EE 418: Handout #2

Polyalphabetic Cryptosystems and Cryptanalysis

Lecture notes of Professor Radha Poovendran
Network Security Lab (NSL)
Dept. of Electrical and Computer Engineering
University of Washington

Abstract. This lecture presents Euclidean algorithm, and then introduces Vigenère, Hill, permutation and stream ciphers. We define attack models for performing cryptanalysis on different ciphers, and show how statistical analysis of the English language can be used to cryptanalyze cryptosystems with large key spaces. Readings from Chapter 2 of D. Stinson, 4th ed.

1 The Euclidean Algorithm(s)

Many of the crypto systems presented during the course requires finding the multiplicative inverse of an integer a, denoted as a^{-1} under modulo arithmetic with base integer b. The Euclid's algorithm and the extended version become handy in solving them. We will first review the basic Euclid's algorithm for finding the greatest common divisor between two integers a, b, with the assumption $a > b$. We then state the condition for the equation $ax \equiv 1$ modulo b to have a solution. We present the extended Euclidean algorithm that helps to find the a^{-1} under modulo arithmetic with base b.

Lemma 1. *Let a and b be integers. Then there exists a unique integer d satisfying the following properties:*

1. *$d|a$ and $d|b$*
2. *If c is another integer such that $c|a$ and $c|b$, then $c|d$.*

d is defined to be the greatest common divisor (gcd) *of a and b.*

The Euclidean algorithm can be used to find the gcd of two integers. It is given by Algorithm 6.1 of Stinson, 4th ed (pg. 189), reproduced in Figure 1 for your convenience.

EUCLIDEAN ALGORITHM
Input: Positive integers a and b
Output: Greatest common divisor d of a and b
$r_0 \leftarrow a$
$r_1 \leftarrow b$
$m \leftarrow 1$
while $r_m \neq 0$
  $q_m \leftarrow \lfloor \frac{r_{m-1}}{r_m} \rfloor$
  $r_{m+1} \leftarrow r_{m-1} - q_m r_m$
  $m \leftarrow m + 1$
end while
$d \leftarrow r_{m-1}$
return d

Fig. 1: The Euclidean algorithm. Finds the greatest common divisor of a and b. Assume $a > b$.

The Euclidean algorithm finds the gcd through repeated integer division. First, $r_0 = a$ is divided by $r_1 = b$ and the remainder r_2 is found. In the next step, $r_1 = b$ is divided by r_2 and the remainder r_3 is found. The process continues until the remainder of r_{m-1} divided by r_m is zero. The gcd(a,b) = gcd(r_0, r_1) is the last non-zero divisor, namely r_m. The steps of the division algorithm are shown below

$$\begin{aligned} r_0 &= q_1 r_1 + r_2 && (1)\\ r_1 &= q_2 r_2 + r_3 && (2)\\ r_2 &= q_3 r_3 + r_4 && (3)\\ &\cdots\cdots && (4)\\ r_{m-2} &= q_{m-1} r_{m-1} + r_m && (5)\\ r_{m-1} &= q_m r_m && (6)\\ & && (7) \end{aligned}$$

The terms r_i are the remainders at each step of the equations. The terms q_i are the quotients. Now consider the equation $r_i = q_{i+1} r_{i+1} + r_{i+2}$. The relationship between the divisor r_{i+1} and the remainder r_{i+2} is given by $0 \leq r_{i+2} < r_{i+1}$. We also assumed that $r_0 > r_1$. Hence, we can write $r_0 > r_1 > r_2 > \cdots r_m$.

There are several interesting properties that are associated with this division algorithm.

- Algorithm terminates in finite steps.
- r_m is the gcd(a,b) = gcd(r_0, r_1).

The remainder sequence r_i is non-negative and monotonically decreasing. The first term r_0 is finite. Since each remainder is integer, the difference between any two adjacent remainders is at least one. Hence, the sequence must reach the limit value of 0 in finite steps. In the worst case, it will take r_0 steps to terminate.

To show that r_m is the gcd(a, b). Let d = gcd(a, b). Then $d|a, d|b$. Hence $d|r_2$. In addition, since $d|r_1, d|r_2$, and $r_1 = q_2 r_2 + r_3$, we can conclude $d|r_3$. By induction, lets assume that $d|r_i$ for all values of $i < j$. Then $r_{j-2} = q_{j-1} r_{j-1} + r_j$ implies that $d|r_j$. Hence, by induction, d divides all the remainders. In particular, $d|r_m$, the last non-zero divisor.

