Public Key Cryptography

Andrew M. Odlyzko

The concern for secure communications has moved cryptology from an arcane science, best known to the military and diplomatic services, to an integral component for business and personal communications. Public key cryptography, developed within the last two decades for commercial use, has become an exciting technology for the communications industry, as wired and wireless voice, data, and image networks continue to proliferate and interconnect. Public key cryptography helps provide secure communications, at reasonable costs, for general communications. It solves several important technical problems, especially those of key management and digital signatures, that are vital for information processing, but it also presents some drawbacks. This article explains public key cryptography, its benefits and limitations.

Introduction

Mention cryptology to the average person and it brings to mind the coded messages of governments that enemy spies are constantly trying to crack. The exciting stories of the Allies' success in *breaking* the Axis codes are well known. But the less obvious success stories of how the Allies *protected* their codes is less known—but no less important. Just one small example is the *encrypted* private line telephone that can be seen today at the bedside of President Roosevelt's home in Hyde Park, N.Y. That phone provided him secure communications with the White House.

The same can be said for businesses. While business communications have always been subject to eavesdropping—letters can be steamed open, copper wires can be physically tapped, microwave signals can be intercepted and demultiplexed, and even fiber optic facilities can be tapped—eavesdropping itself was never without risk, and certainly was expensive, to whomever attempted it. With the growth and interconnectivity of public and private wired and, especially, wireless data networks, and the proliferation of wireless voice and data terminals, the security of both wired and wireless communications is becoming a more prominent issue in personal and business communications than ever before.

Public key cryptography was invented to provide secure information for civilian systems more easily, and far less expensively, than was possible with the traditional methods used by military and diplomatic agencies. Public key cryptography promises that secure communications channels will be available to businesses and to individuals.

Complexity of Conventional Systems

Conventional cryptosystems can provide security, but at substantial cost. An extreme example is that of the Vernam cipher, invented at AT&T in 1917. Using the Vernam cipher, Alice and Bob can communicate secretly, if they agree ahead of time on a string of randomly generated *encryption* or *key* bits:

$$k_1, k_2, k_3, k_4, \text{ etc.},$$

which will be used to encipher messages.

If Alice wishes to convey to Bob a message that is represented by the *message* bits:

$$m_1, m_2, m_3, m_4, \text{ etc.},$$

she uses the randomly generated key bits to

Acronyms Used in This Paper

DES — Data Encryption Standard
DSS — Digital Signal Standard
IS-54-B — An industry standard for North American digital cellular systems
NIST — National Institute of Standards and Technology, formerly the National Bureau of Standards (NBS)
TIA — Telecommunications Industry Association

encode the message bits before transmitting them to him.

Thus, her *enciphered* message bits would be:

$$e_1 \equiv m_1 + k_1, e_2 \equiv m_2 + k_2,$$
$$e_3 \equiv m_3 + k_3, e_4 \equiv m_4 + k_4 \text{ (mod 2), etc.}$$

Each enciphered bit (e_j) is the exclusive-or of bits m_j and k_j.

Bob, who receives the enciphered bits

$$e_1, e_2, e_3, e_4, \text{ etc.,}$$

recovers Alice's original message bits m_1, m_2, etc., by the operation:

$$m_1 \equiv e_1 - k_1, m_2 \equiv e_2 - k_2 \text{ (mod 2), etc.}$$

If the key bits k_1, k_2, etc., are truly random, and are never used more than once, this cipher is called a "one-time pad." This means that a second message from Alice to Bob, or Bob's reply to the first message from Alice, would use as key bits:

$$k_{n+1}, k_{n+2}, k_{n+3} \ldots,$$

where n is the total *number* of bits previously transmitted. The one-time pad cipher is unbreakable and, indeed, is the only cryptosystem that has been proven to be unbreakable.

The reason that the Vernam one-time pad is regarded as an extreme example of a costly conventional cryptosystem is that it requires huge numbers of random bits, as many bits as Bob and Alice might wish to transmit to each other. Moreover, those key bits have to be created and conveyed securely to just Alice and Bob, without allowing anyone else to learn what they are. This can be done, of course, in cases requiring extreme security. For example, the well-known Washington-Moscow hot-line is reputedly encrypted with the one-time pad. But such a system is not adequate for the civilian marketplace, where the volume of transmitted information is huge.

U.S. Data Encryption Standard

The best known and most widely used cryptosystem today is the U.S. Data Encryption Standard (DES).[1,2] DES uses a 56-bit key. Therefore, if Alice and Bob wish to communicate using DES, they do not need to generate beforehand as many random key bits k_1, k_2, etc., as they had message bits to transmit. Instead, they only need to agree on a single 56-bit DES key.

