Mobile Netw Appl (2007) 12:93–110
DOI 10.1007/s11036-006-0009-6

On the Impact of Quality of Protection in Wireless Local Area Networks with IP Mobility

Avesh K. Agarwal · Wenye Wang

Published online: 22 November 2006

Abstract Wireless local area networks (LANs) are vulnerable to malicious attacks due to their shared medium in unlicensed frequency spectrum, thus requiring security features for a variety of applications even at the cost of quality of service (QoS). However, there is very little work on investigating to what extent system performance is affected by security configurations with respect to mobility scenarios, heterogeneous networks, and different applications. In order to exploit the full potential of existing security solutions, we present a detailed experimental study to demonstrate the impacts of security features on performance by integrating cross-layer security protocols in a wireless LAN testbed with IP mobility. We introduce a *quality of protection (QoP)* model to indicate the benefits of security protocols and then measure the performance cost of security protocols in terms of authentication time, cryptographic overhead and throughput. Our measurements demonstrate that the effects of security protocols on QoS parameters span a wide range; for example, authentication time is between 0.11 and 6.28 s, which can potentially affect packet loss dramatically. We also find that for the same security protocol throughput in non-roaming scenarios can be up to two times higher than that in roaming scenarios. However, some protocols are robust against mobility with little variation in system performance; thus, it is possible to provision steady service by choosing security protocols when users' mobility pattern is unknown. Furthermore, we provide observations on cross-layer security protocols and suggestions to the design of future security protocols for real-time services in wireless LANs.

Keywords wireless local area networks · quality of service · quality of protection · security protocols · mobile IP · performance analysis

1 Introduction

Rapid and widespread deployment of Wireless Local Area Networks (LANs) offers great convenience to access the Internet via wireless devices such as laptop computers and portable data assistants (PDAs) users. In the early stages of deployment [1], wireless LANs were used for non-critical applications such as chat applications, e-mails, web-browsing and so on for proprietary networks such as campus or enterprise networks. Nowadays, many organizations and individuals are using wireless LANs as public access networks for transmitting sensitive and critical data [2]. Therefore, security is of utmost concern in wireless LANs because malicious users can intercept and eavesdrop data in transit on shared and broadcast medium [3]. In response to the demand for security, several security protocols such as wired equivalent privacy (WEP), 802.1x port access control with extensible authentication protocol (EAP) support are designed to address security issues [4–6]. Moreover, IP security protocol (IPsec) used in wired networks is also considered as an alternative for wireless networks as well [7].

This work is supported by the National Science Foundation (NSF) under grant NR-0322893 and the Center for Advanced Computing and Communication (CACC) #04-08.

A. K. Agarwal · W. Wang (✉)
Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695, USA
e-mail: wwang@eos.ncsu.edu

However, secure communications are not gained for free in wireless networks because all security protocols require transmission of users' credentials for identity verification, control messages, as well as data encryption/decryption. With the increasing demand for mobile applications [8], while security protocols are expected to be an integral part of network protocols, they should not undermine the usage of wireless networks because of their effects on system performance. Although it is clearly that there is a trade-off between security and quality of service (QoS) based on either qualitative discussions or intuitive observations, there is very little work on investigating to what extent system performance can be impacted by security protocols with respect to mobility scenarios, heterogeneous networks, and different applications. For example, it is well-known that authentication may cause additional delay which can lead to degradation in QoS; whereas, there is almost no literature showing authentication time in practice and explain how these extra delay may affect user mobility support in wireless LANs. Compared to the extensive studies in provisioning QoS, our understanding of the performance aspects of security protocols is quite limited. As a result, this critical gap in understanding the impacts of security features imposes a hurdle for optimizing system performance when security concerns are present. To a certain extent, the lack of knowing the impacts of security protocols in wireless networks makes the main challenges unclear in developing security protocols for mobile applications.

To evaluate the performance impacts of security protocols, our approach is to take measurements in a testbed because experimental study not only provides realistic results, but more importantly, many issues, such as processing time of mobile devices and interaction of cross-layer protocols, cannot be accurately modeled in simulations or analytical studies. Similarly, several experimental studies have been carried out in various network environments [7, 9, 10]. However, there are several major limitations regarding previous efforts. First, existing studies have focused on improving cryptographic aspects of security protocols in a few network scenarios [4, 11, 12], while not providing detailed quantification of the performance overhead associated with security protocols [7, 9, 10]. Second, IP mobility support is not considered in these studies. Since most of the wireless applications are IP-based such as email and web-browsing service and with the increasing demand for mobile applications [13], it is necessary to conduct experimental study in wireless LANs with IP mobility. Third, previous works have explored the advantages and disadvantages of security protocols in stand-alone mode, focusing on one particular protocol in study. Note that there exist security protocols at different network layers, it is inevitable and intuitive to explore the effects of cross-layer protocols associated with integrated security features at different layers, which will be helpful in understanding the applicability of a particular *policy or feature* to real-time networks while maintaining the required QoS at the same time.

Moreover, there has been a surge of heterogeneous devices recently, such as iPAQs, SharpZaurus and laptop computers using different hardware and software platforms, being used by mobile users in wireless LANs. The diversity of wireless devices is the driving force behind heterogeneous wireless networks. Existing studies without using heterogeneous devices in the experiments and are, therefore, restricted in their relevance to practice [9]. Besides, some of the existing studies are performed at the Windows platform [10]. However, Linux open source community is growing very quickly, leading free source code available to every user. Thus, it is important to evaluate the performance impact of security protocols by using various wireless devices and open source codes in experimental studies so that large community of users can be benefited.

Therefore, we aim to study the cross-layer integration of security protocols to gain a deeper understanding about the trade-offs between performance overhead and security benefits in wireless LANs with IP mobility, by addressing the aforementioned limitations in a variety of network environments. In order to achieve these goals, we setup a real-time experimental testbed, which is a miniature of existing wireless networks to ensure that our experimental scenarios are consistent with typical deployment of wireless LANs, while using mobile IP for roaming support [14–16]. We use iPAQ, SharpZaurus, laptop and desktop machines each equipped with wireless cards to create heterogeneous environments. More than 120 experiments are designed, which include various network elements such as stand-alone security protocols and hybrid security protocols for cross-layer integration; roaming and non-roaming scenarios; heterogeneous network environments; and TCP/UDP traffic streams. One of the main challenges is the implementation complexity associated with configurations of cross-layer security protocols in a real-time testbed.

In addition, we introduce a *quality of protection (QoP) model* to demonstrate the benefits of integrating cross-layer security protocols on system performance. The model consists of a *utility function* which assigns weights to various security features, such as authentication, confidentiality, mutual authentication, data integrity, and non-repudiation, based on their imple-

mentations in a network. Specifically, the utility function offers a *micro view* of the benefits provided by a security protocol. The QoP model also consists of an *additive reward model* to quantify the cumulative benefits provided by a security protocol. The additive reward model offers a *macro view* of the benefits associated with a security protocol. Besides, authentication time, cryptographic overhead and throughput are evaluated as performance metrics evaluated in our testbed for TCP and UDP traffic streams to gain a deep understanding of the trade-off between QoS and security. Also, by performing statistical analysis, we analyze the robustness associated with each individual and cross-layer security protocol because performance robustness can be used to provide steady QoS to mobile users even when their roaming profiles are not known in advance. Our experimental results not only present a quantitative view of performance impacts of security protocols, but also they are helpful in contributing useful insights about the applicability of security protocols to real-time application and design challenges of future security protocols.

The rest of the paper is organized as follows. We describe the methodology of our experimental studies in Section 2, including the details of the testbed, network scenarios, security policies, and data acquisition. In Section 3, we introduce *QoP model* and defines utility function and additive reward model. Cost functions to analyze the interaction between security benefits and performance overhead associated with policies are presented in Section 4. Detailed performance evaluation of the experimental results is provided in Section 5. In Section 6, we present insights drawn from our experimental results, and provide suggestions on security protocols for real-time applications. We discuss about the vulnerabilities associated with individual security protocols and cross-layer integration, as well as robustness of security protocols against user mobility. Finally, Section 7 concludes the paper.

2 Methodology of experimental study on security protocols

In order to study the impact of security protocols in an IP-based wireless LAN, our methodology of this experimental work includes the following components:

- Design and implementation of an IP-based wireless LAN testbed: Establishing a wireless LAN testbed with IP mobility support is the basis of our study. Since our goal is to evaluate the performance of individual security protocols and cross-layer security protocols in heterogeneous environments, we design and setup a wireless LAN by using different mobile devices and equip them with the same protocol stack as explained in Subsection 2.1.
- Classification of network scenarios with mobility support: By taking user mobility into consideration, we classify network scenarios into non-roaming and roaming scenarios with respect to the current location of mobile users rather than the movement patterns of users because the up-to-date locations of users are the results of user mobility as described in Section 2.2.
- Design of security policies: Each security protocol involves many algorithms in providing features such as confidentiality and integrity. Thus, we design *individual* polices based on the services provided in each protocol and *hybrid* policies for cross-layer protocols in Section 2.3.
- Acquisition of experimental data: For each security feature configured in our testbed, measurements are collected in two phases as explained in Section 2.4. The first phase collects measurements during the initial negotiation of protocols. The second phase focuses on generating streams, and then collecting experimental data for different policies. The collected data are analyzed in Section 5.
- QoP model and performance metrics: One of the most challenging issues in the performance evaluation is to quantify the benefits offered by security protocols, and to define performance metrics. We introduce a *QoP model* to indicate the benefits of security features, and define performance metrics such as authentication time, cryptographic overhead, and throughput, which enables us to observe the impact of security policies on system performance as explained in Sections 3 and 4, respectively.

