out upon LA crossings.

Based on the sequence of LAs, visited by a user, the corresponding transition probabilities are updated. More specifically, the transition probability of moving from LA_i to LA_j is defined as:

$$\rho_{ij} = \rho\langle X_n = j | X_{n-1} = i \rangle = \frac{n_{ij}}{n_i}$$

where X_n is a stochastic variable that specifies the LA in which a MS is located after the n^{th} location update, n_{ij} represents the number of times the user had previously crossed from LA_i to LA_j, and n_i denotes the total number of crossings from LA_i to any of the adjacent LAs ($\sum_k n_{ik}$).

Roaming Behavior: Classification Phase

Once the roaming behavior of a user has been adequately characterized by his/her P_R, the classification or detection phase is initiated. The intrusion detection process is carried out by the ID, which compares the current mobility sequence in progress, referred to as the candidate $\hat{C}$, to the P_R. As $\hat{C}$ is composed of k LAs, the average probability $\bar{\rho}_i$, for each LA_i, is determined according to:

$$\bar{\rho}_i = \frac{1}{O_i} \sum_{j=1}^{n} \rho_{ij} = \frac{1-\rho_i}{O_i} \quad i \ \varepsilon[0 \ldots k]$$

where O_i represents the total number of outgoing transitions from LA_i.

The final decision is rendered, as to whether the candidate itinerary is normal or anomalous, by analyzing the doubt factor $V_d \geq 0$. The transition from LA_i to LA_j is

Category	FAR	DR
Business (during business hours)	3-4%	75-90%
Roamer (any time)	5-7%	65-80%

Table 3.9: Roaming Behavior: Simulation Results

considered anomalous if $\rho_{ij} \geq \bar{\rho}_i$. In this case, V_d is reduced by KV_d with $(0 < K < 1)$. Otherwise, V_d is increased by a positive value depending on the following criteria:

- If a transition from LA_i to LA_j exists, then H is added to V_d.
- If it does not exist, then V_d is incremented by $\frac{F}{2^l}$, where l denotes the number of LAs in $\hat{C}$ that have never been incorporated into P_R.

Parameters F and H are dependent on the average length of a user's itineraries.

Therefore, the itinerary is considered anomalous if $V_d > 2F$. Otherwise, the ID updates P_R. The decision criteria accommodates new itineraries of legitimate users who roam into the visited domain. Moreover, the detection algorithm is executed *on-line*, each time a location update signalling message is processed. As a result, deviations from pre-established itineraries are immediately detected. Upon detection of an intrusion, IDAMN signals the network to either activate a call bearing procedure or to disconnect the user.

Roaming Behavior: Simulation Results

Table 3.9 presents the results of FAR and DR, for users in the business and roamer categories, as with the previous section.

In comparison to the results presented in Table 3.8, the FAR is higher while the DR is lower for both categories of users. Using mobility patterns for intrusion detection is challenging at best. It appears as if mobility behavior of users is less consistent than activity-based behavior. This would be particularly true for users, who tend to roam. It would also explain the increase in the FAR between business users and roamers, and between the use of activity and mobility profiles by roamers. In terms of the DR, the lack of consistency in behavior, results in a lower probability of distinguishing intrusive vectors from normal ones.

While these results are promising, there are issues that warrant further research. For example, there is a remote probability that an intruder can successfully impersonate a legitimate user, even if she does not know *a priori* the activity or mobility behavior of the user. Under these circumstances, it will be difficult to achieve a DR of 100%. Furthermore, the innate characteristics of human behavior also rule out the possibility of obtaining a FAR of 0%. Nevertheless, the fact that an alarm can be raised in less than one second, as a result of using optimized algorithms, small profiles and distributed detection, is noteworthy.

GSM-A02/A03-C02: User Mobility Profiles

As presented in section 2.3 and briefly described here, the GSM network is distributed in order to re-use transmission frequencies. Moreover, there are several MSCs and associated databases, e.g. VLRs, which serve their designated areas. A region, which is managed by an MSC, is subdivided into several LAs. These areas are, in turn, subdivided into several cells, the smallest unit in a cellular network.

This infrastructure supports the two primary procedures used for tracking and locating a user, namely location update and paging. Thus, by employing more adaptive and dynamic tracking algorithms, the mobility and behavioral patterns of users can be defined, as suggested by Buschkes *et al.* [142]. Their approach is to identify the routes of a user, such that the GSM network can predict the cell, most likely to be occupied by a MS, at a given point in time. When there is an anomaly, i.e. major deviations from the route, the user can be prompted for his/her current status. Hence, misuse by an attacker can be detected, by comparing the time and place of the call to the normal/standard behavior of the legitimate user.

In order to predict the location (cell) of a MS, two mobility profiles are created using the Bayesian (universally applicable) and mean residence time (domain-specific) algorithms. The use of these algorithms, in particular, the Bayesian algorithm, fulfills the requirements of a statistical ABID. Table 3.10 presents a brief summary of the key characteristics of the IDS.

Components	Details
Features in user profile	Cell location and mean residence time
Classification/Detection	Statistical measure and threshold values

Table 3.10: Components of User-mobility IDS

Bayesian Algorithm

The Bayesian algorithm makes use of the Bayes decision rule for achieving minimum error rate. According to Duda and Hart [44], this decision rule is widely used in statistical pattern recognition systems. A pattern recognition task is carried out by determining the probability of an observed vector x belonging to each class c, i.e. $p(c|x)$. Finally, the c with the highest probability is selected. Probability $p(c|x)$ is formally defined as:

$$p(c|x) = \frac{p(c)p(x|c)}{p(x)}$$

where $p(x|c)$ is the *class conditional* probability density of observing x, $p(c)$ represents the *a priori* probability for c, and $p(x)$ is the probability density of observing vector x.

Within the context of mobility profiling, vector x is represented by a sequence of observations, while class c is equivalent to a cell. In order to calculate $p(c)$ and $p(x|c)$, a histogram is created by dividing the time axis into intervals of length δt, and by categorizing a sequence of observed vectors, e.g. $t_{1,c1}, t_{2,c2}, \ldots, t_{n,cn}$ with $t_{i+1,c_k} - t_{i,c_i} = \delta t$.

Mean Residence time Algorithm

Unlike the Bayesian algorithm, the mean residence time in each cell is obtained as follows: First, different movement patterns of a user are extracted from the observed mobility behavior. Second, residence times, associated with the samples of each movement pattern, are used for calculating the mean residence time in each cell. A movement profile is subsequently constructed, based on the average values.

In light of the fact that a user typically takes the same route, through a cell, minimal residence time T_{min} is determined by $\frac{S_{cell}}{V_{max}}$, whereas actual driving time T is defined as $T_{min} + \Delta T$. While S_{cell} represents the length/distance of the route through

Algorithm	Highway Scenario	City Scenario	Convergence state
Bayesian	0.952-0.955	0.835	15 days
Mean Residence Time	0.952-0.955	0.835	15 days

Table 3.11: Simulation Results

the cell, V_{max} is the maximum speed of a user, and ΔT denotes the variation in driving time. It is restricted to $0 \leq \Delta T \leq \infty$, and is modeled as an exponentially distributed random variable.

General Results

The Mean Prediction Level (MPL) is used for evaluating the performance of the Bayesian and Average algorithms. The MPL represents the empirical probability that a user is actually in the expected cell, at a given time. In addition, it is calculated as $\frac{hit}{hit+miss}$, where *hit* represents the number of successful verifications, as described in the sequel.

In order to determine the impact of different precision levels, i.e. different number of cells in an LA, on MPL values (90% confidence interval), two scenarios are considered, as indicated in Table 3.11.

For both scenarios, the most probable cell is determined using the profile of a user. Its corresponding LA is then compared to the actual LA of the user. As indicated, the MPL values, associated with the highway scenario, are not only identical for both algorithms but are also higher than the city scenario. These results reflect the use of a larger cell size in the highway scenario.

While the results are encouraging, the authors fail to clarify the precise type of mobility behavior used for the calculation of MPLs. However, they do conclude that approximately 75% of the users, e.g. working people and housekeepers, are potential candidates for the successful application of ABID techniques. Only high-mobility users, with chaotic behavioral patterns, are not suited for this purpose. This conclusion is in synch with the criteria used, by the authors of the UMTS Research on Advanced Communications in Europe (RACE) specification, for categorizing mobile users. As a side note, this specification, on mobility management, explicitly takes into

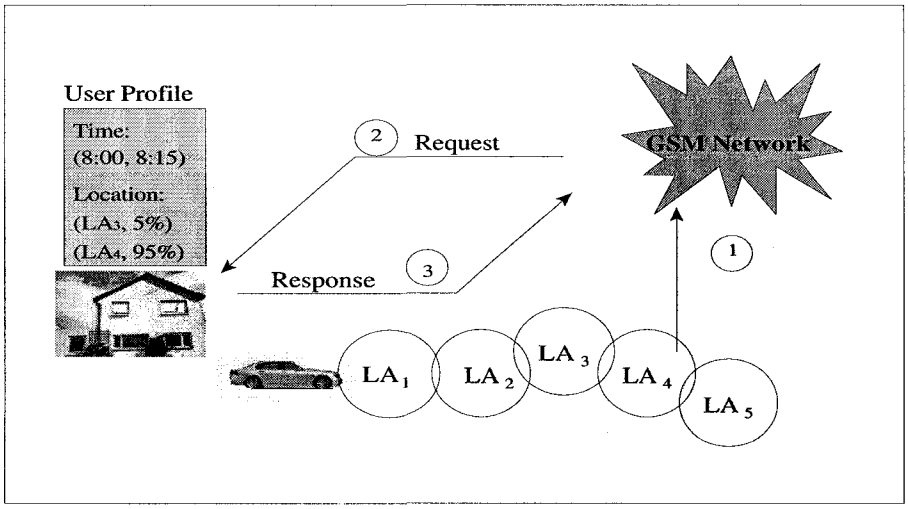

Figure 3.4: Supporting user privacy using HTD

consideration, the different mobility behavior of users. Consequently, the underlying model accommodates different user classes, environments and geographical areas, e.g. workers, driving in a bus, etc.

Privacy of users

In order to accommodate users' need for privacy, the authors propose a trusted and private environment, that is realized using a Home Trusted Device (HTD), see Fig. 3.4.

The details of the HTD model are as follows.

- A MS makes use of the classical location tracking algorithm in order to obtain the daily mobility patterns. This information is subsequently transmitted to the HTD, on a daily basis and in a secure way, through a direct connection to the HTD. The use of secret keys and authentication fulfills the need for adequate security. After this initial phase, a user specific profile is generated by and stored in the HTD.

- In the event that a call has to be established (1), the GSM location tracking algorithm sends the current location of the MS to the corresponding HTD (2).
- The HTD, in turn, compares this information to that stored in the profile. The result of this verification is transmitted to the GSM network (3), which establishes the call, if there is a match. Otherwise, special measures, which are also controlled by the user, are taken (not specified by the authors). Finally, it is assumed that request/response signalling messages are protected using a symmetric cryptographic system.

This model does have a key advantage. In general, the mobile network scenario is not sensitive to certain attacks, which are typically associated with a cellular network. For example, an intruder cannot fool an ABID system, by changing his/her behavior slowly from *good* to *bad*. He must either steal a MS or clone the GSM card within it. In the first case, only one user, with an exact identifier, actually uses the network. Hence, a difference in mobility behavior would be detected by the ABID. In the latter, two users are present in the network. Nevertheless, the mobility behavior of the intruder is likely to raise an alarm. In any event, neither one of the attacks provide an attacker with sufficient time to slowly modify the profile of a user.

GSM-A02/A03-C03: Mobility Profiles using cell identifiers

In contrast to the approach adopted by Buschkes *et al.*, Sun and Yu [167] make use of cell IDs, traversed by users, as the basis for anomaly detection. They propose an on-line detection algorithm that is capable of detecting masqueraders, who impersonate a specific group of legitimate users. This threat can be realized using a stolen phone or a clone. In addition, they not only assume that this group of users, which does not include taxi drivers and those on vacation, typically maintain regular or consistent patterns of mobility, but that an intruder is also incapable of duplicating these patterns, with a high degree of accuracy. These assumptions are consistent with those made by Yu and Leung [195]. These authors propose the use of mobility patterns for improving the performance of QoS provisioning and resource allocation in wireless cellular networks.

Components	Details
Features in user profile	List of cell IDs traversed by user
Profile updates	EWMA
Classification/Detection	Statistical measure and thresholds

Table 3.12: Components of Optional service

Although the motivation of Sun and Yu is to develop an on-line detection system, which alerts legitimate users (via an optional service) upon detecting potential intrusions, the underlying framework is nevertheless similar to other research initiatives.

Other key aspects of the system, see Table 3.12, include the use of: high order Markov model [37] for characterizing mobility patterns of a user; Ziv-Lempel data compression technique [197] for parsing mobility data and storing the resulting statistical information in a mobility trie (profile); Exponentially Weighted Moving Average (EWMA) [94] for updating the mobility trie, i.e. to reflect the current behavior of a user; and thresholds for ABID.

Profiling Phase

The feature extraction process is carried out in order to create a profile for each user. They are stored, along with other information, e.g. billing, in the HLR. The key objective is to extract features that best characterize the mobility patterns of a user. In this case, the cell numbers, traversed by a user, are used to define a profile. According to the authors, it is a reasonable choice, since this feature is relatively stable, i.e. less deviation of mobility behavior over time, and the resulting alphabet is small.

Once a sequence (string) of cell numbers (symbols) has been obtained for a user, a mobility trie is constructed using the Ziv-Lempel algorithm. In particular, the character based version (LZ78) is used for parsing the input S, i.e. a string of symbols, into variable length phrases $x_1, x_2, \ldots, x_n$, based on the prefix property. This property states that, for $j > 1$, there exists $i < j$ such that x_j equals $x_i + c$, and where c is a character in the alphabet. The resulting phrases are stored in a trie, which is a multiway tree with a path from the root to a unique leaf. Moreover, only the unique

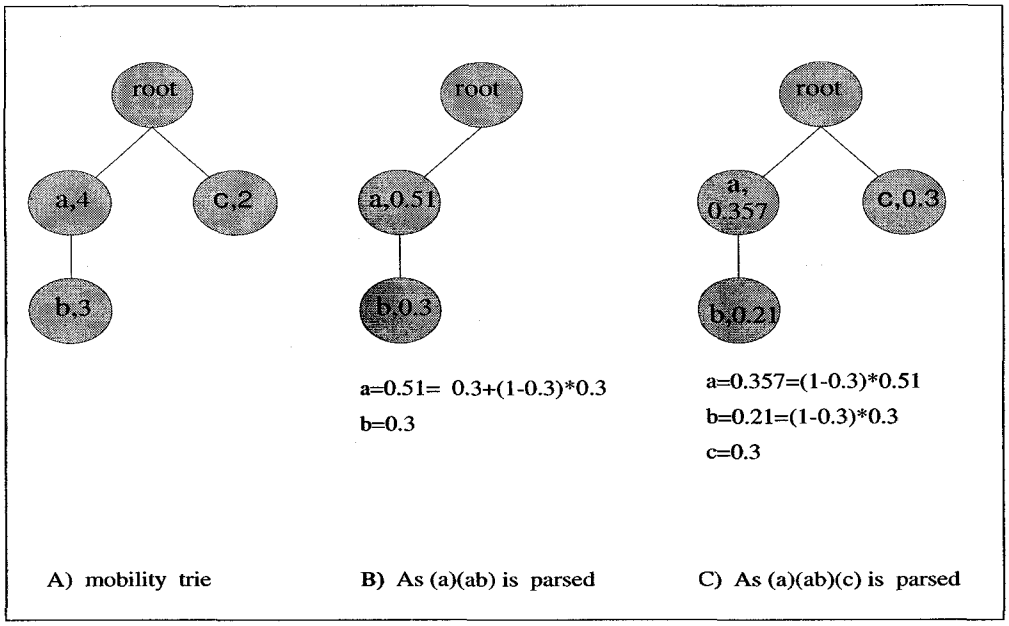

Figure 3.5: Example of a mobility trie

prefix of each string is stored since the suffix can be obtained by performing a string search. This form of data compression proves invaluable in minimizing storage and memory requirements.

Part A in Fig. 3.5 presents an example of a multiway tree that has been created based on the alphabet (a, b, c), string S (aabc) and parsed phrases $(a)(ab)(c)$. The numbers, associated with each of the nodes, represent the frequency with which each node had been traversed by the user. As the mobility tree reflects the transitional probability of S, the probability of a given sub-string can also be determined. So, for example, the probability of $(a) = \frac{4}{6}$ or $\frac{2}{3}$, whereas the probability of $(c) = \frac{1}{3}$.

The m^{th} Markov model, which is based on the prediction by partial matching (PPM) scheme [33], is used for calculating the probability of a given symbol. The calculation is based on the m predecessors of a symbol. Given that the value of m is difficult to predict, the authors adopt a *blending* strategy. This approach is used to first calculate various probabilities, using a number of models with different orders, and then to obtain a weighted sum of these probabilities. In general, the prediction

accuracy of a model increases as the order is increased. Consequently, a larger weight is assigned to models with larger orders. The blended probability of $\rho(\alpha)$ is defined as:

$$\rho(\alpha) = \sum_{i=0}^{m} \omega_i * \rho_i(\alpha)$$

where α is predicted based on the previous i characters, $\rho_i(\alpha)$ denotes the probability assigned to α by a model of order i, and ω_i represents the weight associated with each of the m models. However, when $i = 0$, the probability of α is calculated in an independent manner. Finally, the weights are established based on simulation parameters.

Although the classification or detection phase can theoretically begin at this point, without further modifications to a user's mobility trie, there is an on-going need to maintain the current mobility characteristics of each user. It is precisely this requirement that necessitates the integration of EWMA into the process of trie creation and maintenance. As defined in Eq. 3.3, at time t, the frequency of each node in the trie is updated according to:

$$F_i(t) = \lambda * \delta + (1 - \lambda) * F_i(t - 1) \tag{3.3}$$

where $F_i(t)$ represents the frequency value (FV) of node i at time t, $\lambda = 0.3$ is the smoothing constant that dictates the decay rate, and δ indicates the status of a match between a symbol being observed and node i. Hence, the FV of a node (cell) in the trie, which has not been visited by a user since $t - k$ is decayed by $(1 - \lambda)^k$. Fig. 3.5, parts B and C, depict the creation of a mobility trie, based on S. As S is parsed, and the construction of the trie is initiated, the following algorithm is applied. When a new node is added, its FV is initialized to 0.3. Over time, this value is either exponentially decayed or increased, based on the incorporation of subsequent phrases.

Classification Phase

Using the EWMA-based mobility trie, the current mobility activity of a user,

represented by sequence $S = (X_1, X_2, \ldots, X_n)$ where $X_i = cellnumber$, is classified as normal or anomalous, i.e. signifying a potential intrusion. First, the high order Makov model is used to determine the blending transitional probabilities of S. For a model with order $i \geq 1$, its o^{th} transitional probability is calculated as:

$$p_o = \sum_{i=1}^{n-o} p(X_{i+o}|X_i, X_{i+1}, \ldots, X_{i+o-1}).$$

On the other hand, when the order is zero, the probability is determined by:

$$\rho_o = \sum_{i=1}^{n} p(X_i).$$

In order to calculate the probability of the transition from $X_i, X_{i+1}, \ldots, X_{i+o-1}$ to X_{i+o}, a search is carried out for the path $X_i, X_{i+1}, \ldots, X_{i+o-1}$, starting from the root. If the path is not found, a probability of zero is assigned. Otherwise, the probability $p(X_{i+o}|X_i, X_{i+1}, \ldots, X_{i+o-1})$ is obtained by $\frac{p(X_{i+o})}{p(X_{i+o-1})}$.

Once the blending probabilities have been calculated for all models of orders $1, 2, \ldots, m$, the next step is to determine the overall probability of S. Using a pre-established weight vector $[\omega_0, \omega_1, \ldots, \omega_m]$, this probability is obtained according to:

$$p = \sum_{i=0}^{m} \omega_i * p_i.$$

Next, the normalcy of S is calculated by using the distance measure $D(S) = \frac{p}{Length(S)}$. Given that p increases, as the length of S is increased, $D(S)$ is normalized using the length of S. The intuition behind $D(S)$ is that it is large for users, who consistently follow a given path. Since this path exists in the mobility trie, many of the transitions at different orders will be found, thus confirming the normalcy of the mobility behavior of a user. On the other hand, an intruder is unlikely to reproduce the exact behavior of a user. This results in a lower distance value.

Finally, if $D(S) \geq P_{thr}$, a pre-determined threshold, sequence S is considered normal. Otherwise, it is identified as an anomaly.

In order to evaluate the proposed detection algorithm, the FAR and DR are used. Whereas FAR is defined as $\frac{n}{m}$, where m denotes the total number of normal itineraries and n being classified as abnormal, DR is based on the detection of n of the q abnormal itineraries. Simulation results indicate the following:

FAR The FAR (maximum: 25%) decreases as the level of mobility increases. The primary factor is the increase in the number of cells traversed by users. However, the classification of relatively long itineraries can produce false positives, especially if the path has low probabilities.

DR Unlike the FAR, an increase in mobility produces a corresponding increase in the DR (minimum: 80%). As the level of mobility increases, the number of cells traversed also increases. Hence, the mobility sequence of an intruder has a tendency to deviate more significantly from the normal behavior of a user. Nevertheless, a longer sequence can also increase the probability of finding a segment of this path, in the mobility trie of a user, thus potentially avoiding detection.

Overall Performance The authors acknowledge the fact that it is impossible to obtain 100% DR or 0% FAR.

CDMA-A01: Cloning (CDMA2000)

Although cloning of GSM phones is still rampant in Asia, the emerging crime is that of cloning CDMA phones. What was once thought to be difficult has now become a reality [174].

Apart from terminology, the cloning of CDMA2000-based mobile phones is similar to that of GSM cloning. More specifically, by programming the unique and factory-coded Electronic Serial Number (ESN) and Machine Identification Number (MIN) into the handsets of mobile phones, new clones are produced within minutes [175]. The ESN/MIN pair can be obtained in a number of ways: using an electronic scanning device or digital data interpreters; social engineering, e.g. convincing a legitimate user or re-seller into releasing such information; hacking into networks of service providers; and other avenues. Once this information has been obtained, it is transferred to a clone using a computer and specialized software such as Patagonia [177]. A copycat box or a plug also provides similar functionality. Finally, there are a number of websites, which provide detailed instructions on mobile phone cloning.

Variations in billing and the rate of dropped calls are two symptoms of cloning. Users, who do scrutinize their cell phone bills and notice an increase in the rate of dropped calls, are encouraged to contact their service provider immediately. However, by that time, the damage would have already been sustained. Not only do legitimate users incur additional expenses, they are also subjected to emotional distress and the delay, associated with the restoration of service.

CDMA-A01-C01: Use of count parameter

In order to address cellular phone cloning, an additional parameter is used by each MS, during the SAKA protocol, as previously indicated. The parameter $COUNT$, which represents the number of phone calls initiated by a MS, is maintained by each MS. During authentication, a MS also transmits this value to the VLR, which compares it to a previously stored value. The objective is to determine if this parameter is being incremented over time.

In the case of cloning, the value of this parameter, which is maintained by a legitimate and cloned MS, is likely to be different. For example, if a legitimate MS had placed n calls, since the cloning exercise, then its current value would be $x + n$, where x represents the total number of calls, prior to the cloning exercise. Now, when the cloned MS attempts to place the first call, its current value would be $x + 1$. Therefore, the VLR can detect a cloned phone and respond accordingly. Unfortunately, it is always possible that a cloned MS starts placing the calls, prior to a legitimate MS. In this case, the extent of the damage that can be sustained, by a legitimate subscriber, can be significant.

CDMA-A01-C02: Other Strategies

This situation is likely to change with the adoption and implementation of the 3G AKA protocol. Until then, service providers in India and other countries have deployed the following techniques to address the problem of CDMA cloning [175]:

Duplicate detection As indicative of the name, this method is used to detect the presence of multiple copies (same identifiers) of a mobile phone, which are simultaneously roaming in different parts of the network.

Velocity trap The rationale behind this test is that it is impossible for a user to be at location X at t=1, and at location Y at t=10, given the distance between the two locations. Hence, there must be two phones with the same identity on the network.

Usage profiles As previously discussed, the usage patterns of a user are compared to his/her profile on a monthly basis. Significant deviations alert network administrators to the existence of fraudulent activities.

Speaker identification Otherwise known as voice recognition, this technique permits an IDS to recognize the voice of a user, based on acoustics analysis. The software, developed by the Central Forensic Laboratory at Hyderabad, has already proved useful in identifying two cloned phones in 2005.

Regulations Consumer groups in India have requested the issuance of directives, by the Telecom Regulatory Authority of India, which would hold service providers responsible for any duplication of mobile phones.

Part III

Outstanding problems

Chapter 4

Device Impersonation by Rogue Devices

The characteristics and nuances of the standards-based wireless networks, discussed in sections 2 and 3, include the following: portfolio of the security services that are provided; strength and weaknesses of authentication protocols; and the attacks which exploit them.

While it may be desirable or even possible to eliminate *all* of these problems, history would prove otherwise. Perfect security is unattainable. The reality is that it is an arms race, where winning is defined as leveling the playing field or simply raising the bar for the next round.

With this in mind, we carry out a high-level assessment of the overall situation, while making references to a specific group of attacks, which continues to be problematic. Categorized under the heading of *device impersonation*, this group is represented by RAP/RBS spoofing, MAC address spoofing and cloning of devices. Based on the high level of publicity, e.g. multiple reports of CDMA cloning and the significance attributed to this problem by current research initiatives, e.g. Ernst and Young [48] and vendors such as Newbury [124], it is clear that device impersonation must be addressed. Although we acknowledge the significance of subscription fraud, we consider it to be outside the scope of this work.

An ARA, a variant of the TRA, is subsequently carried out in order to assess their

relative significance.

Finally, given the simplified architecture and availability of multiple 802.11 wireless cards, we target the detection of MAC address spoofing for further investigation. Results from this exercise should prove useful in formulating appropriate strategies for addressing RAP/RBS spoofing and device cloning.

4.1 High-Level Situational Assessment

Based on the information presented in sections 2 and 3, a number of interesting observations can be made.

1. Most weaknesses are being addressed
2. Common methodology used for most attacks: device impersonation
3. Need for prevention and detection strategies
4. Increased use of ABID

Most weaknesses are being addressed

It appears as if most of the key weaknesses, associated with authentication protocols of wireless networks, are being addressed by the research community and standards organization. For example, the lack of mutual authentication in GSM (e.g. GSM-W01), one of the most significant weaknesses exhibited by the authentication protocol, is specifically addressed by UMTS. This strategic direction is also being adopted by other networks including Worldwide interoperability for Microwave Access (WiMax)/802.16 [88]. Nevertheless, until such time as 3G/4G wireless networks have been fully deployed, the vulnerabilities of their predecessors will remain problematic. As far as the less significant weaknesses are concerned, there is always room for improvement over and above the proposed resolution strategies.

Weakness	Attack	Intrusion Detection	Status
	BT-A04	BT_ADDR Spoofing	Not identified
802.11-W04	802.11-A03	MAC address spoofing	Intruder location, AirDefense User mobility profiles
802.11-W06	802.11-A04	RAP-Unauthorized access	MAC filtering, OUI filtering, Embedded IDS, AirMagnet
802.11-W07	802.11-A05	RAP-Spoofing	Location-based approach
	CDMA-A01	Cloning (CDMA2000)	Various countermeasures

Table 4.1: List of outstanding attacks/problems

Common methodology used for most attacks: device impersonation

With the exception of RAPs (802.11-A04), see Table 4.1, all of the remaining attacks share a common goal of *device impersonation*, i.e. they are directed towards the masquerading as or impersonation of devices. Moreover, it can be instantiated through device theft (e.g. cellular phone, SIM card, or wireless NIC) or by duplicating relevant information from one device onto another, e.g. MAC address spoofing, BT_ADDR spoofing, RAP spoofing or cloning.

This observation is rather logical, especially since masquerading as legitimate devices and/or users, has been identified, by the ETSI technical specification [49], as the most common threat in the area of access control. The threat analysis section of this specification also confirms the relatively high impact of this threat, given that an attacker can neutralize the key security objectives, e.g. confidentiality, integrity, accountability, and availability. To this list, one could also add Authentication, Authorization and Accounting (AAA). In other words, should this threat be realized, the consequences to both users/subscribers and service providers could be significant.

Need for prevention and detection strategies

There are many ways to address a given attack. First, it can be *prevented* by eliminating the underlying vulnerability in an authentication system, or by incorporating other essential mechanisms. This would be the most logical and effective approach. Otherwise, an attack can be *detected* and *diverted* by implementing appropriate countermeasures. Under certain circumstances, e.g. device theft, the only solution would

be to adopt a detection strategy. On the other hand, given that there are no authentication systems that are impervious to attacks, it might be prudent to adopt and to implement both strategies in order to harden access control. A brief discussion of the prevention and detection strategies, associated with the aforementioned attacks, is presented next.

RAP - Unauthorized access

As previously indicated, RAPs specifically exploit the lack of AP authentication by wired networks (802.11-W06). In order to address this problem, the use of a RADIUS server (802.11-W06-R01) has been identified as the key prevention mechanism. However, depending on the implementation specificities of this solution, the prevention of RAPs may remain problematic.

Hence, the use of intrusion detection mechanisms such as MAC/OUI filtering (802.11-A04-C01 and 802.11-A04-C02) may prove beneficial. However, as stated previously, the key disadvantage of this approach is the use of identifiers, e.g. MAC address, that are malleable (i.e. easily spoofed). On the other hand, the ABID approach, adopted by the embedded IDS, appears promising. Finally, the significance of this problem is further confirmed by the availability of commercial WIDS, e.g. AirMagnet.

RAP - spoofing

In the case of spoofing attacks that are carried out by RAPs, the vulnerability being exploited is the lack of mutual authentication between users and APs (802.11-W07). More precisely, it is the lack of authentication of APs by users. In order to address the issue of mutual authentication, different user-level authorization mechanisms, e.g. EAP-based, have been introduced in WiFi/802.11 networks [161] and [14]. However, some of the EAP methods are still being defined. Since security flaws are often uncovered in *unproven* mechanisms, a second line of defense would prove essential. Details regarding security vulnerabilities in EAP methods are available in a Web page maintained by Aboba [6]. Another prevention strategy is the use of link layer

solutions (802.11-S04/05), identified in section 2.2. In light of the fact that these solutions are not standards-based, it is highly unlikely that organizations will opt to implement them.

Given the lack of robust prevention mechanisms, it is no wonder that a significant level of effort has and is currently being expended to address this problem. Theoretically, the same countermeasures and commercial WIDS, used for addressing unauthorized RAPs, could also be employed in this case. In addition, the use of location-based IDS may also prove useful in mitigating the threat of AP impersonation.

MAC address spoofing

The use of an ACL, based on MAC addresses of wireless cards, was adopted at a time when these addresses could not be changed. Thus, they satisfied the need for an acceptable form of identification. However, in time, new software tools have been made available, in order to permit network administrators to change MAC addresses, in a dynamic manner. Unfortunately, the same tools can now be used for nefarious purposes, i.e. to mount spoofing attacks (802.11-A03). Hence, the continued use of a malleable identifier, in ACLs, represents a vulnerability (802.11-W04) in this access control mechanism.

While the use of public-key cryptography represents a viable prevention strategy, it requires that an appropriate infrastructure be in place to support this solution. On the other hand, detection strategies, that make use of intruder location (802.11-A03-C01) and/or user mobility profiles (802.11-A03-C02), could also prove useful.

Cloning (CDMA2000)

Cloning of CDMA phones (CDMA-A01), another actualization of device impersonation, does not exploit vulnerabilities in the CAVE algorithm. As a matter of fact, it is either the lack of authentication or implementation flaws, e.g. use of zero for A-keys, which has prompted intruders to undertake this activity.

Although the use of the *count* parameter has been suggested as a means of preventing cloning of CDMA phones, its success rate has yet to be established. As a matter of fact, given the numerous incidents of CDMA cloning in India last year, the effectiveness of this strategy is highly questionable. Of course, it is highly possible that not all network service providers have implemented this enhancement. Perhaps, the replacement of the underlying SAKA protocol with the AKA protocol (3GPP) will address these vulnerabilities in the near future.

In the meantime, the need for countermeasures has prompted service providers to implement various ID mechanisms, including the use of mobility and usage profiles.

Increased use of ABID

Aside from the common use of filtering, e.g. MAC-based filtering for detecting RAPs, what is also evident is the increased use of ABID for detecting both forms of device impersonation. In particular, the implementation of mobility-based user profiles in WLANs and WWANs could very well represent a trend, which is starting to emerge.

4.2 Attack Risk Analysis

As aforementioned, one would typically assess the significance of a given attack, based on various factors including the level of publicity and the number of research papers dedicated to its resolution. While this approach continues to be the norm, a more objective form of analysis, e.g. Quantitative Risk Analysis, would prove useful for not only establishing a quantitative level of significance, but also determining whether or not countermeasures are warranted.

It is a common practice to evaluate security-based and other types of threats using TRA. However, given that the single most significant threat, in the area of access control, is impersonation of legitimate users/devices, we apply the principles of TRA to analyze the attacks, associated with device impersonation. In particular, the following ARA is carried out using the risk measurement criteria identified in [49].

Scale	Occurrence	Technical difficulty	Motivation
1	unlikely	high	low
2	possible	moderate	moderate
3	likely	low	high

Table 4.2: Criteria for ARA: occurrence likelihood

Scale	Impact	Damage
1	low	low
2	medium	moderate
3	high	high

Table 4.3: Criteria for ARA: impact

In order to establish a relative degree of significance, the attacks are categorized according to the *occurrence likelihood* and their *impact*. The product of these two criteria provides a measurement of the *risk*. It is this value that dictates whether or not a mitigation strategy or a contingency plan is warranted.

Table 4.2 identifies the parameters used for the calculation of the occurrence likelihood or probability. The occurrence likelihood is defined by the difficulty of mounting an attack, e.g. cost, and motivation of an attacker, i.e. benefits to be realized. More specifically, a given level of technical difficulty is associated with the following factors: availability of equipment/tools, knowledge or skill set of the attacker, and appropriate operating environment.

A potential attack, regardless of the high occurrence likelihood, may not solicit much attention unless it negatively impacts the service providers and/or users. Three levels of impact, along with the corresponding damage that can be sustained, are presented in Table 4.3. Whereas a low-level impact can be characterized by various types of damage including user frustration, high-level impact may threaten a business and incur significant costs. Ideally, two sets of values should be used to assess the impact on both service providers and users.

The risk, associated with a given attack, is the key indicator that dictates, amongst other things, the level of effort that should be expended to protect the resources of the networks and/or users. A resolution strategy, see Table 4.4, is typically not required when the risk is minor. However, when it is classified as major or critical, it is in

Scale	Risk	Resolution strategy
1,2,3	minor	countermeasure not required
4	major	mitigation or countermeasure
6,9	critical	mitigation or countermeasure
		Values 5,7, and 8 are not possible

Table 4.4: Criteria for ARA: risk

ID	Intrusion Detection	Likelihood	Impact	Risk	Strategy
BT-A04	BT_ADDR Spoofing	2	2	4	C
802.11-A03	MAC address spoofing	3	2	6	M and C
802.11-A04/A05	RAP	3	3	9	
CDMA-A01	Cloning (CDMA2000)	2	3	6	M and C
	C=countermeasure M=mitigation				

Table 4.5: ARA Summary: Problems to be resolved

an organization's best interest to establish/implement a mitigation strategy, i.e. to reduce the occurrence likelihood, and/or a countermeasure.

Using this set of criteria, we carry out an ARA for the purpose of categorizing the attacks based on risk. Although the resulting values are subjective, what is important is the relative risk factor. A summary of the ARA is presented in Table 4.5 and discussed in the sequel. Given that the use of RAPs, for unauthorized access (802.11-A04) and AP spoofing (802.11-A05), serves similar purposes, they will be analyzed as a single entity.

Lowest Risk

At the lowest end of the risk spectrum is BT-A04. It has an overall risk assessment of 4. Consequently, a countermeasure should be developed in order to minimize the impact of this attack. Both the level of motivation, i.e. the value of the information to be obtained, and technical difficulty, which must be surmounted in order to mount a successful attack, are moderate. Furthermore, the fact that an attacker's BT device must be within close proximity to that of the victim may cease to be a hinderance, if a BT scanning device, e.g. BlueSniper, is used. Under these circumstances, both the likelihood and overall risk would increase, thus necessitating a more immediate form

of intervention. Finally, a moderate level of impact reflects the level of damage, e.g. disruption to the piconet, that can be sustained by a BT network.

