Pemecahan Sandi Caesar Dengan Bantuan Algoritma *String Matching* dan *Brute-Force*

Naufal Syifa Firdaus - 13521050
Program Studi Teknik Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13521050@std.stei.itb.ac.id

***Abstract*—Sandi Caesar adalah salah satu dari metode enkripsi teks yang paling sederhana dan populer. Penggunaannya dapat ditelusuri dari abad 58 sebelum masehi pada zaman dictator Julius Caesar untuk mengenkripsi pesan militer. Pada studi kali ini, akan dimulai dengan prinsip dasar sandi Caesar dan penggunaanya dalam sejarah. Lalu akan dibahas teori *string matching* dan berbagai metodenya. Perlu beberapa metode lain untuk memecahkan sandi Caesar diantaranya *Brute-Force* dan analisis frekuensi yang keduanya akan dibahas dalam makalah berikut. Sebagai hasil, studi ini akan mencoba untuk mengajukan sebuah metode dan serangkaian algoritma untuk memecahkan sandi Caesar dengan teori-teori fundamental yang akan dibahas dan juga menganalisis efektivitas dan efisiensi metode yang diajukan menggunakan studi kasus teoritis. Metode yang diajukan akan menggunakan teori analisis frekuensi dikombinasikan dengan algoritma *Brute-Force* dan *String Matching* untuk mempercepat komputasi. Lebih jauh lagi, studi ini akan menyentuh beberapa faktor yang memengaruhi performa pemecahan seperti Bahasa dan panjang teks.**

Harapannya hasil studi ini akan berkontribui pada bidang ilmu kriptografi dengan menawarkan metode yang dapat memecahkan sandi Caesar yang cukup cepat dan efisien. Selain itu juga, membuka bidang eksplorasi baru terkait teknik enkripsi dan deskripsi yang dapat membukan jalan untuk algoritma enkripsi modern.

***Keywords*—Caesar, Sandi, String Matching, Decode, Algoritma**

I. PENGENALAN

Sandi Caesar, nama lain *Caesar Box* atau *Caesar Cipher*, adalah salah satu sandi kuno yang sudah digunakan dari zaman dahulu. Pada awalnya diktator dan negarawan roma kuno, Julius Caesar, menggunakan sandi sebagai cara untuk merahasiakan isi dari surat yang dikirim untuk kepentingan militer dan pribadinya. Sebagai mana dikutip dari manuskrip "The Lives of The Twelve Caesar".

> *"There are extant some letters of his to the senate, written in a manner never practiced by any before him; for they are distinguished into pages in the form of a memorandum book whereas the consuls and commanders till then, used constantly in their letters to continue the line quite across the sheet, without any folding or distinction of pages. There are extant likewise some letters from him to Cicero, and others to his friends, concerning his domestic affairs; in which, if there was occasion for secrecy, he wrote in cyphers; that is, he used the alphabet in such a manner, that not a single word could be made out. The way to decipher those epistles was to substitute the fourth for the first letter, as d for a, and so for the other letters respectively."*
>
> *Suetonius, De vita Caesarum*

Namun diyakini bahwa Caesar bukan yang pertama kali menggunakan sandi yang menggantikan huruf dalam teks dengan menggeser susunan alfabetnya. Pada zamannya sandi ini quote dapat dikatakan tidak terpecahkan. Bukan hanya karena kebanyakan musuh roma saat itu buta huruf, tapi juga tidak adanya rekaman sejarah pada saat itu yang menyatakan terpecahkannya sandi Caesar dengan metode apapun. Rekaman sejarah paling awal yang ditemukan atas terpecahkannya sandi tersebut adalah pada abad kesembilan di Arab dengan metode analisis frekuensi. Huruf yang tergantikan tetap masih memiliki arti yang sama yang berarti dapat dipecahkan dengan mencocokannya dengan frekuensi penggunaan sebuah huruf dalam suatu bahasa tertentu. Walaupun begitu, sandi Caesar masih digunakan secara luas sampai hari ini. Sebagai contoh, pada Perang Dunia 1, tentara rusia menggunakan sandi Caesar untuk bertukar pesan antar tentara. Sekarang sandi Caesar banyak digunakan pada buku dan permainana anak-anak untuk mengenalkan generasi baru pada bidang ilmu kriptografi. Sandi ini juga berfungsi sebagai dasar untuk sandi yang lebih kompleks dan sistem enkripsi baru. Misalnya, sandi Vigenère, yang ditemukan pada abad ke-16, mengembangkan prinsip sandi Caesar dengan menggunakan kata kunci untuk menentukan nilai pergeseran variabel untuk setiap huruf.

