

Aplikasi Teori Bilangan

✧ *ISBN (International Book Serial Number)*

✧ Fungsi *hash*

✧ Kriptografi

✧ Pembangkit bilangan acak-semu

✧ dll

ISBN

✧ Kode ISBN terdiri dari 10 karakter, biasanya dikelompokkan dengan spasi atau garis, misalnya 0–3015–4561–9.

✧ ISBN terdiri atas empat bagian kode:

- kode yang mengidentifikasikan bahasa,
- kode penerbit,
- kode unik untuk buku tersebut,
 - karakter uji (angka atau huruf X (=10)).

✧ Karakter uji dipilih sedemikian sehingga

$$\sum_{i=1}^{10} ix_i \equiv 0 \pmod{11}$$

$$\sum_{i=1}^9 ix_i \pmod{11} = \text{karakter uji}$$

✧ Contoh: ISBN 0–3015–4561–8

0 : kode kelompok negara berbahasa Inggris,

~~3015~~ : ~~kode penerbit~~ -----


4561 : kode unik buku yang diterbitkan

8 : karakter uji.

Karakter uji ini didapatkan sebagai berikut:

$$1 \cdot 0 + 2 \cdot 3 + 3 \cdot 0 + 4 \cdot 1 + 5 \cdot 5 + 6 \cdot 4 + \\ 7 \cdot 5 + 8 \cdot 6 + 9 \cdot 1 = 151$$

✧ Jadi, karakter ujinya adalah $151 \bmod 11 = 8$.



Catatlah bahwa untuk kode ISBN ini,

$$\sum_{i=1}^{10} ix_i = \sum_{i=1}^9 ix_i + 10x_{10} = 151 + 10 \cdot 8 = 231$$

dan $231 \bmod 11 = 0$ atau $231 \equiv 0 \pmod{11}$.

Tugas

✦ Buatlah sebuah program (C) untuk mencari karakter uji dari

- ✦ ISBN

- ✦ EAN-13

✦ dicetak, didalamnya diberi komentar NIM dan Nama Mahasiswa.

Soal 23 dan 24 (Buku halaman 223)

- ✧ Sembilan angka pertama dari kode ISBN sebuah buku adalah 0-07-053965. Tentukan karakter uji untuk buku ini!
- ✧ ISBN sebuah buku mengenai algoritma adalah 0-201-57P85-1, yang dalam hal ini P adalah angka. Berapa nilai P?

Fungsi *Hash*

✧ Tujuan: pengalamatan di memori

✧ Bentuk: $h(k) = k \bmod m$

- m : jumlah lokasi memori yang tersedia
- k : kunci (*integer*)
- $h(k)$: lokasi memori untuk *record* dengan kunci k

$$h(5) = 5 \bmod 11 = 5$$

0 1 2 3 4 5 6 7 8 9 10

-
- ✧ Kolisi (*collision*) terjadi jika fungsi *hash* menghasilkan nilai h yang sama untuk k yang berbeda.
 - ✧ Jika terjadi kolisi, cek elemen berikutnya yang kosong.
 - ✧ Fungsi *hash* juga digunakan untuk me-*locate* elemen yang dicari.



✧ Area parkir 0-30

✧ Kunci= 327, 100, 121, 310, 414, 110, 017


- ✧ $327 \bmod 31 = 17$
- ✧ $100 \bmod 31 = 7$
- ✧ $121 \bmod 31 = 28$
- ✧ $310 \bmod 31 = 0$
- ✧ $414 \bmod 31 = 11$
- ✧ $110 \bmod 31 = 17$ karena sudah terpakai disimpan (18)
- ✧ $017 \bmod 31 = 17$ karena 17, 18 sudah dipakai, maka (19)

Latihan

- ✧ Tunjukkan bagaimana sekumpulan data dengan kunci
- ✧ 714, 631, 26, 373, 775, 906, 509, 2032, 42, 4, 136, 1028
- ✧ Ditempatkan di dalam memori dengan fungsi hash $h(k) = k \bmod 17$

Kriptografi

- **Pesan:** data atau informasi yang dapat dibaca dan dimengerti maknanya.
Nama lain: **plainteks** (*plaintext*)
- Pesan dapat berupa: teks, gambar, audio, video.
- Pesan ada yang dikirim atau disimpan di dalam media penyimpanan.



