Scribe: Cryptography and Network Security (Class.3.B)

Shubham Mishra

18-Sep-2020

1 Fermat's Little Theorem and Euler's Theorem

1.1 Introduction

Euler's Theorem: If $gcd(a, m) = 1$, then $a^{\phi(m)} \equiv 1\ (mod\ m)$

Here $\phi(m)$ is the Euler Phi Function.

When $m = p$ for a prime p, $\phi(m) = p - 1$. Hence, $a^{p-1} \equiv 1\ (mod\ p)$, which is known as **Fermat's Little Theorem**.

1.2 Proof

Let $R = \{r_1, r_2, ..., r_{\phi(m)}\}$ be the reduced system modulo m.

If $gcd(a, m) = 1$, we see that $\{ar_1, ar_2, ..., ar_{\phi(m)}\}$ is also the same reduced system modulo m, with $gcd(r_i, m) = 1 \forall i$.

This is because R contains all element co-prime to m and smaller, and since $gcd(a, m) = 1$, $gcd(ar_i, m) = 1$, hence it belongs to R. The cardinalities are same as well. Hence the 2 sets above are just permutations of each other.

$\therefore$ The product of elements in both the sets is the same.

$\therefore r_1 \cdot r_2 \cdot ...r_{\phi(m)} = (ar_1) \cdot (ar_2) \cdot ...(ar_{\phi(m)}) \implies a^{\phi(m)} \equiv 1 (mod m)$ which proves the theorem.

[Note that since $gcd(r_1, m) = 1$, r^{-1} exists.]

1.3 Examples

1. Find the remainder when 72^{1001} is divided by 31.

$72 \equiv 10\ (mod\ 31)$

$\implies 72^{1001} \equiv 10^{1001}\ (mod\ 31)$

$10^{30} \equiv 1 (mod 31)$ $[\because\ 31\ is\ prime]$

$\implies (10^{30})^{33} \equiv 1\ (mod\ 31)$

$\implies 10^{990} \equiv 1\ (mod\ 31)$

Now, $10^{1001} = 10^{990} \cdot 10^8 \cdot 10^2 \cdot 10^1 \equiv 1 \cdot (10^2)^4 \cdot 10^2 \cdot 10 \equiv 1 \cdot 7^4 \cdot 7 \cdot 10 \equiv 49^2 \cdot 7 \cdot 10 \equiv (-13)^2 \cdot 7 \cdot 10 \equiv 98 \cdot 10 \equiv 5 \cdot 10 \equiv 19\ (mod\ 31)$

$\implies 72^{1001} \equiv 19\ (mod\ 31)$

2. Find the least residue of 7^{973} $(mod\ 72)$.

$\phi(72) = \phi(3^2 \cdot 2^3) = 72 \cdot (1 - \frac{1}{3}) \cdot (1 - \frac{1}{2}) = 24$

$\implies 7^{24} \equiv 1\ (mod\ 72)$

Now, $7^{973} = 7^{960} \cdot 7^{13} \equiv 7^{13} = (7^3)^4 \cdot 7 \equiv (-17)^4 \cdot 7 \equiv 1 \cdot 7 \equiv 7\ (mod\ 72)$

3. $k|a \wedge k|b$ and let $gcd(k, m) = d$. Prove that $a \equiv b\ (mod\ m) \implies \frac{a}{k} \equiv \frac{b}{k}\ (mod\ m)$.

$\because gcd(k, m) = d$

$\therefore k = d\alpha\ and\ m = d\beta\ where\ gcd(\alpha, \beta) = 1$

$\because k|a\ and\ k|b$

$\therefore a = kp = d\alpha p\ and\ b = kq = d\beta q$

$a \equiv b\ (mod\ m) \implies m|(a - b) \implies m|k \cdot (a - b)$

$\implies k \cdot (p - q) = \gamma m \implies d\alpha(p - q) = \gamma d\beta \implies d(p - q) = \gamma\beta$

$\implies \beta|\alpha(\frac{a}{k} - \frac{b}{k})$

$\because gcd(\alpha, \beta) = 1$

$\therefore \beta|(\frac{a}{k} - \frac{b}{k})$

$\frac{a}{k} \equiv \frac{b}{k}\ (mod\ \beta) \implies \frac{a}{k} \equiv \frac{b}{k}\ (mod\ \frac{m}{d})$

4. Let $a \equiv b\ (mod\ p^k)$. Prove that $a^p \equiv b^p\ (mod\ p^{k+1})$ $(k \geq 1)$.

$a \equiv b\ (mod\ p^k) \implies a - b = \alpha p^k \implies a = b + \alpha p^k$

$\implies a^p = (b + \alpha p^k)^p = b^p + p \cdot b^{p-1} \cdot (\alpha p^k) + \binom{p}{2} b^{p-2} (\alpha p^k)^2 + ... + (\alpha p^k)^p$

$\because k \geq 1,\ nk \geq k + 1$

$\therefore Taking\ modulo\ p^{k+1},\ a^p \equiv b^p + 0\ (mod\ p^{k+1})$

2 Polyalphabetic Ciphers

As opposed to monoalphabetic ciphers, polyalphabetic ciphers can map one character of the plaintext alphabet to multiple characters to the ciphertext alphabet.

