



An Introduction to Cryptographic Security Methods and Their Role in Securing Low Resource Computing Devices

*An Overview of Public-key Cryptosystems based on RSA,
Diffie-Hellman and the Next Generation of Public-key
Cryptographic Security for Low-Resource Computing Devices -
the Algebraic Eraser™*

May 2010

SecureRF Corporation
175 Post Road West
Westport, CT 06880

203-227-3151
info@SecureRF.com
www.SecureRF.com

1. Introduction

The need for secure communications dates back to antiquity. Whether we are looking at the 5000 year old cylinder seals of ancient Mesopotamia, which were used for authentication, or the Caesar cipher, which was used to protect military messages, a theme quickly emerges: information needs to be secured.

There are two distinct paradigms for approaching security. One is the symmetric or private-key path, which can be viewed as a descendant of the very early attempts at security. Any such system assumes that, if two individuals wish to communicate securely, they must both possess a unique common secret key, which is used for both encryption and decryption. While many such systems have evolved over time, it is this very assumption, and the difficulties of sharing and distributing the common key and keeping it a secret, that leads to their vulnerability and the need for a new paradigm. By the early 1970's a significantly new and different perspective on security began to take shape contributing to the emergence of public-key cryptography.

With public-key cryptography, each user has a specific private-key, which only the user knows, and a mathematically-related public-key which can be made public and freely distributed. These public-key systems fall into two categories: the Diffie-Hellman protocol published by Whitfield Diffie and Martin Hellman in 1976, and the RSA protocol, as publicly described by Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. Before focusing on the structure of these distinct approaches, it is worth taking a moment to consider the broader impact public-key methods have had.

As ubiquitous as the Internet is today, it was not until the 1990's that the necessary public-key security protocols were put in place that enabled users to function securely while online. For example, the online revolution in the financial sector, e-commerce, and medical records, all rely on the security breakthrough provided by public-key cryptography. In many respects we are at a similar evolutionary point with embedded systems, wireless sensor networks, and radio frequency identification (RFID) technologies. The solutions that enabled the Internet revolution are not suitable to these low-resource environments; a new generation of public-key infrastructure needs to emerge. The basic theme, which dates back to early civilization, persists: information needs to be secured.

2. A Heuristic View of Cryptographic Structures

When viewed diagrammatically at a high level, private-key methods of encryption all take the following form. Two users, Alice and Bob, each possess the same secret key, κ , which is used for both encryption and decryption (hence the term *symmetric*). When Alice wishes to communicate securely with Bob she inputs her message (often termed the plaintext), along with her key κ , into her encryption protocol, which then outputs the message in an encrypted form, referred to as the ciphertext, as seen in Figure 1.

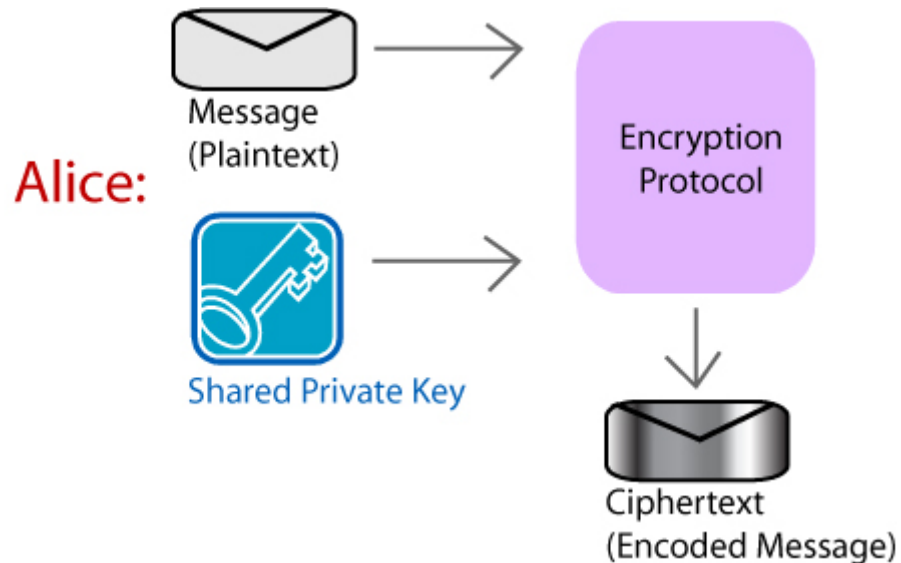


Figure 1: Alice creates an encoded message with a private-key she shares with Bob.