When DES was first proposed as a standard, there were suspicions that it might contain "trapdoors" through which government agencies could easily decrypt business and personal transmissions. These concerns have been allayed, however, by research done over the last two decades. Now, the general consensus is that DES is a strong system—for its key size.

However, it is felt that a 56-bit key is, in general, too short for many critical applications. For about one million dollars, one can build a parallel computer that would typically require only four hours of processing to perform an exhaustive search of the 2^{56} possible keys (72,057,594,037,927,936) that DES might have.[3] For many applications, this level of security is not adequate, especially since key search machines are becoming faster and cheaper to build.

Exhaustive key attacks can be thwarted, however, by using ciphers that are more complex than DES. One such system is triple-DES, which consists of three encryptions with the basic DES, controlled by two keys, for an effective key size of 112 bits.[1,2]

Problems of Key Distribution and Signatures. The levels of security, and the probabilities of DES, triple-DES, and other similar systems being broken over time can be estimated by skilled cryptographers. Since the key sizes are moderate, the main disadvantage of the Vernam cipher—the enormous number of randomly selected keys required—is avoided. However, conventional cryptosystems have two serious problems:

- Distributing the key to its users, and
- Providing digital signatures.

Key Distribution. For Alice and Bob to communicate using DES, they need to have a 56-bit key that nobody else knows. If Bob and Alice were going to

Panel 1. The Diffie-Hellman Key Exchange Method

Suppose that Alice and Bob wish to establish a secret key that only they will possess. To do this, they agree on a large prime p and an integer g, which can be transmitted publicly. (Indeed, these numbers p and g might be the same for many people. We will say more about their choice below.) Then Alice chooses a random integer a, $1 \le a \le p-2$, and Bob chooses a random integer b, $1 \le b \le p-2$. The integers a and b are kept secret from anyone else. Alice then computes A:

$$A \equiv g^a \pmod{p}, \quad 1 \le A \le p-1,$$

and Bob computes B:

$$B \equiv g^b \pmod{p}, \quad 1 \le B \le p-1,$$

where $x \equiv y \pmod{p}$ means that the remainders obtained by dividing x and y by p are the same. The computations of A and B can be carried out quickly, with no intermediate results larger than p^2.

Then Alice transmits A to Bob over an open channel, and Bob transmits B to Alice. Next, Alice computes:

$$X \equiv B^a \pmod{p}, \quad 1 \le X \le p-1,$$

and Bob computes:

$$Y \equiv A^b \pmod{p}, \quad 1 \le Y \le p-1.$$

Since

$$B \equiv g^b \pmod{p}$$

and

$$A \equiv g^a \pmod{p},$$

we find that:

$$X \equiv g^{ba} \equiv g^{ab} \equiv Y \pmod{p},$$

and, therefore, $X = Y$.

Hence, Alice and Bob do obtain the same integer $X = Y$, which can then be used to derive a key for a conventional cryptosystem.

An eavesdropper, who listens to the conversation between Alice and Bob, sees A and B. However, to derive X from A and B, it appears (although this has never been proved) that the eavesdropper has to compute either a or b. Computing either of these numbers is an instance of the discrete logarithm problem, and in general appears very hard to do. Some precautions have to be observed (the prime p has to be large, $p-1$ has to have at least one large prime factor, the multiplicative order of g modulo p has to be large, etc.). This protocol is widely used.

communicate only with each other, deciding on a key would be a simple matter. But what if they wanted to communicate with a number of other people as well? With n people or computers that might need to communicate with each other, the number of keys necessary is $n(n-1)/2$.

Since there are already over 20 million users of the increasingly popular Internet network, just to allow any two users to communicate in secret would require that 200 trillion keys be available, and each user would have to keep a file of 20 million keys, one for each potential correspondent. This is clearly a major defect of conventional cryptosystems, and was the main motivator for the invention of public key cryptography.

Digital Signatures. Another area where conventional cryptosystems are deficient is digital signatures, guaranteeing to the recipient that the message has not been altered and was sent by the person who claims to have sent it.

While key management can often be handled using classical cryptographic methods (as will be explained in "Limitations of Public Key Cryptosystems"), there is no effective way to ensure a digital document is authentic without using public key encryption methods to protect the digital signature from unauthorized use. With the rapid spread of electronic transactions of all sorts, the need to provide digital signatures presents a serious problem.

