Attacks Against Ciphers

When decrypting a cryptosystem, the attacker can be interested in obtaining the plaintext for a given ciphertext or in detecting the key used to produce the ciphertext. As a key may have been used to encode many messages, detecting the key, in general, is the more rewarding activity.

Cryptanalytic attack can be divided into the following categories:

- ***Exhaustive search***. The most obvious generic attack is the *exhaustive key search or brute force attack* which simply involves trying all possible keys. The difficulty of the attack corresponds to the size of the key space; thus, a very large key space can make this attack unfeasible. This attack was implemented using machines starting from the 1930s, such as in the cryptanalysis of the Enigma systems.
- ***Ciphertext only attack.*** The attacker only knows the ciphertext from which the plaintext or key is to be obtained. The difficulty of this attack is based on the redundancy present in the ciphertext and the available ciphertext length. This type of attack will yield no information about the plaintext (except its length), in cases where no redundancy exists in the ciphertext. Decryption methods of this type make heavy use of the source language statistics and often involve a guess of the likely parts of plaintext.
- ***Known plaintext attack.*** In this case, the attacker knows some plaintext with its corresponding ciphertext. This information, for many classical ciphers, allows the key to be trivially detected. This helps in reading other messages enciphered using the same or similar keys.
- ***Chosen plaintext attack.*** With this method, the attacker can choose his own set of plaintexts to be encoded. This enables the attacker to select a particular plaintext or plaintexts which might increase the opportunity of determining the key.
- ***Chosen ciphertext attack***. Here the attacker is able to choose ciphertexts to be decrypted in order to obtain their corresponding plaintexts. This attack is used only when the cryptanalyst wants to determine the key being used. Public key ciphers are often vulnerable to this form of attack.
- ***Chosen key (related-key) attack***. In this case, the attacker knows the key in advance. During the design of a cipher, the effect of different keys on the same plaintext is investigated in this form of attack.

When evaluating the security of modern cryptosystems, it is usual to assume that a known plaintext attack and a chosen plaintext attack (in most cases) are feasible. Security always depends on several different factors while the importance of the message to be transmitted will determine the precautions to be taken. The strength of a cryptosystem is a negative quality in that security relies on the inability of attackers to find a feasible way to break it. The best way to prove the difficulty of breaking a cryptosystem is to show that its decryption operation is equivalent to solving some generally agreed computational problems that do not have a polynomial time solution.

Stream and Block Ciphers

One final characterization of encryption algorithms relates to the nature of the data to be concealed. Suppose you are streaming video, perhaps a movie, from a satellite. The stream may come in bursts, depending on such things as the load on the satellite and the speed at which the sender and receiver can operate. For such application you may use what is called ***stream encryption***, in which each bit, or perhaps each byte, of the data stream is encrypted separately. A model of stream enciphering is shown in Figure 3. Notice that the input symbols are transformed one at a time. The advantage of a stream cipher is that it can be applied immediately to whatever data items are ready to transmit. But most encryption algorithms involve complex transformations; to do these transformations on one or a few bits at a time is expensive.

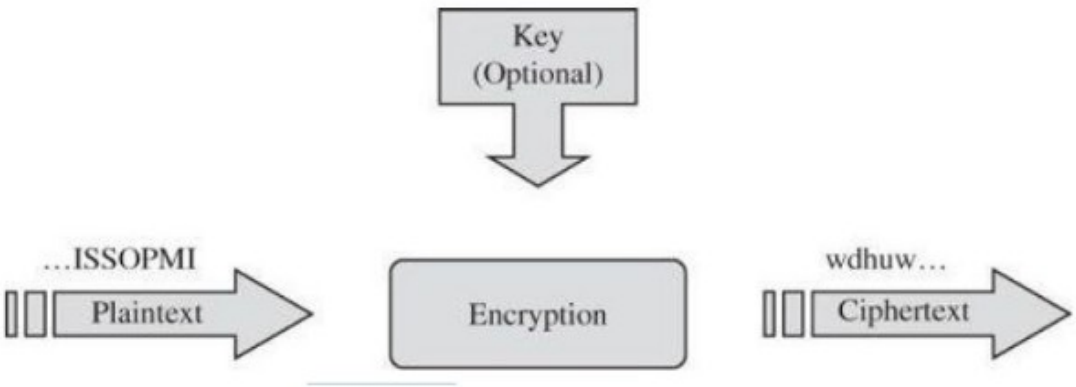

Figure 3: Stream Enciphering

To address this problem and make it harder for a cryptanalyst to break the code, we can use block ciphers. A block cipher encrypts a group of plaintext symbols as a