Moderate Risk

Within the category of moderate risk (value of 6) are MAC address spoofing and CDMA cloning.

One of the widely recognized problems in 802.11 networks is MAC address spoofing (802.11-A03). Unlike BT address spoofing (BT-A04), MAC address spoofing can be carried out, with relative ease, using any of the readily available tools and at a distance that is considerably further from the victim's device. Moreover, potential access to the resources of the core network provides sufficient incentive for pursuing this attack. Depending on the authorization mechanism in place, e.g. device-based, an attacker could obtain access to various types of resources, including highly sensitive information. In addition, he could initiate other attacks at the link layer, such as the de-association or de-authorization attack. Even worse, he could cause service disruptions for a considerable period of time.

In contrast to GSM cloning, the likelihood, associated with CDMA cloning (CDMA-A01), is moderate. Although software and instructions for cloning these devices are available on the Internet, some technical expertise is required in order to successfully perform the required operations. However, as with GSM, the revenue to be generated through the sale of cloned phones (many can be cloned within a short period of time) and the availability of free phone services, are often more than sufficient to provide the required level of motivation. As aforementioned, recent incidents, e.g. [175], [174] and [177] of CDMA cloning in India, attest to this conclusion.

Unfortunately, the potential level of damage, sustained by service providers and users, can be significant. Whereas increased frustration, breach of privacy, and financial theft (if e-commerce or m-commerce is supported) represent some of the resulting costs, borne by an individual user, the consequences are much more acute for service providers. They are ultimately responsible for the illegitimate calls and the resulting loss in revenues.

A strategy, which may prove beneficial in mitigating the risk of cloning, is the incorporation of biometrics into the phone. Depending on the accuracy of this technique, this approach may successfully prevent unauthorized access to the phone, if it is not already operational. Of course, one or more countermeasures that are implemented as ABID mechanisms, would also provide another level of defense.

Highest Risk

The attack that carries the highest level of risk is the use of RAPs, as correctly portrayed by the media, research teams and vendors alike.

Using RAPs, a malicious attacker could inflict an unprecedented level of damage including long disruptions to services. Furthermore, there are no technical difficulties to overcome. In brief, an attacker captures the identity of a legitimate AP, using any one of the readily available tools. It subsequently generates frames, using the stolen identity, and injects the crafted messages when the medium becomes available. Hence, RAP-based attacks are both possible and likely to occur.

When EAP mutual authentication is used, the likelihood of the attack is mitigated to some degree. In any event, given the security vulnerabilities of EAP and the reluctance of service providers to implement it, the risk associated with a RAP attack is critical.

4.3 Problem addressed: Detection of address spoofing

As previously stated, the use of a *single malleable* identifer in ACLs, e.g. MAC address, renders this access control mechanism susceptible to address spoofing. In order to address this problem, resolution strategies and countermeasures have been proposed, see Tables 2.2 and 3.2.

In terms of resolution strategies, the use of public-key cryptography, although theoretically feasible, has some disadvantages including the use of quasi-static data, administrative overhead and increased computational requirements.

As the public/private key pair represents static data (unless it is changed periodically and that is unlikely), it can potentially be discovered using OTA and other mechanisms. Perhaps, when tamper-resistant hardware become more affordable [167], this limitation may cease to exist. Another disadvantage, identified by Laing [103], is the time required to manually type each MAC address and its associated public key into each AP. Unless the cost of administration is reduced via automation, this solution may not be suitable but for smaller networks. Finally, the resources required for public key cryptography are currently unavailable in hand held devices. As confirmed by Barbeau *et al.* [21], even the use of elliptic key cryptography demands a level of resources that exceeds current availability.

Given these limitations and requirements, organizations may opt to address this problem using countermeasures, including intruder location by Adelstein *et al.* [9], commercial IDSs, e.g. AirDefense [89], and UMPs by Spencer [163]. Unlike the use of public-key cryptography, the use of intruder location or user mobility profiles, is less susceptible to forgery and impersonation attacks. For one thing, as an ABID mechanism, both countermeasures make use of behavioral data, which are more difficult to forge or replicate. Whereas the intruder location mechanism (802.11-A03-C01) examines the signal strength (device characteristic) of WiFi/802.11 wireless nodes, the use of UMPs (user characteristic) is adopted in (802.11-A03-C02). Second, both strategies require that an association, between a given MAC address and its corresponding profile, be maintained for the purpose of detecting MAC address spoofing.

As far as commercial products are concerned, AirDefense does prevent MAC address spoofing by looking at the address prefix. However, this approach is limited in that the IDS makes a distinction between devices based only on the manufacturer's identification. Hence, the need to identify devices, from the *same* manufacturer, remains unfulfilled.

In light of these circumstances, there is an opportunity to further explore the use of device-based and user-based profiles for addressing the aforementioned problem.

Part IV

Approach: ABID using device-based profiles

Chapter 5

Radio Frequency Fingerprinting

While RF has provided the fundamental mode of communications in wireless networks, its use has also been exploited in other applications.

As aforementioned, RF signatures have been used by for locating wireless devices, as indicated by Newbury Networks [124].

Another application, which is being touted as the next big thing on the technology radar, is Radio Frequency Identification (RFID) [82] and [168]. One of the early adopters of this technology, Walmart, has been aggressively pursuing its implementation in 600 of its stores. By January 2006, the next 200 suppliers are also expected to go live.

RFID is an identification method that is based on the use of two key components [185]: an RFID tag or transponder and reader (transceiver). In a typical RFID system, objects are equipped with a small, inexpensive tag. The tag contains a transponder with a digital memory chip that has been pre-programmed with a unique electronic product code. On the other hand, an RFID reader, also referred to as an interrogator, i.e. an antenna packaged with a transceiver and decoder, emits a signal that activates the RFID tag. Hence, when an RFID tag is within the RF zone, it detects the activation signal, generated by the reader, and transmits its data. The reader, in turn, decodes the data in the tag's silicon chip and forwards them to the host computer. Finally, the application on the host processes the data, typically using Physical Markup Language (PML).

Current applications include animal identification, tracking of pallets, building and vehicle access control, electronic cash using smart cards and human implants. In addition, other interesting applications of RFID are forthcoming. They include the replacement of Universal Product Code (UPC), health care, patient identification, and intelligent traffic signs or road beacons for vehicular positioning purposes.

What is of interest to us, is the use of RFF for the purpose of device identification. RFF is a technology designed to capture the unique characteristics of the radio frequency energy of the transceiver, for the purpose of identifying cell phones and other devices. Nevertheless, the underlying principle applies equally to all wireless devices.

Pioneered by the military to track the movement of enemy troops, it has been subsequently implemented by some cellular carriers, e.g. Bell Nynex, as an authentication mechanism, to combat cloning fraud [147]. The key benefit of employing this technique is the increased level of difficulty, associated with the replication of a transceiverprint, i.e. set of features extracted from the transient of a signal. As illustrated in Figure 5.1, the transient of a signal is associated with the start-up period of a transceiver prior to transmission. Even more importantly, it reflects the unique hardware characteristics of a transceiver and other related components. These characteristics are produced as a result of the manufacturing process and the tolerance limits of the underlying components. Consequently, it cannot be easily forged, unless the entire circuitry of a transceiver can be accurately replicated (e.g. requiring the theft of an authorized device). It is precisely this feature that is being exploited for the purpose of identifying RF-based transceivers. However, in the case of theft, it is assumed that other mechanisms, such as a black list and/or user-based profiles, are used to address this issue.

5.1 Related Work

Since 1995, the level of interest in RFF continues to rise, partly motivated by the need to identify malfunctioning or illegally operated radio transmitters, in support of radio spectrum management practices. This section provides a brief overview of the

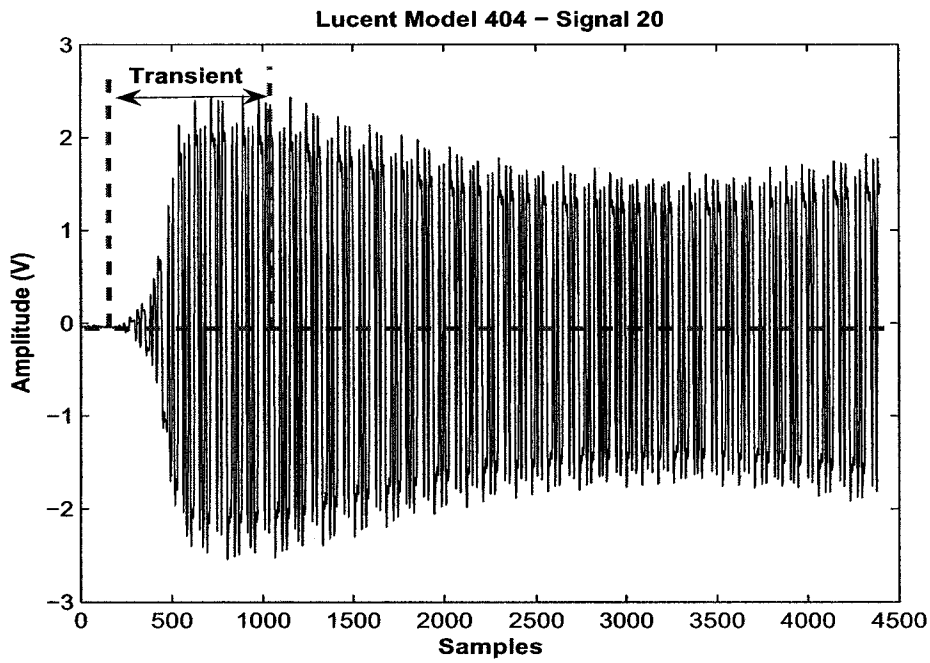

Figure 5.1: Signal from a 802.11b Transceiver

various research initiatives that have been undertaken in the area of RFF. Although a detailed description of the process is presented in the sequel, a condensed version is provided here in order to categorize the various research efforts.

The RFF process begins with the extraction of the turn-on transient of a signal. Fig. 5.1 illustrates the location of the transient from a Lucent 802.11b wireless device (ID:404) that has been manufactured using the Orinoco chip set. Once the transient has been isolated, various components, namely instantaneous amplitude, phase and frequency are subsequently obtained. Next, one or more features, from each of these components, are extracted. This set of features represents a fingerprint of the transceiver, or in other words, a transceiverprint. The transceiverprint is, in turn, classified as belonging to one of the profiled transceivers.

Radio transmitter fingerprints

In the paper by Ellis and Serinken [45], the authors examine the amplitude and phase components of signals, captured from various transceivers (some from the same manufacturer). The overall objective is to determine the degree of variability, asso-

ciated with amplitude and phase profiles of all transceivers. The general conclusion is that all transceivers do possess features (derived from amplitude and phase components) that are consistent (within each transceiver), although they may not be necessarily unique (between transceivers). Moreover, the characteristics of the fingerprints are likely to change as a result of Doppler shift, multipath propagation, fading, temperature variation, battery condition and aging.

Detection of start of transients

Proposed by Shaw and Kinsner in 1997, the Threshold detection approach [157] makes use of data from the discriminator output of a general coverage communications receiver. In brief, one of the key functionalities of a discriminator system is to convert changes in frequency into amplitude. The key objective is to calculate the variance in amplitude for each consecutive portion/window of the signal and to compare each of these values, in sequence, to a predetermined threshold. The start of a transient is located when the variance exceeds the threshold by a given margin. The end of a transient is determined in an experimental manner. The most significant drawback of this approach is the level of effort required to establish a system-wide threshold. Moreover, the underlying algorithm does not address spikes within the ambient channel noise. Finally, the authors have imposed a minor limitation, i.e. the ambient channel noise segment should contain a minimum of one quarter of the samples of a raw signal.

Another approach, which also makes use of the discriminator output, is the Bayesian Step Change Detector (BSCD). Proposed by Ureten and Serinken [180], the underlying technique transforms a change in the variance into a change in the mean value, which is subsequently used by the BSCD to detect the start of a transient. Unlike the previous approach, the detection of a transient is based exclusively on the characteristics of the data. Consequently, this technique can theoretically be used with various types of signals. However, the performance is less than optimal for signals, e.g. 802.11 and BT, that exhibit a gradual change in power at the start of a transient. In order to accommodate these signals, the authors have recently proposed

an enhanced detection method, referred to as the Bayesian Ramp Change Detector (BRCD) [153].

Feature selection

As far as the selection of features is concerned, the use of Neural Networks (NN) represents one option, although they are typically used for classification purposes.

A NN represents a model for classifying a given sequence of data into one of many classes. Typically, they are trained using samples (transceiverprints) from a given class (e.g. transceivers). The classification process is carried out by calculating the distance between a new transceiverprint and those in the training database.

The use of Probabilistic Neural Network (PNN) [40] for feature selection is explored by Hunter in [87]. Each feature that has been extracted, either participates in the definition of the transceiverprint or not. This condition is represented by using binary values (0-no participation, 1-participation). Thus, the transceiverprint consists of a binary string, which indicates the participation status of the features, at a given point in time. By classifying the transceiverprint and observing the results, the composition of the fingerprints can be adjusted accordingly.

Classification of transceiverprints

As the classification process of RFF is often based on the concept of pattern recognition, there are many options available.

The use of a pattern-based classifier, such as the PNN, is advocated by many research teams including Shaw [157], Hunter [87] and Tekbas *et al.* [173].

An alternative to NNs is the use of the Self Organizing Map (SOM) [50]. In the paper by Somervuo *et al.* [162], the authors make use of the SOM and a learning vector quantization (LVQ) algorithm to support variable-length feature sequences. The novelty of their work is the association of a feature vector *sequence* with each SOM node, unlike the traditional method of using single feature vectors. The LVQ is used to optimize the sequences for optimal class separation. This technique may

prove useful for unsupervised clustering of input (sequence) whose temporal structure is of interest.

An interesting and unique approach is adopted by Hippenstiel [83]. The frequency component (coefficients of the Discrete Wavelet Transform (DWT) [114]) of the transient is first obtained using the Daubechies Polynomial of order 8, a wavelet filter typically used in DWT. Since the maxima of the modulus of the wavelet coefficients contains approximately the same amount of information as the transient, the wavelet coefficients at each scale are replaced by their extrema. Using this reduced set of coefficients (transceiverprint), classification is carried out by determining the Euclidean Distance (ED) between a given transceiverprint and the templates in each of the different classes (transceivers).

Finally, the use of genetic algorithms for classification purposes is explored by Toonstra and Kinsner [178]. Mimicking the natural evolution of biological species, genetic algorithms have been used to obtain a near optimal solution to a problem. The algorithm's three main components: crossover, selection of the fittest and mutation are applied recursively until the stop criterion is met. This concept is used to not only reduce the number of wavelet coefficients in the transceiverprint but to determine the best match between the transceiverprint and the class of transceivers. Aside from obtaining an optimal solution, this approach is rather resource-intensive. Hence, the use of genetic algorithms may not be appropriate for resource-constrained devices.

5.2 Description of Solution

In order to address the threat of device impersonation by rogue devices, we have focused our efforts on the development of a countermeasure for MAC address spoofing. However, as aforementioned, a generic solution should fulfill similar requirements in other RF-based networks, namely BT [118] and cellular networks.

Whereas a brief overview of the key components is presented in this section, evaluation exercises, carried out using WiFi/802.11 and BT devices, are discussed in subsequent chapters.

Profiling As stated previously, a profile of a transceiver (created using RFF) is used for ABID. By associating a MAC address of a wireless device (e.g. 802.11 wireless card, RAP) with its corresponding transceiver profile, the capabilities of a wireless IDS can be further enhanced. This speculation is based on the premise that the transceiverprints, of the illegitimate device, are sufficiently different, and hence, would not match the profile of the legitimate device.

Classification In terms of classification, a MSPC technique, namely Hotelling's T^2, and a threshold that has been established using the F distribution, are used to classify a transceiverprint. That is, the classification process determines if a transceiverprint matches the profile of the transceiver with the claimed MAC address. Although the T^2 distribution itself can be used, the F distribution takes not only the number of features/variabiles and training samples into consideration, but also accommodates the use of specific confidence coefficients.

Decision Engine It is generally known that current IDSs render a decision, as to whether an observed behavior is normal or anomalous, based on a *single* observation. In an environment that is characterized by interference and noise, delaying the decision until *multiple* observations have been processed reduces the level of uncertainty and results in a lower false-alarm rate (i.e. normal behavior classified as anomalous).

Concept Drift Lastly, the notion of concept drift is addressed by continuously updating the profile of a transceiver using currently observed transceiverprints. The application of this strategy not only results in a low FAR but also increases the detection (i.e. anomalous behavior is correctly identified) rate.

Before proceeding to the following section, readers are encouraged to consult Appendix A. It provides a brief introduction to the representation of signals and the extraction of its components (e.g. amplitude) for RFF.

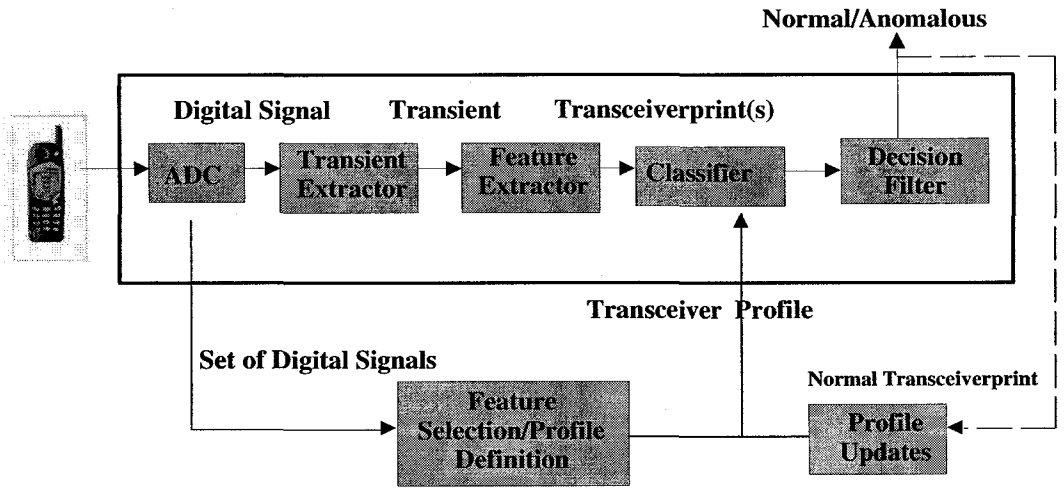

Figure 5.2: ABID using RFF

5.2.1 Intrusion Detection Framework

This section describes the framework and key activities that are undertaken to fulfill the two primary objectives: the creation of a profile for each transceiver and the specification of a classification system. The information flow within the anomaly-based IDS is illustrated in Fig. 5.2.

The flow of information, associated with the detection phase, begins with the conversion of an analog signal to a digital signal using the Analog to Digital Conversion (ADC), which will not be covered in detail. As previously indicated, the transient of a digital signal is extracted, by the *transient extractor*. The extractor exploits the phase characteristics of a signal, in order to detect the start of the transient, as illustrated by Hall, Barbeau and Kranakis [71]. Upon isolating the transient, its amplitude, phase and frequency are subsequently extracted by the *feature extractor*. In turn, these components are used for the extraction of specific features, which define a transceiverprint.

The *classifier* is then used to carry out the identification or recognition of Wi-Fi/802.11 transceivers. The identification of a transceiver requires that the classifier

match an observed transceiverprint to one of the transceiver profiles in the IDS, as illustrated by Hall, Barbeau and Kranakis [72]. On the other hand, the recognition or verification of a transceiver is carried out, by comparing an observed transceiverprint to the profile of the transceiver, which is associated with a given MAC address. Finally, a decision is rendered, regarding the status (normal/anomalous) of a *set* of transceiverprints, by the *decision filter*.

As far as the transceiver profiles are concerned, they are created during the profiling phase, by extracting the transceiverprints from a set of captured signals and storing the corresponding centroid and covariance matrix (discussed in section 7.2.1) in a profile. This exercise is undertaken prior to the evaluation and classification phases. Furthermore, due to factors, such as transceiver aging, there is a need to periodically update a profile in order to reflect the altered characteristics of a transceiver. One possible strategy is to continuously recalculate the centroid and covariance matrix (two key elements used for classification purposes) using one or more transceiverprints, which have been classified recently as normal, and the Moving Average Filter (MAF) [108].

Chapter 6

Profiling Phase

6.1 Transient Extractor

As the unique characteristics of a transceiver are primarily manifested in the transient, a key task is to extract it from a digital signal. This proves to be one of the most challenging yet crucial aspects of the RFF process. Inaccurate detection of the start of a transient results in misclassification. Although both premature and delayed detection ultimately influence the classification success rate (correct classifications divided by the total number of classifications), the impact is more pronounced in the case of a delayed detection. In order to ensure a high probability of success, during the classification phase, it is essential that the difference between the estimated detection value and the actual value be minimized.

6.1.1 Related Work

In this section, an analysis of the three key approaches, namely Threshold [157], BSCD [180] and BRCD [153] is presented. They not only exploit the characteristics of the raw signal, but are also based on the premise that these characteristics, associated with the channel noise and transient, differ. Although, this holds true for all signals, the performance of the underlying algorithms is less than optimal, when applied to signals from 802.11b and BT devices. With these signals, the transition

between channel noise and transient occurs more gradually.

Threshold Detection

Proposed by D. Shaw and W. Kinsner in 1997, the Threshold Detection is one of the most recent approaches for detecting the start of a transient.

Phase 1: Extract Features from Signal

It is well known that the Euclidean dimensions of a point, line and plane can be represented by integer values of 0, 1 and 2. However, fractional quantities can also be used to accommodate such objects as signals. Whereas fractals refer to objects that are similar to each other, fractal dimension can be used to represent the *irregularity* of a fractal, as described by Kinsner [99].

With this approach, it is the variance in amplitude that is used to calculate the fractal/variance dimension for successive portions (defined using an overlapping window) of a signal:

$$D(t) = E + 1 - H \tag{6.1}$$

where E represents the Euclidean dimension and has been assigned a value of one for this application. This forces the value of the variance dimension $D(t)$ to fall between 1 (highly correlated portions of the signal) and 2 (uncorrelated white noise). It also implies that the value of H, referred to as the Hurst exponent, will be within the range of [0,1].

The relationship between the variance in amplitude, in a given section, and H is as follows:

$$Var[\Delta X_{\Delta t}] \approx |\Delta t|^{2H} \tag{6.2}$$

where Δt is the time increment and $\Delta X_{\Delta t}$ represents the difference between two samples that are separated by the value of the time increment. What is important is that the variance of $\Delta X_{\Delta t}$ increases in proportion to $|\Delta t|^{2H}$, for large values of t.

The value of H is calculated according to:

$$H = \lim_{\Delta t \to 0} \frac{1}{2} \frac{\log[Var(\Delta X_{\Delta t})]}{\log(\Delta t)} \tag{6.3}$$

Moreover, for a given portion of a signal, H indicates the correlation of $\Delta X_{\Delta t}$ with respect to $|\Delta t|$. Therefore, given that the portions of a signal, within channel noise, are characterized by high amplitude variations, the corresponding values of H would be low, i.e. less than 0.5. On the other hand, H would typically be higher for subsequent portions of a signal. Finally, as $D(t)$s, associated with channel noise and transient, differ to some extent, the start of a transient should be located at the transition point.

Phase 2: Detect start of Transient

Once the variance dimensions have been determined and stored in a feature vector (referred to as the fractal trajectory), the start of a transient is detected according to

$$|D(t) - \mu| > (\tau \times \mu) + \sigma \tag{6.4}$$

where τ is the threshold that has been established experimentally. In addition, μ and σ represent the mean and standard deviation of a segment of channel noise ($t = 1, 2, \ldots, \frac{T}{4}$) in the original signal, see Fig. 6.1. Finally, T denotes the total number of samples.

This algorithm is used to calculate the difference between each variance dimension (element in the fractal trajectory, e.g. m=$\frac{T}{4}+1, \frac{T}{4}+2, \ldots, T$) and the mean value until the condition in Eq. 6.4 is met (e.g. m=3500). Given that the mean value has been calculated, based on a representative segment of channel noise, the absolute difference between the variance dimension and the mean would not satisfy Eq. 6.4, if a variance dimension is associated with channel noise. However, at the start of a transient, where the variance dimension is expected to be significantly lower or higher, the absolute difference would be greater than the sum of the standard deviation and mean (value dictated by threshold). Once detection has been triggered, the corresponding location,

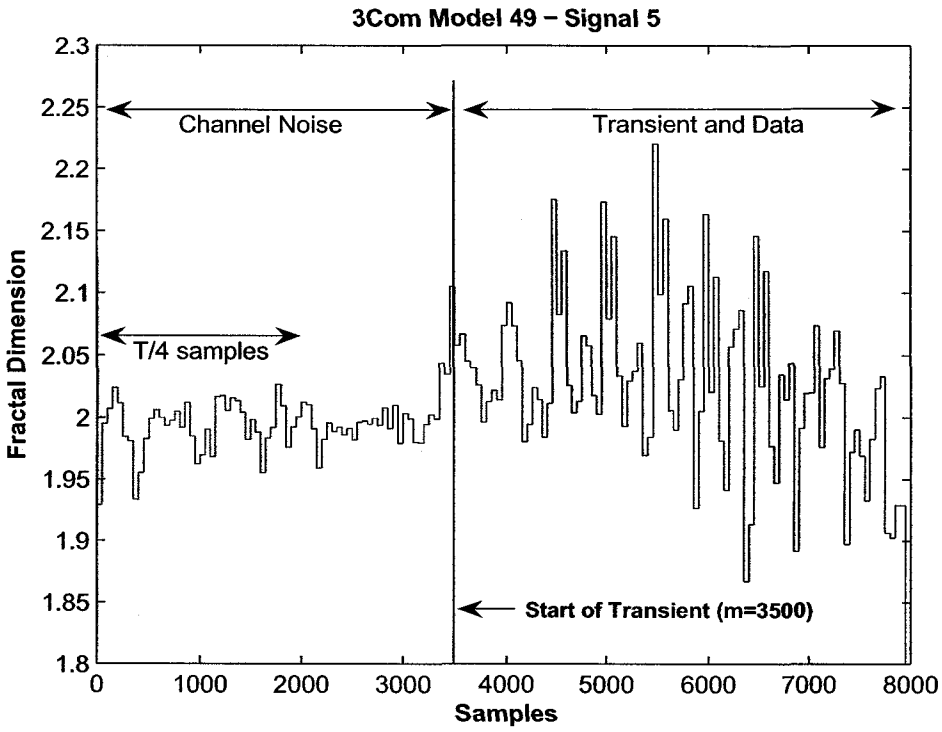

Figure 6.1: Transient Detection using Threshold (BT transceiver)

within the original signal, can be determined.

A variation of this approach, which makes use of complex signals for detecting the start of a transient, is proposed by Tekbas and Serinken [172].

Test Case: Threshold Detection

Fig. 6.2 demonstrates the performance of the Threshold algorithm, when applied to a BT signal from 3Com (Model or ID:49). Based on plot 2, the start of the transient is expected to coincide with sample number 3500 in plot 1. However, since this value is greater than the actual value of 3300 (delayed detection), this test would not be considered a success.

Due to the difficulty in establishing a system-wide threshold for all transceivers, an overall success rate could not be established. Nevertheless, it might be worthwhile to explore the feasibility of establishing transceiver-based thresholds, in a dynamic manner.

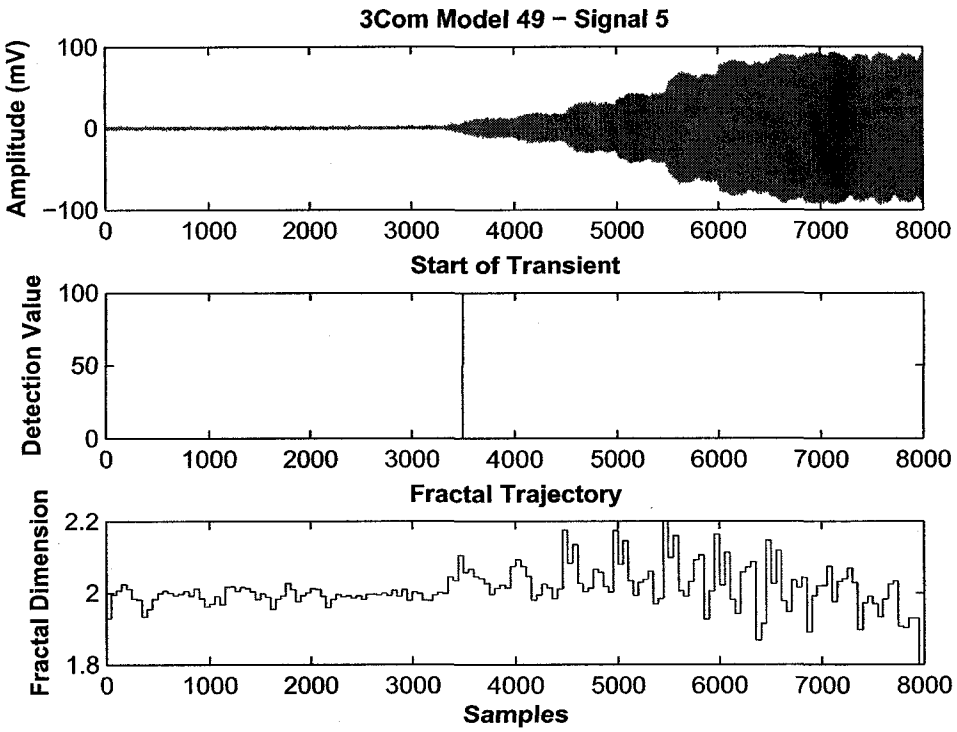

Figure 6.2: Test Case for Threshold Detection (BT transceiver)

Bayesian Step Change Detector

As with the Threshold approach, the BSCD, proposed by O. Ureten in 1999, also makes use of amplitude, associated with changes in frequency, to detect the start of a transient.

Phase 1: Extract Features from Signal

Unlike the Threshold process, the fractal dimension is calculated for successive segments of a signal using Higuchi's method, as demonstrated in [81]. First, subsets of samples e.g. $X(1), X(2), \ldots, X(N)$ of the original signal are created according to :

$$X(m,k); X(m), X(m+k), ..., X\left(m + \left[\frac{N-m}{k}\right] \times k\right)$$

where m and k are integers, which denote the initial time (sample) and interval time (number of samples), respectively.

Thus, for example, setting $k = 3$, $N = 100$ and $m = 1, 2, 3$ will result in the following 3 subsets:

X(1,3); X(1), X(4), ..., X(100),
X(2,3); X(2), X(5), ..., X(98),
X(3,3); X(3), X(6), ..., X(99).

Second, the length of the curve for each of the subset $(X(m,k))$ is calculated using the following :

$$L_m(k) = \left\{ \left(\sum_{I=1}^{\frac{N-m}{k}} |x(m+ik) - x(m+(i-1)k)| \right) \frac{N-1}{\left[\frac{N-m}{k}\right] k} \right\} /k$$

The term, $\frac{N-1}{\left[\frac{N-m}{k}\right]k}$ represents the normalization factor of the curve length.

Finally, the average value $L(k)$, of the k sets of $L_m(k)$, is plotted against k on a log-log scale, followed by the application of the least-square procedure. The slope of the straight line, an output of this procedure, represents the fractal dimension. The resulting data are subsequently used to achieve the same goal as the Threshold approach.

Phase 2: Detect start of Transient

Unlike the previous approach, the detection of the start of a transient is accomplished using the posteriori probability density function of the BSCD.

$$p(\{m\}|d) \alpha \frac{\frac{1}{\sqrt{m(N-m)}}}{\left[\sum_{i=1}^{N} d_i^2 - \frac{1}{m}(\sum_{i=1}^{m} d_i)^2 - (\frac{1}{N-m})(\sum_{i=m+1}^{N} d_i)^2\right]^{\frac{N-2}{2}}} \tag{6.5}$$

The parameters of the function have the following definition: d represents the fractal dimension, N is the number of elements in the fractal trajectory and m repre-

sents a potential change point (start of transient). In addition, while the term $-\frac{N-2}{2}$ is used to accentuate the difference in variance, the numerator $\frac{1}{\sqrt{m(N-m)}}$ provides a weighting function that favors those elements, which are located near the middle of the fractal trajectory.

For each fractal dimension (at point m) in the fractal trajectory, this function is used to calculate the variance of the fractal dimensions for the sequence $[1, \ldots, m]$ and $[m+1, \ldots, N]$. The larger the difference in variance, between these two sequences, the larger the value of the probabilistic density function. Since the fractal dimensions are typically higher for the noise segment of a signal, the difference in variance between two sequences would be the highest at the start of a transient.

Fig. 6.3 illustrates the use of the fractal trajectory (overlapping factor set to one sample) for detecting the start of the transient. The vertical bar at m=4000 (expected start of the transient) is consistent with the results of the probability density function, which are displayed in Fig. 6.4, 2^{nd} plot.

Test Case: BSCD

As depicted in Fig. 6.4, 2^{nd} plot, the detection of the start of the transient has been delayed by 700 samples (from 3300 to 4000). The primary factor is the gradual change in amplitude during the transition from channel noise to the transient.

Bayesian Ramp Change Detector

In order to accommodate WiFi signals, which are characterized by a linear increase in output power level, Ureten and Serinken have proposed the BRCD.

Phase 1: Extract Features from Signal

Given that the following model assumes a linear increase in the power level of the transceiver, during the turn-on transient, the average power level of each section (25 samples), of the original signal, is used as signal data.

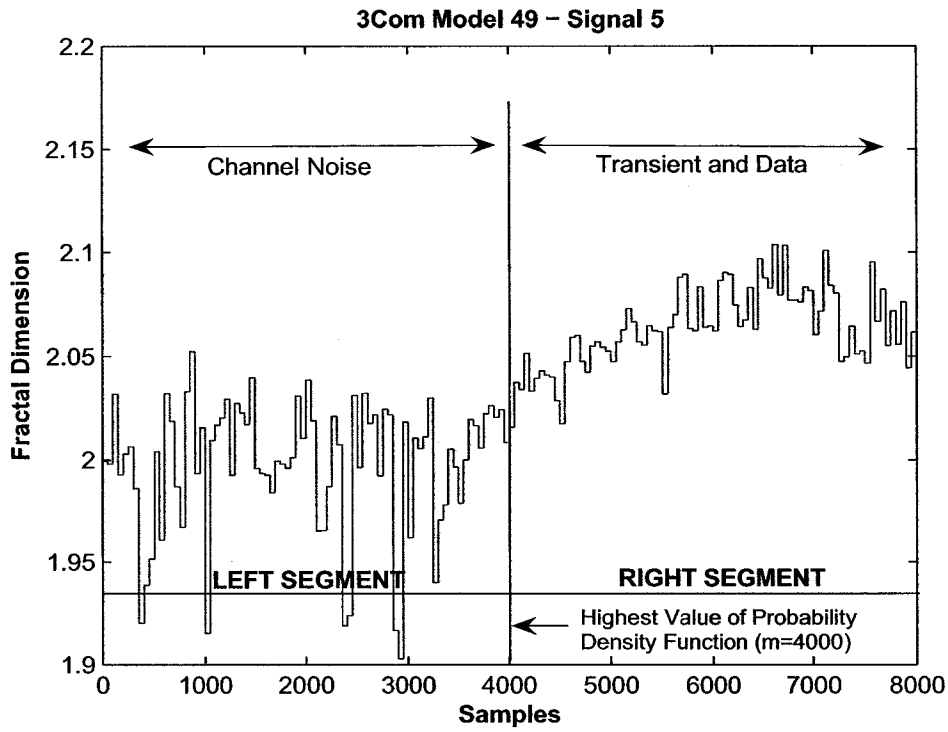

Figure 6.3: Transient Detection using BSCD (BT transceiver)

Phase 2: Detect start of Transient

As with BSCD, the detection of the start of a transient is accomplished using the a posteriori probability density function of the BRCD. However, the samples of the signal data can be modeled using a matrix equation:

$$d = Gb + e \tag{6.6}$$

where d represents an $N \times 1$ matrix of the data points, e is an $N \times 1$ matrix of Gaussian noise samples. In addition, G is an $N \times M$ matrix, where each column represents a basis function that is evaluated at each data point, and each element of the $M \times 1$ matrix b is a linear coefficient. Moreover, the change point m is reflected in the structure of G. For BRCD, it is defined as:

$$G^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & \dots & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 & 3 & \dots & N-m \end{bmatrix} \tag{6.7}$$

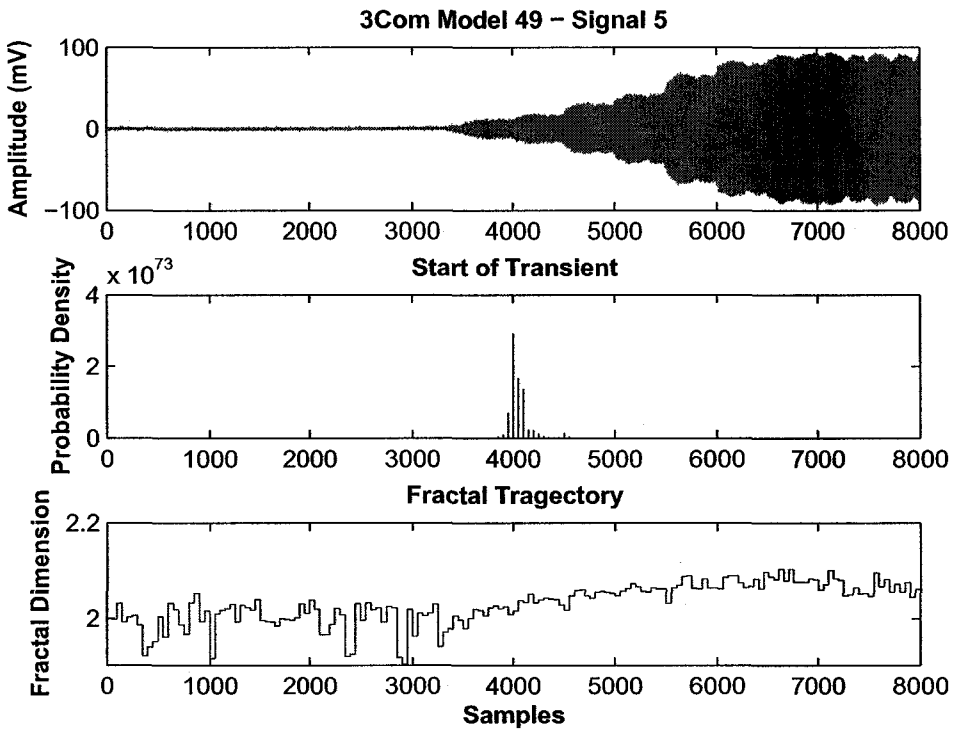

Figure 6.4: Test Case for BSCD (BT transceiver)

Finally, the a posteriori probability density of the change point is calculated based on the following:

$$p(\{m\}|d)\alpha\frac{[d^Td - d^TG(G^TG)^{-1}G^Td]^{-(N-M)/2}}{\sqrt{det(G^TG)}} \tag{6.8}$$

The change point with the highest a posteriori probability represents the expected start of a transient.