✱ **Cipherteks** (*ciphertext*): pesan yang telah disandikan sehingga tidak memiliki makna lagi.

Tujuan: agar pesan tidak dapat dimengerti maknanya oleh pihak lain.

✱ Cipherteks harus dapat diubah kembali ke plainteks semula




Contoh:

Plainteks:

culik anak itu jam 11 siang

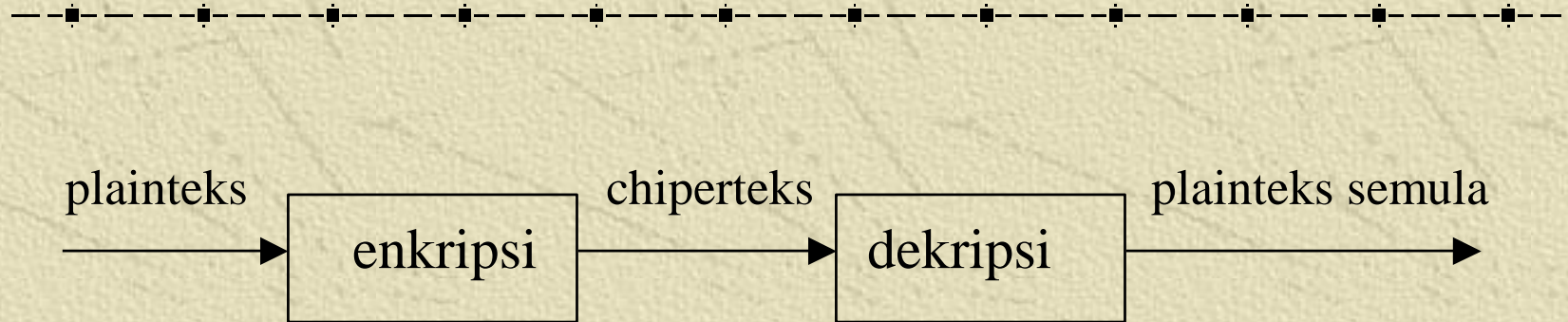
Cipherteks:

t^\$gfUi89rewoFpfdWqL:p[uTcxZ



✦ **Enkripsi** (*encryption*): proses menyandikan plainteks menjadi ciphertek.

✦ **Dekripsi** (*decryption*): Proses mengembalikan cipherteks menjadi plainteksnya.



Gambar 1.1 Enkripsi dan dekripsi

✧ Kriptografi (*cryptography*)

✧ Dari Bahasa Yunani yang artinya “*secret writing*”

✧ Definisi: kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan.

✧ **Algoritma kriptografi (*cipher*)**

- aturan untuk enkripsi dan dekripsi
- fungsi matematika yang digunakan untuk enkripsi dan dekripsi.

✧ **Kunci:** parameter yang digunakan untuk transformasi *enciphering* dan *dechiphering*

✧ Kunci bersifat rahasia, sedangkan algoritma kriptografi tidak rahasia

Sejarah Kriptografi

- ✦ Sudah digunakan di Yunani 400 BC
- ✦ Alat yang digunakan: *scytale*




Gambar 1.2 *Scytale*

Aplikasi Kriptografi

1. Pengiriman data melalui saluran komunikasi

(data encryption on motion).

2. Penyimpanan data di dalam *disk storage*
(data encryption at rest)

- 
-
- ✦ Data ditransmisikan dalam bentuk ciperteks. Di tempat penerima ciperteks dikembalikan lagi menjadi plainteks.
 - ✦ Data di dalam media penyimpanan komputer (seperti *hard disk*) disimpan dalam bentuk ciperteks. Untuk membacanya, hanya orang yang berhak yang dapat mengembalikan ciperteks menjadi plainteks.

Contoh enkripsi pada dokumen

Plainteks (plain.txt):

Ketika saya berjalan-jalan di pantai,
saya menemukan banyak sekali kepiting
yang merangkak menuju laut. Mereka
adalah anak-anak kepiting yang baru
menetas dari dalam pasir. Naluri
mereka mengatakan bahwa laut adalah
tempat kehidupan mereka.