Bob, upon receiving the ciphertext from Alice, can obtain her original message by inputting their shared key κ , along with ciphertext into his decryption protocol, as shown in Figure 2 on the next page.

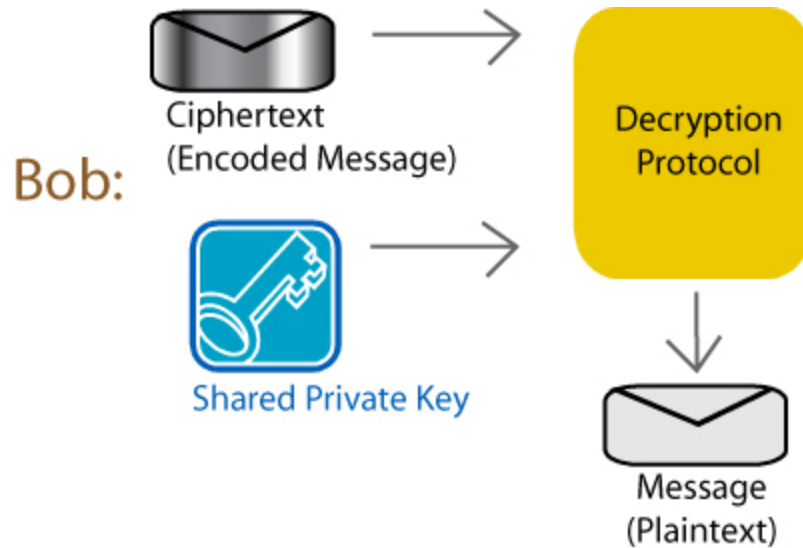


Figure 2: Bob converts encoded message from Alice into plaintext using shared private-key.

As intuitive as such a private-key method is, the vulnerabilities quickly emerge. The security of the system relies on the secret key κ remaining uncompromised at all times. In that there are many cryptographic methods to search for the secret key, a key can only remain secure for so long. With the use of a private-key solution, if one user's private -is compromised, then all users who share that key will be impacted.

As the number of users of a private-key system grows (in the case of RFID - into the billions!), the need to confidentially distribute new keys on a regular basis leads to the progressively unwieldy problem of key management. While it is generally true that private-key systems have a fast running time, which perpetuates their use in many commercial systems today, their intrinsic weaknesses are well known in the security community.

Public-key (or asymmetric) cryptography frees itself from most of the above-discussed difficulties by assigning each user a specific private-key, which only the user knows, and a mathematically-related public-key which can be made public and freely distributed. There are two general methods for using public-key cryptography, RSA and Diffie-Hellman. In both cases, the systems are designed so that the public-keys do not have to be distributed to all parties prior to communicating and the security of the system is not compromised by the publication of the public-keys when a session begins.

In the case of an RSA-type system, if Alice wants to send Bob a message, she uses Bob's public-key to encrypt her plaintext, as seen in Figure 3.