On the other hand, $r_m|r_{m-1}$ at the last step. Looking up one step above the last step, we have $r_{m-2} = q_{m-1} r_{m-1} + r_m$. Since r_m divides the right hand side, $r_m|r_{m-2}$. Continuing this way up, by induction, lets assume that $r_m|r_{m-l}$ for $l < j$. Then looking at $r_{m-j} = q_{m-(j-1)} r_{m-(j-1)} + r_{m-(j-2)}$, the right hand side is divisible by r_m. Hence, $r_m|r_{m-j}$. Hence, by induction, we have that $r_m|b$ and $r_m|a$. Hence, r_m is a common divisor of a, b. Since d=gcd(a,b), by definition, $r_m|d$. We now have $r_m|d$ and $d|r_m$. Hence, $d = r_m = gcd(a, b)$.

Example: Let $a = 87$ and $b = 24$. Then we have:

$$\begin{aligned} 87 &= 3(24) + 15 && (8)\\ 24 &= 1(15) + 9 && (9)\\ 15 &= 1(9) + 6 && (10)\\ 9 &= 1(6) + 3 && (11)\\ 6 &= 2(3) && (12) \end{aligned}$$

Therefore $\gcd(87, 24) = 3$.

When the gcd(a, b) =1, then, the Euclidean algorithm also allows one to find the multiplicative inverse of a under modulo b. the following lemma is key to finding the inverses.

Lemma 2. *Let a and b be positive integers, and let $d = \gcd(a, b)$. Then there exist integers x and y such that*

$$ax + by = d \tag{13}$$

Question: Suppose we have such integers x and y. How can we use them to find the inverse of a modulo b?

Answer: We have seen that a has an inverse mod b iff $\gcd(a, b) = 1$. By Lemma 2, there exist x and y such that

$$ax + by = 1 \tag{14}$$

Then we can write $1 - ax = by$. Hence, $b|(ax - 1)$. By definition then

$$ax \equiv 1 \pmod{b} \tag{15}$$

Hence if we can find x and y satisfying Eq. (15), then we can invert a modulo b. The algorithm for finding x and y: is called the *extended Euclidean algorithm*, presented on pg. 191 of Stinson, 4th ed and in Figure 2.

EXTENDED EUCLIDEAN ALGORITHM
Input: Positive integers a and b
Output: Integers r, s, and t such that
$r = \gcd(a, b)$ and $sa + tb = r$
$a_0 \leftarrow a$
$b_0 \leftarrow b$
$t_0 \leftarrow 0$
$t \leftarrow 1$
$s_0 \leftarrow 1$
$s \leftarrow 0$
$q \leftarrow \lfloor \frac{a_0}{b_0} \rfloor$
$r \leftarrow a_0 - qb_0$
while $r > 0$
  $temp \leftarrow t_0 - qt$
  $t_0 \leftarrow t$
  $t \leftarrow temp$
  $temp \leftarrow s_0 - qs$
  $s_0 \leftarrow s$
  $s \leftarrow temp$
  $a_0 \leftarrow b_0$
  $b_0 \leftarrow r$
  $q \leftarrow \lfloor \frac{a_0}{b_0} \rfloor$
  $r \leftarrow a_0 - qb_0$
end while
$r \leftarrow b_0$
return (r, s, t)

Fig. 2: The extended Euclidean algorithm.

Example: Let $a = 7$, $m = 26$. Find $a^{-1} \bmod m$.

First, let's look at the Euclidean algorithm.

$$26 = 3(7) + 5 \tag{16}$$
$$7 = 1(5) + 2 \tag{17}$$
$$5 = 2(2) + 1 \tag{18}$$

Now, let's rewrite the last equation to put the gcd (which is 1) on to the left-hand side of the equation.

$$1 = 5 - 2(2) \tag{19}$$

From Eq. (17), we have

$$2 = 7 - 5 \tag{20}$$

Substituting Eq. (20) into Eq. (19) yields

$$1 = 5 - 2(7 - 5) = 3(5) - 2(7) \tag{21}$$

We're almost there; the last step is to use Eq. (16), as follows:

$$5 = 26 - 3(7) \tag{22}$$

so that

$$1 = 3(26 - 3(7)) - 2(7) = 3(26) - 11(7) \tag{23}$$

And so $7^{-1} \bmod 26 = -11 \bmod 26 = 15 \bmod 26$.

