Computer network cryptography engineering

by HARRISON R. BURRIS
TRW Systems Group, Inc.
Redondo Beach, California

ABSTRACT

A definition of system security and a unified description of encryption methods is presented as background. Alternatives for five major computer network design decisions related to the employment of cryptography with the network are discussed in terms of efficiency (security achieved) and cost.

INTRODUCTION

This paper discusses several design decisions relating to the employment of cryptographic techniques with computer based networks. Cost (hardware purchase prices and effect on performance parameters other than security) and efficiency (security achieved) measures are used for comparisons of alternatives. A definition of security is presented in section two and a method of describing basic types of encryption is presented in section three. The remainder of the paper compares the various network cryptography design alternatives which must be considered in planning efficient secure computer networks.

Given a professionally designed cryptographic algorithm (ignoring considerations of cryptanalytic resistance), the computer network designer should be aware of the impact of decisions concerning the method of employment of the cryptographic techniques upon the overall performance of the system design.

Following the example of Baran,[1] it is assumed that the system attackers are thoroughly familiar with all aspects of the system security design including cryptography and that only the current cryptographic keys are kept secret from the attackers.

There are three possible system security objectives for any computer based system, from single processor to distributed computer network.[2] These are:

Restriction of information to authorized persons.
Protection of system performance (availability and responsiveness).
Restriction of system resources to authorized persons.

One or more of these security objectives may be appropriate to a particular network depending upon the application. Penetration through the network communications system is a high probability threat to all three security objectives and network cryptography is an important system security technique which, depending upon the design decisions made, can either greatly strengthen or weaken the system's threat resistance.

Restriction of information

Restriction of information refers to the objective of preventing unauthorized persons from obtaining the information present in a system. There is nothing as effective as professionally designed cryptography for securing the information content of a communication.

Kahn[4] provides numerous examples of the ease with which "unbreakable" amateur ciphers have been broken by professionals, and restates the maxim that only the skilled cryptanalysts can determine the cryptographic strength of a cipher algorithm. Given professionally designed encryption techniques (hardware or computer algorithms) it is the job of the network engineer to insure that the method of employment does not provide an opportunity to compromise the encryption system.

Protection of system performance

Protection of System Performance refers to all actions taken to prevent system degradation. Degradation of performance is achieved either by causing a system to function with incorrect data so that the outputs are meaningless, or by slowing the system response time until even a correct response is useless. Degradation of performance can be achieved through the communications links by attacking the information being transferred or by monopolizing a sufficient portion of the available communications resources to slow down the system. The manner in which cryptography is employed with a particular network will either reduce the chances of successful attempts at system degradation or may greatly increase the impact of an attack (i.e., it may take much longer to resynchronize an encrypted communications link than one sending clear text).

Degradation of performance is measured as the increase in processing time or computational inaccuracy when the system is under attack over the processing time or computational inaccuracy when the system is operating in a benign environment. This measure should be distinguished from the security cost measured in increased processing time, hardware complexity, or computational inaccuracy of the secure system design over a system performing similar processing but without security capabilities.

An attacker can achieve degradation of performance by attacking the communications network with any combination of four types of attack: (1) Jamming, (2) Playback, (3) Alteration, and (4) Generation. Jamming refers to the introduction of some signal into the communications stream thereby preventing the reception of legitimate transmission. Playback refers to the recording of a legitimate transmission and then reintroduction of the message into the communications stream at some later time. Alteration refers to introducing changes into legitimate transmissions. Generation refers to introducing new messages into the communications stream. Since generation generally implies breaking the cipher before new messages can be produced it will be considered as precluded by the cryptographic algorithms being used in the network.

While cryptography is the principal tool for information protection its contribution to achieving the other security objectives is largely dependent upon the presence of other system security techniques such as Message Sender Identity Verification.[5] Just as a poorly designed encryption algorithm can cause the compromise of information, a poor application of either amateur or professionally designed encryption can greatly increase the impact of an attack aimed at degrading system performance.

Restriction of system resources

Restriction of System Resources refers to the objective of insuring that the system is used only for intended processing, or from another point of view, that each user pays for the system resources used. As for attacks through the communications links (and hence cryptographic design decisions), threats to this security objective are just another form of degradation of performance. From this view, the unauthorized processing load introduced represents the amount of degradation achieved.

ENCRYPTION METHODS

Most digital cryptography has developed as an aspect of digital communications, and this practical rather than theoretical outlook has resulted in digital cryptography being described in terms of a particular logic implementation. The resulting lack of a clear distinction between the cryptographic principles and the logic implementation presents a formidable barrier in assessing the performance (speed and cost) of a particular algorithm compared to other algorithms with similar properties.

The n character message to be encrypted (plaintext) is represented as a character string $P_1, P_2 \ldots, P_n$ and also as a bit string $B_1, B_2 \ldots, B_\alpha$ where $\alpha = n\beta$ and β is the number of bits per character. The n character long encrypted message (cipher text) is represented as a character string $E_1, E_2 \ldots, E_n$ and as a bit string $Y_1, Y_2 \ldots, Y_\alpha$. The different encryption methods (sometimes called privacy transforms[6,7,8,9]) are categorized according to the manner in which plaintext string P is transformed into the encrypted string E.

There are three major categories of enciphering:[4,8] (1) Transposition, (2) Substitution, and (3) Additive Encoding.

(1) *Transposition*—Transposition enciphers a message by reordering the characters of the plaintext. A transposition cipher is decrypted by reordering the encrypted message according to an inverse of the transform used to encrypt the message.

(2) *Substitution*—Substitution enciphers a message by replacing the characters of the plaintext with other characters, perhaps from another alphabet.

(3) *Additive Encoding*—Additive Encoding enciphers a message by combining the bits of the plaintext with the bits of a binary string using the exclusive OR (binary add) function. The encrypted message is decrypted by repeating the exclusive ORing of the encrypted text with the identical binary string.

Transposition

Define a transposition vector, T, of length n such that each value of T controls the transposition of the corresponding character in P into string E according to

$$E_{T_i} \leftarrow P_i \text{ for } 1 \leq i \leq n \text{ and } 1 \leq T_i \leq n \qquad (1)$$

Clearly, $T_i \neq T_j$ is required for $i \neq j$ and $1 \leq i \leq n$ and $1 \leq j \leq n$. The enciphered message is transposed back to the plaintext (decrypted) by the inverse transposition vector, $\bar{T}$, such that

$$P_{\bar{T}_i} \leftarrow E_i \text{ for } 1 \leq i \leq n \qquad (2)$$

where $\bar{T}$ is related to T by $\bar{T}_{T_i} \leftarrow i$ for $i = 1$ to n.

Plaintext messages of length greater than n can be encrypted using a transposition vector T of length n by partitioning P into a series of n long character strings and transposing each separately. The last string of P can be padded to length n with pseudorandom characters without weakening the transposition system.

Substitution

Define the plaintext alphabet, A, as the ordered set of all characters from which characters can be chosen to generate cleartext message strings. Define $\lfloor a \rfloor A$ as the sequence number of character a in the ordered alphabet A.

Define a substitution alphabet, S, such that for every $a \epsilon \{A\}$ there corresponds an $s \epsilon \{S\}$. The correspondence or mapping of A into S is determined by a substitution table F. The substitution table F is defined such that $j = f_i$ indicates a correspondence between A and S such that a_i corresponds to s_j. Table F could in some cases be represented as a function rather than a table which will be done now for brevity. Table I indicates two alphabets A and S whose characters happen to be mutually exclusive.

TABLE I

I	A	S	I	A	S
1	A	S	10	J	2
2	B	T	11	K	3
3	C	U	12	L	4
4	D	V	13	M	5
5	E	W	14	N	6
6	F	X	15	O	7
7	G	Y	16	P	8
8	H	Z	17	Q	9
9	I	1	18	R	0

For an F table defined according to the function $f_i = i$, the correspondence between the alphabets of Table I would be $Y \epsilon S$ corresponds to $G \epsilon A$. For the function $f_i = n + 1 - i$ with $n = 18$, Table I indicates the correspondence of character $4 \epsilon S$ to $G \epsilon A$.

A plaintext message P is encrypted by substitution on a character by character basis where P_i is encrypted according to

$$E_i \leftarrow S_{f_j} \text{ where } j = \lfloor p_i \rfloor A \tag{3}$$

The encrypted message string is decrypted using the inverse transform vector F defined so that

$$P_i \leftarrow A_{f_j} \text{ where } j = \lfloor s_i \rfloor S \tag{4}$$

Substitution ciphers are not explicitly influenced by the length of the plaintext strings.

Substitution ciphers where only one substitution alphabet and one substitution function are defined are called monalphabetic (including $S \equiv A$ except that for this case $f_i \neq i$ is required to avoid an identity transform). Polyalphabetic substitution ciphers[4,10] can be represented as multiple mappings (multiple F's) into a single S, as multiple alphabets controlled by a single F, or as a combination of multiple S's and F's. In order to complicate the cryptanalysis of substitution ciphers, several characters in S are often defined as being equivalent to one character in the plaintext alphabet. These sets of equivalent substitution characters are called homophones.[4]

Additive encoding

An additive encryption system transforms the plaintext bit string B into the enciphered string Y by applying the exclusive OR operation to string B and ciphering string X on a bit by bit basis where

$$Y_i \leftarrow B_i \oplus X_i \text{ for } i = 1 \text{ to } \alpha. \tag{5}$$

It is the property of the exclusive OR operation that string Y can be decrypted by a repeated application of the transform

$$B_i \leftarrow Y_i \oplus X_i \text{ for } i = 1 \text{ to } \alpha. \tag{6}$$

Partitioning the encrypted bit string into characters shows a key additive encryption to be equivalent to a polyalphabet substitution.

Plaintext strings longer than the coding string can be encrypted by partitioning the plaintext string into a series of bit strings of length α and encrypting each separately. The last string can be padded (random characters should be used for padding to strengthen the crypto system).

Encryption primitive security considerations

Jamming and Playback are not directed against the cryptographic primitives. For this presentation, generation has been eliminated as a threat (professional cryptographic algorithms), so only alteration remains as an attack directly influenced by the cryptographic primitives embodied in an implementation.

With a transposition cipher the plaintext characters are replicated in the ciphertext string so the attacker can precisely determine the plaintext character that result if a change to a ciphertext character is made. However, since the T vector is unknown, the correct position occupied by each character in the plaintext string cannot be determined. With both substitution and additive ciphers, the exact position of an altered character is known to the attacker but the exact plaintext character represented by each ciphertext character cannot be determined. Numerous highly effective probabilistic attacks can be made against substitution and additive cipher systems.[5] Cryptographic implementations employing multiple transposition and substitution or additive primitives can counter all but brute force alteration attacks since both position and resulting plaintext are unknown.

SYNCHRONIZATION

Synchronization of cryptographic devices is the process by which the encrypter at the sending end and decrypter at the receiving end are kept in step with each other. Three synchronization alternatives exist. The first two are in general use, the third is considered a practical proposal.

(1) *Link Synchronous Encryption*—The term link

synchronous is applied to an employment of cryptographic devices in which a one directional (simplex) point-to-point communications channel is enciphered such that a continuous stream of encrypted characters appear on the communications link and the receiving crypto device (the decrypter) is kept in step with the key of the sending encryption device by counting characters in the received data stream. In order to maintain the continuous character stream when no data is available to be sent, the encrypting end of the link generates a string of pseudorandom padding characters which are switched in and out of the transmission stream as required.

Link synchronous encryption is highly susceptible to degradation of performance attacks, since once the encryption devices are forced out of synchronization (i.e., by jamming) it requires a relatively long period to reestablish the network. Thus, a short duration jamming attack on the part of the attacker can deny use of the communication link for a period much longer than the period of jamming.

Because the communications link is continuously in operation, the network attacker is denied information about the volume and frequency of message traffic. This is called transmission security (TRANSEC).

The costs of this synchronization method are extremely high in terms of encryption equipment and communications resources (radio frequency spectrum or wire lines) required. The upper bound for a fully interconnected n node network is n^2-n links (dedicated frequencies or wire lines) and 2(n^2-n) encryption devices.

(2) *Packet Synchronous*—A packet is a block of characters, which may be either a segment of a message or an entire message, and may or may not be of fixed length. The term packet synchronous will be used to describe methods of synchronization which rely upon the appending of crypto synchronization information to the header of the packet in order to set the decryption device to the appropriate key.

In this mode, packets may be deleted without detection, and playback is possible. These attacks are facilitated by packet synchronization because as long as the synchronization and message text are associated together, both can be sent to a receiver at a later time and still be decrypted correctly. This method does not require the time consuming resynchronization processes of link synchronous systems since each packet carries its own synchronizing information.

Since a network can be established in which each node (and encryption device) recognizes its own address in the packet header, the costs of this method are considerably less than those for link synchronization since dedicated links and encryption devices are no longer required. For an n node network offering fully interconnected routing only n links and n devices are required.

(3) *Clock Synchronous*—Clock synchronous is a term proposed for the following method of encryption synchronization. The use of extremely accurate atomic (Rubidium or Cesium Beam) clocks for achieving synchronization of communications devices has been proven and portable clocks are available.[11] It should be possible to use the same methods to synchronize encryption.

In this mode a clock at each node is used to control the advance of the key. The clock time at which encryption was begun is appended to the message packet and serves as the synchronization information to set the key at the receiving node. If the message is not received within a set time period after encryption, it is rejected. Aside from the reduction of the synchronization bit string (often longer than the data in the packet) to a string just long enough to contain the start time to the required accuracy, the clock synchronous method offers no additional benefits over the packet synchronous mode. An atomic clock is required at each node for the clock synchronous method.

IMPLEMENTATION OF ENCRYPTION DEVICES

Presupposing the cryptographic algorithms are highly resistant to cryptanalysis it is extremely important to insure that the information to be protected is not compromised in some other fashion. It is possible that a circuit failure could result in plaintext being passed through a failed encryption device without detection. It is possible the electromagnetic radiation caused by the encryption device could radiate the plaintext data.[9,12-Ch. 29]

While status indicators or software checks are available to detect the failure of an applique or main processor cryptographic process, these are not nearly as reliable for preventing accidental information release as using an LSI cryptographic device tied to some other critical circuit such as the main processor instruction sequence controller (CPU master clock) so that if the encryption chip failed, the main processor would stop within one instruction!

There are some applications where it is critical that the processing functions be performed even when the security system has failed. In these circumstances, the strong argument for LSI becomes a liability. Bypassing a failed applique is usually accomplished by a switch action or at most replugging a patchboard, bypassing circuits at the LSI or even card level can be a more difficult problem. However, as LSI availability increases, multiple encryption chips could be used with "hot spares" switched in after a failure.

KEYING METHODS

The autokey (or ciphertext autokey) method was developed for use with polyalphabetic substitution ciphers.[4] For a plaintext alphabet of n characters define n substitution functions (and inverse functions)

$F^1, F^2, \ldots F^n$ and $\bar{\bar{F}}^1, \bar{\bar{F}}^2, \ldots \bar{\bar{F}}^n$ each of which specifies a transform of every character in A into S and vice versa. The notation F_j^i is defined for multiple functions where i specifies which substitution function is to be used and j is the input (sequence number) to the function.

Encryption begins by placing an extra plaintext character which indicates the start of the key at the beginning of the plaintext and enciphered strings ($P_0 \equiv E_0$). The plaintext is then enciphered according to

$$E_i \leftarrow S_j \text{ for } j = F^{P_{i-1}}_{\lfloor P_i \rfloor A} \text{ and } i = 1 \text{ to } n. \qquad (7)$$

where P_{i-1} determines which substitution function F will be used to transform P_i into E_i. The decryption process is

$$P_i \leftarrow A_j \text{ for } j = \bar{\bar{F}}^{P_{i-1}}_{\lfloor E_i \rfloor S} \text{ and } i = 1 \text{ to } n. \qquad (8)$$

To begin the decryption, recall that $P_0 \equiv E_0$.

It appears practical to extend the autokey strategy to transposition and additive encryption methods. Both of these methods may employ a fixed length transform of n characters at a time. Define a key as a set of transforms for either method with one transform for every possible character in the plaintext alphabet A. Then designate one character position in the n character plaintext string as the position controlling the selection of the transform to be used for encrypting the next n character string. This completely defines an autokeying encryption process. Similar to substitution, decryption is controlled by the key selection character position of the most recently decrypted plaintext string.

Some autokey systems have not been popular for computer network applications because of their tendency to propagate communications errors (or attacker induced changes). A transmission error in one position causes the wrong inverse transform to be selected for decrypting the next character. However, good autokey systems can be devised to be self-synchronizing after an error. Error propagation increases the attacker's leverage for denying the use of the communications resources. However, it makes attacks relying upon the acceptance of an altered message almost impossible. The autokey methods proposed above for transposition and additive systems do not perform as well for detecting alteration, as in many applications it would be unacceptable to wait until receipt of the next message before determining that the system has been penetrated. Even this after-the-fact indication would not be present if the attacker were careful to avoid altering the character position controlling selection of the next key. Error propagation would occur for these transforms only for the case where the key selecting character was in error.

