PEARSON IT CYBERSECURITY CURRICULUM

THIRD EDITION

NETWORK DEFENSE AND COUNTERMEASURES

Principles and Practices

CHUCK EASTTOM

Network Defense and Countermeasures

Principles and Practices

Third Edition

Chuck Easttom

Pearson

800 East 96th Street, Indianapolis, Indiana 46240 USA

and you choose to shift by two letters, then the message becomes

```
C ecv
```

Or, if you choose to shift by three letters, it becomes

```
D fdw
```

In this example, you can choose any shifting pattern you want. You can shift either to the right or left by any number of spaces you like. Because this is a simple method to understand, it makes a good place to start your study of encryption. It is, however, extremely easy to crack. You see, any language has a certain letter and word frequency, meaning that some letters are used more frequently than others. In the English language, the most common single-letter word is *a*. The most common three-letter word is *the*. Knowing these two characteristics alone could help you decrypt a Caesar cipher. For example, if you saw a string of seemingly nonsense letters and noticed that a three-letter word was frequently repeated in the message, you might easily surmise that this word was *the*—and the odds are highly in favor of this being correct.

Furthermore, if you frequently noticed a single-letter word in the text, it is most likely the letter *a*. You now have found the substitution scheme for *a*, *t*, *h*, and *e*. You can now either translate all of those letters in the message and attempt to surmise the rest or simply analyze the substitute letters used for *a*, *t*, *h*, and *e* and derive the substitution cipher that was used for this message. Decrypting a message of this type does not even require a computer. Someone with no background in cryptography could do it in less than ten minutes using pen and paper.

Caesar ciphers belong to a class of encryption algorithms known as substitution ciphers. The name derives from the fact that each character in the unencrypted message is substituted by one character in the encrypted text. The particular substitution scheme used (for example, 12 or 11) in a Caesar cipher is called a substitution alphabet (that is, *b* substitutes for *a*, *u* substitutes for *t*, etc.). Because one letter always substitutes for one other letter, the Caesar cipher is sometimes called a mono-alphabet substitution method, meaning that it uses a single substitution for the encryption.

The Caesar cipher, like all historical ciphers, is simply too weak for modern use. It is presented here just to help you understand the concepts of cryptography.

ROT 13

ROT 13 is another single alphabet substitution cipher. All characters are rotated 13 characters through the alphabet.

The phrase

```
A CAT
```

becomes

```
N PNG
```

ROT 13 is a single-substitution cipher.

Atbash Cipher

Hebrew scribes copying the book of Jeremiah used the Atbash cipher. Using it is simple; you just reverse the alphabet. This is, by modern standards, a primitive and easy-to-break cipher. However, it will help you get a feel for how cryptography works.

The Atbash cipher is a Hebrew code that substitutes the first letter of the alphabet for the last and the second letter for the second to the last, etc. It simply reverses the alphabet; for example, *A* becomes *Z*, *B* becomes *Y*, *C* becomes *X*, etc.

This, like the Caesar and ROT 13 ciphers, is also a single-substitution cipher.

Multi-Alphabet Substitution

Eventually, a slight improvement on the Caesar cipher was developed, called multi-alphabet substitution (also called polyalphabetic substitution). In this scheme, you select multiple numbers by which to shift letters (that is, multiple substitution alphabets). For example, if you select three substitution alphabets (12, 22, 13), then

```
A CAT
```

becomes

```
C ADV
```

Notice that the fourth letter starts over with another 12, and you can see that the first *A* was transformed to *C* and the second *A* was transformed to *D*. This makes deciphering the underlying text more difficult. Although this is harder to decrypt than a Caesar cipher, it is not overly difficult to decode. It can be done with simple pen and paper and a bit of effort. It can be cracked quickly with a computer. In fact, no one would use such a method today to send any truly secure message, for this type of encryption is considered very weak.