single block. A block cipher algorithm performs its work on a quantity of plaintext data all at once. These type of algorithms capitalize on economies of scale by operating on large amounts of data at once.
Blocks for such algorithms are typically 64, 128, 256 bits or more. The block size need not have any particular relationship to the size of a character. Block ciphers work on blocks of plaintext and produce blocks of ciphertext, as shown in Figure 2-21. In the figure, the central box represents an encryption machine: The previous plaintext pair is converted to po, the current one being converted is IH, and the machine is soon to convert ES.

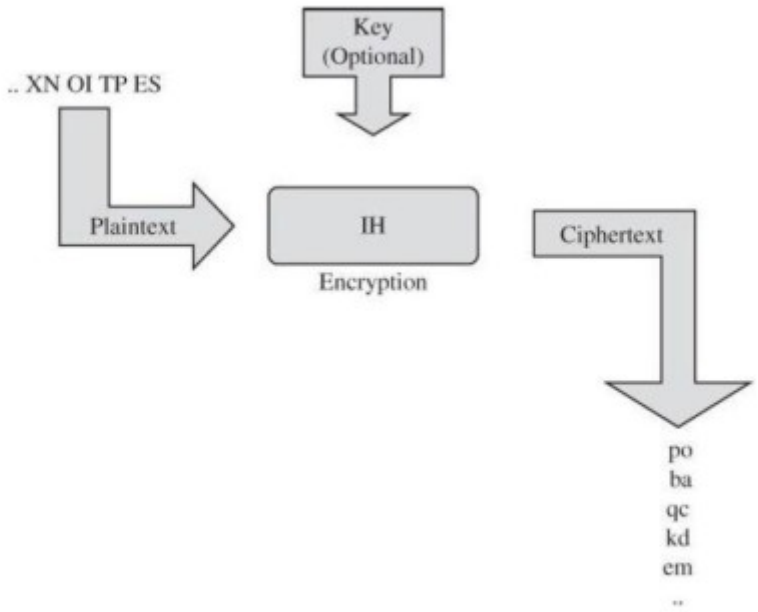

Figure 4: Block Cipher

Stream ciphers encrypt one bit or one byte at a time; block ciphers encrypt a fixed number of bits as a single chunk.

Stream Ciphers

Advantages:

- Speed of transformation: algorithms are linear in time and constant in space. Each symbol can be encrypted as soon as it read.
- Low error propagation: an error in encrypting one symbol likely will not affect subsequent symbols.

Disadvantages:

- Low diffusion: all information of a plaintext symbol is contained in a single ciphertext symbol. Each symbol is separately encrypted. Therefore all the information of that symbol is contained in one symbol of ciphertext.
- Susceptibility to insertions/ modifications: an active interceptor who breaks the algorithm might insert spurious text that looks authentic.

Block Ciphers

Advantages:

- High diffusion: information from one plaintext symbol is diffused into several ciphertext symbols.
- Immunity to tampering: difficult to insert symbols without detection. Because the length of the block would then be incorrect.

Disadvantages:

- Slowness of encryption: an entire block must be accumulated before encryption / decryption can begin.
- Error propagation: An error in one symbol may corrupt the entire block.
- Padding: a final short block must be filled with irrelevant data to make a full size block.

Representing Characters

We want to study ways of encrypting any computer material, whether it is written as ASCII characters, binary data, object code, or a control stream. However, to simplify the explanations, we begin with the encryption of messages written in the standard 26- letter English alphabet, A through Z.