Examples: Vigenere Cipher, Hill Cipher etc.

3 Vigenere Cipher

3.1 Introduction

Vigenre Cipher is one of the earliest known Polyalphabetic ciphers.

- It named after Blaise de Vigenere[2] and a method for its cryptanalysis was given by Friedrich Kasiski.
- The key here is an m character long string and the cipher encrypts m characters of plaintext at a time.
- We divide the plaintext in blocks of m characters and add (modulo the alphabet size) the characters in each block with its corresponding character in the key.

3.2 Example

Consider the plaintext: `thiscryptosystemisnotsecure`, and the key: `cipher`

Here $m = 6$.

We shall write the plaintext and the key into residue modulo 26, as shown below:

Plaintext	Key	(Plaintext + Key) mod 26	Ciphertext
t = 19	c = 2	21	v
h = 7	i = 8	15	p
i = 8	p = 15	23	x
s = 18	h = 7	25	z
c = 2	e = 4	6	g
r = 17	r = 17	8	i

Plaintext	Key	(Plaintext + Key) mod 26	Ciphertext
y = 24	c = 2	0	a
p = 15	i = 8	23	x
t = 19	p = 15	8	i
o = 14	h = 7	21	v
s = 18	e = 4	22	w
y = 24	r = 17	15	p

Plaintext	Key	(Plaintext + Key) mod 26	Ciphertext
s = 18	c = 2	20	u
t = 19	i = 8	1	b
e = 4	p = 15	19	t
m = 12	h = 7	19	t
i = 8	e = 4	12	m
s = 18	r = 17	9	j

Plaintext	Key	(Plaintext + Key) mod 26	Ciphertext
n = 13	c = 2	15	p
o = 14	i = 8	22	w
t = 19	p = 15	8	i
s = 18	h = 7	25	z
e = 4	e = 4	8	i
c = 2	r = 17	19	t

Plaintext	Key	(Plaintext + Key) mod 26	Ciphertext
u = 20	c = 2	22	w
r = 17	i = 8	25	z
e = 4	p = 15	19	t

Hence the ciphertext will be `vpxzgiaxivwpubttmjpwizitwzt`.

3.3 Size of the key space

If we consider English lower case characters as alphabet, our key size is 26^m.

Note that, $26^5 \sim 1.1 \cdot 10^7$

So even with a key size of 5, brute forcing for the key is not feasible.

3.4 Implementation

The following functions in C can be used to encrypt and decrypt a plaintext p of size n to a ciphertext c using a key k of size m.

```
char* encrypt(char* p, int n, char *k, int m){
    char *c = (char *)malloc(n * sizeof(char));

    for (int i = 0; i < n; i++){
        c[i] = (p[i] - 'a' + k[i % m] - 'a' + 26) % 26 + 'a';
    }

    return c;
}

char* decrypt(char* c, int n, char *k, int m){
    char *p = (char *)malloc(n * sizeof(char));

    for (int i = 0; i < n; i++){
        p[i] = (c[i] - 'a' - k[i % m] - 'a' + 26) % 26 + 'a';
    }

    return p;
}
```

4 Cryptanalysis

4.1 Introduction

Cryptanalysis is the *art of breaking a cipher*. Cryptography and Cryptanalysis together form the subject of **Cryptology**.

4.2 Models of Cryptanalysis

- **Cipher-text only**: Adversary possesses a string of ciphertext.
- **Known plaintext**: Adversary possesses a known plaintext, x and its corresponding ciphertext y.
- **Chosen plaintext**: Attacker can choose plaintext and obtain the corresponding ciphertext.
- **Chosen ciphertext**:
  - The attacker has temporary access to the decryption oracle (he doesn't know how it works).
  - He can chose ciphertexts and decrypt to obtain the corresponding plaintexts.

It is evident that the list above is in increasing order of strength.

Generally, the attacker has access to the encryption oracle in a public key cryptosystem since we can encrypt our messages using the receiver's public key.