The math behind the computation of x, y, has its own update equations as shown below. The main statement is the following:

Lemma 3. *Let $r_0 = a$ and $r_1 = b$ be positive integers, and let $d = \gcd(a, b)$. If the sequences $r_i, i > 1$ satisfy the division algorithm such that $r_{i-2} = q_{i-1}r_{i-1} + r_i$, Then there exist integers x_i and y_i such that*

$$ax_i + by_i = r_i. \tag{24}$$

The logical proof goes via induction as follows: We know from $r_0 = q_1 r_1 + r_2$ that $r_2 = q_0 - q_1 r_1 = a - q_1 b$. Similarly, from $r_1 = q_2 r_2 + r_3$, we have $r_3 = r_1 - q_2 r_2 = b - q_2(a - q_1 b) = (1 + q_1 q_2)b - q_2 a$. Hence, letting $x_2 = 1, y_2 = -q_1, x_3 = -q_2, y_3 = (1 + q_1 q_2)$, we have $r_2 = ax_2 + by_2$ and $r_3 = ax_3 + by_3$. Let this be true for all values of $i < j$. Then we can write $r_j = r_{j-2} - q_{j-1} r_{j-1}$. But by induction, we then have $r_j = (ax_{j-2} + by_{j-2}) - q_{j-1}(ax_{j-1} + by_{j-1})$ leading to $r_j = a(x_{j-2} - q_{j-1}x_{j-1}) + b(y_{j-2} - q_{j-1}y_{j-1})$. Letting $x_j = x_{j-2} - q_{j-1}x_{j-1}$ and $y_j = y_{j-2} - q_{j-1}y_{j-1}$ leads to $r_j = ax_j + by_j$. Hence by induction, at the final step of the division algorithm, we have $r_m = d = gcd(a, b) = ax_m + by_m$. Setting the final values to $x = x_m$ and $y = y_m$ leads to $d = ax + by$ as desired.

2 The Vigenère Cipher

The idea employed in this cryptosystem is to use a vector of m keys, i.e., $\underline{\mathbf{K}} = (K_1, K_2, .., K_m)$.
$\mathscr{P} = \mathscr{C} = \mathscr{K} = (\mathbb{Z}_{26})^m$ where $(\mathbb{Z}_{26})^m$ is an m-tuple. The difference between the Vigenère cipher and the Shift, Substitution, and Affine ciphers is that each alphabetic character is not uniquely mapped to another alphabetic character.

$$y = e_K(x_1, x_2, .., x_m) = (x_1 + K_1, x_2 + K_2, .., x_m + K_m) \bmod 26, \tag{25}$$
$$d_K(y_1, y_2, .., y_m) = (y_1 - K_1, y_2 - K_2, .., y_m - K_m) \bmod 26. \tag{26}$$

PLAINTEXT:	21 4 2 19	14 17
KEY:	2 4 6 7	2 4
CIPHER:	23 8 8 0	16 21
	X I I A	Q V

As an example, let $m = 4$, $\underline{\mathbf{K}} = (2, 4, 6, 7)$, and the plaintext be *vector*. From the correspondence table we have $\underline{\mathbf{x}} = (21, 4, 2, 19, 14, 17)$. Then:

To decrypt, the same keyword is used, but modulo subtraction is performed instead of modulo addition. The number of possible keywords of length m is 26^m, so even for small m an exhaustive search attack requires a long time.

3 The Hill Cipher

Consider the affine cipher, where $e_{(a,b)}(x) = ax + b \bmod m$, and suppose that $b = 0$, so that encryption is given by $e_{(a,0)}(x) = ax \bmod m$, i.e. multiplication by the secret key a modulo m. Decryption is then given by $d_K(y) = a^{-1}y \bmod m$, provided that $\gcd(a, m) = 1$.

Question: How can we generalize this from x corresponding to a single letter to x corresponding to a string of letters?

The idea is:

1. Choose an integer $m > 0$. Let us take for example $m = 2$.
2. Choose an $m \times m$ matrix K. For example,

$$K = \begin{pmatrix} 2 & 3 \\ 5 & 7 \end{pmatrix}. \tag{27}$$

3. Plaintext is written as row matrices. For example, if plaintext is ***test*** then we write it as

$$\begin{pmatrix} 19 & 4 \end{pmatrix}, \tag{28}$$

$$\begin{pmatrix} 18 & 19 \end{pmatrix}. \tag{29}$$

4. Encryption of te is:

$$\begin{pmatrix} 19 & 4 \end{pmatrix} \begin{pmatrix} 2 & 3 \\ 5 & 7 \end{pmatrix} = \begin{pmatrix} 38\text{+}20 & 57\text{+}28 \end{pmatrix} = \begin{pmatrix} 6 & 7 \end{pmatrix} \mod 26. \tag{30}$$

Encryption of st is:

$$\begin{pmatrix} 18 & 19 \end{pmatrix} \begin{pmatrix} 2 & 3 \\ 5 & 7 \end{pmatrix} = \begin{pmatrix} 1 & 5 \end{pmatrix} \mod 26. \tag{31}$$

Hence, the cipher is:

$$\begin{pmatrix} 6 & 7 & 1 & 5 \end{pmatrix}, \tag{32}$$

which is $GHBF$.

