

Introduction to Cryptology: a Workbook

Howard A. Blair

Copyright © 2010 Howard A. Blair. All commercial rights reserved.

1 Polyalphabetic and polygraphic cyphers

1.1 The Vigenère Cipher

Note 1.1: We begin with the useful conversion: list the members of the English alphabet in standard order, beginning with $a_0 = a$, and ending with $a_{25} = z$.

input	output
a_0	a
\vdots	\vdots
a_5	f
\vdots	\vdots
a_{10}	k
\vdots	\vdots
a_{15}	p
\vdots	\vdots
a_{20}	u
\vdots	\vdots
a_{25}	z

Note that Lewand's index of the English alphabet begins with 1 and ends with 26.

△

Note 1.2: We represent plaintext and ciphertext messages as strings without spaces or punctuation, as in the plaintext

mollywillneverbreakthis

Since we are used to representing strings as arrays of characters indexed from 0 to $n - 1$, where n is the length of the array, we will indicate strings as the message above by

$$\alpha_0 \alpha_1 \dots, \alpha_{n-1}$$

where n is the length of the string. For example, if the string under consideration is `mollywillneverbreakthis`, then e.g. $\alpha_0 = \text{m}$, $\alpha_4 = \text{y}$ and $\alpha_{10} = \alpha_{12} = \text{e}$. $n = 23$.

△

Note 1.3: Now, choose another string of letters from the English alphabet. Generally, this string is chosen to be considerably shorter than the plaintext message. Following Lewand's example, choose `chaos`. The chosen string is called the *key*. We then form a *keysting* of the same length as the plaintext by repeating the key over and over, truncating the last occurrence of the key as necessary. If we place the two strings in one-to-one correspondence as indicated below, then, in this example, we obtain

c	h	a	o	s	c	h	a	o	s	c	h	a	o	s	c	h	a	o	s	c	h	a
m	o	l	l	y	w	i	l	l	n	e	v	e	r	b	r	e	a	k	t	h	i	s

△

Note 1.4: Consider any string ρ . Let n be the length of ρ and let $0 \leq j \leq n - 1$. Let $\text{index}(\rho_j)$ be the index of the letter ρ_j in the table in note 1.1. For example, if $\rho = \text{mollywillneverbreakthis}$, then $\text{index}(\rho_4) = \text{index}(\text{y}) = 24$ and $\text{index}(\rho_{17}) = \text{index}(\text{sfa}) = 0$.

△

Note 1.5: Let α be the plaintext string and let κ be the keysting. We encipher α as follows. Let γ be the ciphertext string. Then

$$\text{index}(\gamma_j) = \text{index}(\kappa_j) + \text{index}(\alpha_j)$$

We decipher γ by

$$\text{index}(\alpha_j) = \text{index}(\gamma_j) - \text{index}(\kappa_j)$$

△

Definition 1.1: The cipher described in the above notes is called the Vigenère Cipher.

△

Problem 1.1: Poke around on the internet to get a little information about Vigenère.

△

Problem 1.2: Use the rules we gave above to verify the ciphertext in Lewand's example.

△

Problem 1.3: Implement a simple program to encipher and decipher strings of lowercase letters.

△

1.2 Miscellaneous problems

Problem 1.4: Let ρ be a string of length n . Suppose the letter **a** occurs n_0 times in ρ . What is the probability that the letter occurring in a randomly chosen position in the string is **a**?

△

Problem 1.5: Suppose the letter **a** occurs n_0 times in ρ . What is the probability that if two positions in the string are randomly chosen, the letter occurring in both positions is **a**?

△

Problem 1.6: Suppose for each $0 \leq i \leq 25$, the letter a_i occurs n_i times in ρ . What is the probability that if two positions in ρ are randomly chosen, the letter occurring in both positions is the same?

△

Definition 1.2: The probability that if two positions in string ρ are randomly chosen, the letter occurring in both positions is the same is called the *index of coincidence*, or *IC* of ρ .

△

Problem 1.7: What is the expected number of occurrences of letter a_i in a string derived from English text?

△

Problem 1.8: Note how Lewand determines a useful approximation to the IC of a long string. Explain how Lewand calculates the average IC of all strings of English text. Then explain how this average IC can be used to give a probabilistic indication of whether the cipher is monoalphabetic or polyalphabetic.