Public Key Cryptosystems

There are several public key systems that are widely used.

Diffie-Hellman Key Exchange. Public key cryptography was invented in the 1970s by Whit Diffie, Ralph Merkle, and Martin Hellman at Stanford University[1,2]. The first practical public key system was the Diffie-Hellman key exchange system, presented in Panel 1. If Alice and Bob wish to communicate in secret, they can use the

Panel 2. The RSA Cryptosystem

The RSA cryptosystem relies for its security on the difficulty of factoring an integer into primes. If Alice wishes to allow secret messages to be sent to her, she chooses two large primes p and q, and forms $n = p \cdot q$. She then selects a random integer e, $1 < e < n$, such that e has no integer divisors > 1 in common with either $p - 1$ or $q - 1$. She then publishes the pair (n,e) as her public key, but keeps p and q secret.

To send a message to Alice, Bob transforms it into blocks of integers, each of $\leq \log_2 n$ bits. If a particular block is regarded as the binary representation of an integer m, $0 \leq m < n$, then Bob computes:

$$c \equiv m^e \pmod{n}, \quad 0 \leq c < n,$$

using the same consecutive squaring method as in the Diffie-Hellman method, and transmits c to Alice.

To decrypt the transmitted message c, Alice uses a procedure similar to the encrypting one, namely:

$$m \equiv c^d \pmod{n},$$

where d is her secret decryption exponent. If the factors p and q are known, d can be computed easily from e, since we need:

$$ed \equiv 1 \pmod{p-1},$$

and

$$ed \equiv 1 \pmod{q-1}.$$

There is no known way to break the RSA system without finding the prime factors p and q of n.

Diffie-Hellman technique to establish a common secret key through the exchange of public messages. This secret key can then be used to encrypt a transmission using a conventional cryptosystem, such as DES.

If suitable precautions are observed, an eavesdropper who intercepts the entire exchange of messages still will not be able to recover the key and, thus, will not be capable of interpreting the communication.

The Diffie-Hellman scheme is, perhaps, the most commonly used public key system, and is incorporated, along with some special enhancements, in AT&T's portable Surity™ 3600 telephone device, marketed by AT&T Secure Communications Products Business Unit.

RSA Algorithm. The most famous public key cryptosystem is the RSA algorithm,[1,2,4] invented by, and named after Ron Rivest, Adi Shamir, and Len Adleman of the Massachusetts Institute of Technology (MIT). It was discovered shortly after the Diffie-Hellman method was announced. RSA, described in Panel 2 and in D'Angelo,[5] enables people who have not had a chance to establish a common secret key to communicate privately. RSA also provides numerous other capabilities.

Key exchange is simple to implement using RSA. If Alice and Bob wish to establish a secret key for use with DES, or any other conventional cryptosystem, Alice can simply select a secret key of her own and send it to Bob encrypted with the *public half* of Bob's encryption key, which would be available in a publicly accessible database. Bob can then decrypt Alice's message, which can only be done by Bob when he uses the second, *private half* of his encryption key. It should be noted that Alice can encrypt a message with Bob's public key, but that public key cannot be used to decrypt the message—only the private half of Bob's key will do that. Now, once Bob has decrypted Alice's secret key, he can send a message to her using her secret key.

Also important is the digital signature capability of RSA, discussed in Panel 3.

Digital Signal Standard. There are many other public key cryptosystems. For example, there are digital signature schemes that are based on discrete logarithms, and not on RSA, such as the proposed U.S. Digital Signature Standard (DSS).[1]

Identity-Based Systems. There also are various systems with additional capabilities, such as the so-called identity-based cryptosystems,[1-2,4] in which users obtain certificates (a string of bits that validates the user) from a central authority (such as a corporate security department) that encode their basic identification information, limits of validity, and so on. These systems enable any two participants to generate a common secret key, while simultaneously verifying each other's identity, without having to access any database of public information.

Limitations of Public Key Cryptosystems

Public key cryptosystems are already widely used, and are likely to become even more widespread. They do have limitations however, that prevent them

Panel 3. Digital Signatures

There are methods for creating digital signatures using conventional cryptosystems, but they are clumsy. In contrast, public key systems provide a very elegant solution to this problem. The U.S. Digital Signature Standard (DSS), proposed by the National Institute of Standards and Technology (NIST), is based on discrete logarithms, as is the Diffie-Hellman system. Here we show a solution based on the RSA cryptosystem, which is described in Panel 2. Suppose that Alice's public key consists of (n,e). To sign a message m, when m is an integer in the range $0 \le m < n$, Alice attaches to it the integer:

$$x \equiv m^d \pmod{n}, \quad 0 \le x < n,$$

where d is Alice's secret decoding exponent. Since Alice knows d, this is something she and she alone can do. Bob, to verify Alice's signature of m, computes:

$$y \equiv x^e \pmod{n}, \quad 0 \le y < n.$$

Since n and e are public, Bob can perform this operation. The property of the RSA cryptosystem discussed in Panel 2 guarantees that $y = m$. This proves to Bob that x was indeed generated by Alice since there is no known way to generate x from m without knowledge of the secret integer d.