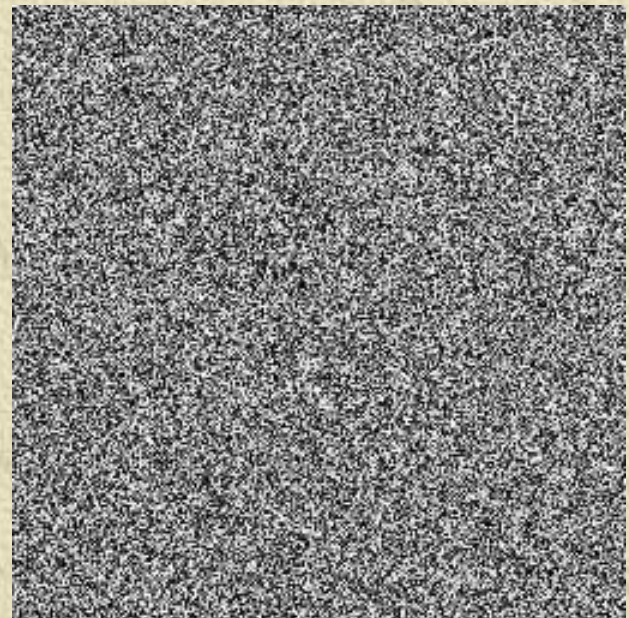
Cipherteks (cipher.txt):

Ztâxzp/épêp/qtüyp{p}<yp{p}/sx/□p}âpx;
épêp/|t}t|äzp}/qp}êpz/étzp{x/z t□xâx
}v êp}v/|tüp}vzpz/|t}äyâ/{päâ=/\tütz
p psp{pw/p}pz<p}pz/z t□xâx}v/êp}
v/qpüä |t}tâpé/spüx/sp{p|/□péxü=/
p{äüx |ttüzp/|t}vpâpzp}/qpwâp/{päâ
/psp{pw â t|□pâ/ztwxsä□p}/|tützp=

Plainteks (lena.bmp):



Cipherteks (lena2.bmp):



Plainteks (siswa.dbf):

NIM	Nama	Tinggi	Berat
000001	Elin Jamilah	160	50
000002	Fariz RM	157	49
000003	Taufik Hidayat	176	65
000004	Siti Nurhaliza	172	67
000005	Oma Irama	171	60
000006	Aziz Burhan	181	54
000007	Santi Nursanti	167	59
000008	Cut Yanti	169	61
000009	Ina Sabarina	171	62

Cipherteks (siswa2.dbf):

NIM	Nama	Tinggi	Berat
000001	tüp}vzpz/ t}äyâ/{ää	äzp}	épêp
000002	t}tâpé/spüx/sp	péxü=	ztwxsä□
000003	ât □pâ/ztwxsä□p}/	}/ tü	spüx/
000004	épêp/ t}t äzp}/qpêpz	qp}êpz	wxsä
000005	étzp{x/zt□xâx}v êp}	pää/psp	étzp{
000006	spüx/sp{p /□péxü=/>]	xâx}v	ttüzp/
000007	Ztâxzp/épêp/qtüypp}<	äzp}	}äyâ/{
000008	qpwwâp/{pää/psp{pw	Ztwxs	xâx}v
000009	}t äzp}/qp}êpz/ép{	qp}êp	äzp}/qp

Keterangan: hanya *field* Nama, Berat, dan Tinggi yang dienkrpsi.

Notasi Matematis

Misalkan:

C = ciperteks


P = plainteks dilambangkan

Fungsi enkripsi E memetakan P ke C ,

$$E(P) = C$$

Fungsi dekripsi D memetakan C ke P ,

$$D(C) = P$$



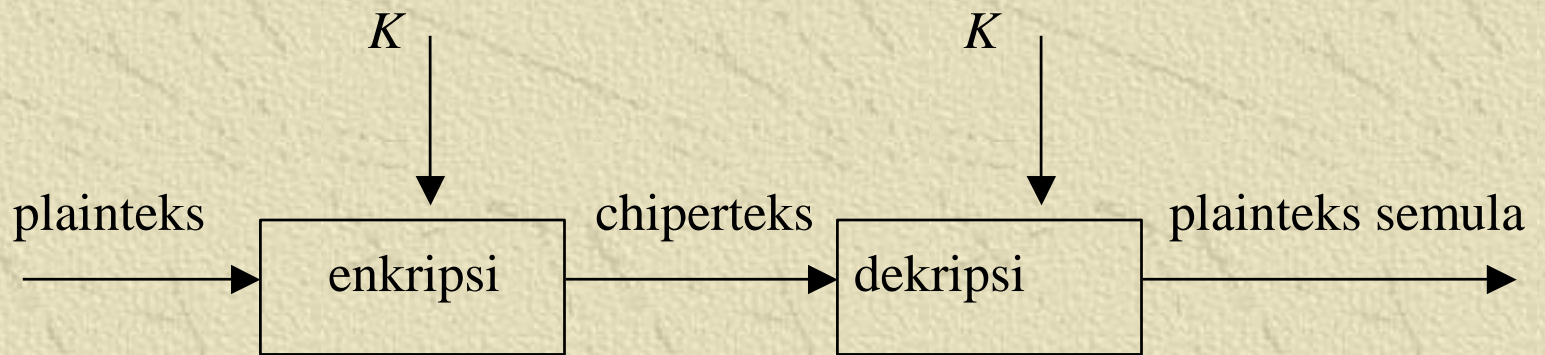
Dengan menggunakan kunci K , maka fungsi enkripsi dan dekripsi menjadi

$$E_K(P) = C$$

$$D_K(C) = P$$

dan kedua fungsi ini memenuhi

$$D_K(E_K(P)) = P$$



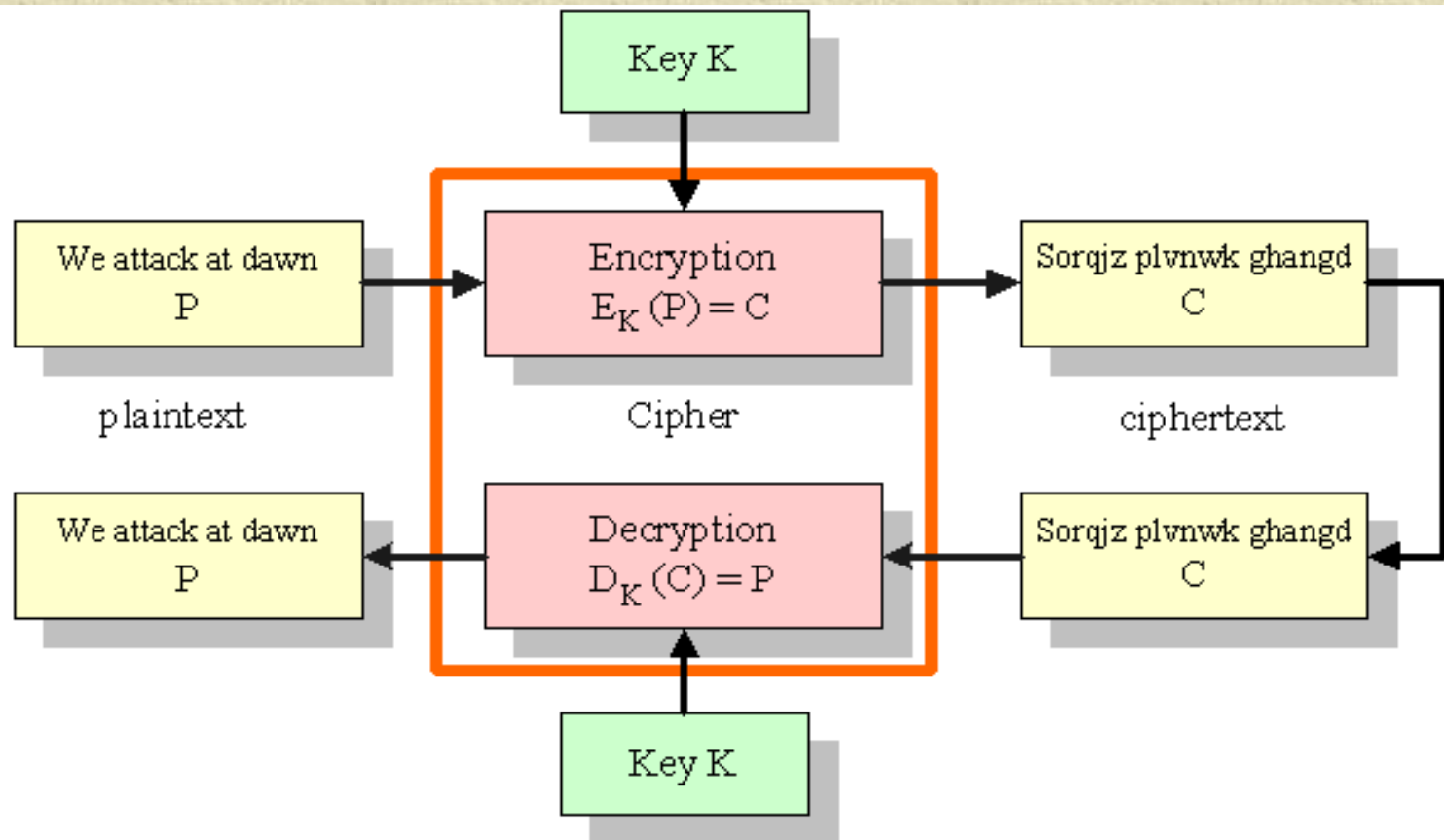
Gambar 1.3 Enkripsi dan dekripsi dengan kunci

