Performance of secured communications over Bluetooth

S. A. Shaikh and S. Mills
Broadlands 202, Park Campus
Department of Multimedia and Computing
University of Gloucestershire Business School
Cheltenham Spa, GL52 2QF, UK
Tel: +44 (0)124 254 3354
Fax: +44 (0)124 254 3327
Email: {sshaikh, smills}@glos.ac.uk

Abstract

The concept of a Virtual Private Network (VPN) offers a considerably cheaper alternative to dedicated secure networks in corporate networks and Internet environments. The choice, however, of specific security protocols affect issues such as data throughput and performance throughout the network. Deploying a VPN over a wireless network adds yet another dimension to this. This paper looks into the performance of a VPN deployed over a Bluetooth network. Two security protocols are used in this investigation: IPSec and SSH, both running on a Windows XP platform. Performance of the two protocols is, subsequently, analysed and discussed.

Keywords: Bluetooth, IPSec, SSH, wireless, security, performance

1 Introduction

The Department of Multimedia and Computing at University of Gloucestershire recently carried out a study aimed at investigating the performance of secured communications over Bluetooth networks. One of the disadvantages of enforcing network security is the performance trade-offs. The network intensive applications used these days and the increasing of use wireless networks lead to more traffic on networks with lesser throughputs. Any cryptographic processing only adds to that load.

The purpose of this paper is to attempt to evaluate the performance of built-in security in the Bluetooth standard [1]. We then deploy two industry-standard security protocols, IPSec [2] and SSH [3] [4], and evaluate their performance over a simple Bluetooth network. The purpose of this effort is to evaluate some possible security solutions for small peer-to-peer Bluetooth networks.

1.1 Remainder of this Paper

The rest of this paper is organised as follows. Section 1.2 introduces the Bluetooth standard and Section 1.3 describes our experimental Bluetooth network. Section 2 introduces IPSec and SSH in detail relevant to this paper. Section 3 presents the results of our performance tests. Section 4 discusses the results further and highlights relevant issues. Section 5 finally concludes the paper.

1.2 Bluetooth standard and security

Bluetooth [1] is a Personal Area Networking (PAN) protocol designed as a cable-replacement technology providing a low cost and low power radio communications, operating in the 2.4 GHz ISM band and therefore requiring no line of sight. Originating in 1999, it intended to provide robust services to small ad hoc networks supporting speeds of 1-2 Mbps. It supports multipoint along with point-to-point connections and works in a small confined area with a range of no more than 10 metres.

The original Bluetooth specifications include security mechanisms at various levels of their protocol stack. The standard uses a combination of a Personal Identification Number (PIN) and a hardware address to identify other devices. Encryption can be used to further enhance privacy. The widely accepted algorithm SAFER+ (Secure And Fast Encryption Routine) [5], [6] is used for key generation/exchange and data encryption. At the bit level, the user controls authentication by using a 128 bit key and radio signals can be coded with 8 bits (or up to 128 bits). This can be triggered via the Host Controller Interface (HCI) commands. The transmission scheme at the radio level attempts to provide another level of security as instead of transmitting over one frequency within the 2.4 GHz band, Bluetooth radios use a Fast frequency-Hopping Spread Spectrum (FHSS) technique allowing only synchronized receivers to access the transmitted data.

1.3 Bluetooth Test Network

This section describes the physical and logical aspects of the test network. The network's physical arrangement is intended to remain constant throughout the testing phase; its software configuration will change. Figure 1 shows the physical setup of the network. The network is configured as a single network and the nodes are assigned a Class C network address. Two portable PCs were used as nodes with each running a Pentium IV processor with processor speed of 2.4 GHz supported by 512 MB of RAM.

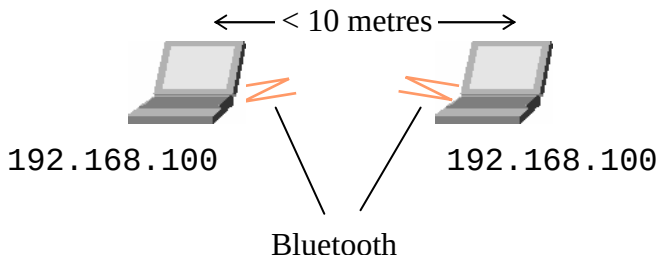

Figure 1: Test Network

The setup in Figure 1 depicts a very simple IP network based on two nodes connected through 3Com [7] Bluetooth network adapters. The nodes form a Wireless Personal Area Network

(WPAN) that essentially allows these nodes to form a small IP-based wireless LAN along with any other nodes that are in-range and ready to join.