In these lectures, we switch back and forth between letters and the numeric encoding of each letter as shown here.

Letter	A	B	C	D	E	F	G	H	I	J	K	L	M
Code	0	1	2	3	4	5	6	7	8	9	10	11	12
Letter	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Code	13	14	15	16	17	18	19	20	21	22	23	24	25

Thus, the letter A is represented by a zero, B by a one, and so on. This representation allows us to consider performing arithmetic on the "letters" of a message. That is, we can perform addition and subtraction on letters by adding and subtracting the corresponding code numbers. Expressions such as A + 3 = D or K - 1 = J have their natural interpretation. Arithmetic is performed as if the alphabetic table were circular. In other words, addition wraps around from one end of the table to theo ther so that Y + 3 = B. Thus, every result of an arithmetic operation is between 0 and 25.

There are many types of encryption. In the next two sections we look at two simple forms of encryption: substitutions, in which one letter is exchanged for another, and transpositions, in which the order of the letters is rearranged. The goals of studying these two forms are to become familiar with the concept of encryption and decryption, to learn some of the terminology and methods of cryptanalysis, and to study some of the weaknesses to which encryption is prone. Once we have mastered the simple encryption algorithms, we explore "commercial grade" algorithms used in modern computer applications.

Classical Ciphers

This section provides some examples of the classical cryptosystems. These examples illustrate different types of encoding.

Simple Substitution Ciphers

A simple substitution cipher (also called a monoalphabetic cipher) replaces each character in the plaintext with another predetermined character to form the ciphertext. Formally, let P be a plaintext alphabetic character of size n, where P in {p_0, p_1, . . . , p_{n-1}} and C is a ciphertext alphabetic character of size n, C in {f(p_0), f (p_1),...,f(p_{n-1})}. Each symbol of P has a one-to-one mapping to the corresponding symbol of C, f: P ----> C. In this section, we study several kinds of substitution ciphers.

The Caesar Cipher

The Caesar cipher has an important place in history. Julius Caesar is said to have been the first to use this scheme, in which each letter is translated to the letter a fixed number of places after it in the alphabet. Caesar used a shift of 3, so plaintext letter pi was enciphered as ciphertext letter ci by the rule

$$c_i = E(p_i) = p_i + 3$$

A full translation chart of the Caesar cipher is shown here.

```
Plaintext:  ABCDEFGHIJKLMNOPQRSTUVWXYZ
Ciphertext: defghijklmnopqrstuvwxyzabc
```

Using this encryption method, the message “CRYPTOGRAPHY DEMO”
Would be encrypted as:

```
CRYPTOGRAPHY DEMO
fubswrjudskb ghpr
```

This type of cryptosystem is easy to implement and to use. However, it is not difficult to crack, as it does nothing to conceal the statistical properties of the language. Hence, it does not provide much security and can be easily broken by frequency distribution analysis. However, during Caesar's lifetime, the simplicity did not dramatically compromise the safety of the encryption because anything written, even in plaintext, was rather well protected; A sender in the field could write out a plaintext and a ciphertext alphabet, encode a message to be sent, and then destroy the paper containing the alphabets. The following Sidebar describes actual use of a cipher similar to the Caesar cipher.

Sidebar . : Mafia Boss Uses Encryption

Arrested in Sicily in April 2006, the reputed head of an Italian Mafia family, Bernardo Provenzano, made notes, pizzini in the Sicilian dialect. When arrested, he left approximately 350 of the notes behind. In the pizzini he gives instructions to his lieutenants regarding particular people.

Instead of writing the name of a person, Provenzano used a variation of the Caesar cipher in which letters were replaced by numbers: A by 4, B by 5, … Z by 24 (there are only 21 letters in the Italian alphabet). So in one of his notes the string "…I met 512151522 191212154 and we agreed that we will see each other after the holidays…," refers to Binnu Riina, an associate arrested soon after Provenzano [LOR06]. Police decrypted notes found before Provenzano's arrest and used clues in them to find the boss, wanted for 40 years.