To decrypt, we will use K^{-1} as the decryption key. This begs the following question.

Question: What does it mean for a matrix to be invertible mod 26?

Answer: Much like with real numbers, there is an identity matrix over the integers mod n. The $m \times m$ identity matrix mod n (denoted I_m) has 1's along the diagonal and 0's elsewhere. As with the reals, for any matrix K, we have $KI_m = I_m K = K$. A matrix K is invertible mod n when there exists a matrix K^{-1} such that $KK^{-1} = K^{-1}K = I_m$.

Recall that a matrix K is invertible over the real numbers when its *determinant* is non-zero (see Stinson, 4th ed, pg 29 for a definition of the determinant). Analogously, K is invertible over $\mathbb{Z}_n$ when $\det K$ is invertible mod n, i.e. when $\gcd(\det K, n) = 1$.

Fair enough. But then how do we compute K^{-1} mod 26?

We will make use of Theorem 2.3 of Stinson, 4th ed:

Theorem 1. *Let K be a matrix such that $\gcd(\det K, n) = 1$. Then*

$$K^{-1} \bmod n \equiv (\det K)^{-1} K^* \bmod n \tag{33}$$

where the (i, j)-th entry of K^* is equal to $(-1)^{i+j} \det K_{ji}$ and K_{ji} is obtained by deleting the j-th row and i-th column of K.

When K is equal to the above encryption matrix, we have

$$K = \begin{pmatrix} 2 & 3 \\ 5 & 7 \end{pmatrix} \tag{34}$$

and

$$K^* = \begin{pmatrix} 7 & -3 \\ -5 & 2 \end{pmatrix} \equiv \begin{pmatrix} 7 & 23 \\ 21 & 2 \end{pmatrix} \bmod 26 \tag{35}$$

Furthermore, we have

$$(\det K)^{-1} \bmod 26 \equiv 25^{-1} \bmod 26 \equiv 25 \bmod 26 \tag{36}$$

Hence

$$K^{-1} \bmod 26 = (\det K)^{-1} K^* \bmod 26 = 25 \begin{pmatrix} 7 & 23 \\ 21 & 2 \end{pmatrix} \bmod 26 \tag{37}$$

$$= \begin{pmatrix} 175 & 575 \\ 525 & 50 \end{pmatrix} \bmod 26 = \begin{pmatrix} 19 & 3 \\ 5 & 24 \end{pmatrix} \bmod 26 \tag{38}$$

To decrypt with the Hill cipher, we multiply the ciphertext by K^{-1}. We leave it as an exercise to verify that yK^{-1} is equal to the original plaintext in this case.

Stated formally, the Hill cipher has $\mathscr{P} = \mathscr{C} = (\mathbb{Z}_{26})^m$, where $m \geq 2$. $\mathscr{K}$ ={set of all $m \times m$ invertible matrices over $\mathbb{Z}_{26}$}. For $K \in \mathscr{K}$:

$$e_K(x) = \underline{\mathbf{x}} K, \tag{39}$$

$$d_K(y) = \underline{\mathbf{y}} K^{-1}. \tag{40}$$

Question: Is the Hill cipher encryption an injective function?

Remark 2 *The permutation cipher is a special case of the Hill Cipher. Consider the following encryption rule* $\pi(x)$.

j	1	2	3	4	5	6
$\pi(j)$	3	5	1	6	4	2

It can be written as a Hill encryption matrix K_π *as follows:*

$$K_\pi = \begin{pmatrix} 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1 \\ 1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \end{pmatrix} \tag{41}$$

and the decryption matrix is:

$$K_\pi^T = \begin{pmatrix} 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0 \\ 1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1 \\ 0\,0\,0\,1\,0\,0 \\ 0\,1\,0\,0\,0\,0 \end{pmatrix} \tag{42}$$

Note that the decryption matrix is the transpose *of the encryption matrix, i.e. we obtain the decryption matrix by interchanging the rows and columns of the encryption matrix.*

