Office (**ARPA**/IPTO), this protocol has come to be known as the Transmission Control Protocol/Internet Protocol or **TCP/IP**. It is the protocol that has made a global **Internet** possible.

In their article, Kahn and Cerf identified a significant technical problem and proposed a protocol design and philosophy for its solution. The problem, which became evident in the early 1970s, is: How is it possible to interconnect diverse packet networks and to make resource sharing possible across the boundaries of dissimilar networks?

The **ARPANET** had solved the problem of creating a **packet-switching** network to connect diverse computers and diverse **operating systems**. To interconnect diverse packet networks, however, introduced new problems and complexities, including variations in packet format and size, in addressing mechanisms, and other conventions. A design for a protocol for internetwork packet communication would need to be able to accommodate such differences.

In early 1973, shortly after he joined ARPA/IPTO, Kahn began considering the problem of providing an architecture to accommodate heterogeneous packet networks. He identified these ground rules:

1. No changes would be required in the internal operations of participating networks.
2. Gateways would provide the means to reformat packets to meet the requirements of different networks and to route packets.
3. Communication would be on a best-effort basis, using transmission and retransmission of packets to get them to their final destination.
4. There would be no global control at the operations level.

These rules form the basis for an open networking architecture. Kahn recognized the need for a protocol to embody these ground rules and invited Cerf to collaborate with him on the design. They presented a draft article describing the design and philosophy for such a protocol in September 1973 at a meeting of networking researchers in Sussex, England. The published version of the article appeared in *IEEE Transactions on Communications* in May 1974.

The protocol design provides the means to create a metalevel architecture by designing software for host computers on the diverse networks and for gateways to interface between them. This makes it possible to set up an "association" between hosts on diverse networks without regard to determining any particular path for data transmission. It identifies the need for a means of addressing that will be understood by the gateways and the hosts on the diverse networks. The gateways make it possible to reformat packets to accommodate the different packet sizes among different networks and to route packets. The flow control and windowing mechanism make it possible to transmit and retransmit packets until they are received and reassembled at the destination host. The article includes a number of other concepts to implement an internetworking protocol in diverse networks.

Rarely has a technical article had such an impact. It provides the design for the infrastructure for the global Internet. To understand the nature of the Internet, it is important to read, study, and understand the philosophy and design of the internetworking protocol described in this seminal article.

FURTHER READING

Abbate, Janet. *Inventing the Internet.* Cambridge, Mass.: MIT Press, 1999.

Cerf, Vinton G., and Robert E. Kahn. "A Protocol for Packet Network Intercommunication." *IEEE Transactions on Communications*, Vol. 22, No. 5, May 1974, pp. 637–648.

—*Ronda Hauben*

Public Key Cryptography

The term *cryptography* refers to the encoding of information in such a way as to make it incomprehensible for a third party. In public key cryptography, a public key is used to encode the information and a secret key is used to decode it. The *public key* is made known to the general public in a "telephone book," while the *secret key*, as the name implies, is known only to its owner. The advantage of this split-key method is that anybody can encrypt information for anybody else without having to arrange a previous exchange of secret keys. However, the encrypted

information can be decrypted only by the intended recipient of the information.

The principle behind public key cryptography and the most common **algorithms** can best be explained by example. Assume that a sender, Alice, and a receiver, Bob, want to communicate. Alice's message, which is a long series of bits in the computer, can be considered to be just a number. For example, all letters in this sentence could be encoded using the **ASCII** 8-bit code and its concatenation would produce a binary number with around 1000 bits. This is the message M that Alice wants to transmit.

Since the message M is a number, we could use another number, which we call the public key P, to transform the message into something very difficult to interpret. We could raise the number M to the power P, yielding an enormous number in which the original information has been scrambled. Alice sends M^P to Bob, and the message is decrypted by extracting the Pth root of this number, getting the original message M.

In this case the public key is P and the secret key is the inverse of P (i.e., $1/P$). Extracting the Pth root of a number corresponds mathematically to raising that number to the $(1/P)$th power. In this simple example, the method is not yet secure because from the knowledge of the public key P, the inverse $1/P$ can readily be obtained and anybody can decrypt the "secret" message.

However, this simple approach can be made secure by employing another type of arithmetic, called *modulo arithmetic*. Modulo arithmetic is done with a finite set of numbers: for example all numbers between 0 and 15 (i.e., 16 numbers). Addition is done in the normal way, but when the result is greater than 15, it "wraps around" and starts again from zero. In modulo-16 arithmetic, 2 plus 2 is 4, but 15 plus 1 is 0, 15 plus 2 is one, and so on. The result of the addition of two numbers in this arithmetic is the rest of the normal addition result when divided by 16.

Modulo arithmetic has many interesting properties. The addition operation can be defined as explained above. Every number has an inverse: for example, $4 + 12 = 0$. This means that the additive inverse of 4 is the number 12. The additive inverse of 8 is 8 itself. Additive inverses are like negative numbers in the usual arithmetic. Subtraction can then be defined as addition with the additive inverse of the second argument.

The important point, though, is that if addition can de defined, the multiplication operation can be defined too. To multiply 4 by 5 we add five times 4 to itself, yielding 20 in the normal arithmetic but 4 in the modulo-16 arithmetic. Having multiplication, it is also possible to raise a number to any power P, because this corresponds to P multiplications of the number with itself.