One of the most widely known multi-alphabet ciphers was the Vigenère cipher. This topic is discussed in detail later in this chapter. This cipher was invented in 1553 by Giovan Battista Bellaso. It is a method of encrypting alphabetic text by using a series of different mono-alphabet ciphers selected based on the letters of a keyword. This algorithm was later misattributed to Blaise de Vigenère, and so it is now known as the "Vigenère cipher," even though Vigenère did not really invent it.

Multi-alphabet ciphers are more secure than single-substitution ciphers. However, they are still not acceptable for modern cryptographic usage. Computer-based cryptanalysis systems can crack historical cryptographic methods (both single alphabet and multi-alphabet) easily. The single-substitution and multi-substitution alphabet ciphers are discussed just to show you the history of cryptography, and to help you get an understanding of how cryptography works.

Rail Fence

All the preceding ciphers we examined are substitution ciphers. Another approach to classic cryptography is the transposition cipher. The *rail fence* cipher may be the most widely known transposition cipher. You simply take the message you wish to encrypt and alter each letter on a different row. So "attack at dawn" is written as

A t c a d w
 t a k t a n

Next, you write down the text reading from left to right as one normally would, thus producing

atcadwtaktan

In order to decrypt the message, the recipient must write it out on rows:

A t c a d w
 t a k t a n

Then the recipient reconstructs the original message. Most texts use two rows as examples; however, this can be done with any number of rows you wish to use.

Vigenère

As we previously discussed, a polyalphabetic cipher uses multiple substitutions in order to disrupt letter and word frequency. Let us consider a simple example. Remember a Caesar cipher has a shift, for example a shift of +2 (two to the right). A polyalphabetic substitution cipher would use multiple shifts. Perhaps a +2, –1, +1, +3. When you get to the fifth letter, you simply start over again. So, consider the word Attack, being encrypted

A (1) + 2 = 3 or C

T (20) –1 = 19 or S

T (20) +1 = 21 or U

A (1) +3 = 4 or D

C (3) +2 = 5 or E

K (11) –1 = 10 or J

So, the ciphertext is CSUDEJ. Given that each letter has four possible substitutions, the letter and word frequency is significantly disrupted.

Perhaps the most widely known polyalphabetic cipher is the Vigenère cipher. This cipher was actually invented in 1553 by Giovan Battista Bellaso, though it is named after Blaise de Vigenère. It is a method of encrypting alphabetic text by using a series of different mono-alphabet ciphers selected based on the letters of a keyword. Bellaso added the concept of using any keyword one might wish, thereby making the choice of substitution alphabets difficult to calculate.

Enigma

It is really impossible to have a discussion about cryptography and not talk about Enigma. Contrary to popular misconceptions, the Enigma is not a single machine but rather a family of machines. The first version was invented by German engineer Arthur Scherbius near the end of World War I. It was used by several different militaries, not just the Nazi Germans.

Some military texts encrypted using a version of Enigma were broken by Polish cryptanalysts Marian Rejewski, Jerzy Rozycki, and Henryk Zygalski. The three basically reverse engineered a working Enigma machine and used that information to develop tools for breaking Enigma ciphers, including one tool named the cryptologic bomb.

The core of the Enigma machine was the rotors, or disks, that were arranged in a circle with 26 letters on them. The rotors were lined up. Essentially, each rotor represented a different single substitution cipher. You can think of the Enigma as a sort of mechanical polyalphabetic cipher. The operator of the Enigma machine would be given a message in plaintext and then type that message into Enigma. For each letter that was typed in, Enigma would provide a different ciphertext based on a different substitution alphabet. The recipient would type in the ciphertext, getting out the plaintext, provided both Enigma machines had the same rotor settings.

There were actually several variations of the Enigma machine. The Naval Enigma machine was eventually cracked by British cryptographers working at the now famous Bletchley Park. Alan Turing and a team of analysts were able to eventually break the Naval Enigma machine. Many historians claim this shortened World War II by as much as two years. This story is the basis for the 2014 movie *The Imitation Game*.

Binary Operations

Part of modern symmetric cryptography ciphers involves using binary operations. Various operations on binary numbers (numbers made of only zeroes and ones) are well known to programmers and programming students. But for those readers not familiar with them, a brief explanation follows. When working with binary numbers, three operations are not found in normal math: AND, OR, and XOR operations. Each is illustrated next.