4 Stream Ciphers

The idea of a stream cipher is to generate a keystream $z = z_1 z_2 \cdots$ and encrypt each character x_i of the plaintext with a different key z_i.

$$y = y_1 y_2 \cdots = e_{z_1}(x_1) e_{z_2}(x_2) \cdots . \tag{43}$$

In its simplest form the keystream is generated from a unique key K. This type of stream cipher is known as the "*synchronous*" cipher since the keystream is synchronized with the plaintext. Note that the keystream is independent of the plaintext. The encryption and decryption operations are given by:

$$y = e_z(x) = (x + z)(mod\, m),$$
$$x = e_z(y) = (y - z)(mod\, m),$$

Note: Often stream ciphers are described in terms of binary alphabets, i.e., $\mathscr{P} = \mathscr{C} = \mathbb{Z}_2$. In that case, encryption and decryption operations are just addition modulo 2. That is,

$$y = e_z(x) = (x + z)(mod\, m),$$
$$x = e_z(y) = (y + z)(mod\, m),$$

The ciphers we have seen so far, which are referred to as "*block*" ciphers, can be considered as a special case of stream ciphers in which the keystream is periodic with some period d. As an example, a Vigenère cipher with keyword length m can be defined as a stream cipher with the keystream z being:

$$z_i = \begin{cases} K_i & 1 \leq i \leq m \\ z_{i-m} & i \geq (m+1), \end{cases} \tag{44}$$

where $\underline{K} = \{K_1, K_2, \ldots, K_M\}$.

Generating a keystream

A keystream can be easily generated using a linear recurrence of degree m. Consider a binary alphabet. Then:

$$z_{i+m} = \sum_{j=0}^{m-1} c_j z_{i+j} \mod 2, \tag{45}$$

$\forall i \geq 1$, and $c_0, c_1, \ldots, c_{m-1}$ being some pre-specified linear weights. Note that to generate z we need to initialize the first m values of the keystream (this would be the key from which the stream is generated). The initialization vector should not consist entirely of zeros to avoid generating the zero keystream. For any other initialization, the resulting key stream will be periodic. The maximum possible period is $2^m - 1$, although not all initialization vectors will result in this period.

Example: Let $m = 4$ and

$$z_{i+4} = (z_i + z_{i+1}) \mod 2. \tag{46}$$

Assume a key $(1, 0, 0, 0)$. The keystream is periodic with period 15 and is as follows:

$$100010011010111 \cdots . \tag{47}$$

Initialization with any other key will give a cyclic permutation of the same keystream.

5 Cryptanalysis

Kerchoff's principle: The details of the cryptosystem used to secure the channel between Alice and Bob are known to Eve. In other words, Eve knows the tuple $(\mathscr{P}, \mathscr{C}, \mathscr{K}, \mathscr{E}, \mathscr{D})$ defining the cryptosystem.

5.1 Attack models

The goal of the adversary is to determine the key K that is used for the encryption/decryption process. The adversary may have different amount of information at its disposal, captured in the following attack models.

Type of attack	Description
Ciphertext only attack	Eve only observes the ciphertext y
Known plaintext attack	Eve knows the ciphertext y corresponding to plaintext x
Chosen plaintext attack	Eve has *temporary* access to an encryption box. The encryption box takes as input any chosen plaintext x and outputs the ciphertext y
Chosen ciphertext attack	Eve has *temporary* access to a decryption box. The decryption box takes as input any chosen ciphertext y and outputs the plaintext x

Based on these models we can analyze the security of each cryptosystem.

5.2 Cryptanalysis of the Shift Cipher

– **Ciphertext only:** Let $K = 3$ and the plaintext be *shift*. We then get *VKLIW* as the cipher (for a right shift). Assume Eve knows only the ciphertext $VKLIW$. Eve also knows that a shift cipher algorithm is used for encryption. Given the small cardinality of the key space, Eve can try all the possible 26 shifts in right direction. Upon shifting, the following plaintexts are obtained:

$vkliw \overset{1^{st} left\ shift}{\longrightarrow} ujkhv \overset{2^{nd} left\ shift}{\longrightarrow} tijgu \overset{3^{rd} left\ shift}{\longrightarrow} shift$, and so on. Since "shift" is the only dictionary word in the list of 26 possible words, Eve assumes that it is indeed the plaintext that was encrypted. Therefore, Eve can also infer the original key $K = 3$.

