IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4671

Wireless Cryptographic Systems

Amanda Dambrouckas
Bay Path College/Rensselaer University/Stanpak Systems
53 Lakeview Hgts., Tolland, CT 06084
(870) 871-6547, amanda@stanpak.com

The science of cryptology has been the center of research projects for decades. As we know, the explosion of sophisticated mobile devices is a much more recent event. Now, we are facing the task of fusing these two technologies to implement secure mobile communications. As we will see, there are many constraints to implementing secure encryption for wireless devices. While many of these issues lie in the area of hardware capabilities, we must consider the public transmission medium used in these networks. Despite these obstacles, I do believe that we can have working, secure encryption for mobile devices in the near future. I think that this task can be solved via communal development and testing, new calculation methods, processing/storage efficiency, and forethought.

FORETHOUGHT

Corporations tend to get overly anxious when preparing to release a new product. This often leads to situations in which security is left to an afterthought (or even worse, seen in hindsight). This is especially common in the tech industry where it is essential to be the first, and the newest innovator. Due to the types of information that we are using wireless technology to transmit/store (e-cash, stock quotes, mail, conversation, account access), we must pay special attention to information security.

A developer and consultant for the US Department of Justice described the Internet surge in late 90's as having only security as an afterthought. She explained that security should be "woven into entire lifecycle." She pointed out that right now, wireless transmissions are not as fast as wired communication. This causes wireless networks to be less attractive targets for hackers. If we wish to enjoy improved mobile technologies, we will have to concentrate on security in order to counteract attacks.[1]

It may seem ironic that encryption algorithms are publicly available. However, not only is the publication of encryption algorithms independent from security, it drives extensive testing and validation of a cryptographic process. The more access cryptanalysts have to these algorithms, the more they can attempt to surpass them. As we have learned from open source projects, more minds yield superior accomplishments. Any person who requires validation of this fact will find proof in the recent advent of AES. The Advanced Encryption Standard was made possible via the efforts of many researchers who shared information and worked collectively to produce the technology.

What does matter in terms of security of cryptographic processes is the key. If the key is private, and it is impossible to fabricate key generation, the algorithm can be publicly available. In fact, experienced cryptographers make the assumption that cryptanalysts will have access to algorithms when designing encryption processes. My theory is that we can improve wireless security practices by concentrating on encryption keys rather than the overall crypto processing. We already have well known blueprints for encryption in wired environments. Now, we can modify these existing methods by replacing tools within the process and customizing for the wireless arena.

Cryptography is a science that offers us humans two main benefits. The first is the ability to conceal the meaning of messages. Second, cryptography allows us to authenticate parties. Wired encryption can only give us message security during storage and transmission. We will see that there are ways in which we can implement ad-hoc network encryption during processing as well.

THE WIRELESS ARENA

When we consider the technological implications of wireless connectivity, we must realize that there are effects on encryption. The process of signal interpretation processing and communications can be encrypted in addition to the usual states of storage and communication. As there are always drawbacks associated with capabilities, the open-air transmission achieved in wireless environments creates more opportunity for interception than we see in wired networks. These added points of possible attack increase the opportunities for malicious eavesdroppers to gain knowledge of the key, and/or message.

WLANS

In order to truly understand the complications of security, and thus the cryptography implemented in wireless settings, we should understand how the environmental architecture differs from that seen in wired networks. A standard (IEEE802.11) WLAN contains a set of radio transceivers, various clients (PCs, workstations, printers, PDAs), access points and other equipment. The base stations (access points) act similarly to the wired bridge, connecting network segments. One of these segments may be a wired LAN, via which the WLAN obtains Internet access.

What makes wireless transmissions vulnerable to security breaches is the medium of transmission. Rather than constraining packets within a protected wire, mobile devices transport packets over untrusted/public mediums. These may include radio frequencies. This leaves the possibility for eavesdroppers to listen to transmissions, simply because the communications take place over a public medium. This point exemplifies the importance of cryptography in wireless settings. We must have some way to conceal data when the transmission channel provides no security. Theoretically, we should feel confident transmitting confidential information over insecure mediums if we know that the information is unintelligible.

WIRELESS ENCRYPTION EQUIPMENT

We can approach the task of implementing strong wireless cryptography with a specialized toolset. Our tools work with keys, rather than with the cryptographic process. We can replace symmetric and asymmetric tools that are in place inside wireless networks. In this toolset, we can include ECC where RSA is currently in place. We can use Rjindael in place of DES and WTLS in place of SSL.

