Evaluating the Impact on Data Reception and Energy Consumption of Mobile Devices using IPSec to securely access WiFi Networks

Fernando da Costa Junior, Luciano Gaspary, Jorge Barbosa, Gerson Cavalheiro, Luciano Pfitscher[1], and Jose Dirceu G. Ramos[2]

[1] Universidade do Vale do Rio dos Sinos
Programa Interdisciplinar de Pós-Graduação em Computação Aplicada
Av. Unisinos, 950 – 93.022-000 – São Leopoldo, Brazil
fcaprio@exatas.unisinos.br
[2] Hewlett Packard do Brasil

Abstract. Despite offering the possibility to develop and distribute a new set of applications to its users, the widespread and unrestricted use of mobile computing depends on the provisioning of a secure network environment. Regarding the communication established from mobile devices such as PDAs (Personal Digital Assistants), one of the most currently used standards is the IEEE 802.11b, which presents known security flaws. To overcome them, some alternative setups are commonly deployed, based on link, network, transport or application-layer. In this paper we evaluate the impact on data reception rate and energy consumption of IPSec-based PDAs access to 802.11b (WiFi) wireless LANs. As a result of this work we identify the overhead imposed by the security mechanisms and the capacity of the device to run CPU and network-intensive applications.

1 Introduction

[3]The miniaturization of electronic components and the growing offer of wireless communication technologies have stimulated the development of small and high capacity computational devices, which enable the concrete implementation of the *mobile computing* concept. In a mobile context it is common to have portable devices such as PDAs interconnected to the wired network infrastructure through wireless links. The easiness of connection and physical mobility of these devices leads to the possibility of providing the users of this technology with a new set of applications (e.g. location-aware and video on demand). However, in order to execute these applications in a production environment some security issues need to be addressed.

One of the most currently used standards to allow network connectivity from mobile devices is the IEEE 802.11b, which has several security flaws [1]. In order to overcome them, some alternative security setups, ranging from link to

[3] This work was partially developed in collaboration with HP Brazil R&D.

application layer, have been widely deployed in production environments (e.g. IPSec and SSL). These additional components are essential to enable the secure communication of millions of devices using IEEE 802.11b that have already been sold, and cannot be replaced by other equipment without extra investments.

Regardless of the security mechanism used, it leads to an overhead in terms of both the data sent/reception rates achieved by the mobile device and its energy consumption. Identifying this overhead and determining which applications can be executed by mobile devices such as PDAs (keeping their autonomy) is valuable, because one can adjust security mechanisms to achieve the best tradeoff between security and consumption.

In this paper we evaluate the impact on data reception rate and energy consumption of IPSec-based PDAs access to 802.11b wireless LANs. We have chosen IPSec because it is the most current, widely adopted setup. Furthermore, since it is a network-layer technology, all applications can take advantage of the security mechanisms that it provides: authentication, privacy, and integrity.

The paper is organized as follows: section 2 describes some related work. In section 3 we revisit some vulnerabilities of the IEEE 802.11b standard. Section 4 presents the setup configured to achieve a secure wireless network environment. In sections 5 and 6 we detail the experiments carried out. Section 7 presents some final considerations.

2 Related Work

Measuring and characterizing the current limits of portable devices in terms of both communication capabilities and energy consumption, to mention just a few aspects, are issues that have been gaining attention recently. This topic grows in importance when secure wireless communications are demanded. Since a lot of extra computation is required to guarantee properties such as authentication, privacy, and integrity, the feasibility to run a variety of applications is directly affected.

Potlapally, Ravi, Raghunathan, and Jha present in [2] an analysis of the energy consumed by mobile devices when using several combinations of security mechanisms in SSL-based applications. Various cryptography (RSA, DSA, and ECDSA) and hashing (MD2, MD4, MD5, SHA, SHA1, and HMAC) algorithms have been used in the experiments.