✧ Jika kunci enkripsi sama dengan kunci dekripsi, maka sistem kriptografinya disebut **sistem simetri** atau **sistem konvensional**.

✧ Algoritma kriptografinya disebut algoritma simetri atau algoritma konvensional .

✧ Contoh algoritma simetri:

- *DES (Data Encryption Standard)*
- Rijndael



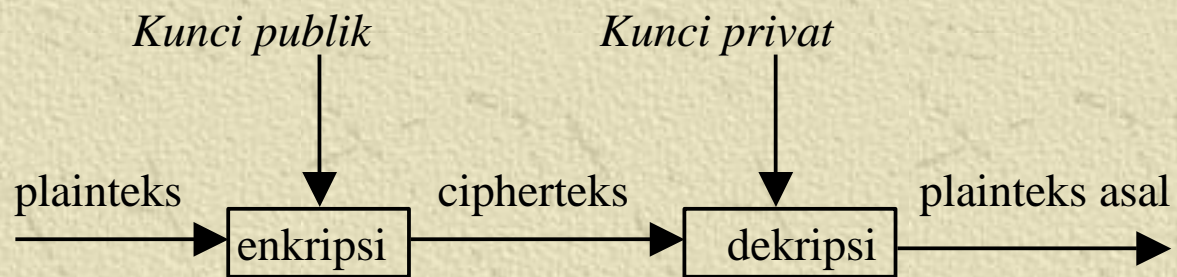
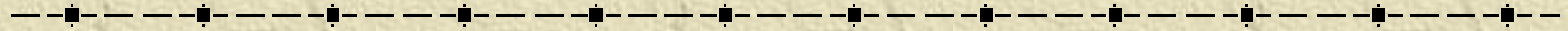
Skema algoritma simetri

- ✧ Jika kunci enkripsi tidak sama dengan kunci dekripsi, maka sistem kriptografinya disebut **sistem nirsimetri** (*asymmetric system*)

-
- ✧ Nama lain: **sistem kriptografi kunci-publik**

karena, kunci enkripsi bersifat publik (*public key*) sedangkan kunci dekripsi bersifat rahasia (*private key*).

- ✧ Pengirim pesan menggunakan kunci publik si penerima pesan untuk mengenkripsi pesan
- ✧ Penerima pesan mendekripsi pesan dengan kunci privatnya sendiri.
- ✧ Contoh algoritmai: *RSA*



Caesar Cipher

✠ Tiap huruf alfabet digeser 3 huruf ke kanan

p_i	:	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
c_i	:	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Contoh:

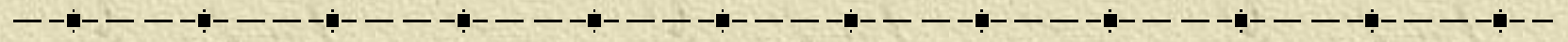
Plainteks: AWASI ASTERIX DAN TEMANNYA OBELIX

Cipherteks: **DZDVL DVWHULA GDQ WHPDQQBA REHOLA**

✦ Misalkan $A = 0, B = 1, \dots, Z = 25$, maka secara matematis caesar *cipher* dirumuskan sebagai berikut:


Enkripsi: $c_i = E(p_i) = (p_i + 3) \bmod 26$

Dekripsi: $p_i = D(c_i) = (c_i - 3) \bmod 26$



$$\begin{aligned} p_1 = \text{'A'} = 0 &\rightarrow c_1 = E(0) = (0 + 3) \bmod 26 = 3 = \text{'D'} \\ p_2 = \text{'W'} = 22 &\rightarrow c_2 = E(22) = (22 + 3) \bmod 26 = 25 = \text{'Z'} \\ p_3 = \text{'A'} = 0 &\rightarrow c_3 = E(0) = (0 + 3) \bmod 26 = 3 = \text{'D'} \\ p_4 = \text{'S'} = 18 &\rightarrow c_4 = E(18) = (18 + 3) \bmod 26 = 21 = \text{'V'} \\ \text{dst...} \end{aligned}$$

✧ Alternatif lain: gunakan tabel substitusi



✦ Jika pergeseran huruf sejauh k , maka:

Enkripsi: $c_i = E(p_i) = (p_i + k) \bmod 26$

Dekripsi: $p_i = D(c_i) = (c_i - k) \bmod 26$

k = kunci rahasia

```

program enkripsi;
{ Mengenripsi berkas 'plain.txt'
  menjadi 'cipher.txt' dengan
  metode caesar cipher }
uses
  crt;
var
  F1, F2 : text;
  p : char;
  c : integer;
  k : integer;

begin
  assign(F1, 'plain.txt');
  reset(F1);

  assign(F2, 'cipher.txt');
  rewrite(F2);

  write('k = ?'); readln(k);
  while not EOF(F1) do
    begin
      while not EOLN(F1) do
        begin
          read(F1, p);
          c := (ord(p) + k) mod 256;
          write(F2, chr(c));
        end;
      readln(F1);
      writeln(F2);
    end;
  close(F1);
  close(F2);
end.

```

```

program dekripsi;
{ Mendekripsi berkas 'cipher.txt'
  menjadi 'plain2.txt' dengan
  metode caesar cipher }
uses
  crt;
var
  F1, F2 : text;
  p : char;
  c : integer;
  k : integer;

begin
  assign(F1, 'cipher.txt');
  reset(F1);

  assign(F2, 'plain2.txt');
  rewrite(F2);

  write('k = ?'); readln(k);
  while not EOF(F1) do
    begin
      while not EOLN(F1) do
        begin
          read(F1, p);
          c := (ord(p) - k) mod 256;
          write(F2, chr(c));
        end;
      readln(F1);
      writeln(F2);
    end;
  close(F1);
  close(F2);
end.

```


Algoritma RSA

- ✧ Ditemukan oleh tiga peneliti dari *MIT* (*Massachussets Institute of Technology*), yaitu Ron Rivest, Adi Shamir, dan Len Adleman, pada tahun 1976.
- ✧ Termasuk algoritma kriptografi nirsimetri.



✦ Setiap pengguna mempunyai sepasan kunci:

1. Kunci publik: untuk enkripsi
2. Kunci privat: untuk dekripsi

✦ Kunci publik tidak rahasia (diketahui semua orang), kunci privat rahasia (hanya diketahui pemilik kunci saja)



Pembangkitan pasangan kunci

1. Pilih dua bilangan prima, a dan b (rahasia)
2. Hitung $n = a b$. Besaran n tidak perlu dirahasiakan.
3. Hitung $m = (a - 1)(b - 1)$.
4. Pilih sebuah bilangan bulat untuk kunci publik, sebut namanya e , yang relatif prima terhadap m .
5. Hitung kunci dekripsi, d , melalui $d \equiv 1 \pmod{m}$.



Enkripsi

1. Nyatakan pesan menjadi blok-blok plainteks: p_1, p_2, p_3, \dots (harus dipenuhi persyaratan bahwa nilai p_i harus terletak dalam himpunan nilai $0, 1, 2, \dots, n - 1$ untuk menjamin hasil perhitungan tidak berada di luar himpunan)
2. Hitung blok cipherteks c_i untuk blok plainteks p_i dengan persamaan

$$c_i = p_i^e \bmod n$$

yang dalam hal ini, e adalah kunci publik.



Dekripsi

Proses dekripsi dilakukan dengan menggunakan persamaan

$$p_i = c_i^d \bmod n,$$

yang dalam hal ini, d adalah kunci privat.

