

Afleveringsopgave 21.c

Af Wel Rachid, 20051177

Først og fremmest starter vi med at lave en frequency analyse. Dvs vi ser hvor ofte bestemte bogstaver forekommer for derefter at bestemme hvilke bogstaver disse er.

Da jeg syntes det ville tage for lang tid at "tælle op" syntes jeg det kunne være lidt sjovere bare at lave et stykke kode der kunne gøre dette for mig. Resultatet af det blev:

Alfabetisk sorteret	Sorteret efter antal forekomster
Array ([a] => 13 [b] => 21 [c] => 32 [d] => 9 [e] => 13 [f] => 10 [g] => 0 [h] => 1 [i] => 16 [j] => 6 [k] => 20 [l] => 0 [m] => 0 [n] => 1 [o] => 2 [p] => 20 [q] => 4 [r] => 12 [s] => 1 [t] => 0 [u] => 6 [v] => 4 [w] => 0 [x] => 2 [y] => 1 [z] => 4)	Array ([l] => 0 [t] => 0 [w] => 0 [g] => 0 [m] => 0 [y] => 1 [s] => 1 [n] => 1 [h] => 1 [x] => 2 [o] => 2 [v] => 4 [z] => 4 [q] => 4 [u] => 6 [j] => 6 [d] => 9 [f] => 10 [r] => 12 [e] => 13 [a] => 13 [i] => 16 [p] => 20 [k] => 20 [b] => 21 [c] => 32)

Som det fremgår af højre side af tabellen er de to mest forekomne bogstaver c og b og jvf tabellen på side 27 er det hhv e og t.

Da vi har at gøre med en affine ciphertext kender vi encrypt algoritmen og decrypt algoritmen.

Encrypt foregår således: $E(x) = (ax + b) \bmod n$, hvor n er 26 her (da vi har 26 bogstaver i vores alfabet a...z)

Decrypt foregår således: $D(c) = a^{-1}(c - b) \bmod n$, hvor n er 26 her (da vi har 26 bogstaver i vores alfabet a...z)

a^{-1} er den modulære multiplikative inverse af a. Dvs at $aa^{-1} = 1$

For at finde a og b skal vi starte med at mappe vores bogstaver over til 26 forskellige tal... dvs fra 0...25 hvor a = 0 og 25 = z.

For at finde a og b skal vi løse følgende ligninger.

$$"e" a + b = "c" \Rightarrow 4a + b = 2$$

$$"t" a + b = "b" \Rightarrow 19a + b = 1$$

For at opfylde at $ax \equiv b \pmod n$ skal have en unique løsning $x \in Z_n$ for ethvert $b \in Z_n$, hvis og kun hvis $\gcd(a,n) = 1$

Det giver følgende valgmuligheder til a: 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23 og 25.

13 er ikke mulig da $\gcd(13,26)=2$

$$"e" a + b = "c" \Rightarrow 4a + b = 2$$

$$"t" a + b = "b" \Rightarrow 19a + b = 1$$

Det giver os $-15a = 1$

Herefter skal vi løse denne ligning med en ubekendt og det giver os et $a = 19$, da $"-15*19\%26 = 1"$

Vi kender nu a og skal derfor vælge b.

$$4*19 + b = 2$$

$$76 + b = 2$$

$$b = -74 \pmod n$$

$$b = 4$$

Vi kan tjekke ved at indsætte $(a,b) = (19,4)$ i ligningen $19a+b=1 \Rightarrow (19*19+4)\%26 = 1$

Da vi ikke har kendskab til nogen algoritmer (endnu) til at finde a^{-1} må vi i første omgang gætte os frem til dette. Det gøres ved trial and error.

$$19*i\%26 = 1, \text{ hvor } i = 1...26$$

Resultatet er at $i=11$ er et godt valg.

Vi har nu alt hvad vi skal bruge for at dekryptere koden.

Til at foretage selve dekrypteringen har jeg også valgt at computeren skal lave det, da det er sjovere.

Måden det sker på er ved at bruge $D(c) = a^{-1}(x - b) \bmod n$ til at vælge hvilket bogstav der oprindeligt var.

For at teste med de første par bogstaver i vores ciphertext (kqerej) indsætter vi følgende

$$D("k") = a^{-1}("k" - b) \bmod n \Rightarrow 11(10 - 4) \bmod 26 \Rightarrow 66 \bmod 26 \Rightarrow 14, \text{ hvor bogstav 14 er "o"}$$

$$D("q") = a^{-1}("q" - b) \bmod n \Rightarrow 11(16 - 4) \bmod 26 \Rightarrow 132 \bmod 26 \Rightarrow 2, \text{ hvor bogstav 2 er "c"}$$

$$D("e") = a^{-1}("e" - b) \bmod n \Rightarrow 11(4 - 4) \bmod 26 \Rightarrow 0 \bmod 26 \Rightarrow 0, \text{ hvor bogstav 14 er "a"}$$

$$D("r") = a^{-1}("r" - b) \bmod n \Rightarrow 11(17 - 4) \bmod 26 \Rightarrow 143 \bmod 26 \Rightarrow 13, \text{ hvor bogstav 13 er "n"}$$

$$D("e") = a^{-1}("e" - b) \bmod n \Rightarrow 11(4 - 4) \bmod 26 \Rightarrow 0 \bmod 26 \Rightarrow 0, \text{ hvor bogstav 0 er "a"}$$

Scriptet giver mig følgende cleartext (sepereret ved hver 5. Tegn)

OCANA DATER REDEN OSAIE UXTON FRONT ESTCE INTDE FLEUR ONSGL ORIEU XCART ONBRA SSAIT PORTE
RLEPE EILSA ITPOR TERLA CROIX TONHI STOIR EESTU NEEPO PEEDE SPLUS BRILL ANTSE XPLOI TSETT AVALE
URDEF OITRE MPEEP ROTEG ERANO SFOYE RSETN OSDRO ITS

Udførelse af kode kan findes på <http://rachid.dk/occur.php>

Selve kildekoden kan findes på <http://rachid.dk/occur.phps>