The digital signature scheme described above can be used to show some of the pitfalls of using cryptosystems. It is possible for a system to fail even if the basic algorithm is secure. This can happen to both public key and conventional cryptosystems. For example, Alice should use separate keys (n,e) and (n',e') for encryption of information (which is sent to her) and for digital signatures (which she generates). To see the reason for this, suppose Alice uses a single key (n,e). Suppose that an eavesdropper overhears Bob sending to Alice the message:

$$c \equiv m^e \pmod{n}.$$

All an eavesdropper has to do to obtain m is to persuade Alice to sign c, since the signature of c is:

$$c^d \equiv m \pmod{n}.$$

Thus Alice's use of the same modulus and exponent in two different cryptosystems allows the eavesdropper to break them with Alice's unwitting cooperation. This is a case of protocol failure, and is one of the main vulnerabilities to be guarded against.

from being used as universally as their earliest proponents expected.

Computational Burdens. The primary limitation on public encryption systems comes from the computational burden they impose. Almost all the public key cryptosystems that are regarded as secure are based on number theory techniques that involve the multiplication of large integers. Intensive research over the last two decades has complicated the matter further by proving that the sizes of the numbers had to be increased to provide adequate security.

For example, Rivest, Shamir, and Adleman published a challenge in 1977 to break a version of the RSA system that relied on 129-digit integers. At that time, they fully expected this problem to remain unbroken at least until the end of this century. However, improved algorithms and a large distributed computation capability, which involved using the idle time on hundreds of computers around the world, succeeded in solving this challenge in 1994.

For information on the methods used in these and related attacks, and guidelines on recommended sizes of keys, see Odlyzko.[6] The computational requirements of public key cryptography are not as much of a barrier to its use today as was the case a decade ago, when special high-speed processing hardware was necessary. At that time, AT&T even produced a special modular multiplication chip for this purpose. Today, the increased capabilities of microprocessors have made them fast enough to carry out the necessary computations. Still, these computations are a burden, especially where power is limited and only simple processors can be used.

In contrast, conventional cryptosystems are still 10 to 1000 times faster than public key ones. Therefore, the encryption of messages on public and private networks, where throughput speed and the cost of network time are still issues, invariably is being done using conventional cryptosystems. Public key schemes are used only for special tasks in which their unique capabilities are needed, such as exchanging keys between users, authenticating users to systems and other users, and appending digital signatures to messages.

Competitiveness of Conventional Systems. Another reason public key schemes are not used more widely is that many of their capabilities can be obtained from conventional cryptosystems. For example, the section "Problems of Key Distribution Systems" explained the key management problem; if there are n users in a system, then $n(n-1)/2$ keys are needed to allow any two users to communicate, and every user has to store $n-1$ keys. If public key systems are used, then only n keys are

needed, as only a single key for each user has to be stored. This key does not have to be safeguarded, as it can, and should, be placed in a publicly accessible database. This database has to be secure against unauthorized modifications. (Identity-based cryptosystems[1] can sometimes even eliminate the need for this database.)

In many situations, however, an almost equally satisfactory solution can be constructed with conventional cryptosystems. If there is a secure, trusted center in the system, then each user needs only a single secret key that is shared with the center. If Alice wishes to communicate with Bob, she sends a message to the center, encrypted with the key she and the center share, requesting that a key be generated for the Alice-Bob conversation. The center creates such a key and sends it to Alice (encrypted with the key Alice and the center share) and to Bob (this time encrypted with the key that Bob and the center share). Bob and Alice can then decrypt the center message containing the temporary, common key, by each using his or her own key. Now, Alice and Bob can communicate using the common key that the center provided.

They obtain not only privacy of the communication, but also assurance of each other's authenticity, via the center's validation of each of them. The disadvantage of this approach is that the center has to be reachable at all times. It also has to be extremely secure from hackers, and absolutely trustworthy, since it possesses the means to listen in on all conversations in the system.

