Proceedings published by International Journal of Computer Applications® (IJCA)
International Conference on Computer Communication and Networks CSI- COMNET-2011

A Hybrid Encryption Technique to Secure Bluetooth Communication

P S Patheja
BIST Bhopal

Akhilesh A. Waoo
BIST Bhopal

Sudhir Nagwanshi
BIST Bhopal

ABSTRACT
A proprietary open wireless technology standard for exchanging data over short distances from one device to another is not much secure. As in traditional method of Bluetooth communication between two or more devices a 128-bit symmetric stream cipher called E0 is used which seems to be week under some conditions it may be broken under certain conditions with the time complexity O (2^{64}). To improve security of data we propose a hybrid encryption technique. In this technique we use triple DES for encryption of the key for which we use Tiger algorithm. In Tiger algorithm there is double protection of Data using triple DES and with the help of this algorithm transmission of data will be more secure for exchanging data over short distances from one device to another.

Keywords
Bluetooth, E0 key stream, hybrid encryption algorithm, data transmission.

1. INTRODUCTION
Bluetooth is a high-speed, low-power microwave wireless link technology, which is based on chip that provide a wireless link to connect phones, laptops and other portable equipment together. Bluetooth is used for a short-range radio frequency (RF) technology that operates at 2.4 GHz and is capable of transmitting voice and data from one device to other. The effective range of Bluetooth devices is 32 feet (10 meters). Bluetooth transfers data at the rate of 1 Mbps, which is from three to eight times to the average speed of parallel and serial transmission. It was invented to get rid of wires[1]. Bluetooth is more suited for connecting two point-to-point devices. Security issues arising from Bluetooth are relatively less publicized, possibly due to less critical nature of information at stake - an individual's cell phone data against corporate data thefts and associated hard losses. Nevertheless, with respect to personal privacy and perimeter security, an insecure Bluetooth device or technology can pose serious risk of information compromise. Now let's explore the categories in which Bluetooth hacking is often classified. This will show how real the issue of security in Bluetooth devices is.

Bluetooth hacks are categorized broadly among:

Blue Jacking: It is the simplest of the four. The hacker uses it by making an attempt to send a phone contact or business card to another nearby phone. The 'name' field of the contact can be misused by replacing it with a suggestive text so that the target device reads it as a part of intimation query displayed on its screen.

Bluesnarfing: It goes a step further and actually accesses or steals data like messages, calendar, phone book etc., from the target device in an unauthorized manner which includes bypassing the usual paring requirement. Here, the problem is bigger since there have been reports of the tools that use methods such as device address guessing and brute force in order to break-in, even when device is configured as 'invisible'.

The next level of sophistication in Bluetooth hacking is Blue bugging where the victim device is controlled by the attacker who sends commands to perform actions as if having physical access to the device this is a functionality analogous to Trojans.

Bluetoothing: Lastly, it is Bluetoothing which typically means social networking in short range, and there is a possibility of harassment from the security point of view. There are programmers for Bluetooth PIN code cracking as well.

The encryption algorithm using in Bluetooth encryption process is the E0 stream cipher. However, this algorithm has some shortcomings, The 128-bit E0 stream ciphers can be cracked with the time complexity O(2^{64}) in some cases. So, for most applications which need top priority to confidentiality, the data security is not enough if we are using the traditional E0 Bluetooth Algorithm.

In this paper we are introducing the Hybrid Encryption algorithm for solving the current security risks in Bluetooth data transmission, but before that we will discuss on basic Bluetooth Mechanism and also about its disadvantages.

2. OVERVIEW
2.1 Security Mechanism in Bluetooth
The Bluetooth defines three security modes:

2.1.1 Safe Mode 1
No safe mode, which has the lowest security level.

2.1.2 Safe Mode 2
Service-oriented security model, which start after the establishment of the channel.

2.1.3 Safe Mode 3
link-oriented security model, which install and initial before communication link is established.

Bluetooth system provides safety precautions in the application layer and link layer, and both sides achieve authentication and encryption in the same way. Link layer uses four entities to ensure the safety:

1. 48-bit of the Bluetooth device address, which is global uniqueness decided by the IEEE;
2. The authentication key for entity authentication is 128-bit;
3. The secret key for data encryption is 8 ~ 128-bit;

4. 128-bit random number trades once, changes once.