Test Case: BRCD

As depicted in Fig. 6.5, 2^{nd} plot, the detection of the start of the transient is premature. Two contributing factors are the length of the channel noise, in comparison to the subsequent segment of the signal, and the quasi-linear characteristics of the power-on ramp. It should be noted that the amplitude in the 1^{st} plot has been

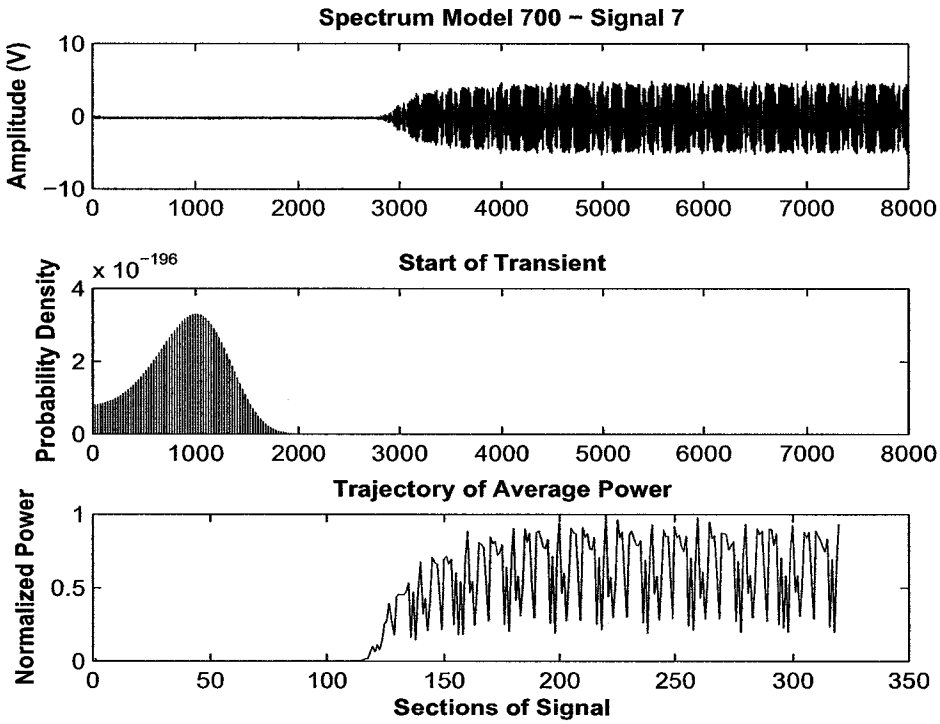

Figure 6.5: Test Case for BRCD (802.11b transceiver)

multiplied by 200, in order to derive the probability distribution. Thus, the actual range of the y-axis is between -0.05 and 0.05.

6.1.2 New Approach

After having implemented and analyzed both approaches, the need to enhance the transient detection process became evident. Thus, a new algorithm, which accommodates signals with varying transitional characteristics, was introduced by Hall, Barbeau and Kranakis [71].

Transient Detection using Phase Characteristics (TDPC)

The TDPC exploits the phase characteristics (features) of signals for detecting the start of transients, in a more efficient (lowest running complexity) and effective (higher success rate) manner.

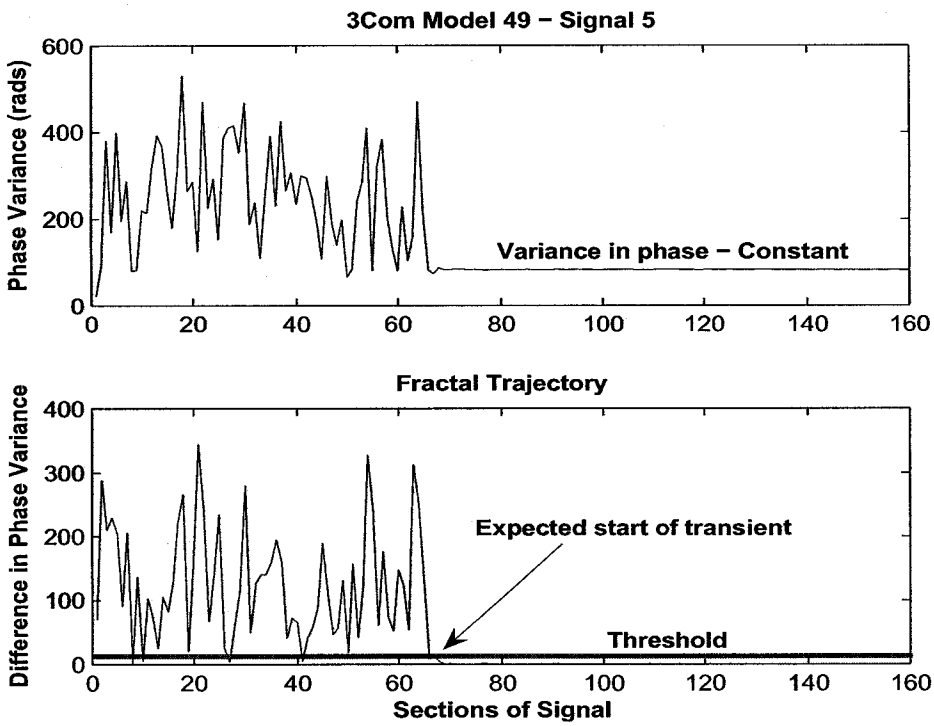

Figure 6.6: Detecting the start of the transient (BT transceiver)

Phase 1: Extract Features from Signal

Using the phase characteristics of a signal has a number of advantages. First, as the phase of a signal is less susceptible to noise and interference, it does not exhibit the same degree of fluctuations. Second, there is a significant difference in phase variance, associated with channel noise and the transient, see Fig. 6.6 1^{st} plot. This property should simplify the task of establishing a threshold by visual inspection and thus permit us to leverage the principle of threshold detection. Finally, the use of phase variance should also render the detection algorithm more accurate, especially, when it is applied to signals with a less abrupt change at the start of the transient, see Fig. 6.8, 1^{st} plot.

The variance fractal trajectory is created as follows:

The instantaneous phase of an observed signal $O(t)$ is unwrapped using Eq. A.5. The resulting vector $\overrightarrow{AV}$ has the same length N as that of $O(t)$. The absolute value

of each element in $\overrightarrow{AV} = (AV_1, AV_2, \ldots, AV_N)$ is then obtained. In order to magnify the variation between the noise and transient segment in $\overrightarrow{AV}$, an overlapping window of size w, with the sliding factor of s, is used to extract the variance for successive segments of $\overrightarrow{AV}$. Both w and s are selected based on the accuracy with which the start of a transient is to be determined. The resulting values (variances) are then stored in a temporary vector $\overrightarrow{TV}$ of length $\frac{N-w}{s} + 1$.

$$\overrightarrow{TV}(i) = \sum_d^g (AV_d - \frac{1}{g-d} \sum_d^g AV)^2$$

where $i = 1, 2, \ldots, N/s$, $g = i \times s$, $d = g - s + 1$ and $\overrightarrow{TV}(i)$ represents the variance for a given section of the signal starting at value d and finishing at value g. An indirect benefit of using a smaller vector is the reduction in processing time, associated with the detection phase. Finally, the difference between any two subsequent values in the $\overrightarrow{TV}$ is obtained in order to create the variance fractal trajectory $\overrightarrow{VT}$.

$$\overrightarrow{VT}(i) = |TV_i - TV_{i+1}|$$

for $i = 1, 2, \ldots, \frac{N-w}{s}$, see Fig. 6.6, 2^{nd} plot.

Phase 2: Detect start of Transient

The successful detection of the start of the transient is based on the fact that the slope, associated with the difference in phase variance, has different properties before and after the start of the transient. In the case of BT technology, which makes use of Gaussian Frequency-Shift Keying, the slope becomes and remains constant after the start of the transient. However, with 802.11 signals, the slope remains relatively constant until the start of the transient, see Fig. 6.7. This phenomenon is to be expected given that 802.11b transceivers employ Quadrature Phase Shift Keying for modulation. Irrespective of the modulation type, an abrupt shift in phase variance is observed after channel noise. It is this property that is being exploited to determine the start of the transient. As a side note, this information can also be used for identifying the modulation type (e.g. frequency or phase) of a signal.

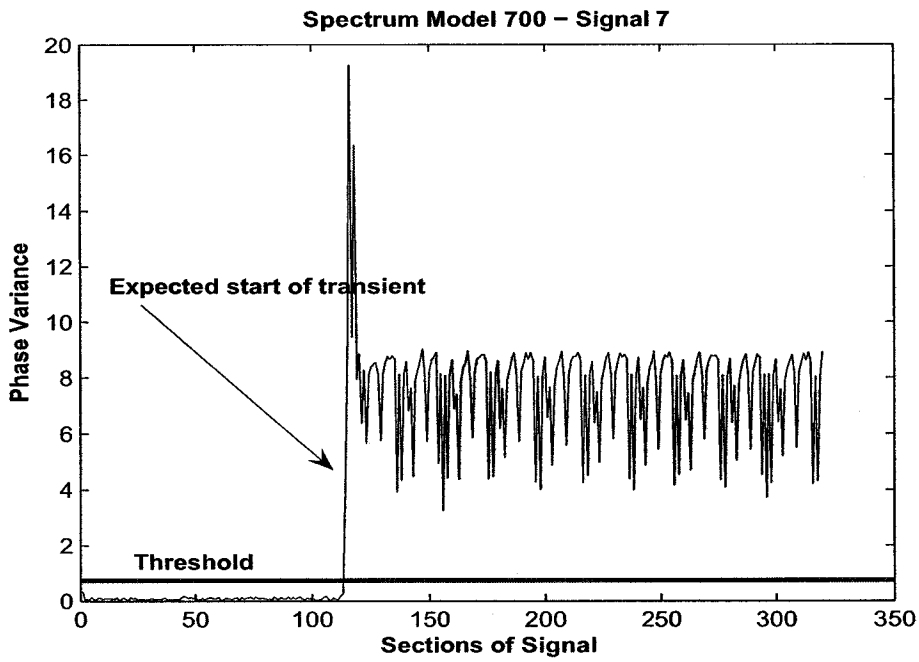

Figure 6.7: Detecting the start of the transient (802.11b transceiver)

The detecting process is carried out by comparing each of the four consecutive elements in the $\overrightarrow{VT}$ to a threshold τ (determined in an experimental manner) until the following condition is met.

BT Transceiver

$$VT(i), VT(i+1), \ldots, VT(i+3) \leq \tau$$

802.11b Transceiver

$$VT(i), VT(i+1), \ldots, VT(i+3) \geq \tau$$

for $i = 1, 2, \ldots, \frac{N-w}{s} - 2$.

At this point, the estimated start of the transient is typically located within 50-150 samples prior to the actual starting point of the transient. Fig. 6.7 illustrates the application of this technique to a 802.11b signal from transceiver (ID:700) by Spectrum.

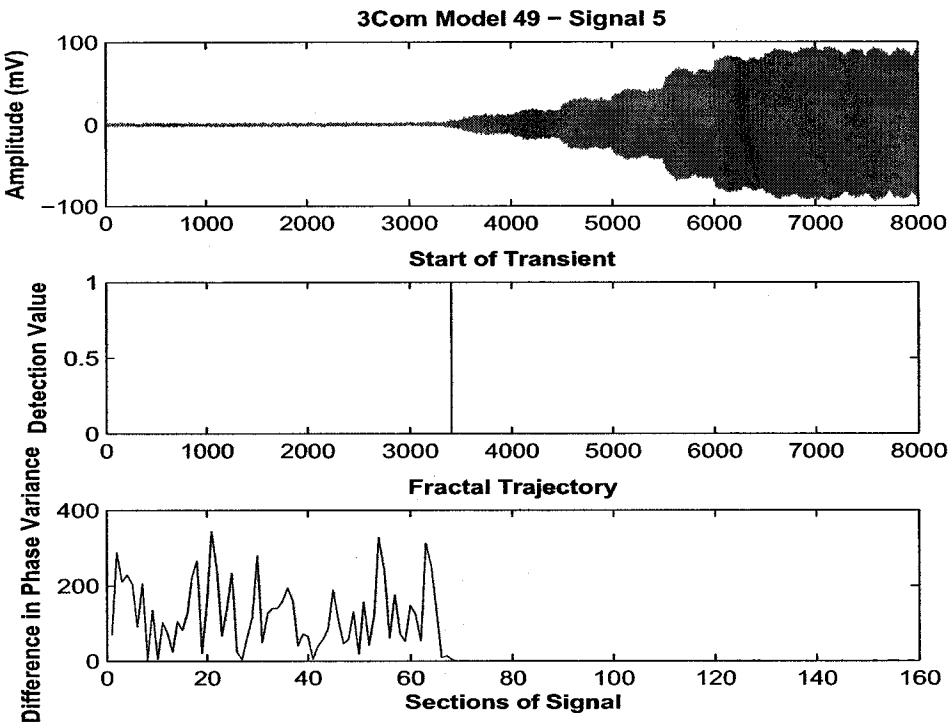

Figure 6.8: Test Case for TDPC (BT transceiver)

Test Case: TDPC

Figure 6.8, 2^{nd} plot, illustrates the application of the new detection algorithm to the same BT signal used in the previous approaches. The performance of the algorithm is better given that the estimated start of the transient occurs prior to the actual value and that the difference between the two values is approximately 50.

On the other hand, Fig. 6.9 illustrates the application of this algorithm to a signal from an 802.11b transceiver (Spectrum Model 700). As indicated by the detection value, TDPC is equally applicable to signals from WiFi transceivers.

Figure 6.10 presents the overall success rate of the detection algorithm. More specifically, the success rate is defined as the number of times where the estimated starting point was within 100-200 samples prior to the actual start of the transient, divided by the total number of test signals. While the overall success rate is 89.5%, the success rate of each model is as follows: 3Com (91%), Ericsson (84.5%), Test Radio (96.5%). Additionally, the overall mean and standard deviation are 90.67 and

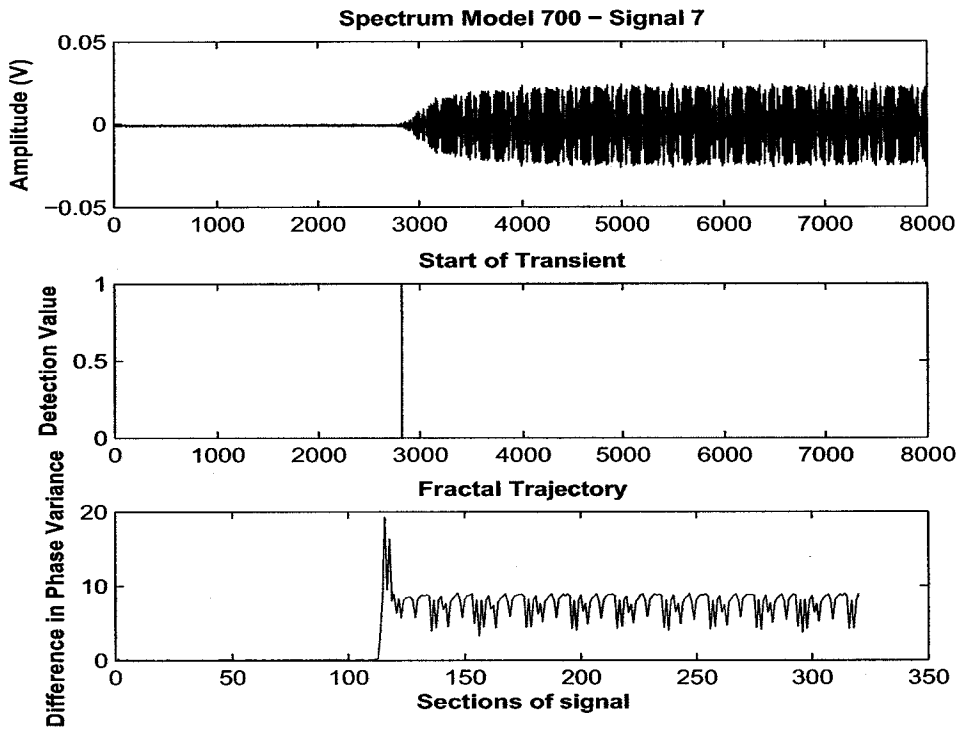

Figure 6.9: Test Case for TDPC (802.11b transceiver)

6.007 respectively.

During the process of testing, some general observations were made:

- Signals, from the transceivers used by 3Com cards and the Test Radios, were very consistent and so were the phase variance of the noise and transient portion of the signal. On the other hand, those from Ericsson transceivers exhibited much more variation, resulting in a lower success rate;
- The frequency hopping behavior of the transceivers was clearly noticeable, although it did not affect the performance of the detection algorithm; and
- As the accuracy of the detection algorithm is based on the phase characteristics (not amplitude) of the signal, it is feasible that this algorithm can be applied to signals from other wireless devices, in addition to 802.11 transceivers.

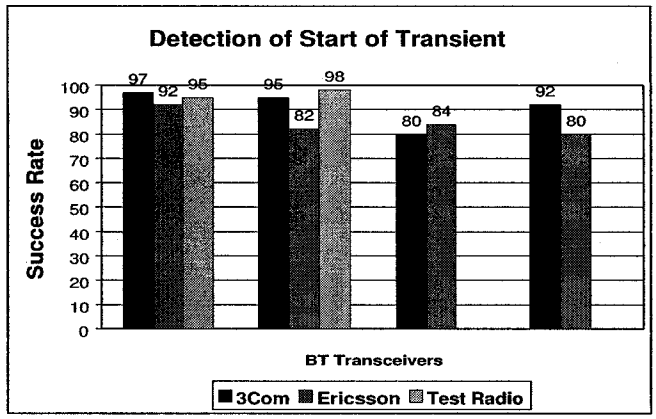

Figure 6.10: Success Rate of Transient Detection (BT transceivers)

6.1.3 Comparison of Approaches

Transient Detection using BT Transceivers

A brief comparison of the Threshold, BSCD and TDPC approaches is provided in Table 6.1.

Threshold In terms of performance, this detection algorithm, with a worst case running time of order n, works well for signals with an abrupt change at the start of the transient. However, establishing an appropriate threshold proves to be challenging. Moreover, the algorithm does not take into account any abrupt spikes, after the first $T/4$ samples, but within the noise segment of a signal. This will result in a premature detection of the start of a transient.

BSCD In comparison to the Threshold detection algorithm, the BSCD is less efficient with a worse case running time of order n^2. Although, it could be considered

Factors	Threshold	BSCD	TDPC
Strength	Algorithms can be easily modified to improve performance	No threshold required; can be used with various types of signals	Does not exhibit some of the weaknesses of other approaches
Weaknesses	Threshold is difficult to establish due to the choice of signal features; does not handle abrupt spikes within channel noise	Poor detection - does not handle 1) spikes in channel noise and 2) signals with slow rate of change at the transition point	Use of system-wide threshold is not highly robust
Complexity	O(n)	$O(n^2)$	O(n)
Success Rate	Not Available. Experiments were discontinued.	80-85%	85-90%

Table 6.1: Comparison of Transient Detection Approaches

more robust than the Threshold mechanism, it does not perform as well with signals, which exhibit similar characteristics as the waveform, depicted in Fig. 6.8, 1^{st} plot. The key advantage, nevertheless, is that it can be applied to various types of signals without having prior knowledge of their specific characteristics.

TDPC With an order of n, the success rate of the algorithm is approximately 85-90%. In comparison to our implementation of the BSCD (approx. 80-85%), the new algorithm is more suited to the characteristics of BT and 802.11b signals. The success rate can be further improved through the use of transceiver-based thresholds.

Transient Detection using 802.11b Transceivers

The histograms, which depict the range of detection errors for TDPC and BRCD, are presented in Fig. 6.11 and Fig. 6.12 respectively. Whereas positive values signify premature detection, i.e. estimated start point is prior to the actual value, the opposite holds true for negative values. Additionally, the bins are equally spaced (50) between the minimum and maximum detection error. In terms of other statistics, e.g. the mean and standard deviation, for confidence level of 95%, are presented in Table 6.2.

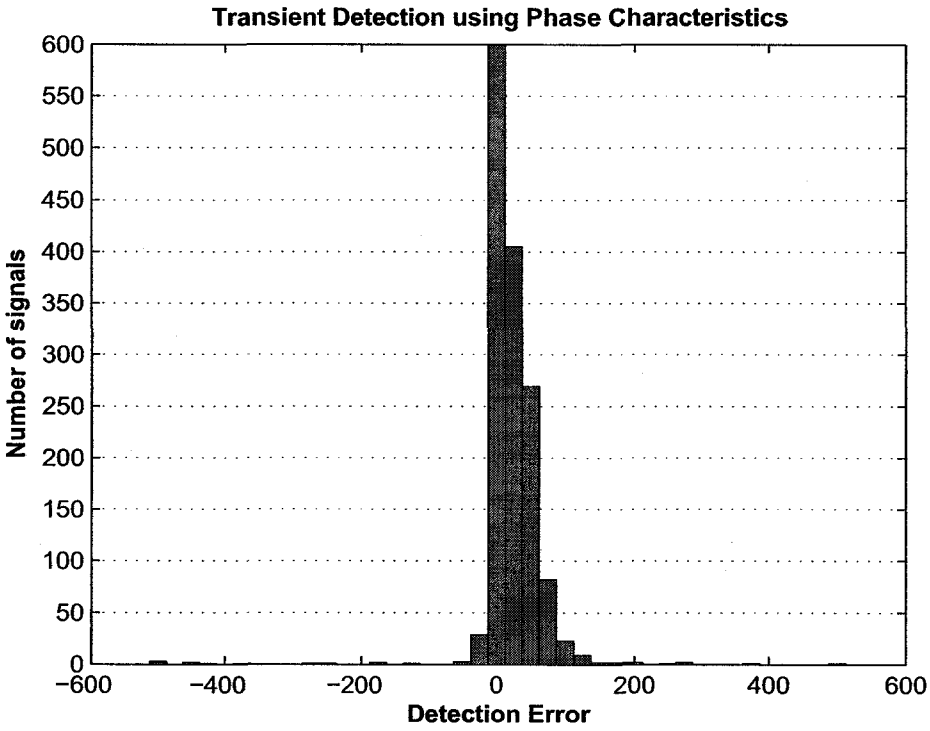

Figure 6.11: Detection Results for TDPC (802.11b transceivers)

Approach	Mean Value	Standard Deviation	CI-Lower	CI-Higher
TDPC	21.22315	51.47388	18.56726	23.87903
BRCD	-594.7457	1082.248	-650.5862	-538.9051
CI=Confidence Interval				

Table 6.2: Comparison of TDPC and BRCD statistics

As depicted in Fig. 6.11, approximately 2.083% (30/1440)of detection errors are associated with delayed detection (range: 25-50 samples), whereas the remaining errors are related to premature detection (range: 0-200 samples). As previously stated, premature detection is preferable over delated detection, since the latter results in the loss of valuable transient data. However, the difference between the estimated and actual start point should be minimized, in order to accurately capture the transient data.

In contrast, approximately 95% of detection errors are related to delayed detection (range: 25-2500) in the case of BRCD. The remaining 5% of the errors are distributed

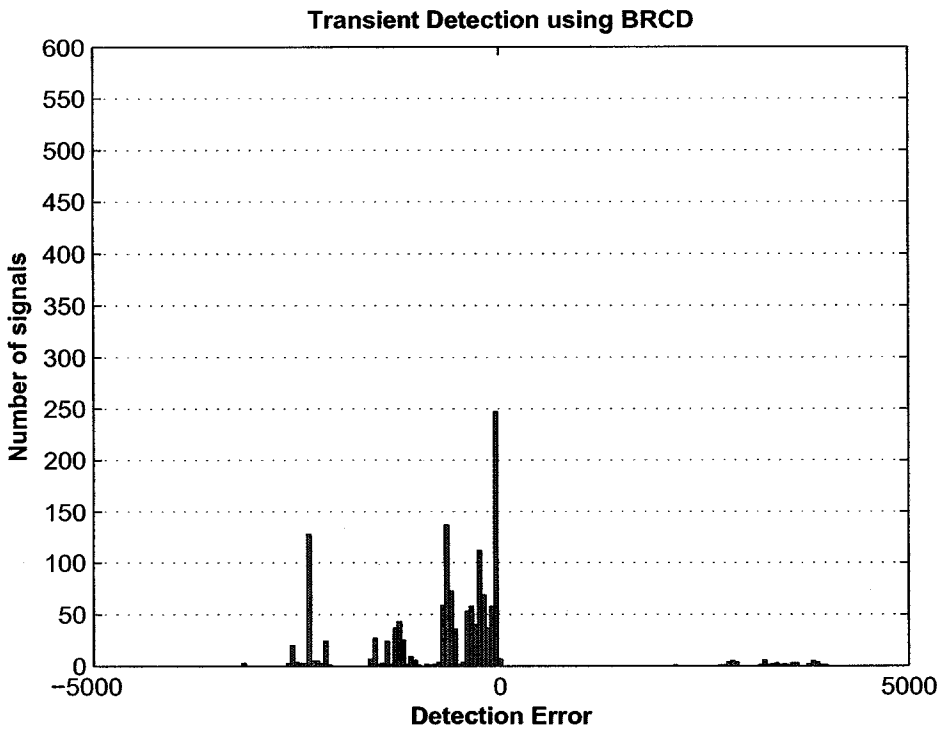

Figure 6.12: Detection Results for BRCD (802.11b transceivers)

between 3000-4000 samples after the actual start point. Although these results would render BRCD unsuitable for this set of 802.11b transceivers, additional tests, using other models, are required in order to fully assess the benefits of this algorithm.

6.1.4 Evaluation

BT Transceivers

In order to capture signals from BT transceivers, the following procedure was used, see Fig. 6.13. The 2400-2483.5 MHz signals were captured using an Omni (3 dBi) antenna. Using the Rohde & Schwarz RF generator, with the output level set to +3dB, and the Watkins Johnson MIG mixer, an intermediate frequency (IF) signal (5-105 MHz) was produced. This signal was then filtered twice for higher fidelity, using a MiniCircuits BBLP-156 LPF with a cutoff frequency of $\sim$ 90 MHz. The filtered signal, in turn, was sampled at 500 MHz using the LeCroy 9354L digital oscilloscope (8 bit ADC). Resulting samples were converted from binary to ASCII

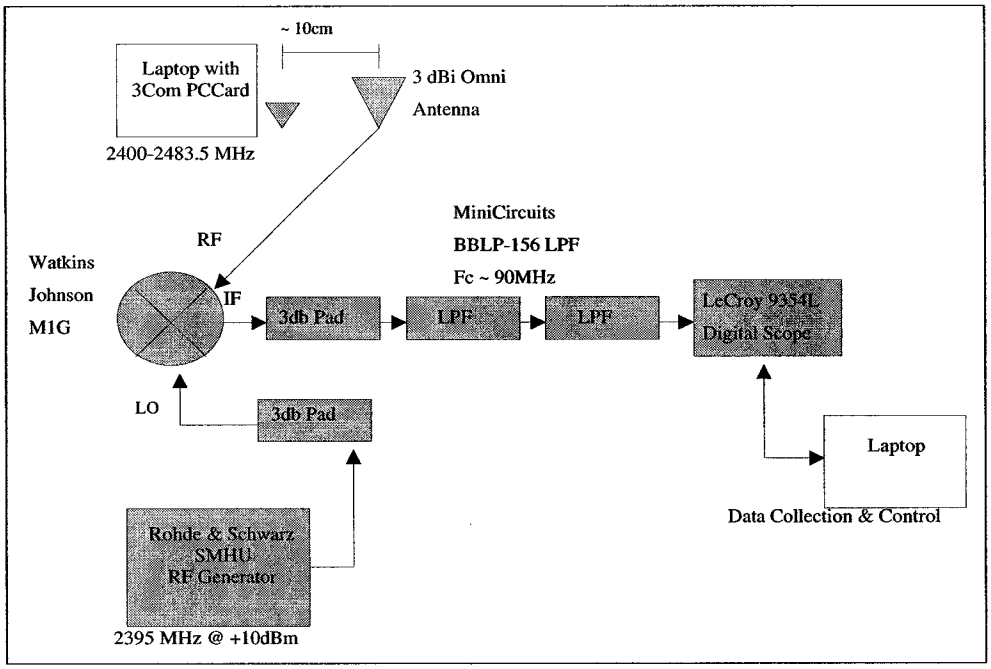

Figure 6.13: Infrastructure for Signal Capture (BT)

format. The raw signal (real data) were converted to analytic pairs (complex data), in software, using the Hilbert transform.

In order to evaluate the aforementioned detection strategies, on a variety of transceivers, three different models were used: 3Com Model 3CRWB6096 (4 units), Ericsson Model ROK101008/21 (4 units), Test Radios Model 3CW1057-E (2 units) for a total of (10) transceivers. One hundred signals per transceiver were captured resulting in a test base of 1000 signals.

All subsequent processing and evaluations were carried out using Matlab software and associated tools. As far as the evaluation platform is concerned, a notebook (HP Pavilion N5445) with 256 Mbs of memory and running XP and Matlab software was used.

802.11b Transceivers

Signals from 802.11b transceivers were captured using the following procedure,

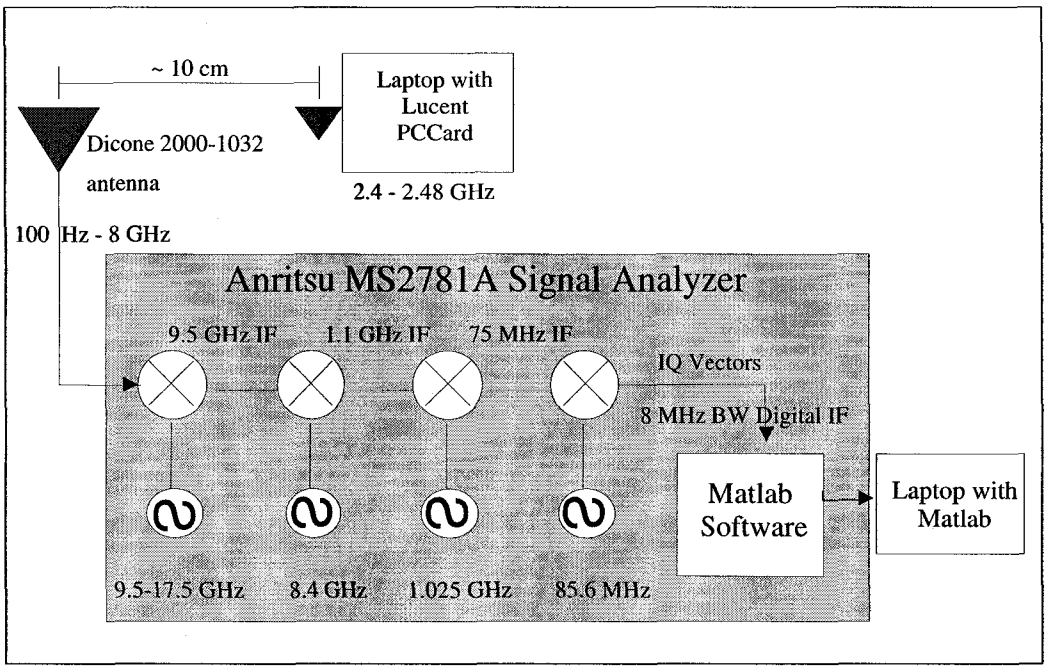

Figure 6.14: Infrastructure for Signal Capture (WiFi/802.11)

see Fig. 6.14. A laptop, equipped with an 802.11b wireless PC card (configured for a specific ad-hoc network), was used to generate the signals. The signals were captured using the (Dicone 2000-1032) antenna with the range of 100Hz-8 GHz. The captured signals, with a central frequency of 2.422 GHz (channel 3), were then processed by the Anritsu MS2781A Signal Analyzer. First, the analog signals were upconverted to 9.5 GHz, using a 11.9 GHz fixed Local Oscillator frequency. This intermediate frequency (IF) was subsequently downconverted to approximately 1.1 GHz followed by a second downconversion to 75 MHz. A 30 MHz bandwidth anti-aliasing filter was then applied to the third IF prior to being digitized at 100 MHz. This action resulted in the creation of an alias at 25 MHz (100-75 MHz) with a 30 MHz bandwidth. In addition, other filtering and gain were applied to the signal from the input to the digitizer. The digitized signal was downconverted to in-phase and quadrature (I/Q) vectors at 0 Hz. Finally, it was resampled at 8x the symbol rate of 11 MHz, resulting in a final sampling rate of 88 MHz that satisfies the Nyquist sampling rate.

The resulting I/Q vectors were eventually transferred by the analyzer to the Matlab environment (as variables) for further processing.

For the purpose of RFF, 105-110 signals, from each of the 802.11b transceivers, were captured. Two sets of transceivers were used: 15 transceivers (different models) for transient detection and 30 transceivers, from Lucent Technologies, for the remaining components of RFF. The decision to use 30 transceivers was influenced by the standard for simulation. Furthermore, according to the central limit theorem, a sampling size of 30 is deemed sufficient for obtaining a normal distribution and establishing various confidence intervals (e.g. 95%). All subsequent processing and evaluations were carried out using the aforementioned infrastructure.

6.1.5 Extraction of Transients

Once the start of a transient has been detected, the end point is determined based on the length of the transient, and in consultation with technical specifications. As far as 802.11b transceivers are concerned, section 18.4.7.6 of the specification dictates that the transmit power-on ramp, from 10% to 90% of maximum power, be limited to 2 microseconds. Therefore, based on the sampling rate of 88 MHz, the length of the transient is approximately 1024 samples. Hence, this number of samples, from the start of a transient, are extracted for further analysis.

6.2 Component Extractor

Once the transient of a signal has been identified, the next requirement is to extract the three primary components of the transient. The instantaneous amplitude, phase, and frequency components are obtained using Eqs. A.3, A.4, A.7 and A.8 respectively.

These components, from two Lucent (802.11b) transceivers (404 and 665), are illustrated in Figs. 6.15 and 6.16 respectively. While the original signal, as well as the transient (segment of the signal between two vertical lines) is displayed in the first plot, the remaining three plots present the three components of the transient.