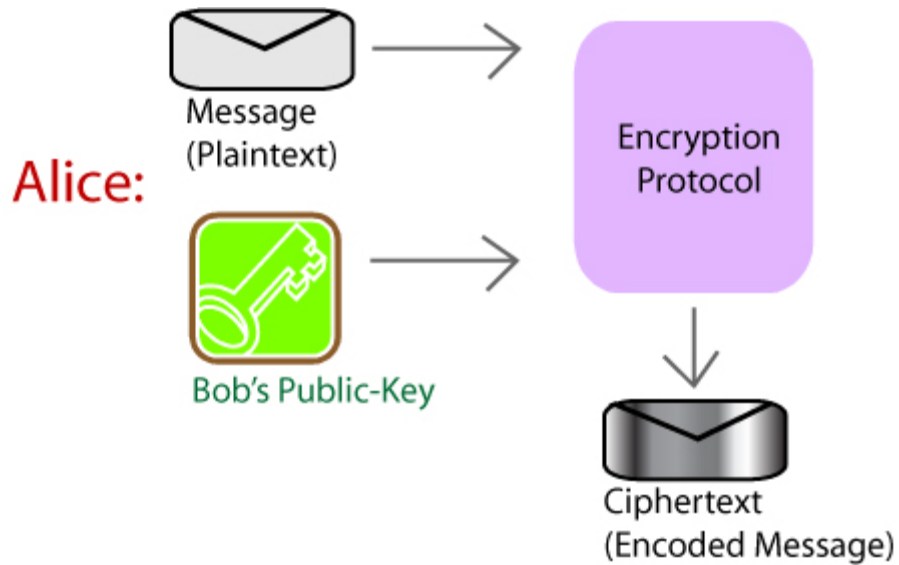


Figure 3: RSA type system: Alice uses Bob's Public-Key to encrypt her message to Bob.

Bob, upon receiving the ciphertext, can obtain Alice's original message by inputting his private-key κ , along with ciphertext into his decryption protocol, as seen in Figure 4.

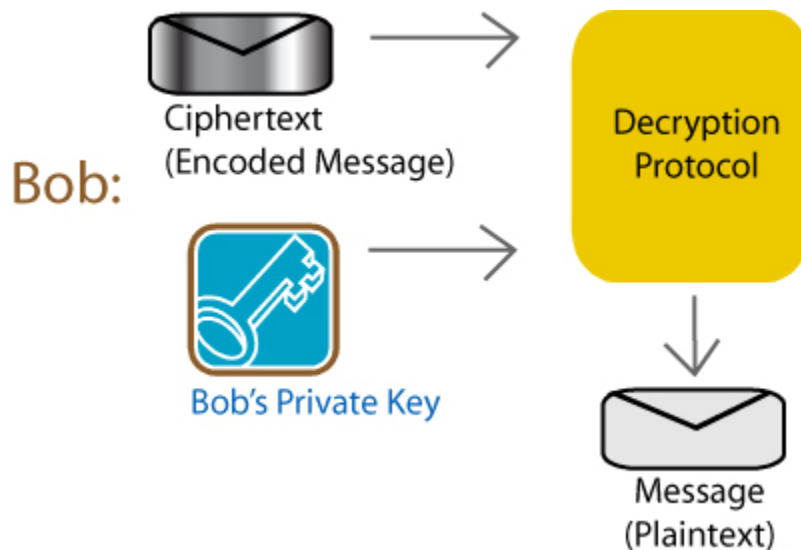


Figure 4: RSA type system: Bob uses his own private-key to decrypt the encoded message.

With an RSA-type system, only Bob has a copy of his private-key so anyone intercepting the encrypted message from Alice will be unable to decipher it.

In the case of a Diffie-Hellman-type system Alice and Bob use each other's public-keys, together with their own respective private-keys, to establish a common secret key, which can be

used for encryption and decryption of their message. Only Alice and Bob, who are participating in this session, will have access to the necessary key to encrypt or decrypt a message.

The Diffie-Hellman type systems are structurally distinct as can be seen in Figure 5.