Initially two keys are generated and do not open, encryption key is generated during certification process from the authentication key, but it is different from the authentication key, a new secret key is generated every time when we activate the encryption. Authentication key is more stable, after key generation[1]. The random numbers of Bluetooth demands "random generation" and "non-repeatability", that is to say the random numbers are almost impossible to duplicate and can not be significantly greater than zero probability estimate of the random numbers in the authentication key life.

2.2 Authentication and encryption

Bluetooth security-mechanism is divided into three modules key generation, authentication and encryption, and adopts four kinds of algorithms as E0 E1, E2, and E3. Link layer is responsible to provides authentication, encryption and key management. PIN code was entered by user, by means of the E2 algorithm for generating the link key, by means of E3 algorithm, getting encryption key, make use of E0 algorithm generated key stream, and encrypt plaintext, then get cipher text. Figure 1 is the process of Bluetooth encryption.

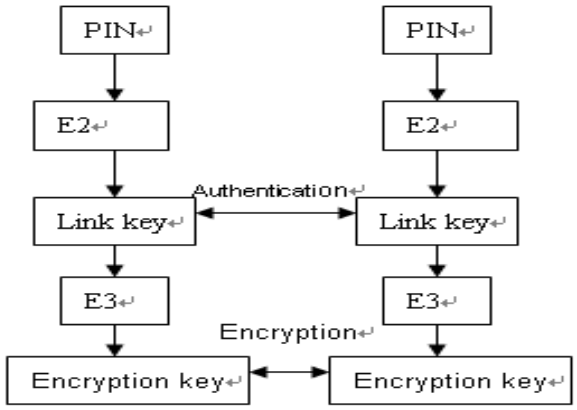

Figure 1 Bluetooth encryption process.

The three modules of Figure 1 are as follows:

1) Key generation module, algorithm E2 is used for generating the link key, and its input parameter is a 4-digit passwords number which is entered by the user, the algorithm E3 calculates encryption key KC by the use of E2 link key encryption key as input parameters.

2) Encryption module, algorithm E0 can be used for generating keys stream to encrypt the original data.

3) Authentication module, algorithm E1 is the crucial algorithms in the authentication process, the two units in need of certification use each authentication algorithm E1 to generate identification word and compare, then complete certification.

2.3 Analysis of E0 Algorithms

E0 belongs to stream encryption method, that is to say it takes data flow and the key bit stream Exclusive-or operation. The payload of each packet is encrypted separately, and the encryption occurs before MPE-FEC, after the cyclic redundancy check. The main principle is to use linear feedback shift register to generate pseudo-random sequence, after that form key stream that can be used for encryption, and then take the key stream and data stream that need encryption Exclusive-or operation, and achieve encryption. During decryption, the cipher text takes Exclusive-or operation once more, re-plaintext can be obtained

2.4 Bluetooth communication

Connection types define the possible ways Bluetooth Devices can exchange data. Bluetooth has three connection types: ACL (Asynchronous Connection-Less), SCO (Synchronous Connection-Oriented) and eSCO. ACL links are for symmetric (maximum of 1306.9 kb/s for both directions) or asymmetric (maximum of 2178.1 kb/s for send and 177.1 kb/s for receive) data transfer. Retransmission of packets is used to ensure integrity of data. SCO links are symmetric (maximum of 64 kb/s for both directions) and they are used for transferring real-time two-way voice. Retransmission of voice packets is not used. Therefore, when the channel BER is high, voice can be distorted. eSCO links are also symmetric (maximum of 864 kb/s for both directions) and they are used for transferring real-time two-way voice. Retransmission of packets is used to ensure the integrity of data (voice). Because retransmission of packets is used, eSCO links can also carry data packets, but they are mainly used for real-time two-way voice. Only Bluetooth 1.2 or 2.0+EDR devices can use eSCO links, but SCO links must also be supported to provide backward-compatibility. Bluetooth devices that communicate with each other form a piconet. The device that initiates a connection is the piconet master. One piconet can have maximum of seven active slave devices and one master device. All communication within a piconet goes through the piconet master. The clock of the piconet master and frequency hopping information are used to synchronize the piconet slaves with the master. Two or more piconets together form a scatternet, which can be used to eliminate Bluetooth range restrictions..