Other work related to PDA energy consumption was published by Karri and Mishra in [3]. The authors measure the energy consumed by the device (i) when secure WAP (Wireless Application Protocol) sessions are established and (ii) during secured data transfer. An additional contribution of the paper is the proposal of techniques to reduce energy consumption. By applying techniques based on information compression, session negotiation protocol optimization, and hardware acceleration of crypto-mechanisms, the energy consumed for session establishment has been reduced by more than 6.5 times, when compared to the normal power consumption. Similarly, the energy for data transmission has diminished more than 1.5 times.

The overhead introduced by WEP (Wireless Equivalent Privacy) and IPSec protocols in IEEE 802.11b wireless networks has been measured by Maciel et al. in [4]. The data throughput achieved by desktop computers (with wireless cards attached to them) has been calculated under two different configurations: employing (i) solely WEP and (ii) both WEP and IPSec. This comparison is of little practical utility, however. WEP becomes unnecessary when IPSec is used, because besides being vulnerable, the first leads to an undesired additional overhead.

In this paper we measure the data reception rate and the energy consumed by a Personal Digital Assistant with and without the employment of IPSec protocol. We identify the type of applications that can be efficiently executed by the portable device even when security mechanisms are employed. We also characterize how much these mechanisms impact the autonomy of the PDA.

3 A Brief Review of IEEE 802.11b Vulnerabilities

The IEEE 802.11b standard uses the Wireless Equivalent Privacy (WEP) mechanism to protect MAC protocol data units (MPDU). It employs the default key (also known as key-mapping key or KeyID) and the RC4 algorithm for encryption. Integrity is achieved through CRC-32, which is used to compute an Integrity Check Value (ICV) over the MPDU data. The resulting 32-bit ICV is appended to the MPDU prior to encryption. In turn, the RC4 key is composed of a 24-bit Initialization Vector (IV) value concatenated with the default key (previously defined in the client and in the access point) to form a per-packet key. The MPDU data and ICV are then encrypted using the per-packet key. The IV and key identifier are pre-pended to the encrypted MPDU data field, forming the complete WEP Protocol data unit as shown in figure 1.

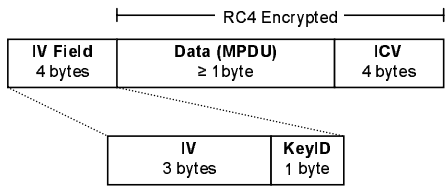


Fig. 1. WEP protocol data unit

The WEP protocol uses a single shared key, which is common to all the wireless network users. Whenever this key is compromised, it needs to be changed in all devices. The most serious problem, however, is that WEP encryption keys can be recovered through cryptanalysis. In August 2001 Fluhrer, Martin and Shamir described a new attack (FMS attack) on this construction [5]. They showed that an eavesdropper who can obtain several million encrypted packets, whose first

byte of plain text is known, could deduce the base RC4 key by exploiting properties of the RC4 key schedule. This kind of attack compromises WEP security. In the moment that the RC4 key is discovered, a malicious user could decrypt intercepted packets and also read encrypted traffic, violating the requirement of confidentiality. The user could also forge new encrypted packets, which would be accepted by the access point, defeating the integrity and authentication goals. Other vulnerabilities of the standard are described in [1].

The Temporal Key Integrity Protocol (TKIP - IEEE 802.11i), initially referred to as WEP2, is an interim solution that fixes WEP key reuse problem. It uses a 128-bit temporal key shared among clients and access points. This short-term key is combined with the client MAC address and a 16-bit IV to produce the key that will be used to encrypt data. Although this scheme can be adopted in current access points (through firmware upgrade), not all companies have provided a patch for their devices.

The recent IEEE 802.11X standard is based on authentication protocols such as EAP-TLS (Extensible Authentication Protocol) and LEAP (Lightweight Extensible Authentication Protocol), and provides port-based access control and mutual authentication via an authentication server. It uses digital certificates for authentication and dynamic distribution of encryption keys to solve the 802.11b key problem. The main drawback of this solution, however, resides on the fact that it is not compatible with currently available access points.