△

1.3 The Hill cipher

Problem 1.9: The following ciphertext was produced by a Hill digraph encipherment. Decrypt it.

TURMET ZLBZTU WZHAGG LZUKCO LKAP

Note: the rightmost letter in the block *LZUKCO* may have been incorrectly given in class on Monday, March 1st.

△

2 Public-Key Cryptography: RSA

2.1 “Euler’s φ ”

Definition 2.1: Let \mathbb{N}^+ be the set of positive integers. Let $\varphi(n)$, where

$$\varphi : \mathbb{N}^+ \longrightarrow \mathbb{N}^+$$

be the number of positive integers that are both less than or equal to n and relatively prime to n .

△

Note 2.1: We will need two main theorems about Euler’s φ to see that the RSA Cryptosystem works. The first theorem tells us how to calculate $\varphi(n)$ and the second

theorem, which depends on the first theorem, tells us how to invert elements in modular arithmetic. A proof of a theorem is a way of seeing that the theorem is true. There is often more than one proof of a theorem, although apparently distinct proofs, from a deeper point of view, can sometimes be different realizations of the same thing. The proof of the first of the two theorem's about Euler's φ is organized around two main simple ideas which are embodied in two lemmas, i.e. subordinate theorems, and several additional supporting lemmas. The proofs of these lemmas were covered in class, but we will state the lemmas, most without proof in our presentation of RSA when we need them. We now state the first of the two main theorems.

△

Theorem 2.1: Let n be a positive integer with unique prime factorization given by

$$n = p_1^{e_1} \cdot \dots \cdot p_n^{e_n}$$

Then

$$\varphi(n) = (p_1^{e_1} - p_1^{e_1-1}) \cdot \dots \cdot (p_n^{e_n} - p_n^{e_n-1})$$

■

Example 2.1:

$$\varphi(1000) = \varphi(2^3 \cdot 5^3) = (2^3 - 2^2) \cdot (5^3 - 5^2) = 400$$

△

Note 2.2: We can restate the previous theorem as follows.

△

Theorem 2.2: Let n be a positive integer with unique prime factorization given by

$$\varphi(n) = n \prod_{p|n} \frac{p-1}{p}$$

Then

$$\varphi(n) = (p_1^{e_1} - p_1^{e_1-1}) \cdot \dots \cdot (p_n^{e_n} - p_n^{e_n-1})$$

where p is a prime-valued variable.

■

Note 2.3: The previous theorem is just a restatement of the first of the two main theorems. The second main theorem remains to be given.

△

Example 2.2:

$$\varphi(1000) = 1000 \cdot \prod_{p|1000} \frac{p-1}{p} = 1000 \cdot \frac{2-1}{2} \cdot \frac{5-1}{5} = 400$$

△

Note 2.4: The value of an empty sum is considered to be 0 and the value of an empty product is considered to be 1. Therefore,

$$\varphi(1) = 1 \cdot \prod_{p|1} \frac{p-1}{p} = 1 \cdot 1 = 1$$

In other words, since no primes divide 1,

$$\prod_{p|1} a_p = 1$$

△

Note 2.5: We will show in the next problem that the previous two theorems are restatements of each other.

△

Problem 2.1: $p_i^{e_i-1}$ can be factored out of each factor $p_1^{e_i} - p_i^{e_i-1}$ in the product $(p_1^{e_1} - p_1^{e_1-1}) \cdot \dots \cdot (p_n^{e_n} - p_n^{e_n-1})$; i.e.

$$p_1^{e_i} - p_i^{e_i-1} = p_i^{e_i-1}(p-1)$$

Use this factorization to show that

$$(p_1^{e_1} - p_1^{e_1-1}) \cdot \dots \cdot (p_n^{e_n} - p_n^{e_n-1}) = n \cdot \prod_{p|n} \frac{p-1}{p}$$

△

Note 2.6: The next two lemmas are at the heart of Euler's φ .