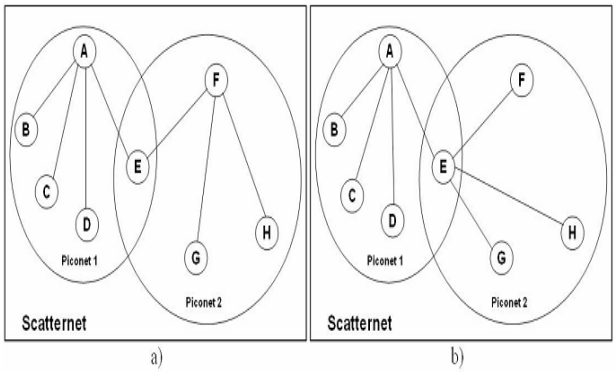

Figure 2 a) Bluetooth topology when ACL links are used. b) Bluetooth topology when SCO

BScatternet environment requires that different piconets must have a common device, so-called scatternet member, to relay data between the piconets. Figure 1 illustrates Bluetooth topology, when ACL or SCO/eSCO links are usedluetooth protocol stack is illustrated in Figure 3. Protocols below HCI (Host Controller Interface) are built-in to the Bluetooth microchip and protocols above HCI are located as a part of the host device's software package. HCI is needed between the hardware and software protocols. The purpose of HCI is to

enable manufacturer-independent combining of Bluetooth chips (Host Controller) and the actual host device. HCI takes care of security communication between the host and Bluetooth module.

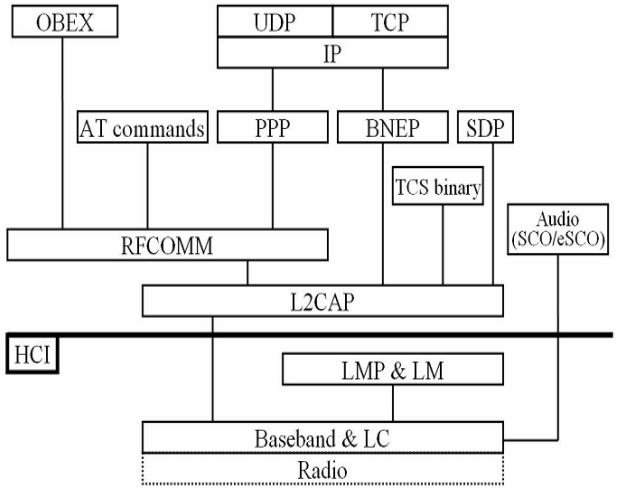

Figure 3 Bluetooth protocol stack

Baseband and LMP (Link Manager Protocol) together enable the physical RF connection. LC (Link Controller) is a state machine that defines the current state of Bluetooth device.

LM (Link Manager) acts as a liaison between the application and the LC on the local device, and it also Communicates with the remote LM via PDUs (Protocol Data Units) using the LMP, i.e. the LM communicates with three different entities during a Bluetooth session: the local host through HCI, the local LC (local operations), and the remote LM (link configuration, link information, and link management operations). The PDU is acknowledged at the Baseband level, but it is acted upon by the LM. The local LM usually resides on the Bluetooth module as a complete host-module implementation. The remote LM can be defined as the LM at the other end of the Bluetooth link. SDP (Service Discovery Protocol) is used to find the services of Bluetooth devices in the range.TCS(Telephony Control protocol Specification) binary defines the call control signaling for the establishment/release of speech and data calls between Bluetooth devices, BNEP (Bluetooth Network Encapsulation Protocol) is used to provide networking capabilities for Bluetooth devices[2].

3. SYSTEM DRAWBACK

3.1 Weakness of E0 stream

If a pseudo-random sequence makes have any error, it will make the whole cipher text wrong, it also possible cipher text cannot recover back as plaintext.

If its output is endless sequence of 0, then the cipher text is the plaintext, so that the whole system is worthless.

if its output is a periodic 16-bit mode, then the algorithm is only an Exclusive-or operation which can ignore security;

If the output is a series of endless random sequence (which is truly random, non-pseudo-random), then there is one-time pad and very perfect safety. The security of actual stream cipher algorithm depends on a simple Exclusive-or operation and the one-time pad[1].