Based on our measurements in terms of performance metrics for each security policies and network scenario, we present experimental results and analysis, followed by observations and remarks.

2.1 IP-based wireless LAN testbed

We design and implement a wireless LAN testbed with Mobile IP for roaming and routing [14–16]. A mobile node (MN) is defined as the wireless node which is able to change its point of attachment by following the terminology in mobile IP [14–17]. This testbed is used as a platform on which we carry out various experiments. We use various hardware and mobile devices to make this testbed a heterogeneous network. Mobile devices

include iPAQ (Intel StrongARM 206 MHz), Sharp Zaurus (Intel XScale 400 MHz) and Dell Laptop (Celeron Processor, 2.4 GHz). Thus, we create a heterogeneous environment that captures mobile scenarios of wireless LANs with multiple subnets.

In each subnet, there is a home agent (HA) to which a mobile node registers its permanent IP address. In our testbed, these home agents are also gateways and their functions are implemented on Dell PC with Pentium IV 2.6 GHz. By following the definition in mobile IP, foreign agents (FA) are the gateways in a foreign network where a mobile node obtains a new IP address to access to a network. Home agents also have the functionalities of foreign agents in our testbed and they are connected to Cisco Access Points (Aironet 1200 series) to provide wireless connectivity. In addition, home agents have functions of IPsec gateways and RADIUS server for IPsec and 802.1x, respectively. An IPsec tunnel is setup between home agents to provide security over the wired segment in our testbed. End systems are Dell PC with Pentium IV 2.6 GHz, act as wired correspondent nodes in different subnets. Cisco Catalyst 1900 series is used as a network switch to provide connectivity between two subnets via the router. For all mobile devices, we use Netgear MA 311 and Lucent Orinoco Gold wireless cards for wireless connections.

For each device, a protocol stack from the medium access control (MAC) to application layer is installed. At MAC layer, WEP/802/1x is used; at network layer, IP, IPsec, and mobile IP are used; at transport layer, TCP/UDP, SSL, and TLS are used; at application layer, RADIUS authentication protocol is used. All end systems are installed with Redhat Linux 9.0 kernel 2.4.20; and other open-source software components for the protocol stack in the testbed are: (1) *FreeSwan* open source is installed on home agents and mobile nodes for IPsec functionality [18]; (2) *Xsupplicant*, which provides 802.1x client functionality, has been installed on mobile nodes [19]; (3) RADIUS server functionality has been provided by *FreeRadius* and has been installed on home agents [20]; (4) *OpenSSL* open source software is installed on home agents [21]; (5) To introduce user mobility in our network, *Mobile IP* implementation from Dynamic is installed on mobile nodes and home agents [22]. Finally, we use *Ethereal* packet analyzer for packet capturing, also *Iperf* and *ttcp* are used for generating TCP/UDP traffic streams.

2.2 Network scenarios

By taking user mobility into account, *network scenarios* can be classified into non-roaming ($\mathcal{N}$) and roaming ($\mathcal{R}$) based on a user's current location, i.e., whether a user is in its home domain or foreign domain, respectively. To make the description of scenarios clear, we assume that subnet A is the home domain for mobile nodes $A1$ and $A2$; and subnet B is the home domain for mobile nodes $B1$ and $B2$. All scenarios are demonstrated in Fig. 1. Non-roaming scenarios, represented as $\mathcal{N}$, are defined as the scenarios when both communicating mobile users are in their home domain. Following are the details of various non-roaming scenario configured in the testbed.

- *Scenario N1:* When two mobile nodes belong to the same home domain, they communicate using their home agents, e.g., the communication between $A1$ and $A2$ occurs as shown in Fig. 1a.
- *Scenario N2:* Mobile nodes communicate with their home agent that provides services to mobile clients in the network. In this case, a mobile node like mobile node $A2$ in subnet A requests service from home agent A-HA directly as shown in Fig. 1b.
- *Scenario N3:* When two participating mobile nodes are in different domains, they must communicate through their own home agents connected via the Internet. For example, mobile nodes $A1$ or $A2$ in subnet A communicates with nodes $B1/B2$ in subnet B as shown in Fig. 1c.

When at least one of two communicating mobile users is in a foreign domain, that is, outside of its home network, we refer it as *roaming scenario*, represented as $\mathcal{R}$. The following roaming scenarios are configured in our experimental testbed.

- *Scenario R1:* If one end node which is in a foreign domain, e.g., node $A2$ in subnet B in Fig. 1d, communicates with another node (e.g., $A1$) in its home domain, these two nodes are in different domains while one roaming node is outside of its home network.
- *Scenario R2:* When both nodes are in the same domain but one of them (e.g., node $B1$) is in a visiting network (e.g., subnet A) as shown in Fig. 1e, the current network (subnet A) is the foreign domain for the roaming node $B1$, whereas it is the home domain for the other node $A1$.

2.3 Design of security policies

Security policies are designed to show the security benefits provided by the integration of protocols at different layers. Various policies are configured in our testbed by combining features from different security protocols. Let $\mathcal{P} = \{P_1, P_2, \ldots, P_{12}\}$ represent the set of individual and hybrid policies configured in the testbed.

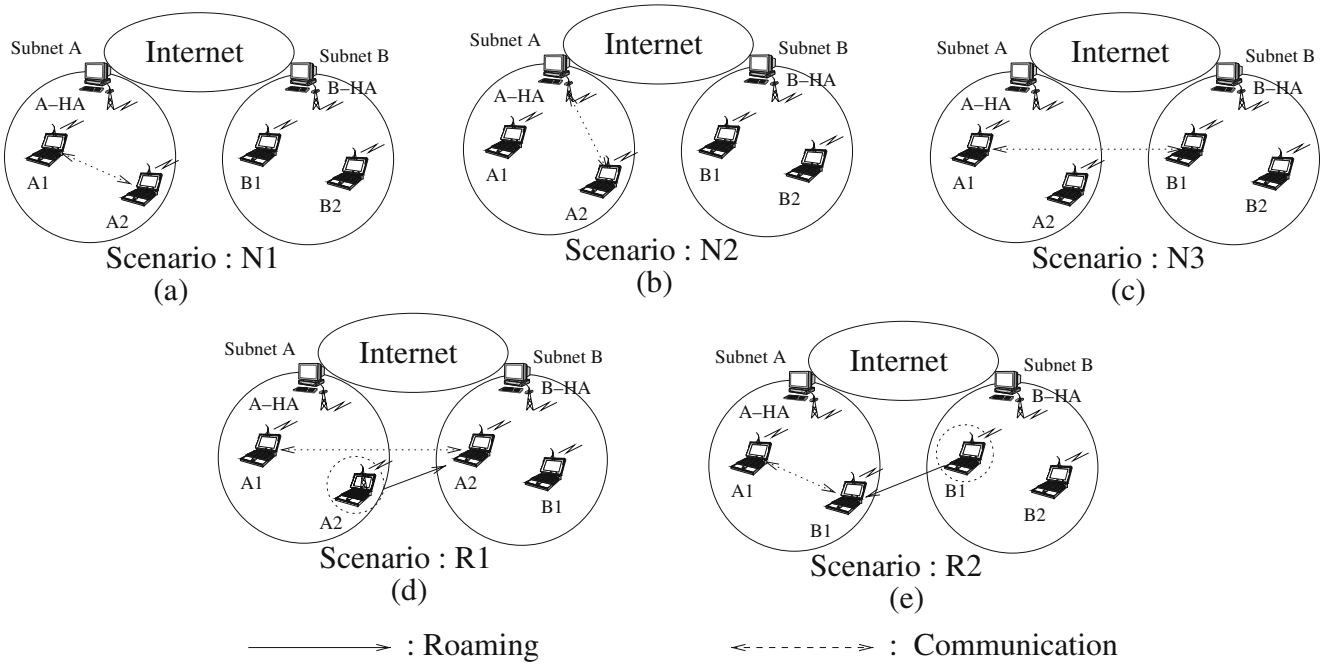

Fig. 1 Non-roaming and roaming network scenarios

Now, we explain these policies and their significance in detail.

2.3.1 *Individual security policies*

When a policy involves mechanisms in a single security protocol, it is called an *individual security policy*. "No security" means that there is no security feature enabled in the network. "No Security" policy helps us in comparing the overhead associated with others with regard to performance metrics. In the following paragraphs we discuss security policies for each security protocol.