Some Problems Are Relative. While public key cryptography provides ways to solve the problems of conventional cryptosystems, these drawbacks are often perceived as not being very significant, or else as being preferable to not having to implement more cumbersome public key schemes.

For example, consider the situation in which a customer of a mobile phone service has subscribed to an encryption capability that encodes all communications to and from the subscriber. When the mobile customer "roams" outside the geographic area served by his or her cellular system, how is that encryption service to be provided by the "foreign" cellular carrier?

IS-54-B Authentication. One answer is found in the IS-54-B authentication system[7] for North American digital cellular systems, which has been adapted for other wireless schemes as well. Designed by Jim Reeds and Phil Treventi of AT&T Bell Laboratories, the IS-54-B system uses a hierarchy of temporary *shared secrets*, strings of bits that permit foreign cellular systems to provide encryption services to a roamer customer. When a customer makes the first call in the geographic area of a foreign cellular system, that system can query the home system to determine how that call and future ones by that customer should be encrypted. Thus, the home system temporarily shares with other cellular systems a form of the encryption, to provide the roamer with an encrypted transmission. The IS-54-B system also provides challenge-response techniques to verify the identity of users' mobile units without the use of public key cryptosystems.

If, instead of using IS-54-B, a public key system were employed to support the communications between the home and foreign cellular systems, it would:

- Reduce the need for the very efficient communication now required by IS-54-B between different cellular systems when they are trying to encrypt the transmissions of roamers from other systems, and
- Prevent the denial of service, which might sometimes occur as a result of communication overloads or breakdowns.

There are, as always, tradeoffs to be considered. Since there are relatively few cellular operators in this country, and the Reeds-Treventi system requires only occasional communication between them, the industry standards group, the Telecommunications Industry Association (TIA), has decided not to use public key cryptography in the immediate future. Public key cryptosystems are under consideration, however, for other wireless systems.

Licensing Issues. Another reason why public key cryptosystems are not used more widely involves patent licensing issues. Most of the basic public key algorithms are patented, and several of the more prominent ones are controlled by a private company, Public Key Partners. Many corporations, including AT&T, Apple, Lotus, and Microsoft, have licenses to these patents. However, there are still many unresolved issues, especially those concerning the proposed DSS. The U.S. government has publicly stated that it is committed to a no-fee access policy with this standard. However, Public Key Partners claims that DSS infringes on its patents, and no agreement has been reached as yet on this question.

Conclusions

Public key cryptosystems are valuable security tools, as they offer essentially the only way to provide

digital signatures, and are often the preferred method for authentication and key distribution. However, public key systems currently don't appear to be good candidates for encrypting general traffic, and can often be dispensed with in networks that have a highly secure and trusted central database.

It will be interesting to follow the trends in computational capabilities, database management, and user demands to see how public key cryptology will evolve.

Acknowledgements

The author thanks Joan Feigenbaum, David Maher, and Michael Reiter for their comments.

References

1. B. Schneier, "Applied Cryptography," Wiley, 1994.
2. G. Simmons, ed., "Contemporary Cryptology," IEEE Press, 1991.
3. M. J. Wiener, "Efficient DES Key Search," *TR-244*, May 1994, School of Computer Science, Carlton University, Ottawa, Canada. Paper presented at the Rump Session of Crypto '93.
4. C. Pomerance, ed., "Cryptology and Computational Number Theory," American Mathematics Society, *Proceedings of Symposia in Applied Mathematics*, No. 42, 1990.
5. D. M. D'Angelo, B. McNair, J. E. Wilkes, "Security in Electronic Messaging Systems," *AT&T Technical Journal*, May/June 1994, pp. 7–13.
6. Andrew M. Odlyzko, Discrete logarithms and smooth polynomials, in "Finite Fields: Theory, Applications and Algorithms," G. L. Mullen and P. Shiue, eds., American Mathematics Society, 1994, in press.
7. "Cellular System Dual-Mode Module Station—Base Station Compatibility Standard (Rev. B)," Telecommunications Industry Association (TIA)

(Manuscript approved June 1994)

Andrew M. Odlyzko *is head of the Mathematics of Communications and Computer Systems Department at AT&T Bell Laboratories in Murray Hill, New Jersey. His organization is involved in studies concerning cryptology, formal verification methods, probability theory, and parallel processing. He joined the company in 1975. He has B.S. and M.S. degrees in mathematics from California Institute of Technology in Pasadena and a Ph. D. in mathematics from Massachusetts Institute of Technology in Cambridge.*