The test setup in Figure 1 is re-presented as a peer-to-peer VPN setup in Figure 2. In this setup, two independent nodes are part of a Bluetooth wireless network and use some form of security to communicate with each other. This *secured communication* essentially is a logical tunnel that the two nodes can use to communicate *privately* in an otherwise *public* wireless environment.

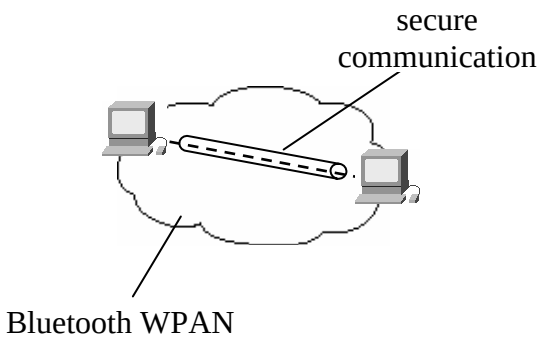

Figure 2: Peer-to-peer VPN model

2 Security protocols

This section would discuss the two security protocols used in this investigation: IPSec and SSH.

2.1 Internet Protocol Security (IPSec)

IPSec [2] is a protocol suite that provides a secure way of communicating over the TCP/IP protocol. The IPSec protocol is a set of security extensions developed by the IETF [8] and provides what is known as *packet-level security*.

IPSec provides strong authentication preventing any interception of data by falsely claimed identities; use of IP authentication headers and variations of hash message authentication code ensuring data integrity during communications; confidentiality services to prevent unauthorized access to sensitive data while it passes between communicating parties; dynamic re -keying during ongoing communications helping to protect against replay attacks; transparency by existing below the transport layer, making it transparent to applications and users.

To protect the contents of an IP packet, the data is transformed using cryptography. There are two main transformation types that form the building blocks of IPSec, the Authentication Header (AH) transformation (responsible for authentication) [9], and the Encapsulating Security Payload (ESP) transformation (responsible for data encryption) [10]. These are configured in a data structure that is called a Security Association (SA). It provides an open industry-standard alternative to proprietary IP encryption technologies; the resulting interoperability is understandably beneficial.

Top

2.2 Secure Shell (SSH)

SSH Secure Shell [3] is a program that allows secure network services over an insecure network. The concept originated in UNIX as a replacement for the insecure "Berkeley services", that is, the *rsh*, *rcp*, and *rlogin* commands. It replaces other, insecure terminal applications (such as Telnet and FTP). It allows secure login to remote host computers, to execute commands safely in a remote computer, to securely copy remote files, to forward X11 sessions (on UNIX), and to provide secure encrypted and authenticated communications between two non-trusted hosts. Also arbitrary TCP/IP ports can be forwarded over the secure channel, enabling secure connections.

Primarily based on the UNIX platform, it is currently available for the Windows platform and was created by SSH Communications Security, a Finland-based corporation. Its main advantages are its comparatively cheaper cost and free licensing for educational and non-profit enterprises. This makes SSH much more attractive for large-scale deployments. An earlier version has been succeeded by Version 2, known as SSH2, reducing some known security vulnerabilities that the SSH1 had. SSH [3] presently does not provide a complete VPN solution, but it does offer features such as TCP port forwarding, which can be used to tunnel other TCP applications securely through it. It does provide support for a variety of encryption and message authentication algorithms.

3 Performance Results

This section discusses the performance tests and how they are carried out followed by the results of testing security features built-in in the Bluetooth standard and the tests carried out deploying IPSec and SSH protocols over Bluetooth.

3.1 Performance Testing

An attempt will be made to determine the performance levels for the test network setup without any security implementation (encryption) and compare it with the implementation of IPSec and SSH tunnelling. To determine the performance of the network, data is transmitted from one node to another. To calculate the throughput, files were transferred using File Transfer Protocol (FTP) and then the time taken to transfer the complete file was logged. The file size was 5 MB. Each time the file was transferred three times and only the average of the three transfers was recorded. The data throughput is then calculated. The measure used throughout the tests is the data throughput through the channel. The throughput determined in Kilobytes per second (denoted by Kbps) reflects the efficiency of the communication channel. The throughput calculated for different setups and configurations, is then compared with each other in order to establish a comparative viewpoint. The higher the throughput, the more effective and quicker a setup is. Differences in throughputs achieved are also worked out as network latency as a percentage. The latency is an important measure as it gives an idea how much overhead does a secured setup introduce in a Bluetooth environment. It is also possible for a secured setup to increase the throughput i.e. be more efficient than the regular setup (as can be seen in the later sections). A negative latency means that the performance has gone down while a positive latency measure would mean that the performance of the network has improved. The formula used to calculate latency (as a percentage) for a test setup *setup 1* against a *regular setup* is:

$$Latency = \frac{Throughput\ difference}{regular\ setup\ performance} \times 100$$

where

$$Throughput\ difference = setup\ 1\ performance - \ regular\ setup\ performance$$