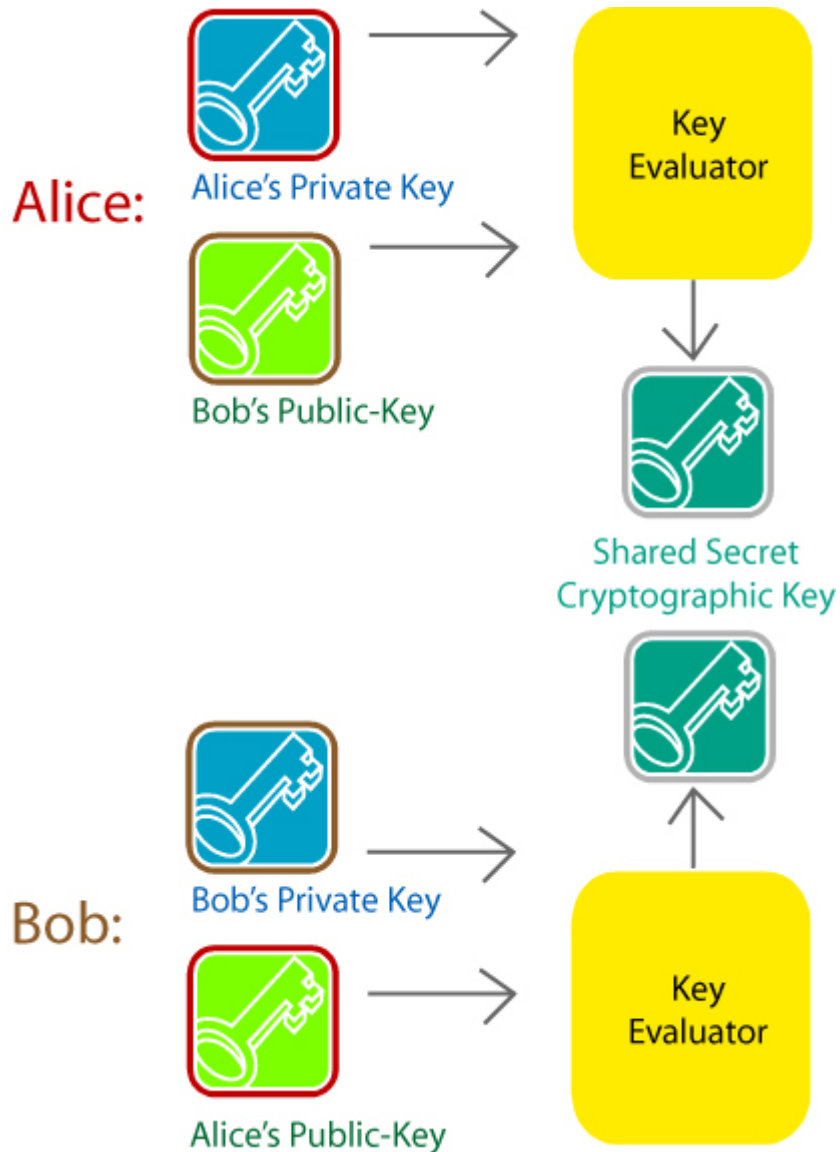


Figure 5: Diffie-Hellman type system

The output of the Diffie-Hellman type system is a shared secret cryptographic key, the value of which is the same for both Alice and Bob. This shared key can then be used for encryption and decryption of the message passed between them, similar to a private key system. In comparison, the output of an RSA type system is the encrypted text itself. In both of these asymmetric systems, each user's private-key is unique to the user.

A succinct view of various public-key and private-key systems, including the Algebraic Eraser™ which will be discussed in the next section, is in the concept tree shown in Figure 6.

