WIRELESS NETWORK SECURITY

Stefaan Seys **Dave Singelée** **Bart Preneel**

K.U.Leuven, Department Electrical Engineering–ESAT, SCD/COSIC
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

ABSTRACT

This article briefly describes the most important security protocols for wireless networks. It focuses on the the Bluetooth standard for Personal Area Networks (PAN) and on the IEEE 802.11 standard for Wireless LANs (WLAN). The strengths and weaknesses of these solutions is discussed and perspectives on further developments and improvements are presented. Finally the security issues for mobile ad hoc networks are introduced.

Key words: wireless security, PAN, WAN, ad hoc networks.

1 INTRODUCTION

Mobile devices and wireless technology are evolving at a rapid rate. The lines between handheld devices are beginning to blur, as cellular phones merge with PDAs, and notebooks take on desktop power. To be truly mobile, all should incorporate one or more wireless technologies, offering them the capability to form peer-to-peer Personal Area (PAN) and access Local Area (LAN) networks. Given that wireless radio frequency (RF) signals disperse beyond walls and buildings, the task of securing wireless RF transmissions is complex. Fortunately, these security issues are being solved and solutions have been standardized.

This paper gives an overview of the security techniques adopted by two common wireless technologies: Bluetooth and IEEE 802.11. Next to the techniques and protocols used to add security, we also discuss the weaknesses and solutions security experts have identified so far. Finally we also discuss the specific security issues of wireless ad hoc networks.

1.1 A security architecture for wireless networks

A security architecture consists of all security measures and techniques that are put into place in order to protect some system. Such an architecture can be seen as a layered structure in which the measures in the higher levels use the services provided by the lower levels:

1. *Cryptographic primitives* are building blocks that are used by all higher level security protocols. Symmetric key block ciphers and digital signatures are well known examples.
2. *Key management and authentication* are probably the most important security protocols as they are necessary for more advanced protocols such as secure routing systems. A key management scheme should provide functionality to securely distribute secret keys between the different nodes in the network.
3. The *security policy* defines the "rules" that the nodes have to obey. For example only allow authorized users to login to the wireless network or only allow RSN-secured connections (see section 3.4)). Obviously setting a policy is difficult and will have to be continuously reviewed.

2 BLUETOOTH SECURITY

In February 1998, the *Bluetooth Special Interest Group (SIG)* [3] was founded by the leaders in the telecommunications industries. The major task of this trade association was creating the Bluetooth specification which describes how mobile phones, computers, PDAs, headsets and other mobile devices can communicate wireless with each other. In June 2000, the Bluetooth standard was included in the Wireless Personal Area Network Working Group (IEEE 802.15 [8]). The Bluetooth wireless technology realizes a low cost short-range wireless voice- and data-connection through radio propagation. We will now discuss the security architecture of Bluetooth and give an overview of its security weaknesses.

2.1 Security architecture

The two most crucial parts of the security architecture are the key establishment protocol and the encryption algorithm. Suppose that two Bluetooth devices (called A and B) want to communicate securely. Initially these devices do not have a common shared secret. That is why they will perform the following steps:

Generation of the unit key: When a Bluetooth device is turned on for the first time, it calculates a so-called *unit key*. This is a key that is unique for every device and that is almost never changed. It is a function of a random number and the Bluetooth address (which is a factory-established parameter unique for every device).

Generation of the initialization key: The Bluetooth devices still do not share a session key. This will be done in different steps. First, an *initialization key* is generated. This temporary key is a function of a random number (which is generated by the device that initiated the communication), a shared Personal Identification Number (PIN) and the length L of the PIN. The PIN should be entered in both devices and the length can be chosen between 8 and 128 bits. If one of the devices does not have an input interface, a fixed PIN is used (often, the default value is 0000). The result is a temporary shared key (the *initialization key*).