Tool Table (Example)

Old Tool	*New Tool*
MD5	*SHA1*
RSA	*ECC*
DES and RC4	*RJINDAEL*
SSL	*WTLS*

REAL ENCRYPTION

There are two primary variants of cipher text generation. These entail symmetrical (single shared key) and asymmetrical (two different yet related keys) encryption. We can refer to an encryption algorithm as being "strong" when the possible number of keys makes a brute force attack unreasonable. We must note the "unreasonable" element in this determination is derived from available processing power, not from human capabilities. Even the long standard DES algorithm has seen the last of its days as a "strong" algorithm. This is because of the fact that the 70 quadrillion possible keys (56bit) are no match for current (and future) processing power.[2]

NEW TOOLS

1. (Symmetric tool) Rjindael is a symmetrical cryptographic technique that gives implementers of the algorithm a great deal of flexibility. Rijndael has been implemented in various dedicated hardware environments, and in smart cards.[3]

2. (Symmetric tool) TDEA (Triple Data Encryption Algorithm) key consists of two or three distinct DES keys. The initial plaintext message is encrypted three times using these keys and the standard DEA/DES logic. This is more secure than its DES ancestor.
3. (Asymmetric tool) Elliptic Curve Cryptosystem (ECC) addresses the issues of processing, storage, and bandwidth limitations faced in wireless devices. This is important because wireless devices tend to provide relatively low memory, processing and storage capabilities. This makes ECC a hopeful method in which to implement secure cryptography (signatures or entire messages) in smart cards, PDAs, ATM machines, remote access systems, electronic cash, and cellular phones.[4]
4. (SHA1). The SHA algorithm produces a fixed length digest, regardless of the plain text message that is input into the function. The length of input is always equal to 2^64 bits, and the resulting digest is 160 bits in length. The message digest is unique for the initial message; that is, no two messages will produce identical hash values. The "Secure" in Secure Hashing Algorithms is due in part to the fact that the logic is designed to prohibit the generation of plain text messages from the message digest.

WIRELESS SOLUTIONS: FOCAL POINTS

I believe that the keys (no pun intended) to successful and practical wireless encryption methods fall into two main categories. Specifically, realistic wireless encryption must be computationally efficient, and require minimal storage capacity. Techniques including hashing algorithms such as SHA-1 and mathematical methods can be of aid to wireless security engineers. These topics address two primary complications of wireless communications, namely processing/bandwidth capabilities and storage capacity.

1. *Mathematics*

We can overcome many of the limitations imposed by wireless device capabilities through the implementation of mathematical routines that best fit the wireless environment.

ECC offers us an alternative method for public key, and for digital signature generation. ECC systems implement a variant of DLP (Discrete Logarithm Problem) using groups of points on elliptical curves. These points are used in formulas to determine the private/public key relationship.

ECC is able to provide a high level of security to these devices without creating unreasonable computational overhead.

This is vital for resource restrained wireless hardware.

A study of ECC efficiency concluded that RSA and DSA would need a 1024 bit key while ECC needed only a 160bit key to perform strong encryption. Furthermore, we can significantly increase the security provided using ECC in small key size increments.

Key Size Equivalency Table

RSA	*ECC*
1024	*160*
2048	*210*[5]

Implementation

a) 3com's palm computing division (palm VII)
b) Motions's Blackberry pager[6]

All of these facts lead one expert to say: "You will start to see people adopt ECC as an option, after it's a pretty common option, it will start to become the default."[7]

2. *Hashing Algorithms*

Hashing algorithms reduce the size of the data that must be transmitted in enciphering routines. This reduces the burden on bandwidth. Message digests produced via hashing reduce the computation time/resources required for mobile devices to perform private key signing and public key authentication. This becomes especially important when we consider public key encryption, which is inherently slower than private key encryption (yet ever so common in e-commerce).

The message summary is a compressed "image" of the original message. It is far faster to perform encryption operations on these summaries than it is to encrypt entire messages. From a security standpoint, hashing algorithms such as SHA-1 are very attractive, due to the fact that they operate in a uniform direction.

Implementation

SHA-1 has already found a niche in the arena of wireless encryption services. HORNET is an efficient stream cipher that implements the SHA-1 algorithm in key generation and has been implemented in ASICs for wireless phones.

HORNET family routines have been implemented to form impressive security measures when teamed with ECC mathematical routines and the Rjindael encryption algorithm.

There are no documented cases of breaches in SHA-1 security. The two most likely attack methods have both failed to break SHA digests. [8]

OLD TOOLS

1. *(Symmetric tool) DES*

DES takes 64 bits of plain text and transforms them into a different 64 bits. The transformed bits comprise the cipher text. First, the Initial Permutation (IP) is applied to the 64bits of plain text to be encrypted. The initial 64 bits of plain text are simply rearranged.

The Problem

Today, it is far too easy for diligent hackers to break 56 bit keys. We simply cannot rely on this level of security for anything more than trivial transmissions. This is where the ECC alternative shines.

2. *WEP*

The Wired Equivalent Protocol WEP (is a security standard used in WLAN environments. WEP implements symmetric keys and is configurable on an 802.11 network.

The Problem

The secret key for WEP implementations ranges from 40-128 bits, and is generally physically typed into the hardware device. This stored key is retained for continued use. Due to the fact that each mobile device associated with a WEP WLAN uses the same key, if one device is compromised then all devices must alter their keys. [9]

CASES FOR IMPROVEMENT

We can examine many cases in which we can improve wireless security. Most of the weak points in these scenarios are obvious or explained below. We can look to our tool table and replace the 'old tools' with the 'new tools' for our new wireless environments.