The nice property of modulo arithmetic is, first, that all results are bounded by the maximum representable number. In modulo-16 arithmetic, for example, no result can be larger than 15, no matter how many multiplications we perform or how large the exponent P of a number is. Second, although raising numbers to a known power is easy, finding the inverse of the power, in order to extract the Pth root, can be made extremely difficult by carefully choosing the modular arithmetic range that we want to use and the number P. Using modulo arithmetic, Alice still sends to Bob the number M^P (now computed in a modular setting), but extracting the Pth root of this number becomes almost impossible. In other words, although P is known, its multiplicative inverse $1/P$ is now not so easy to compute.

The most famous public key algorithm is the RSA method, named after its inventors, Ron Rivest, Adi Shamir, and Len Adleman. The public key is actually a pair of numbers (e,n), where e is the power to which the message will be raised and n the modular arithmetic range. For real applications these numbers are very large, usually in the range of several thousand bits. That is, Alice's message is actually the number M raised to the eth power, modulo n. The secret key is another pair (d,n), where d is just the number needed to invert the previous exponentiation operation. As said before, from the knowledge of (e,n) it is very difficult to compute d. The public and secret key are chosen in such a way that computing d requires the factorization of a very large number. Prime number factorization is a problem for which no efficient algorithm is known. RSA keys can in principle be computed and the method

can be compromised, but in practice the keys are so large and the needed computational effort so large that the algorithm is considered secure.

Public key cryptography was first proposed by Whitfield Diffie (1944–) and Martin Hellman (1945–) in 1976, but their method had some shortcomings that were solved with the RSA approach, introduced in 1977. U.S. Patent 4,405,829 was awarded to the inventors and is known as the *RSA Patent.*

FURTHER READING

Rivest, Ronald L., Adi Shamir, and Len Adelman. "On Digital Signatures and Public Key Cryptosystems." *MIT Laboratory for Computer Science Technical Memorandum* 82, Apr. 1977.

Schneier, Bruce. *Applied Cryptography: Protocols, Algorithms Source Code.* New York: Wiley, 1995.

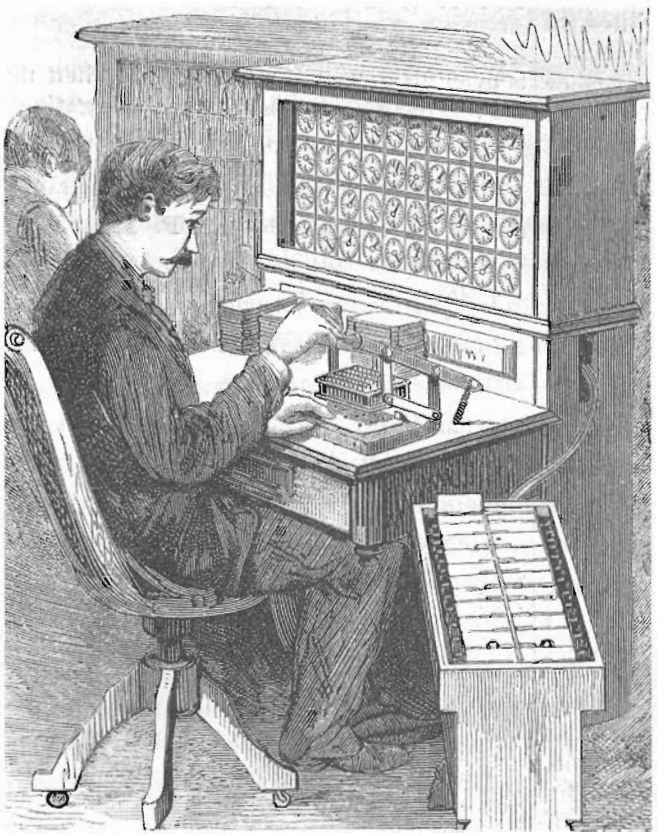
Stallings, William. *Data and Computer Communications.* Upper Saddle River, N.J.: Prentice Hall, 1996.

—Raúl Rojas

Punched Card Systems

Punched card systems stored data as combinations of holes in paper cards, to be processed by machines such as sorters and tabulators; cards were first punched and verified, and then sorted for tabulation. The first punched card systems were built during the 1890s to process counting statistics. During the next half century, they were used for summation statistics, bookkeeping, and for managing records. The machines evolved to enable quicker processing and to enable more complex operations.

The engineer **Herman Hollerith** (1860–1929) invented the first punched card system in the 1880s to process the 1890 census of the United States. Since 1850, U.S. censuses encountered problems due to a growing population, increasing exactness requirements, and the absence of a permanent institution to manage census processing. Hollerith's Electric Tabulating System was created to bring greater speed and accuracy to the 1890 census, the most extensive information processing effort yet attempted. The system was also used for census statistics in four European countries with limited success—of the four countries, only Norway used the system to process a second census.



A punched card tabulating machine. (© Bettmann/ CORBIS)

Population statistics were comparatively simple, requiring only counting and some sorting of records. The first punched card system utilized punched cards of 6⅝ by 3¼ inches (in.) (c. 16.8 by 8.3 centimeters [cm]) divided into 24 columns each of 12 punching positions, and it was comprised of two simple constructs with manual card feed: a key punch and a non-printing tabulator with a sorting box. The results were read from counters.

The standardized general statistics punched card system was developed by Hollerith from 1892 to 1907. It featured standard punched cards of 7¾ by 3¼ in. (c. 17.7 by 8.3 cm) with 45 columns each of 12 punching positions, round holes, and a numeric punching code; an adding nonprinting tabulator with mechanical feed; and a sorting machine with mechanical feed. Later, this system was improved through the use of automatic group control, the first method of record-controlled processing. Automatic group control enabled the tabulator to stop for manual reading or later automatic printing of a subtotal—for example,