- *WEP Policies:* WEP supports 40-bit and 128-bit encryption keys. As we observed little variations in measurements for the two key sizes, we present WEP only with 128-bit due to longer key size. Although WEP has been shown vulnerable to many attacks [4], we study WEP for two reasons. First, WEP is still being used in many networks for dynamic session keys along with other protocols such as EAP-TLS with 802.1x framework [6]. Second, comparing WEP's performance with others provides a complete study of the performance impact of existing security protocols for WLANs. P_2 is the only individual WEP policy configured in the testbed.
- *IPsec Policies:* IPsec protocol supports a large set of algorithms, and provides strong security. Since we use *Freeswan* [18] for IPsec functionality, our analysis is restricted to the security features provided by Freeswan open source implementation. Freeswan includes 3DES as an encryption algorithm, and MD5 and SHA as authentication algorithms. Since IPsec tunnel mode is considered better by providing stronger security features than IPsec transport mode, we analyze only IPsec tunnel mode in our setup. P_3 is the only individual IPSEC policy configured in the testbed.
- *802.1x Policies:* In case of 802.1x, we use RADIUS as a backend server maintaining users' credentials. EAP is used as a transport mechanism which involves MD5 and TLS modes. Although EAP-MD5 is not considered a very strong authentication implementation for WLANs [23], it can provide better security when configured with other security protocols. Therefore, we believe that inclusion of EAP-MD5 makes our study complete. Moreover, as discussing performance aspects of various security protocols is the main objective of this paper, inclusion of EAP-MD5 enables us to provide comprehensive performance measurements of the existing security protocols in WLANs. Policies P_5 and P_6 are the two 802.1x individual policies configured in the testbed.

2.3.2 *Hybrid security policies*

When policies involve mechanisms belonging to multiple protocols, they are called *hybrid security policies*.

Table 1 Features of security policies

Policy	Security policies	Authentication	Confidentiality	Data integrity	Non repudiation	Mutual auth
P_1	No security (NS)					
P_2	WEP-128 bit key	Y	Y			
P_3	IPsec-3DES-SHA	Y	Y	Y	Y	Y
P_4	IPsec-3DES-SHA-WEP-128	Y	Y	Y	Y	Y
P_5	8021x-EAP-MD5	Y				
P_6	8021x-EAP-TLS	Y			Y	Y
P_7	8021X-EAP-MD5-WEP-128	Y	Y			
P_8	8021X-EAP-TLS-WEP-128	Y	Y		Y	Y
P_9	8021X-EAP-MD5-WEP-128-IPsec-3DES-MD5	Y	Y	Y	Y	Y
P_{10}	8021X-EAP-TLS-WEP-128- IPsec-3DES-MD5	Y	Y	Y	Y	Y
P_{11}	8021X-EAP-MD5-WEP-128-IPsec-3DES-SHA	Y	Y	Y	Y	Y
P_{12}	8021X-EAP-TLS-WEP-128-IPsec-3DES-SHA	Y	Y	Y	Y	Y

Such policies are required, if visiting users have security support at more than one network layer. Therefore, the network can fulfill the needs of a large number of users. Also, security functionalities required by a network may not be fulfilled by one security protocol, leading to the need for configuration of more than one protocol in the network. Our study incorporates services provided by WEP, IPsec and 802.1x in different ways. Initially we focus on the combination of IPsec and WEP. We first analyze the overhead associated with IPsec (3DES, and SHA) and WEP (128 bits). Then we perform experiments with 802.1x and WEP to capture combined effects of all security features at MAC layer and transport layer. Finally, we unite different services of 802.1x, WEP and IPsec together for analysis. P_4, P_7, P_8, P_9, P_{10}, P_{11} and P_{12} are hybrid policies configured in our testbed. Integration of different protocols helps us answer a vital question, i.e., whether it is beneficial to combine security features at different network layers at the cost of adding extra overhead. A subset of security policies and associated features are shown in Table 1.

2.4 Data acquisition

For each security policy, measurements are collected in two phases. In the *First phase*, we concentrate on taking data that is related to initial negotiations, which take place during the handshake stage of any protocol. We use *Ethereal* network packet analyzer to capture the packets exchanged in handshake phase. Using time-stamp option provided in every packet, we record the time difference between the first and last packet of the negotiation phase. Since in our analysis, we interpret initial negotiation phase as the authentication phase, data obtained in this manner is used to compute and compare authentication time for different security policies. The *Second phase* in our study includes generating different traffic streams in the network between two participating nodes. We use *ttcp* and *Iperf* traffic generators, because they can generate TCP and UDP traffic. Moreover, these utilities provide different types of statistics such as end-to-end delay, throughput, packet loss, and so on. Also, we can verify whether measurements provided by one tool are in consistent with experimental data provided by other tools.

Throughout all experiments, the transmission rate for each wireless card has been set to 11 Mbps. In addition, TCP and UDP streams consist of 16 MB data, as differences in results were not visible for smaller data. Moreover, we repeat experiments more than 15 times to obtain accurate results. The average values of these results are further used in our analysis and comparison.

3 Quality of protection (QoP) model

One of the most challenging issues in the performance evaluation of security policies is to define metrics to indicate their benefits. In general, the benefits of a feature in a security policy depend upon which cryptographic algorithm has been used to implement the feature. For instance, the confidentiality feature can be implemented using 3DES or WEP-128. However, as 3DES uses longer key of 192 bits as compared to 128-bit key used in WEP-128, it is assumed to deliver better confidentiality than WEP-128. Though it may be concluded qualitatively that 3DES offers better confidentiality than WEP-128, it is very difficult to quantify the absolute difference. In existing studies, this qualitative nature of various security features based on their implementation choices has been discussed as

quality of protection (QoP) of a particular feature in a security policy [24]. Research is going on to shift from qualitative paradigm to quantitative paradigm so that QoP for security policies can have same notion as QoS in Networking.

In this work, we look at the problem of quantification from different perspectives of protecting communications, and aim to quantify in the sense of *relative benefits* of various security features based on their implementation. In other words, we examine the benefits offered by an implementation of a security feature to application users. For example, we argue that if we assign a higher number to 3DES, say 2, and a lower number to WEP-128, say 1, it is easy to tell by looking at numbers that 3DES seems to offer better benefits that WEP-128. We extend this notion to quantify the cumulative benefits offered by various features in security protocols. Therefore, we discuss quality of protection (QoP) associated with various features of security protocols based on their implementations. Then, we define a simple *utility function* to quantify the benefits offered by various features of security protocols to application users. The utility function maps QoP of a feature of a security policy to a numeric value for quantification. The utility function provides a *micro view* of the benefits offered by the features of security policies. In addition, we also define a simple *additive reward model* to quantify the cumulative benefits offered by security policies. The additive reward model offers a *macro view* of the benefits provided by a security policies.

The main advantage of our model is that application users or system designers can assess security policies by looking at their micro view and macro view, and then decide whether a particular policy is suitable for their needs or not. Besides, our model provides fine granularity so that it becomes easier to categorize various security policies into a broader range.

3.1 Related work

There has been extensive research in the past to define QoP of a system. Various security models such as The Orange Book [25] have been proposed to assess the QoP of a system. As there are only four levels involved for defining QoP of a system as discussed in The Orange book, it may be difficult to distinguish the advantages offered by the security policies belonging to a same level [25]. In [26], authors use matrix approach to quantify and define security levels of a Voice over IP infrastructure during its design phase. Weights to individual features in a policy are assigned and specified in matrix form. By computing Euclidean distance of two matrices, comparison against a reference policy and a system policy is made to quantify QoP of a system. Although, this approach is simple and effective, matrix computation may be costly in terms of processing time and battery power. More importantly, it is almost impossible to find a one-to-one correspondence between a security level and a practical security policy in real systems.

Besides, authors in [27] proposes a QoP framework to provide different security levels for mobile multimedia applications. Their framework defines QoP meta data which includes security features and QoP parameters. Authors suggest that key lengths of cryptographic algorithms or time interval during which the information is valid, can be used as QoP parameters. Besides, authors define QoP reward profile as the security benefit an application user receives from the QoP parameters specified. We adopt the same framework as discussed in [27] to define our utility function. In addition, we extend their framework by defining an additive reward model to quantify the cumulative benefits offered by a security policy to application users.

3.2 Utility function

The motivation for utility function in our work is taken from the multidimensional reward function discussed in [27], while utility function has been used in many previous work to quantify the benefits of networking performance. In this work, we use *utility function* to indicate the strength of protection as a result of implementing security features. In general, the time period to compromise a particular implementation of a security feature depends upon the length of cryptographic key used in it. For instance, assume that the similar computing power is used to break 3DES and WEP-128. Then, it is obvious that it will take longer to compromise 3DES with 192-bit key than WEP-128 with 128-bit key. However, whether a security feature can be compromised, depends not only on the key lengths, but also on how secure or insecure the method of message exchange related to that security feature is. For example, authentication with 802.1x-EAP-MD5 uses 128-bit key to hash passwords. However passwords are sent in clear text. Therefore, any malicious user can sniff the password and replay that. So, we consider that time to compromise a particular implementation depends upon many factors such as key length, the method of message exchange and so on. Therefore, our utility function also becomes directly proportional to such factors, which in turn, imply the QoP associated with a particular security feature qualitatively. Our utility function maps the QoP of a security feature into a numeric weight. A summary of weights assigned to

Table 2 Weights assigned to various implementations of security features

Security feature	Authentication (w_A)	Mutual authentication (w_M)	Confidentiality (w_C)	Data integrity (w_T)	Non-repudiation (w_R)
WEP-128 (shared)	1	–	1	–	–
802.1x-EAP-MD5	2	–	–	–	–
IPsec	3	1	2 (3DES)	2 (SHA)	1 (ESP)
IPsec/802.1x-EAP	–	–	–	1 (MD5)	–
802.1x-EAP-TLS	4	2	–	–	2

each feature in different protocols is shown in Table 2 ("–" means not specified). A detailed explanation how these weights are assigned is as follows.