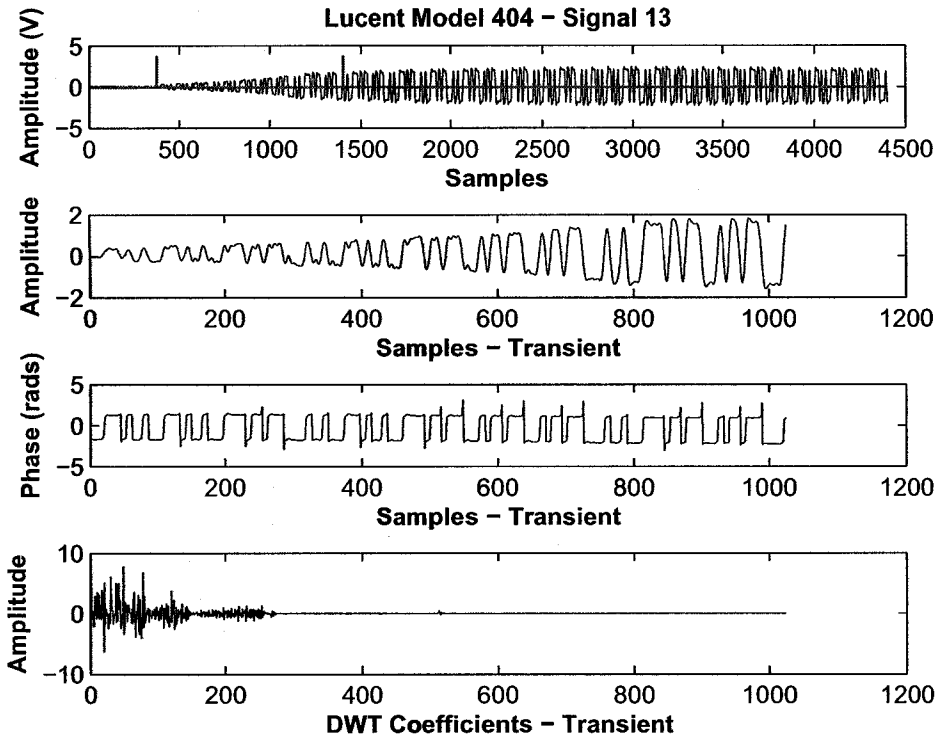

Figure 6.15: Components of a transient: Transceiver 404

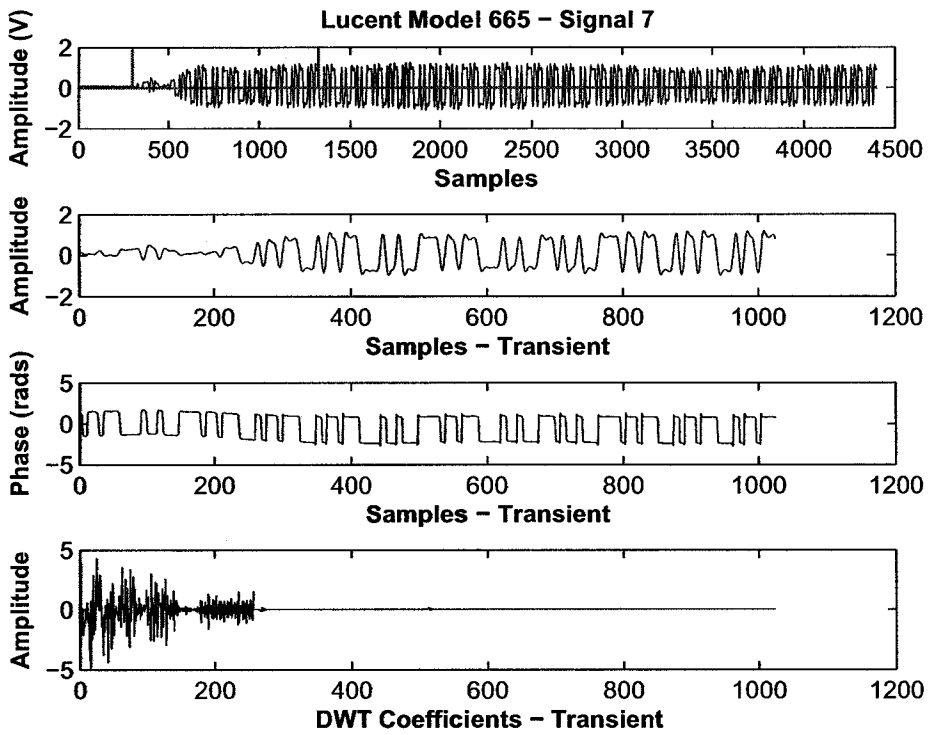

Figure 6.16: Components of a transient: Transceiver 665

There are some observations that are noteworthy. First and foremost, the individual components of the two transients are dissimilar to some extent, especially the DWT coefficients. As a side note, due to its low computational complexity and its ability to provide perfect reconstruction of a signal, as stated by Choe *et al.* [31] and Hippenstiel and Payal [83], the Daubechies wavelet/filter is used to obtain the DWT coefficients. In any event, even a small variance, between each of the corresponding components, proves to be useful in classifying transceivers from the same manufacturer. In fact, the specific features that are extracted from these components (discussed in section 6.3) serve to highlight the dissimilarities between all transceivers.

Second, the DWT coefficients of significance are located in the lower half of the spectrum (i.e. coefficients 1 to 250 for both transceivers). Thus, a reduction in the number of coefficients used, improves the performance of the feature extraction process. Lastly, slight nuances, e.g. small oscillations (not noise) between samples 350-450 (first plot) in Fig. 6.16, are also exploited to improve the characterization of different transceivers.

Once these primary components have been extracted, a feature vector, which represents a transceiverprint, is created. It is defined as

$$\overrightarrow{FV} = (F_1, F_2, \ldots, F_p)$$

where F_i represents a feature that is extracted from one of the components and p specifies the number of features being used. Please note that the terms transceiverprint and feature vector are used interchangeably in the following sections.

6.3 Feature Extractor

In order to define a transceiver profile, one must first determine the composition of a transceiverprint. Although there are many suitable techniques, e.g. the use of fractal, information and correlation dimensions, as proposed by Ureten and Serinken [181], the underlying requirement is the same: to select one or more features that have low *intra-transceiver* variability (within a transceiver) and high *inter-transceiver* variability (between transceivers). While high inter-transceiver variability is crucial for

Number	Feature	Equations used
F1	Normalized DWT coefficients	(A.6)
F2	Normalized amplitude	(A.3)
F3	Normalized phase	(A.5)
F4	Variance in amplitude fluctuation	(A.3)
F5	Normalized in-phase data	(A.3)
F6	Normalized quadrature data	(A.2)
F7	Normalized amplitude (mean centered)	(A.3)
F8	Power per section	Defined in the sequel
F9-15	Normalized DWT coefficients by levels	(A.6)

Table 6.3: Features in a transceiverprint

maximizing DRs (distinguishing transceivers from the same manufacturer), low intra-transceiver variability is equally important for minimizing FARs.

The selection of features is carried out through an iterative process. First, a preliminary set of features is selected, based on similar research initiatives undertaken by other research teams. Second, both the inter and intra-transceiver variability are assessed. Finally, based on the results, the preliminary set of features is altered in an iterative manner.

The final set of features (used in this iteration) is enumerated in Table 6.3 and is further defined in the sequel. Please note that the standard deviation has been used for all features in order to minimize the difference, between the range of values, associated with each feature.

Normalized DWT coefficients (F1)

$$DWTC = \sqrt{(DWTC_n - M_{nd})^2}$$

where $DWTC_n$ represents the normalized amplitude of a DWT coefficient and is denoted as $\frac{DWTC_i}{M_d}$. While $DWTC_i$ represents the amplitude at time instant t ($i = 1, 2, \ldots, 512$), $M_d = max\{DWTC_i\}$ and it is the maximum of the amplitudes of the coefficients. Finally, the mean of the normalized coefficients M_{nd}

is defined as $\frac{1}{512}\sum_{n=1}^{152} DWTC_n$.

Normalized amplitude, phase, in-phase and quadrature (F2,F3,F5,F6)

$$SNA = \sqrt{(A_n - M_{na})^2}$$

where A_n represents the normalized instantaneous amplitude and is denoted as $\frac{A_i}{M_a}$. While A_i represents the amplitude at time instant t ($i = 1, 2, \ldots, N$), $M_a = max\{A_i\}$ and it is the maximum of the instantaneous amplitudes. Finally, the mean of the normalized amplitudes M_{na} is defined as $\frac{1}{N}\sum_{n=1}^{N} A_n$ whereas N signifies the total number of samples.

The standard deviation of the normalized phase (SNP), in-phase (SNI), and quadrature data (SNQ) are also obtained by replacing A_i with P_i, ID_i and Q_i respectively.

Variance in amplitude fluctuation (F4)

$$VAD = \sqrt{(AD_n - M_{ad})^2}$$

where AD_n is the difference between each two subsequent values of the instantaneous amplitudes in the transient and $n = 1, 2, \ldots, N-1$. M_{ad} represents the average value of AD_n and is defined as $\frac{1}{N}\sum_{n=1}^{N} AD_n$.

Normalized amplitude (mean centered) (F7)

$$\sigma_{An} = \sqrt{(A_c - M_{an})^2}$$

where A_c represents the normalized mean-centered instantaneous amplitude and is denoted as $\frac{A_i}{M_a} - 1$. While A_i represents the amplitude at time instant t ($i = 1, 2, \ldots, N$), $M_a = \frac{1}{N}\sum_{i=1}^{N} A_i$ and it is the average of the instantaneous amplitudes. Finally, the mean of the normalized amplitudes M_{an} is defined as $\frac{1}{N}\sum_{c=1}^{N} A_c$.

Power per section (F8)

$$PPS = \sqrt{(P_s - M_s)^2}$$

where P_s represents the power associated with a given subset of instantaneous amplitudes. While $s = 1, 2, \ldots, N$, $N = \frac{T}{W_{size}}$ with T representing the size of the transient and W_{size} the predefined window size. The power value for set $s = 1$ ($P_{s=1}$) is defined as $\sum_{(W_{size}\times(s-1))+1}^{W_{size}\times s} A_i^2$. Other values for P_s are calculated in a similar manner by shifting the window by one (e.g. $s = 2$ and $W_{size} = W_{size} \times s$). Lastly, M_s is defined as $\frac{1}{N}\sum_{s=1}^{N} P_s$.

Normalized DWT coefficients by levels (Start=129:End=256) (F9-F15)

$$DWT = \sqrt{(DWT_n - M_{dwt})^2}$$

where DWT_n represents the normalized DWT coefficients and is denoted as $\frac{DWT_i}{M_{wc}}$. While DWT_i represents the amplitude of the coefficients at time instant t ($i = S, S+1, \ldots, E$), $M_{wc} = \frac{1}{E-S+1}\sum_{i=S}^{E} DWT_i$ and it is the mean of the coefficients for a given level. Lastly, the mean of the DWT coefficients M_{dwt} is defined as $\frac{1}{E-S+1}\sum_{n=S}^{E} DWT_n$. The coefficients for (S=65:E=128), (S=33:E=64), (S=257:E=512), (S=9:E=32), and (S=1:E=8) are obtained in a similar manner.

Next, in order to determine the two classes of variability, the use of ED and clustering techniques of MVA are employed [95].

The extraction of the preliminary set of features from each of the transients, associated with a given transceiver, results in a set of feature vectors referred to as a cluster. In order to assess the inter-transceiver variability between two or more transceivers (clusters), a centroid (composed of the average value of each of the features in the clusters) is created for each transceiver according to Eq. 7.2. The inter-transceiver variability, between three 802.11b transceivers from Lucent, is illustrated in Fig. 6.17.

What is of interest are the features (x-axis), which have noticeably different values

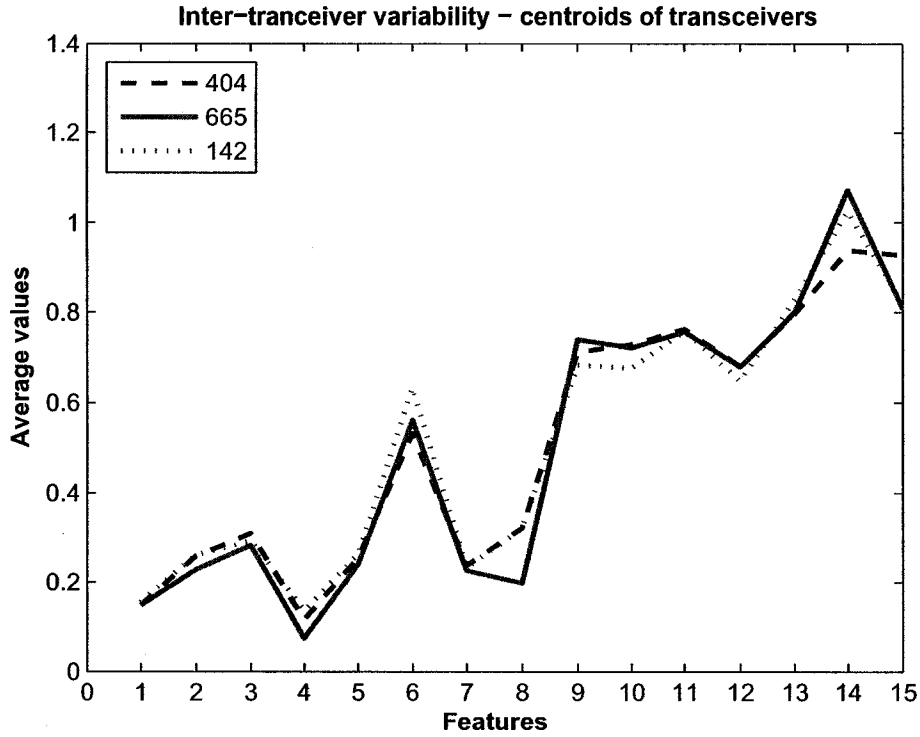

Figure 6.17: Inter-transceiver Variability

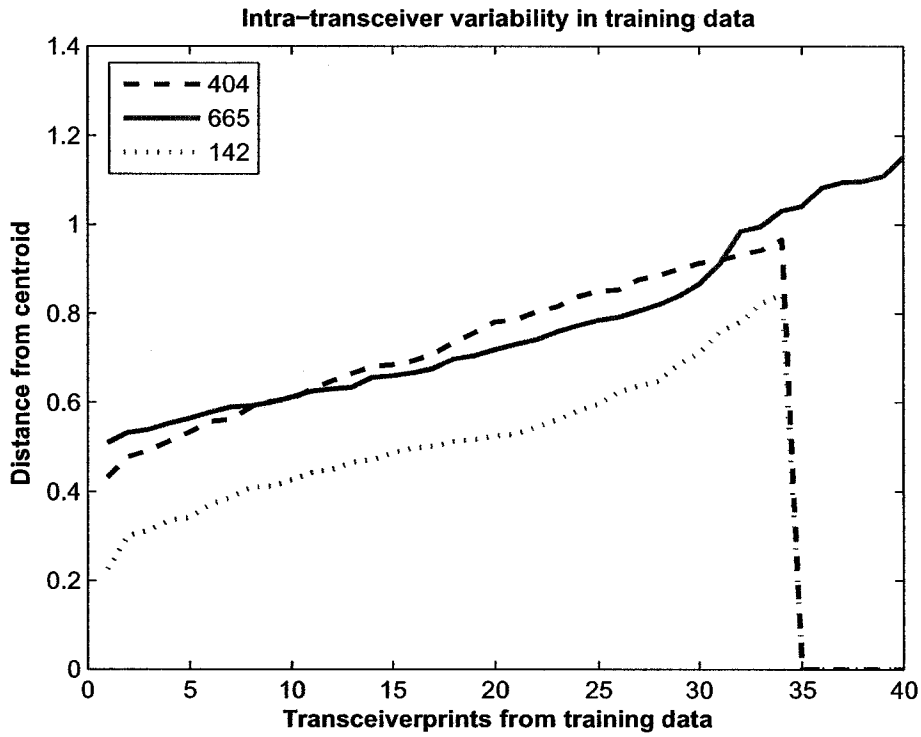

Figure 6.18: Intra-transceiver Variability

(y-axis) for each of the transceivers being analyzed. Hence, for example, features 4 and 14 would prove more useful in distinguishing transceivers from the same manufacturer.

Intra-transceiver variability, on the other hand, is depicted in Fig. 6.18. The individual data curves represent the ED (y-axis) of each of the transceiverprints (x-axis) from the corresponding centroids. The transceiverprints have been sorted based on the ED in order to determine the range of variability/dispersion. Although an optimal range of intra-transceiver variability would be represented by a horizontal line (no variability), the slow rising slopes of the data curves for transceivers 142, 665 and 404 demonstrate a low degree of variability (e.g. 0.6 for transceiver 142) amongst the transceiverprints in the corresponding clusters.

While normalizing one or more features does reduce intra-transceiver variability to some extent, variability caused by factors, such as internal temperature (within the wireless card), cannot be eliminated. Nevertheless, the range of intra-transceiver variability, as defined by the variability of each feature, can be represented in a profile using the covariance matrix.

Once the composition of a transceiverprint has been established, through an iterative process, a subset of the transceiverprints, from the original data set, is used to create a profile for each transceiver. This subset is selected using K-means clustering [110]. However, a more optimal subset can subsequently be selected based on the results of the evaluation process.

6.4 Profile Definition

A formal definition of a transceiver profile is presented in this section. With the exception of elements E7 and E8, which are only required for dynamic profiles (i.e. for profile updates), all other elements in Table 6.4 are stored on a permanent basis.

A detailed description of the individual elements is as follows:

Centroid A centroid is a (1 × p) vector, which represents the average value (calculated using a set of transceiverprints) of each of the p features. It is also

Element	Description	Use
E1	Centroid	Classification
E2	Covariance matrix	Classification
E3	Transceiver identification code	Profile selection
E4	Upper ED threshold	Classification
E5	Lower ED threshold	Classification
E6	Transient threshold	Transient Extraction
E7	Intra-transceiver variability	Profile update
E8	Set of transceiverprints	Profile update

Table 6.4: Elements in a transceiver profile

used to determine the Hotelling T^2 value of a test transceiverprint during the classification process.

Covariance matrix A covariance matrix of dimension (p x p) characterizes the intra-transceiver variability of each feature with respect to one another. As with the centroid, it is also used to determine the T^2 value of a test transceiverprint.

Transceiver identification code This identification code of three bytes is used as an index to a database of transceiver profiles.

Upper ED threshold The upper ED threshold is used to exclude test transceiverprints, which are considered outliers, i.e. those exceeding the threshold, from the classification process. The threshold itself is determined by analyzing the ED of all transceiverprints of a given transceiver. Fig. 6.19 depicts the upper and lower thresholds for transceiver 55 from Lucent.

Lower ED threshold In contrast to the previous threshold, the lower ED threshold provides the lower bound.

Transient threshold As illustrated in Fig. 6.7, this value is used to determine the start of a transient. Although a system-based threshold may suffice, experimentation supports the use of transceiver-based values for optimal performance.

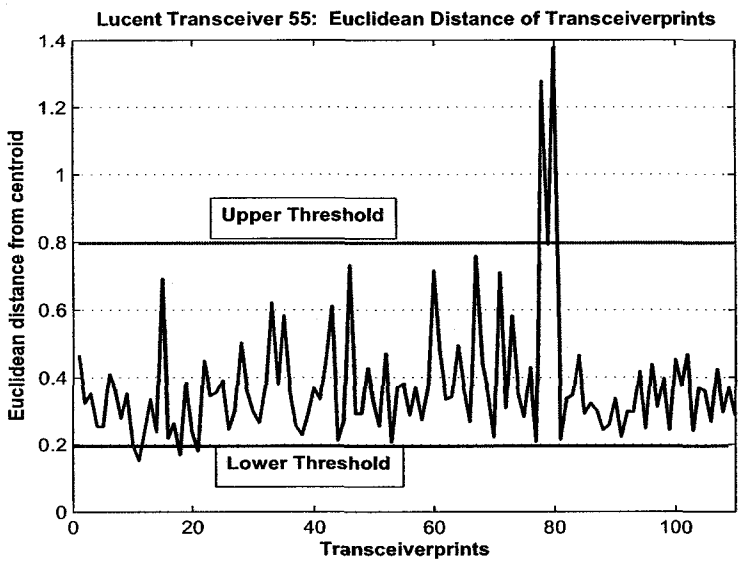

Figure 6.19: Upper and Lower Thresholds

Intra-transceiver variability As aforementioned, the variance of the ED (of all transceiverprints from a transceiver) indicates the degree (e.g. high or low) to which these transceiverprints are similar to the centroid. This value is used for two purposes: 1) selecting a transceiverprint from a set of test transceiverprints, which has been classified as normal, to be incorporated into the set of transceiverprints (E8) and 2) selecting a transceiverprint from E8 for eviction. Thus, for example, a test transceiverprint with the highest ED from the centroid is selected if the intra-transceiver variability is high, thus preserving the overall intra-transceiver variability of a transceiver.

Set of transceiverprints The requirement for storing a set of transceiverprints, which characterize a transceiver, is necessitated by the profile update process, as described in section 6.5.

6.5 Profile Update

As stated in section 5.2.1, a profile must be updated periodically so that it continues to represent the current characteristics of the corresponding transceiver. To this end, the

use of a MAF is employed. Although the Exponentially Weighted Moving Average Filter (EWMAF) [108], another MSPC technique, is commonly used with process measurement data that is acquired over a long (e.g. days or months) period of time, its use in profile updates is not suitable for two reasons. First, the EWMAF is rarely used for updating the profile of a given process. It is used instead for emphasizing the importance of the most recent measurement data. Second, in light of the fact that a transceiver profile is being updated continuously, the benefits of employing the EWMAF become negligible.

Under these circumstances, one simple strategy is to update the profile by replacing one of the transceiverprints in the set (E8) with a test transceiverprint, which has been recently classified as normal. A transceiverprint in E8 is selected for expulsion using First In First Out (FIFO). On the other hand, the test transceiverprint is selected according to E7. Once it has been incorporated into the set (E8), the centroid and covariance matrix are recalculated and stored in the profile. Although it is possible to update these elements, without having to store the transceiverprints, as suggested by Samfat and Molva [149]. This technique will be explored in the next iteration. Hence, for the time being, a performance penalty is incurred. For example, the inversion of a covariance matrix has a worse case running time of $O(p^3)$ [36], where p represents the number of features associated with a transceiverprint.

Chapter 7

Classification Phase

As you may recall, the key requirement to be fulfilled, during the classification phase, is the determination of whether a transceiverprint is normal or anomalous.

7.1 Identification of Transceivers

With regards to the identification of transceivers, the guiding factor is the need to determine the classification probability of a single transceiverprint. More precisely, the probability of a match between an observed transceiverprint and each of the transceiver profiles in the IDS, must be determined. The resulting *set* of probabilities is subsequently used by the Bayesian filter (discussed in the sequel) to identify the transceiver, which is most likely (i.e. highest probability) to have generated the signal/transceiverprint.

In order to determine the probability of a match, a statistical classifier is employed. It uses a set of variables, in this case, a set of features, to represent a vector that is to be classified. The probability of a match is calculated using a simplified version (no multiplication by constants) of the Kalman filter from Bar-Shalom [17]. It is based on the Hotelling T^2 statistical measure and is defined as follows:

$$P(FV_i) = \exp\left[-\frac{1}{2}\left((FV_i - \overline{FV})^T S^{-1}(FV_i - \overline{FV})\right)\right] \tag{7.1}$$

where FV_i denotes a feature vector (transceiverprint) to be classified, $\overline{FV}$ corresponds to the centroid (vector composed of the average value of each feature) and S^{-1} is the inverse of the covariance matrix S, which characterizes the dispersion or variability of each feature with respect to one another.

The centroid is defined as:

$$\overline{FV} = (\overline{F}_1, \overline{F}_2, \ldots, \overline{F}_p) \tag{7.2}$$

where $\overline{F}_1$ represents the average of the first column in the table of features or variables.

In addition, the covariance of two variables (e.g. the Standard Deviation of Normalized Amplitude v and Standard Deviation of Normalized Phase w) is defined according to

$$cov(F_v, F_w) = \frac{1}{R}\sum_r \left(((r, F_v) - \overline{F_v})((r, F_w) - \overline{F_w})\right)$$

where r represents a row and R is the total number of rows in the table. The $p \times p$ covariance matrix S is created by incorporating the covariances of all the variables:

$$\begin{pmatrix} cov(1,1) & cov(1,2) & \ldots & cov(1,p) \\ cov(2,1) & cov(2,2) & \ldots & cov(2,p) \\ \ldots & \ldots & \ldots & \ldots \\ cov(p,1) & cov(p,2) & \ldots & cov(p,p) \end{pmatrix}$$

Finally, the inverse of S is calculated according to pre-established mathematical algorithms.

Eq. 7.1 returns a probability based on the relationship between an observed tranceiverprint and the profile of a transceiver. First, the difference between the transceiverprint and the centroid $(FV_i - \overline{FV})$ is determined. The smaller the deviation, the more likely the transceiverprint belongs to this transceiver. Furthermore, the likelihood of membership is also established, by assessing the variability of each of the features in the transceiverprint, with respect to the covariance matrix.

7.1.1 Bayesian Filter

As previously mentioned, most of the current IDSs render a verdict, as to whether or not an observed behavior (e.g. user's session data, system calls and network traffic) is normal or anomalous, after a single observation. This approach may be adequate, under certain circumstances, where normal behavior can be profiled accurately. However, in a wireless environment, characterized by noise and interference, there is a potential for increased variability between signals transmitted from a given transceiver. Under these circumstances, the application of the Bayesian filter is useful for reducing the number of false positives or false alarms.

The following discussion focuses on the operational details of the filter, as it is used during the evaluation phase. The key difference between the evaluation and detection phases is the use of a predefined threshold for determining/detecting whether or not the transceiverprint does in fact belong to the target transceiver.

The Bayesian filter probabilistically estimates the state of a system from noisy observations. In RFF, the state is defined as the transceiver model(s) (from the list of profiled transceivers) to which the extracted transceiverprint (observation) is closely related to. At each point in time t, a set of probabilities, called *belief*, over the state (x_t), represents the uncertainty and is denoted as $Bel(x_t)$. Hence, at $t = 1$, the set of probabilities for the first transceiverprint is obtained from the classification process. Since the final probability of the state is based on multiple observations, the filter sequentially estimates such beliefs over the transceiver space. Hence, the belief is estimated for $(t = 1, 2, \ldots, 10)$. Finally, as indicated by Eq.7.3, the belief at time t represents the current probability that has been influenced by the probability of the previous observation o_{t-1} at $t - 1$.

$$Bel(x_t) = p(x_t|o_t)Bel(x_{t-1}) \tag{7.3}$$

The result of applying the Bayesian filter for transceiver 404 is shown in Fig.7.1. The belief/probability of a match between a given transceiverprint (y-axis) and each of the 30 profiled transceivers (x-axis) is indicated by the z-axis.

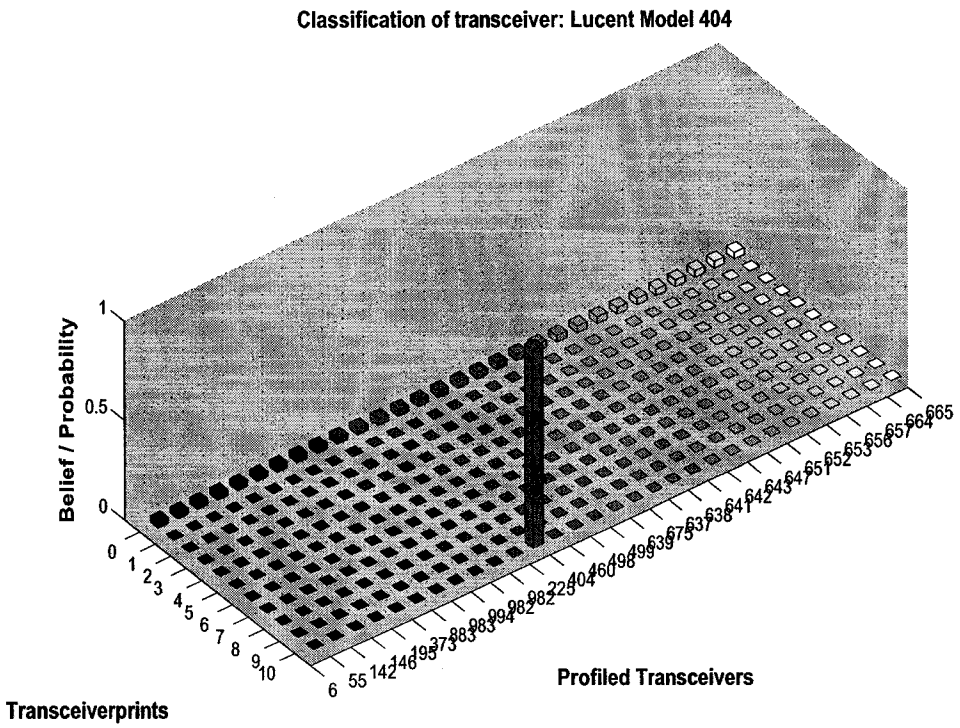

Figure 7.1: Application of the Bayesian filter

Let us take a closer look at the filtering process. Initially the $Bel(x_0)$ is uniformly distributed, as indicated by the similar height of the vertical bars for (y=0) along the z-axis. This reflects the fact that the filter has no prior knowledge about the transceiver that is most likely to be associated with the first transceiverprint. Once the set (30 transceivers) of probabilities has been obtained for the first transceiverprint (y=1), it is multiplied by $Bel(x_0)$. Essentially, the probability assigned to each of the transceivers at $t = 1$, by the classification process, is multiplied by its corresponding value at $t = 0$. This process continues for t iterations (10 in this case), at which time, the final probability distribution is normalized to unit length. The transceiver with the highest probability is expected to be the correct transceiver. As indicated by Fig.7.1, the transceiver most likely to have transmitted the signals is 404 with a probability of 100%. A final probability of less than 100% is indicative of sub-optimal characterization, i.e. the selection of features and/or transceiverprints for profiling purposes, of a transceiver.

7.1.2 Details of Evaluation

The purpose of the evaluation exercise was to primarily assess the composition of a transceiverprint based on the classification success rate (metric). More specifically, the following steps were carried out:

Step 1: For each transceiver being profiled, the transceiverprints were obtained from the transients of corresponding signals. The transient itself was extracted using the approach described in [71]. Although the BSCD and Threshold approaches were also implemented, they were not suitable for 802.11b signals, as indicated by the results (80-85%).

Step 2: A subset (approximately 35-40) of the transceiverprints was selected (based on the ED) and subsequently used to create a centroid and covariance matrix. The remaining transceiverprints (approximately 60) were used for testing purposes.

Step 3: The evaluation exercise was carried out by: selecting a transceiver to be tested (from a list); obtaining a set of 10 consecutive test signals (for that transceiver) from a given starting point (changed between each iteration); extracting the transceiverprints from each signal; classifying each test transceiverprint and thus obtaining the necessary set of probabilities (one for each transceiver); determining the transceiver with the highest probability using the Bayesian filter; and finally incrementing the counter of this transceiver (used for calculating success rate), if the test signals did in fact originate from it.

7.1.3 Evaluation Results - RFF and Bayesian Filter

After 50 iterations of the evaluation exercise, the following classification success rate (number of correct classification / number of iterations) was achieved:

Based on evaluation results, there are some observations that are noteworthy. First, the high success rates for most of the transceivers attest to the overall contribution of the features in the transceiverprint and to the quality of the characterization of the transceivers. It is feasible, however, to optimize the feature vector by eliminat-

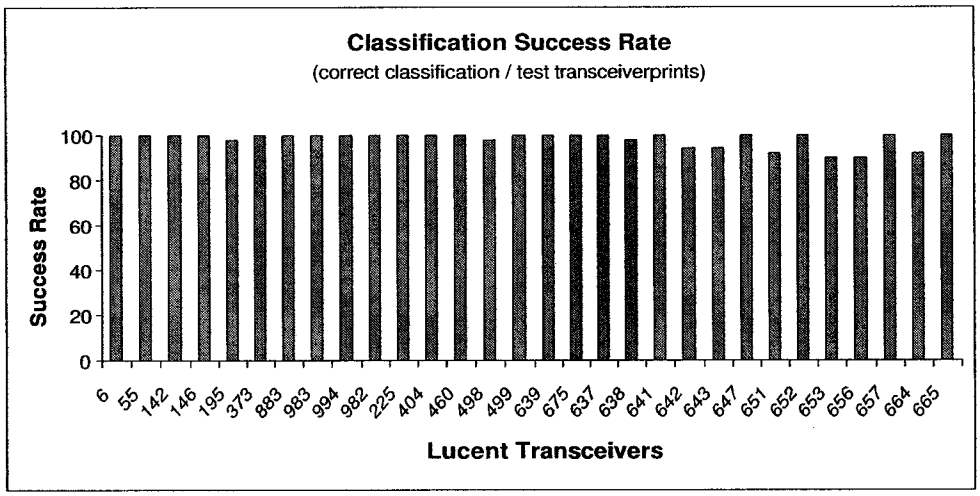

Figure 7.2: Classification Success Rate (802.11b transceivers)

ing features that provide only minimal contribution. The use of principle component analysis will prove beneficial in this regard.

Second, the success rate for transceivers, e.g. 656, can be further improved by characterizing the transceivers more accurately. Techniques such as the SOM, k-means clustering and others can not only be used to improve characterization but to reduce memory requirements as well.

Finally, an improvement in the success rates achieved by the classifier (e.g. from 92% to 100% and from 95% to 100% for transceivers 225 and 460) provides evidence to support the use of the Bayesian filter.

Although it would prove beneficial to compare the classification success rate with those obtained by other similar initiatives, it tends to be rather difficult due to the following reasons:

First, the use of statistical classifiers for the purpose of RFF is rather uncommon. Second, classifiers used for classification purposes are typically based on some variant

of neural networks e.g. PNN and ANN [198], and SOMs [98]. Finally, classification results obtained by various research teams, in the area of RFF, are very much application dependent and are based on various parameters including the size of the training set and number/type of features used.

Nevertheless, the type of research carried out by Choe [31] is similar to some degree. However, the number of profiled transceivers was limited to three (2-Motorola HT-220, 1-Motorola MX-330) in comparison to the 30 802.11b transceivers used in this project. Despite the increased complexity, the average success rate of $95\% \pm 0.894$ (95% confidence interval), achieved using RFF and Bayesian filter, is consistent with their results of (94%).

7.1.4 Memory Requirement and Running-Time Complexity

In this section, we briefly analyze both memory requirements and running-time complexity of the classification algorithm, which is used for identifying the transceiver most likely to have transmitted a set of observed signals.

Memory Requirements

One of the main reasons for exploring the use of a statistical classifier is to minimize both memory and running-time requirements. Although a PNN could have been used for the same purpose, the underlying algorithm is somewhat different. For example, in order to obtain a probability of a match, between an observed transceiverprint and a transceiver profile, a one (transceiverprint) to many (transceiverprints or training patterns in the profile) comparison must be carried out. In order to facilitate this process, all of the training patterns must be loaded into memory. When the size of a transceiverprint is fairly large (e.g. 1024-2048 bytes), and a large number of training patterns are required, the issue of scalability (memory requirement per profile (MPP)) prohibits its use for on-line systems. Of course, there are data compression techniques for reducing the size of transceiverprints, however, not without compromising the accuracy of the classifier to some degree.

In contrast, the MPP of a statistical classifier is more modest and is defined by

$$MPP(b, f) = bf + b(f^2)$$

where f is the number of features, b is the size in bytes, and bf as well as bf^2 represent the memory requirement for the centroid and covariance matrix respectively. Thus, for example, setting f to 15 and b to 4, results in a MPP of 960 bytes in comparison to 1,800 bytes (60 bytes per training pattern multiplied by 30 patterns) required for PNN.

Running-time Complexity

Before proceeding with the discussion of running-time complexity, a high-level overview of Algorithm 1 would prove useful.

In terms of prerequisites, the following elements are required: a profile, i.e. centroid FV_m and inverse of the covariance matrix S_m^{-1}, for each m of the $M = 30$ transceivers; and a set of $N = 10$ transceiverprints or feature vectors fv_n to be classified.

1. In order to accommodate the use of the Bayesian filter, a matrix $BM_{n+1,m}$ is used. Although all the elements of the matrix are initialized to zero, prior to the execution of the algorithm, the first row $BM_{1,M}$ is initialized to $1/M$ to indicate equal probability.