– **Known plaintext:** If Eve knows a (plaintext, ciphertext) pair, then Eve can find the key by subtracting the plaintext from the ciphertext mod 26. For instance, if Eve knows that plaintext b corresponds to ciphertext E, then Eve can determine that $K = 3$.

– **Chosen plaintext:** Choose letter a as plaintext; the resulting ciphertext will be the key. For example, if the ciphertext is P then $K = 15$.

– **Chosen cipher:** Choose A as the ciphertext. The plaintext is then the negative of the key K.

5.3 Remarks on Letter Distribution of the English Language

English language text has different frequencies for different alphabets. An estimate of relative frequencies (probabilities) of the 26 letters are as indicated in Table 1. Note that the letter **e** has the maximum relative frequency of 0.127.

Table 1: Probabilities of occurrence of the 26 alphabets.

A	B	C	D	E	F	G	H	I	J	K	L	M
0.082	0.015	0.028	0.043	0.127	0.022	0.020	0.061	0.070	0.002	0.008	0.040	0.024
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0.067	0.075	0.019	0.001	0.060	0.063	0.091	0.028	0.010	0.023	0.001	0.020	0.001

Similarly we can define frequencies of digrams, trigrams, initial letters, final letters, etc. We can then use the statistical properties of the English language to perform cryptanalysis. We will provide a simple example on the Affine Cipher.

A key observation to note is that the vowels "a, e, i, o" and the letters "t, s, b, h, d" have relatively high probability of appearance in English language. The table below indicates the rank order of vowels based on their frequencies.

The table below indicates the rank order of consonants "t, s, d, n, h" based on their frequencies.

Table 2: Rank order of the probabilities of occurrence of the vowels.

E	0.127
A	0.082
I	0.075
O	0.070
u	0.028

Table 3: Probabilities of most frequently occurring consonants.

T	0.091
S	0.063
N	0.067
H	0.061
D	0.043

5.4 Cryptanalysis of the Affine Cipher

– **Ciphertext only attack:** Eve has intercepted the following ciphertext:

FMXVEDKAPHFERBNDKRXRSREFMORUDSDKDVSH
VUFEDKAPRKDLYEVLRHHR

The most frequent letters are R with 8 occurrences, D with 7, E, K, H with 5 and F, V, S with 4. First guess is that $R = e$ and $D = t$. Given the encryption function

$$e_K(x) = ax + b \tag{48}$$

we get the following linear system:

$$4a + b = 17 \tag{49}$$
$$19a + b = 3. \tag{50}$$

Solving the system we obtain the unique solution $a = 6, b = 19$ (note solution must be in $\mathbb{Z}_{26}$). But for the affine cipher a has to be relatively prime to 26. Given that $gcd(26, 6) = 2$, $a = 6, b = 19$ is not a valid key. Second guess $R = e$ and $E = t$. Solving the linear system yields $a = 13$ which again is not a legal key. Third guess, $R = e$ and $K = t$, which yields $a = 3$, and $b = 5$. Since this is a valid key we decrypt the entire ciphertext to see if we get a meaningful English text.

$algorithms\ are\ quite\ general\ definitions\ of\ arithmetic\ processes$

Besides the statistical analysis, Eve could have tried all possible 312 pairs (a, b) that constitute a valid key for the affine cipher.

– *Known plaintext attack*: Let Eve know that $uw = 20\ 22$, has cipher $KQ = 10\ 16$. It can then determine the following linear system:

$$10 = 20a + b \pmod{26}, \tag{51}$$
$$16 = 22a + b \pmod{26}. \tag{52}$$

Equation 51 - Equation 52 gives:
$6 = 2a \ mod \ 26$. i.e. $2a = q \times 26 + 6 \Rightarrow a = 3, 16$. But $gcd(16, 26) \neq 1 \Rightarrow a = 3$. Then from Equation 51 we get:

$$10 = 20 \times 3 + b \pmod{26}, \tag{53}$$

$$i.e. \ -50 = b \pmod{26} \tag{54}$$

$$i.e. \ b = q \times 26 + (-50) \Rightarrow q = 2 \Rightarrow b = 2. \tag{55}$$

Hence Eve only needs to know two pairs of (cipher, plaintext) pairs.

– *Chosen plaintext*: Choose $ab = 0\ 1$ as plaintext. The cipher will be:

$$0 \times a + b \equiv b \pmod{26}, \tag{56}$$

$$1 \times a + b \equiv a + b \pmod{26}. \tag{57}$$

Hence Eve can find the key K.

– *Chosen ciphertext*: Choose AB as cipher and proceed as above.