AND

To perform the AND operation, you take two binary numbers and compare them one place at a time. If both numbers have a one in both places, then the resultant number is a one. If not, then the resultant number is a zero, as you see here:

```
1 1 0 1
1 0 0 1
-------
1 0 0 1
```

OR

The OR operation checks to see whether there is a one in either or both numbers in a given place. If so, then the resultant number is one. If not, the resultant number is zero, as you see here:

```
1 1 0 1
1 0 0 1
-------
1 1 0 1
```

XOR

The XOR operation impacts your study of encryption the most. It checks to see whether there is a one in a number in a given place, but not in both numbers at that place. If it is in one number but not the other, then the resultant number is one. If not, the resultant number is zero, as you see here:

```
1 1 0 1
1 0 0 1
-------
0 1 0 0
```

XORing has a an interesting property in that it is reversible. If you XOR the resultant number with the second number, you get back the first number. And, if you XOR the resultant number with the first number, you get the second number.

```
0 1 0 0
1 0 0 1
-------
1 1 0 1
```

Binary encryption using the XOR operation opens the door for some rather simple encryption. Take any message and convert it to binary numbers and then XOR that with some key. Converting a message to a binary number is a simple two-step process. First, convert a message to its ASCII code, and then convert those codes to binary numbers. Each letter/number will generate an eight-bit binary number. You can then use a random string of binary numbers of any given length as the key. Simply XOR your message with the key to get the encrypted text, and then XOR it with the key again to retrieve the original message.

This method is easy to use and great for computer science students; however, it does not work well for truly secure communications because the underlying letter and word frequency remains. This exposes valuable clues that even an amateur cryptographer can use to decrypt the message. Yet, it does provide a valuable introduction to the concept of single-key encryption, which is discussed in more detail in the next section. Although simply XORing the text is not the method typically employed, single-key encryption methods are widely used today. For example, you could simply include a multi-alphabet substitution that was then XORed with some random bit stream—variations of which do exist in a few actual encryption methods currently used.

Modern cryptography methods, as well as computers, make decryption a rather advanced science. Therefore, encryption must be equally sophisticated in order to have a chance of success.

What you have seen so far regarding encryption is simply for educational purposes. As has been noted several times, you would not have a truly secure system if you implemented any of the previously mentioned encryption schemes. You might feel that this has been overstated in this text. However, having an accurate view of what encryption methods do and do not work is critical. It is now time to discuss a few methods that are actually in use today.

The following websites offer more information about cryptography:

- Cryptography I course on Coursera: https://www.coursera.org/course/crypto
- Applied cryptography course on Udacity: https://www.udacity.com/course/applied-cryptography--cs387
- Cypher Research Laboratories: www.cypher.com.au/crypto_history.htm

Understanding the simple methods described here and other methods provided by the aforementioned websites should give you a sense of how cryptography works as well as what is involved in encrypting a message. Regardless of whether you go on to study modern, sophisticated encryption methods, having some basic idea of how encryption works at a conceptual level is important. Having a basic grasp of how encryption works, in principle, will make you better able to understand the concepts of any encryption method you encounter in the real world.

FYI: Careers in Cryptography

Some readers might be interested in a career in cryptography. Basic knowledge of cryptography is enough to be a security administrator, but not enough to be a cryptographer. A strong mathematics background is essential for in-depth exploration of cryptography, particularly when pursuing a career in this field. An adequate background includes a minimum of the complete calculus sequence (through differential equations), statistics through basic probability theory, abstract algebra, linear algebra, and number theory. A double major in computer science and mathematics is ideal. A minimum of a minor in mathematics is required, and familiarity with existing encryption methods is critical.

Learning About Modern Encryption Methods

Not surprisingly, modern methods of encryption are more secure than the historical methods just discussed. All the methods discussed in this section are in use today and are considered reasonably secure. Note that DES is an exception, but only due to its short key length.