2. For each feature vector fv_n, the probability T_m, associated with each m of the transceivers is determined. These values are subsequently stored in $BM_{n,m}$, once they have been multiplied by the corresponding transceiver probability in row $n-1$.

3. Once the N feature vectors have been processed, the last row of probabilities $BM_{N=10,1:M}$, is normalized to unit length.

4. The maximum value of $BM_{N=10,1:M}$ is determined.

5. The transceiver number m, corresponding to the maximum value, is returned.

Algorithm 1 Transceiver Identification

$BM \leftarrow 0$; $BM_{1,M} \leftarrow 1/M$ {Determine probability for each transceiver for each feature vector}
for $n = 2toN$ **do**
  for $m = 1toM$ **do**
    $T_m \leftarrow \exp\left[-\frac{1}{2}(fv_n - FV_m)' \times S_m^{-1} \times (fv_n - FV_m)\right]$
    $BM_{n,m} \leftarrow T_m \times BM_{n-1,m}$
  end for
end for
{Determine sum of the last row of probabilities}
$sum \leftarrow 0$
for $m = 1toM$ **do**
  $sum \leftarrow sum + BM_{10,m}$
end for
{Normalize last row of probabilities and determine max value}
$maxValue \leftarrow 0; transceiver \leftarrow 0$
for $m = 1toM$ **do**
  $BM_{N+1,m} \leftarrow BM_{N+1,m}/sum$
  if $BM_{N+1,m} \geq maxValue$ **then**
    $maxValue \leftarrow BM_{N+1,m}$; $transceiver \leftarrow m$
  end if
end for
return *transceiver*

A high-level analysis of Algorithm 1 is presented next. The calculation of the set of transceiver probabilities m, for each of n feature vectors (lines 2-7), requires $n(m[p^2 + 1])$ operations. A discussion of the derivation of p^2 is presented in the sequel. Continuing with the algorithm, the normalization of the last row in $BM_{N+1,m}$ (lines 9-11 and 13-18) result in $4m$ operations. Thus, a total of $nmp^2 + nm + 4m$ operations are required for the identification of transceivers. In a real-world example, where profiles of many transceivers, e.g. $m = 1000$ must be consulted, scalability could become an issue.

Running-time for T_m

The term p^2 represents the estimated running-time for the calculation of T_m, the most time consuming aspect of the classification process. The analysis of S^{-1}, the inverse of the covariance matrix, is a good place to start. One option is to store the covariance matrix in the profile and to calculate its inverse, prior to the classification of each set of transceiverprints. However, a more suitable strategy would be to calculate and to store S^{-1} in the profile, thus eliminating the need to recalculate it during the classification process. Hence, this component does not negatively impact the overall performance of the classifier.

The next operation, represented by $((fv_n - FV_m)')(S^{-1})$, can be summarized as a multiplication of a matrix S^{-1} of dimension $p \times p$ by a vector of dimension $1 \times p$. According to Strassen's recursive algorithm [36], the multiplication of square matrices carries with it a worse-case running time of $O(p^2)$. Finally, the resulting vector $1 \times p$ and the last component $fv_n - FV_m$ of dimension $p \times 1$ are multiplied together within $O(p)$ time. Therefore, the worse-case running time of the classifier is approximately $O(p^2)$.

7.2 Verification of Transceivers

Unlike the previous classification exercise, the key objective, in this case, is to verify if a given transceiver had transmitted an observed set of transceiverprints. Hence, a

slightly different approach has been adopted to fulfill this objective.

7.2.1 Statistical Classifier

As with the previous form of classification, the Hotelling's T^2 is once again used for the verification of transceivers. However, this time, the T^2 statistic simply reflects the deviation of a transceiverprint from the centroid and covariance matrix in the target profile. In other words, the statistic represents the similarity of the transient to the corresponding transceiver profile, and is defined as:

$$T^2(FV_i) = (FV_i - \overline{FV})^T S^{-1} (FV_i - \overline{FV}) \tag{7.4}$$

In order to determine whether or not a given level of deviation is normal, the T^2 statistic is transformed to follow an F distribution by multiplying the statistic by the constant $n(n-p)/(p(n+1)(n-1))$, where n represents the sample size and p the number of features or variables. If the transformed value is greater than the F value of 2.20 (for 0.05 level of significance), the transceiverprint is classified as anomalous, i.e. it does not belong to the target transceiver.

7.2.2 Decision Filter

It is well known that intrusion prevention systems, e.g. network authentication, tend to make use of less dynamic data, e.g. passwords, for granting access to network resources and services. Thus, a single authentication event is sufficient for rending a binary decision. On the other hand, most anomaly-based IDSs, that rely on behavioral characteristics of users and/or devices, also render a decision based on the classification probability of a single observation or event.

While using behavioral-based characteristics does tend to lower the success rate of attacks, associated with impersonation, there are two problems associated with their use. First, it is difficult to identify *all* the unique behavioral patterns of a user or device for the purpose of profiling. Second, it is well known that behavior, in general, does change and with a frequency that is dictated by the behavior being

characterized. In addition, behavior is also influenced by environmental and other factors. The end result is that the use of single observation for rendering a normal vs. anomalous decision is not optimal.

In the case of transceiverprints, which are susceptible to environmental factors, such as noise and interference, the classification of a *set* of transceiverprints and the application of a decision filter, addresses the second problem. In other words, a final decision of normal or anomalous is rendered by the decision filter, based on the classification results obtained for a given set of transceiverprints. More specifically, the filter requires that 80% of the transceiverprints in the set be classified as normal. However, not only can this percentage be changed to reflect the requirements of the application, but other criteria, such as the average T^2 value of a set of transceiverprints, can also be used.

In terms of updating a profile based on currently observed behavior (first problem), this issue is addressed in section 6.5.

7.2.3 Details of Evaluation - WiFi/802.11 devices

The purpose of the evaluation is two-fold: 1) to primarily assess the composition of the transceiverprint, using static profiles, based on the False Alarm (FA)s and DRs (metrics) and 2) to determine the impact of profile updates, i.e. using dynamic profiles, on these metrics. However, a secondary objective is to determine the impact of the length of a transient on FAs and DRs. Thus, the use of transients with 1024 and 2048 samples was also explored.

In order to evaluate the use of RFF and static profiles (first objective), the following steps were carried out:

Step 1: For each transceiver being profiled, the transients of the captured signals were extracted using the approach presented in section 6.1. In turn, the transceiverprints were extracted from the transients. Once the outliers were identified and excluded, the remaining transceiverprints were used for profiling and evaluation purposes.

Step 2: A subset (approximately 35-40) of the transceiverprints was selected using

K-means clustering and subsequently used to define the elements of a profile, including elements E7 and E8 (although not used with static profiles). The remaining transceiverprints (60) were used for testing/evaluation purposes.

Step 3a: The actual evaluation exercise was carried out by: selecting a transceiver to be tested (from a list); obtaining the first set $(1, 2, \ldots, 10)$ of chronologically ordered test signals (for that transceiver); extracting the transceiverprints from each signal; and classifying or matching each test transceiverprint with the corresponding transceiver profile (for detecting false alarms) and with the remaining 29 transceiver profiles (for detecting intrusions). The decision filter is then applied to the classification results of the transceiverprints to determine whether the *set* is normal or anomalous. If less than 8 test transceiverprints (80%) of transceiver X match the profile of X, the set is considered anomalous (a false alarm) thus increasing the FAR by one. On the other hand, if less than 8 test transceiverprints of transceiver X match the profile of transceiver Y, an intrusion against Y is suspected and the DR rate of Y is increased by one. The evaluation was repeated 50 times using the next set or window of transceiverprints (e.g. $2, 3, \ldots, 11$) that is continuously shifted by one.

The use of test transceiverprints, in an overlapping fashion, had been adopted to simulate the initial capture and subsequent classification of a set of transceiverprints, which start at different points in time. This strategy has also been adopted by Lane for the purpose of creating mobility patterns, using the IBL technique [104]. Nevertheless, we acknowledge the lack of independence of test samples, i.e. some transceiverprints used in multiple classification events, required for statistical results. In the future, we will make use of non-overlapping windows and a larger set of signals to further assess the performance of the classification system.

In order to fulfill the second objective, i.e. using a dynamic profile, the previous procedure, in particular step 3a, was modified as follows.

Step 3b: Once a set of transceiverprints has been classified as normal, the profile of the target transceiver is updated according to the procedure presented in section 6.5. The updated profile is then used for the next iteration of the evaluation. All other details remain unchanged.

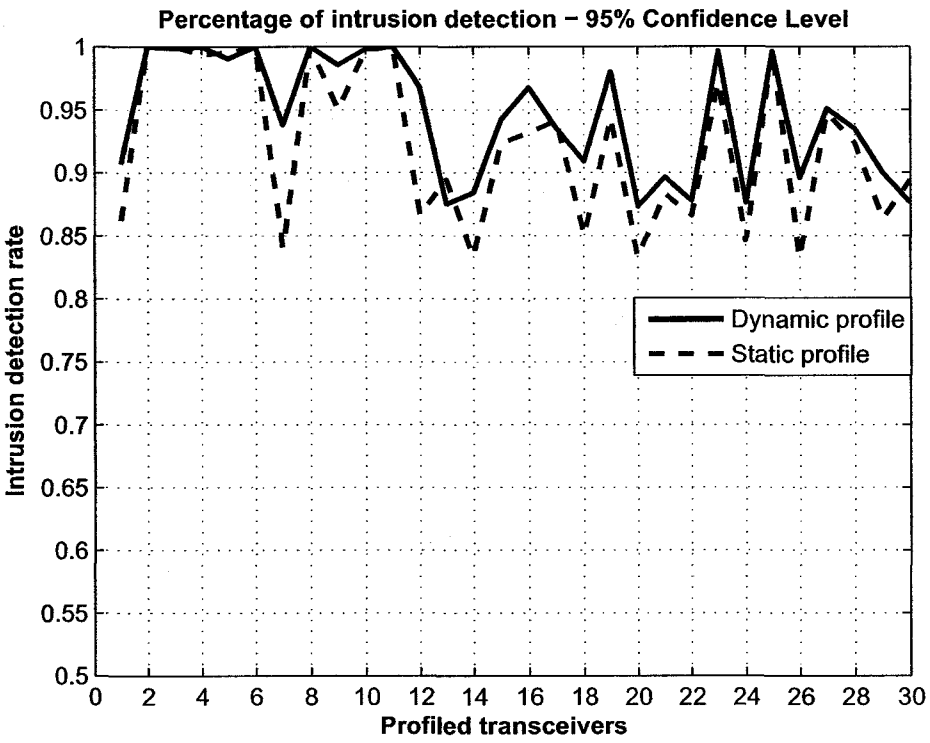

Figure 7.3: Intrusion Detection Rate (1024 samples)

As far as the infrastructure is concerned, a similar setup, see Fig. 6.14, was used for the data capture and evaluation exercises. However, approximately 120 signals from each of the 30 802.11b transceivers were captured for the purpose of RFF.

7.2.4 Evaluation Results - RFF and T^2 Hotelling Statistic

After executing the evaluation procedure 50 times, using 50 sets of 10 transceiverprints each, the following FAs and DRs were obtained for each of the 30 profiled transceivers. FAR for a given transceiver, e.g. 665, is defined as (the number of anomalous test transceiverprints (sets) / total number of test transceiverprints (sets)) of transceiver 665. On the other hand, the DR for 665 is defined as the (number of anomalous test transceiverprints (sets) / total number of test transceiverprints (sets) of the remaining 29 transceivers).

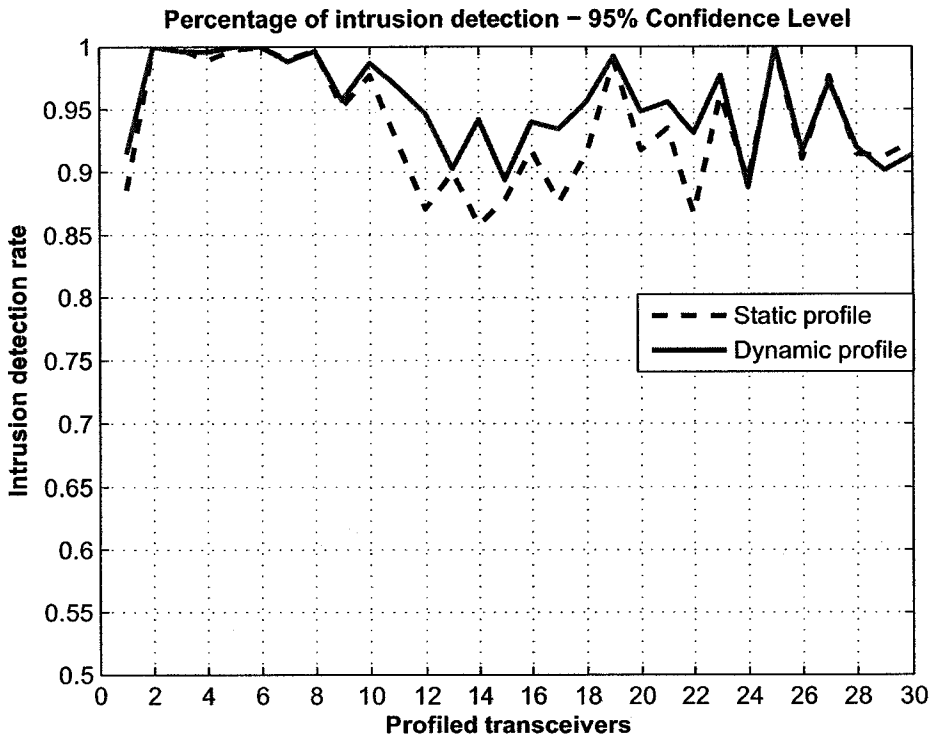

Figure 7.4: Intrusion Detection Rate (2048 samples)

False Alarm Rate

Using transients with 1024 or 2048 samples

The FAR of all profiled transceivers is 0%. Most importantly, this rate illustrates the feasibility of accurately characterizing the behavior of the transceivers. Moreover, this rate is obtained when using both static and dynamic profiles. When a static profile is used, the FAR provides an indication as to the accuracy with which the set of transceiverprints (E8) has been selected, using K-means clustering, for profiling purposes. As this profile is updated in a dynamic manner, the use of the upper/lower ED thresholds (E4 and E5) and intra-transceiver variability (E7), permits the general characteristics of a transceiver to be preserved, without introducing abnormal behavior, e.g. outliers.

Detection Rate

Using transients with 1024 samples

The DR, associated with the use of static profiles, is typically lower with the mean and standard deviation of 92.27% and 6.27% respectively, see Fig. 7.3. One of the key factors responsible for the lower DR is the use of a static covariance matrix. As previously discussed, this matrix represents the variability of each of the features in E8 with respect to one another. However, the selected transceiverprints in E8 may not reflect the full range of variability of the corresponding transceiver. Thus, it is possible for a transceiverprint from transceiver Y to be mistakenly classified as X, thus lowering the DR of transceiver X. Although the centroid is also used in the classification/detection process, its contribution is not as significant.

This situation is remedied, to some extent, by continuously updating a given profile, i.e. recalculating the centroid and covariance matrix to reflect currently observed behavior (transceiverprints). Hence, as the covariance matrix is altered, it begins to reflect the true behavior of a transceiver, a critical element for distinguishing between transceivers from the same manufacturer and production line. As a result, the mean DR (94.5%) is increased, while the standard deviation (4.91%) is decreased, thus supporting the use of dynamic profiles.

Using transients with 2048 samples

As indicated by Fig. 7.4, the statistics, associated with the use of static (mean: 93.74% and std: 4.81%) and dynamic profiles (mean: 95.46% and std: 3.67%) are marginally improved. Due to the increased number of samples in the transients, the intra-variability of the DWT-related features is further reduced, resulting in a higher DR.

In order to make a proper comparison of the evaluation results, an identical set of transceiverprints (E8) was used for both types of profiles. However, the selection of a more optimal set of transceiverprints can further improve the DRs. Nevertheless, the average rate of 94.5% is encouraging. It justifies the use of multiple features for distinguishing between transceivers from the same manufacturer.

Although it would prove beneficial to compare FAs and DRs, with those obtained by other similar research initiatives, one must take the following factors into consideration. First, although transceiver *identification* is similar to transceiver *verification*, the performance metrics of the underlying classification systems are different. For example, in the case of transceiver verification, the use of FAR and DR is common. On the other hand, the mean, standard deviation and percent of correct classification, are often associated with transceiver identification. In the past, most research teams have focussed on transceiver identification in order to detect unauthorized use of the electromagnetic spectrum.

Second, it is uncommon for statistical classifiers, which are not categorized as neural networks or self-learning systems (e.g. SOMs), to be used for RFF. Classifiers, which make use of distributions, such as the F-distribution, are typically used for detecting variances in the quality of a given process or product.

Lastly, a comparison between a statistical classifier and a variant of neural networks, e.g. ANN, would equate to a comparison between apples and oranges. The performance of each system is dictated by the selection of various parameters. Additionally, the use of PNN, a statistical classifier, for determining FAR and DR is also inappropriate, due to their intended behavior.

Nevertheless, the type of research carried out by Ye *et al.* [194] is similar to some degree. The authors make use of two MSPC techniques, namely T^2 and EWMA for ABID on a host machine. Whereas T^2 is used for creating a profile of normal events and determining a classification threshold ($\bar{T}^2 + 3s_{T^2}$), EWMA is used for incorporating the time characteristics into each of the observed events to be classified. In other words, the values, associated with each event, are decayed exponentially with the passage of time.

In comparison to our work, there are a few dissimilarities. First, the ABID exercise is carried out off line. As the IDS makes use of audit data, that is collected over a period of time, the use of the EWMA filter is appropriate. Second, neither the centroid (sample mean $\bar{X}$) nor the covariance matrix is updated using the normal test data. Thus, the profile remains static throughout the testing period. Third, the

key objective of the authors is to differentiate between normal and intrusive events that exhibit significant differences in their respective mean and standard deviation. All of these factors combined permit the system to distinguish between attack and normal sessions (one correct classification per session) with a false-alarm rate of 2.13% and DR of 100%.

7.2.5 Details of Evaluation - Bluetooth devices

As with WiFi/802.11 devices, the purpose of the evaluation exercise is to assess the composition of the transceiverprint based on the classification success rate. The following steps were carried out using a set of signals captured from BT transceivers:

For each transceiver being profiled, the aforementioned features were extracted from the transients. Once the outliers (approximately 5-10) were removed, a subset (approximately 30-40) of the transceiverprints was selected using k-means clustering and subsequently used to calculate a centroid and covariance matrix. The remaining transceiverprints (50) were used for testing purposes. In order to evaluate the profiling and classification aspects of the proposed technique, signals from each of the 10 BT transceivers (3COM-4, Ericsson-4, Test Radios-2) were captured. All subsequent processing and evaluations were carried out using the Matlab software and associated tools.

7.2.6 Evaluation Results - RFF and Statistical Classifier

The FAR and DR served as our primary metrics. Additionally, 40 iterations were used for the purpose of assessing these metrics.

False Alarm Rate

All of the 10 transceivers, with the exception of E4 (five percent), have a FAR of zero percent. Unlike the high intra-transceiver variability of E4, the others have a low to moderate level of variability. This characteristic permits the k-means clustering algorithm to select a set of transceiverprints, which accurately characterizes

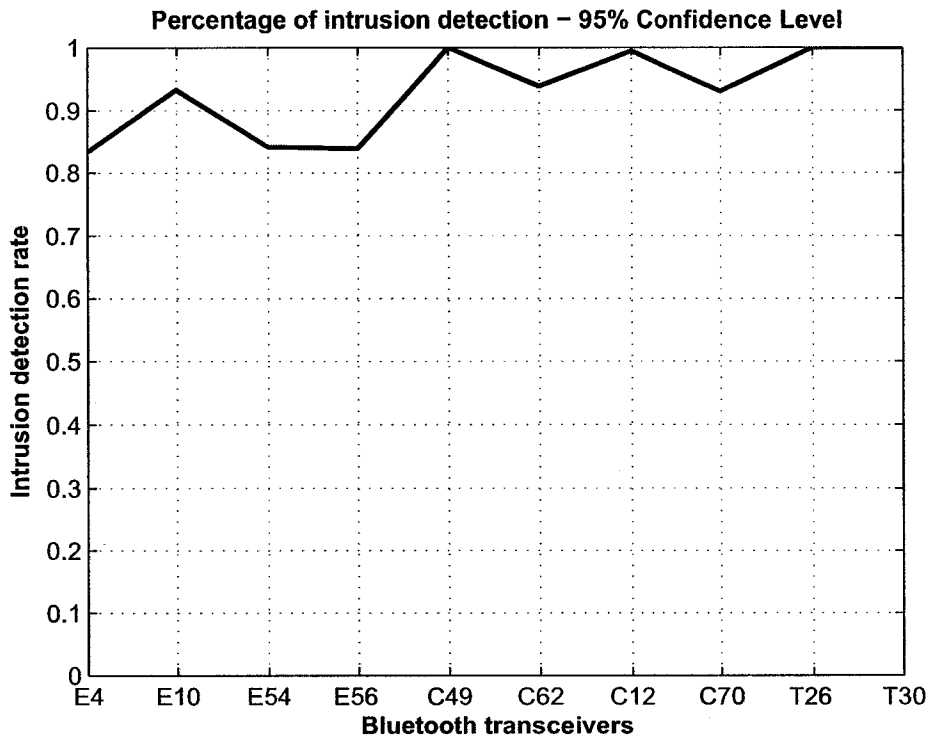

Figure 7.5: Intrusion Detection Rate (2048 samples)

a transceiver via the centroid and covariance matrix. Finally, the overall mean and standard deviation are five percent and 0.025 respectively.

Detection Rate

Figure 7.5 illustrates the DR for the profiled transceivers. The x-axis represents the transceivers, which are identified using the term $M\#$. Whereas M represents the manufacturer, e.g. E=Ericsson, C=3Com and T=Test Radio, # is the identifier of the transceiver.

There are a few observations that are noteworthy. First, the overall mean of ninety-three percent and standard deviation of seven suggest the presence of inter-transceiver variability between the 10 BT transceivers. Second, the average DR for the test radios is the highest (100%), followed by those from 3Com (90%) and Ericsson (87%). These results support the different levels of both inter-transceiver and intra-transceiver variability. In particular, it is interesting to note that E4 has one of the

lowest DR as a result of its large intra-transceiver variability. Finally, although not depicted in the figure, there is a degree of similarity between the transceivers, i.e. signals, from the same manufacturer.

7.2.7 Memory Requirement and Running-Time Complexity

Given that the same transceiver profiles are being used for the purpose of transceiver verification, the memory requirements remain unchanged. Please refer to section 7.1.4 for a detailed discussion.

Running-time Complexity

Unlike the classification algorithm, associated with the identification of transceivers, Algorithm 2 is relatively less complex. Although the same number of feature vectors fv_n are used, they are only compared to the profile of the target transceiver.

1. For each feature vector fv_n, the $T_{squared}$ value, associated with the target transceiver, is determined. Moreover, this value is transformed to an equivalent value $fValue$ in the F distribution. The values s and p denote the number of samples, i.e. transceivers, and features respectively.

2. If $fValue$ is less than or equal to the pre-established threshold, for 0.05 level of significance, the feature vector is considered normal and the *count* of normal vectors is incremented by one.

3. If *count* is greater than or equal to the *decisionFilter*, then the set of feature vectors is considered normal.

Proceeding with the analysis of the aforementioned algorithm, the number of operations required, for determining the $fValue$ for each feature vector fv_n and maintaining a running count (lines 3-9), is $n(p^2 + 3) = np^2 + 3n$. Once again, the term p^2 is associated with the calculation of the $T_{squared}$ value.

Algorithm 2 Transceiver Verification

1: $count \leftarrow 0$; $decisionFilter \leftarrow 8$; $F_{thresholds} \leftarrow 2.20$; $s \leftarrow 30$; $p \leftarrow 15$ {Determine T^2 and convert it to F distribution for each feature vector}
2: **for** $n = 1toN$ **do**
3: $T_{squared} \leftarrow (fv_n - FV)' \times S^{-1} \times (fv_n - FV)$
4: $fValue \leftarrow T_{squared} \times s(s-p)/(p(s+1)(s-1))$
5: **if** $fValue \leq F_{threshold}$ **then**
6: $count \leftarrow count + 1$
7: **end if**
8: **end for**
{Determine status of feature vectors based on value of count}
9: **if** $count \geq decisionFilter$ **then**
10: $status \leftarrow NORMAL$
11: **else**
12: $status \leftarrow ANOMALOUS$
13: **end if**
14: **return** *status*

Part V

Approach: ABID using user-based profiles

Chapter 8

User Mobility Patterns

In the past, the mobility patterns of users have been used to address the inefficiencies of location-area based update schemes, e.g. Wong [189] and Ma [109], and to enhance routing in wireless mobile ad-hoc networks, e.g. Wu [190].

Regardless of the underlying algorithms and their implementation, the success rate is based on the consistency of a user's mobility patterns. Therefore, it stands to reason that the success rate will be less than optimal, for users with chaotic behavior.

8.1 Related Work

As aforementioned, their use in ABID has been investigated by Spencer [163]. Moreover, in the cellular network domain, the incorporation of user profiles into an ABID system has been evaluated by Samfat and Molva [149] (GSM-A02/A03-C01) as well as by Sun and Yu [167] (GSM-A02/A03-C03). Samfat and Molva have also studied the use of usage patterns (GSM-A02/A03-C01) in anomaly detection. The key novelty and weaknesses of these two solutions are identified next.

The key novelty of the approach, adopted by Samfat and Molva, is the ability of the distributed IDS to detect mobile intruders, in the visitor location and in *real-time*, i.e. within the duration of a typical call. Hence, all algorithms have been optimized to support this goal. As far as usage profiles are concerned, the feature set consists of call and session vectors. Moreover, concept drift is taken into consideration by

continuously updating both profiles of the users. In terms of weaknesses, the accuracy of the mobile station traffic generator is questionable, even if the behavior of a user is modeled using the exponential (duration of calls) and poisson distributions (arrival of calls).

In contrast to the previous approach, Sun and Yu propose an *on-line* anomaly detection algorithm. The key distinguishing characteristic is the use of a high-order Markov model [143] for predicting the next cell and its corresponding probabilities. However, as with Samfat and Molva, the authors do take into consideration the need to address concept drift. In this case, the Exponentially Weighted Moving Average is used to update the mobility trie. The first key weakness is the fact that profiling and classification is carried out using a group of users with consistent itineraries. Although evaluation results do confirm the above-average performance of the proposed detection algorithm, a more accurate assessment can be made by using different groups of users, as with the first approach. Second, the authors do not specifically address the space and time complexity aspects of this approach.

Although different profiling and classification mechanisms have been used in the previous approaches, the common goal is to provide anomaly-based detection. In addition, since they specifically address *phone* theft, it is not surprising that the underlying implementation strategies leverage the existing infrastructure of cellular networks. In fact, the use of cells, as the primary feature in mobility profiles, is to be expected. Last, but not least, a common *limitation* is the use of simulated data for both profiling and classification purposes.

Finally, commercial systems, namely the Fraud Management System by Hewlett-Packard (FMS-HP) [80] and Compaq (FMS-C) [34] also make use of service usage profiles. They are built using calling patterns, call frequency, call times and duration, wireless home/roaming behavior and other call-related information. Although both FMSs offer some services, which permit them to be differentiated, they both detect multiple types of fraud, by examining all calls (e.g. streams of call detail records used for billing purposes) and other-related events (event records).

8.2 Key Requirements

In order to develop a solution, which is both robust (i.e. resilient against various attacks) and practical, the following requirements must be taken into consideration:

Generic Solution A solution that can be readily integrated into various wireless networks, e.g. WLANs, would prove more useful, and thus would command a higher value.

Characteristics of System An adequate level of system performance, high scalability ratio, and semi-autonomous operations represent the key characteristics that must be exhibited by the proposed IDS. In terms of system performance, both space and time complexity of the system should be minimized in order to permit on-line detection of intruders. Likewise, it is critical that the system be capable of accommodating a significant number of users. With respect to semi-autonomous operations, the key processes of the system should be automated to the greatest extent possible. This would alleviate some of the responsibilities from systems' administrators.

Privacy of users Last, but not least, the fundamental requirement for user privacy cannot be overstated. Thus, various data transformation strategies, such as the use of aliases and hash codes, should be explored in order to achieve the stated goals without compromising the privacy of users. The key factors, in instilling confidence in users, are to solicit their approval, via service agreement or optional service, and to repatriate control, of their sensitive information, back to the users.

8.3 Description of Solution

In this section, a brief overview of the use of mobility patterns for ABID is presented. In addition to the nuances, associated with the profiling and classification activities, other key characteristics are as follows:

Profiling A profile for each user is created using unique sequences (feature) of LCs. These geographical coordinates are obtained from Location Broadcast (LB)s, which are transmitted by users on a voluntary basis.

Classification The classification of an observed sequence of LCs is carried out using the IBL system [10], a general class of machine learning techniques. Furthermore, the thresholds, used for classification purposes, are based on the mobility characteristics of each individual user. While the accuracy, with which the mobility behavior of users is characterized, does dictate the classification performance of the system, other parameters, namely the SL and PL, also influence FAs and DRs (defined in the sequel).

Evaluation Finally, evaluation exercises are based on LBs from users, who make use of public transportation, e.g. bus, in the area of Los Angeles. The high density of these users increases the probability of intrusions, a necessary prerequisite for evaluating the performance of the system.

8.3.1 Framework

This section provides an overview of the proposed ABID system. As with most IDSs, the two key objectives are to define user mobility-based profiles, and to design an appropriate classification system.

Details of the framework, which has been used for the implementation of the ABID system, are illustrated in Fig. 8.1. It is important to note that the detection process, as described in the sequel, is applied to each authorized user. Moreover, during the profiling phase, the subset of the activities, from data collection to the definition of mobility profiles, is typically carried out on a one-time basis and prior to classification. However, in order to address the issue of concept drift, i.e. the inherent tendency for users' mobility patterns to change with time, it is essential that the profiles be updated periodically. The use of EWMA would prove useful in fulfilling this requirement.

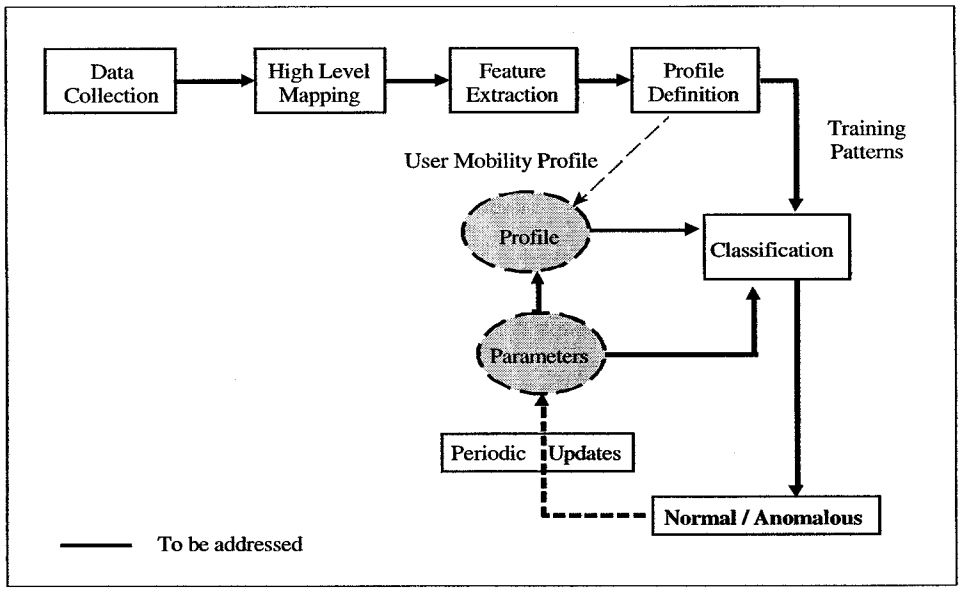

Figure 8.1: ABID using Mobility Patterns

The intrusion detection process begins with the data collection exercise (see section 10.3.1). Once the LBs, which contain LCs and other data, have been captured for a period of 3-6 months, a high-level mapping (HLM) is applied. The objective of the HLM is to decrease the precision of the LCs in order to increase the similarity between a pair of LCs. Upon completion of this phase, the LCs are extracted from each broadcast during feature extraction. A set (defined by SL) of these chronologically-ordered LCs are subsequently concatenated to define a mobility sequence. This process continues until all the mobility sequences (referred to as a data set) have been created. The unique sequences (training patterns), from the first four of the six partitions of the data set, are stored in the UMP, along with other user-related information. During the classification phase, an observed set of mobility sequences of a user is compared to the training patterns in his/her profile. If the Noise suppressed Similarity Measure to Profile (NSMP) value falls within the pre-established thresholds, the mobility sequences are considered normal, otherwise a flag is raised.

Chapter 9

Profiling Phase

As with device-based profiling, a profile for each user is created during this phase. Although the profiling exercise itself is relatively short in duration, as aforementioned, it could take several months to collect a sufficient amount of mobility data.

An interesting aspect of the profiling process, which is rather uncommon, is the use of a classifier for establishing user-based thresholds, as discussed in section 10.1.

9.1 High-Level Mapping

The term *intra-user* variability refers to the difference between the LCs (j represents the latitude and i represents the longitude) that are transmitted by user A as he/she travels using routes one (solid line) and two (dashed line), see Fig. 9.1. So, for example, if a LC, in the area of $(j+5, i+4)$, is sent while on routes one and two, these coordinates could potentially be different. Let us assume that the sequence of 10 LCs, associated with route one, has been captured and stored in the profile (as a training pattern) of user A. If the sequence of LCs, corresponding to route two, is compared to this training pattern, it would result in a similarity value of zero.

Therefore, one strategy, for reducing this type of variability, is to define different degrees of similarity that are based on the PL of LCs, see Table 9.1. As indicated, the HLM transforms the original LCs according to the PL. Consequently, the similarity between the 10 corresponding LCS in the two sequences (from routes one and two)

PL	LC from APRS	HLM LC
3	33.14623,114.26874	33.10,114.25
2	33.14623,114.26874	33.1,114.2
1	33.14623,114.26874	33,114

Table 9.1: HLM using different PLs

is increased.

This mapping process, which is applied to the LC in each LB, is carried out as follows. The original format of the LC is (###.#####) and (###.#####), where the first and second terms (###.#####) represent the latitude and longitude respectively. Based on the PL, the LC is truncated and rounded to the specified number of digits after the decimal point. For example, with level three (highest precision), the specified digit of the first and second terms (###.##) is rounded to 0 if it is within 0-4 and to 5 if it is within 5-9 range. Thus, for example, the LC 33.14623,114.26874 is mapped to 33.10,114.25. Similarly, the HLM for levels two and one are (###.#) and (###.0) respectively. The choice of PL is explored in Section 10.2.

Caution must nevertheless be exercised, since minimizing intra-user variability will also minimize *inter-user* variability (difference between LCs from different users). As depicted in Fig. 9.1, the same logic would apply to intruders as well, resulting in a potentially successful impersonation attempt.

As a rule of thumb, inter-user variability must be maximized in order to correctly distinguish between legitimate users and intruders.

9.2 Feature Extraction

The extraction of high-level mapped LCs (feature), from LBs, results in a HLM data stream that is used for creating mobility sequences. A mobility sequence is defined as a chronologically-based sequence of LCs. The selection of the appropriate SL is also addressed in Section 10.2.

Using the data stream of LCs, the feature extraction process concatenates the first

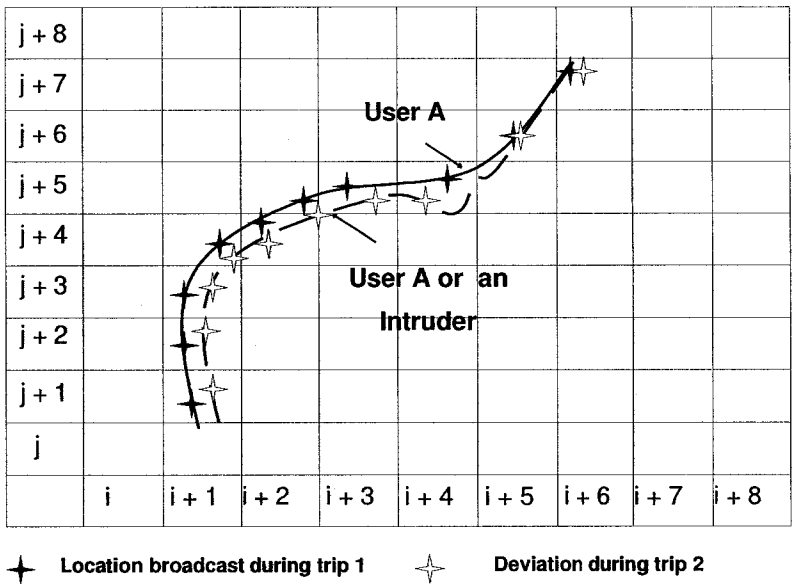

Figure 9.1: Intra-user and inter-user variability

set (e.g. ten) of LCs into a single sequence. The next sequence is created by using the LCs at the $i+1$ to $j+1$ indexes of the HLM data stream, where i (LC1) and j (LC10) represent the first and last LC of the first sequence. Hence, each subsequent sequence is obtained by incrementing (by one) the indexes of the previous sequence, as suggested by Lane and Brodlay [104]. The purpose of using an overlapping window (shifted by one) is to accommodate different sequences that start at different LCs. In other words, it permits each LC to become a starting point of a sequence.