4 A Secure Wireless Local Area Network Setup

In addition to the approaches just mentioned, there are other that can be applied to secure current IEEE 802.11b wireless networks with no extra investments in hardware: IPSec (IP Security) [6], CIPE (Crypto IP Encapsulation) [7], and VTUN (Virtual Tunnel) [8] at the network-layer; SSL (Secure Socket Layer) [9] at the transport-layer; SET (Secure Electronic Transaction) [10], and OpenVPN [11] at the application layer.

CIPE, VTUN, and OpenVPN are not supported by mobile device operating systems such as PalmOS and PocketPC 2003. SSL and IPSec are by far the most deployed schemes. The former is used to provide application-specific end-to-end encrypted transfers. The latter, on the other hand, offers a general purpose cryptographic tunnel capable of providing secured communication to any application running on the PDA. Due to this generality, we have chosen to use IPSec in our experiments.

The setup is composed of a L2TP (Layer 2 Tunneling Protocol) [12] and an IPSec server (FreeS/WAN [13]) running on the gateway (figure 2). L2TP/IPsec is one the mechanisms that can be used by Pocket PC 2003 to acquire a virtual IP address from the internal network. This scheme has been chosen because (i) it is used by Pocket PC 2003's built-in VPN client (which is free!) and (ii) it is an official IETF standard.

We have configured the IPSec server to run in tunnel mode, i.e. both the header and the payload of packets sent/received by the PDA to/from the gateway

are encrypted. Although this is a very conservative setup, it has been used so that worst case measurements could be made.

The authentication process used was PSK (Pre-shared Key). A Pre-shared Key is a secret password that is shared by both sides of the IPSec tunnel. Preferably, the PSK is distributed through "out-of-band" medium, such as phone call, paper, face to face, and should not be transmitted over public networks.

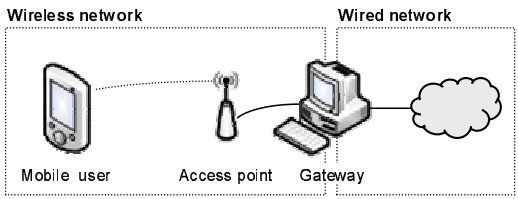


Fig. 2. IPSec-based PDAs access to 802.11b wireless LANs

5 Experimental Setup

Mobile devices face problems on battery working time and packet processing, which get critical when CPU and network-intensive applications are executed. Due to these restrictions, it is important to figure out the impact of the security mechanisms intended to be used in the wireless network infrastructure (the idea is to avoid imposing many extra limitations to the use of the device). To better understand the relation between data reception rate and energy consumption under different scenarios, we have carried out some experiments (described henceforth).

5.1 Testing environment

The setup of the experiments was composed of a client and a server (gateway). The client was an iPAQ 5550 with a 400MHz Xscale processor 128MB RAM, running Pocket PC 2003 operating system. The gateway was an Intel Celeron 500MHz 128MB RAM. In order to provide support for IPSec in the gateway, we have installed the following software: Debian Linux, kernel 2.6 [14] (with native support for IPSec), FreeS/WAN[4], and L2TP. The communication between the client and the gateway was done through a Linksys WAP11 access point (IEEE 802.11b), located around 15 meters far from the client.

The energy consumed by the mobile device during the data transmissions has been measured through the battery output voltage and the electric current data, which were acquired with an oscilloscope. The circuit implemented for acquiring these signals is showed in figure 3. The oscilloscope used was an Agilent 54622D MegaZoom, 100MHz, 200MSa/s. To monitor the electric current, a shunt resistor of 0.1Ω has been applied.

[4] FreeS/WAN had to be patched to support NAT-T (Traversal NAT for UDP packets).