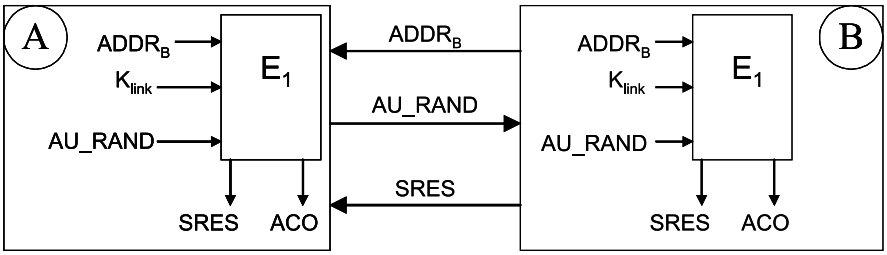

Figure 1: Mutual authentication protocol.

Mutual authentication: Both devices now share a key and will perform a mutual authentication protocol which is based upon a challenge-response protocol. This protocol is performed twice. First, B authenticates itself to A, as shown in Fig. 1. If this authentication is successful, the roles are switched (B becomes the verifier and A the prover). The authentication goes as follows. A generates a random number AU_RAND and sends this to B. This random number is called the *challenge*. Both devices now calculate a *response* $SRES = E_1(ADDR_B, K_{link}, AU\_RAND)$. The authentication function E_1 is based on the encryption function SAFER+. The algorithm is an enhanced version of an existing 64-bit block cipher SAFER-SK128 [15]. K_{link} is the shared key (*the initialization key*). If the response of B corresponds to the value that A has calculated, the authentication is successful. We omit the details of the protocol.

Generation of the link key: Both devices now share an initialization key. This key will be used to agree on a new, semi-permanent key (called the *link key*). The link key will be stored on both devices so that it can be used for future communication. If one of the devices does not have enough memory to store keys, the link key will be the unit key of the memory constrained device. This unit key will be encrypted with the initialization key and sent to the other device. If both devices have enough memory, the link key will be a combination key derived from the input of both devices and the shared initialization key.

Mutual authentication: After the generation of the link key, the old temporary initialization key is definitively discarded and a mutual authentication is started with the exchanged link key.

Generation of the encryption key: After a successful generation of the link key, the encryption key can be generated. This encryption key depends on the shared link key, a random number (generated by the device that initiated the communication) and the value ACO (which is the outcome of the mutual authentication protocol, as can be seen in Fig. 1). The encryption key can be reduced in length if necessary.

Generation of the key stream: The encryption key (or the length-reduced key) is fed to the stream cipher E_0 together with the Bluetooth address and the clock of the master. The result is a key stream which will be XORed with the data that has to be encrypted. The master clock is used in order to make the key stream harder to guess.

2.2 Security weaknesses

There are several security weaknesses in the Bluetooth standard (see for example Jakobsson and Wetzel [10]). Some of these problems are easily exploited by an attacker, other security weaknesses are rather theoretical. A brief overview of the most important problems will now be given.

Security depends on security of PIN: The initialization key is a function of a random number, a shared PIN and the length of the PIN. The random number is known by an attacker who is present during the initialization phase. This means that if an attacker obtains the PIN, she knows the initialization key. It even gets worse! Since all the other keys are derived from the initialization key, they will also be known by the attacker. The security of the keys depends on the security of the PIN. If it is too short or weak (e.g., 0000), it is very easy for an attacker to guess the PIN. Note that it is always possible to verify a PIN. The reason is that a mutual authentication protocol is executed after the generation of the initialization key. If an attacker observes this protocol, he obtains a challenge and the corresponding response. The attacker calculates for every guess of the PIN the corresponding response and when this is equal to the observed response, it is very likely that the correct PIN was used. The shorter the PIN, the faster this brute force attack can be executed.

Unit key: The unit key is used if one of the Bluetooth devices does not have enough memory to store session keys. This key is stored in non-volatile memory; it is almost never changed. The unit key is sent encrypted (with the initialization key) to the other device. This opens the door for an impersonation attack. It is recommended to avoid the use of unit keys.