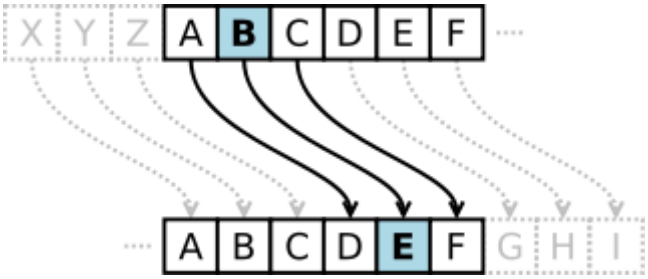
Pada saat ini, sandi Caesar tidak lagi digunakan untuk mengenkripsi teks yang memerlukan keamanan menengah keatas. Karena prinsip dasarnya yang sederhana dan metode pemecahannya sudah diketahui secara luas, sandi ini dinilai tidak bisa mengikuti kebutuhan keamanan data saat ini. Namun, sandi ini tetap menjadi sebuah bagian dari sejarah kriptografi dan memegang peranan penting terhadap perkembangan enkripsi umat manusia. Manusia berangkat dari sandi Caesar di masa lalu sampai sekarang, enkripsi publik yang tidak terpecahkan bahkan oleh komputer paling cepat sekalipun.

Pada studi kali ini akan dibahas metode-metode yang dapat dengan mudah memecahkan sandi Caesar dan mempercepat prosesnya menggunakan algoritma *Brute-Force* dan *String matching*. Studi kali akan menggabungkan metode yang telah ada sehingga menghasilkan suatu teori metode yang mampu memecahkan sandi Caesar secara cepat dan akurat.

II. DASAR TEORI

A. *Sandi Caesar*

Seperti yang telah disebutkan pada kutipan buku "The Lives of Twelve Caesar" pada bahasan sebelumnya, sandi Caesar adalah suatu teknik mengenkripsi sebuah teks dengan mengganti huruf dalam teks tersebut dengan huruf lain. Pada sandi ini, huruf digantikan dengan huruf lain pada urutan alfabet sesuai dengan jumlah pergeseran yang telah ditentukan dan relatif kepada urutan huruf yang digantikan. Sebagai contoh, untuk sandi Caesar dengan jumlah pergeseran 4 maka huruf A akan menjadi E dan B akan menjadi F. Jumlah pergeseran ini dapat berupa angka berarpapun yang telah disepakati oleh pengirim dan penerima pesan. Selain itu perlu ditentukan juga arah pergeserannya, kiri atau kanan.



Gambar I. Pergeseran Huruf denga jumlah 3. Sumber: [1]

Untuk mempermudah enkripsi sandi, dapat digunakan tabel yang mengandung huruf dan sandinya dalam satu kolom yang sama sebagai representasi visual dari sandi Caesar tersebut. Lebih jauh lagi, tabel yang digunakan dapat berbentuk sirkuler yang menggambarkan sifat sirkuler dari pergeseran sandi yang dibuat. Dibawah ini adalah contoh tabel sirkuler sandi Caesar dengan pergeseran empat kearah kanan, seperti yang dideskripsikan oleh Suetonius dalam biografinya tentang Julius Caesar.

Gambar II.Tabel sirkuler Caesar. Sumber:[1]

Tabel diatas dapat mempermudah enkripsi sandi Caesar dengan hanya melihat dan mencocokan alfabet yang diatas dengan sandi yang dibawah pada kolom yang sama. Sebagai contoh, teks pangram, kalimat yang mengandung semua huruf dalam alfabet, dan hasil sandi dibawah ini yang menggunakan tabel Caesar diatas.