This process is repeated until all the LCs in the data stream have been exhausted. The resulting set of sequences, henceforth referred to as original sequences, serves as input to the profiling and classification phases.

The use of a single feature (i.e. sequence of LCs) is intentional. One of the key objectives is to determine the maximum success rate possible using mobility sequences. Additional features, such as timeframe, will be investigated in the future for maximizing *inter-user* variability.

Element	Description
E1	Identifier
E2	Training Patterns
E3	Window Size
E4	Minimum Threshold
E5	Maximum Threshold

Table 9.2: Elements in a profile

9.3 Profile Definition

Once the mobility sequences have been obtained, the next step is to create UMPs. A detailed description of each element in a UMP, see Table 9.2, is provided in the sequel. Although, not explicitly stated, all of the elements are used for classification purposes.

Identifier Otherwise, known as call sign, it represents the unique identification of the user, which has been issued by Industry Canada. It is transmitted with all LBs.

Training Patterns As aforementioned, these unique mobility sequences characterize the mobility behavior of a user.

Window Size Due to factors, such as traffic and weather, a mobility sequence of a user may deviate from the norm. This deviation is referred to as noise, which must be minimized. *Window size* refers to the number of consecutive mobility sequences to be used for obtaining the average or NSMP value. The benefits of noise reduction, not withstanding, the use of this parameter also results in a proportional time delay (corresponding to the number of LCs required) before a detection verdict can be rendered. Although the window size is identical for all users (in this iteration), this parameter can be customized to reflect the level of noise, within the mobility patterns of a given user. One possible strategy, for identifying the level of noise or intra-user variability, is to determine the number and frequency of unique mobility sequences in the training data. Whereas a

small number of unique sequences, with high frequencies, supports the notion of consistent behavior, the inverse exemplifies a more chaotic behavior.

Minimum/Maximum Thresholds Whether or not an observed set of mobility sequences reflect normal behavior is dictated by the minimum and maximum thresholds. If the NSMP value, of these sequences, falls within the thresholds, it is considered normal, otherwise, a potential intrusion is suspected. The values of the thresholds are determined by obtaining a distribution of the NSMP values, which are calculated using the training patterns and parameter sequences (5^{th} partition of the data set), and by applying the desired level of false alarms γ (application-dependent).

As previously stated, a mobility profile is created, for each user, prior to classification. However, in order to address the issue of concept drift, it is essential that these profiles be updated periodically. One approach is to maintain a window of training patterns that is continuously shifted in time, as new patterns are added (analogous to the use of EWMA). As the window is shifted, the minimum and maximum thresholds, are updated accordingly. This should not only reduce the rate of false alarms, but also maintain a given level of performance (currently being investigated).

Chapter 10

Classification Phase

The final step, in anomaly-based detection, is the classification of an observed behavior as normal or anomalous.

As indicated in section 10.3, an observed set of mobility sequences of a user is compared to a set of training patterns in his/her profile. For each mobility sequence being compared to the training patterns, the maximum similarity value (discussed in the sequel) is obtained. If the average of these values falls within the pre-established thresholds, then the user is considered legitimate, otherwise a flag is raised.

10.1 Instance-Based Learning Classifier

This section provides a brief overview of the key concepts, which are defined in IBL and enumerated in Table 10.1. Readers are encouraged to consult the paper by Lane and Brodlay [104] for a more detailed discussion of the underlying framework.

Similarity Measure

As you may recall, a mobility sequence is defined as a chronologically ordered set of LCs, and that these sequences are used for training, establishment of parameters and test/evaluation (final partition of the data set) purposes.

Therefore, the *similarity* of two sequences X (from the set of test sequences) and

Concepts	Description
Similarity Measure (SM)	Similarity between a test sequence and training pattern
Similarity Measure to Profile (SMP)	SM between a test sequence and all training patterns
Noise Suppression	Average SMP for 10 consecutive test sequences
Decision Rule	Classification of a test sequence as normal or anomalous

Table 10.1: Key concepts associated with IBL classification

Y (from the set of training patterns) of equal length l is defined as follows:

$$sim(X,Y) = \sum_{i=0}^{l-1} w(X,Y,i)$$

with:

$$w(X,Y,i) = \begin{cases} 0 & \text{if } i < 0 \text{ or } x_i \neq y_i \\ 1 + w(X,Y,i-1) & \text{if } x_i = y_i \end{cases}$$

where i represents the index of a LC in a sequence. Thus $w(X,Y,i)$ equals zero if the LCs of the X and Y sequences at index i are not identical. Otherwise, a value of one is added to the outcome of $w(X,Y,i)$ at $i-1$, see Fig. 10.1.

Similarity Measure to Profile

Whereas a SM is determined based on a one to one comparison of the LCs between a test sequence and training pattern, a SMP is calculated by performing a one to many comparison of an observed test sequence X, with *all* the training patterns in a profile D. It is defined as:

$$sim_D(X) = max_{Y \in D} sim(Y,X).$$

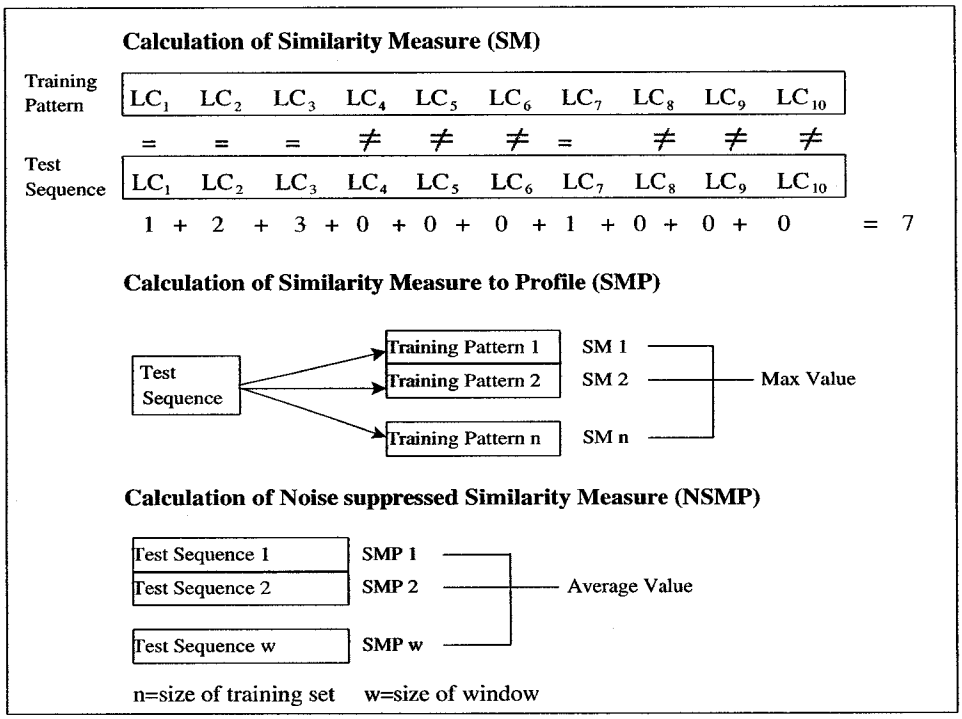

Figure 10.1: Key concepts in IBL

The maximum value of $sim_D(X)$, for a given SL, is:

$$\sum_{i=1}^{l} i = \frac{l(l+1)}{2}.$$

Thus, the SMP is the maximum of the SM values.

Noise Suppression

In section 9.2, the concept of using each LC (from a long stream of LCs), as a starting point i of a sequence of length l, was introduced. This form of segmentation results in a set of sequences (original sequences), whereby a sequence, starting at i, is called the i-th sequence.

As with all chaotic systems, noise is inherent and it reflects the deviation of a mobility sequence from the patterns in the profile. A degree of *intra-user* variability is to be expected, since it is a function of many factors, including traffic conditions and weather. Nevertheless, noise can be suppressed by calculating the average SMP of a *set* of W test sequences, where W is established based on application requirements.

Thus, the average SMP, over a window of length W, is defined as:

$$v_D(i) = \frac{1}{W} \sum_{j=i-W+1}^{i} sim_D(j).$$

where *jandi* are the start and end test sequences respectively. The term $v_D(i)$ is thus referred to as the *NSMP value.*

Decision Rule

Whether or not a given set of observed sequences exhibit normal mobility behavior can be determined by comparing the resulting NSMP value to the pre-established minimum t_{min} and maximum t_{max} thresholds. While t_{min} is used to detect sequences, which have low NSMP values, t_{max} proves useful in detecting sequences that have unusually high similarity to the profiled behavior, perhaps an indication of a replay attack.

The calculation of t_{min} and t_{max}, for each user, is carried out by specifying an acceptable level of false alarms γ to a Normalized Probability Distribution (NPD) of NSMP values. Thus, t_{min} and t_{max} are dependent on γ and NPD.

The parameter γ dictates the width of the acceptance region (between t_{min} and t_{max}) on the x-axis, see Fig. 10.2. Consequently, it represents a trade-off between FAs and DRs. Hence, a smaller value of γ corresponds to a wider acceptance range. As a result, the rate of false alarms is decreased. However, the expanded region also causes the DR to decrease.

As far as the NPD is concerned, it is obtained by using the parameter sequences and the training patterns, obtaining a distribution/histogram of NSMP values (in the range of $0, \ldots, l(l+1)/2$), and normalizing this distribution based on the probability of each NSMP value.

Finally, t_{max} and t_{min} are established using $\gamma/2$ quantiles (upper and lower) of the NPD, as proposed by Lane and Brodlay. The number of sequences and the actual sequences (training vs parameter) used for the calculation of the NPD are important factors to be considered. As far as the number of sequences are concerned, it is

dependent on the intra-variability of the original LCs and the PL used to minimize this variability. Using a low PL results in sequences, more specifically LCs, being more similar, and thus, reduces the number of sequences in the training set. The number of sequences allocated to the parameter set is not as significant, so long as the sequences fully reflect the mobility behavior of a user.

As to which sequences, from the initial set of sequences, should be used for training, parameter and test/evaluation data represents a more challenging problem. One option, which has been implemented in this iteration, is to divide the initial set of sequences into partitions of 4/1/1. The first 4/6 of the sequences are allocated for training, followed by 1/6 for establishing parameters (e.g. thresholds), and the last 1/6 for testing purposes. By allocating the first and the largest set of sequences for training, the probability of accurately characterizing the mobility behavior of a user is increased. This is, of course, based on the assumption that the mobility patterns of a user is typically established within a given timeframe. The shape of the NPD reflects the accuracy with which the mobility behavior of a user has been characterized. Regardless of the allocation strategy, it is essential that the profiles of users (i.e. the set of training patterns) be updated periodically. Moreover, a replacement strategy, which favors the most recent patterns, should be employed in order to limit the storage space.

Fig. 10.2 illustrates the application of $\gamma = 0.05$ to the NPD of user 19, who was selected at random. In this figure, the x-axis represents the spectrum $(0, \ldots, l(l+1)/2)$ of the similarity values that are possible for a sequence of length 10. Please note that the actual values are in the range of $(1, \ldots, l(l+1)/2) + 1$ for improved graphical representation. The y-axis represents the probability of each NSMP value in the NPD. Both the minimum and the maximum thresholds are indicated using vertical lines. What is illustrated in the figure is the width of the acceptance region, which is a function of NPD and γ. The narrow acceptance region, located at the higher end of the spectrum, is desirable. In particular, the location of the minimum and maximum thresholds at NSMP values of 38 and 56 respectively, reflects the high level of consistency between the training patterns and parameter sequences. In other

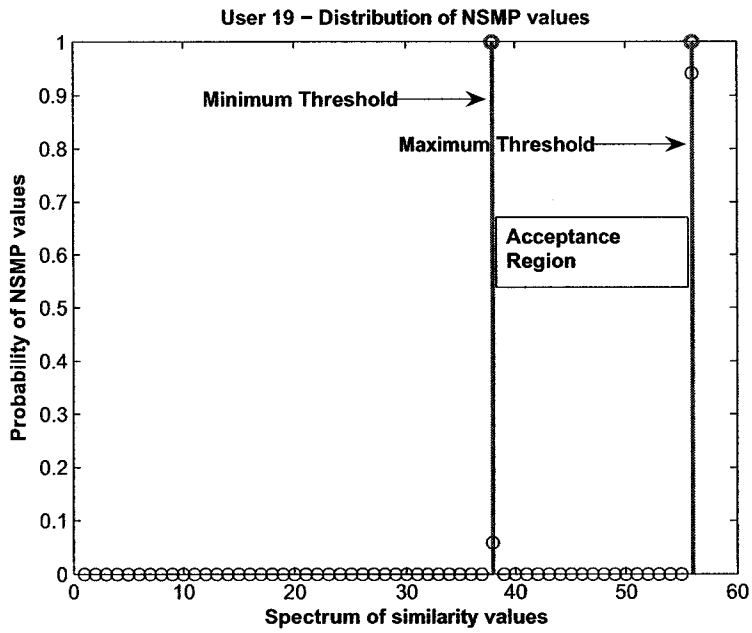

Figure 10.2: Minimum/Maximum thresholds

words, the mobility behavior of user 19 is fairly consistent during this time period (i.e. when LCs for training and parameter sets were acquired). This characteristic should support a high DR. Furthermore, if this behavior remains consistent (i.e. high similarity between training patterns and testing sequences), then a low FAR can also be expected.

10.2 Empirical Analysis of System Parameters

In the previous sections on HLM and feature extraction, we had indicated that the PL and SL are of significance and that an appropriate value had to be selected.

Aside from stating the obvious, our first objective is to determine the impact of these parameters on the characterization of users (distribution of the NSMP values) and intrusions (successful impersonation attempts against a user). We address the impact of these parameters on FAs and DRs in the section on evaluation.

Given that the mobility behavior of the 50 profiled users does differ to some extent, and that this variability is likely to influence the analysis of both parameters, we have categorized these users based on the consistency with which the training patterns

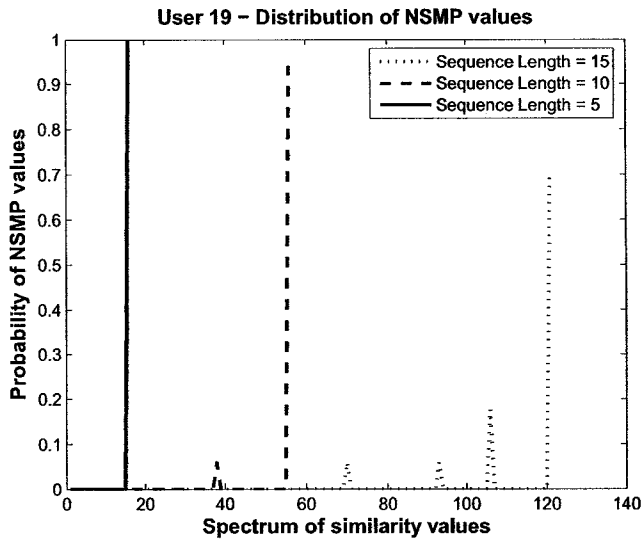

Figure 10.3: Characterization using different sequence lengths

are being followed (repetitions). The three classes are defined as follows. Whereas class one represents users who exhibit consistent behavior, class two and three are associated with those with progressively more chaotic behavior. We focus on the results obtained for user 19 (class 1 with 40% of users) as they illustrate the expected behavior, associated with an adequate level of characterization. Nevertheless, we briefly comment on results obtained for user 23 (class 2 with 56%) and user 41 (class 3 with 4%).

10.2.1 Sequence Length

Fig. 10.3 illustrates the use of three different lengths (5,10,15) for the mobility sequences and their impact on the characterization of user 19. Values of NSMP, which are located at the lower-end of the SM spectrum, are vulnerable to the choice of γ. Since γ dictates the width of the acceptance region, in particular the minimum threshold, all values of NSMP that are less than the threshold are treated as FAs.

Other parameters used include the window size of 100, PL of one (PL1), and minimum threshold of two. The maximum threshold, however, was based on the SL.

In Fig. 10.3, the x-axis represents the spectrum of similarity values for all three SLs. Since the results, associated with each length, have been incorporated into one

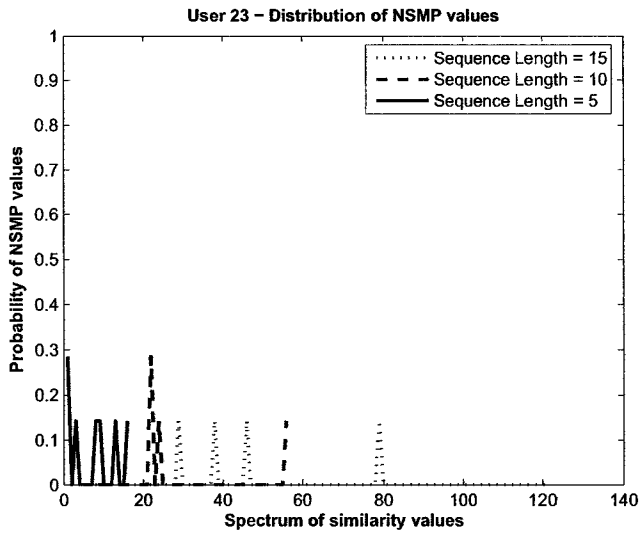

Figure 10.4: Characterization using different sequence lengths

plot, the range of the x-axis is actually from 1 to 121 (for SL15). In other words, results obtained for SL5 (length of five) are localized towards the lower end of the spectrum. As far as the normalized NSMP values are concerned, they are indicated via the y-axis.

What is being illustrated is as follows: as the SL is increased, the percentage of NSMP values, located at the higher-end of the SM spectrum starts to decrease. In this case, the NSMP values are located precisely at 15, 55 and 120 on the x-axis. Furthermore, as the percentage of these values decreases, they are distributed towards the other (i.e. lower) end of the spectrum. This behavior is logical since the probability of achieving a high NSMP value decreases as the SL is increased. Therefore, should the NPD of a user be localized towards the higher end of the spectrum, selecting a larger SL is not prudent, since it shifts the NPD further towards the left. However, if the NPD is located at the lower end of the spectrum, see Fig. 10.5, it is advantageous to use a larger SL, as it causes the NPD to shift towards the higher end of the spectrum. On the other hand, when the NPD is distributed between the lowest and highest similarity values, see Fig. 10.4, a larger SL is also desirable for shifting the NPD towards the center, and away from the lower end of the spectrum.

We continue our analysis of the impact of SL on the distribution of potential

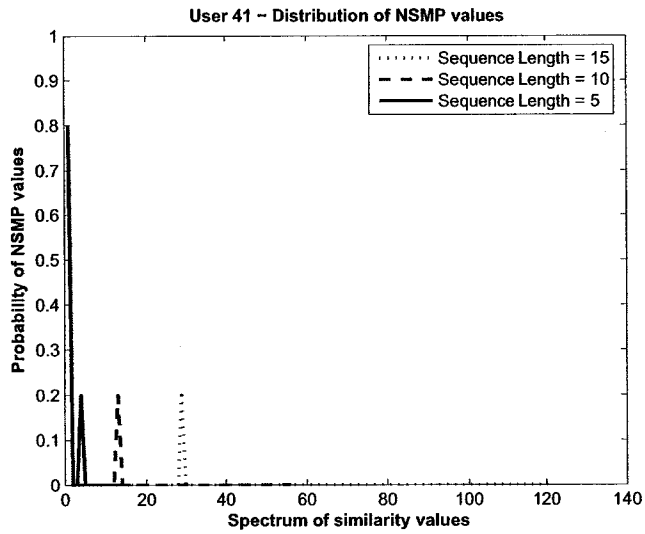

Figure 10.5: Characterization using different sequence lengths

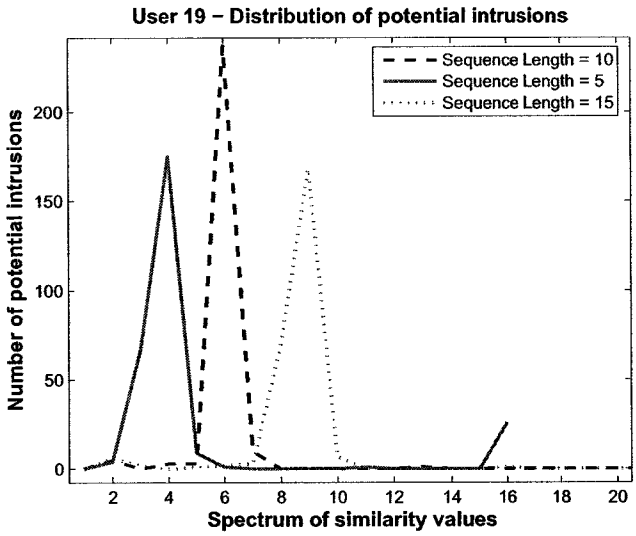

Figure 10.6: Intrusions at different sequence lengths

intrusions. All parameters, which were used in the previous test, remain the same, with one exception. The NSMP values of potential intrusions, are calculated using the training patterns of user 19 and test sequences of the remaining 49 users.

Fig. 10.6 depicts the distribution of intrusions, associated with each of the three SLs. It is important to note that we have zoomed in on the range of SM values between 1-16, since most of the intrusions are located in this range. The original x-axis, however, does cover the range of 1-121.

This figure demonstrates the fact that, as the SL is increased, the distribution shifts towards the higher end of the SM spectrum. This behavior is justified since there is a higher probability of achieving a high NSMP value, when the SL is longer. The key difference between user 19 and users 23 and 41 is the magnitude of the distribution. Due to the more chaotic behavior, the magnitude is higher for user 23 and even more so for user 41.

The last detail to note is the small number of intrusions at location 16 on the x-axis. It is an indication that the mobility sequences of one or more of the 49 users are identical (based on PL1) to user 19. A closer inspection reveals the identify of the potential intruder (user 13) responsible for most of these intrusions. Increasing the PL, discussed next, addresses this problem.

10.2.2 Precision Level

The analysis of PL, and its impact on the characterization of users and number of potential intrusions, is presented next. Given that our goal is to minimize the number of intrusions first and then to address the problem of characterization, we have used a SL of five, see Fig. 10.3. An expanded view of the similarity values, between 1 and 16, is presented in Fig. 10.7. It indicates that the distribution of NSMP, associated with a given PL, shifts towards the lower end of the spectrum, as the PL is increased, e.g. from PL2 to PL3. This behavior was observed with all three classes of users, see Fig. 10.8 and 10.9. Therefore, a lower PL can be used to improve characterization. Doing so, increases the probability of a match between a training pattern and a parameter sequence, resulting in a higher NSMP value.

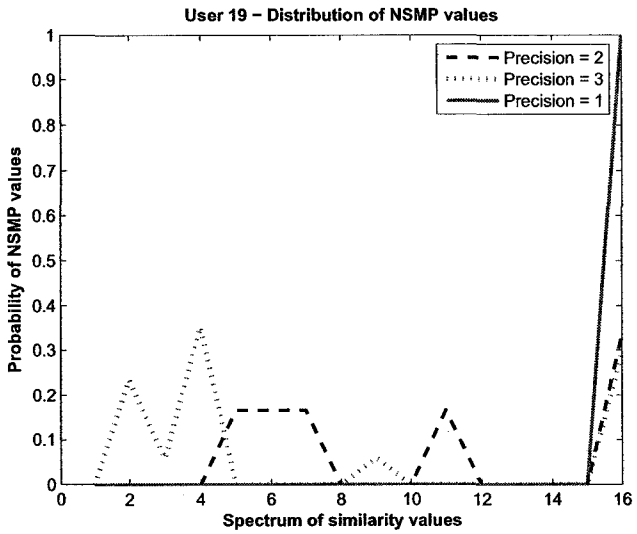

Figure 10.7: Characterization using different precision levels

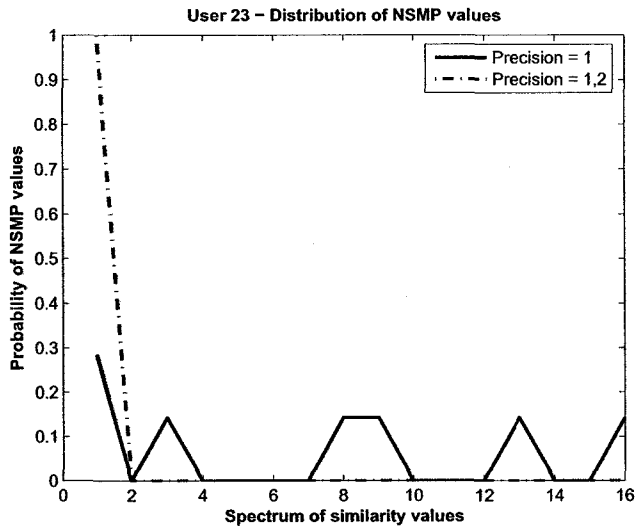

Figure 10.8: Characterization using different precision levels

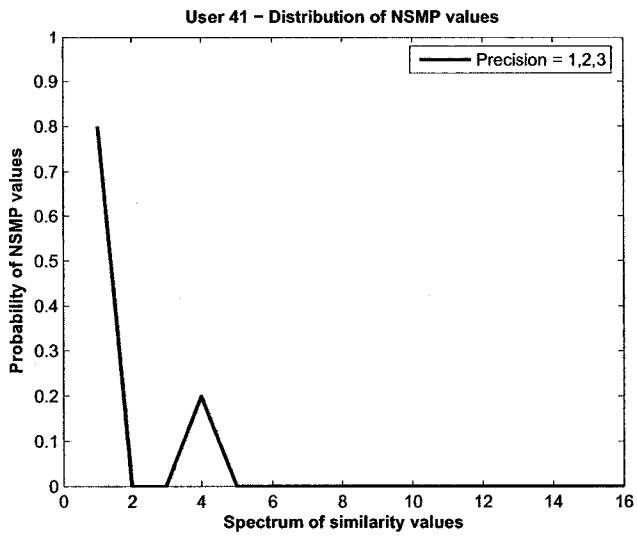

Figure 10.9: Characterization using different precision levels

Although the use of a lower PL is desirable for characterization purposes, it becomes problematic with respect to intrusions, see Fig. 10.10. What is exemplified, in this figure and applicable to all classes of users, see Fig. 10.11 and Fig. 10.12, is the fact that the distribution shifts towards the higher end of the spectrum, as the PL is decreased. On the other hand, the intrusions at SM value of 16 are eliminated when PL2 and PL3 are used. This should not come as a surprise since increasing the PL also decreases the similarity between two LCs. As a result, the probability of obtaining a high NSMP value is reduced, as indicated by the distribution of intrusions for PL3. Thus, the use of a higher PL would reduce the number of intrusions and improve the corresponding DR.

In summary, the selection of values for both the SL and PL is a challenging task given that many of the possible permutations produce undesirable results. Nevertheless, an optimal strategy would produce NSMP values, which are localized towards the higher end of the spectrum (characterization) and towards the lower end (intrusions) as well. This would set the stage for low FAs and high DRs.

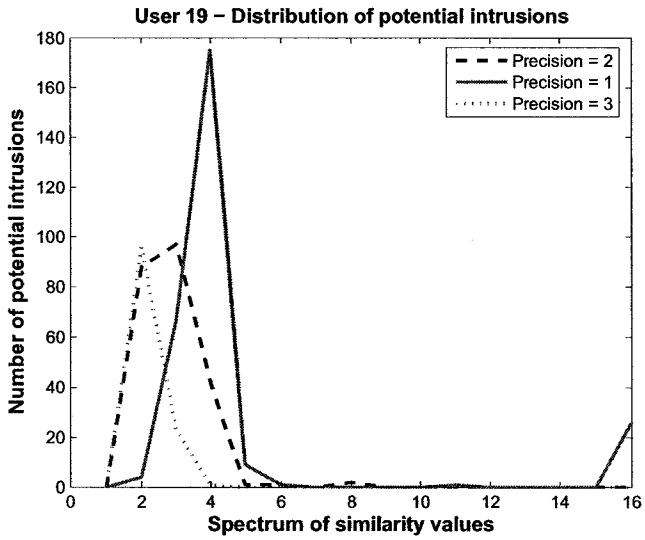

Figure 10.10: Intrusions at different precision levels

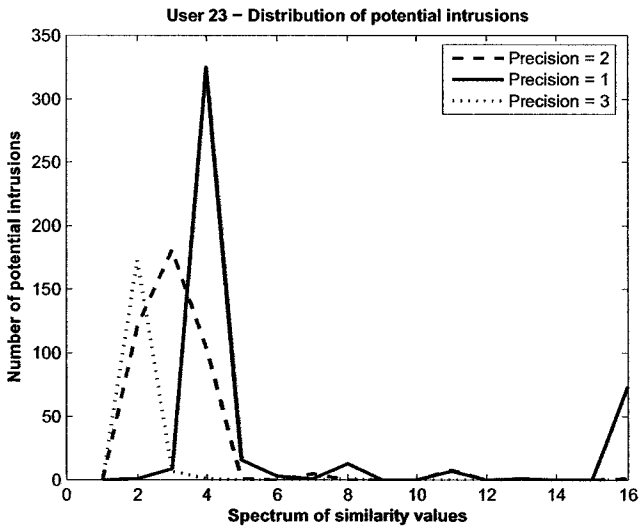

Figure 10.11: Intrusions at different precision levels

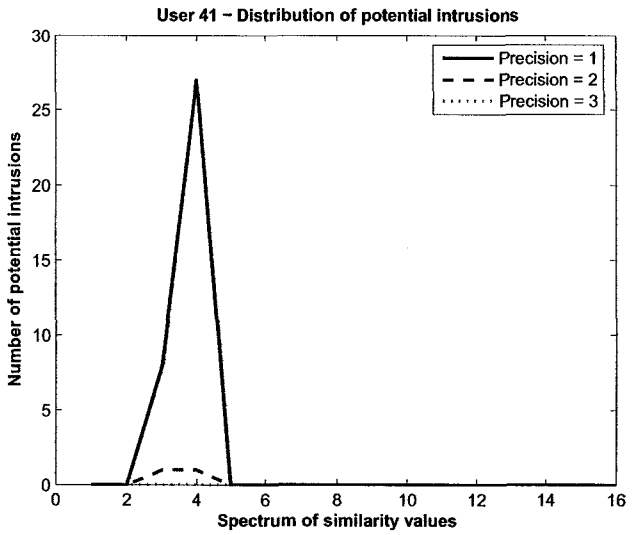

Figure 10.12: Intrusions at different precision levels

10.3 Evaluation

The primary objective of this exercise was to determine the impact of the PL on FAs and DRs (metrics). We relaxed the use of various SLs for the time being. We were also interested in the correlation between the quality of characterization, which can be attained using IBL, and the resulting FAs and DRs.

10.3.1 Evaluation Infrastructure

Details of the evaluation infrastructure are as follows. The acquisition of the LBs was carried out using Automatic Position Reporting System (APRS) and appropriate hardware (e.g. receiver and antenna). APRS is a packet radio-based system for tracking mobile objects. It captures and reports on locations, weather and other information for a geographical area, e.g. country or city. A detailed discussion of the APRS architecture and its use in supporting location-based service development is provided by Filjar and Desic [51].

Fig. 10.13 represents the infrastructure used to collect and process the LCs. The flow of information originating from users is as follows:

1. Location-related data, transmitted by user A, who is within the range of the

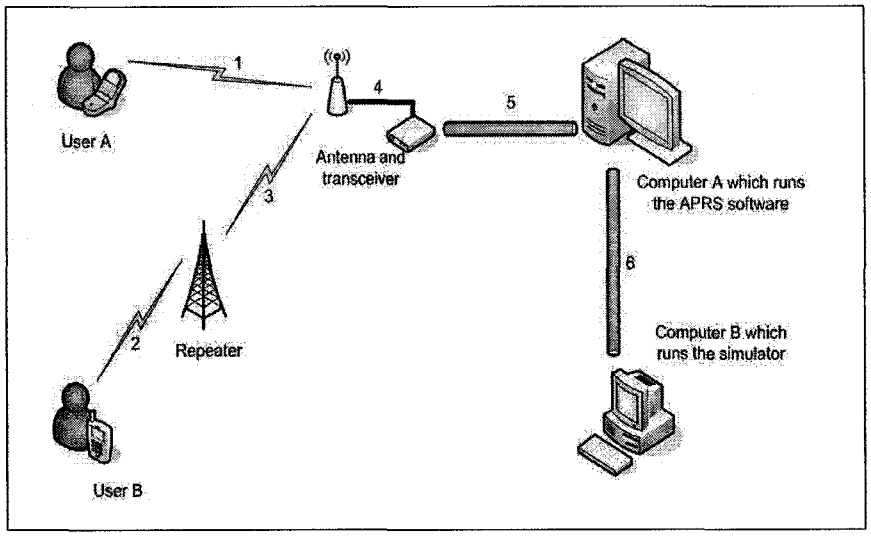

Figure 10.13: Infrastructure for data capture

antenna (1), is received by the transceiver. On the other hand, transmissions from user B, which are outside the range of the antenna, are first received by a repeater (2) and subsequently retransmitted (3) to the transceiver.

2. The transceiver demodulates the LBs and transfers the data to computer A (5), which hosts the APRS software. This data (e.g. LCs) are not only maintained in a log file by APRS, but are displayed on a map of a given region, e.g. city or province, in realtime.

3. Finally, the log file is transferred from computer A to a MySQL database, in computer B (6), for further processing. All subsequent analysis and evaluation exercises were carried out using Matlab software.

A screen capture of the users in the area of Ottawa-Carleton is presented in Fig. 10.14.

It has been suggested by Markoulidakis [115] that nearly 50% of all mobile users of public transportation, e.g. buses, can be characterized. This statistic has been confirmed to some extent by Wu [190]. Users, who took buses in the area of Los Angeles, were selected for this study. The city of Los Angeles was also chosen due to the high density of APRS users. The final set of 50 users was established based on the highest number of LBs transmitted by each user.