4.3 Statistical Analysis

If the letters in the English language were equally likely to occur, the probability of occurrence of each would have been $\frac{1}{26}$.

That would make the cryptanalysis difficult.

However, due to the structure of the words in English language, there exists a non-uniform probability distribution of the occurrence of the letters.

- *E* has highest probability, 0.120.
- *T, A, O, I, N, S, H, R* have probability in the range 0.06 to 0.09.
- *D, L* have probability of 0.04.
- *C, U, M, W, F, G, Y, P, B* have probability in the range 0.06 to 0.09.
- *V, K, J, X, Q, Z* have probability $\leq$ 0.01.

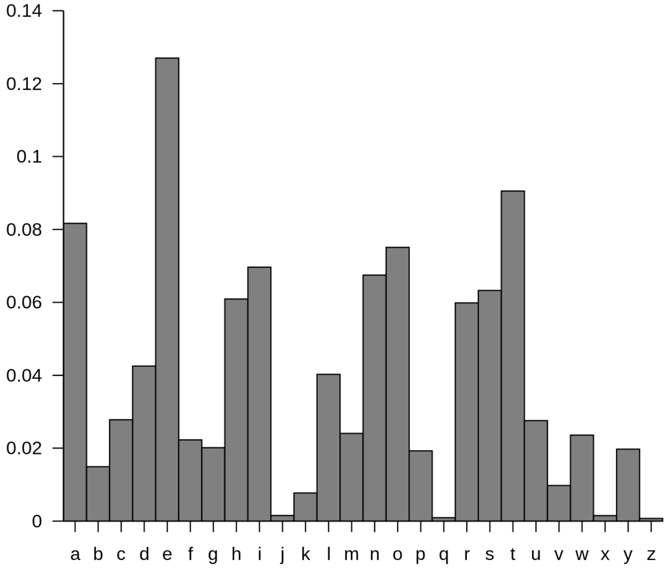

Figure 1: Distribution of frequencies[1]

Similar distribution exists in bigrams and trigrams as well.

Some common bigrams are: *TH, HE, IN, ER, AN, RE,* etc. Some common trigrams are: *THE, ING, AND, HER, ERE,* etc.

4.4 Cryptanalysis of Monoalphabetic Ciphers

No matter how complicated a Monoalphabetic Cipher is, all we need to know in order to decrypt it is a Look-up table consisting of the mapping of plaintext characters to the cipher text characters.

Since monoalphabetic ciphers have one-to-one correspondence between plaintext and ciphertext characters, the same probability distribution, albeit with the characters changed, applies to the ciphertext as well.

For example, if in a long enough ciphertext, we find that the letter T occurs around 12% of the time, we can be fairly sure that E is mapped to T.

This knowledge is enough to break a Shift cipher.

For affine cipher, we need one more additional information (since we need to determine two parts of the key). Similar statistical analysis with other characters will give the information.

Note that, since this is a statistical method, it's correctness has to be proved by decrypting the text and finding out if it makes sense.

4.4.1 Example of decrypting Affine Cipher

Suppose the ciphertext at hand is:

`FMXVEDKAPHFERBNDKRXRSREFNORUDSDKDVSHVUFEDKAPRKDLYFVLRHHRH` Suppose the affine cipher key is (a, b). The letter counts are `R: 8; D: 7; E, H, K: 5; F, S, V: 4`.

- First Guess: $R \leftarrow e, D \leftarrow t$
  - Thus $e_k(4) = 17, e_k(19) = 3$
  - So, $4a + b = 17;\ 19a + b = 3$
  - This gives $a = 6$, $b = 19$, since $gcd(6, 26) = 2$, so incorrect.
- Next Guess: $R \leftarrow e, E \leftarrow t$, the result is $a = 13$, not correct.
- Next Guess: $R \leftarrow e, H \leftarrow t$, the result is $a = 8$, not correct.
- Next Guess: $R \leftarrow e, K \leftarrow t$, the result is $a = 3,\ b = 5$.
  - The decrypted text is: `algorithmsarequitegeneraldefinitionsofarithmeticprocesses`

5 Conclusion

In today's lecture, we learnt about different models of cryptanalysis along with a polyalphabetic cipher.

Clearly, no matter how cleverly designed the cipher is, generally one can find ways to cryptanalyse the same.

So ultimately, it is up to the human beings to be more vigilant about the data they send.

References

[1] Nandhp. Letter frequency.

[2] Wikipedia contributors. Vigenère cipher — Wikipedia, the free encyclopedia, 2020. [Online; accessed 20-September-2020].