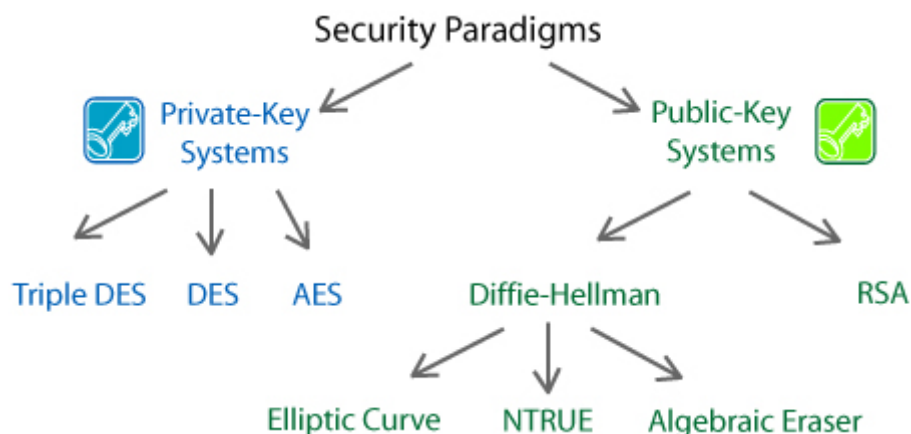


Figure 6: Categorization of private-key and public-key security paradigms

In a private-key situation the security of the system relies on the secret key κ remaining uncompromised at all times. In contrast, with a public-key system, if any one user's private-key becomes known the other users are insulated from having their communications compromised. This is because the private-key is not common across the entire system as found in a symmetric system, but rather the private-key is unique to each individual. In a public-key scenario the mathematical relationship between a user's private and public key is designed so that it is infeasible to derive the private-key from the public-key (such a relationship is referred to as a one-way function). To be able to effectively reverse engineer a public-key would be as difficult as a mathematically intractable problem. That is, the value of breaking the public-key is similar to a mathematical problem that may be solvable but is so difficult or timely to complete that the solution will provide no benefit once derived. The security of these systems lies in this intractability.

Public-key systems do have issues to overcome. Ensuring that the distributed public-keys are authentic is essential to security of the system and this issue is known as the "man-in-the-middle attack." Protocols to prevent malicious use of public data also need to be put in place in order to prevent "replay attacks." While technically challenging, these obstacles can be mitigated.

A more profound issue with today's commonly implemented RSA and Diffie-Hellman type public-key protocols, including NTRU and Elliptic Curve Cryptography (ECC), concerns the computational footprint they entail. While memory and energy usage is not a primary concern for most environments requiring cryptographic security, these issues, along with runtime, lie at the heart of any small computing device security discussion. Every one of these cryptographic systems at its core utilizes multiplication of large numbers. As a result the computing resources required to achieve security grow rapidly as the level of security is increased.

As the market for smaller and more mobile/wireless computing devices continues to grow, so does the need for stronger and more efficient security solutions to address these markets. As a result, the need for a next generation of public-key cryptography becomes evident.

3. The SecureRF Approach

With the goal of facilitating the security of low-resource computing platforms including RFID, microcontrollers, wireless sensor networks and integrated circuits, SecureRF has introduced the Algebraic Eraser™ public-key cryptosystem whose characteristics are specifically suited to these low-resource environments.

The Algebraic Eraser public-key cryptosystem has as its foundation in three distinct areas of mathematics and on which its security is based: the theory of braids, the theory of matrices with polynomial entries (expressions of finite length constructed from variables), and modular arithmetic. The quality that makes braid theory desirable for cryptography is akin to the fact that a fishing line can get entangled in a few seconds while it may take hours to untangle. With mathematical braids, calculating in one direction is easy, but reversing it is extremely difficult. At the Algebraic Eraser's core is a highly specialized function (replacing the standard system's operations), known as E-Multiplication™, which brings together these mathematical tools and enables the system to provide high-speed security without overwhelming the memory and power available. This core function is highly resistant to reverse engineering due to connections to mathematically intractable problems.

In January 2010, the United States Patent and Trademark Office granted SecureRF Corporation U.S. Patent 7,649,999, entitled "Method and apparatus for establishing a key agreement protocol," for its technology invention in the field of cryptography. The reader may learn more about the company's methods from reviewing this patent. The method is summarized in its abstract stating:

"A system and method for generating a secret key to facilitate secure communications between users. Public keys are exchanged between first and second users. Each user's private key may be iteratively multiplied by the other user's public key to produce a secret key. Secure communication may then occur between the first and second user using the secret key."

Structurally, SecureRF's Algebraic Eraser public-key cryptosystem is of the Diffie-Hellman type. Alice and Bob each use their own private-key and the other person's public-key to generate a shared secret key, via the Algebraic Eraser E-multiplication function, as can be seen in Figure 7 on the following page.

