

# Kryptografia

## Podpisy cyfrowe

**dr Robert Borowiec**

Politechnika Wrocławska

Instytut Telekomunikacji i Akustyki

pokój 908, C-5

tel. 3203083

e-mail: [robert.borowiec@ita.pwr.wroc.pl](mailto:robert.borowiec@ita.pwr.wroc.pl)

www: [lstwww.ita.pwr.wroc.pl/~RB/](http://lstwww.ita.pwr.wroc.pl/~RB/)

**Wykład IX**

**1-godzina**

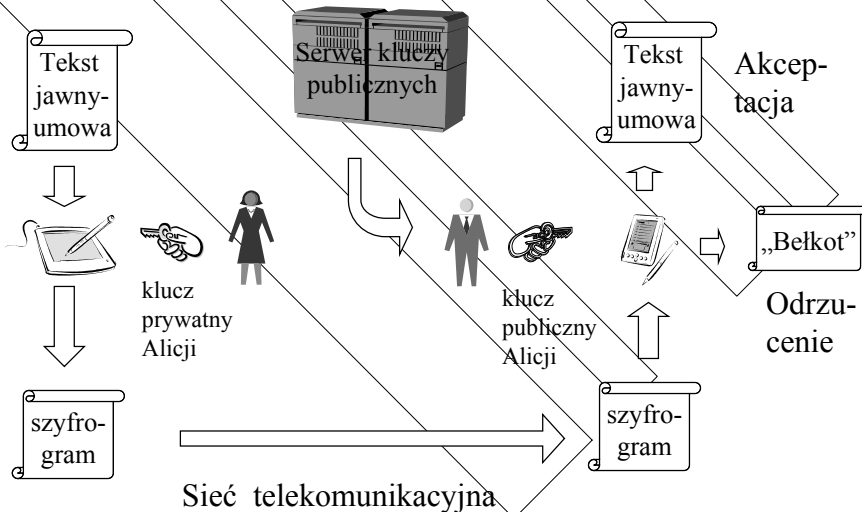
## Cechy podpisu tradycyjnego

- Podpis jest niepodrabialny (przynajmniej teoretycznie), czyli jest świadectwem, że podpisujący świadomie go złożył;
- Podpis jest autentyczny, czyli podpisujący rozważnie go złożył;
- Złożony podpis nie nadaje się do ponownego użycia, czyli jest nieprzenoszalny pomiędzy dokumentami;
- Treść podpisanego dokumentu nie może być zmodyfikowany;
- Autor nie może się wyprzec swojego podpisu, czyli zaprzeczyć, że podpisał dokument.

