Kryptologia przykład metody RSA

przygotowanie:

- niech p=11, q=23

  n= p*q = 253

- funkcja Eulera phi(n)=(p-1)*(q-1)=220

- teraz potrzebne jest e które nie jest podzielnikiem phi; na przykład liczba pierwsza e=29

- potrzebne jest d: (d*e) mod[(p-1)(q-1)] = 1

- d=129 (rozszerzony algorytm Euklidesa)

  (129 * 29) mod 220 = 1

- przyjacielowi przesyłamy dwie liczby (e,n)

Kryptologia przykład metody RSA

kodowania dokonuje przyjaciel:

- np. chce przesłać literę h; jego wartość ASCII to 101
- C = pow(101,29) mod (253) = 50
- przysyła nam wartość C

Kryptologia przykład metody RSA

dekodowanie:

-otrzymaliśmy C=50

- wyliczamy M = pow(C,d) mod 253 = 101

-101 to litera h; przekaz się udał !

Kryptologia przykład metody RSA

Oto poszczególne kroki metody

1) wybierz duże liczby pierwsze p oraz q
2) zachowaj je wyłacznie dla siebie
3) wylicz n = p * q
4) wylicz phi = (p-1) * (q-1)
5) wybierz e, takie że phi oraz e nie mają wspólnych podzielników oprócz 1; np. e=13

Kryptologia przykład metody RSA

6) wylicz d z równania (e*d)mod phi = 1 ; takie d na pewno istnieje

7) przekaż liczby (e,n) przyjacielowi, który będzie szyfrował wiadomości przesyłąne do ciebie

8) kodowanie: C = pow(M,e)

9) dekodowanie M = pow(C,d)

Kryptologia przykład metody RSA (odrobina matematyki)

pow(a, phi(n) + 1)(modulo n) = a to własność funkcji Eulera

z tego wynika, że

e * d = phi(n) +1 teraz dzielimy modulo phi(n)

(e * d) modulo phi(n) = 1

to powyższe równanie daje się rozwiązać (algorytm Euklidesa)

Jeśli n=pq, gdzie p, q to liczby pierwsze,

to phi(n)=(p-1)(q-1)

Uzupełnienie dotyczące kryptologii
”Diffie-Hellman method of key exchange”

Osoba A chce do osoby B przesłać wiadomość, używając np. standardowego algorytmu kryptograficznego takiego jak DES (Data Encryption Standard). Potrzebny jest jakiś klucz szyfrowy (załóżmy, że na steganografię nie ma szans, przeciwnik pilnie nadsłuchuje wszystkich transmisji).

(metodę wymyślono w 1976 r., autorzy to Whitfield Diffie, Martin Hellman, Ralph Merkle)

Uzupełnienie dotyczące kryptologii

1.Osoba A wymyśla jakąś liczbę całkowitą dodatnią g oraz liczbę pierwszą p taką, że $g < p$

2.A wybiera „tajemny całkowity wykładnik m”, taki że $0<m<p$

3.A wylicza a = pow(g,m) modulo p, oczywiście może stosować metodę binarnego kwadratu i mnożenia

4.A przesyła liczby p, g, a do osoby B. Otwartym tekstem, przeciwnik (nieprzyjaciel C) słyszy.

5.B wybiera „tajemny całkowity wykładnik n”, taki że $0<n<p$

Uzupełnienie dotyczące kryptologii

6.B wylicza b = pow(g,n) modulo p

7.B przesyła b do osoby A (otwartym tekstem, przeciwnik słyszy)

8.A wylicza klucz szyfrowy k=pow(b,m) modulo p

9.B wylicza klucz szyfrowy k=pow(a,n) modulo p

10. Zarówno A, jak i B mają ten sam klucz szyfrowy k !!
(wynegocjowali go)

typowo dobiera się liczbę pierwszą p rzędu 300 cyfr dziesiętnych; liczby m oraz n rzędu 100 cyfr dziesiętnych, jako liczbę g wybiera się najczęściej liczbę 5 .

Uzupełnienie dotyczące kryptologii

Jak przeciwnik mógłby poznać klucz szyfrowy k ?

Mógłby np. rozwiązać zadanie a = pow(g,m) modulo p, w celu odnalezienia m, a następnie policzyć – tak jak osoba A – k=pow(b,m) modulo p.

Otóż rozwiązanie a = pow(g,m) modulo p, tak by znaleźć m jest szalenie żmudne (problem nosi nazwę dyskretnego problemu logarytmicznego)

Uzupełnienie dotyczące kryptologii

Jeszcze jedna cecha podnosząca bezpieczeństwo: liczby **m** oraz **n** można teraz nieodwracalnie zniszczyć, nie są już potrzebne !