All notes appear to use the same encryption, making them trivial to decrypt once police discerned the pattern.

Suggestions we might make to Sig. Provenzano: use a strong encryption algorithm, change the encryption key from time to time, and hire a cryptographer.

Advantages of using a Caesar cipher include:

- One of the easiest methods to use in cryptography and can provide minimum security to the information
- Use of only a short key in the entire process
- One of the best methods to use if the system cannot use any complicated coding techniques
- Requires few computing resources

Disadvantages of using a Caesar cipher include:

- Simple structure usage
- Can only provide minimum security to the information
- Frequency of the letter pattern provides a big clue in deciphering the entire message

Cryptanalysis of the Caesar Cipher

Let us take a closer look at the result of applying Caesar's encryption technique to "CRYPTOGRAPHY DEMO", If we did not know the plaintext and were trying to guess it, we would have many clues from the ciphertext. For example, the break between the two words is preserved in the ciphertext. We might also notice that when a letter is repeated, it maps again to the same ciphertext as it did previously. So the letters R, Y, P and O always translate to u, b, s and r. These clues make this cipher easy to break.

By using frequency analysis of individual letters, the cryptanalyst can readily decrypt a ciphertext manually if it uses this cryptosystem. With a few attempts and after trying some of the possibilities, the cryptanalyst will be able to find the correct substitution. Digram and trigram distributions provide more useful information that can also be accessed by the cryptanalyst. Many digrams could occur more frequently than some single letters while other digrams such as ‘qj’ rarely occur in English. Typically, different languages have different letter frequencies. Thus, it is possible to determine the plaintext language before starting to decrypt the ciphertext if the ciphertext provided is of sufficient length.

Suppose you are given the following ciphertext message, and you want to try to determine the original plaintext.

wklv phvvdjh lv qrw wrr kdug wr euhdn

The message has actually been enciphered with a 27-symbol alphabet: A through Z plus the "blank" character or separator between words. As a start, assume that the coder was lazy and has allowed the blank to be translated to itself. If your assumption is true, it is an exceptional piece of information; knowing where the spaces are allows us to see which are the small words. English has relatively few small words, such as am, is, to, be, he, we, and, are, you, she, and so on. Therefore, one way to attack this problem and break the encryption is to substitute known short words at appropriate places in the ciphertext until you have something that seems to be meaningful. Once the small words fall into place, you can try substituting for matching characters at other places in the ciphertext.

In fact, in most encryption schemes, spaces between words often are deleted, under the assumption that a legitimate receiver can break most messages into words fairly easily. For ease of writing and decoding, messages are then arbitrarily broken into blocks of a uniform size, such as every five characters, so that there is no significance to the places where the message is broken.

Look again at the ciphertext you are decrypting. There is a strong clue in the repeated r of the word wrr. You might use this text to guess at three-letter words that you know. For instance, two very common three-letter words having the pattern xyy are see and too; other less common possibilities are add, odd, and off. (Of course, there are also obscure possibilities like woo or gee, but it makes more sense to try the common cases first.) Moreover, the combination wr appears in the ciphertext, too, so you can determine whether the first two letters of the three-letter word also form a two-letter word.

For instance, if wrr is SEE, wr would have to be SE, which is unlikely. However, if would be TO, which is quite reasonable. Substituting T for w and O for r, the message becomes

```
wklv phvvdjh lv qrw wrr kdug wr euhdn
T--- ------- -- -OT TOO ---- TO -----
```

The OT could be cot, dot, got, hot, lot, not, pot, rot, or tot; a likely choice is not. Unfortunately, q = N does not give any more clues because q appears only once in this sample.

The word lv is also the end of the word wklv, which probably starts with T. Likely two-letter words that can also end a longer word include so, is, in, etc. However, so is unlikely because the form T-SO is not recognizable; IN is ruled out because of the previous assumption that q is N. A more promising alternative is to substitute IS for lv throughout, and continue to analyze the message in that way.