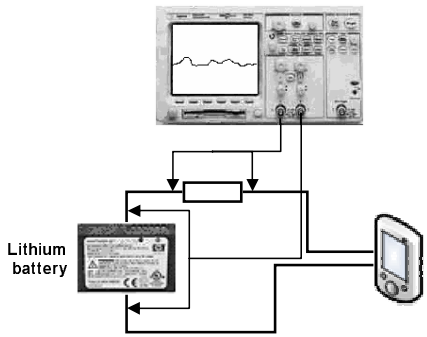


Fig. 3. Circuit for measuring the energy consumed by the mobile device

The determination of the energy consumed (W) has been obtained by the integration of the voltage x electric current, and can be expressed by the following equation: $W = \int v.i.dt(Joule)$.

5.2 Experiments

We have carried out three experiments in order to verify the impact of IPSec on the mobile device. Each experiment has been repeated five times to calculate the average and the standard deviation. To execute them we have developed two applications. The first, running on the gateway, was responsible to send UDP packets to the mobile device. The second, running on the PDA, was responsible to analyze the number of received UDP packets so that we could measure the reception rate in Mbps.

The first experiment assessed the maximum reception rate achieved by the mobile device (using IPSec). In this test we analyzed the data flow with no speed control during a 180-second period. We have used different PDU sizes: 256, 512, and 1024 bytes. 3DES and SHA1 algorithms have been applied, respectively, for encryption and integrity checking.

The second experiment aimed at measuring energy consumption and packet reception rate in controlled speed UDP stream transmissions. We have divided it in two groups: low speed (56, 128, and 256Kbps) and high speed stream transmissions (1, 2, 4, and 8Mbps). In low speed tests each transmission lasted 8 minutes (for better accuracy), while in high speed tests they ran for 3 minutes. For all these tests 1024-byte PDUs have been used. Again, IPSec traffic has been encrypted using 3DES and SHA1.

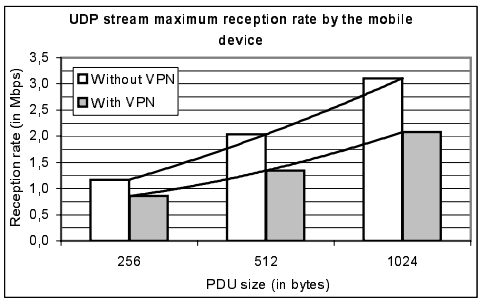
In order to get a more refined view on energy consumption and packet reception rate we have repeated the previous UDP stream transmission at 2Mbps using several PDU sizes (256, 512, 1024, and 2048 bytes).

In the third experiment we have evaluated the impact of using some combinations of encryption and hashing algorithms on the PDA energy consumption. We transmitted 1024-byte UDP packets at 2Mbps from the gateway to the PDA,

using DES and 3DES cryptographic algorithms combined with SHA1 and MD5 integrity algorithms.

6 Results

Figure 4 shows the maximum UDP stream reception rate achieved by the mobile device. From the graph one can observe that although the nominal capacity of WiFi wireless LANs is 11Mbps, the maximum reception rate achieved by the PDA was 3,109Mbps (using 1024-byte PDUs). This low rate is due to the mobile device limited CPU, which is not able to process so many packets in a short time period. As expected, the PDU size affects directly the reception rate regardless of whether IPSec is used or not. However, the use of IPSec always cause the reception rate to decrease compared to the non-encrypted transmission: 27% with 256-byte PDUs, 34% with 512-byte PDUs, and 33% with 1024-byte PDUs.



PDU	Without VPN	With VPN
	Pkts rcvd	Pkts rcvd
256	107.644	78.762
512	93.756	61.820
1024	71.633	47890

PDU	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
256	1,168	0,039	0,855	0,004
512	2,035	0,013	1,342	0,048
1024	3,109	0,066	2,079	0,012