✧ **Contoh 21.** Misalkan $a = 47$ dan $b = 71$ (keduanya prima), maka dapat dihitung

$$n = a \times b = 3337$$

$$m = (a - 1) \times (b - 1) = 3220.$$

✧ Pilih kunci publik $e = 79$ (yang relatif prima dengan 3220 karena pembagi bersama terbesarnya adalah 1).

✧ Nilai e dan n dapat dipublikasikan ke umum.

✧ Selanjutnya akan dihitung kunci dekripsi d dengan kekongruenan:

$$e \times d \equiv 1 \pmod{m}$$

$$d = \frac{1 + (k \times 3220)}{79}$$

Dengan mencoba nilai-nilai $k = 1, 2, 3, \dots$, diperoleh nilai d yang bulat adalah 1019. Ini adalah kunci dekripsi.

✧ Misalkan plainteks $P = \text{HARI INI}$

atau dalam desimal ASCII: 7265827332737873

Pecah P menjadi blok yang lebih kecil (misal 3 digit):

$$p_1 = 726$$

$$p_4 = 273$$

$$p_2 = 582$$

$$p_5 = 787$$

$$p_3 = 733$$

$$p_6 = 003$$

✧ *Enkripsi setiap blok:*

$$c_1 = 726^{79} \bmod 3337 = 215$$

$$c_2 = 582^{79} \bmod 3337 = 776$$

dst untuk sisa blok lainnya

Keluaran: chiperteks $C = 215\ 776\ 1743\ 933\ 1731\ 158$.

✧ *Dekripsi (menggunakan kunci privat $d = 1019$)*

$$p_1 = 215^{1019} \bmod 3337 = 726$$

$$p_2 = 776^{1019} \bmod 3337 = 582$$

dst untuk sisi blok lainnya

Keluaran: plainteks $P = 7265827332737873$

yang dalam ASCII karakternya adalah HARI INI.

-
1. $729^2 \bmod 3337 = (729 \bmod 3337) \cdot (729 \bmod 3337) \bmod 3337$
 2. $729^4 \bmod 3337 = \{729^2 \bmod 3337 \cdot 729^2 \bmod 3337\} \bmod 3337$
 3. $729^8 \bmod 3337 = \{729^4 \bmod 3337 \cdot 729^4 \bmod 3337\} \bmod 3337$
 4. $729^{16} \bmod 3337 = \{729^8 \bmod 3337 \cdot 729^8 \bmod 3337\} \bmod 3337$
 5. $729^{32} \bmod 3337 = \{729^{16} \bmod 3337 \cdot 729^{16} \bmod 3337\} \bmod 3337$
 6. $729^{64} \bmod 3337 = \{729^{32} \bmod 3337 \cdot 729^{32} \bmod 3337\} \bmod 3337$
 7. $729^{72} \bmod 3337 = \{729^{64} \bmod 3337 \cdot 729^8 \bmod 3337\} \bmod 3337$
 8. $729^{76} \bmod 3337 = \{729^{72} \bmod 3337 \cdot 729^4 \bmod 3337\} \bmod 3337$
 9. $729^{78} \bmod 3337 = \{729^{76} \bmod 3337 \cdot 729^2 \bmod 3337\} \bmod 3337$
 10. $729^{79} \bmod 3337 = \{729^{78} \bmod 3337 \cdot 729^1 \bmod 3337\} \bmod 3337$

Kekuatan dan Keamanan RSA

- ✧ Kekuatan algoritma RSA terletak pada tingkat kesulitan dalam memfaktorkan bilangan non prima menjadi faktor primanya, yang dalam hal ini $n = a \times b$.
- ✧ Sekali n berhasil difaktorkan menjadi a dan b , maka $m = (a - 1) \times (b - 1)$ dapat dihitung. Selanjutnya, karena kunci enkripsi e diumumkan (tidak rahasia), maka kunci dekripsi d dapat dihitung dari persamaan $e \times d \equiv 1 \pmod{m}$. Ini berarti proses dekripsi dapat dilakukan oleh orang yang tidak berhak.

✧ Penemu algoritma *RSA* menyarankan nilai a dan b panjangnya lebih dari 100 digit. Dengan demikian hasil kali $n = a \times b$ akan berukuran lebih dari 200 digit.

✧ Menurut Rivest dan kawan-kawan, usaha untuk mencari faktor bilangan 200 digit membutuhkan waktu komputasi selama 4 milyar tahun! (dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik).