Introduction to Cryptanalysis and Public Key Cryptography

APPLICATION REPORT: SSSA001

Gene Lin
DSP Application Engineer
Texas Instruments Taiwan Limited

Digital Signal Processing Solutions
November, 1997

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain application using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

Figures

Introduction to Cryptanalysis and Public Key Cryptography

Abstract

Cryptanalysis is the process of finding a shortcut method not envisioned by the designer for decrypting an enciphered message when the key used to encrypt the message is not known.

This report discusses three methods of cryptanalysis:

- Chosen-plaintext attack
- Known-plaintext attack
- Ciphertext-only attack.

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.

This report introduces two types of public key cryptography:

- Diffie-Hellman Exchange Algorithm
- Rivest-Shamir-Adleman (RSA) Algorithm

The report also introduces two conventional cryptosystems:

- Vernam cipher
- US Data Encryption Standard (DES)

Product Support

World Wide Web

Our World Wide Web site at www.ti.com contains the most up to date product information, revisions, and additions. Users registering with TI&ME can build custom information pages and receive new product updates automatically via email.

Email

For technical issues or clarification on switching products, please send a detailed email to dsph@ti.com. Questions receive prompt attention and are usually answered within one business day.

Introduction

Transforming messages into unintelligible data by means of an encryption algorithm is known as *encrypting* or *enciphering*. The algorithm produces an output that can only be understood by someone who knows how to reverse the transformation, known as *decrypting* or *deciphering*.

In the cryptographic community, the original message is commonly called *plaintext* and the output of the encryption algorithm is called *ciphertext*. Because the encryption algorithm is often well known, additional secret information, known as the *key*, is used to change the algorithm in some specific manner. Knowledge of the ciphertext and key enables the intended recipient to decrypt the message.

Cryptanalysis

Cryptanalysis is the process of finding a short-cut method, not envisioned by the designer, for decrypting an enciphered message when the key used to encrypt the message is not known.

There are many complex encrypting methods, and there is always someone cracking them. This introduces the following methods of cryptanalysis:

- Substitution cipher
- Chosen-plaintext attack
- Known-plaintext attack
- Ciphertext-only attack.

Substitution Cipher Algorithm

The *substitution cipher* is among the simplest of cryptosystems. Though no longer used to protect data, this cipher effectively demonstrates the basic principles of cryptanalysis. Encryption involves substituting each plaintext character with another character to produce ciphertext. Usually, the character sets used for plaintext and ciphertext are the same, so that the encrypted message can be transmitted using the same channel or medium over which the plaintext is transmitted. Here is an example of the substitution cipher.

The cipher is:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

S N P H U A T V B K W J Z Y M Q G X F C R O E I L D
```

The original plaintext is:

This is a test.
Cryptanalysis is defined as the process of attempting to find a shortcut method, not envisioned by the designer.

After the substitution cipher, the encrypted ciphertext is:

Cvbf bf s cufc.
Pxlqcsysjlfbf bf huabyuh sf cvu qxmpuff ma sccuzqcbyt cm abyh s fvmxcprc zucvmh, ymc uyobfbmyuh nl cvu hufbtyux.

Chosen-Plaintext Attack

The *chosen-plaintext attack* allows a choice of which messages the cryptanalyst can encrypt under the unknown secret key. Any message the attacker considers necessary may be enciphered with some unknown key. The attacker then compares the plaintext with the ciphertext and searches for patterns that may lead to determining the mechanics of the algorithm. Given this process, the obvious choice when attacking a substitution cipher is to input whatever alphabet the algorithm uses as the initial plaintext. In English, we would encrypt *ABCD .... XYZ*, which would result in the output of ciphertext that is the translation table, as shown below:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
(Input)

S N P H U A T V B K W J Z Y M Q G X F C R O E I L D
(Output)
```

This translation table enables the decryption of any received ciphertext encrypted by this cryptosystem.

Known-Plaintext Attack

The *known-plaintext attack* limits the messages to whatever plaintext and corresponding ciphertext the cryptanalyst can gather. This approach is more restrictive than the chosen-plaintext attack. The cryptanalyst is not allowed to encipher just any message. Instead, a set of plaintext messages, and their corresponding ciphertext, is made available to the cryptanalyst.

Given the following short plaintext message and its associated ciphertext, the mechanics of a substitution cipher attack are quite simple:

The plaintext:

This is a test.
Cryptanalysis is defined as the process of attempting to find a shortcut method, not envisioned by the designer.

The encrypted ciphertext:

Cvbf bf s cufc.
Pxlqcsysjlfbf bf huabyuh sf cvu qxmpuff ma sccuzqcbyt cm abyh s fvmxcprc zucvmh, ymc uyobfbmyuh nl cvu hufbtyux.

The cryptanalyst constructs the translation table by determining which characters have been translated to other character. The resulting translation table is composed as follows:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
(Input)

S N P H U A T V B _ _ J Z Y M Q _ X F C R O _ _ L _
(Output)
```

It is highly likely that any ciphertext encrypted using this cryptosystem can be decrypted, because the remaining *mystery letters* (J, K, Q, W, X, and Z) are not widely used in English. In actually, the remainder of the translation table will probably be determined by decrypting future messages.

Ciphertext-Only Attack

The *cipher-only attack* requires the least difficult form of data collection: there is no plaintext available to the cryptanalyst. This attack is also the most restrictive because no plaintext in available.

The classic ciphertext-only attack on a substitution cipher is *frequency analysis* of the ciphertext. This process involves counting how many times certain patterns occur and comparing the frequency of those occurrences to the frequency of known patterns in English. Figure 1 shows the frequency distribution of characters in English. Figure 2 shows the frequency distribution of cybertext.

Figure 1. English Character Frequency Distribution

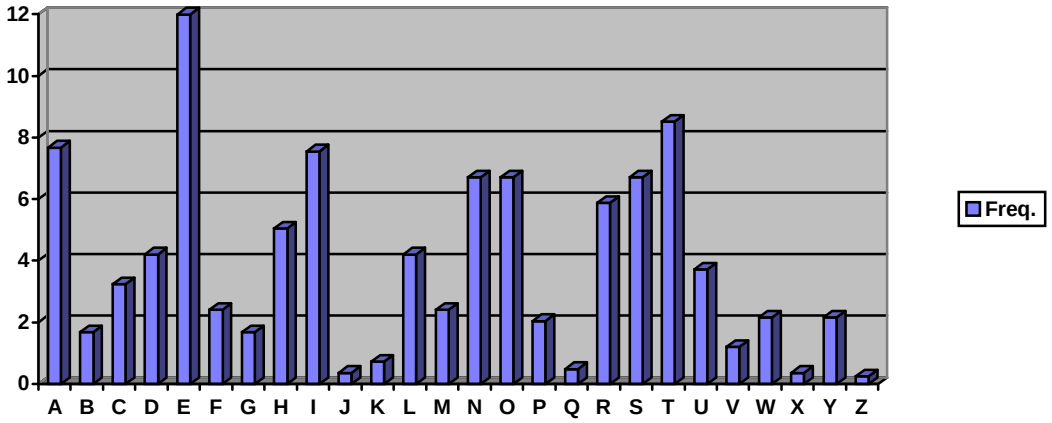
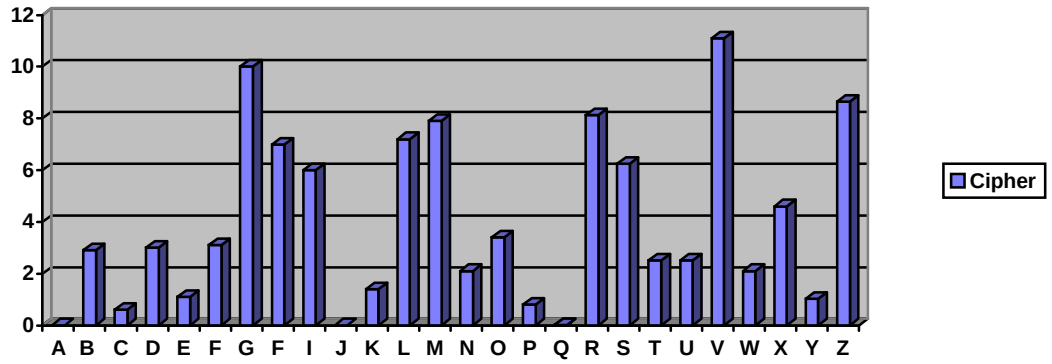


Figure 2. Ciphertext Frequency Distribution



Assume that G or V in the ciphertext translates to E in plaintext. This is because E is the most popular letter in English, and the other two letters form a significant fraction of the ciphertext. It is also likely that whichever letter does not translate to E translates to T, because T is the second most popular in English.

An sample program could count the frequency distribution of English-alphabet letters and use the cipher-only attack to crack the ciphertext. The program also provides the manual changing translation table function to decrypt the ciphertext more correctly.

The ciphertext is:

Qrnjbp wul pxlqcmtxsqvl, huoujmquh ebcvby cvu jsfc cem hupshuf amx pmzzuxpbsj rfu, vsf nupmzu sy uipbcbyt cupvymjmtl amx cvu pmzzrybpscbmyf byhrfcxl, sf ebxuh syh ebxujuff ombpu, hscs, syh bzstu yucemxwf pmycbyru cm qxmjbauxscu syh bycuxpmyyupc. Qrnjbp wul pxlqcmtxsqvl vujqf qxmobhu fuprxu pmzzrybpscbmyf amx tuyuxsj pmzzrybpscbmyf.

After running the program to decrypt, the sample reads:

Mublic veh crhmtogramyh, dekelomed wityin tye last two decades for coppercial use, yas becope an exciting tecynologh for tye coppunications industrh, as wired and wireless koice, data, and ipage networvs continue to mroliferate and interconnect. Mublic veh crhmtogramyh yelms mrokide secure coppunications for general coppunications.

After this first pass at decryption, the program changes the next set of characters, such as y and m. After this pass, the decrypted paragraphs become:

Public vey cryptography, dekeloped within the last two decades for commercial use, has become an exciting technology for the communications industry, as wired and wireless koice, data, and image networvs continue to proliferate and interconnect. Public vey cryptography helps prokide secure communications for general communications.

The final unencrypted paragraph is:

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 for general communications.

A comparison of these two paragraphs shows they are almost the same. It is possible to use the ciphertext-only attack to decrypt the ciphertext, and carefully correct every character.

Public Key Cryptography

The term cryptography may suggest coded messages of governments that enemy spies are constantly trying to crack. This is just one kind of cryptograph. Another is encrypted private line telephones provide secure communications.

The current growth of public or private wired and wireless data networks, the proliferation of wireless voice and data terminals, and the security of both wired and wireless communication are becoming prominent issues in personal or business communications.

Public key cryptography provides secure information for civilian business or individual systems easier and cheaper than the traditional methods used by military and diplomatic agencies.

This section introduce three public key cryptosystems:

- Diffie-Hellman Exchange Algorithm
- Rivest-Shamir-Adleman (RSA) Algorithm
- Digital Signature

The section also introduces two conventional cryptosystems:

- Vernam cipher
- US Data Encryption Standard (DES)

Diffle-Hellman Key Exchange

Public key cryptography was first invented in the 1970s by Whit Diffie, Ralph Merkle, and Martin Hellman at Stanford University and was named *Diffe-Hellman Key Exchange Method*.

A and **B** want to communicate securely. They agree on a large prime *p* and an integer *g*, which can be transmitted publicly. This means that *p* and *g* may be the same for many people to transmit. **A** then chooses an integer *a, 0 < a < p-1,* and **B** chooses an integer *b, 0 < b < p-1,* too, then **A** computes *I* and **B** computes *J*:

$$I = g^a \pmod{p}$$

$$J = g^b \pmod{p}$$

A then transmits *I* to **B** and **B** transmits *J* to **A**. After that, **A** receives *J* and computes *X*:

$$X = J^a \pmod{p}$$

B receives *I* computes *Y:*

$$Y = I^b \pmod{P}$$

It is easy to see that *X=Y*. Both **A** and **B** can use *X* or *Y* to derive a key for a conventional cryptosystem. If there is an eavesdropper who gets both *I* and *J*, it is very difficult for him to derive *X* or *Y* from *I* and *J* without *a* or *b*. Thus, this protocol is widely used.

RSA Cryptosystem

The most famous current public key cryptosystem is the *RSA algorithm*, invented by Ron Rivest, Adi Shamir, and Len Adleman, three professors of Massachusetts Institute of Technology.

The RSA Cryptosystem relies for its security on the difficulty of factoring an integer into primes. If **A** wants secret messages sent to him, he first choose two primes *p* and *q*, and let $n = p \bullet q$. Next, he chooses another integer *e*, $1 < e < n$, such that *e* has no integer divisors greater than *1* in common with either *p-1* or *q-1*. He publishes the pair *(n, e)* as his public key, and keeps *p* and *q* secret.

B wants to send a message to **A**. He first transforms the message into blocks of integers, each with no more than *log2n* bits. If a particular block is regarded as the binary representation of an integer *m*, $0 \le m < n$, then he computes:

$$c = m^e \pmod{n}$$

Using the same consecutive squaring method as in the Diffe-Hellman method, **B** then transmits *c* to **A**.

To decrypt the transmitted message *c*, **A** uses a procedure like the encrypting one. First he tries to find an integer *d*, that *d* satisfies:

$$ed=1 \pmod{p-1}$$

and

$$ed=1 \pmod{q-1}$$

Then **A** computes the integer *m* by

$$m = c^d \pmod{n}$$

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

Digital Signature System

Applying the RSA cryptosystem helps develop a *Digital Signature System.* Suppose **A**'s public key is (*n , e*). To sign a message *m*, when *m* is an integer of the range $0 \leq m < n$. **A** attaches it to the integer *x*, which *x* is:

$$x = m^d \pmod{n}$$

Here, *d* is **A**'s secret decoding exponent. **A** sends *x* to someone who wants to verify the signature. Only **A** knows *d*, and anyone who wants to verify **A**'s signature of *m*, only needs to computes the integer *y*

$$y = m^d \pmod{n}$$

The property of this system ensures that, if the signature is true, *y* will be definitely equal to *m* since (*n , e*) is known. The system should work without problems.

Vernam Cipher System

A conventional Cryptosystem can provide security, but at substantial cost. In 1917, the *Vernam cipher* was invented at AT&T. If **A** and **B** want to communicate secretly with this cipher, they agree ahead of time on a string of randomly generated encryption or key bits:

k1, k2, k3, k4, ... etc.

which will be used to encipher message.

If **A** wants to convey to **B** a message such as:

m1, m2, m3, m4, ...etc.

He uses the randomly generated key bits to encrypt the message before transmitting:

e1 = m1 + k1 (mod 2)

e2 = m2 + k2 (mod 2)

e3 = m3 + k3 (mod 2)

e4 = m4 + k4 (mod 2) . . . . . .etc.

When **B** receives the *ei* bits, he decrypts them:

m1 = e1 - k1 (mod 2)

m2 = e2 - k2 (mod 2)

m3 = e3 - k3 (mod 2)

m4 = e4 - k4 (mod 2) . . . . . .etc.

B decrypts the message correctly. If the key bits *k1, k2* ... are truly random and never used more than once, and if they can be created and conveyed securely, the Vernam cipher will be unbreakable, but it is also expensive . The Washington-to-Moscow hot line is one example using this method. Such a system usually is not appropriate for civilian use because of the huge volume of transmitted information.

DES Key System

To save cost, the US Data Encryption Standard uses a 56-bit key, which is called a *single 56-bit DES key.* This means there are only 2 power 56 possible keys. A computer finds the exact key quickly and inexpensively. For many applications, this level of security is not adequate. There is an enhanced system, the *triple-DES*, which consists of three encryptions of the basic DES and is controlled by two keys, for an effective key size of 112 bits. This system has more possible keys and is more expensive to crack.

Conventional Cryptosystem Disadvantage

There is an additional problem for either DES or Vernam cipher. When *n* people or computers communicate with each other, the number of necessary keys is *n(n-1)/2*. Since there are already more than 20 million users in the Internet network, any two users communicating in the Internet securely requires maybe 200 trillion keys to be available. This is a huge cost and the major disadvantage of conventional cryptosystems,. Reducing this cost is the main reason for the invention of public key cryptography.

Conclusion

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 do not appear to be good candidates for encrypting general traffic, and are often unnecessary within networks that have a highly secure and trusted central database.