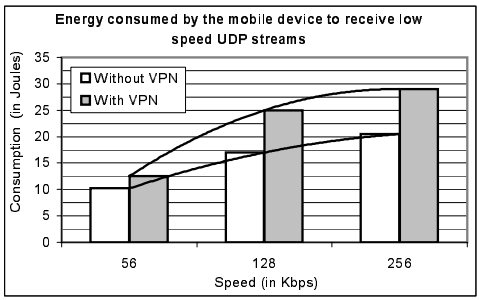
Fig. 4. Maximum reception rate achieved by the mobile device

In the second experiment two aspects have been analyzed: successful packet reception rate and energy consumption overhead[5]. As already mentioned in section 5.2, this experiment has been divided in two groups: high speed and low speed stream transmissions. The packet reception rate for the low speed data stream transmissions was almost 100%. This was expected, since in the previous experiment we have shown the PDA is able to cope with the 3,109Mbps reception rate.

Figure 5 shows the energy overhead consumed by the mobile device to receive low speed UDP streams. The consumption is directly affected by the transmission speed (the higher the speed, the higher the number of packets to be processed). It is worth observing in the graph how much the use of IPSec affects energy consumption in each transmission rate. The overhead of using IPSec at 56kbps is

[5] The overhead has been calculated by decreasing the PDA absolute energy consumption value after a fixed period data transmission from the energy consumed by the device during an equivalent time period when it was idle.

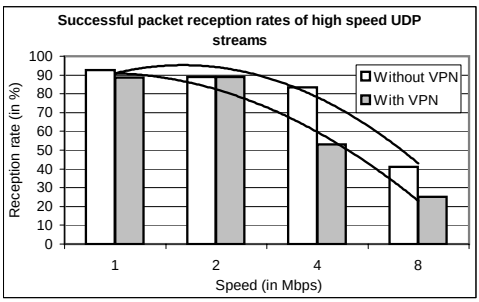
18%. This difference gets bigger as the transmission rate increases, achieving 47% for 256kbps. Taking into account the total battery energy is 13.72KJ (measured prior to the experiments), the consumption of 25J to receive a 8-minute UDP stream at 128Kbps using 1024-byte packets corresponds to 0,001% of the battery capacity.



Speed	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
56	10,200	23,091	12,500	12,261
128	17,000	2,646	25,000	9,899
256	20,500	2,121	29,000	11,314

Fig. 5. Mobile device energy consumption overhead to receive low speed UDP streams

For the high speed data stream transmissions it is valuable to illustrate two graphs: successful packet reception rate (figure 6) and energy consumption overhead (figure 7). The first shows the number of UDP packets received and processed by the mobile device in relation to the number of packets transmitted by the gateway under different speeds (from 1 to 8Mbps). From the graph one can infer that 2Mbps is the maximum reception rate the PDA using IPSec is able to handle with less than 20% of packet loss. When VPN is not used this rate grows up to 4Mbps. Above these rates the PDA receives less than 60% of the packets, which is not acceptable for applications such as video streaming.

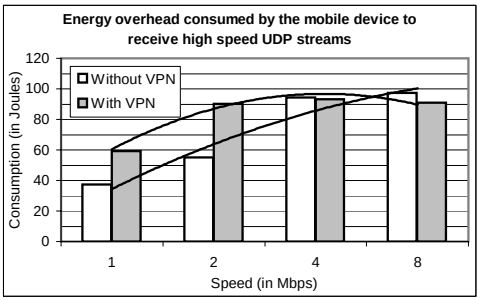


Speed	Without VPN	With VPN
	Pkts rcvd	Pkts rcvd
1	20.864	19.975
2	39.990	39.995
4	75.203	47.834
8	73.885	45.125

Speed	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
1	92,729	9,793	88,778	0,050
2	88,867	0,031	88,878	0,003
4	83,558	0,034	53,148	0,276
8	41,047	0,192	25,069	0,004

Fig. 6. Mobile device successful packet reception rates of high speed UDP streams

The second graph (figure 7) shows the energy overhead consumed by the mobile device to receive high speed data stream transmissions. Since its maximum reception rate is 3.109Mbps (figure 4), the energy consumption reaches the maximum value between 2 and 4Mbps (around 90J). The PDA consumes 37% more energy when IPSec is used to transmit/receive streams at 1Mbps, and 64% at 2Mbps. At 4Mbps and 8Mbps the energy consumed does not grow proportionally, because the number of packets received and processed is similar to what happens at 2 Mbps (figure 6); most of the packets are lost in these rates.