Since WEP-128, 802.1x-EAP-MD5, IPsec and 802.1x-EAP-TLS provide authentication feature in our testbed, four different weights are assigned to each of them. WEP-128 is assigned the lowest weight of 1 due to its weak cryptographic algorithm [4]. 802.1x-EAP-TLS is assigned the highest weight of 4 as it uses digital certificate for signing private keys. IPsec has weight of 3 which is lower than the weight assigned to 802.1x-EAP-TLS, because IPsec uses public key cryptography without certificates unlike 802.1x-EAP-TLS. Although digital certificate can be used with IPsec as well, but we use IPsec without certificate due to some practical problems in configuring them together in the testbed. On the other hand, 802.1x-EAP-MD5 is assigned weight of 2 which is lower than those of IPsec and 802.1x-EAP-TLS, because it uses weak plain text user-password [28].

In case of mutual authentication, IPsec and 802.1x-EAP-TLS protocols are considered, and are assigned weights of 1 and 2, respectively. The reason for assigning a higher weight to 802.1x-EAP-TLS than IPsec is the same as we described for the authentication feature.

In addition, WEP-128 and 3DES offer confidentiality feature for various security policies in the testbed. 3DES encryption algorithm is allocated higher weight of 2 than the weight of 1 assigned to WEP-128, because 3DES provides complex and more secure cryptographic algorithm than WEP-128. IPsec with SHA/MD5 and 802.1x-EAP-MD5 provide the data integrity security feature. Since SHA uses longer keys than MD5 [29], IPsec with SHA is assigned a higher weight of 2 than those of IPsec with MD5 and 802.1x-EAP-MD5. IPsec with MD5 and 802.1x-EAP-MD5 are assigned the same weight of 1, because both of them use MD5 algorithm.

In general, we notice that TLS has been assigned a higher weight than MD5, because it makes use of digital certificates which provide stronger authentication feature than MD5 [28]. Note that the weight assigned to an implementation of a security feature signifies only its relative benefits corresponding to other implementations. These weights do not imply absolute quantification of security benefits associated with a security feature. For instance, if two implementations providing authentication feature are assigned weights of 4 and 1, respectively, the one with weight 4 is not necessarily four times stronger than the one with weight 1 with respect to authentication feature. The weight assignment only signifies that the implementation with weight 4 is stronger than the implementation with weight 1 with respect to authentication feature, whereas there are other implementations providing authentication features with benefits in between.

3.3 Additive reward model

Authors in [27] have discussed a reward model to quantify benefits of security features. However, a security policy includes more than one security feature such as authentication, mutual authentication, confidentiality, data integrity and non-repudiations. The benefits provided by a security policy can not be determined by considering just one feature. In addition, security features in a wireless LAN may not be provided by an individual protocol, instead, cross-layer security protocols may be used. For instance, both IPsec and 802.1x can be configured in a laptop computer. Therefore, we extend the reward model in [27], and defines an additive reward model which quantifies cumulative benefits of an individual or hybrid security policy by considering all its security features.

Before defining the additive reward model, we discuss the goals to be achieved by it: (1) Different security features provided in each policy should be considered. These features include authentication, mutual authentication, confidentiality, data integrity and non-repudiations; (2) One security policy may include *multiple* security implementation, such as EAP-MD5, EAP-TLS, IPsec, and WEP, which consist of different algorithms. Furthermore, implementation may provide *multiple* security features. For instance, EAP-MD5 can

Table 3 Additive rewards (σ)

Security protocol	P_1	P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
σ	0	2	3	5	8	9	10	11	12	13	18	19
σ (Normalized)	0	10.5	15.8	26.3	42.1	47.4	52.6	57.9	63.2	68.4	94.7	100

provide authentication and data integrity security features. Therefore, security features and their implementations must be considered in a quantitative metric; (3) The approach to quantifying benefits of a security policy should be simple and practically feasible with regards to processing time and implementation, so that it can be implemented even in resource constrained environments such as wireless networks; (4) A quantitative metric should have sufficient fine granularity so that it can be used to identify clear distinction among different security policies.

Therefore, we define an *additive reward model* to quantify the benefits achieved by different security policies. The additive reward model is based on a linear sum of weights assigned to various features in a policy. The weights of security features are obtained by using the utility function discussed above. We define *additive reward model* by considering five security features, *authentication*, *mutual authentication*, *confidentiality*, *data integrity* and *non repudiation*. Let

- U_A^i be the weight associated to authentication provided by an implementation i.
- U_C^i be the weight associated to confidentiality provided by an implementation i.
- U_T^i be the weight associated to data integrity provided by an implementation i.
- U_R^i be the weight associated to non-repudiation provided by an implementation i.
- U_M^i be the weight to mutual authentication provided by an implementation i.

Assume a security policy P_α consists of n security implementations. Then, *additive reward* of security protocol P_α is a metric which is defined as

$$\sigma(P_\alpha) = \sum_{i=1}^{n} U_A^i \mathcal{I}_\mathrm{A} + U_C^i \mathcal{I}_\mathrm{C} + U_T^i \mathcal{I}_\mathrm{T} + U_R^i \mathcal{I}_\mathrm{R} + U_M^i \mathcal{I}_\mathrm{M}. \quad (1)$$

In the above expression, $\mathcal{I}_{(\cdot)}$ is an indicator function, which equals to 1 if that particular security feature is provided by the implementation i, otherwise zero. Let us walk through an example of how σ is obtained. We notice from Tables 1 and 2 that P_{12} (802.1x-EAP-TLS-WEP-128-IPsec-3DES-SHA) consists of three implementations: IPsec-3DES-SHA, WEP-128 and 802.1x-EAP-TLS. These three implementations consist of ten features: five by IPsec-3DES-SHA, two by WEP-128, and three by 802.1x-EAP-TLS. Let i, j and k represent IPsec-3DES-SHA, WEP-128 and 802.1x-EAP-TLS, respectively. By using Table 2, weights of features provided by IPsec-3DES-SHA are $U_A^i = 3$, $U_M^i = 1$, $U_C^i = 2$, $U_T^i = 2$, and $U_R^i = 1$. The corresponding weights of features in WEP-128 are $U_A^j = 1$, $U_M^j = 0$, $U_C^j = 1$, $U_T^j = 0$, and $U_R^j = 0$. With 802.1x-EAP-TLS, we obtain the weight as $U_A^k = 4$, $U_M^k = 2$, $U_C^k = 0$, $U_T^k = 0$, and $U_R^k = 2$. Although 802.1x-EAP-TLS can provide confidentiality and data integrity, but in our testbed, it is used for authentication in network without its confidentiality and data integrity features. Therefore, we do not take into these features account in 802.1x-EAP-TLS. By substituting the weights of various features, the value of σ for protocol P_{12} is $3+1+2+2+1+4+2+2+1+1=19$. For comparative study, we normalize σ values of other protocol based on the highest value of 19 of P_{12}. Table 3 lists actual and normalized σ values of different protocols in the increasing order.

It can be easily observed that weights assignments in our work is such that a security policy with stronger security features obtains higher value of σ. Although, the weight assignment by using utility function to various security features is unique to this study, our definition of σ can be extended in other scenarios not explored here. If security policies with other security implementations are considered, then weights assignments may require the modifications to various weights to accommodate new security implementations.

3.4 Discussions

Here we discuss the drawback associated with additive reward model and remedies to correct them. Assume that a policy P_α has one implementation of authentication which is very strong, and another policy P_β has four implementations of authentication which are relatively weak. In addition, assume that the implementation of authentication feature in policy P_α is assigned a weight of 3 since it is considered stronger, and four weak implementations of authentication are assigned weight of 1 each. If we compute the values of σ for P_α and P_β, we obtain 3 and 4 for P_α and P_β, respectively. It implies that P_β seems stronger than P_α, however in

reality P_α is stronger that P_β. One solution to this inconsistency is to assign weight of 5 to the authentication feature in P_α. In this way, we can conclude correctly by using additive reward model that P_α is stronger than P_β.

Therefore, we argue that it is not a problem associated with additive reward model. However, it is the problem how the utility function assigns weights to different security features. As we said above, we reemphasize that weight assignment should be done based on what are the security policies available in a network, and which way those protocol will be configured in the network. A good utility function is the one which assigns weights to different security features based on their implementations so that there is no inconsistency.