Encryption algorithm: Bluetooth uses the encryption algorithm E_0. This stream cipher has some security flaws. Several attacks on E_0 are published, but most of these attacks do not work on the algorithm which implements E_0 in Bluetooth. There are however exceptions. Golic [9] has found an attack on the Bluetooth stream cipher that can reconstruct the 128-bit secret key with complexity about 2^{70} from about 45 initializations. In the precomputation stage, a database of about 2^{80} 103-bit words has to be sorted out. The attack uses a general linear iterative cryptanalysis method for solving binary systems of approximate linear equations. This attack has been further generalized in [11].

Denial-of-service attacks: Mobile networks are vulnerable to denial-of-service attacks. They consist of mobile devices and these devices are often battery fed. Bluetooth is no exception. An attacker can send dummy messages to a mobile device. When this device receives a message, it consumes some computation (and battery) power. After some time, all battery power will be consumed. This exhaustion of the battery power is called the *sleep deprivation attack* which was first introduced by Stajano and Anderson in [18] (see also 4.1). There are also some denial-of-service attacks caused by implementation decisions. A good example is the *black list* which is used during the mutual authentication protocol. To avoid that a device would start the authentication protocol over and over again, each device has a black list of the Bluetooth addresses of the devices which failed to authenticate themselves correctly. These devices can not start an authentication protocol during some period. This mechanism can be exploited in several denial-of-service attacks [6].

Location Privacy: When two or more Bluetooth devices are communicating, the transmitted packets always contain the Bluetooth addresses of the sender and the receiver. When an attacker eavesdrops on the transmitted data, she can keep track of the place and the time the two devices were communicating. This is a violation of the privacy of the user. The location information could be sold to other persons and used for location dependent commercial advertisements.

Bluesnarf attack: Recently, the *Bluesnarf attack* [14] was discovered. It is possible, on some mobile phones, to connect to the device without alerting the owner of the target device of the request, and gain access to restricted portions of the stored data in the phone, including the entire phone book (and any image or other data associated with the entries), calendar, real-time clock, business card, properties, change log, IMEI (*International Mobile Equipment Identity*,) ... This is normally only possible if the device is in *discoverable mode*, but there are tools available on the Internet that allow this safety net to be bypassed.

Bluejacking: When two Bluetooth devices are paired, they will send their "name" to each other. This user defined name can be up to 248 characters and will be displayed on the output-interface of the other device. The *Bluejacking attack* [2] exploits this name to send advertisements to Bluetooth devices.

3 WIRELESS LAN SECURITY

In June 1997, the IEEE body that defined the dominant 802.3 Ethernet standard, released the 802.11 standard for wireless local area networking. The IEEE 802.11 standard supports transmission in infrared light and two types of radio transmission within the unlicensed 2.4 GHz frequency band: Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). Today several 802.11 standards exist. 802.11b is an expansion of the standard that allows transmission speeds of up to 11 Mbps. The newer 802.11a and 802.11g versions support speeds as high as 54 Mbps.

3.1 Overview

IEEE 802.11 designers recognized the inherent difference between the wired and wireless environments, particularly with regard to medium access. Because the transport medium is shared, any client in transmission range of another client can process packets originating from that host. The three goals that need to be met to prevent unauthorized access to data and services in a wireless environment like WLAN are (1) authentication and authorization, (2) confidentiality and (3) integrity. The 802.11 standard has been through multiple changes to try to achieve these goals.

The Wired Equivalent Privacy (WEP) protocol was designed to protect data at the link layer and prevent unauthorized access to 802.11 data frames. Soon after the release of the first 802.11 compliant products (1999–2000) multiple weaknesses were discovered and tools like AirSnort [1] to break WEP security were developed [5, 7]. In 2001 the IEEE 802.11 Task Group i (TGi) was created to adapt the 802.11 Medium Access Control (MAC) in order to enhance security and authentication mechanisms. The enhancements proposed by the TGi would lead to the 802.11i (MAC Enhancements for Enhanced Security) standard.