By now, you might notice that the ciphertext letters uncovered are just three positions away from their plaintext counterparts. You (and any experienced cryptanalyst) might try that same pattern on all the unmatched ciphertext. The completion of this decryption is left as an exercise.
The cryptanalysis described here is ad hoc, using deduction based on guesses instead of solid principles. But you can take a more methodical approach, considering which letters commonly start words, which letters commonly end words, and which prefixes and suffixes are common. Cryptanalysts have compiled lists of common prefixes,common suffixes, and words having particular patterns. (For example, sleeps is a word that follows the pattern abccda.)

Other Substitutions

In substitutions, the alphabet is scrambled, and each plaintext letter maps to a unique ciphertext letter. We can describe this technique in a more mathematical way. Formally, we say that a permutation is a reordering of the elements of a sequence. For instance, we can permute the numbers 1 to 10 in many ways, including the permutations $\pi 1 = 1, 3, 5, 7, 9, 10, 8, 6, 4, 2$; and $\pi 2 = 10, 9, 8, 7, 6, 5, 4, 3, 2, 1$. A permutation is a function, so we can write expressions such as $\pi 1(3) = 5$ meaning that the letter in position 3 is to be replaced by the fifth letter. If the set is the first ten letters of the alphabet, $\pi 1(3) = 5$ means that c is transformed into E.
One way to scramble an alphabet is to use a key, a word that controls the permutation.
For instance, if the key is word, the sender or receiver first writes the alphabet and then writes the key under the first few letters of the alphabet.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
word
```

The sender or receiver then fills in the remaining letters of the alphabet, in some easy-to-remember order, after the keyword.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
wordabcefghijklmnpqstuvxyz
```

In this example, the key is short, so most plaintext letters are only one or two positions off from their ciphertext equivalents. With a longer keyword, the distance is greater and less predictable, as shown below. Because π must map one plaintext letter

to exactly one ciphertext letter, duplicate letters in a keyword, such as the second s and o in professional, are dropped.

ABCDEFGHIJKLMNOPQRSTUVWXYZ
profesinalbcdghjkmqtuvwxyz

Notice that near the end of the alphabet replacements are rather close, and the last seven characters map to themselves. Conveniently, the last characters of the alphabet are among the least frequently used, so this vulnerability would give little help to an interceptor.
Still, since regularity helps an interceptor, it is desirable to have a less regular rearrangement of the letters. One possibility is to count by threes (or fives or sevens or nines) and rearrange the letters in that order. For example, one encryption uses a table that starts with
ABCDEFGHIJKLMNOPQRSTUVWXYZ
adgj

using every third letter. At the end of the alphabet, the pattern continues mod 26, as shown below.
ABCDEFGHIJKLMNOPQRSTUVWXYZ
adgjmpsvybehknqtwzcfilorux

There are many other examples of substitution ciphers. For instance, the following Sidebar describes a substitution cipher called a poem code, used in the early days of World War II by British spies to keep the Germans from reading their messages.

Sidebar : Poem Codes

During World War II, the British Special Operations Executive (SOE) produced codes to be used by spies in hostile territory. The SOE devised poem codes for use in encrypting and decrypting messages. For security reasons, each message had to be at least 200 letters long.

To encode a message, an agent chose five words at random from his or her poem, and then assigned a number to each letter of these words. The numbers were the basis for the encryption. To let the Home Station know which five words were chosen, the words were inserted at the beginning of the message. However, using familiar poems created a huge vulnerability. For example, if the German agents knew the British national anthem, then they might guess the poem from fewer than five words. As Marks explains, if the words included "'our,' 'gracious,' 'him,' 'victorious,' 'send,' then God save the agent" [MAR98].

For this reason, Leo Marks' job at SOE was to devise original poems so that "no reference books would be of the slightest help" in tracing the poems and the messages.