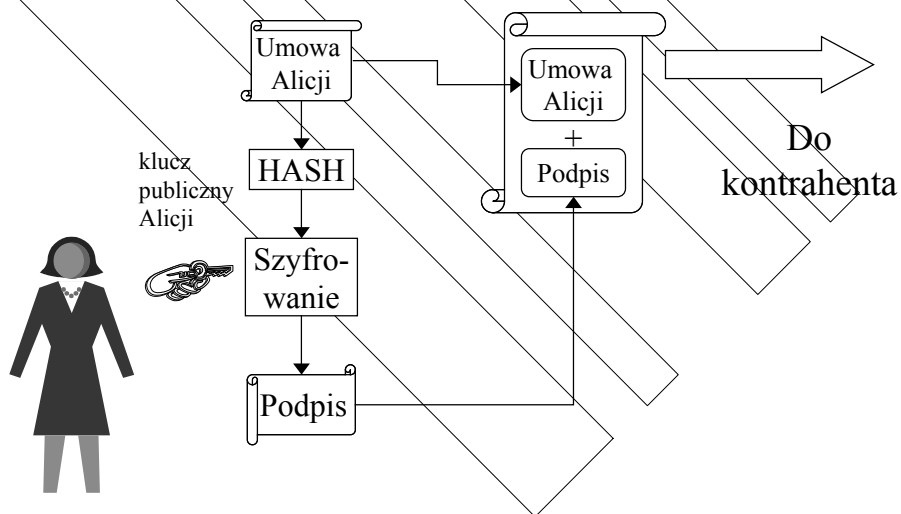
# Podpisy elektroniczne

- Podpisy cyfrowe w symetrycznych systemach kryptograficznych (niepraktyczne, gdyż wymagają strony trzeciej-zaufanego arbitra).
- Podpisy cyfrowe oparte na algorytmach niesymetrycznych
  - ⇒ RSA
  - ⇒ ElGamala
  - ⇒ DSA
  - ⇒ ESIGN
  - ⇒ Okamoto

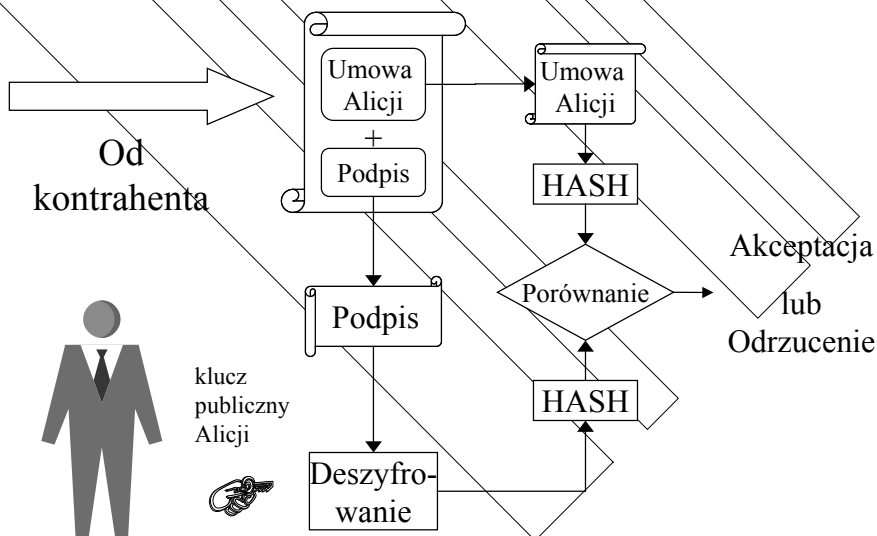
## Procedura podpisu elektronicznego I



## Procedura podpisu elektronicznego II



## Weryfikacja podpisu elektronicznego II



## Własności podpisu cyfrowego

- Podpis jest niepodrabialny, bo tylko osoba podpisująca zna swój klucz prywatny i może go wygenerować
- Podpis jest autentyczny, gdyż wiadomość można odszyfrować tylko kluczem publicznym, będącym parą do klucza prywatnego, będącego tylko w gestii nadawcy wiadomości.
- Podpis nie może być przeniesiony do innego dokumentu;
- Podpisany dokument nie może być zmieniony, bo nie będzie go można rozszyfrować, bądź rozszyfrować skrótu;
- Nie można się wyprzec złożonego podpisu, bo tylko właściciel klucza prywatnego mógł go użyć.

## Podpisy wielokrotne

- Oba przedstawione systemy podpisu elektronicznego mogą być użyte do podpisywania dokumentu przez więcej osób.
- Lepszym rozwiązaniem przy podpisie wielokrotnym jest system podpisu elektronicznego II.