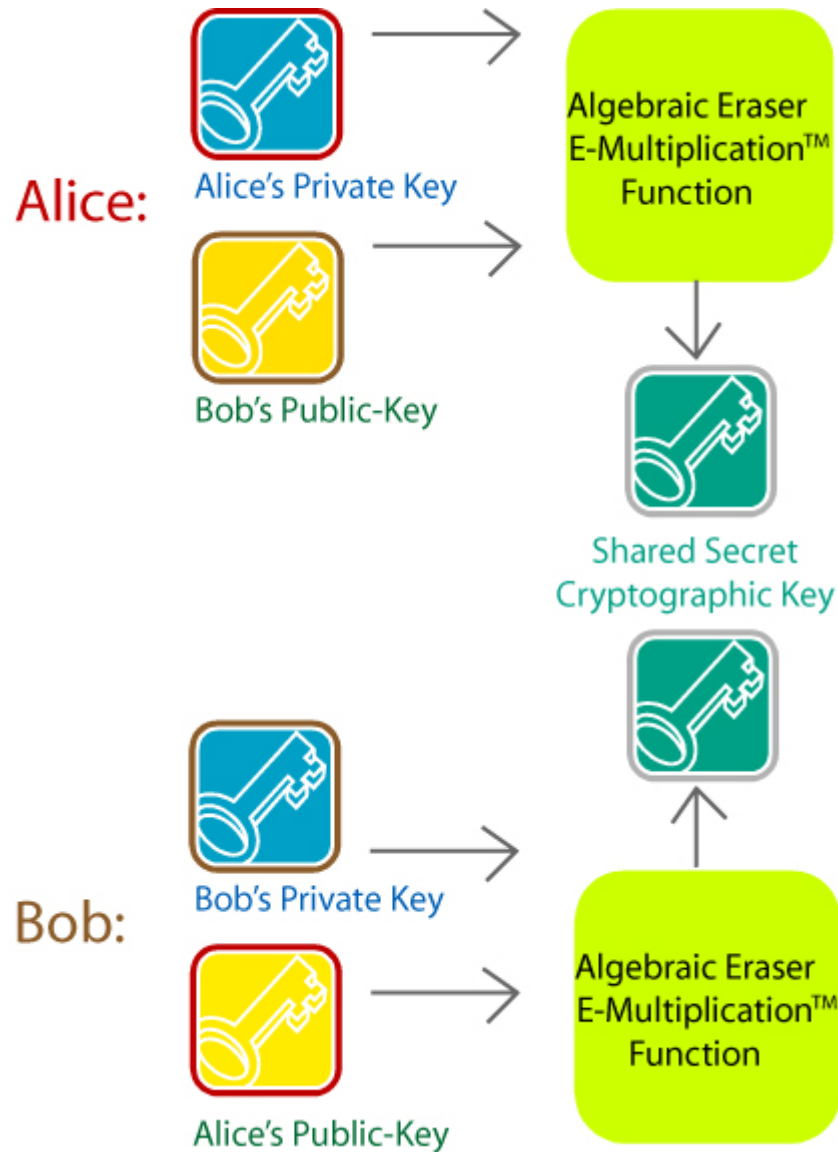


Figure 7: SecureRF's Algebraic Eraser based cryptosystem

The output of the Algebraic Eraser system is also a shared secret cryptographic key that can then be used as a private-key system for encryption and decryption of the message passed between Alice and Bob.

Some of the significant features of SecureRF's approach include:

- Low power consumption for the processor, high speed implementation for real-time processing and a small computational footprint.
- Key management is mitigated because secure disposable keys can be generated for each communication session.
- Algebraic Eraser protocols are secure against replay attacks and man in the middle attacks.

- All Algebraic Eraser protocols run in linear time in the key size, while all other systems, including RSA and Diffie-Hellman protocols scale quadratically. When viewed from a user resource perspective, the graph in Figure 8 gives a high level indication of the intrinsic benefits of the Algebraic Eraser, especially as the security strength levels increase. The Algebraic Eraser is much less resource intensive and can run on devices that have limited computing power.