Every other element involved in the test is kept constant and unchanged throughout the different configurations. There were a few settings that were constant; the channel capacity as Bluetooth was used throughout the tests; the services being run on the machines are the same and no extra load was introduced at any stage. The readings that were recorded throughout the tests are not guaranteed to be accurate. They were obtained from the various tools used, each of which may not be dead precise or accurate. It was made sure that no external or internal problems or factors are present that may affect the test network's apparent capability. Every measurement was repeated three times and an average of multiple samples obtained was taken. The best tools and techniques were employed that were available for the purpose.

3.2 Performance of Bluetooth standard

The first tests conducted were to look at the security features that are built in to the Bluetooth standard and then to compare their performances. The three levels of security are *non-secure*, *service level* and *link level* [1]. Whereas *non-secure* does not provide any form of protection, *service* and *link* level each provide a higher level of security than the previous level and is concerned with various aspects of communication, including user authentication.

The conducted tests revealed an interesting picture. Table 1 below shows the results of the network throughput for each level.

Table 1: Performance of Bluetooth security modes

Bluetooth Security		
Level of security	**Throughput** (Kbps)	**Latency** %
Non-secure	82.98	-
Service level	87.34	(-)5.25
Link level	100.16	(-)20.70

The latency column on the far right represents the percentage of latency introduced at each level against the *non-secure* mode. It is interesting (and unexpected) to see that the performance at any secured level is comparatively better than the unsecured level. This suggests that *service* and *link* level enforce some sort of mechanism that allows for greater throughput. One explanation is, of course, any advantages gained by using dedicated hardware used for the purpose of packet processing. This is common for many implementations of secured transmission technologies.

3.3 Performance of IPSec over Bluetooth

The Bluetooth security tests were followed by the deployment of IPSec and observing its performance over the Bluetooth network. The two nodes were running the Windows XP operating system, which provides built-in support for IPSec. The IPSec was configured to operate in transport mode such that all traffic between the two text nodes would be encrypted. One of the advantages with IPSec is the choice of many authentication and encryption

algorithms that can be used. The two tests conducted both used SHA-160 (Secure Hash Algorithm) [11] as the authentication method. SHA-160 is part of a Secure Hash Standard and is one of the effective ways to provide data integrity with origin identification.

In the two tests the encryption algorithm used was DES (Data Encryption Standard) [12] in one and Triple-DES [12] in the other. DES is one of the most commonly used symmetric encryption algorithms. This means the sender and receiver of the data must know the same secret key being used to encrypt and decrypt the data. It is a block cipher that transforms 64-bit data blocks under a 56-bit key, by means of permutation and substitution. The key size for the single DES is actually 64-bit but 8 parity bits are removed from it. DES has been recently criticised as being weak in some respects [13]; therefore Triple-DES was also tested to see whether it performed at least reasonably well. The results of the test are given in Table 2 below.

Table 2: Performance of IPSec over Bluetooth

IPSec		
	Throughput (Kbps)	**Latency** %
Non-secure	82.98	-
IPSec using DES	38.39	53.74
IPSec using Triple DES	37.89	54.34

The results show that IPSec is much slower, which is understandable since the extra amount of encryption and decryption processing involved. The latency introduced in the order of 53.74 % is considerably high and means that the effective throughput is cut by almost half. Given the Bluetooth channel capacity is only about 1 Mbps, the deployment of IPSec may not be an ideal solution. It is interesting to see, however, that the difference between the latencies of using DES and using Triple-DES is not much whereas the strength of the security provided by Triple-DES compared to DES is considerably higher, thus suggesting Triple-DES may generally be preferable over DES.