In 2002, stating that the industry could not wait for the 802.11i's ratification, the industry consortium Wi-Fi Alliance introduced Wi-Fi Protected Access (WPA). It is a subset of the abilities of 802.11i, including better encryption, easier setup using a pre-shared key, and the ability to use RADIUS-based 802.1X authentication of users. In June 2004 the IEEE finally ratified the 802.11i standard. 802.11i has all the abilities of WPA and adds the possibility to use Advanced Encryption Standard (AES) for encryption of data. The downside is that AES support may require new hardware for many existing WLANs, as it needs a dedicated chip to handle the encryption and decryption.

3.2 Wired Equivalent Privacy (WEP) protocol

The first 802.11 hardware included the WEP protocol that could be used to optionally secure the connection between the Mobile Station (ST) and the Access Point (AP) [12]. A first drawback of the proposed security architecture is the absence of key management. Users have to manually input up to four secret keys into their device (and the AP). The key that was actually used also has to be selected manually by the user. If WEP is used then all the traffic between the ST and AP is encrypted and authenticated using WEP and the selected secret key. Two mandatory access control mechanisms are described in the standard. The default being "open system" or no access control and the second a challenge-response scheme based on WEP. Next to these, multiple vendor specific schemes were added. Fig. 2 shows the WEP encryption scheme. First a linear CRC-32 checksum (Integrity Check Value or ICV) is computed and appended to the payload. Next the stream cipher RC4 [16] is used to encrypt the plaintext. The key used to initialize RC4 consists of the concatenation of a 24-bit Initial Value (IV) and the secret key (42 or 104 bits). The resulting ciphertext together with the IV are sent to the receiver. The IV is changed for every packet.

As we already mentioned, multiple weaknesses were discovered soon after the release of the standard [5]. These weaknesses were due to three main problems with the scheme:

1. Small IV space: 24 bit is not sufficient. The birthday paradox states that after 5000 packets (with random IV choices) a first collision occurs with high probability. When an IV-collision occurs, then both resulting RC4 streams R are equal. An attacker can now XOR the two corresponding ciphertexts $C_1 = M_1 \oplus R$ and $C_2 = M_2 \oplus R$, resulting in $C_1 \oplus C_2 = M_1 \oplus M_2$. Because the sum of the two ciphertexts no longer contains the pseudo-random stream R, statistical attacks can be applied to obtain M_1 and M_2 from their sum.
2. The use of a linear, key-independent CRC-32 checksum in combination with a stream cipher. The CRC-32 computation involves no secret keying information and is intended for detection of *random* errors (for example introduced by the transport layer). In WEP, CRC-32 is used for data authentication, and the design rational was that an adversary would not be able to compute a correct CRC-32 checksum because it is protected by the RC4 encryption. In practice this is not the case [5]. An adversary can change messages of other users and insert messages of her own while still preserving a valid checksum, without knowing the secret key. This means that WEP fails to provide data authentication.

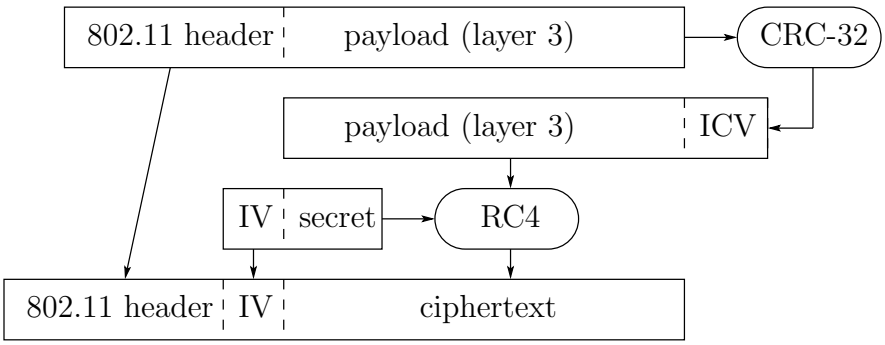

Figure 2: WEP data encryption and encapsulation.