△

Lemma 2.1:

$$\gcd(m \bmod n, n) = \gcd(m, n)$$

■

Note 2.7: We have an immediate consequence (i.e. corollary) of this lemma:

△

Corollary 2.1: Let $\{a_0, a_1, \dots, a_{n-1}\}$ be any set of distinct integers such that

$$a_i \bmod n = a_j \bmod n \text{ iff } a_i = a_j$$

for all $0 \leq i, j \leq n-1$. Then, the subset of $\{a_0, a_1, \dots, a_{n-1}\}$ all of whose members are relatively prime to n , is the set $\{a_i \mid \gcd(a_i \bmod n, n) = 1\}$.

■

Note 2.8: Lemma 2.1 and its corollary are “main lemma number one”. The next lemma is “main lemma number two”.

△

Lemma 2.2: If m and n are relatively prime, and then k is relatively prime to their product mn iff k is relatively prime to both m and n .

■

Problem 2.2: Without looking it up, try to prove the lemma 2.2 yourself.

△

Problem 2.3: Again, without looking it up, try to prove the next lemma yourself.

△

Lemma 2.3: If m and n are relatively prime, and then

$$\varphi(mn) = \varphi(m)\varphi(n)$$

■

Problem 2.4: Once again, without looking it up, try to prove the first of the main theorems, theorem 2.1 yourself, using lemma 2.3.

△

Note 2.9: We now state the second main theorem about Euler's φ .

△

Theorem 2.3: If a and n are relatively prime positive integers, then

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

Proof: Let

$$\{a_1, \dots, a_{\varphi(n)}\}$$

be the subset of $\{0, \dots, n-1\}$ whose members are relatively prime to n .

The following sequence of problems are the main steps of the proof of theorem 2.3. The hypothesis that a and n are relatively prime positive integers applies in each of the problems until the proof of the theorem is finished.

Problem 2.5: Show that if $(a \cdot i) \bmod n = (a \cdot j) \bmod n$, then $a \cdot i = a \cdot j$, for all $0 \leq i, j < n$. [Hint: Use: if a and n are relatively prime and $n|(a \cdot k)$, then $n|k$].

△

The result in the preceding problem shows that the function that maps an integer k to $k \bmod n$ is 1-to-1 on the set $\{a \cdot 0, \dots, a \cdot (n-1)\}$.

Problem 2.6: What lemma shows that for any $0 \leq i < n$, $a \cdot i$ is relatively prime to n iff $(a \cdot i) \bmod n$ is relatively prime to n ?

△

Problem 2.7: Let

$$a'_i = (a \cdot a_i) \bmod n$$

Show that

$$a \cdot a_i \equiv a'_i \pmod{n}$$

△

Problem 2.8: Show that

$$(a \cdot a_1) \cdot \dots \cdot (a \cdot a_{\varphi(n)}) \equiv a'_1 \cdot \dots \cdot a'_{\varphi(n)} \pmod{n}$$

△

Problem 2.9: Show that

$$\{a_1, \dots, a_{\varphi(n)}\} = \{a'_1, \dots, a'_{\varphi(n)}\}$$

△

Problem 2.10: Show that

$$a^{\varphi(n)} \cdot a_1 \cdot \dots \cdot a_{\varphi(n)} \equiv a_1 \cdot \dots \cdot a_{\varphi(n)} \pmod{n}$$

△

Let $x = a_1 \cdot \dots \cdot a_{\varphi(n)}$. By the result of the previous problem,

$$a^{\varphi(n)} \cdot x \equiv x \pmod{n}$$

Problem 2.11: Show that if $\gcd(k, n) = 1$, then there exists positive integer k' such that $kk' \equiv 1 \pmod{n}$.

△

Problem 2.12: Show that $\gcd(x, n) = 1$.

△

Problem 2.13: Finish the proof of theorem 2.3.

△

Problem 2.14: Show by numerical calculation that

$$31^{\varphi(128)} \equiv 1 \pmod{128}$$

△

Problem 2.15: **Hand in for midterm problem set** Verify that 221 and 1025640 are relatively prime. Then use Euler's theorem to solve the congruence

$$221 \cdot x \equiv 1 \pmod{1025640}$$

△

Proposition 2.1: Let

$$ac \equiv 1 \pmod{n}$$

for some c . Then a and n are relatively prime.

■

Problem 2.16: **Hand in for midterm problem set** Prove the previous proposition.