- Teks: Saya lihat foto Hamengkubuwono XV bersama enam zebra purba cantik yang jatuh dari Aquarium
- Sandi: Wece pmlex jsxs Leqirkoyfyasrs BZ fivweqe ireq difve tyvfe gerxmo cerk nexyl hevm Euyevmyq

Dengan menganggap sebuah huruf sebagai representasi urutan alfabetnya, A = 0, B = 1 dan seterusnya, sandi Caesar dapat diformulasikan sebagai berikut.

$$E_n(x) = (x + n) \mod 26.$$
$$D_n(x) = (x - n) \mod 26.$$

Gambar III. Persamaan matematis sandi Caesar. Sumber: []

Persamaan E adalah untuk enkripsi sedangkan D adalah untuk dekripsi sandi dengan x urutan huruf yang diganti dan n jumlah pergeserannya.

Sandi Caesar memiliki banyak kelemahan yang membuatnya tidak aman untuk enkripsi informasi yang penting. Kelemahan tersebut adalah sebagai berikut:

- **Ruang Kunci Terbatas:** Cipher Caesar memiliki ruang kunci yang relatif kecil, dengan hanya 25 nilai pergeseran potensial untuk alfabet (selain pergeseran 0, yang akan menghasilkan teks biasa yang sama). Karena ruang kunci sandi yang kecil, serangan *brute-force*, di mana penyerang mencoba setiap nilai pergeseran yang dapat dibayangkan untuk memecahkan kode, dimungkinkan.
- **Analisis Frekuensi:** Bahasa Indonesia adalah salah satu dari beberapa bahasa di mana beberapa huruf lebih sering digunakan daripada yang lain. Seorang penyerang dapat memperkirakan arit dari sebuah sandi dengan memeriksa frekuensi huruf di dalamnya.
- **Kurangnya Lapisan Keamanan**: Tidak ada lapisan keamanan tambahan atau metode enkripsi canggih yang digunakan dalam sandi Caesar. Karena kemudahan analisis dan penguraiannya, enkripsi lebih rentan terhadap penyerangan.
- **Kurangnya kerahasiaan:** Enkripsi Caesar tidak memberikan kerahasiaan yang sangat tinggi pada teks biasa. Karena nilai pergeseran tetap dan seringkali kecil, penyerang dapat dengan cepat menguji semua nilai pergeseran potensial untuk menemukan pesan aslinya.

B. Brute Force

Algoritma sangat penting dalam bidang ilmu komputer dan kriptografi untuk menyelesaikan masalah yang kompleks. Algoritma *Brute-Force* adalah salah satu metode ampuh untuk menemukan solusi melalui pencarian menyeluruh. Algoritma *Brute-Force*, atau disebut juga sebagai algoritma *Exhaustive Search*, adalah metode mudah untuk memecahkan masalah yang memerlukan pengujian mendalam setiap jawaban yang berpotensial sampai jawaban yang paling tepat ditemukan

(solusi paling optimal). Dalam algoritma ini semua kemungkinan solusi diperiksa dan dievaluasi kebenarannya dalam lingkup masalah yang diberikan.

Pendekatan brute force digunakan di berbagai bidang, termasuk enkripsi, masalah pengoptimalan, dan kombinatorial. *Brute-Force* berfungsi sebagai pendekatan mendasar dalam kriptografi untuk memecahkan sandi substitusi sederhana dan sandi Caesar.

Dengan menganalisis setiap kombinasi atau permutasi solusi yang berpotensial, *Brute-Force* dapat digunakan untuk memecahkan masalah pengoptimalan. Meskipun mahal secara komputasi, strategi ini memastikan hasil terbaik bila digunakan pada masalah yang sederhana. Selain itu, dengan melihat setiap kemungkinan kombinasi, pendekatan brute force sering digunakan untuk menyelesaikan teka-teki kombinatorial seperti Sudoku atau Traveling Salesman Problem.