3. No RC4 key-stream reuse detection. Once an attacker has captured one plaintext/ciphertext pair and hence the key-stream that was used to encrypt, this key-stream can be reused until the secret key is changed. Particularly if challenge / response is used for device authentication, then the attacker can easily obtain a plaintext (challenge) and ciphertext (response) pair and will be able to reuse the resulting key-stream.

The lack of key management even enhances these attacks because frequently changing the key is tiresome and consequently the key is hardly ever renewed in many situations.

Finally in July 2001, Fluhrer et al. presented an attack on RC4 as it is used in WEP [7]. This attack enables an attacker to compute the secret key that is used after passively capturing a sufficient number of cryptograms (1.3 million for a 128-bit WEP key; on a heavily loaded AP this would take less than 1 hour). This attack is linear with respect to the key size, thus enlarging the secret key does not help here. This attack was implemented in different tools and should be considered a real threat to 802.11 WEP-based security.

3.3 Wi-Fi Protected Access (WPA)

In 2002 the industry consortium Wi-Fi Alliance introduced Wi-Fi Protected Access (WPA) to provide a short-term solution for WLAN security. The Wi-Fi Alliance stated that WPA should be compatible with existing 802.11 hardware. This obviously limited the improvements that could be achieved because WEP is still used and the attack by Fluhrer et al. still applies. A new encryption/authentication scheme "Temporal Key Integrity Protocol" (TKIP) was built around WEP to enhance its security. TKIP uses a larger IV space (48 bit) with sequencing rules and adds a mechanism to derive per-packet WEP keys from a temporal secret key, the MAC address of the device and the packet sequence number. This ensures unique keys even if multiple STs share the same secret key. The temporal key itself is frequently changed (every 10000 packets) and based on a pairwise master key (PMK) and nonces that are exchanged between the AP and ST. Next to this, a new 64-bit keyed message integrity code (MAC), called Michael, is applied to the payload before encrypting it with WEP. Michael had to be very efficient (< 5 instructions/byte) and is not very strong, but it is a substantial improvement over CRC-32. Through these enhancements, TKIP addresses all WEP's known vulnerabilities.

Finally WPA also enables the use of the IEEE 802.1X framework for user or device authentication [13]. 802.1X offers an effective framework for authenticating and controlling user traffic to a protected network, as well as dynamically varying encryption keys. 802.1X ties a protocol called EAP (Extensible Authentication Protocol) to both the wired and wireless LAN media and supports multiple authentication methods, such as token cards, Kerberos, one-time passwords, and public key authentication. For details on EAP specifically, refer to IETF's RFC 2284 [4]. Initial 802.1X communications begins with an unauthenticated supplicant (ST) attempting to connect with an authenticator (AP). The AP responds by enabling a port for passing only EAP packets from the client to an authentication server located on the wired side of the access point. The access point blocks all other traffic until the access point can verify the client's identity using an authentication server (e.g., RADIUS) or a pre-established shared key. Once authenticated, the access point opens the client's port for other types of traffic. 802.1X can also be used to securely exchange PMKs.

Where the use of WEP was optional in the original 802.11 standard, the use of TKIP and 802.1X are mandatory in WPA.

3.4 802.11i and Robust Security Network (RSN)

In June 2004 the IEEE ratified the 802.11i standard. IEEE 802.11i defines a new type of wireless network called a robust security network (RSN). In order to join an RSN, a wireless device has to have the capacities discussed in the previous section. However, to facilitate an extended upgrade period, 802.11i defines a transitional security network in which both RSN and WEP systems can operate in parallel. WPA and RSN share a common architecture and approach. WPA has a subset of capacities focused specifically on one way to implement a secure network, whereas RSN allows more flexibility. RSN also supports the AES algorithm in addition to TKIP. Since current hardware does not support the full 802.11i standard, WPA provides for the needs of the current WLAN users, while in the long term the full RSN allows more flexibility.