# Algorytm ElGamala

- Algorytm ElGamala może być wykorzystywany do podpisów cyfrowych oraz do szyfrowania.
- Bezpieczeństwo algorytmu opiera się na problemie obliczenia logarytmu dyskretnego
- Algorytm może być używany do przesyłania informacji podprogowych

# Algorytm ElGamala

*(tworzenie kluczy)*

- Wybierana jest liczba pierwsza  $p$  oraz dwie liczby losowe  $g$  i  $x$
- Oblicza się  $y = g^x \bmod p$
- Klucz jawny stanowią liczby  $y$ ,  $g$  i  $p$
- Klucz tajny to liczba  $x$

# Algorytm ElGamala

## (podpisywanie informacji)

- Wybieramy liczbę  $k$ , taką że  $\text{NWD}(k, p-1)=1$
- Obliczamy podpis  $a$ ,  $a=g^k \bmod p$
- Obliczamy podpis  $b$  z rozszerzonego algorytmu Euklidesa z równania

$$M = ax + kb \bmod (p-1)$$

- Liczby  $a$ ,  $b$  stanowią podpis cyfrowy, a  $k$  jest utrzymywane w tajemnicy

# Algorytm ElGamala

## (sprawdzenie podpisu)

- Sprawdzenie podpisu następuje przez sprawdzenie równania

$$y^a a^b \bmod p = g^M \bmod p$$

- Jeżeli równanie jest spełnione to oznacza, że podpis jest prawdziwy

# Algorytm ElGamala

*(przesyłanie informacji podprogowych)*

- Zamiast dowolnej liczby  $k$  mamy wiadomość  $M'$ , którą chcemy ukryć, przy czym  $\text{NWD}(M', p-1)=1$
- Obliczamy podpis  $a$ ,  $a=g^{M'} \bmod p$
- Obliczamy podpis  $b$  z rozszerzonego algorytmu Euklidesa z równania, w którym  $M$  jest informacją nieistotną

$$M = ax + M'b \bmod (p-1)$$

- Liczby  $a$ ,  $b$  stanowią podpis cyfrowy.
- Odbiorca musi znać tajny klucz  $x$  aby odczytać informacje podprogową.

# Algorytm ElGamala

*(Odczyt informacji podprogowych)*

- Cenzor może tylko sprawdzić poprawność podpisu pod listem

$$y^a a^b \bmod p = g^M \bmod p$$

- Odbiorca, ponieważ zna klucz prywatny nadawcy  $X$  może odczytać wiadomość ze wzoru

$$M' = b^{-1}(M - x \cdot a) \bmod (p-1)$$

- a także sprawdzić, czy nie nastąpiła ingerencja cenzora w treść listu

$$g^{x \cdot a} a^b \bmod p = g^M \bmod p$$

# Algorytm DSA

## (Digital Signature Algorithm)

- Algorytm podpisów DSA jest standardem od 1991 roku i jest oparty na algorytmie ElGamala
- Bezpieczeństwo algorytmu DSA opiera się na dwóch problemach tj.: znalezieniu logarytmu dyskretnego w ciele skończonym modulo  $p$ , a drugi to logarytmowanie w podgrupie cyklicznej rzędu  $q$ .
- Algorytm DSA nadaje się tylko do podpisów cyfrowych. Oparty jest na algorytmie niesymetrycznym. Nie nadaje się do szyfrowania informacji.
- Tak jak w algorytmie ElGamala można w nim przesyłać informację podprogową.
- DSA wykorzystuje w trakcie generowania podpisu jednokierunkową funkcję skrótu  $H(x) \Rightarrow \text{SHA-1}$

# Algorytm DSA

## (tworzenie kluczy)

- Wybierana jest liczba pierwsza  $q$  taka, że  $2^{159} < q < 2^{160}$ .
- Wybieranie liczby pierwszej  $p$  z zakresu  $2^{511+64t} < p < 2^{512+64t}$  *takiej*, że  $q$  dzieli  $p-1$ . Liczba  $t$  przyjmuje wartości z zakresu  $\{0, 1, \dots, 8\}$ .  $p$  ma długość od 512 do 1024;
- Obliczamy  $g = h^{(p-1)/q} \bmod p$ ,  $h$  jest dowolną liczbą mniejszą od  $p-1$ , ale taką że  $h^{(p-1)/q} \bmod p > 1$ ;
- Wybieramy losowo liczbę  $x$  o dł. 160 bitów, taką że  $x < q$ ;
- Obliczamy  $y = g^x \bmod p$
- Liczby  $p$ ,  $q$  i  $g$  są jawne i mogą być używane wspólnie przez grupę użytkowników (wchodzą w skład klucza jawnego). Liczba  $x$  jest kluczem prywatnym, liczba  $y$  jest kluczem jawnym.