4 Performance metrics

Now we describe metrics associated with policies in terms of authentication time, cryptographic overhead and throughput. Authentication time and cryptographic cost are dependent on control signaling in an authentication phase and encryption/decryption process of a policy, respectively. In addition, throughput helps us in quantifying QoS degradation for a particular policy.

Authentication Time We consider authentication time, represented as T_A, as the time associated with authentication phase of a policy. It is due to the fact that time involved in an authentication phase is one of the important factors contributing towards performance impact in a network. Here, we describe a simple method to obtain the authentication time T_A based on real-time measurements. Let the total time involved in transmitting, receiving and processing the kth packet by P_α during its authentication phase be denoted as $t_A(k, P_\alpha)$. By exchanging n packets during authentication phase, the total authentication time can then be obtained by $T_A(P_\alpha) = \sum_{k=1}^{n} t_A(k, P_\alpha)$.

Cryptographic Overhead It represents the performance cost associated with a policy. Since we consider the time involved in authentication phase as *authentication time* of a policy, we now define cryptographic overhead as the bit rate that involves overhead due to other features such as encryption/ decryption, data integrity and non-repudiation. Below we describe the procedure for calculating this cost of a policy.

Let P_1 denote the case without policy configured in the network and $P_\alpha(\alpha \neq 1)$ denote a policy with certain security features. Let $t_C^s(k, P_\alpha)$ denote the time required to process the kth packet by a sender S during the configuration of policy P_α in our testbed. The time duration, $t_C^s(k, P_\alpha)$, involves adding extra header by a policy and encrypting a packet. Let $t_C^r(k, P_\alpha)$ denote the time required to process the kth packet by a receiver R during the configuration of policy P_α. The time duration, $t_C^r(k, P_\alpha)$, involves removing extra header of policy and decryption of a packet.

Further we denote $t_C^{sr}(k, P_\alpha)$ as the time taken by the kth packet in traversing the network between the sender and the receiver during security policy P_α. Therefore, the total time involved in processing the kth packet, denoted by $t_C(k, P_\alpha)$, between the sender and the receiver during policy P_α is the sum of three time periods defined above, and is given by,

$$t_C(k, P_\alpha) = t_C^s(k, P_\alpha) + t_C^r(k, P_\alpha) + t_C^{sr}(k, P_\alpha). \quad (2)$$

Assume that n packets are transmitted between the sender and the receiver, then the total time required for processing n packets during security policy P_α is the sum of time involved in processing all n packets, $T_C(P_\alpha)$, is, $T_C(P_\alpha) = \sum_{k=1}^{n} t_C(k, P_\alpha)$.

Consider that the number of bits in each packets may be different, for example, the size of the kth packet is b_k bits. Then the total number of bits in n packets, denoted as B_n, is, $B_n = \sum_{k=1}^{n} b_k$.

Now we compute bit rate associated with various policies to measure the cryptographic overhead for each policy. Let $R_B(P_\alpha)$ denote the bit rate (bits/sec) that can be experienced during policy P_α and $R_B(P_1)$ denote the bit rate (bits/sec) obtained during policy P_1 (that is, no security), respectively. Assume that $C_C(P_\alpha)$ denotes the cryptographic overhead associated with policy P_α. In this work, we evaluate the cryptographic overhead as the difference between the bit rates for security policies P_α and P_1. Then by using $T_C(P_\alpha)$ and B_n, the cryptographic overhead for policy P_α is determined by

$$C_C(P_\alpha) = R_B(P_1) - R_B(P_\alpha) = \frac{B_n}{T_C(P_1)} - \frac{B_n}{T_C(P_\alpha)}, \quad (3)$$

Note that one of the distinguished properties of the definition here is that it is measurable rather than a denotation for analysis.

Throughput (bits/second) In this work, we are interested in observing the effective data rate, which is called *throughput* between participating nodes with the configuration of a security policy in the network. Considering that in an experimental study, transmitted data is represented by measured in bits, the throughput associated with a security policy is the same as the bit rate associated with a security policy which we computed

previously during the calculation of cryptographic overhead. Therefore, throughput (η) during security policy P_α can be obtained by

$$\eta(P_\alpha) = \frac{B_n}{\sum_{k=1}^{n}\{t_C^s(k, P_\alpha) + t_C^r(k, P_\alpha) + t_C^{sr}(k, P_\alpha)\}}. \quad (4)$$

Therefore, these performance metrics are not only consistent with performance evaluation of quality of service, in particular, they are useful in experimental studies too because the parameters used in these definitions are measurable in real systems.

5 Experimental results and analysis

We classify the results based on policies into three groups as *low*, *middle* and *high* to evaluate the performance impact of afore-mentioned policies in various mobility scenarios. For *low security group* (LSG), we consider policies that have normalized σ values below 45%, including policies P_2, P_5, P_6 and P_7. This group of policies provide basic security protection such as authentication and one or more other protection. The *middle security group* (MSG) includes policies that have normalized σ values between 45 and 70%, such as P_3, P_4, P_8, P_9 and P_{11}. In this group, security is provided by either IPsec or 802.11x with security protection by one or two protocols. Policies that have normalized σ values approaching to 100% belong to the *high security group* (HSG) such as P_{10} and P_{12}. These two policies are provided by the strongest implementations in security protocols.

5.1 Authentication time

Authentication time is associated with the initial phase of a policy as defined in Section 4. During this period, a mobile node provides its credentials to the authentication server, such as home agent or foreign agent in the testbed, to access a network. Messages exchanged during the initial phase of a policy vary with the security protocols involved in the policy. Moreover, authentication time for various policies is obtained for non-roaming and roaming scenarios, respectively. Table 4 demonstrates the components of authentication time associated with each security policy in various scenarios. Since WEP does not involve exchange of control messages, there is no authentication time involved with it. Since Mobile IP is used for enabling mobility in the testbed, authentication time (T_A) for IPsec and 802.1x includes Mobile IP registration time as well. Fig. 2 demonstrates the authentication time versus normalized σ in an increasing order and corresponding security policies.

We observe from Fig. 2a that 802.1x-EAP-TLS policies produces the longest authentication time among all policies. This is due to the fact that 802.1x-EAP-TLS uses digital certificate for mutual authentication, which involves exchanging several control packets. We find that a total of 17 control packets are exchanged during the initial phase of 802.1x-EAP-TLS, which is much more than eight and nine control packets exchanged in 802.1x-EAP-MD5 and IPsec authentication phases, respectively. Moreover, IPsec policies generate longer authentication time than 802.1x-EAP-MD5 (without IPsec) policy because of the tunnel establishment in IPsec. In addition, we can see that authentication time in roaming scenarios is much longer than non-roaming scenarios due to the re-authentication in a foreign network for all security policies. Besides these general observations, we notice that authentication time does not increase monotonically with respect to *sigma* values of security policies. For example, policy P_3 (IPsec) induces lower authentication time than P_6 (802.1x-EAP-TLS) in all scenarios although it has higher σ value than P_6. Although P_{10} and P_{12} cause longer authentication time than other policies but these policies consist of highest σ values because security features are enabled in more than one security protocols.

We observe that policy P_{12} (with the highest σ value) yields the longest authentication and incurs around 7 and 3 times longer authentication time than P_5 which has the shortest authentication time in non-roaming and roaming scenarios, respectively. This observation suggests that variations in authentication time values are less in roaming scenarios than in non-roaming scenarios and that even policies with lower σ values induce higher authentication time in roaming scenarios. The reason for this phenomenon is that registration to a foreign agent takes a very long time (e.g., 1.4 s).

However, we find that although σ value of P_4 is higher than that of P_8, authentication time of P_4 is less than that of P_8 in both scenarios. Therefore, it can be concluded that authentication time for security policies does not increase monotonically with σ.

Further, authentication time of *high security group* is up to two times longer than that of the middle group, e.g., IPsec policies in both roaming and non-roaming scenarios. We discover that security policies, which are in the middle security group do not exhibit much variations in authentication time, and IPsec policies, P_3 and P_4, induce the lowest authentication time (1.4 s in non-roaming and 2.8 s in roaming) among them. Based on these observations, we conclude that policies in the middle security group provide the best trade-off

Table 4 Authentication time computation

Scenario/policy	IPsec	802.1x-EAP(MD5/TLS) without IPsec	802.1x-EAP(MD5/TLS) with IPsec
Non-roaming ($\mathcal{N}$)	IPsec tunnel establishment time in HN + MN registration time to HA	802.1x -EAP(MD5/TLS) authentication time in HN + MN registration time to HA	IPsec tunnel establishment time in HN + 802.1x-EAP(MD5/TLS) authentication time in HN + MN registration time to HA
Roaming ($\mathcal{R}$)	IPsec tunnel establishment time in HN + MN registration time to HA + MN registration time to FA	802.1x-EAP(MD5/TLS) authentication time in HN + MN registration time to HA	IPsec tunnel establishment time in HN + 802.1x-EAP(MD5/TLS) authentication time in HN + MN registration time to HA

between security and performance overhead, and IPsec policies, P_3 and P_4, are the best among them. On the other hand, P_{12} (802.1x-EAP-TLS with IPsec) is best suitable for networks carrying very sensitive data.