4 SECURITY FOR MOBILE AD HOC NETWORKS

An ad hoc network is a collection of wireless mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration. Due to the limited transmission range of wireless network interfaces, multiple network "hops" may be needed for one node to exchange data with another across the network.

4.1 Security threats

Are there ad hoc network specific attacks that do not exist in other distributed networks, such as the Internet? In general, the answer is negative. The main difference is that for ad hoc networks the devices are physically accessible at every node. Thus traditional access

control mechanisms such as firewalls cannot be applied, as these systems assume a few well controlled access points through which all traffic to and from the "outside" world is channeled. In an ad hoc network, there is no inside and outside.

What can happen if no security is provided in the system?

- Eavesdropping of wireless communication is fairly easy. This means that without security an adversary could easily extract useful information from conversations between nodes.
- Without proper authentication mechanisms unauthorized people or devices could request services or data of the unprotected nodes. In many cases these services or data may not be public. Malicious users could also try to join the network undetected by impersonating as some other, trusted node. As a "trusted" node, it will now have access to private data or it can disrupt the normal network operations.
- Without proper security measures it is possible to trace the actions of any node in the network. If one or more of the nodes are carried by a user, an adversary is able to pinpoint the location of that user at any given time.
- A denial-of-service attack specific to (unattended) ad hoc networks is battery power exhaustion. Battery life is the critical parameter for the nodes in power-restrained networks (for example sensor networks) and many techniques are used to maximize it; in one technique, for example, nodes try to spend most of the time in a sleep mode in which they only turn on the radio receiver, or even the processor, once in a while. In this environment, energy exhaustion attacks are a real threat: without sufficient security, a malicious node could prohibit another node to go back to sleep causing the battery to be drained [18].

4.2 Challenges

- For several mobile ad hoc networks, security protocols and cryptographic algorithms have to operate at ultra-low energy budgets. Current state-of-the-art protocols and algorithms are not being developed for this purpose. Next to this, communications in the form of radio transmissions have to be weighted against computations.
- There might be large amounts of nodes, for example, thousands in ad hoc sensor networks. This means that ad hoc peer to peer communications are established, without the availability of a trusted higher authority that oversees distribution of keys or verifies identities.
- Because unattended ad hoc networks (for example sensor networks) are physically accessible, *tamper-resistance* is also required. This will be a difficult task, especially within the cost constraints of these (small) mobile nodes.

4.3 A security architecture for wireless ad hoc networks

Due to the specific properties of wireless ad hoc networks, a security architecture for these networks will have to meet specific demands:

1. *Cryptographic primitives.* Power consumption of these building blocks is critical in this mobile, energy constrained environment.
2. *Key management and authentication* has to be provided in an ad hoc fashion, without the use of online servers For example in [19] a distributed CA (Certification Authority) is proposed. Using the different properties of advanced secret sharing schemes, the distributed CA and the resulting PKI (Public Key Infrastructure) can be made very flexible and robust. Broadcast authentication is very useful to provide hop-by-hop data authentication in multi-hop networks [17].
3. *Security policy.* Stajano and Anderson propose a policy for ad hoc networks based on the behavior of ducklings: the first node a new node (duckling) connects to becomes his master (mother). From then on it will only listen to this mother node [18]. Different policies are required for other types of ad hoc networks like sensor networks.

5 CONCLUSIONS

We have given an overview of the security techniques included in Bluetooth and IEEE 802.11. Next to this we have also discussed the security issues of wireless ad hoc networks. It is clear from these discussions that providing security for wireless networks is hard and that great care and evaluation by experts is required to get it right. The problems with WEP are severe and it is not surprising that the IEEE has created a task group to remedy these problems quickly. Although Bluetooth security is not perfect, it seems that the problems are not considered severe enough by the industry to push the Bluetooth SIG to improve the security mechanisms.