Przy okazji widać, jak przydatna w języku programowania jest możliwość wygodnego dokonywania dzielenia modulo, czyli posiadanie operatora % by można wyliczać

x % y

resztę z dzielenia modulo.

Uzupełnienie dotyczące kryptologii

Pojęcie klucza szyfrowego – zastosowanie na przykładzie tablicy Vigenera

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

Figure 4 - Vigenère Table

Uzupełnienie dotyczące kryptologii

Czy algorytm RSA jest bezpieczny ?

Konkurencja: Algorytm plecakowy (Ralph Merkle, Martin Hellman, ”Knapsack method”, 1976)

http://www.time.com/time/magazine/article/0,9171,953625,00.html

.....ten zakład został przegrany...algorytm plecakowy przegrał, RSA jeszcze nie przegrał...

Algorytm plecakowy (*knapsack method*)

Następujący pomysł: kluczem jest uszeregowany zbiór liczb. Klucz prywatny to liczby uporządkowane rosnąco. Komunikujący się posiadają pewną *tajną transformację modulo*, którą przekształca się klucz prywatny w klucz publiczny. Klucz publiczny może poznać każdy. Klucz prywatny musi być utajniony.

Teraz mamy liczbę **a** (reprezentującą tekst pierwotny, niekodowany) który chcemy przesłać. Liczbę tą przedstawiamy w układzie dwójkowym. Z tego przedstawienia oraz klucza publicznego tworzy się tekst kodowany. Ten tekst kodowany zostaje przesłany (wtedy w szczególności może go poznać nieprzyjaciel; nic nie szkodzi).

Tekst zakodowany po dotarciu na miejsce zostaje poddany operacji *odwrotnej do tajnej operacji modulo.* Dzięki temu otrzymuje pewną liczbę w; teraz należy znaleźć – to łatwe – które liczby z klucza sumują się do liczby w. Tak otrzymuje się liczbę-tekst pierwotny **a**.

Algorytm plecakowy (*knapsack method*)

Przykład:

1.klucz prywatny (1,3,5,10,20)

2.tajna operacja modulo **p**=7, **b**=40 (mnożenie przez **p**, następnie dzielenie modulo przez **b**)

3.znalezienie operacji odwrotnej do tej z punktu 2; jest to mnożenie przez 23 a następnie dzielenie modulo przez 40 (można sprawdzić, (7*23) mod 40 = 1. Operacja taka istnieje, gdyż 7 oraz 40 są względnie pierwsze

4. przy pomocy tajnej operacji z punktu 2 przekształcamy klucz z punktu 1

Algorytm plecakowy (*knapsack method*)

5.otrzymujemy klucz publiczny, w tym przykładzie to (7,21,35,30,20)

6.klucz publiczny przesyłamy przyjacielowi, klucz publiczny może poznać nieprzyjaciel

7.przyjaciel ma coś nam przesłać, np. chce przesłać literę m

8.litera m to w ASCII wartość 13 dziesiętnie czyli 01101 binarnie

9.przyjaciel korzysta teraz z klucza publicznego

10.wylicza (0*7+1*21+1*35+0*30+1*20) = 76

Algorytm plecakowy (*knapsack method*)

11. tekst kodowany czyli 76 zostaje przesłany do nas; zapewne nieprzyjaciel również poznaje ten tekst

12. tekst kodowany poddajemy transformacji odwrotnej:
    (23*76) mod 40 = 28

13. odnajdujemy, które liczby z klucza prywatnego należy dodać, aby otrzymać 28; trzeba dodać 20+5+3

14. czyli wiadomość od przyjaciela jest binarnie 01101

15. to znak ASCII m; transfer wiadomości powiódł się.

Dlaczego są potrzebne metody szyfrowania z parą kluczy: prywatny oraz publiczny ?

Rozpatrzmy układ N użytkowników. Niech będzie tak, że chcą się porozumiewać parami, by nikt inny ich rozmowy nie słyszał. Potrzeba wtedy 0.5 *N*(N-1) kluczy szyfrowych typu DES czy innych kluczy symetrycznych.

Tymczasem w przypadku szyfrowania z parą kluczy (publiczny, prywatny) każdy z użytkowników przygotowuje sobie te dwa klucze dla siebie, po czym klucz publiczny może dać WSZYSTKIM INNYM. I tak tylko on potrafi zaszyfrowaną wiadomość odszyfrować.

Czyli potrzeba tylko N par kluczy (prywatny+publiczny).

Kryptologia – zagrożenia dla metody RSA

- funkcja dzeta Riemanna
- czy istnieją dostatecznie wielkie liczby pierwsze ?