Algoritma brute force memiliki sejumlah keunggulan yang kuat. Pertama, pemrogram yang tidak berpengalaman dapat menggunakannya karena secara teori mudah dan sederhana untuk diterapkan. Kedua, solusi yang dihasilkan dapat dipastikan adalah solusi paling optimal jika tersedia waktu dan sumber daya yang memadai. Ketiga, dapat diadaptasi dan dapat digunakan untuk menyelesaikan berbagai masalah. Teknik *Brute-Force* dapat diaplikasikan dengan cepat karena tidak memerlukan informasi awal tentang masalah yang diberikan.

C. *String Matching: Boyer-Moore*

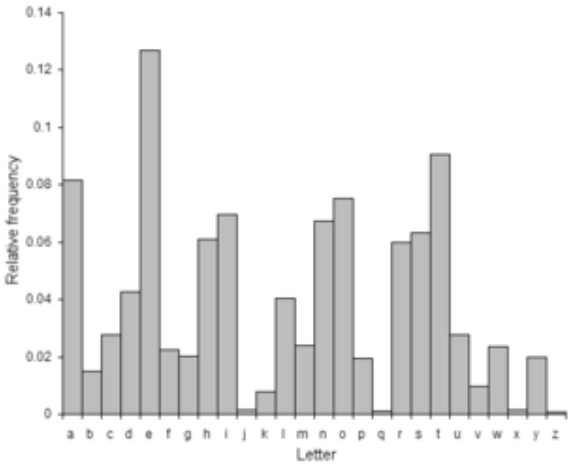
Metode dengan efektivitas tinggi untuk menemukan contoh pola di dalam teks atau string yang lebih panjang adalah algoritma pencocokan string. Metode Boyer-Moore adalah salah satu teknik pencocokan string yang ampuh dan efektif dalam pencarian pola. Metode Boyer-Moore, dibuat pada tahun 1977 oleh Robert S. Boyer dan J. Strother Moore, sangat meningkatkan keefektifan pencocokan string. Algoritma ini mengurangi jumlah operasi yang diperlukan dengan melewatkan perbandingan yang tidak berguna dan menggeser pola dengan jarak sesuai yang telah ditentukan.

Algoritma menempatkan kemunculan paling kanan karakter yang tidak cocok dalam pola dan memindahkan pola sehingga karakter ini sejajar dengan karakter teks yang cocok. Dengan demikian, ini menghilangkan perbandingan yang tidak berguna dan meningkatkan keefektifan umum algoritma. Berikut adalah langkah-langkah algoritma Boyer-Moore.