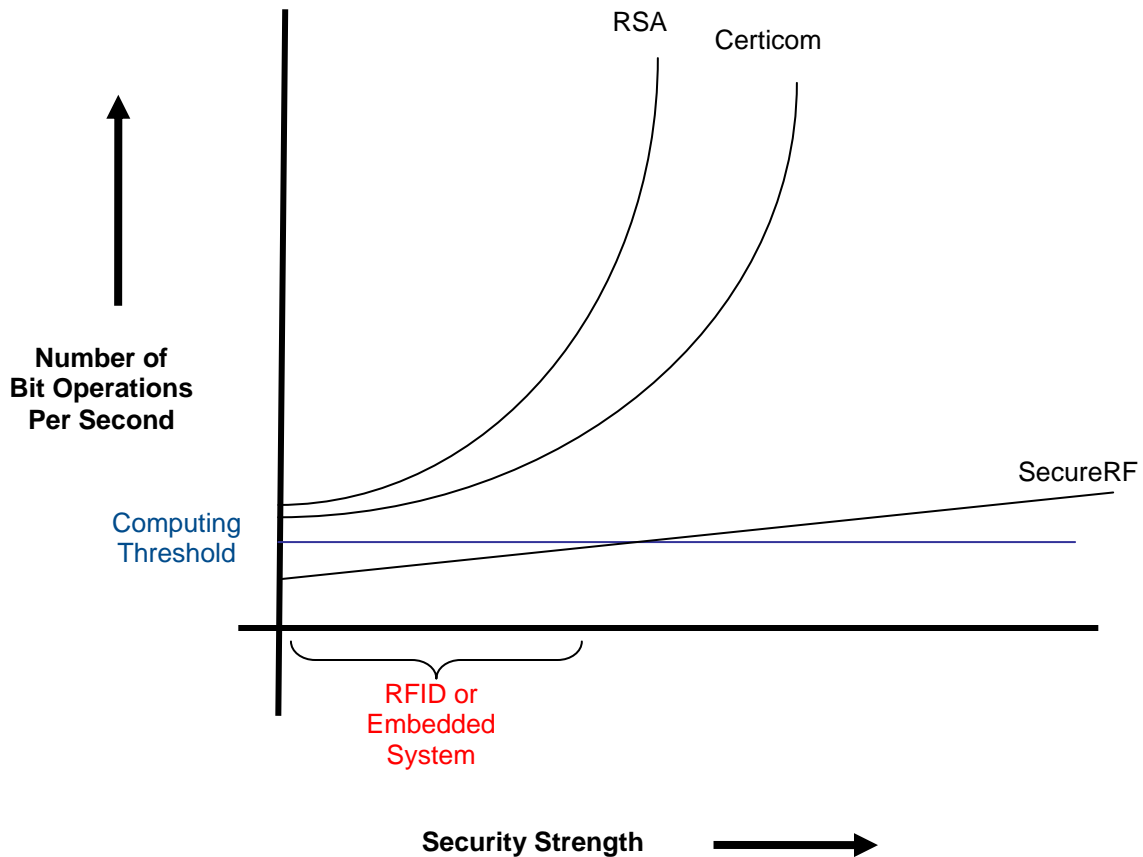


Figure 8: Comparison of computing resources required as security strength increases

Security protocols which are based on secret methods are often insecure because they have not had the benefit of widespread testing and analysis. Strong cryptographic security methods are published for peer review. Details about how the Algebraic Eraser’s key agreement protocol for public-key cryptography is suitable for low resource devices, such as RFID tags, have been published by The American Mathematical Society in the peer-reviewed book “[Algebraic Methods in Cryptography](#)” in the section titled *Key agreement, the Algebraic Eraser™ and Lightweight Cryptography*. A version of this paper is available on SecureRF’s web site (<http://www.securerf.com/white.shtml>).