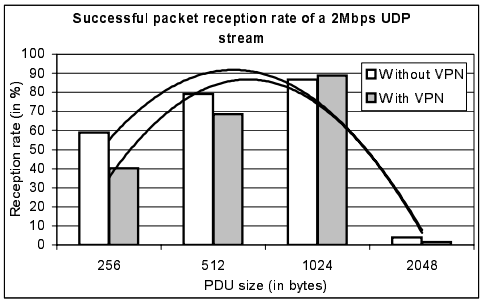
Speed	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
1	37,333	4,243	59,333	0,000
2	55,083	5,620	90,333	0,000
4	94,333	8,185	93,333	0,000
8	97,333	2,828	90,833	0,707

Fig. 7. Mobile device energy consumption overhead to receive high speed UDP streams

Figures 8 and 9 illustrate a zoomed view of the mobile device successful packet reception rate and energy consumption overhead when UDP streams at 2Mbps are transmitted to it. As one can notice in figure 8, the reception rate increases as larger packets are used (up to 1024 bytes). When 2048-byte long packets are transmitted by the gateway, they are fragmented and the mobile device reception rate drops unexpectedly to less than 10%. This is a good indicative that network intensive applications such as streaming video should be tuned to use the largest packet size that can be sent without fragmentation.

The energy overhead consumed by the mobile device to receive a 2Mbps UDP stream is shown in figure 9. The use of IPSec imposes a considerable overhead in the energy consumption compared to the non-encrypted transmissions: 3,72% for 256 byte packets, 27,67% for 512 byte packets, 50,22% for 1024 byte packets, and 53,12% for 2048 byte packets. Regarding the consumption associated with the first three transmissions, they are almost equivalent. This is explained in figure 8, where one may notice the packet reception rates increase as less, larger packets are transmitted by the gateway. When 256 byte PDUs were used, 72.514 packets have been received and processed by the PDA. On the other hand, only 39.983 packets have been received when 1024 byte PDU were transmitted. A lot of the consumption in the first case is related to header protection. Therefore, the best tradeoff between successful reception rate and energy consumption is reached when 1024-byte long packets are used.

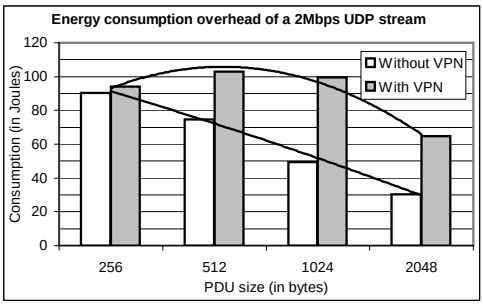
Figure 10 shows the results of the third experiment, where we have assessed the impact of using different encryption (DES and 3DES) and integrity checking



PDU	Without VPN	With VPN
	Pkts rcvd	Pkts rcvd
256	106.273	72.514
512	71.167	61.748
1024	39.058	39.983
2048	867	324

PDU	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
256	59,041	8,630	40,285	5,452
512	79,074	1,758	68,609	2,377
1024	86,796	0,566	88,850	0,024
2048	3,851	0,651	1,440	0,314

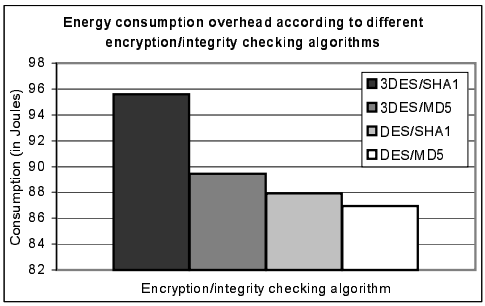
Fig. 8. Packet reception rate of a 2Mbps UDP stream using different PDU sizes