3.2 Believability of PIN

Bluetooth communication uses non-standard 4-digit PIN code and another variable to generate the link key and encryption key. Actually, 4-digit PIN code is the only variable which is the real key generated, resulting only one key (a random number) transport in the air. In the process of creating a link key, intruder intercepts the communication data packet in the first communication process. In order to derive a variety of relevant parameters, including the link key, try brute force attack on the PIN. Thus, the trust of the PIN code is poor; 4 bits PIN code only has 10,000 possibilities. One solution is to choose and use longer PIN code, or use the public key system. This solution increase the difficulty of the attacker, but it is inconvenient, each and every time when a secure connection required, we should have to enter a PIN code.

3.3 Address Spoofing

Every Bluetooth device has a unique Bluetooth device address. Its uniqueness raises new problems. Once the ID links with a fixed device, this device can be tracked and their activities can easily be recorded. In this case, the individual's privacy will be violated. These problems can leads to believe Bluetooth security system is highly unreliable, but there is a fact can not be ignored is that: in general, the data transmitted via Bluetooth connection is not very important. Now, Bluetooth standard is only applicable in smaller networks because considered security technology, if the network nodes are more complex and multiple, the existing key distribution and authentication based on point to point can not meet the demand. Bluetooth technology provides data security measures for small-scale applications, but any sensitive data or the data that may cause problems should not be transferred via Bluetooth directly. In order to uses Bluetooth technology more widely, we can use other more powerful encryption algorithms, such as triple DES and SHA hybrid encryption algorithm.

4. LITERATURE OF TRIPLE DES

A DES key consists of 64 binary digits ("0"s or "1"s) of which 56 bits are randomly generated and used directly by the algorithm. The other 8 bits, which are not used by the algorithm, may be used for error detection. The 8 error detecting bits are set to make the parity of each 8-bit byte of the key odd, i.e., there is an odd number of "1"s in each 8-bit byte0. A TDEA key consists of three DES keys, which is also referred to as a key bundle. Authorized users of encrypted computer data must have the key that was used to encipher the data in order to decrypt it. The encryption algorithms specified in this standard are commonly known among those using the standard. The cryptographic security of the data depends on the security provided for the key used to encipher and decipher the data.

4.1 Triple Data Encryption Algorithm

Let EK(I) and DK(I) represent the DES encryption and decryption of I using DES key K respectively. Each TDEA encryption/decryption operation (as specified in ANSI X9.52) is a compound operation of DES encryption and decryption operations. The following operations are used. A TDEA mode of operation is backward compatible with its single DES counterpart if, with compatible keying options for TDEA operation[2][10].

4.1.1 TDEA encryption operation

The transformation of a 64-bit block I into a 64-bit block O that is defined as follows:

$$O = EK3(DK2(EK1(I))).$$

An encrypted plaintext computed using a single DES mode of operation can be decrypted correctly by a corresponding TDEA mode of operation

TDEA decryption operation

The transformation of a 64-bit block I into a 64-bit block O that is defined as follows:

$$O = DK1(EK2(DK3(I)))$$

An encrypted plaintext computed using a TDEA mode of operation can be decrypted correctly by a corresponding single DES mode of operation. When using Keying Option 3 (K1 = K2 = K3)[2].

4.2 Keying options

The standard specifies the following keying options for bundle (K1, K2, K3)

4.2.1 Keying Option 1

K1, K2 and K3 are independent keys;

4.2.2 Keying Option 2

K1 and K2 are independent keys and K3 = K1;

4.2.3 Keying Option 3

K1 = K2 = K3.

Keying option 1 is the strongest, with 3 × 56 = 168 independent key bits .Keying option 2 provides less security, with 2 × 56 = 112 key bits. This option is stronger than simply DES encrypting twice, e.g. with K_1 and K_2, because it protects against meet-in-the-middle attacks. Keying option 3 is equivalent to DES, with only 56 key bits. This option provides backward compatibility with DES, because the first and second DES operations cancel out. It is no longer recommended by the National Institute of Standards and Technology (NIST), and is not supported by ISO/IEC 18033-3.