Complexity of Substitution Encryption and Decryption

An important issue in using any cryptosystem is the time it takes to turn plaintext into ciphertext, and vice versa. Especially in the field (when encryption is used by spies or decryption is attempted by soldiers), it is essential that the scrambling and unscrambling not deter the authorized parties from completing their missions. The timing is directly related to the complexity of the encryption algorithm. For example, encryption and decryption with substitution ciphers can be performed by direct lookup in a table illustrating the correspondence, like the ones shown in our examples.
Transforming a single character can be done in a constant amount of time, so we express the complexity of the algorithm by saying that the time to encrypt a message of n characters is proportional to n. One way of thinking of this expression is that if one message is twice as long as another, it will take twice as long to encrypt.

Cryptanalysis of Substitution Ciphers

The techniques described for breaking the Caesar cipher can also be used on other substitution ciphers. Short words, words with repeated patterns, and common initial and final letters all give clues for guessing the permutation.
Of course, breaking the code is a lot like working a crossword puzzle: You try a guess and continue to work to substantiate that guess until you have all the words in place or until you reach a contradiction. For a long message, this process can be extremely tedious. Fortunately, there are other approaches to breaking an encryption. In fact, analysts apply every technique at their disposal, using a combination of guess, strategy, and mathematical skill.

Cryptanalysts may attempt to decipher a particular message at hand, or they may try to determine the encryption algorithm that generated the ciphertext in the first place (so that future messages can be broken easily).

To see why, consider the difficulty of breaking a substitution cipher. At face value, such encryption techniques seem secure because there are 26a possible different encipherments. We know this because we have 26 choices of letter to substitute for the a, (all but the then 25 (all but the one chosen for a) for b, 24 ones chosen for a and b) for c, and so on, to yield 26 * 25 * 24 *...* 2 * 1 = 26a possibilities. By using a brute force attack, the cryptanalyst could try all 26a permutations of a particular ciphertext message. Working at one permutation per microsecond (assuming the

cryptanalyst had the patience to review the probable-looking plaintexts produced by some of the permutations), it would still take over a thousand years to test all 26a possibilities.
We can use our knowledge of language to simplify this problem. For example, in English, some letters are used more often than others. The letters E, T, O, and A occur far more often than J, Q, X, and Z, for example. Thus, the frequency with which certain letters are used can help us to break the code more quickly. We can also recognize that the nature and context of the text being analyzed affect the distribution. For instance, in a medical article in which the term x-ray was used often, the letter x would have an uncommonly high frequency. When messages are long enough, the frequency distribution analysis quickly betrays many of the letters of the plaintext.

How difficult is it to break substitutions? With a little help from frequency distributions and letter patterns, you can probably break a substitution cipher by hand. It follows that, with the aid of computer programs and with an adequate amount of ciphertext, a good cryptanalyst can break such a cipher in an hour. Even an untrained but diligent interceptor could probably determine the plaintext in a day or so. Nevertheless, in some applications, the prospect of one day's effort, or even the appearance of a sheet full of text that makes no sense, may be enough to protect the message. Encryption, even in a simple form, will deter the casual observer.

Despite the fact that simple substitution ciphers are not typically used today in real-world encoding systems, many effective and secure modern ciphers use substitution ciphers in combination with other ciphers, for example, transposition ciphers, modular arithmetic, Boolean algebra and so on. This powerful combination is an important innovation as it results in a method that is stronger than its original components. Although monoalphabetic ciphers are not considered secure, they are frequently used as building blocks in larger state-of-the-art cryptographic systems. Consequently, it is very important to understand the vulnerability of these simple ciphering systems, to help with building more complex ciphers.

Classical ciphers generally fall into two main categories: substitution ciphers and transposition ciphers. Modern encryption systems have now superseded the classical systems; however, the cryptanalysis of classical ciphers remains the most popular cryptological application and implementation for metaheuristic search research. The essential concepts of substitution ciphers and transposition ciphers are still widely used today in the Advanced Encryption Standard (AES) and the International Data

Encryption Algorithm (IDEA). As long as the operations and concepts of classical cipher systems are the basic building blocks of more secure modern ciphers, then classical ciphers will typically be the first ciphers considered in the case of investigating and examining new attacks.