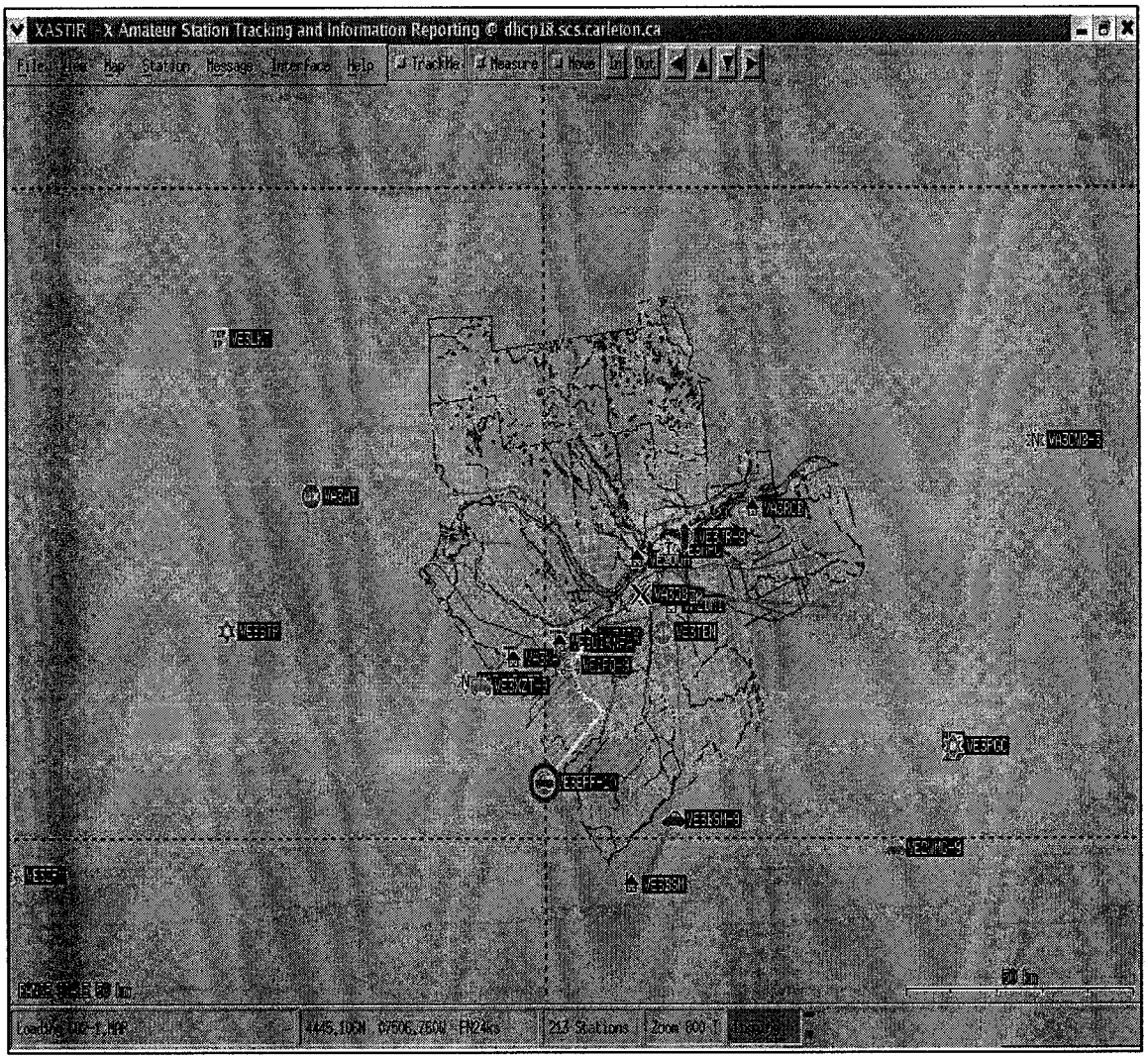

Figure 10.14: Screen shot of APRS

10.3.2 Details of Evaluation

The evaluation was carried out for each of the 50 profiled users. In order to determine the percentage of FAs, a comparison was made between the sequences in the test data of the user and his/her training patterns. The resulting NSMP values, which fell outside the acceptance region, were considered FAs. On the other hand, the DR or True Detect (TDs) was obtained by comparing the test sequences of the remaining users to the training patterns of the user being evaluated. As with FAs, all NSMP values, outside the acceptance region, were considered TDs. Statistics, corresponding to these metrics, were obtained for all profiled users.

10.3.3 Evaluation Results

We focus on the results obtained for representatives from each class, namely users that we number 19, 23 and 41 respectively in classes one, two and three.

False Alarm and Detection Rates

Figure 10.15 illustrates the percentage of FAs and TDs corresponding to each of the three PLs used. We begin by analyzing the results for user 19. We observe that there are no FAs for all three PLs. This is due to the fact that the three minimum thresholds of (16,5,2) associated with PLs 1,2, and 3, see Fig. 10.7, are all greater than the value of one. It is an indication that the mobility sequences in the test data are similar to those in the parameter data, which had been used, during the profiling phase, to establish the thresholds. In fact, it is the minimum threshold, at PL3, which most accurately reflects the similarity between these two sets of sequences.

In terms of TDs, the DRs decrease, as the PL is increased. Further scrutiny reveals that this behavior is appropriate, in light of the fact that the distribution of NSMP values shifts towards the lower end of the SM spectrum, see Fig. 10.10. Therefore, as the minimum thresholds shift towards the lower end, the probability of incorrectly classifying intrusions as normal behavior (i.e. intrusions that are within the acceptance range), becomes higher. This results in a decrease in the TD rate.

The characterization of user 23, on the other hand, is not as optimal. In fact, the NSMP values are distributed between the SM values of 1 and 16 (figure not shown) for PL1. The wide acceptance region and the minimum threshold of zero reflect the absence of mobility sequences (parameter data) in the training data. Although the test sequences may or may not be similar to those in the parameter set, the NSMP values of these test sequences have, nevertheless, fallen within the thresholds, resulting in zero FAs. These two factors, in particular, the low value of the minimum threshold, have also permitted all intrusions to remain undetected, resulting in a TD rate of zero. As the PL is increased to two and the maximum threshold becomes equivalent to the minimum threshold, it becomes more evident that the test sequences are dissimilar to those in the parameter data, but are nevertheless similar, to some degree, to those in the training data. Consequently, the FA rate becomes 100%. The corresponding TD rate, at PL2, also increases due to the fact that the intrusions, which had fallen inside the minimum and maximum thresholds in PL1, are now being detected at this level. Finally, as the PL is increased to three, the number of FAs decreases, as a result of the increase in intra-user variability between the parameter sequences and the training patterns. As expected, the TD rate also decreases as the PL is increased. Simply stated, the increase in inter-user variability, in conjunction with the pre-established thresholds, has influenced the DR of intrusions.

Finally, results for user 41 are very interesting, although somewhat misleading. We observe that, as with user 19, there are zero FAs for all three PLs. However, unlike user 19, the minimum and maximum thresholds of zero and four respectively, for all PLs, have permitted the NSMP values of all test sequences to fall within the narrow acceptance region. Similarly, the minimum threshold of value zero has also prevented all intrusions from being detected, even when the test sequences of all other users are dissimilar to the training patterns of user 41.

Enhanced Characterization

What can be ascertained, from the previous evaluation exercise, is the need to shift the minimum threshold towards the higher end of the spectrum, such that it is

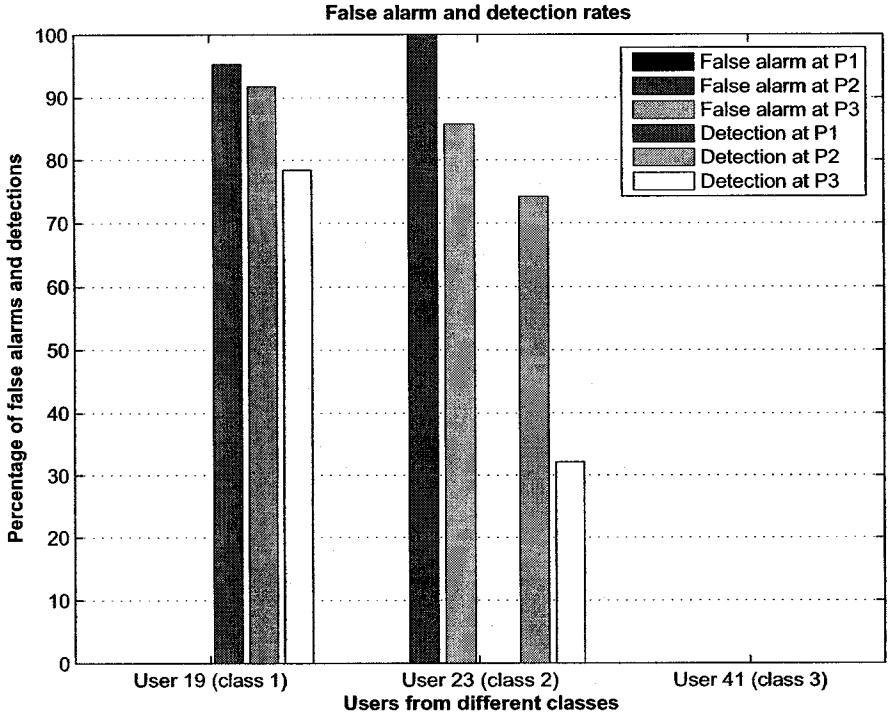

Figure 10.15: False alarms and detection rates for different precision levels

greater than zero. One simple strategy is to incorporate the mobility sequences from the parameter data, which have a NSMP value of zero, into the training data. This strategy reduces the width of the acceptance region and shifts the NPD, especially the minimum threshold, towards the higher-end of the spectrum. You may recall that the NSMP value, for a set of sequences, is calculated during the profiling (i.e. establishment of thresholds) phase.

Fig. 10.16 demonstrates the application of this strategy and the resulting impact on FA and TD rates. With user 19, the FAs remain unchanged whereas the TD rates (for all PLs) have increased, as expected. Moreover, the largest increase of 19% is associated with PL3, a desirable outcome. As far as user 23 is concerned, the three TD rates, associated with PL1,PL2 and PL3, have increased by 20%, 33% and 233% respectively. However, the FAs for PL3 has also increased due to the dissimilarity of some of the test sequences to those in the parameter set. Finally, the results for user 41 exemplify the potential benefit of this strategy. Although a 5% increase in the FAs

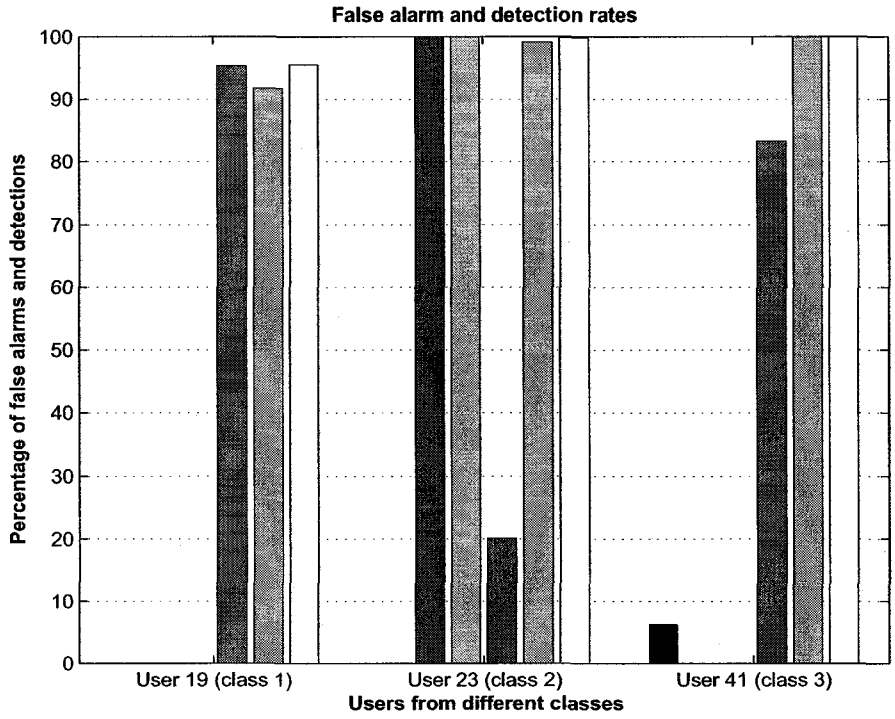

Figure 10.16: False alarms and detection rates using enhanced characterization

(at PL1) has been incurred, there is, nevertheless, a significant improvement in the TDs (85%, 100%, 100%), associated with the three PLs.

10.3.4 Memory Requirement and Running-Time Complexity

As with Hotelling's T^2 classifier, we perform similar analysis of the IBL classification algorithm, with respect to memory requirements and running-time complexity.

Memory Requirements

Requirements for memory are dictated primarily by two variables: the number of LCs or (p) and number of location sequences in both the set n (to be classified) and profile m. Whereas, the number of bytes, per location sequence, is represented by $4p$, total requirements for memory is approximately $4p(n+m) = 4pn + 4pm$.

Running-time Complexity

The classification of an observed set of location sequences is carried out using Algorithms 3 and 4. Algorithm 3 calculates the degree of similarity between two location sequences.

1. For each LC lco_p in the observed sequence LSO, a comparison is made to the corresponding coordinate lcp_p in the sequence, associated with the profile LSP.

2. If these two coordinates are identical, then the current value of the similarity measure $smValueCurrent$ is incremented by one.

3. Once all the LCs have been processed, the total value of the similarity measure $tsmValue$, i.e. sum of values derived in the previous step, is returned.

Algorithm 3 Similarity Measure(LSO,LSP)

```
1: smValueCurrent, tsmValue ← 0; LC ← 10 {p denotes a LC}
2: for p = 1toLC do
3:    if lco_p = lcp_p then
4:       smValueCurrent ← smValueCurrent + 1
5:    else
6:       smValueCurrent ← 0
7:    end if
8:    tsmValue ← tsmValue + smValueCurrent
9: end for
10: return tsmValue
```

Using the previous algorithm, the classification of a set of location sequences, using IBL, is executed as follows:

1. For each location sequence lso_n in the set of observed sequences OS, the similarity measure $smValue$, between lso_n and each sequence lsp_m in the profile PP, is obtained. In addition, for each lso_n, the maximum value of the set of similarity measures is determined.

2. Once all the sequences in OS have been analyzed, the average value $nsmpValue$ of the set of maximum values is calculated.

3. If $nsmpValue$ falls within the pre-established minimum $minThreshold$ and maximum $maxThreshold$ thresholds, OS is considered normal. Otherwise, a *status* of anomalous is returned by the algorithm.

Algorithm 4 IBL Classification

```
1: smValue, maxsmValue, largestValue, nsmpValue ← 0 {Obtain nsmpValue for
   set of sequences in OS}
2: for n = 1toOS do
3:   for m = 1toPP do
4:     smValue ← SimilarityMeasure(lso_n, lsp_m)
5:     if smValue > largestValue then
6:       largestValue ← smValue
7:     end if
8:   end for
9:   maxsmValue ← maxsmValue + largestValue
10: end for
11: nsmpValue ← maxsmValue/n
    {Return normal if nsmpValue is within thresholds}
12: if minThreshold ≥ nsmpValue ≤ maxThreshold then
13:   status ← NORMAL
14: else
15:   status ← ANOMALOUS
16: end if
17: return status
```

As far as the running-time is concerned, the key task of determining the $nsmpValue$ is performed using $n(m[4p+2]+1)$ or $nmp+2nm+n$ operations, where $4p$ represents the requirement for Algorithm 3.

Part VI

Post Review

Chapter 11

Conclusions and Future Initiatives

This chapter presents the conclusions of the research initiatives, undertaken to date. While evaluation results are promising, there is ample room for future enhancements. Given that these results are primarily intended to validate proof of concept, further research is required in order to: fully assess the feasibility of the proposed methodologies; test their effectiveness in real-time systems; and to eventually integrate them into future IDSs for wireless networks. Furthermore, future research initiatives, associated with the two ABID components as well as other related areas, are also identified.

11.1 ABID using RFF

Detection of start of transients

The Threshold, BSCD and BRCD approaches, which utilize amplitude (discriminator output) characteristics of signals, can be employed to detect the start of a transient. However, a detection algorithm, that exploits the phase characteristics of a signal, provides better performance. For example, it accommodates signals where the transition, between channel noise and transient, occurs more gradually. Table 11.1 highlights the key aspects of this algorithm.

As indicated, it is particularly useful when processing signals, associated with BT and WiFi/802.11 transceivers. The higher success rate is attributable to two key factors. First, the susceptibility of the signal phase to noise and interference is minimal.

Strengths	Success rate (BT:85-90% WiFi:95%) and O(n)
Weaknesses	Dependence on thresholds
Appropriate use	BT and WiFi/802.11 signals

Table 11.1: TDPC: Strengths and weaknesses

Strengths	Average success rate (94-100%), low space requirements
Weaknesses	Worst case running time of $O(nmp^2 + nm)$, scalability
Appropriate use	Identify a transceiver

Table 11.2: Transceiver identification: Strengths and weaknesses

Second, as aforementioned, there is a noticeable difference in the characteristics of the slope, i.e. difference in phase variance, during channel noise and transmission. Moreover, the time complexity of O(n) is similar to other algorithms, although techniques for optimizing its performance would prove beneficial.

As far as weaknesses are concerned, the need to establish system-wide thresholds may be construed as a potential weakness. Given the inter-transceiver variability, it may prove advantageous to establish transceiver-based thresholds.

While preliminary results are promising, the detection algorithm should be further enhanced in order to achieve a higher success rate and to accommodate signals from other wireless devices. Benefits accrued from this exercise will no doubt play a vital role in the next phase, the classification of transceivers.

Transceiver Identification (RFF and Bayesian)

Based on evaluation results, see Table 11.2, the use of RFF and Bayesian filtering for ABID is technically feasible. More specifically, the characterization of transceivers using multiple features, derived from the amplitude, phase and frequency components of the signal, has proven to be effective (high success rate) for classification purposes. In addition, the use of a statistical classifier minimizes both memory and storage requirements, i.e. only the covariance matrix and centroid are required for each transceiver.

Furthermore, delaying the decision making process until a sufficient number of transceiverprints have been classified (using the Bayesian filter), addresses the uncertainty associated with current IDSs. By accommodating some variability or deviations

Strengths	Success rate (WiFi): FAR(0%) and average DR(95%) Success rate (BT): average FAR (0.005) and DR (93%)
Weaknesses	Worst case running time of $O(np^2 + 3n)$
Appropriate use	Verify a transceiver

Table 11.3: Transceiver verification: Strengths and weaknesses

from normal behavior, e.g. transceiverprints, an increase in the confidence level and classification success rate would ensue.

The worst case running time of $O(nmp^2 + nm)$ raises some concern. Since the goal of the classification process is to identify the most probable transceiver, a one (set of transceiverprints) to m (transceiver profiles) comparisons are required. With thousands of transceiver profiles in the IDS, scalability becomes an issue.

There are also other issues, which warrant further attention. For example, the lower success rates for certain transceivers must be addressed. These rates can be improved by enhancing the composition of the transceiverprints. One option is to extract features from the turn-off transient at the end of signals. Other options include the use of filters, signal to noise ratio and other RF-based characteristics.

Another interesting initiative worth pursuing is the use of $NtoM$ ratios, where N represents a set of authorized transceivers and M the unauthorized ones (intruders), for assessing the performance of the classification component.

Transceiver Verification (RFF and Hotelling's T^2)

In terms of transceiver verification, results from evaluation exercises, see Table 11.3 support the use of RFF and Hotelling's T^2 for ABID.

As with transceiver identification, the characterization of transceivers using *multiple* features is equally effective for verification purposes. In fact, the same transceiverprints are used for both types of classification. In addition, the Hotelling's T^2 classifier is memory conscious, i.e. only 960 bytes are required per transceiver profile. Although the term p^2 can be treated as a constant, the running time of $O(n)$ can be further optimized for supporting on-line detection schemes. Another similarity is the use of multiple transceiverprints by the decision filter. This strategy improves both the FAR and DR. Finally, the consistency of the FAR and the improved DR support the use of

dynamic profiles, which are updated continuously using one or more of the currently observed transceiverprints. Nevertheless, a detailed analysis of the memory requirement and running-time complexity, associated with the use of dynamic profiles, must be carried out. Results from this exercise may influence the frequency with which the profiles are updated.

While preliminary results are encouraging, there are some issues that should be addressed through further research. First and foremost, as with transceiver identification, the DR could be further improved by optimizing the composition of the transceiverprints and validating them using a larger set of transceivers from the same manufacturer. In fact, the success rate, associated with the *NtoM* ratio, should be further investigated. Moreover, it would prove useful to determine the response of the classification system to various percentages, used in the decision filter.

Second, although the signals were captured in a quasi-controlled environment, i.e. indoors with a few other APs, it is important to take into consideration such factors as mobility and ambient temperature. Given that these factors are likely to change the characteristics of the transceivers and their corresponding transceiverprints, the data capture and profiling exercises should be repeated periodically. The transceiverprints, obtained from this exercise, can subsequently be used to update the profiles of the transceiver, using the EWMA technique.

Finally, simulations and field tests should be carried out to determine system performance in a more realistic setting.

11.2 ABID using UMP

Profiling and Classification

The key advantage of this approach, see Table 11.4 is the use of IBL. The adoption of the IBL classification technique is suitable since the definition of the similarity measure is comparable to that of the Euclidian distance. Supplemented by the high level mapping exercise, which reduces the intra-user variability between mobility sequences and training patterns, this technique performs well, as indicated by the FAs

Strengths	Use of IBL
Weaknesses	Characterization, worst case running time of $O(nmp + 2nm)$

Table 11.4: IBL classification: Strengths and weaknesses

and DRs obtained for all three classes of users, i.e. using the enhanced characterization strategy.

In terms of weaknesses, the strategy of using the first n partitions of mobility patterns (in the data set), for training purposes, is only appropriate for characterizing users who exhibit consistent mobility behavior. Other users with less consistent or chaotic behavior cannot be characterized accurately using this approach. One simple strategy, which enhances the characterization of all users and increases the DR at a minimal cost (low percentage of FAs), is to incorporate the missing parameter sequences into the training patterns. A more effective approach would require the use of clustering techniques for selecting training patterns.

Furthermore, the issue of concept drift should also be addressed by continuously monitoring the FAR and selectively incorporating newly observed mobility sequences into the training patterns. The use of a window that is shifted in time (analogous to exponentially weighted moving average) would prove beneficial. Moreover, the selection criteria can be based on pre-established thresholds, such as the frequency of all new sequences encountered over a period of time.

Once the characterization of users has been adequately addressed, the selection of specific values for SL and PL should be based on the level of intra-user variability. These values could then be incorporated into a user's profile. Categorizing users into different classes, based on the level of variability, represents an alternate strategy.

Another potential weakness, is the worst case running time of $O(nmp + 2nm)$. In particular, the large number of mobility patterns m in a profile will most likely dominate the performance (one to many comparisons) of the classification algorithm. Not surprisingly, it is similar to $O(nmp^2 + nm)$, associated with the classification algorithm for transceiver identification.

As far as other research initiatives are concerned, the following issues could be pursued in the future: *user privacy*; the expansion of the feature set, e.g. time frame

and other relevant features, for improving DR; a comprehensive analysis of system performance for comparison purposes; and the use of different parameter values, which reflect the mobility behavior of users.

11.3 Other research initiatives and applications

Research in the area of ABID, in wireless networks, is perhaps in its adolescence, and hence, necessitates further study. While some direction is provided in this section, there is no doubt that other initiatives will become more apparent over time.

IDS using multiple contributors

As aforementioned, our secondary objective is to make use of multiple IDS modules for enhancing the overall detection capabilities of a WIDS. More specifically, the use of a probabilistic model, e.g. Dempster-Shafer, can permit a WIDS to render a final decision of normal or anomalous, based on the combined belief and plausibility for a set of events, e.g. potential attacks. Details of this model are presented in Appendix B.

Other Potential applications

Other potential applications, which could be realized by the aforementioned research objectives, are presented in this section. Although the following applications are technically feasible, we acknowledge the critical need to preserve the privacy of users.

Detection of cellular phone cloning

In addition to the detection of MAC and BT_ADDR address spoofing, a device-based IDS can also prove useful for identifying clones in cellular networks.

Location confirmation of devices

Another potential application is the determination and/or confirmation of the location of devices.

By associating an identifier of a device with its corresponding characteristics, the location of a given device can be determined/confirmed by a network entity, e.g. IDS. This technique is equally applicable to all transceivers used in BT, WiFi/802.11 and cellular devices.

While it is also possible for an intruder to obtain this information for nefarious purposes, the required tools and technical expertise are currently available but to a very small percentage of users.

Detection of abnormal mobility behavior in robotics

Another potential application of a mobility-based WIDS mechanism, is the detection of unauthorized mobility patterns, associated with automated and autonomous systems, e.g. robotics. Once the mobility patterns have been profiled, they can prove useful in monitoring the on-going behavior of these systems. Given that these mobility patterns are more or less static, it should be fairly straightforward to detect deviations in behavior, using appropriate thresholds.

Appendix A

RF Signals

A.1 Representation of Signals

While a brief introduction is provided in this section, there are many references on digital signal processing, including [140], available for consultation.

An analog signal can be defined as a real or complex signal. A real signal, such as a cosine wave, is typically represented by positive frequency components that can be analyzed in the frequency domain. In fact, the same signal contains both positive and negative frequencies, when examined in the complex domain.

A real signal is represented as

$$X(t) = \cos \omega t \tag{A.1}$$

where $\omega = 2\pi f$, an angular frequency, and t is time. In the complex domain, its representation consists of two complex signals with positive and negative frequency components.

$$X(t) = \frac{1}{2}[(\cos \omega t + j \sin \omega t) + (\cos \omega t - j \sin \omega t)]$$

The addition of the real or in-phase terms ($\cos\omega t$) and the elimination of the imaginary or quadrature terms ($\pm j \sin\omega t$) result in $\frac{1}{2}(2\cos\omega t)$, which is identical to (A.1).

Given that the negative frequency component of $X(t)$ is redundant, this signal can be converted to a complex signal containing only positive frequency components. This conversion is carried out by first generating a quadrature signal $Q(t)$ using the following quadrature filter or Hilbert Transform [130]

$$Q(t) = \frac{1}{\pi t} * I(t) \tag{A.2}$$

whereby all frequencies in an in-phase signal $I(t)$, which is equivalent to a real signal $X(t)$, are phase-shifted by 90°. Next, a complex signal with positive frequency components, referred to as an analytic pair, is obtained via $I(t) + jQ(t)$, where j represents $\sqrt{-1}$. Please note that the $(*)$ represents the convolution of $I(t)$ with the function $\frac{1}{\pi t}$ in the time domain.

Although only real signals have been used extensively in the past, the use of complex signals provides one key advantage. Since the amplitude and phase components of the digital signal are adequately preserved during the analog-to-digital (A/D) conversion, as indicated by Ellis and Serinken [45], they can be used to enhance the profiling phase of the RFF process.

A.2 Signal Components

The instantaneous amplitude and phase of an analytic pair can be calculated as follows.

$$a(t) = \sqrt{I^2(t) + Q^2(t)} \tag{A.3}$$

$$\theta(t) = \tan^{-1}\left[\frac{Q(t)}{I(t)}\right] \tag{A.4}$$

To simplify the analysis of phase characteristics, the instantaneous phase $\theta(t)$ is unwrapped in order to remove discontinuities that occur at multiples of 2π radians. It is denoted here as $S_h(t)$, and is defined as

$$S_h(t) = \left\{ \begin{array}{ll} \theta(t) & \text{if} \quad \mid \theta(t-1) - \theta(t) \mid \leq \pi \\ \theta(t) \pm 2\pi & \text{otherwise} \end{array} \right\} \tag{A.5}$$

While the use of instantaneous frequency, as defined in

$$f_i(t) = \frac{1}{2\pi}\frac{d}{dt}S_h(t)Hz$$

can be used to determine the unique frequency characteristics of signals, the preferred approach for obtaining the frequency component of a non-stationary signal (e.g. transient, see Fig. 5.1) is the application of the Wavelet Transform (WT) [134]. This technique decomposes the signal into its frequency components while preserving both time and frequency resolution.

In order to provide time localization of frequency components, the WT employs a set of functions, a wavelet or basis function and a scaling function, that are associated with a high pass and low pass filter respectively. The WT coefficients are obtained by forming the inner product of the scaling and wavelet coefficients with the signal data.

The formal definition of the WT is as follows:

$$\begin{aligned} \gamma(j,k) &= < A(t), \Psi_{j,k}(t) > \\ \Psi_{j,k}(t) &= \frac{1}{\sqrt{s_0^j}}\Psi\left(\frac{t - k\tau_0 s_0^j}{s_0^j}\right) \\ \Psi(t) &= 2\sum_{k=1} g(k)\Phi(2t-k) \\ \Phi(t) &= 2\sum_{k=1} h(k)\Phi(2t-k) \\ g(k) &= (-1)^k h(N-k) \end{aligned} \tag{A.6}$$

where $\gamma(j,k)$ are the WT coefficients at level/scale j and time (delay) k. While the

$\Psi_{j,k}(t)$ represents the wavelet function at scale j and time k, $\Psi(t)$ is the original wavelet function that is defined in terms of the translated scaling function $\Phi(t)$. The dilation step and translation factor are defined as $s_0^j = 2$ and $\tau_0 = 1$. Finally, $g(k)$ is the high pass filter, which is derived from the low pass filter with a length of N.

Although a sampled version of the Continues Wavelet Transform (CWT) can be used, the resulting information is highly redundant as far as the reconstruction of the signal is concerned. Moreover, it requires a significant level of computational resources. On the other hand, the DWT provides sufficient information for analysis, without the aforementioned overhead. Furthermore, the underlying principle is the same as it is in CWT, and a time-scale or time-frequency representation of a digital signal is obtained through the application of digital filters. More specifically, the signal is passed through a sequence of high pass and low pass filters in order to analyze the high and low frequencies respectively. A single level of decomposition can be expressed mathematically as:

$$y_{high}[k] = \sum_n x[n] \cdot g[2k - n] \tag{A.7}$$

$$y_{low}[k] = \sum_n x[n] \cdot h[2k - n] \tag{A.8}$$

where $x[n]$ is the discrete version of $X(t)$, and $y_{high}[k]$ and $y_{low}[k]$ represent the outputs of the high pass $g[n]$ and low pass $h[n]$ filters respectively, after subsampling by two.

A brief summary of DWT, also known as subband coding, pyramidal coding and multiresolution analysis, is as follows:

- The discrete signal $x[n]$, of length L is passed through $g[n]$ and $h[n]$. The output of $g[n]$ is referred to as level one DWT coefficients. The $g[n]$ reduces the frequency band of $x[n]$ by half, i.e. from (0 to π) to ($\frac{\pi}{2}$ to π), and decimates the number of samples by two to eliminate redundancy, e.g. $\frac{L}{2}$. On the other hand, the frequency band of $h[n]$ is also reduced, i.e. from (0 to $\frac{\pi}{2}$).

- Using the resulting samples ($\frac{L}{2}$) of $h[n]$ as input, the process of filtering and

decimating is repeated. This results in $\frac{L}{4}$ level two DWT coefficients, which represent ($\frac{\pi}{4}$ to $\frac{\pi}{2}$) frequency band. As with the previous decomposition, the frequency band, associated with $h[n]$, becomes (0 to $\frac{\pi}{4}$). This process continues until the last DWT coefficient is computed.

- The DWT of $x[n]$ is subsequently obtained by concatenating the coefficients at each level, starting with the last level of decomposition. Fig. 6.15, last plot, represents the DWT coefficients of a transient from transceiver (ID 404).

Appendix B

Dempster-Shafer Theory

In the area of ABID, the research community has come to acknowledge the following statements:

1. It is difficult to achieve an FAR of 0%.

2. It is difficult to achieve an DR of 100%.

Naturally, the most optimal rates, which can be achieved using a given ABID component, depend on two key factors. For one thing, the implementation of the underlying mechanisms, e.g. profiling and classification techniques, can influence both FAR and DR. However, even more importantly, these rates are dictated by the characteristics or features used in device or user profiles. What is generally accepted is that all behavioral characteristics do have a tendency to change, from time to time.

On one hand, device-based characteristics, e.g. transceiverprints, are more likely to change, as a result of transceiver aging, in a gradual manner, and over a period of time. The primary factor is the narrow tolerance levels defined by specifications. Therefore, by accurately profiling these devices and by addressing concept drift, it is feasible to achieve and to maintain a low FAR. However, as specifications become more stringent, it could become increasingly difficult to make a distinction between two devices, e.g. WiFi/802.11 wireless cards, from the same manufacturer. More specifically, the uncertainty, associated with the classification process, could result in

a decrease in the DR. This phenomenon is equally applicable to other transceivers, including those used in cell phones.

On the other hand, user-based characteristics, e.g. mobility patterns, can either be very consistent, as with domestic users, or even more chaotic, e.g. roamers, than those based on devices. Consequently, the FAR and DR are influenced by the category of the user. Thus, for example, users exhibiting consistent mobility behavior are represented by a low FAR and high DR. In contrast, a high FAR and high DR are typically associated with individuals, whose behavior lies at the opposite side of the mobility spectrum. Furthermore, an intruder's ability to successfully masquerade as a legitimate user is also influenced by these characteristics.

So, in order to render a final verdict, i.e. normal or anomalous, one of two strategies can be adopted:

1. Accept the classification uncertainty of a given ID component, and adjust the classification thresholds based on such factors as application requirements; or

2. One can reduce the classification uncertainty by incorporating classification results from multiple ABID components, e.g. mobility-based and activity-based profiles, as suggested by Samfat and Molva [149].

B.1 Overview

The details of the Dempster-Shafer theory [155], which can be used for the reduction of classification uncertainty, are presented next.

The ultimate goal of this theory is to permit a system, e.g. ABID, to render a decision, based on the *belief* and *plausibility* that are derived from a set of contributors, e.g. ABID components. Whereas belief represents the degree to which an event, e.g. classification of an observation as being normal, is supported, plausibility is defined as the lack of evidence to oppose the former. These two values provide the basis for any belief-based decision making system.

Details

The following list outlines the key elements of this theory.

Responsibility of each contributor Each contributor assigns a weight or probability (between 0 and 1) to each member of the set of mutually exclusive events under consideration.

Assignment of weights The empty set {} is always initialized to 0. Furthermore, the sum of all weights over all subsets must equal the value of one. A non-zero weight, assigned to a set with more than one event, indicates a degree of ignorance about the events in the set. Finally, a non-zero value of a set of all events, e.g. $\{A, B\}$ in the two-event case, represents overall ignorance.

Definition of Belief Belief, of a given event, is calculated by summing the weights of all the subsets that contain that event.

Definition of Plausibility On the other hand, plausibility is obtained by subtracting the complement, i.e. all other events, from the value of one. In other words, it is defined as the sum of all the weights, which correspond to all subsets containing the complement. So, in the case of events A and B, plausibility(A)= 1 - W(B) - W(A,B) and vice versa.

Decision Criteria If both belief and plausibility for a given event (e.g. A) is higher than the corresponding values of all other events, then event A is declared the most likely. However, in general, the event with the highest plausibility may not be the same as that with the highest belief. Under these circumstances, other heuristics must be applied.

The calculation of belief and plausibility is performed by integrating the results from multiple contributors. In particular, it is carried out by multiplying the corresponding weights from each contributor. Hence, the aggregated $belief(A) = W_1(A) * W_2(A)$, where the subscripts, associated with the weights, denote the contributor. Finally, the calculation of the aggregated plausibility is carried out in a similar manner.

Contributor	Normal	Anomalous
C1	0.60	0.40
C2	0.30	0.70

Table B.1: Weights assigned by all contributors

B.2 Application

We illustrate the application of the theory to an ABID system in cellular networks. As stated previously, the goal is to determine if an observed behavior of a user is normal or anomalous, i.e. an intruder. This decision is rendered by using two independent ABID components, which make use of activity (Contributor 1) and mobility (Contributor 2) data.

Table B.1 identifies the weights arbitrarily assigned by each contributor to the two events.

In order to take the conflicting views of C1 and C2, i.e. C1's result indicates normal behavior whereas the opposite is supported by C2, into consideration, the Degree of Conflict is defined as follows:

Calculation of Degree of Conflict:

$$\text{DoC} = 1 - (W_1(Normal) * W_2(Anomalous)) = 1 - (0.60 * 0.70) = 0.58$$

Calculation of Belief:

$$\text{Belief(Normal)} = \frac{W_1(Normal)*W_2(Normal)}{DoC} = \frac{0.6*0.3}{0.58} = 0.31$$

$$\text{Belief(Anomalous)} = \frac{W_1(Anomalous)*W_2(Anomalous}{DoC} = \frac{0.4*0.7}{0.58} = \mathbf{0.48}$$

Calculation of Plausibility:

Plausibility(Normal) = 1 - 0.48 = 0.52

Plausibility(Anomalous) = 1 - 0.31 = **0.69**

Given that both belief and plausibility, associated with the classification result of *anomalous*, are higher, it would appear as if the observed behavior is not consistent with the profile of the user. In other words, the aggregated results are indicative of

a potential intrusion. You may have already expected this outcome, in light of the fact that C2's value of belief(anomalous) is higher than that of C1's belief(normal). Although it would be logical in this case, this observation may not necessarily hold when more than two events are being considered.

B.3 Proposed Extension

As illustrated previously, the application of the theory to ABID in cellular and other wireless networks would prove useful. However, the theory does not provide a specific mechanism for indicating the relative accuracy of the underlying component. Assigning weights to each of the contributors, which are normalized to the value of one, would enhance the interpretation of the individual and aggregated results. Thus, for example, belief(normal) would be defined as:

$$\text{belief(normal)} = \frac{(C_1 * W_1(Normal)) * (C_2 * W_2(Normal))}{DoC}$$

where C_i represents the degree of accuracy assigned to a contributor. The same principle would apply to all other calculations also, including the degree of conflict.