REFERENCES

[1] AirSnort. ⟨`http://airsnort.shmoo.com/`⟩.

[2] Bluejacking. ⟨`http://www.bluejackq.com/`⟩.

[3] Bluetooth Special Interest Group. ⟨`http://www.bluetooth.com/`⟩.

[4] L. Blunk and J. Vollbrecht, "PPP extensible authentication protocol (EAP)," RFC 2284, 1998.

[5] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: The insecurity of 802.11," in *Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking*, pp. 180–189, ACM Press, July 2001.

[6] C. Candolin, "Security issues for wearable computing and Bluetooth technology," 2000. ⟨`http://www.tml.hut.fi/~candolin/Publications/BT/`⟩.

[7] S. Fluhrer, I. Mantin, and A. Shamir, "Weaknesses in the key scheduling algorithm of RC4," in *Proceeding of the Eighth International Annual Workshop on Selected Areas in Cryptography – SAC'01* (S. Vaudenay and A. M. Youssef, eds.), vol. 2259 of *Lecture Notes in Computer Science*, Springer-Verlag, 2001.

[8] IEEE Computer Society, "Wireless medium access control (MAC) and physical layer (PHY) specifications for: Wireless personal area networks, IEEE standard 802.15," 2002. ⟨`http://www.ieee802.org/15/`⟩.

[9] V. B. J. Golic and G. Morgari, "Linear Cryptanalysis of Bluetooth Stream Cipher," in *Advances in Cryptology - EUROCRYPT 2002*, vol. 2332 of *Lecture Notes in Computer Science*, pp. 238–255, Springer-Verlag, 2002.

[10] M. Jakobsson and S. Wetzel, "Security Weaknesses in Bluetooth," in *Proceedings of the Cryptographer's Track at the RSA Conference (CT–RSA '01)*, vol. 2020 of *Lecture Notes in Computer Science*, pp. 176–191, Springer-Verlag, 2001.

[11] P. Junod and S. Vaudenay, "Optimal key ranking procedures in a statistical cryptanalysis," in *Fast Software Encryption, FSE '03* (T. Johansson, ed.), vol. 2887 of *Lecture Notes in Computer Science*, (Lund,SE), pp. 235–246, Springer-Verlag, 2003.

[12] LAN/MAN Standards Committee of the IEEE Computer Society, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802.11, 1999 Edition.* 1999. ⟨`http://standards.ieee.org/getieee802`⟩.

[13] LAN/MAN Standards Committee of the IEEE Computer Society, *IEEE Standards for Local and Metropolitan Area Networks: Port-Based Network Access Control, IEEE Standard 802.1X, 2001 Edition.* 2001. ⟨`http://standards.ieee.org/getieee802`⟩.

[14] A. Laurie and B. Laurie, "Serious flaws in Bluetooth security lead to disclosure of personal data.". ⟨`http://bluestumbler.org/`⟩.

[15] J. L. Massey, "SAFER K-64: A byte-oriented block ciphering algorithm," in *Proceedings of Fast Software Encryption (FSE' 04)*, vol. 809 of *Lecuture Notes in Computer Science*, pp. 1–16, Springer-Verlag, 1994.

[16] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography.* CRC Press, 1997.

[17] A. Perrig and J. Tygar, *Secure Broadcast Communication in Wired and Wireless Networks.* Kluwer Academic Publishers, 2003.

[18] F. Stajano and R. Anderson, "The resurrecting duckling: Security issues in ad-hoc wireless networks," in *Proceedings of the 7th International Workshop on Security Protocols* (B. Christianson, B. Crispo, and M. Roe, eds.), vol. 1796 of *Lecture Notes in Computer Science*, Springer-Verlag, 1999.

[19] L. Zhou and Z. Haas, "Securing ad hoc networks," *IEEE Network Magazine Special Issue on Network Security*, vol. 13, no.6, Nov./Dec. 1999.