5.2 Cryptographic overhead

By analyzing cryptographic overhead, we capture encryption and decryption time associated with security policies during the data transmission. Tables 5 and 6 show cryptographic overheads in non-roaming and roaming scenarios for TCP and UDP traffic streams for policies 2 to 12 because there is no cryptographic overhead for policy 1, respectively. In these tables, values presented in italics represent the recommended security policies in each security *group* for a scenario; values presented in bold face indicate the overall recommended security policy for a particular network scenario. Security policies are arranged in an increasing order of normalized σ as shown in Table 3. For example, in the middle group, normalized σ of policy P_4 is higher that of policy P_8.

We notice from Table 5 that for low security group, P_5 and P_6 exhibit negligible cryptographic overheads, which is due to the fact that these policies do not consist of any confidentiality feature associated with them. Although, theoretically, cryptographic overheads of policies P_5 and P_6 should be zero, but we obtained small values in real-time environments, which may be caused by broadcast messages or polling messages at the MAC layer. Within each group, we recommend security policies by giving normalized σ values higher priority, e.g., we recommend policy P_6 shown in italics for this group even though its cost is a bit higher than policy P_5. In the middle security group, we observe that cryptographic overheads associated with policies P_4, P_9, P_{11}, P_{10} and P_{12} in non-roaming scenarios are very close to each other, showing little variations. This is due to the fact that these policies use the same IPsec and WEP protocols which are the dominating factors contributing towards their cryptographic overheads. Generally, the policies P_4, P_9, P_{11}, P_{10} and P_{12} exhibit 16% higher cryptographic overheads than P_3, and around 3.5 times higher than that of P_2, P_7

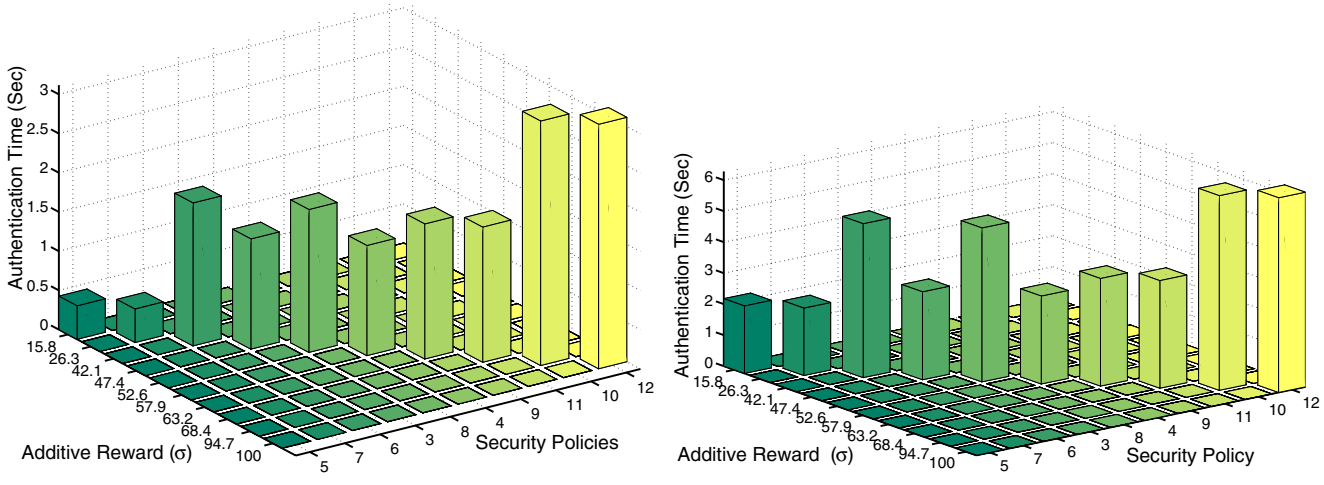

Fig. 2 Authentication time vs. normalized additive reward (σ): **a** non-roaming scenarios, **b** roaming scenarios

Table 5 TCP cryptographic overhead (kbits/sec) under various network scenarios

Network scenarios		Security group										
		Low				Middle					High	
		P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
Non-roaming ($\mathcal{N}$)	N1	71.10	11.88	75.88	*11.47*	264.90	**77.21**	302.80	286.89	313.94	291.48	*301.77*
	N2	70.90	2.09	101.88	*3.92*	273.45	**57.15**	311.70	347.33	298.78	299.25	*296.13*
	N3	108.78	7.29	105.11	*4.33*	304.59	**118.71**	331.68	343.84	378.10	382.87	*343.52*
Roaming ($\mathcal{R}$)	R1	90.43	1.97	104.81	*6.15*	209.54	*92.19*	216.27	246.49	251.56	259.97	**260.81**
	R2	208.04	25.60	232.66	*1.79*	318.32	*230.78*	367.53	393.13	391.12	381.70	**395.29**

and P_8. The reason is that policies P_4, P_9, P_{11}, P_{10} and P_{12} have more than one level of encryption and decryption processing associated with them. A closer look at the table reveals that cryptographic overhead increases corresponding to σ values. However, we see that P_8 is the policy with a higher σ value but with lower cryptographic cost. Specifically, P_8 exhibits almost half of cryptographic overhead of policies P_4, P_9, P_{11}, P_{10} and P_{12}, and almost similar to policies P_2 and P_7. Therefore, we recommend policy P_8 not only for the middle group, but also for TCP streams in non-roaming scenarios and for UDP streams in roaming scenarios, which are shown in bold face. In other words, policy P_8 provides a good balance between security benefits and low cryptographic overhead.

We also notice the similar behavior for UDP in various non-roaming scenarios from Table 6. However, cryptographic costs of policies for UDP are less than that of TCP. It is due to the fact that TCP requires acknowledgment for each packet, leading to the transmission of more number of packets through the networks than UDP. So TCP results in higher encryption and decryption processing overhead, leading to increased cryptographic cost. Therefore, the observations suggest that P_8 (802.1x-EAP-TLS with WEP) provides the best trade-off for both types of traffic streams in non-roaming scenarios. However, we recognize from Table 6 that, in non-roaming scenarios during UDP, difference between cryptographic overheads of policies P_9, P_{10}, P_{11}, P_{12} (with high σ values), and P_8 is relatively less. Therefore, P_{12} is a good choice with little extra overhead in these scenarios due to its very strong security features and it is also an alternative in roaming scenarios.

5.3 Throughput

To understand the impact of policies on the network performance, Tables 7 and 8 present the throughput resulting from policies in different network scenarios for TCP/UDP streams. In the similar way, we use bold face to highlight the recommended policy for each scenario, but not for each group because variations in throughput across security policies are not very significant.

We observe that the highest variations in throughput for various security policies during TCP traffic are between 12–15%; whereas for UDP traffic they are between 6–8% for non-roaming scenarios, respectively. In roaming scenarios, R1 and R2, exhibit variations up to 10–16% for TCP traffic and they are up to 7–12% for UDP traffic, respectively. So the variations are in general around 10% in most of the scenario, which suggests that the effect of security policies over throughput during data transmission is not very sig-

Table 6 UDP cryptographic overhead (kbits/sec) under various network scenarios

Network scenarios		Security group										
		Low				Middle					High	
		P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
Non-roaming ($\mathcal{N}$)	N1	97.68	0.54	111.59	*5.70*	174.50	*95.98*	186.58	163.15	185.81	164.66	**198.75**
	N2	51.77	6.21	42.74	*7.26*	101.20	*55.28*	173.18	177.83	194.41	170.30	**175.01**
	N3	139.14	41.23	162.31	*53.36*	193.05	*168.99*	289.85	284.47	304.68	289.38	**286.03**
Roaming ($\mathcal{R}$)	R1	64.84	20.90	69.59	*5.84*	164.96	**73.87**	227.47	384.51	399.53	354.55	*375.46*
	R2	72.71	3.47	88.14	*10.73*	172.47	**99.27**	184.49	241.26	241.26	241.26	*241.26*

Table 7 TCP throughput (mbits/sec) under various network scenarios

Network scenarios		Security group											
		NS	Low				Middle					High	
		P_1	P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
Non-roaming ($\mathcal{N}$)	N1	2.90	2.83	2.89	2.83	2.89	2.64	2.83	2.60	2.62	2.59	2.61	**2.60**
	N2	5.64	5.51	5.64	5.45	5.64	5.11	5.53	5.04	4.97	5.06	5.06	**5.07**
	N3	2.97	2.86	2.96	2.86	2.97	2.67	2.85	2.64	2.63	2.59	2.59	**2.63**
Roaming ($\mathcal{R}$)	R1	2.83	2.74	2.83	2.73	2.83	2.62	2.74	2.62	2.59	2.58	2.57	**2.57**
	R2	2.86	2.65	2.83	2.62	2.86	2.54	2.63	2.49	2.46	2.47	2.48	**2.46**

nificant. This is based on the fact that we have not taken into account the cost of authentication time for calculating throughput, because throughput for a data stream is calculated by using the total time involved in transmission of the entire data after authentication phase is over. Therefore, variations in throughput values presented in this paper are caused by cryptographic overheads only. Another reason to segregate authentication phase from throughput phase is to measure the authentication time independently, which would be helpful in comparing authentication time and cryptographic overhead because the former is mainly caused by exchanging signaling messages and the latter one is due to computation.