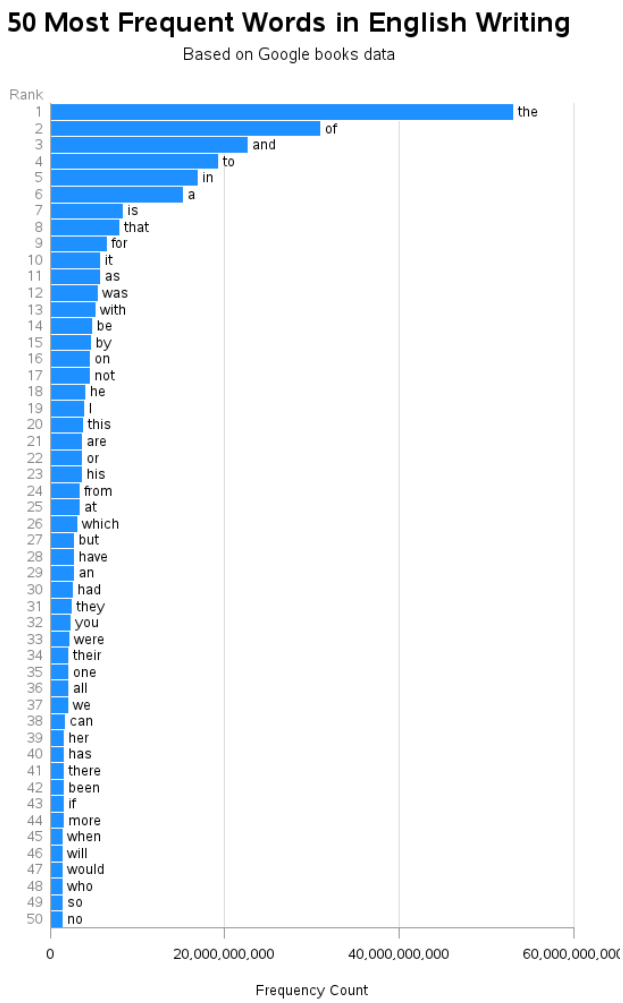
1. Tetapkan teks dan pola yang ingin Anda cari terlebih dahulu.
2. Tentukan panjang teks dan polanya.
3. Untuk menyimpan indeks paling kanan dari setiap karakter dalam pola, buat tabel karakter
4. Tetapkan semua karakter ke -1 dalam tabel karakter, yang umumnya adalah berupa larik berukuran 256 (dengan karakter ASCII).
5. Dari kiri ke kanan, ulangi urutannya.
6. Perbarui entri basis data karakter buruk untuk setiap karakter yang ditemui dengan indeks paling kananya.
7. Mulailah mencari pola dengan menyelaraskan ujung kanan pola dengan posisi yang sesuai di teks.
8. Bandingkan karakter pola dan bagian teks yang disejajarkan dari kanan ke kiri.
9. Jika terjadi ketidaksesuaian:
    a. Ambil karakter yang tidak cocok dari teks.
    b. Cari tabel karakter untuk menemukan kemunculan paling kanan dari karakter yang tidak cocok di dalam pola.
    c. Hitung jarak pergeseran dengan mengurangi indeks paling kanan dari indeks saat ini.
    d. Geser pola ke kanan dengan jarak pergeseran yang dihitung.
    e. Lanjutkan membandingkan karakter dari kanan ke kiri.
10. Jika ditemukan kecocokan:
    a. Pindah ke kiri pada pola dan teks untuk melanjutkan perbandingan.
11. Ulangi langkah 8-9 hingga seluruh teks telah dicari atau polanya ditemukan.

D. Analisis Frekuensi

Analisis frekuensi adalah metode yang efektif untuk memecahkan atau mendekripsi sebuah enkripsi. Metode ini membuat prediksi informasi tentang isi atau kunci enkripsi dengan menggunakan statistik suatu bahasa, seperti frekuensi kemunculan huruf atau pengelompokan huruf.. Prinsip dasar analisis frekuensi adalah bahwa berbagai huruf atau kombinasi huruf memiliki peluang berbeda untuk muncul dalam bahasa tertentu. Misalnya, huruf "E" paling sering digunakan dalam bahasa Inggris, diikuti huruf "T", "A", dan seterusnya. Dimungkinkan untuk menarik kesimpulan tentang bahasa enkripsi tersebut dan bahkan menentukan metode atau enkripsi yang digunakan dengan melihat frekuensi huruf dalam sebuah teks enkripsi.



Gambar VI. Frekuensi Kemunculan Huruf Inggris.
Sumber[1]



Gambar VI. Frekuensi Kemunculan kata Inggris.
Sumber[https://blogs.sas.com/content/sastraining/files/2015/11/word_frequency.png]

III. Metode Pemecahan

Sandi Caesar dapat dengan mudah dipecahkan dengan menguji semua nilai pergeseran yang mungkin, *brute-force*. Namun, mungkin sulit untuk memilih dekripsi yang tepat dari 25 opsi. Oleh karena itu metode *string matching* dapat ditambahkan ke pendekatan *brute-force* untuk meningkatkan akurasi decoding sandi Caesar. Dengan menguji setiap nilai pergeseran, pendekatan *brute-force* untuk memecahkan sandi Caesar menciptakan teks bisa. Setiap huruf dalam teks enkripsi digeser dalam alfabet dengan iterasi melalui jumlah pergeseran dari 0 sampai 25. Ada 25 kemungkinan teks hasil sebagai hasil dari prosedur ini, satu untuk setiap nilai pergeseran.