TDEA Encryption Operation:

$$I \rightarrow \boxed{DES\ E_{K1}} \rightarrow \boxed{DES\ D_{K2}} \rightarrow \boxed{DES\ E_{K3}} \rightarrow O$$

TDEA Decryption Operation:

$$I \rightarrow \boxed{DES\ D_{K3}} \rightarrow \boxed{DES\ E_{K2}} \rightarrow \boxed{DES\ D_{K1}} \rightarrow O$$

4.3 Encryption of more than one block

As with all block ciphers, encryption and decryption of multiple blocks of data may be performed using a variety of modes of operation, which can generally be defined independently of the block cipher algorithm. However ANS X9.52 specifies directly, and NIST SP 800-67 specifies via SP 800-38A, that some modes shall only be used with certain constraints on them that do not necessarily apply to general specifications of those modes. For example, ANS X9.52 specifies that for cipher block chaining, the initialization vector shall be different each time, whereas ISO/IEC 10116 does not. FIPS PUB 46-3 and ISO/IEC 18033-3 define only the single block algorithm, and do not place any restrictions on the modes of operation for multiple blocks[9].

4.4 Security

In general Triple DES with three independent keys (keying option 1) has a key length of 168 bits (three 56-bit DES keys), but due to the meet-in-the-middle attack the effective security it provides is only 112 bits. Keying option 2 reduces the key size to 112 bits. However, this option is susceptible to certain chosen-plaintext or known-plaintext attacks and thus it is designated by NIST to have only 80 bits of security. The best attack known on keying option 1 requires around 2^{32} known plaintexts, 2^{113} steps, 2^{90} single DES encryptions. This is not currently practical. If the attacker seeks to discover any one of many cryptographic keys, there is a memory-efficient attack which will discover one of 2^{28} keys, given a handful of chosen plaintexts per key and around 2^{84} encryption operations.

5. LITERATURE OF TIGER

5.1 Introduction

Tiger is a cryptographic hash function designed by Ross Anderson and Eli Biham in 1995 for efficiency on 64-bit platforms. The size of a Tiger hash value is 192 bits. Truncated versions (known as Tiger/128 and Tiger/160) can be used for compatibility with protocols assuming a particular hash size. Unlike the SHA-2 family, no distinguishing initialization values are defined; they are simply prefixes of the full Tiger/192 hash value.

Tiger2 is a variant where the message is padded by first appending a byte with the hexadecimal value of 0x80 as in MD4, MD5 and SHA, rather than with the hexadecimal value of 0x01 as in the case of Tiger. The two variants are otherwise identical. Tiger is designed using the nearly universal Merkle-Damgård paradigm. The one-way compression function operates on 64-bit words, maintaining 3 words of state and processing 8 words of data. There are 24 rounds, using a combination of operation mixing with XOR and addition/subtraction, rotates, and S-box lookups, and a fairly intricate key scheduling algorithm for deriving 24 round keys from the 8 input words. Although fast in software, Tiger's large S-boxes (4 S-boxes, each with 256 64-bit entries totals 8 KiB) make implementations in hardware or small microcontrollers difficult[14].

5.2 Cryptanalysis

Unlike MD5 or SHA-0/1, there are no known attacks on the full 24-round Tiger except for pseudo-near collision. While MD5 processes its state with 64 simple 32-bit operations per 512-bit block and SHA-1 with 80, Tiger updates its state with a total of 144 such operations per 512-bit block, additionally strengthened by large S-box look-ups. John Kelsey and Stefan Lucks have found a collision-finding attack on 16-round Tiger with a time complexity equivalent to about 2^{44} compression function invocations and another attack that finds pseudo-near collisions in 20-round Tiger with work less than that of 2^{48} compression function invocations. Florian Mendel et al. have improved upon these attacks by describing a collision attack spanning 19 rounds of Tiger, and a 22-round pseudo-near-collision attack. These attacks require a work effort equivalent to about 2^{62} and 2^{44} evaluations of the Tiger compression function, respectively.

6. THE IDEAS AND PROCESSES OF HYBRID ENCRYPTION ALGORITHM

6.1 Process of encryption

During the process of sending encrypted information, the random number generator uses 64-bit triple DES session key only once, it encrypt the plaintext to produce cipher text. On the other hand, the sender get debit's public key from public key management center, and then using Tiger to encrypt session key. Finally, the combination of the session key from Tiger encryption and the cipher text from triple DES encryption are sent out.