We believe that, in the future, as hardware becomes faster, cryptographic overhead (i.e., time involved in encryption/decryption process) may be reduced further. Moreover, based on our previous observations from Fig. 2, authentication time in roaming scenarios is very high, and it may affect mobile applications significantly as user mobility increases. Compared to the difference in authentication time, we consider that QoS degradation in a network may be more significant due to the authentication time than the cryptographic overhead in the design of security protocols for wireless networks. More discussions will be presented in Section 6.

6 Observations and suggestions

The work described in this paper is our study in building a complete wireless LAN system with IP mobility for evaluating the performance impact of security protocols. This experimental work has been valuable because it has demonstrated that some QoS metrics are significantly affected by security policies. As a result, these well-designed security protocols may not be applicable in real systems. Also, some intuitive assumptions made about the trade-off between QoS and security in much current research do not hold well in practice. The observations we gained from this work will have great impacts on some of the design choices in providing secure, high quality, mobile services in wireless LANs. In this section, we summarize our findings based on previous discussions of experimental results and analysis. In addition, we provide several suggestions regarding to the design of security protocols for wireless networks in the future.

Table 8 UDP throughput (mbits/sec) under various network scenarios

Network scenarios		Security group											
		NS	Low				Middle					High	
		P_1	P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
Non-roaming ($\mathcal{N}$)	N1	3.53	3.43	3.53	3.42	3.52	3.35	3.43	3.34	3.37	3.34	3.36	**3.33**
	N2	6.36	6.27	6.35	6.29	6.35	6.18	6.27	6.05	6.04	6.01	6.06	**6.05**
	N3	3.64	3.50	3.60	3.48	3.58	3.45	3.47	3.35	3.35	3.33	3.35	**3.35**
Roaming ($\mathcal{R}$)	R1	3.59	3.52	3.57	3.52	3.58	3.42	3.52	3.36	3.20	3.19	3.23	**3.21**
	R2	3.54	3.46	3.53	3.45	3.53	3.36	3.44	3.35	3.31	3.45	3.28	**3.34**

1. *Performance Impacts of Security Polices.* We found that the performance impact on wireless LAN can vary significantly with different policies for TCP/UDP traffics. For instance, authentication time varies from 0.11 s for Mobile IP only to 6.28 s when policy 802.1x-EAP(TLS) with IPsec is applied. These variations will cause completely different effect on wireless services. When end-to-end delay is below 0.15 s, there is no audible delay for real-time voice traffic; 0.40 s is the upper limit for communications with QoS [30]; and above 0.60 s, communications are not possible. Therefore, some policies cannot be used in wireless networks when QoS is required. Therefore, we conclude that security protocols have a significant impact on system performance.
2. *Performance Impacts of Mobility Scenarios.* The performance of polices can vary significantly when the same policy is used with different mobility. For example, P_{12} leads to cryptographic overhead of around 301 and 395 Kbps, and TCP throughput achieved for P_1 is 5.6 and 2.8 Mbps in non-roaming and roaming scenarios, respectively. As real-time voice requires 64 Kbps per call, P_{12} results in 8 and 14% reduction in voice calls in non-roaming and roaming scenarios, respectively. It shows that non-roaming scenarios are more favorable to policy P_{12} than roaming scenarios. Therefore, it concludes that mobility scenarios also affect the choice of policies and should be considered during the network design for deployment of security protocols.
3. *Trade-off between Security and Performance.* We observed that in general, performance degrades as security policies provide more benefits, though there are very few exceptions. For instance, P_{10} and P_{12} deliver high throughput (3 Mbps), but with high cryptographic overhead (395 Kbps) and authentication time (6.28 s). Whereas, P_8 in MSG causes small cryptographic overhead (80 Kbps) and authentication time (4.96 s), but offers high throughput (3 Mbps). Since, real-time services, such as voice over IP (VoIP) and video conferencing, require low packet loss (1–3%) and transmission delay (0.15–0.4 s) [30], P_8 can be better suitable to them than P_{10} and P_{12}. Whereas, for service, such as email, file-transfer, web-browsing with no constraints on QoS, P_{10} and P_{12} can deliver better performance to them. Therefore, our results can enable network designers to find policies providing trade-offs suitable to their networks with regard to real time services.
4. *Integration of Cross-Layer Protocols.* We observed that integration of protocols at different layers is able to counter more attacks than individual protocols and offers performance comparable to individual protocols. In LSG, P_2 is prone to *Authentication Forging or Message Fabrication* (AF/MF) due to its small key space and the reuse of the same initialization vector [4]. P_5 and P_7 are prone to AF due to their MD5 algorithm which user clear-text passwords [23]. Only P_6 in LSG is immune to AF due to digital certificate used in its authentication process. However, cross-layer policies in MSG and HSG either include IPsec or EAP-TLS using public key cryptography or digital certificate, and therefore, are not prone to AF. Similarly, though policies in LSG are susceptible to message fabrication, but policies in MSG and HSG are not due to their IPsec or TLS protocols. Also, P_5 and P_7 in LSG transmit *response* to challenge in plain-text form, and therefore, are prone to *Man in the Middle Attack* (MITM) [23]. However, policies with IPsec and TLS protocols are not vulnerable to MITM. Although IPsec employs public key cryptography prone to MITM attack, ISAKEMP key agreement protocol used in it can overcome MITM.

   In addition, policies in LSG and some (e.g.,P_3 and P_4) in MSG are not strong in authentication at the user level because IPsec works at layer 3, which provides system authentication but not user authentication [18]. Therefore, for user authentication, P_3 and P_4 must be employed with higher layer security protocols. For instance, P_{10} and P_{12} provide strong security at layer 3 due to IPsec, and enable user authentication by using 802.1x-EAP-TLS. P_{10} and P_{12} in HSG can overcome most of the malicious attacks discussed above, while providing strong access control. In addition, P_{10} and P_{12} can offer throughput (3 Mbps) similar to policies in LSG and MSG.

   In general, we see that hybrid policies are able to overcome the attacks associated with individual policies, therefore, cross-layer integration of protocols is an advantageous choice in providing better security solutions for many wireless applications. Therefore, we conclude that integration of security protocols can be realized in wireless networks to offer stronger security.
5. *Robust Security Policies.* When we evaluate performance of a network, it is very important to understand the stability of QoS. In addition, it may not always be possible to know the roaming profile of a user in advance. As networks may have been configured with different policies not in accordance with the required QoS by a user, the user may experience large variations in QoS. In order to

provide a steady service, we need to find policies that provide the trade-off between security and low variations in performance impact when user mobility is unknown. Thus, we examine statistical variations in measurements exhibited by policies in different network scenarios. We characterize these statistical variations of each policy in different network scenario as *robustness* against mobility of that particular policy. Specifically, we perform statistical analysis using *mean* and *variance* of measurements. As an example, statistical variations of cryptographic overhead (C_C) and throughput (η) for TCP are in Table 9.

From the table, we find P_6 (without confidentiality) with very little variations and conclude that P_6 is robust to user mobility; whereas P_2 is an alternative when cryptographic feature is desired. Meanwhile, P_3 yields the smallest variance than other policies in MSG and the variance shows an increasing trend with respect to σ except policy P_8 which demonstrates unusually high variance of 4.68. The reason is that the encryption/decryption processing times by TLS used in P_8 demonstrate large differences for the entire data stream in different scenarios, which leads to high variance for P_8. Thus, we notice that P_3 provides the best trade-off but P_8 is a good alternative too. Whereas, P_{10} and P_{12} in HSG exhibit almost similar variations with respect to each other. So P_{12} can be regarded as the best policy with high robustness and strong security. Moreover, we observe that policies with lower σ values have smaller variations than that of with higher σ values. In case of throughput, we observe that variance associated with policies are very close to each other. Therefore, it can be suggested that P_{12} provides the best trade-off between robustness and security benefits, because of its highest σ value along with robustness comparable with other policies.

In reality, mobility scenarios are usually unknown, then network administrators should choose security policy to use in advance. However, if the selection of policy is random or based on impractical assumptions, it can degrade performance of real-time services in different mobility scenarios. For example, streaming audio and video can not tolerate packet loss of higher than 1% [30]. If a policy causes large performance variations in different scenario, high packet losses can occur during handoffs. We find that IPsec policy (P_3) (with encryption feature) causes the lowest variation (1.78), whereas P_{11} leads to largest variation (3.35) in different scenarios. Therefore, choosing P_{12} rather than P_3 for a user, with unknown mobility pattern and using streaming audio and video, can lead to poor user perceptive QoS. Therefore, it is concluded that robustness against mobility varies for different security policy and applications.