Untuk meningkatkan akurasi penguraian sandi Caesar, dapat menggunakan daftar kata populer dalam bahasa enkripsi dan membandingkannya dengan teks biasa yang dihasilkan. Metode ini berdasarkan teori analisis frekuensi yang menyatakan bahwa beberapa kata lebih sering digunakan daripada yang lain dalam suatu bahasa.

Pertama, harus diperoleh daftar istilah yang paling sering digunakan dalam bahasa enkripsi. Kamus atau basis data frekuensi kata dapat digunakan untuk membuat daftar ini. Istilah-istilah dalam daftar harus sering digunakan dan harus diatur dalam frekuensi dari paling sering dipakai ke paling jarang dipakai (menurun).

Lalu bandingkan setiap dekripsi yang dihasilkan algoritma *Brute-Force* dengan daftar istilah populer. Teks yang didekripsi dibandingkan dengan kata-kata dalam daftar, dan kecocokan dihitung. Prosedur ini membantu menemukan kemungkinan dekripsi yang cocok dengan bahasa yang diprediksi.

Dengan mengevaluasi jumlah kecocokan untuk setiap dekripsi potensial,dapat ditentukan dekripsi yang menunjukkan jumlah kecocokan tertinggi dengan kata-kata populer. Dekripsi khusus ini dianggap paling dekat dengan teks yang dapat dibaca manusia dan paling mungkin menjadi solusi yang tepat.

Metode pencocokan pola menawarkan peningkatan yang untuk prosedur dekripsi sandi caesar brute force. Penulis dapat menentukan mana dari berbagai dekripsi yang paling mungkin dengan mempertimbangkan pola dan struktur bahasa yang unik. Dengan menggunakan metode ini, sandi Caesar dapat dipe dipecahkan dengan lebih cepat dan akurat.

IV. Studi Kasus

Bayangkan sebuah situasi di mana pesan rahasia yang dienkripsi sandi Caesar adalah "Wece pmlex jsxs Leqirkoyfyasrs BZ fivweqe ireq difve tyvfe gerxmo cerk nexyl hevm Euyevmyq " adalah ciphertext yang diberikan. Pesan akan didekode menggunakan pendekatan brute force, yang akan menghasilkan semua dekripsi potensial dan menguji setiap nilai pergeseran antara 0 dan 25. Namun, untuk menemukan jawaban yang paling mungkin, akan digunakan algoritma pencocokan pola Boyer Moore sebagai kebalikan dari pemeriksaan individual setiap dekripsi potensial. Langkah-langkahnya adalah sebagai berikut.

1. Hasilkan dekripsi potensial: Iterasi melalui semua kemungkinan nilai pergeseran, dekripsi ciphertext menggunakan algoritma Brute force.
2. Buat daftar kata populer: Dapatkan daftar kata yang sering digunakan dalam bahasa target.
3. Bandingkan dekripsi dengan daftar kata: Untuk setiap potensi dekripsi, hitung jumlah kecocokan dengan daftar kata populer.
4. Pilih dekripsi dengan jumlah kecocokan tertinggi: Identifikasi potensi dekripsi yang menunjukkan kecocokan paling banyak dengan kata-kata populer. Dekripsi ini dianggap paling dekat dengan teks yang dapat dibaca manusia dan kemungkinan besar merupakan solusi yang tepat.

Hasil dari pemecahan algoritma brute force adalah sebagai berikut.