# Algorytm DSA

## (Podpisywanie i weryfikacja)

### Tworzenie podpisu

- ⇒ Wybierana jest losowa i tajna liczba całkowita  $k$ , mniejsza od  $q$ .
- ⇒ Obliczana jest pierwsza część podpisu  $r = (g^k \bmod p) \bmod q$ .
- ⇒ Obliczana jest druga część podpisu  $s = k^{-1}(H(m) + x \cdot r) \bmod q$
- ⇒ Podpis wiadomości  $m$  stanowi para  $(r, s)$

### Weryfikacja podpisu

- ⇒ Obliczana jest liczba  $w = s^{-1} \bmod q$ ;
- ⇒ Obliczana jest wartość  $u1 = H(m) \cdot w \bmod q$  oraz  $u2 = r \cdot w \bmod q$ ;
- ⇒ Obliczamy  $v = (g^{u1} \cdot g^{u2} \bmod p) \bmod q$ ;
- ⇒ Jeśli  $v = r$  to podpis jest prawdziwy.

## Łączenie podpisów z szyfrowaniem

Protokół przesłania podpisanej i zaszyfrowanej informacji od Alicji do Roberta:

1. Alicja podpisuje pewną wiadomość  $M$  swoim kluczem prywatnym (szyfruje ją kluczem prywatnym) tj.  $S_A(M)$ .
2. Alicja szyfruje tę wiadomość za pomocą klucza publicznego Roberta, czyli oblicza  $E_B(S_A(M))$  i wysyła całość do niego.
3. Robert Deszyfruje otrzymaną wiadomość swoim kluczem prywatnym, czyli oblicza:  $D_B(E_B(S_A(M))) = S_A(M)$
4. Robert sprawdza i odtwarza wiadomość za pomocą klucza publicznego Alicji.  $V_A(S_A(M)) = M$ .
5. Algorytm szyfrujący powinien być inny niż stosowany przy podpisie cyfrowym

## Atak na korespondencję podpisaną i szyfrowaną tym samym algorytmem

Atak na pocztę jest możliwy jeżeli odsyłamy potwierdzenia w postaci zaszyfrowanej poczty do nadawcy.

Trzy osoby Alicja, Robert, Zenek-podsluchujący

1. Alicja wysła list do Roberta-poprzedni algorytm.
2. Zenek przechwytuje list do Roberta, tj.  $E_B(S_A(M))$ , podpisuje go swoim kluczem prywatnym i szyfruje kluczem publicznym Roberta i wysła do niego list  $E_B(S_Z(E_B(S_A(M))))$ .
3. Robert sądzi, że jest to legalna poczta od Zenka. Deszyfruje ją i sprawdza podpis Zenka,  $D_R(V_Z(E_B(S_Z(E_B(S_A(M)))))) = E_B(S_A(M))$ .
4. Treść listu jest oczywiście bezsensowna i nieczytelna. Jednak Robert pilnuje się protokołu i wysła potwierdzenie do Zenka  $E_Z(S_R(E_B(S_A(M))))$ .
5. Zenek teraz odszyfruje wiadomość swoim kluczem prywatnym i sprawdzi podpis Alicji jej kluczem publicznym w wyniku czego odzyska  $M$ , tj.:  $V_A(D_Z(E_Z(S_R(E_B(S_A(M)))))) = M$ .

## Atak na korespondencję podpisaną i szyfrowaną tym samym algorytmem

A series of parallel diagonal lines crossing the slide from the top-left to the bottom-right.

KONIEC

Dziękuję za uwagę