List of Acronyms

ABID Anomaly-based Intrusion Detection

ACL Access Control List

ADC Analog to Digital Conversion

AES Advanced Encryption Standard

AH Authentication Header

ANN Artificial Neural Network

AP Access Point

APRS Automatic Position Reporting System

ARA Attack Risk Analysis

AuC Authentication Center

BRCD Bayesian Ramp Change Detector

BSC Base Station Controller

BSCD Bayesian Step Change Detector

BSS Basic Service Set

BSS_2 Base Station Subsystem

BT Bluetooth

BTS Base Transceiver Station

CAVE Cellular Authentication and Voice Encryption

CDMA Code Division Multiple Access

CRC Cyclical Redundancy Check

DAP Data Access Point

DCS Digital Cellular System

DHCP Dynamic Host Configuration Protocol

DoS Denial of Service

DR Detection Rate

DWT Discrete Wavelet Transform

EAP Extensible Authentication Protocol

ED Euclidean Distance

ESP Encapsulating Security Payload

ETSI European Telecommunications Standards Institute

EWMA Exponentially Weighted Moving Average

FA False Alarm

FAR False Alarm Rate

FTP File Transfer Protocol

GPRS General Packet Radio Service

GSM Global System for Mobile Communications

HLR Home Location Register

HMAC Hash Message Authentication Code

HTTP HyperText Transfer Protocol

IBL Instance-Based Learning

IDS Intrusion Detection System

IMSI International Mobile Subscriber Identity

IMT International Mobile Telecommunications

IP Internet Protocol

IPSec IP Security

ISDN Integrated Services Digital Network

IV Initialization Vector

LA Location Area

LAN Local Area Network

LB Location Broadcast

LC Location Coordinate

MAC Media Access Control

MAC_2 Message Authentication Code

MAF Moving Average Filter

MN Mobile Node

MS Mobile Station

MSC Mobile Switching Center

MSPC Multivariate Statistical Process Control

MVA Multivariate Analysis

NPD Normalized Probability Distribution

NSMP Noise suppressed Similarity Measure to Profile

PAN Personal Area Network

PIM Personal Information Management

PIN Personal Identity Number

PKI Public Key Infrastructure

PL Precision Level

PMAKAP Password Mutual Authentication and Key Agreement Protocol

PNN Probabilistic Neural Network

PPP Point to Point Protocol

PRNG Pseudo Random Number Generator

PSTN Public Switched Telephone Network

QoS Quality of Service

RADIUS Remote Authentication Dial-In User Service

RAND Random Number

RAP Rogue Access Point

RAS Remote Access Server

RBAC Role-Based Access Control

RBS Rogue Base Station

RFCOMM RF Communications

RFF Radio Frequency Fingerprinting

RFID Radio Frequency Identification

SAKA Subscriber Authentication and Key Agreement

SBRS Steel-Belted Radius Server

SIG Special Interest Group

SIM Subscriber Identity Module

SL Sequence Length

SM Similarity Measure

SMP Similarity Measure to Profile

SNMP Simple Network Management Protocol

SOM Self Organizing Map

SRP Secure Remote Password

SSID Service Set Identifier

SSL Secure Sockets Layer

TDMA Time Division Multiple Access

TDPC Transient Detection using Phase Characteristics

TKIP Temporal Key Integrity Protocol

TLS Transport Layer Security

TMSI Temporary Mobile Subscriber Identity

TRA Threat Risk Analysis

UMP User Mobility Pattern

UMTS Universal Mobile Telecommunications System

UTRA UMTS Terrestrial Radio Access

VLR Visitor Location Register

VPN Virtual Private Network

WAP Wireless Application Protocol

WEP Wired Equivalent Privacy

WIDS Wireless Intrusion Detection System

WiFi Wireless Fidelity

WiMax Worldwide interoperability for Microwave Access

WLAN Wireless Local Area Network

WNIC Wireless Network Interface Card

WPA WiFi Protected Access

WTLS Wireless Transport Layer Security

WWAN Wireless Wide Area Network

Bibliography

[1] 3GPP. Specification of the MILENAGE algorithm set. http://www.3gpp.org. TS 35.205 - accessed January 2006.

[2] 3GPP. Technical Specifications Group Services and System Aspects; 3g Security Architecture. http://www.3gpp.org, September 2003. TS 33.102 - accessed January 2006.

[3] 3GPP2. Signaling Link Access Control (LAC) Standard for cdma2000 Spread Spectrum Systems. http://www.3gpp2.org/C-S0004-C.v1.0.pdf, May 2002. C.S0004-C - accessed January 2006.

[4] 3GPP2. Upper Layer (Layer 3) Signaling Standard for cdma2000 Spread Spectrum Systems. http://www.3gpp2.org/C-S0005-C.v1.0.pdf, May 2002. C.S0005-C - accessed January 2006.

[5] B. Aboba. http://www.drizzle.com/ aboba/ieee/11-01-253r0-i-wep2securityanalysis.ppt. WEP2 Security Analysis, May 2001. Doc: IEEE 802.11-00/253 - accessed in January 2006.

[6] B. Aboba. The Unofficial 802.11 Security Web Page - Security vulnerabilities in EAP methods. www.drizzle.com/ aboba/IEEE/, May 2005. Accessed in January 2006.

[7] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. Extensible authentication protocol (EAP). The Internet Engineering Task Force - Request for Comments: 3748, June 2004.

[8] J. Abramowitz. Wireless LANs - poised for untethered growth. http://www.wlana.org/pdf/wlana-industry.pdf, 2001. Accessed in January 2006.

[9] F. Adelstein, P. Alla, R. Joyce, and G.G. Richard III. Physically locating wireless intruders. In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, pages 482–489, 2004.

[10] D. Aha, D. Kibler, and M. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[11] S. Al-hajeri, M. Merabti, and B. Askwith. Analysis of IPSec services. In *Proceedings of the Wireless and Optical Communications Conference*, pages 326–331, Banff, Canada, July 2003. ACTA Press.

[12] Wi-Fi Alliance. Deploying Wi-Fi Protected Access (WPA) and WPA2 in the Enterprise. http://www.wi-fi.org/getfile.asp?f=WPA-02-27-05-WPA-WPA2-WhitePaper.pdf, 2005 March. Accessed in January 2006.

[13] Wi-Fi Alliance. Wi-Fi Alliance Introduces Next Generation of Wi-Fi Security. http://www.wifialliance.com/OpenSection/ReleaseDisplay.asp, 2004 September. Accessed in January 2006.

[14] WiFi Alliance. Wi-fi protected access (WPA) enhanced security implementation based on IEEE p802.11i standard, version 3.1, August 2003.

[15] Wireless Ethernet Compatibility Alliance. 802.11b Wired Equivalent Privacy (WEP) Security. http://webpage.pace.edu/zf76248n/report.html, 2001. Accessed in January 2006.

[16] W.A. Arbaugh, N. Shankar, and Y.C. Justin Wan. Your 802.11 wireless network has no clothes. *IEEE Wireless Communications*, pages 44–51, 2002.

[17] Infrared Data Association. IrDA Infrared Mobile Communications specification v1.1. http://www.irda.org/displaycommon.cfm?an=1subarticlenbr=7, 1999. Accessed in January 2006.

[18] Telecommunications Industry Association. Introduction to cdma2000 Spread Spectrum Systems - TIA-2000.1-D. http://www.tiaonline.org/standards/sfg/imt2k/cdma2000/, July 2004. Accessed in January 2006.

[19] J.S. Balasubramaniyan, J.O. Garcia-Fernandez, D. Isacoff, E. Spafford, and D. Zamboni. An architechture for intrusion detection using autonomous agents. Technical report, COAST Laboratory Purdue University, 1998.

[20] M. Barbeau, J. Hall, and E. Kranakis. Detecting Impersonation Attacks in Future Wireless and Mobile Networks. In *Proceedings of MADNES 2005 - Workshop on Secure Mobile Ad-hoc Networks and Sensors - Held in conjunction with ISC'05*, Singapore, September 20-22 2005. SVLNCS.

[21] M. Barbeau and J-M. Robert. Perfect identity concealment in UMTS over radio access links. In *Proceedings of the Wireless and Mobile Computing, Networking and Communications*, pages 72–77, Montreal, Canada, August 2005.

[22] R. Barber. Security in a mobile world - is Bluetooth the answer? *Computers and Security*, 20:374–379, 2001.

[23] T. Bass. Intrusion detection systems and multisensor data fusion. *Communications of the ACM*, 43:99–105, 2000.

[24] N. Borisov, I. Goldberg, and D. Wagner. Intercepting mobile communications: The insecurity of 802.11. In *Proceedings of the SIGMOBILE*, pages 180–188, Rome, Italy, July 2001. ACM.

[25] A. Bria, F. Gessler, O. Queseth, R. Stridh, M. Unbehaun, J. Wu, and J. Zander. 4th-Generation Wireless Infrastructures: Scenarios and Research Challenges. *Personal Communications*, 2001.

[26] M. Briceno, I. Goldberg, and D. Wagner. An implementation of the GSM A3A8 algorithm. http://www.scard.org/gsm/a51.html, 1999. Accessed January 2006.

[27] N. Cam-Winget, R. Houseley, D. Wagner, and J. Walker. Security Flaws in 802.11 Data Link Protocols. *Communications of the ACM*, 46(5):35–39, May 2003.

[28] J. Cannady and J. Harrell. A comparative analysis of current intrusion detection technologies. Presented at Technology in Information Security Conference (TISC), 1996.

[29] C.F. Chiasserini and A. Ganz. Security in Wireless LAN, Draft of Wireless LAN lab., 1995.

[30] M.K. Chirumamilla and B. Ramamurthy. Agent based intrusion detection and response system for wireless LANs. In *IEEE International Conference on Communications (ICC'03)*, volume 1, pages 492–496. IEEE, May 2003.

[31] H. Choe, C.E. Poole, A.M. Yu, and H.H. Szu. Novel identification of intercepted signals from unknown radio transmitters. *SPIE*, 2491:504–516, 1995.

[32] Cisco. Overview: Wireless LAN Security. http://www.cisco.com/warp/public/cc/pd/witc/ao350ap, 2001. Accessed in January 2006.

[33] J.G. Cleary and I.H. Witten. Data compression using adaptive coding and partial string matching. *IEEE Transactions on Communications*, 32(4):396–402, April 1983.

[34] Compaq. Compaq Fraud Management System v7.0. h18000.www1.hp.com/info/SP6140/SP6140PF.pdf, October 2001. Accessed in January 2006.

[35] D. Coppersmith and M. Jakobsson. Almost optimal hash sequence traversal. In *Proceedings of the Fifth Conference on Financial Cryptography*, pages 102–119. IEEE, February 2002.

[36] T. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms.* McGraw-Hill, second edition, 2001.

[37] M.H.A Davis. *Markov Models and Optimization.* Chapman and Hall, 1953.

[38] J. L. DeBoer. Digital matrix. http://home.attbi.com/ digitalmatrix/airsnare/, January 2003. Accessed in July 2005.

[39] M. Delio. Wireless networks in big trouble. Technical note by AirSnort, August 2001. http://wired-vig.wired.com/news/wireless/0,1382,46187,00.html - accessed in July 2005.

[40] D.F.Specht. Probabilistic neural networks for classification mapping or associative memory. In *IEEE International Conference on Neural Networks*, pages 525–532. IEEE, 1988.

[41] P. Drabwell. Bluetooth Security - Fact or Fiction? *Lecture Notes in Computer Science*, 2467:221–228, 2002.

[42] R. Droms and W. Arbauth. Authentication for DHCP Messages. RFC 3118, June 2001. Nework Working Group.

[43] D. Dubie. Surveying security on wireless LANs. http://www.itworldcanada.com, September 2002. Accessed in September 2005.

[44] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* J. Wiley, New York, 1973.

[45] K.J. Ellis and N. Serinken. Characteristics of radio transmitter fingerprints. *Radio Science*, 36:585–597, 2001.

[46] C. Ellison. Exploiting and protecting 802.11b wireless networks. *PC Magazine*, September 2001.

[47] Black Alchemy Enterprises. Black Alchemy Weapons Lab: Fake AP, October 2002. http://www.blackalchemy.to/Projects/fakeap/fake-ap.html - accessed in July 2005.

[48] Ernst and Young. The necessity of rogue wireless device detection. White Paper, 2004.

[49] ETSI. Telecommunications and internet protocol harmonization over networks TIPHON release 4; protocol framework definition; methods and protocols for security; part 1: Threat analysis. Technical Specification ETSI TS 102 165-1 V4.1.1, 2003.

[50] L. Fausett. *Fundamentals of Neural Networks - Architectures, Algorithms and Applications.* Prentice Hall PTR, 1994.

[51] R. Filjar and S. Desic. Architecture of the automatic position reporting system (aprs). In *Proceedings of 46th International Symposium on Electronics in Marine*, pages 331–335, June 2004.

[52] D. Fisher. Wireless spec no security elixir. *eWeek*, 19:12, 2002.

[53] M. Flament, F. Gessler, F. Lagergren, O. Queseth, R. Stridh, M. Unbedaun, J. Wu, and J. Zander. An Approach to 4th Generation Wireless Infrastructures - Scenarios and Key Issues. In *Proceedings of the 49th Conference on Vehicular Technology*, pages 1742–1746, Houston Texas, July 1999.

[54] Working Group for Wireless Local Area Networks. IEEE Standard for Wireless LAN MAC and PHY Specifications. http://standards.ieee.org/wireless, 1997. Accessed in January 2006.

[55] Working Group for Wireless Local Area Networks. IEEE 802.11 Standard, Wireless LAN Standard. http://standards.ieee.org/getieee802/download/802.11-1999.pdf, 1999. Accessed in January 2006.

[56] R. Gennaro and P. Robatgi. How to Sign Digital Streams. In *Proceedings of the 1997 Conference on Advances in Cryptography*, pages 180–197, 1997.

[57] J. Girard. WLANs demand a strong security watch. http://techrepublic.com.com/5100-1035-11-5084890.html, October 2003. Accessed in January 2006.

[58] M. Gray. How to crack, compromise and circumvent WLAN security. webcast from SANS, September 2005. Chief Technology Officer at Newbury Networks.

[59] C. Grecas, S. Maniatis, and I. Venieris. Introduction of the asymmetric cryptograpy in GSM, GPRS, UMTS, and its public key infrastructure integration. *Mobile Networks and Applications*, 8:145–150, 2003.

[60] Bluetooth Special Interest Group. Bluetooth Protocol Architecture. http://www.bluetooth.org, 1999. Accessed in January 2006.

[61] Bluetooth Special Interest Group. Bluetooth specification v1.1, baseband specification. http://www.bluetooth.org/spec, 2001. Accessed in January 2006.

[62] Bluetooth Special Interest Group. Specification of the Bluetooth System, Profiles, Generic Object Exchange Profile. http://www.bluetooth.org/spec, 2001. Accessed in January 2006.

[63] Bluetooth Special Interest Group. Specification of the Bluetooth System, volume 1. http://www.bluetooth.org/spec, 2001. Accessed in January 2006.

[64] Bluetooth Special Interest Group. Specification of the Bluetooth System, Volume 2. http://www.bluetooth.org/spec, 2001. Accessed in January 2006.

[65] Bluetooth Special Interest Group. Bluetooth Security White Paper. http://www.bluetooth.org, 2002. Accessed in January 2006.

[66] CDMA Development Group. 3g-cdma2000. http://www.cdg.org/technology/3g.asp. Accessed in January 2006.

[67] HomeRF Working Group. A Comparison of Security in HomeRF versus IEEE802.11b. www.cazitech.com/HomeRF%20WP%20-%20Security.PDF, 2001. Accessed in January 2006.

[68] Trusted Computing Group. Specification version 1.2 revision 85. https://www.trustedcomputinggroup.org/groups/tpm/mainP1DP-rev85.zip, November 2004. Accessed in January 2006.

[69] C.T. Hager and S.F. Midkiff. An analysis of Bluetooth security vulnerabilities. In *Proceedings of the Conference on Wireless Communications and Networking*, volume 3, pages 1825–1831. IEEE, March 2003.

[70] C.T. Hager and S.F. Midkiff. Demonstrating vulnerabilities in Bluetooth security. In *Proceedings of the Global Telecommunications Conference*, volume 3, pages 1420–1424. IEEE, December 2003.

[71] J. Hall, M. Barbeau, and E. Kranakis. Detection of Transient in Radio Frequency Fingerprinting using Signal Phase. In *Proceedings of the Wireless and Optical Communications Conference*, pages 13–18, Banff, Canada, July 2003. ACTA Press.

[72] J. Hall, M. Barbeau, and E. Kranakis. Enhancing intrusion detection in wireless networks using radio frequency fingerprinting. In *Proceedings of the 3rd IASTED International Conference on Communications, Internet and Information Technology (CIIT)*, pages 201–206, St. Thomas, U.S. Virgin Islands, November 2004.

[73] J. Hall, M. Barbeau, and E. Kranakis. Anomaly-based intrusion detection using mobility profiles of public transportation users. In *Proceedings of the Wireless and Mobile Computing, Networking and Communications*, pages 17–24, Montreal, Canada, August 2005.

[74] J. Hall, M. Barbeau, and E. Kranakis. Radio Frequency Fingerprinting for Intrusion Detection in Wireless Networks. *IEEE Transactions on Dependable and Secure Computing*, 2005. Submitted in 2005.

[75] J. Hall, M. Barbeau, and E. Kranakis. Detecting rogue devices in Bluetooth Networks using Radio Frequency Fingerprinting. In *Proceedings of the IASTED International Conference on Communications and Computer Networks*, page 6, Lima, Peru, October 2006.

[76] J. Hall, M. Barbeau, and E. Kranakis. Using Mobility Profiles for Anomaly-based Intrusion Detection in Mobile Networks. Network and Distributed System Security Symposium (NDSS) Pre-conference Workshop on Wireless and Mobile Security, 2005 April. San Diego, US.

[77] L. Harn and W-J. Hsin. On the security of wireless network access with enhancements. In *Proceedings of the WiSE'03 conference*, pages 88–95, San Diego, California, USA, September 2003. ACM.

[78] J.J. Harrington and D.A. Pritchard. Concepts and Applications of Wireless Security Systems for Tactical, Portable, and Fixed Sites. In *Proceedings of the 31st Annual International Carnahan Conference on Security Technology*, pages 133–139. IEEE, October 1997.

[79] S. Harris. 802.11 Security. *Windows 2000 Magazine*, pages 47–51, 2001.

[80] Hewlett-Packard. Hp Fraud Management System. http://h71028.www7.hp.com/enterprise/downloads/solutionbrief.pdf, February 2003. Accessed in January 2006.

[81] T. Higuchi. Approach to an irregular time series on the basis of the fractal theory. *Physica D*, 31:277–283, 1988.

[82] V. Himmelsbach. Pushing RFID from the top down: Wal-mart starts a game of tag. www.itbusiness.ca, April 2005. Computing Canada.

[83] R.D. Hippenstiel and Y. Payal. Wavelet based transmitter identification. In *International Symposium on Signal Processing and its Applications*, Gold Coast Australia, August 1996.

[84] S. Hofmeyr, S. Forrest, and A. Somayaji. Intrusion detection using sequences of system calls. *Computer Security*, 6(3):151–180, 1998.

[85] W-C. Hsieh, C-C. Lo, J-C. Lee, and L-T. Huang. The implementation of a proactive wireless intrusion detection system. In *Proceedings of the Fourth International Conference on Computer and Information Technology (CIT'04)*, pages 581–586, September 2004.

[86] A. Huger. The honeynet project. http://www.honeynet.org/misc/project.html, April 2005. Accessed in January 2006.

[87] A. Hunter. Feature Selection using Probabilistic Neural Networks. *Neural Computing and Applications*, 9:124–132, 2000.

[88] IEEE. IEEE 802.16, Standard for Local and Metropolitan Area Networks, part 16, "air interface for fixed broadband wireless access systems. IEEE Press, April 2002.

[89] AirDefense Inc. http://www.airdefense.net. Accessed in February 2004.

[90] European Telecommunications Standards Institute. Digital cellular telecommunications system: security aspects, 1999. version 8.0.1.

[91] M. Jakobsson. Fractal hash sequence representation and traversal. In *Proceedings of the International Symposium on Information Theory*, pages 437–444. IEEE, July 2002.

[92] M. Jakobsson and S. Wetzel. Security Weaknesses in Bluetooth. *Lecture Notes in Computer Science*, pages 176–191, 2001.

[93] D.D. Janowski and S. Chang. The lay of the wireless LAN. www.cisco.com/global/BE/press/pdfs/pcmag52102final1.pdf, May 2002. Accessed in January 2006.

[94] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice Hall, 1998.

[95] Jr. Joseph, F. Hair, E. Anderson, W. Black, and R. Tatham. *Multivariate Data Analysis.* Prentice Hall PTR, 1998.

[96] M. Just, E. Kranakis, and T. Wan. Resisting malicious packet dropping in wireless ad-hoc networks using distributed probing. In *Proceedings of 2nd Annual Conference on Adhoc Networks and Wireless (ADHOCNOW'03)*, pages 151–163, Montreal, Canada, 2003.

[97] T. Karygiannis and L. Owens. Wireless Network Security: 802.11, Bluetooth and Handheld Devices. Technical report, NIST, 2002.

[98] H.G. Kayacik, A.N. Zincir-Heywood, and M.I. Heywood. On the Capability of an SOM based Intrusion Detection System. *Neural Networks*, pages 1808–1813, July 2003.

[99] W. Kinsner. A unified approach to fractal and multifractal dimensions. Technical report, University of Manitoba, 2001.

[100] C. Ko, M. Ruschitzka, and K. Levitt. Execution Monitoring of Security-Critical Programs in Distributed Systems: A Specification-based Approach. In *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, pages 175–187, May 1997.

[101] T. Kohno, A. Broido, and KC Claffy. Remote physical device fingerprinting. In *Proceedings of the 2005 IEEE Symposium on Security and Privacy*, May 2005.

[102] P. Kyasanur and N.H. Vaidya. Detection and handling of MAC layer misbehaviour in wireless networks. Technical report, Digital Equipment Corporation, 2002.

[103] A. Laing. The Security Mechanism for IEEE 802.11 Wireless Networks. http://rr.sans.org/wireless/, November 24 2001. Accessed in January 2006.

[104] T. Lane and C.E. Brodley. Temporal Sequence Learning and Data Reduction for Anomaly Detection. *ACM Transactions on Information and System Security*, 2(3):295–331, August 1999.

[105] J. Leyden. Tool dumbs down wireless hacking. *The Register*, 2001.

[106] Y-X. Lim, T. Schmoyer, J. Levine, and H.L. Owen. Wireless intrusion detection and response. In *Proceedings of the IEEE Information Assurance Workshop on Systems, Man and Cybernetics*, pages 68–75, June 2003.

[107] Prism Holdings Limited. Prism develops cryptographic protection against SIM fraud. http://www.itweb.co.za/office/prism/0302050753.htm, February 2003. Accessed in January 2006.

[108] C.A. Lowry, W.H. Woodall, C.W. Champ, and S.E. Rigdon. A multivariate exponentially weighted moving average control chart. *Technometrics*, 34:46–53, February 1992.

[109] W. Ma and Y. Fang. A new location management strategy based on user mobility pattern for wireless networks. In *Proceedings of the 27th Annual Conference on Local Computer Networks*, 2002.

[110] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. University of California Press, 1967.

[111] C. Madson and R. Glenn. The use of HMAC-MD5-96 within ESP and AH. Request for Comments 2403 - November 1998, 1998. Internet Engineering Task Force.

[112] C. Madson and R. Glenn. The use of HMAC-SHA-1-96 within ESP and AH. Request for Comments 2404 - November 1998, 1998. Internet Engineering Task Force.

[113] Air Magnet. http://www.airmagnet.com, January 2003. Accessed in July 2005.

[114] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1999.

[115] J. Markoulidakis, G. Lyberopoulos, D. Tsirkas, and E. Sykas. Evaluation of location area planning scenarios in future mobile telecommunication systems. *Wireless Networks*, 1:17–29, February 1995.

[116] P. McFedries. Bluetooth Cavities. *IEEE Spectrum*, 42(6):88–89, June 2005.

[117] P.C. Mehta. Wired Equivalent Privacy Vulnerability. http://rr.sans.org/wireless/equiv.php, 2001. Accessed in January 2006.

[118] B.A. Miller. *Bluetooth Revealed.* Prentice Hall PTR, 2001.

[119] A. Mishra and W.A. Arbaugh. An Initial Security Analysis of the IEEE 802.1x Standard. Technical report, University of Maryland, 2002.

[120] M. Mitchell. *An Introduction to Genetic Algorithms (Complex Adaptive Systems).* The MIT Press, February 1998.

[121] B. Morin and H. Debar. Correlation of intrusion symptoms: an application of chronicles. In *Proceedings of the International Conference on Recent Advances in Intrusion Detection*, pages 94–112, Berlin Heidelberg, 2003.

[122] P. Morrissey and D. Advani. Sneak an AiroPeek at WLAN Stats. http://www.networkcomputing.com/1311/1311f3.html, May 2002. Accessed in January 2006.

[123] M. Mouly and M-B. Pautet. *The GSM System for Mobile Communications.* Telecom Publishing, 1992.

[124] Newbury Networks. The power of location-based WLAN security and management. White Paper, April 2005.

[125] L. Nguyen, R. Safavi-Naini, W. Susilo, and T. Wysocki. Secure authorization, access control and data integrity in Bluetooth. In *Proceedings of the 10th International Conference on Networks*, pages 428–433. IEEE, 2002.

[126] R.K. Nichols and P.C. Lekkas. *Wireless Security Models, Threats, and Solutions.* McGraw-Hill, 2002.

[127] C. Nobel. New options help sort out 802.11. *eWeek*, 19:9–10, 2002.

[128] Newsfactor Magazine Online. New qualcomm chipsets to enable cellular and wi-fi phones. http://blog.newsfactor.com/?b=67, September 2005. Accessed in September 2005.

[129] K. Pahlavan and P. Krishnamurthy. *Principles of Wireless Networks.* Prentice Hall PTR, 2002.

[130] A. Papoulis. *The Fourier Integral and Its Applications.* McGraw-Hill, New York, 1962. 198-201.

[131] S.H. Park, A. Ganz, and Z. Ganz. Security protocol for IEEE 802.11 wireless local area network. *Mobile Networks and Applications*, 3:237–246, 1998.

[132] C. Peikari and S. Fogie. *Maximum Wireless Security.* Sams, 2002.

[133] D.L. Pepyne, Y-C. Ho, and Q. Zheng. SPRiNG Synchronized Random Numbers for Wireless Security. *Wireless Communications and Networking*, pages 2027–2032, 2003.

[134] D. Percival and A.T. Walden. *Wavelet Methods for Time Series Analysis.* Cambridge University Press, 2002.

[135] S. Phan. Creating wireless security without WEP. http://www.networkmagazineindia.com/200111/focus2.htm, 2001. Accessed in January 2006.

[136] L. Phifer. Wireless Privacy: An Oxymoron ? http://www.wifiplanet.com/columns/article.php/786641, April 2001. Accessed in January 2006.

[137] P. Porras and P. Neumann. EMERALD: Event monitoring enabling responses to anomalous live disturbances. In *Proceedings of the Twentieth National Information Systems Security Conference*, pages 353–365, 1997.

[138] BSNL Portal. Phones can use Wi-Fi, cellular networks. http://www.bsnl.in/telecomtrends.asp?intNewsId=53442, August 2005. Accessed in September 2005.

[139] B. Potter. Wireless Intrusion Detection. *Wireless Security*, 2004(4):4–5, 2004.

[140] J.G. Proakis and D.G. Manolakis. *Digital Signal Processing.* Prentice Hall PTR, 1996.

[141] J. Quirke. Security in the GSM system. http://www.ausmobile.com, May 2004. Accessed in January 2006.

[142] P. Reichl R. Buschkes, D. Kesdogan. How to increase security in mobile networks by anomaly detection. In *Proceedings of the Computer Security Applications Conference*, pages 3–12, Phoenix AZ USA, Dec. 1998.

[143] L.R. Rabiner and B.H. Juang. *An introduction to hidden markov models.* Prentice Hall PTR, 1986.

[144] P.C. Kishore Raja and Suganthi. VLSI approach to wireless security mechanism. In *Proceedings of the IEEE International Conference on Personal Wireless Communications (ICPWC)*, pages 429–433, Jan 2005.

[145] S. Ravi, A. Raghunathan, and N. Potlapally. Securing Wireless Data: System Architecture Challenges. In *Proceedings of the 15th International Symposium on System Synthesis*, pages 195–200, Kyoto, Japan, October 2002. ACM.

[146] P. Reddy, V. Krishnan, K. Zhang, and D. Das. Authentication and Authorization of Mobile Clients in Pubic Data networks. *Lecture Notes in Computer Science*, pages 115–128, 2002.

[147] M.J. Riezenman. Cellular security: better, but foes still lurk. *IEEE Spectrum*, 37(6):39–42, June 2000.

[148] S.J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall PTR, 2002.

[149] D. Samfat and R. Molva. IDAMN: an intrusion detection architecture for mobile networks. *IEEE Journal on Selected Areas in Communications*, 15(7):1373–1380, Sept. 1997.

[150] R. Sandhu and P. Samarati. Access Control: Principles and Practice. *IEEE Communications*, 32(9):40–48, November 1994.

[151] J. Schiller. *Mobile Communications.* Addison-Wesley, 2000.

[152] Beyond Security. Kismet - Wirless Network Sniffer. http://www.securiteam.com/tools/5EP091F8UM.html, January 2003. Accessed in January 2006.

[153] N. Serinken and O. Ureten. Bayesian detection of Wi-Fi transmitter RF fingerprints. *Electronic Letters*, 41(6):373–374, March 2005.

[154] Information Handling Services. Electro/Telecom Industry Trends. http://electronics.ihs.com/newsletters/tele-sept3-1.jsp, September 2003. Accessed in January 2006.

[155] G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, NJ, USA, 1976.

[156] V. Sharma. Intrusion detection in infrastructure wireless LANs. *Bells Labs Technical Journal*, 8(4):115–119, 2004.

[157] D. Shaw and W. Kinsner. Multifractal Modelling of Radio Transmitter Transients for Classification. In *Proceedings of the Conference on Communications, Power and Computing*, pages 306–312, Winnipeg Manitoba, May 1997. IEEE.

[158] R. Shorey and B.A. Miller. The Bluetooth Technology: Merits and Limitations. In *Proceedings of the International Conference on Personal Wireless Communications*, pages 80–84. IEEE, 2000.

[159] W. Simpson. PPP Challenge Handshake Authentication Protocol, 1996.

[160] B. Sklar. *Digital Communications: Fundamentals and Applications.* Prentice-Hall, 1988.

[161] IEEE Computer Society. IEEE Std 802.11i-2004 IEEE standard for information technology- telecommunications and information exchange between systems-local and metropolitan area networks- specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: Medium access control (MAC) security enhancements. Standard Number IEEE Std 802.11i-2004, 2004.

[162] P. Somervuo and T. Kohonen. Self-organizing Maps and Learning Vector Quantization for Feature Sequences. *Neural Processing Letters*, 10:151–159, 1999.

[163] J. Spencer. Use of an artificial neural network to detect anomalies in wireless device location for the purpose of intrusion detection. In *Proceedings of the IEEE SoutheastCon*, pages 686–691, April 2005.

[164] W. Stallings. *Cryptography and Network Security.* Prentice-Hall, Inc., 1999.

[165] W. Stallings. *Data and Computer Communications.* Prentice-Hall, Inc., 2000.

[166] D.R. Stinson. *Cryptography Theory and Practice.* CRC Press LLC, 1995.

[167] B. Sun and F. Yu. Mobility-based anomaly detection in cellular mobile networks. In *Proceedintgs of the International Conference on WiSe 04*, pages 61–69, Philadelphia, Pennsylvania, USA, 2004.

[168] N. Sutton. Panel urges suppliers to start developing their RFID strategies. www.itbusiness.ca, April 2005. Computing Canada.

[169] TamoSoft. An advanced wireless packet sniffer can help you get the full picture of your 802.11 WLAN traffic. http://www.tamos.com/products/commwifi/wifi-sniffer.htm. Accessed in January 2006.

[170] P. Tao, A. Rudys, A.M. Ladd, and D.S. Wallach. Wireless LAN location-sensing for security applications. In *Proceedings of the Workshop on Wireless Security*, pages 11–20. ACM Press, 2003.

[171] J. Taschek. How much wireless security is enough? *eWeek*, 19:62, 2002.

[172] O.H. Tekbas and N. Serinken. Transmitter Fingerprinting from Turn-on Transients. In *Proceedings of the NATO RTO Sensors and Electronics Technology Panel Symposium on Passive and LPI Radio Frequency Sensors*, Warsaw Poland, April 2001.

[173] O.H. Tekbas, O. Ureten, and N. Serinken. Improvement of transmitter identification system for low SNR transients. *Electronic Letters*, 40(3):182–183, February 2004.

[174] Financial Times. CDMA phone easy to clone. LexisNexis(TM) Print Delivery - lexisnexis@prod.lexisnexis.com, May 2005. Accessed in January 2006.

[175] Financial Times. Mobile Cloning. LexisNexis(TM) Print Delivery - lexisnexis@prod.lexisnexis.com, March 2005. Accessed in January 2006.

[176] Financial Times. SIM card cloning case detected. LexisNexis(TM) Print Delivery - lexisnexis@prod.lexisnexis.com, February 2005. Accessed in January 2006.

[177] Financial Times. What is mobile cloning. LexisNexis(TM) Print Delivery - lexisnexis@prod.lexisnexis.com, May 2005. Accessed in January 2006.

[178] J. Toonstra and W. Kinsner. Transient Analysis and Genetic Algorithms for Classification. In *Proceedings of the Conference on WESCAN*. IEEE, 1995.

[179] M. Traskback. Security of Bluetooth - An overview of Bluetooth security. http://www.cs.hut.fi/Opininot/Tik-86.174/Bluetooth-Security.pdf.

[180] O. Ureten and N. Serinken. Detection of radio transmitter turn-on transients. *Electronic Letters*, 35(23):1996–1997, November 1999.

[181] O. Ureten and N. Serinken. Detection, Characterization and Classification of Radio Transmitter Turn-on Transients. In *Proceedings of the NATO Advanced Study*, pages 611–616. Institute on Multisensor Data Fusion, Kluwer Academic Publishers, 2002.

[182] J. Vainio. Bluetooth Security. http://www.niksula.cs.hut.fi/ jiitv/bluesec.html, 2000. Accessed in January 2006.

[183] J.R. Walker. Unsafe at Any Key Size: An Analysis of the WEP Encapsulation. http://grouper.ieee.org/groups/802/11/Documents/DocumentHolder/0-362.zip, 2000. Accessed in January 2006.

[184] S. Weatherspoon. Overview of IEEE 802.11b Security. White paper by Network Communications Group, 2000.

[185] Wikipedia. RFID. http://en.wikipedia.org/wiki/RFID, January 2006. Accessed in January 2006.

[186] J. Williams. The IEEE 802.11b Security Problem, Part 1. *IT Professional*, 3(6):91–96, 2001.

[187] J. Williams. Providing for Wireless LAN Security, Part 2. *IT Professional*, 4(6):44–48, 2002.

[188] C. Wingert and M. Naidu. CDMA 1XRTT security overview. White paper by Qualcomm, August 2002.

[189] V. Wong and V. Leung. Location management for next generation personal communications networks. *IEEE Network*, pages 18–24, Sept. 2000.

[190] K. Wu, J. Harms, and E.S. Elmallah. Profile-based protocols in wireless mobile ad hoc networks. In *Proceedings of the 26th Annual Conference on Local Computer Networks*, pages 568–575. IEEE Computer Society, November 2001.

[191] T. Wu. The secure remote password protocol. In *Proceedings of the Internet Society Network and Distributed System Security Symposium*, pages 97–111, 1998.

[192] T.G. Xydis and S. Blake-Wilson. Security Comparison: Bluetooth Communications vs. 802.11. Technical report, Bluetooth Security Experts Group, 2002.

[193] H. Yang, L. Xie, and J. Sun. Intrusion detection for wireless local area network. In *Proceedings of the Canadian Conference on Electrical and Computer Engineering 2004*, pages 1949–1952, Niagara Falls, Canada, May 2004.

[194] N. Ye, Q. Chen, S.M. Emran, and S.Vilbert. Hotelling's t^2 multivariate profiling for anomaly detection. In *Proceedings of the Workshop on Information Assurance and Security*, pages 180–186, West Point NY, June 2000. IEEE.

[195] F. Yu and V.C.M. Leung. Mobility-based predictive call admission control and bandwidth reservation in wireless cellular networks. *Elsevier Computer Networks*, 38(5):577–589, April 2002.

[196] Y. Zhang and W. Lee. Intrusion detection in wireless ad-hoc networks. In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking*, pages 275–283, 2000.

[197] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5):530–536, September 1978.

[198] J. Zuidweg and H. Zuidweg. *Next Generation Intelligent Networks.* Artech House, 2002.