DISTRIBUTION OF KEY MATERIAL

The particular key used with a set of crypto-algorithms is usually changed after a stated period of time (the crypto period). This change makes the playback of previous messages difficult and increases the work of the cryptanalysts since the statistics collected on the previous key must now be repeated. Key material is also changed or updated whenever it is suspected that the previous key has been compromised.

There are two principal alternative methods which can be used to distribute the new keying material: (1) the keys can be transmitted over the network, or (2) the keys can be distributed by an independent distribution system.

Manual key distribution systems require extra storage capability for keys, special handling, such as couriers, and require a relatively long time (manual action) to insert the new key. The strength of this method, which is the standard practice today, is that it requires a physical compromise of the key material in order for the encryption system to be broken during key distribution. Distribution of new key material over the network would have the advantage of changing crypto periods at machine (computer and communications) speeds. However, if the network is used to distribute new keys using an old key, compromise of this key potentially compromises all subsequent keys.

EXTENT OF ENCRYPTION

The network design decision to encrypt all messages (encrypt-all) or to encrypt only messages requiring protection (encrypt-select) must be considered by the network architect. Enciphering equipment is expensive, especially for remote terminals that do not handle information that must be protected. This observation has led to the development of several networks where some nodes are interconnected by encrypted communications and other nodes of the network are interconnected by unencrypted communications.[1,12] Another implementation with similar properties is a network where information requiring protection is sent enciphered (ciphertext) and information not requiring protection is sent unencrypted (cleartext). The advantages of these cleartext/ciphertext systems are that (1) low cost, portable nodes can be used where encryption is not necessary and (2) the network can be easily reconfigured for emergency transmission of priority information even if some encryption devices have failed.

The advantages of encrypt-select designs are transitory at best, since LSI encryption devices should soon eliminate the cost/size/power objections to encryption of even the lowest priority remote terminals. More importantly, the increased complexity in message switching software and encryption status checking hardware in the processors far outweighs the ease of reconfiguration achieved. Once total encryption becomes feasible these bimodal networks will probably cease to exist.

SUMMARY

Several network design considerations involving cryptography were discussed in terms of security and cost. In nearly every case each design alternative traded increased security from one form of attack at the expense of increased susceptibility to another attack. The particular choice of a method therefore is left to the designer depending upon the intended application of the network.

REFERENCES

1. Baran, P., *On Distributed Communications: IX. Security, Secrecy, and Tamper-free Considerations*, Doc. RM-3765-RP, Rand Corp., Santa Monica, Calif., August 1964.
2. Burris, H. R., "System Security Planning Guide: Concepts and Projects," *System Security Series No. 1*, Office of the Project Manager Army Tactical Data Systems, Ft. Monmouth, N. J., October 1972.
3. Girsdansky, M. B., "Cryptology, the Computer and Data Privacy," *Computers and Automation*, Vol. 21, No. 4, April 1972, pp. 12-19.
4. Kahn, D., *The Codebreakers*, The Macmillan Company, New York, 1967.
5. Burris, H. R., "Performance Comparison of Authentication and Serialization Message Sender Identity Verification Techniques," *System Security Series No. 4*, Office of the Project Manager Army Tactical Data Systems, Ft. Monmouth, N. J., February, 1973.
6. Carroll, J. M. and P. M. McLelland, "Fast Infinite Key Privacy Transformation for Resource Sharing Systems," *Proc. AFIPS 1970 FJCC*, Vol. 26, AFIPS Press, Montvale, N. J., pp. 223-230.
7. Hoffman, L. J., *Security and Privacy in Computer Systems*, Melville Publishing Company, Los Angeles, Calif., 1973.
8. Petersen, H. E. and R. Turn, "System Implications of Information Privacy," *Proc. AFIPS 1967 SJCC*, Vol. 30, AFIPS Press, Montvale, N. J., pp. 291-300.
9. Turn, R. and N. Z. Shapiro, *Privacy and Security in Databank Systems: Measures of Effectiveness, Costs, and Protector-Intruder Interactions*, Report P-4871, The Rand Corporation, 1972.
10. Skatrud, R. O., "The Applications of Cryptographic Techniques to Data Processing," *Proc. AFIPS 1969 FJCC*, Vol. 34, AFIPS Press, Montvale, N. J., pp. 111-117.
11. *Electronic Instruments and Systems*, Hewlett Packard, 1975.
12. Martin, J., *Security, Accuracy, and Privacy in Computer Systems*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1973.