There are several groups working on the mathematical security using braids. The single conjugate work of Ko, et al from Korea is a very different method than the “dual conjugate”

protocol that SecureRF's uses. There are several published papers pointing out that the attacks on the Korean's method do not work on the dual-conjugate braid methods used by SecureRF.

More information about how the Algebraic Eraser can be used for RFID, embedded systems and other platforms with low computing power can be found in SecureRF's white paper – *Security in Low Resource Devices*, available at <http://www.securerf.com/white.shtml>.

4. SecureRF's Mathematician/Cryptographers

The inventors of SecureRF's public-key cryptosystem are co-founders Dr. Michael Anshel, Dr. Dorian Goldfeld and Dr. Iris Anshel.

Dr. Michael Anshel is a security thought-leader and world-class mathematician with expertise in the field of cryptography. Dr. Anshel has authored and co-authored numerous papers in the area of public-key cryptography, is the co-inventor of four patents in the area of cryptography, zeta-one-way functions, and braid groups and has received numerous fellowships and honors. He is a professor in the Department of Computer Science at The City College of New York.

Dr. Goldfeld is a world-class mathematician who has published over 50 papers and lectured internationally on a wide range of cryptographic topics and methods including applications of elliptic curves, quadratic fields, zeta functions, public-key cryptography, and group theoretic approaches to public-key cryptography. In 2009 he was inducted as a Fellow of the prestigious American Academy of Arts and Sciences. He is the co-inventor of three patents in the areas of multistream encryption systems, high-speed cryptography, and cryptographically secure algebraic key establishment protocols based on monoids. He has been a professor in the Faculty of Mathematics at Columbia since 1985.

Dr. Iris Anshel, SecureRF's Chief Scientist, is an accomplished mathematician and cryptographer. In addition to extensive research and publications, Dr. Anshel has experience in the commercialization of security technology. As a co-founder of Arithmetica, she was responsible for documenting methods for commercial deployment of new cryptography protocols including the AAG Braid Group Cryptosystem and supported sales and business development activity.

5. About SecureRF

SecureRF Corporation provides security solutions for embedded systems and radio frequency identification (RFID) technology used in high value asset tracking, monitoring and anti-counterfeiting applications in the pharmaceutical, food, defense, homeland security and other sectors. The company's technology, based on a breakthrough in cryptography that is very computationally efficient yet highly secure, provides strong authentication and data protection.

More information about SecureRF can be found on its Web site at www.SecureRF.com. SecureRF's insights into RFID Security can be found on its blog at www.SecureRF.com/RFID-Security-blog/.

SecureRF, SecureRF logos, Algebraic Eraser, E-Multiplication and E-Multiply are trademarks of SecureRF Corporation.

6. Glossary of Terms

Asymmetric - Two parts of a thing are not similar. In asymmetric cryptography each user has a different private-key, which only the user knows, and a mathematically-related public-key which can be made public and freely distributed, i.e. “public-key cryptography.”

Cipher – A pair of algorithms used in cryptography to implement a particular form of encryption and decryption. The term “cipher” is used to describe symmetric key techniques.

Ciphertext - The encrypted text of a message which is unreadable.

Cryptosystem – A suite of algorithms needed to implement a particular form of encryption and decryption, usually including an algorithm for key generation. The term "cryptosystem" is used to refer to both symmetric and asymmetric techniques.

Decryption - The process of transforming an encoded message or ciphertext into plaintext.

Encryption - The process of transforming information to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key.

Key management - The processes that deal with the creation, distribution, authentication, safeguarding, storage, destruction and replacement of keys used in a cryptographic system.

Man-in-the-middle attack - A security attack on a communication channel in which the attacker can intercept and possibly modify messages transmitted between two parties without the communicating parties' knowledge.

Plaintext - Information a sender wishes to transmit to a receiver, a message before encryption or after decryption.

PKI – Public-Key Infrastructure.

Replay attack - A security attack on an authentication system by recording and sending a copy of previously sent valid messages (or parts of messages).

Symmetric - Two parts of a thing are similar. In symmetric cryptography the same key is used for both encrypting and decrypting data, i.e. “private-key cryptography.”