1. Vdbd olkdw irwr Kdphqjnxexzrqr AY ehuvdpd hqdp cheud sxued fdqwln bdqj mdwxk gdul Dtxdulxp
2. Ucac nkjcv hqvq Jcogpimwdwyqpq ZX dgtucoc gpco bgdtc rwtdc ecpvkm acpi lcvwj fctk Cswctkwo
3. Tbzb mjibu gpup Ibnfohlvcvxpop YW cfstbnb fobn afcsb qvscb dboujl zboh kbuvi ebsj Brvbsjv.
4. Saya lihat foto Hamengkubuwono XV bersama enam

zebra purba cantik yang jatuh dari Aquarium

5. Rzxz khgzs ensn Gzldmfjtatvnmn WU adqrzlz dmzl ydaqz otqaz bzmshj xzmf izstg czqh Zptzqhtl
6. Qywy jgfyr dmrm Fykcleiszsumlm VT zcpqyky clyk xczpy nspzy aylrgi wyle hyrsf bypg Yosypgs
7. Pxvx ifexq clql Exjbkdhryrtlkl US ybopxjx bkxj wbyox mroyx zxkqfh vxkd gxqre axof XnrxofrjAsdf
8. Owuw hedwp bkpk Dwiajcgqxqskjk TR xanowiw ajwi vaxnw lqnxw ywjpeg uwjc fwpqd zwne Wmqwneqi
9. Nvtv gdcvo ajoj Cvhzibfpwprjij SQ wzmnvhv zivh uzwmv kpmwv xviodf tvib evopc yvmd Vlpvmdph
10. Musu fcbun zini Bugyhaeovoqihi RP vylmugu yhug tyvlu jolvu wuhnce suha dunob xulc Ukoulcog
11. Ltrt ebatm yhmh Atfxgzdnunphgh QO uxkltft xgtf sxukt inkut vtgmbd rtgz ctmna wtkb TjntkbnfDfasd
12. Ksqs dazsl xglg Zsewfycmtmogfg PN twjkses wfse rwtjs hmjts usflac qsfy bslmz vsja Simsjame
13. Jrpr czyrk wfkf Yrdvexblslnfef OM svijrdr verd qvsir glisr trekzb prex arkly uriz Rhlrizld
14. Iqoq byxqj veje Xqcudwakrkmede NL ruhiqcq udqc purhq fkhrq sqdjya oqdw zqjkx tqhy Qgkqhykc
15. Hpnp axwpi udid Wpbtcvzjqjldcd MK qtghpbp tcpb otqgp ejgqp rpcixz npcv ypijw spgx Pfjpgxjb
16. Gomo zwvoh tchc Voasbuyipikcbc LJ psfgoao sboa nspfo difpo qobhwy mobu xohiv rofw Oeiofwia
17. Fnln yvung sbgb Unzratxhohjbab KI orefnzn ranz mroen cheon pnagvx lnat wnghu qnev Ndhnevhz
18. Emkm xutmf rafa Tmyqzswgngiaza JH nqdemym qzmy lqndm bgdnm omzfuw kmzs vmfgt pmdu Mcgmdugy
19. Dljl wtsle qzez Slxpyrvfmfhzyz IG mpcdlxl pylx kpmcl afcml nlyetv jlyr ulefs olct Lbflctfx
20. Ckik vsrkd pydy Rkwoxquelegyxy HF lobckwk oxkw jolbk zeblk mkxdsu ikxq tkder nkbs Kaekbsew
21. Bjhj urqjc oxcx Qjvnwptdkdfxwx GE knabjvj nwjv inkaj ydakj ljwcrt hjwp sjcdq mjar Jzdjardv
22. Aigi tqpib nwbw Piumvoscjcewvw FD jmzaiui mviu hmjzi xczji kivbqs givo ribcp lizq Iycizqcu
23. Zhfh spoha mvav Ohtlunrbibdvuv EC ilyzhth luht gliyh wbyih jhuapr fhun qhabo khyp Hxbhypbt
24. Ygeg rongz luzu Ngsktmqahacutu DB hkxygsg ktgs fkhxg vaxhg igtzoq egtm pgzan jgxo Gwagxoas
25. Xfdf qnmfy ktyt Mfrjslpzgzbtst CA gjwxfrf jsfr ejgwf uzwgf hfsynp dfsl ofyzm ifwn Fvzfwnzr

Setelah menerapkan teknik pencocokan pola ke ciphertext yang diberikan, dekripsi dengan jumlah kecocokan tertinggi diidentifikasi sebagai solusi yang paling mungkin. Yang adalah iterasi keempat.