3.4 Performance of SSH over Bluetooth

The IPSec test was followed by the SSH performance test. In order to facilitate a SSH secured connection a SSH server was installed on one of test nodes so a secured FTP (SFTP) [14] service could be hosted for the other test node to act as a SSH client. The SSH Server is configured from an inbuilt configuration utility. It was configured for at least a single connection to the server and using port 22 by default. The listening address was configured to “0.0.0.0”, allowing the SSH Server to use the node’s designated IP address for listening to any clients requesting a connection. The windows-based SSH client, provides two interfaces, one as a “Secure Shell Client” and the other as a “Secure File Transfer Client”. The secure shell client provides a remote login to the SSH server while the other one provides a secured FTP service. The SSH client was installed on the other test node to access the secured FTP service on the SSH server. SSH also provides a liberal choice of algorithms; for the purpose of the test SHA-160 was used as an authentication method while DES was used for data encryption. The results of the test are given in Table 3 below.

Table 3: Performance of SSH over Bluetooth

SSH		
	Throughput (Kbps)	**Latency** %
Non-secure	82.98	-
SSH using DES	37.40	54.93

It is slightly disappointing to observe that the performance with SSH has not improved over IPSec. The interoperability and low system requirements are one of the reasons why SSH appeals to large enterprises for deploying network security on a larger scale. However, with almost 55 % percent latency introduced into the network, SSH may not be a very good choice for any wireless networks.

4 Analysis of performance results

This section would look at all the results that are obtained and, compare and discuss them. It begins by looking at some of the general issues that may or may not influence the performance of the network.

4.1 General influencing factors

It is important to understand the number of elements that may or may not influence the performance of secured communications of the type that are tested during this study. One of the first elements of communications to consider is the propagation delay. This is the actual time taken for the data to travel across the length of the channel. This factor remains constant in all readings as long as the test arrangements remain constant, i.e. the distance between the nodes and the transmission conditions. Hence, in determining the affects of security on performance this obviously plays no role.

A second influencing factor is the transmission delays in the network. The channel capacity provided by Bluetooth is almost 1 Mbps and it remained constant as well. There are differences in the channel utilisations when the security protocols are introduced. The size of the data packets increases as the packets carry an overhead of IPSec headers and ciphertext. An increased utilisation of the channel may also explain why the performance improves when various levels of Bluetooth security are in effect. During the course of the current study, channel utilisations were not measured due to the lack of resources. A wireless network sniffer capable of sniffing Bluetooth networks would be a useful tool to reflect on the capacity of the channel and its utilisation.

Finally the processing delay is also important. In all security that is applied there is some increase in the processing activity on the involved machines. The nature of increase obviously depends on the nature of the activity. This involves processes involving encrypting and/or decrypting packets, hashing the packets, and also the encryption/decryption key management procedures.

4.2 Comparing the performances

Figure 3 below presents a comparison of the performances of all the test setups that have been observed during this investigation. It is heartening to see that Bluetooth at the *link level* security mode is the most efficient setup even including the regular *non-secure* setup. This is of interest as any level of extra security processing such as encryption/decryption and key

management would introduce a negative latency into the network but given dedicated hardware to carry out this processing, performance has evidently improved.

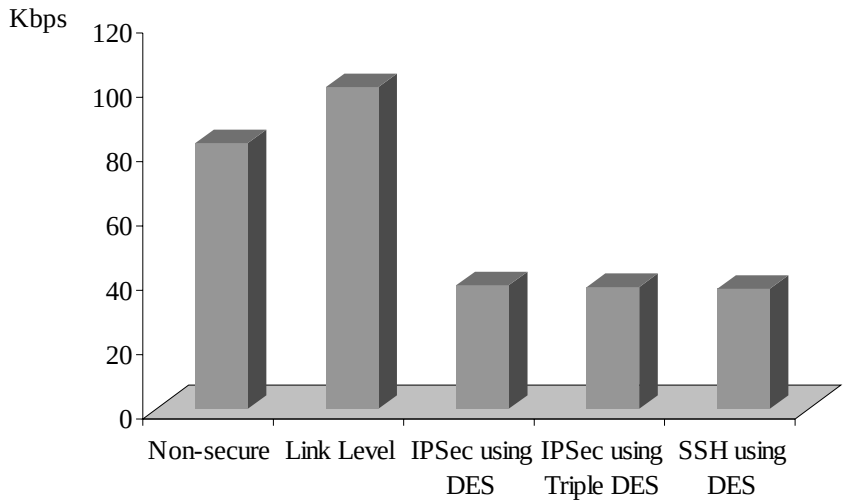

Figure 3: Comparing the performances

The Bluetooth standard [1] uses SAFER+, a byte-oriented block encryption algorithm, the cryptographic properties of which will not be discussed here as they are out of the scope of this discussion; interested readers are advised to consult [15] and [16] for further details. Although the performance (speed) of the algorithm used is not important here, but it is interesting to see that performance of various implementations differ. A review of some of the limited literature available [16] [17] [6] on this suggests that various hardware and software implementations of the algorithm provide unusually large differences in the algorithm's throughput achieved where some are considerably faster than the others. There is no indication as to whether any data compression takes place at any stage of the algorithm processing which may be another factor that contributes to the gain in throughput.