1. *Bluetooth*

Bluetooth is an inexpensive, short-range radio link for mobile devices within WANs. Cryptography in Bluetooth communications is performed symmetrically with SAFER+ (block cipher) and E0 (stream cipher). *The Problem*

SAFER+ keys cannot exceed 128bits, which has been proven insecure in WEP. The E0 cipher also users a 128bit (max) key, and is quite vulnerable to divide and conquer attacks. "Bluetooth is simply scrambled, spreading its data over numerous different spectrums, rather than encrypted in complex algorithms."[10] This leaves privacy/integrity to a game rather than a dependable method of information security.

Experts have identified four main areas of concern in Bluetooth technology:[11]

Device address discovery — authenticate device, not user.
Key management — transmission in plain text
Pin code attacks — could fix with public key crypto (ECC)
No user authentication — could fix with robust symmetric ciphers Rjindael

2. *VPNs*

The IPSEC Protocol is an addition to the standard IP protocol, intended to provide security and privacy for TCP/IP sessions. IPSEC is the most widely used protocol for VPNs. Virtual Private Networks are implemented today in many wireless (and wired) networks in order to fill in for the loopholes of WEP. [12]

Current	***Replacement***
1. Key exchange is handled via RSA or D-H	*1. ECC*
2. Session protection is delegated to DES or triple DES (3DES)	*2. Rijandael*
3. HMAC-MD5 algorithms provide data integrity	*3. SHA-1*

3. *WLANS*

The core of WEP logic lies in RC4, a stream cipher symmetric key algorithm. Developers chose this algorithm due to the fact that it was inexpensive to license, and relatively easy to implement. Today, RC4 is not the state of the art encryption algorithm that it once was, but it is still regarded as reasonably secure. RC4 uses a variable length key (1-256 bytes). A psuedo random stream is generated and XORed with the plain message text. Due to export restrictions, the RC4 key is often limited to 40 bits. However, the algorithm can utilize keys from any length between 1 and 2048 bits.

The Problem

Berkley researchers discovered that WEP could be compromised via passive attack strategies. The researchers discovered that a 128-bit WEP key could be recovered using such an attack. This is quite indicative of the amount of processing power that is available to the public today. A 128-bit key generated with high entropy can pose a possible 3.4028236692e38 keys for crackers.

One of the major reasons that the Wired Equivalent Privacy technique faulted was the use of RC4 for multiple WLAN functions. Specifically, the RC4 stream cipher was used for both the authentication and privacy functionality. RC4 warns never to use identical key material repeatedly, because it is a simple XOR stream cipher. [13]

Current	***Replacement***
RC4 (authentication)	*Rijandael/RC5*
RC4 (privacy)	*Rijandael/RC5*

CONCLUSION

We have seen that there are solutions available to implement wireless security. I think that we will see a pattern of public attitude modification, in the area of confidence in ad-hoc technologies. Just as the number of consumers willing to participate in wired e-commerce has grown, I believe that the number of wireless financial transactions will increase. Hopefully, efficient encryption designs will aid in this process. We have discussed several techniques that show great potential in reaching this goal. In my analysis, these methods tend to fall into the computational and hashing/compression paths. In any case, I think that we can indeed have truly secure ad-hoc networks in the near future. The outcome of this project will depend on communal development/testing, and in exploring new tools and techniques.

ENDNOTES

[1] Gaudin Sharon. “Wireless security lesson to learn.” Bluetooth IT management. www.earthweb.com/secu/article.php 10/22/02

[2] US Department of Commerce and The National Institute of Standards and Technology. Data Encryption Standard (DES).

[3] Daemen, Rijmen. Proton World & COSIC. “Rijndael: Vincent meets Joan.” ProtonWorld, NISSC 2000. October 23, 2000.

[4] Certicom. The Elliptic Curve Cryptosystem: Current Public-Key Cryptographic Systems. July 2000.

[5] Ntru Press Room. Fastest, Smallest Security Toolkit for Palm. 3G Strategies for operators. Issue 5.

[6] Lee, Tom. The Industrial Physicist. The American Institute of Physics. August 2000.

[7] IBM Developerworks. What’s what in wireless surveying the wireless landscape victor marks software engineer. IBM May 2001.

[8] IBM schedules TeleHubLInk’s wireless encryption microchip for manufacturing. www.semiconductorfabtech.com/sit-global/news

[9] Nichols, Lekkas. Wireless Security Models, Threats, and Solutions. McGraw Hill TELELCOM. 2002. 226-241.

[10] Nichols, Lekkas. Wireless Security Models, Threats, and Solutions. McGraw Hill TELELCOM. 2002. 415.

[11] Nichols, Lekkas. Wireless Security Models, Threats, and Solutions. McGraw Hill TELELCOM. 2002. 415.

[12] Cisco Technologies Documentation. IPSec Network Security. www.cisco.com/univercd/cc/td/doc

[13] Nichols, Lekkas. Wireless Security Models, Threats, and Solutions. McGraw Hill TELELCOM. 2002. 415.