6. *Application-Oriented Security.* The selection of security policy may require an application pattern which significantly influences protocol security. For applications that require low end-to-end delay and low security, such as real-time video and audio services [30], IPsec policies (P_3, P_4) and 802.1x-EAP(TLS)-WEP policy (P_8) with low cryptographic overhead (from 57 to 300 Kbps) and high throughput (from 2.6 to 6.2 Mbps) are the most suitable. Whereas, integrated policy P_{12} (802.1x-EAP(TLS) with IPsec and WEP) is especially suitable for nomadic roaming wireless applications, such as email, file-transfer and web-browsing.
7. *Further Research on Security Protocols.* By our experiments, we find that P_{12} (802.1x-EAP(TLS) with IPsec and WEP) can offer the strongest security regarding protection against various attacks, such as authentication-forging, MITM and message forging. In addition, P_{12} provides strong user access control. However, it is also noticed that policy P_{12}

Table 9 Robustness analysis for TCP (mbits/sec)

Performance metrics	Statistical Analysis	Security group										
		Low				Middle					High	
		P_2	P_5	P_7	P_6	P_3	P_8	P_4	P_9	P_{11}	P_{10}	P_{12}
C_C	Mean	109.85	9.77	124.07	5.53	274.16	115.21	306.00	323.54	326.70	323.05	319.50
	Var	3.26	0.10	3.83	**0.01**	**1.78**	4.68	3.13	3.28	3.35	3.14	**2.66**
η	Mean	3.32	3.43	3.30	3.44	3.12	3.32	3.08	3.05	3.06	3.06	3.07
	Var	1.50	1.53	1.45	1.52	1.25	1.54	1.21	1.15	1.26	1.25	**1.26**

yields the largest authentication delay (6.28 s) and cryptographic overhead (395 Kbps), which is not a good option for applications with low packet loss (1–3%) and transmission delay (0.15–0.4 s), such as voice over IP (VoIP) and video conferencing [30]. As policy P_{12} is sensitive to mobile scenarios, it is not a good option for users with unknown mobility patterns due to high authentication time. Therefore, reducing authentication delay is one of the most challenging issues in the design of wireless security protocols.

7 Conclusions

The performance impact of security protocols in wireless LANs with IP mobility has been rarely studied due to the lack of quantitative study, though there are considerable efforts on improving security services. To the best of our knowledge, this is the first experimental study which considers IP mobility and cross-layer integration of security protocols in a real-time testbed. In this paper, we presented a comprehensive study on the impact of security policies with mobility support in wireless LANs. We concluded that the integration of security protocols is practically feasible in real systems with regard to system performance, and provides stronger security than individual protocols. Also, we introduced a QoP model to quantify the benefits of security policies and demonstrate the relationship between QoS and QoP. We found that authentication is the most critical factor contributing towards the performance degradation in a network involving users with high mobility and real-time applications. We also observed that security policies exhibit small differences in throughput as compared to other metrics, which taught us that even the strongest policy can be realized in wireless networks with minimal throughput degradation. More importantly, we observed that the selection of a most suitable policy for a network is dependent upon various factors, such as mobility scenarios, applications, and QoS requirements.

References

1. Gast M (2002) 802.11 network deployment, 802.11 wireless networks: the definitive guide. O'Reilly, Sebastopol, CA, April
2. Hannikainen M, Damalainen TD, Niemi M, Saarinen J (2002) Trends in personal wireless data communications. Comput Commun 25(1):84–99
3. Karygiannis T, Owens L (2002) Wireless network security 802.11, bluetooth and handheld devices, National Institute of Technology, Special Publication, pp 800–848, November
4. Borisov N, Goldberg I, Wagner D (2001) Intercepting mobile communications: the insecurity of 802.11. In: Proc of the ACM MobiCom'01, ACM, New York, pp 180–189, July
5. IEEE Std 802.1x-2001x: Port-based network access control, http://www.ieee802.org/1/pages/802.1x.html, June 2001
6. IEEE 802 Standards, http://standards.ieee.org/getieee802
7. Qu W, Srinivas S (2002) IPsec-based secure wireless virtual private networks. In: Proc of the IEEE MILCOM'02, vol 2, IEEE, Anaheim, CA, pp 1107–1112, October
8. Liu W, Lou W, Fang Y (2005) An efficient quality of service routing algorithm for delay-sensitive aplications. Comput Networks 47:87–104, January
9. Godber A, Dasgupta P (2002) Secure wireless gateway. In: Proc of the ACM WiSe'02, ACM, New York, pp 41–46, September
10. Baghaei N, Hunt R (2004) Security performance of loaded IEEE 802.11b wireless networks. Elsevier Comput Commun 27:1746–1756, November
11. Faria DB, Cheriton DR (2002) DoS and authentication in wireless public access networks. In: Proceedings of the ACM workshop on wireless security (WiSe'02), ACM, New York, pp 47–56, September
12. Arbaugh WA, Shankar N, Wang J, Zhang K (2002) Your 802.11 wireless network has no clothes. IEEE Wirel Commun Mag 9:44–51, December
13. Li M, Zhu H, Sathyamurthy S, Chlamtac I, Prabhakaran B. (2004) End-to-end framework for QoS guarantee in heterogeneous wired-cum-wireless networks. In: Proc of the first international conference on quality of service in heterogeneous wired/wireless networks, IEEE Computer Society, Washington, DC, pp 140–147, October
14. Perkins CE (1998) Mobile networking through mobile IP. IEEE Internet Computing 2:58–69, January-February
15. Ma W, Fang Y (2004) Dynamic hierarchical mobility management strategy for mobile IP networks. IEEE J Sel Areas Commun (Special Issue on All-IP Wireless Networks) 22:664–676, May
16. Ma W, Fang Y (2003) Improved distributed regional location management scheme for mobile IP. In: Proceedings of the IEEE international symposium on personal, indoor and mobile radio communications (PIMRC'2003), vol 3. IEEE, Anaheim, CA, pp 2505–2509, September
17. Perkins C (1996) IP mobility support, http://www.ietf.org/rfc/rfc2002.txt, October
18. IPSEC, http://www.freeswan.org.
19. 802.1x Supplicant, http://www.open1x.org.
20. RADIUS, http://www.freeradius.org.
21. OpenSSL, http://www.openssl.org.
22. Mobile IPv4, http://dynamics.sourceforge.net.
23. Kim I-G, Choi J-Y (2004) Formal verification of PAP and EAP-MD5 protocols in wireless networks: FDR model checking. In: Proc of AINA, vol 2. IEEE Computer Society, Washington, DC, pp 264–269, March
24. Karjoth G (2003) Access control with IBM tivoli access manager. ACM Trans Inf Syst Secur (TISSEC) 6:232–257 (May)
25. DoD Trusted Computer System Evaluation Criteria, http://csrc.ncsl.nist.gov/secpubs/rainbow/std001.txt, December 1985
26. Casola V, Rak M, Mazzeo A, Mazzoccca N (2005) Security design and evaluation in a VoIP secure infrastracture: a policy based approach. In: Proceedings of ITCC'05, vol 1. IEEE Computer Society, Washington, DC, pp 727–732, April
27. Ong CS, Nahrstedt K, Yuan W (2003) Quality of protection for mobile multimedia applications. In: Proceedings of the international conference on multimedia and expo (ICME) '03, vol 2, pp 137–40. Baltimore, Maryland, 6–9 July 2003

28. Aboba B, Simon D (1999) PPP EAP TLS athentication protocol. RFC 2716, RFC Editor, US, October
29. Satoh A, Inoue T (2005) ASIC-hardware-focused comparison for hash functions MD5, RIPEMD-160, and SHS. In: Proceedings of the international conference on information technology: coding and computing (ITCC'05), vol 1. IEEE Computer Society, Washington, DC, pp 532–537, April
30. Zhai H, Chen X, Fang Y (2005) How well can the IEEE 802.11 wireless LAN support quality of service? IEEE Trans Wirel Commun 4:3084–3094, November

Avesh Kumar Agarwal received the B.E. degree from the Motilal Nehru Regional Engineering College (Now MNNIT), India in 2000. He is currently working towards the Ph.D. degree in computer science at the North Carolina State University, Raleigh, NC, USA. His current research interests are in the area of simulations and real-time performance evaluation of security mechanisms and routing protocols in wireless LANs and wireless Ad-Hoc networks.

Wenye Wang (M'98/ACM'99) received the B.S. and M.S. degrees from Beijing University of Posts and Telecommunications, Beijing, China, in 1986 and 1991, respectively. She also received the M.S.E.E. and Ph.D. degree from Georgia Institute of Technology, Atlanta, Georgia in 1999 and 2002, respectively. She is now an Assistant Professor with the Department of Electrical and Computer Engineering, North Carolina State University. Her research interests are in mobile and secure computing, quality-of-service (QoS) sensitive networking protocols in single- and multi-hop networks. She has served on program committees for IEEE INFOCOM, ICC, ICCCN in 2004. Dr. Wang is a recipient of NSF CAREER Award in 2006. She has been a member of the Association for Computing Machinery since 2002.