From a security perspective, the IPSec and SSH protocols used in the test were of interest. Both protocols are widely accepted as reliable and would certainly be more trusted against any inbuilt Bluetooth security. According to the results in Figure 3, the results for both IPSec and SSH are quite similar. The use of the more secure Triple-DES in IPSec is not hugely taxing on the performance either. Compared to the Bluetooth implementations, however, both protocols do not perform that well. This is understandable for reasons discussed earlier and in 4.1. While SSH is still in early stages of development IPSec has received considerable support from the industry, both on the interoperability and deployment front. To deal with the significant processing delay introduced by IPSec, an example of hardware support is discussed in the following section.

4.3 Hardware-supported IPSec processing

The nodes involved in IPSec secured communications spend a large part of their processing resources on the encryption and decryption of the packets. This processing incurs a huge overload and in a large-scale deployment of IPSec, performance is one of the main trade-offs. This trade-off could however be complemented by the use of hardware that supports IPSec packet processing. The hardware in this regard could be the unique network interface cards (NIC), which contribute to some of the IPSec processing. The packets pass through the NIC onto the channel and vice versa. The NICs therefore provide a perfect spot to deal with these

packets, working alongside the computer processor to share some of the processing and hence, offload some (or most) of the IPSec processing. These IPSec-offloading NICs can play a major role in improving the performance. A number of manufacturers provide these IPSec-offloading NICs including some of the popular vendors such as 3Com [7], IBM [18] and Intel [19]. The cards they provide work with a few of the main operating systems predominantly with Microsoft Windows. One of the advantages of using these specific IPSec-offloading NICs is their easy integration with the supported operating systems such as Windows. Due to the enhanced hardware compatibility provided by recent Windows operating systems, these cards usually require no setup or configuration, once their device drivers are procedurally installed.

5 Conclusion

This paper has focussed on client-to-client wireless VPNs. It has focussed on the Bluetooth network, specifically on issues related to secured communications and protocols and their performance. It has presented results and compared them. It has discussed some interesting issues that influence performance in wireless networks such as Bluetooth. It has also raised many queries that require careful attention and some theoretical reflection.

5.1 Recommendations

Performance testing secured communications and networks requires consideration and care in order for the results to be accurate and worthy. The test setups used in this study were inspired by [20] and [21]. It is important to consider more controlled and accurate test setups and adopt more precise testing procedures.

For the purpose of reliable security, it is important to investigate and evaluate the cryptographic properties of the built-in security in the Bluetooth standard; some important weaknesses have been highlighted in [22]. How these weaknesses relate to the security performance in the standard is of interest. IPSec and SSH protocols are independent of a specific standard such as Bluetooth but nevertheless some weaknesses still exist in them, as is highlighted in [23] for IPSec and [24] for SSH. Yet again their relationships with any of the performance measures presented in this paper would be of interest.

More attention should also be given to the nature of the application and the supporting infrastructure when it comes to securing applications within a Bluetooth network. This paper has looked at an IP over Bluetooth network; it will be interesting to see the performance measures of secured Bluetooth connections between ubiquitous and cellular devices. Since new wireless standards such as IEEE 802.11 a and b [25] and HomeRF [26] have emerged it will be interesting to see how the performance of secured communications at the IP level compare between various wireless standards. Some related work has already been undertaken by the authors [27].

References

[1] “Specification of the Bluetooth System”, Specification Vol. 1, Version 1.1, Feb 22, 2001

[2] RFC 2411, “IP Security Document Roadmap”. At http://www.ietf.org/rfc/rfc2411.txt, 2004

Top

[3] Internet Draft, “SSH Protocol Architecture”. At http://search.ietf.org/internet-drafts/draft-ietf-secsh-architecture-07.txt, 2004

[4] N. Ferguson and B. Schneier, “A Cryptographic Evaluation of IPsec”, Counterpane Internet Security, Inc., 2000. At http://www.counterpane.com, 2004

[5] J. L. Massey, G. H. Khachatrian, M. K. Kuregian, “SAFER+ Cylink Corporation’s Submission for the Advanced Encryption Standard”, First Advanced Encryption Standard Candidate Conference, Ventura, CA, August 20-22, 1998

[6] J. L. Massey, G. H. Khachatrian, M. K. Kuregian, “Nomination of SAFER+ as Candidate Algorithm for the Advance Encryption Standard”, First Advanced Encryption Standard Candidate Conference, Ventura, CA, August 20-22, 1998.

[7] Website, 3Com Inc. At http://www.3com.com, 2004

[8] Website, “Internet Engineering Task Force”. At http://www.ietf.org, 2004

[9] RFC 2402, “IP Authentication Header (AH)”. At http://www.ietf.org/rfc/rfc2402.txt, 2004

[10] RFC 2406, “IP Encapsulating Security Payload (ESP)”. At http://www.ietf.org/rfc/rfc2406.txt, 2004

[11] RFC 2404, “The Use of HMAC-SHA-1-96 within ESP and AH”. At http://www.ietf.org/rfc/rfc2404.txt, 2004

[12] U.S Department of Commerce/National Institute of Standards and Technology. FIPS PUB 46-3: Data Encryption Standard (DES), October 25, 1999. Federal Information Processing Standards Publication. At http://csrc.nist.gov/fips/fips46-3.pdf, 2005

[13] G. Sturm, “Encryption Standards: AES vs. DES”. At http://stud3.tuwien.ac.at/~e9825530/computerscience/aes/index.html, 2004

[14] Internet Draft, “SSH Transfer Protocol”. At http://search.ietf.org/internet-drafts/draft-ietf-secsh-filexfer-00.txt, 2004

[15] P. Kitsos, N. Sklavos and O. Koufopavlou, "Hardware Implementation of the SAFER+ Encryption Algorithm for the Bluetooth System", IEEE International Symposium on Circuits and Systems (ISCAS'02), Vol. IV, pp. 878-881, USA, May 26-29, 2002

[16] P. Kitsos, N. Sklavos, K. Papadomanolakis and O. Koufopavlou, "Hardware Implementation of the Bluetooth Security", IEEE Pervasive Computing, Mobile and Ubiquitous Systems, Vol. 2, No. 1, pp. 21-29, 2003

[17] J. L. Massey, “On the Optimality of SAFER+ Diffusion”, Second Advanced Encryption Standard Candidate Conference (AES2), Rome, Italy, March 22-23. At http://csrc.nist.gov/encryption/aes/round1/conf2/aes2conf.htm, 2004

[18] Website, IBM. At http://www.ibm.com, 2004

Top

[19] Website, Intel Inc. At http://www.intel.com, 2004

[20] S. Al-Khayatt, S. A. Shaikh, B. Akhgar and J. Siddiqi, “A Study of Encrypted, Tunneling Models in Virtual Private Networks”, Third IEEE Conference on Information Technology ITCC-2002, Las Vegas, Nevada, USA, April 8-10, 2002

[21] S. Al-Khayatt, S. A. Shaikh, B. Akhgar and J. Siddiqi, “Performance of Multimedia Applications with IPSec Tunneling”, Third IEEE Conference on Information Technology ITCC-2002, Las Vegas, Nevada, USA, April 8-10, 2002

[22] M. Jakobsson and S. Wetzel, “Security weaknesses in Bluetooth,” In Topics in Cryptology – CT-RSA 2001, Proceedings of the Cryptographer’s Track at RSA Conference 2001, Vol. 2020 of Lecture Notes in Computer Science, pages 176–191, Springer-Verlag, 2001

[23] Internet Engineering Task Force, Secure Shell charter. At http://www.ietf.org/html.charters/secsh-charter.html, 2002

[24] M. Bellare, T. Kohno and C. Namprempre, “Authenticated Encryption in SSH: Provably Fixing the SSH Binary Packet Protocol,” CCS’02, Washington, DC, USA, November 18-22, 2002

[25] IEEE 802.11b Standard. At http://standards.ieee.org/getieee802, 2004

[26] Website, “HomeRF”. At http://www.homerf.org, 2004

[27] S. A. Shaikh and S. Al-Khayatt “A Wireless Network for Multimedia Applications – a Case Study”, Third Symposium on Communication Systems, Networks and Digital Signal Processing, CSNDSP’02, Staffordshire, UK, 2002

Top