4. Saya lihat foto Hamengkubuwono XV bersama enam zebra purba cantik yang jatuh dari Aquarium

V. KESIMPULAN

Kesimpulannya, studi membahas penggunaan teknik pencocokan pola untuk meningkatkan pemecahan sandi Caesar. Dengan menggabungkan pencocokan pola dan *brute-force*, khususnya melalui perbandingan teks yang dihasilkan dengan daftar kata-kata populer dalam bahasa target, akurasi dekripsi dapat ditingkatkan.

Pendekatan pencocokan pola berkontribusi pada bidang kriptografi dengan memberikan penilaian dekripsi potensial yang lebih baik. Dengan mempertimbangkan pola linguistik, frekuensi kata, dan struktur bahasa yang banyak digunakan, ini meningkatkan kemungkinan memilih dekripsi yang tepat di antara kemungkinan solusi. Pendekatan ini menghemat waktu dan tenaga dengan menghilangkan kebutuhan untuk evaluasi manual dari setiap potensi dekripsi.

Namun, penting untuk diketahui bahwa keefektifan pencocokan pola bergantung pada keakuratan dan kelengkapan daftar kata populer dalam bahasa target. Jika daftar tidak lengkap atau tidak mewakili bahasa secara memadai, hasilnya dapat tidak maksimal. Selain itu, keberhasilan pencocokan pola sangat bergantung pada karakteristik linguistik dari bahasa yang didekripsi. Struktur bahasa yang tidak biasa atau penggunaan kata yang tidak biasa dapat menyebabkan kesalahan dalam proses dekripsi.

Sangat penting untuk diingat bahwa esai didasarkan pada pengetahuan dan pengalaman penulis yang terbatas. Pelaksanaan metode yang disarankan mungkin dibatasi oleh tingkat kemahiran penulis dalam analisis bahasa, pencocokan pola, dan kriptografi. Untuk dekripsi sandi Caesar dan algoritma enkripsi lainnya yang lebih akurat dan tepat, disarankan untuk berkonsultasi dengan profesional dan menggunakan alat dan prosedur kriptografi yang telah teruji.

Secara keseluruhan, studi ini menekankan potensi metode pencocokan pola untuk meningkatkan dekripsi Caesar cipher. Ini meningkatkan pemilihan dekripsi yang sesuai dengan mempertimbangkan pola linguistik dan frekuensi kata, sebagai alat yang berguna untuk kriptografer dan peneliti di bidang keamanan informasi.

REFERENCES

[1] Tranquillus, C.S. (121AD) 'LVI', in De vita Caesarum (lit. 'On the Life of the Caesars'). Sumber: https://www.gutenberg.org/files/6400/6400-h/6400-h.htm

[2] Caesar Cipher, McGill University, sumber: https://www.cs.mcgill.ca/~rwest/wikispeedia/wpcd/wp/c/Caesar_cipher.htm#:~:text=The%20Caesar%20cipher%20is%20named%20after%20Julius%20Caesar%2C%20who%2C%20according,word%20could%20be%20made%20out.

[3] Caesar Cipher, Wahington University, sumber: https://courses.cs.washington.edu/courses/cse490h1/19wi/exhibit/artifacts/crypto.pdf

[4] K. H. Rosen. "Graph" in *Discrete Mathematics and Its Application*, 7th ed. New York, NY, USA, 2021, ch. 10, sec. 10.1, pp. 641–650.

[5] Pencocokan String, Rinaldi Munir, sumber: https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf

[6] Brute-Force, Rinaldi Munir, https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-(2022)-Bag1.pdf

[7] https://blogs.sas.com/content/sastraining/files/2015/11/word_frequency.png

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis adalah tulisan saya sendiri, bukan saduran atau terjemahan dari makalah orang lain, dan bukan plagiarisme.

Bandung, 22 Mei, 2023

Naufal Syifa Firdaus (13521050)