△

Proposition 2.2: Let

$$ac \equiv 1 \pmod{n}$$

and

$$ad \equiv 1 \pmod{n}$$

Then

$$c \equiv d \pmod{n}$$

■

Problem 2.17: Prove the previous proposition.

△

Definition 2.2: (The RSA Public-key Cryptosystem) Fix a method of converting characters into numbers. For example, identify a character with its ASCII code. Then for each participant, do the following:

1. Choose distinct large prime numbers p and q .
2. Let $n = pq$. Note that $\varphi(n) = (p - 1)(q - 1)$.
3. Choose e such that $e < n$ and e is relatively prime to $\varphi(n)$. Note that e uniquely determines $0 < d < n$ such that $ed \equiv 1 \pmod{\varphi(n)}$.
4. The numbers n and e are issued as the participant's public key.
5. The number d is issued as the participant's private key.

To send a message to a participant, whether or not the sender is a participant in the key distribution system, do:

1. Convert a character block into a plaintext number m such that $m < n$.
2. Calculate the ciphertext number: $c = (m^e) \bmod n$. Note that e and n are public and therefore available to the sender whether or not the sender is a participant

in the key distribution system. The sender and receiver have to agree on the correspondence between plaintext character blocks and plaintext numbers.

3. Send c .
4. Repeat for each character block. (The character blocks do not have to be the same size if there is an agreed upon number of digits - or bits - for indexing each character such as would be the case if ASCII codes were used.)

For a participant to decipher a ciphertext number c , do:

1. Use the private key d to calculate $(c^d) \bmod n$.
2. Convert $(c^d) \bmod n$ to a plaintext character block.

△

Theorem 2.4: Let n, e and d be as specified for RSA. Let

$$c = (m^e) \bmod n$$

Then

$$m = (c^d) \bmod n$$

In the next sequence of problems we will prove the theorem and related lemmas.

Lemma 2.4: Let $n = pq$ and let e and d be as specified for RSA. Then

$$m^{ed} \equiv m \bmod p$$

Proof: e and d were selected such that

$$ed \equiv 1 \bmod \varphi(n)$$

Problem 2.18: **Hand in for midterm problem set** Continue the proof of the lemma by showing that for some integer t ,

$$ed = 1 + t(p - 1)(q - 1)$$

△

We now continue the argument by observing that there are two cases:

1. m and p are not relatively prime. In this case, since p is prime, $p|m$.
2. m and p are relatively prime.

Problem 2.19: Hand in for midterm problem set Continue the proof by showing that in case 1,

$$m \equiv 0 \pmod{p}$$

Show that therefore

$$m^{ed} \equiv 0 \pmod{p}$$

Show that therefore

$$m^{ed} \equiv m \pmod{p}$$

△

We now continue the argument by reasoning within case 2. Since m and p are relatively prime, by Euler's Theorem we have

$$m^{\varphi(p)} \equiv 1 \pmod{p}$$

Proposition 2.3: Let k, v be any integers, and let r, s, u be any positive integers. Let $k^r \equiv v \pmod{u}$. Then

$$(k^r)^s \equiv v^s \pmod{u}$$

■

Problem 2.20: Hand in for midterm problem set Prove the previous proposition. Reason carefully; it's a little harder than it looks.

△

We can now proceed with the proof of the Lemma, still in case 2.

$$\begin{aligned}
m^{ed} &\equiv m^{1+t(p-1)(q-1)} \pmod{p} \\
&\equiv m(m^{t(p-1)(q-1)}) \pmod{p} \\
&\equiv m(m^{(p-1)t(q-1)}) \pmod{p} \\
&\equiv m(m^{\varphi(p)})^{t(q-1)} \pmod{p} \\
&\equiv m(1)^{t(q-1)} \pmod{p} \quad [\text{Problem: Why?}] \\
&\equiv m(1) \pmod{p} \\
&\equiv m \pmod{p}
\end{aligned}$$

This completes the proof of the lemma. ■

By interchanging p and q in the preceding lemma and its argument we obtain

Lemma 2.5: Let $n = pq$ and let e and d be as specified for RSA. Then

$$m^{ed} \equiv m \pmod{p}$$

■

By the two lemmas, each of the primes p and q divide $m^{ed} - m$. Therefore, $n | m^{ed} - m$.
Therefore,

$$m^{ed} \equiv m \pmod{n}$$

■