PDU	Without VPN		With VPN	
	Avg	Std Dev	Avg	Std Dev
256	90,494	12,021	93,994	8,485
512	74,494	0,707	102,994	1,414
1024	49,494	0,707	99,438	0,707
2048	30,438	2,121	64,938	1,414

Fig. 9. Energy consumption overhead to receive a 2Mbps UDP stream

(MD5 and SHA1) algorithms on the PDA energy consumption. As depicted in the graph, the combination 3DES/SHA1 is the most computational intensive. It is explained by the higher complexity of both algorithms compared to DES/MD5 [15].



Algorithms	Consumption	
	Average	Std Dev
3DES/SHA1	95,605	6,658
3DES/MD5	89,438	2,121
DES/SHA1	87,938	0,000
DES/MD5	86,938	1,414

Fig. 10. Energy consumption using different encryption/integrity checking algorithms

7 Conclusions

In this paper we have revisited some of the existing flaws in 802.11b networks. Several approaches to solve them have been commented. Then we have described the IPSec-based setup and the testing environment. The experiments carried out aimed at assessing the impact on data reception rate and energy consumption of IPSec-based PDAs access to 802.11b wireless LANs.

From the results obtained, it is important to highlight the maximum reception rate achieved by the mobile device is less than 50% of the nominal capacity (it gets worse when IPSec is used). The PDA energy consumption increases considerably when the security mechanisms are employed. It is also worth mentioning that the mobile device does not cope well with fragmentation. Depending on the number and size of the UDP packets it is not able to process more than 50% of them. Finally, we found out 3DES/SHA1, which are the the most common used encryption/integrity checking algorithms, are the ones that consume more. In a mobile computing environment, where maximizing the battery working time is highly desirable, DES/MD5 could alternatively be used.

References

1. Cam-Winget, N., Housley, R., Wagner, D. and Walker, J.: Security Flaws in 802.11 Data Link Protocols. Communications of the ACM **46** (2003) 35–39.
2. Potlapally, N., Ravi, S., Raghunathan, A. and Jha, N.: Analyzing the Energy Consumption of Security Protocols. Dept. of Electrical Engineering, Princeton University (2003).
3. Karri, R., Mishra, P.: Optimizing the Energy Consumed by Secure Wireless Sessions: Wireless Transport Layer Security Case Study. Mobile Networks and Applications **8** (2003) 177–185.
4. Maciel, P., Nunes, B., Campos, C. Moraes, L.: Avaliando a Sobrecarga Introduzida nas Redes 802.11 pelos Mecanismos de Segurança WEP e VPN/IPSec. 3rd Brazilian Workshop on Security of Computing Systems (2003) 61–68 (in Portuguese).
5. Fluhrer, S., Mantin, I., and Shamir, A.: Weaknesses in the Key Schedule Algorithm of RC4. Lecture Notes in Computer Science. **2259** (2001) 1–24.
6. Kent, S., Atkinson, R.: Security Architecture for the Internet Protocol. IETF RFC 2401 (2004).
7. Crypto IP Encapsulation. `http://www.extra300.nl/cipe.htm` (2004).
8. Virtual Tunnels over TCP/IP Networks. `http://vtun.sourceforge.net` (2004).
9. Freier, A. O., Karlton, P., Kocher, P. C.: The SSL Protocol Version 3.0. IETF Internet Draft (1996).
10. Secure Electronic Transaction. `http://www.setco.org` (2004).
11. OpenVPN. `http://openvpn.sourceforge.net` (2004).
12. Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., Palter, B.: Layer Two Tunneling Protocol L2TP. IETF RFC 2661 (1999).
13. Linux FreeS/WAN. `http://www.freeswan.org` (2004).
14. Linux Kernel Archives. `http://www.kernel.org/` (2004).
15. Stallings, W.: Network Security Essentials. Prentice Hall (2002).