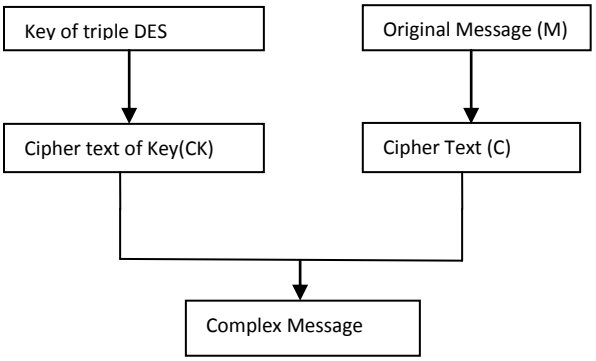

Figure 4 shows the encryption of text

6.2 Process of decryption

The decryption of hybrid encryption algorithm is as follows. The first, the receiver divide received cipher text into two parts, one is cipher text from the Tiger algorithm encryption, the other is cipher text from the triple DES algorithm encryption. The second, the receiver decrypt cipher text by their own private key, receive the key which belongs triple DES algorithm, then decrypt the cipher text to the original text by key. Figure is a decryption of hybrid encryption algorithm.

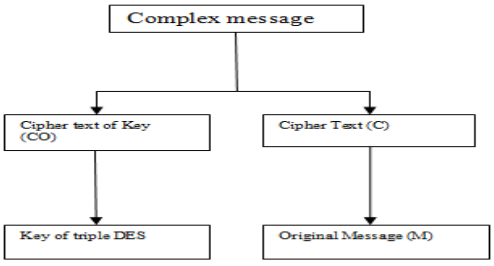

Figure 5 show the decryption process

7. CONCLUSION

Bluetooth technology is a new technology for our transmission method. For communication between devices it uses wireless mode for the transmission of data from one device to another. Compared to the fixed line network Bluetooth network is more vulnerable to be attacked. For those applications that take data security as priori, achieving a high level of data security is essential. Currently, stream cipher E0 used in Bluetooth standard has many shortcomings, while the hybrid encryption algorithm is relatively more secure and easier to achieve, thus it ensures that data transmission between the Bluetooth device are more safe than traditional E0 Bluetooth Algorithm.

8. ACKNOWLEDGMENT

I express my sincere gratitude and acknowledgement towards P S Patheja sir and Akhilesh A Waoo sir, who guided me. It was their constant support and inspiration without which my effort would not have taken this shape. I sincerely thank them for this and seek them support for all my future endeavors

9. REFERENCES

[1] A Hybrid Encryption Algorithm Based on DES and RSA in Bluetooth Communication, Wuling Ren, Zhiqian Miao, College of Computer and Information Engineering Zhejiang Gongshang University

[2] Performance Analysis of SAFER+ and Triple DES security algorithms for Bluetooth Security Systems , Dr.R.Neelaveni,D.Sharmila

[3] Bluetooth Hacking: A Case Study, Dennis Browning, Gary C. Kessler

[4] Zheng Hu. Network and Information Security [M]. Peking: Tsinghua University Pres, 2006.

[5] Man Young Rhee. Network Security Encryption Principle, algorithm and Protocol[M]. Peking: Tsinghua University Pres, 2007.

[6] Suri, P. R.; Rani, S. Bluetooth security Need to increase the efficiency in pairing [J]. IEEE/ Southeastcon, 2008.

[7] Fengying Wang. Dynamic Key 3DES Algorithm of Discrete System Based on Multi-dimension Chaos [J]. Microelectronics and Computer, 2005, 7: 25-28.

[8] Falk A. The IETF, the IRTF and the networking research Community.Computer Communication Review, v35, n5, Oct. 2005:6970.

[9] Yaniv Shaked, Avishai Wool. Cracking t he Bluetooth P[C]. 3rd USEN IX/ ACM Conf. Mobile Systems, Application and Services (MobiSys). Seattle, WA, June 2005:39250.

[10] Data Encryption using DES/Triple-DES Functionality in Spartan-II FPGAs, Amit Dhir

[11] Cryptanalysis of Bluetooth Keystream Generator Two-level E0, Yi Lu? and Serge Vaudenay

[12] On the Existence of low-degree Equations for Algebraic Attacks, Frederik Armknecht?

[13] Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher Revised 19 May 2008 William C. Barker

[14] Tiger: A Fast New Hash Function, Ross Anderson, Eli Biham

[15] Serpent: A New Block Cipher Proposal, Eli Biham, Ross Anderson and Lars Knudsen

[16] Cracking the Bluetooth PIN, Yaniv Shaked and Avishai Wool.

[17]