

Rywalizacja kryptoanalityków i kryptografów.

**Szyfrowanie** – proces, w którym wiadomość (*tekst jawny*) jest przekształcana w inną wiadomość (*kryptogram* – *tekst zaszyfrowany*) za pomocą funkcji matematycznej oraz hasła szyfrowania (*klucza*)

**Deszyfrowanie** – proces, w którym *kryptogram* jest przekształcany z powrotem na oryginalny *tekst jawny* za pomocą pewnej funkcji matematycznej i *klucza*.

### Zastosowanie kryptografii:

- ochrona przed nieautoryzowanym ujawnieniem informacji przechowywanej na komputerze,
- ochrona informacji przesyłanej między komputerami,
- potwierdzanie tożsamości użytkownika,
- potwierdzanie tożsamości programu żądającego obsługi,
- uniemożliwianie nieautoryzowanej modyfikacji danych.

**Szyfrowanie jest tylko jednym z elementów strategii  
utrzymywania bezpieczeństwa**

### Atrybuty systemu szyfrującego

- Algorytm szyfrowania
- Klucze szyfrowania
- Długość klucza
- Tekst jawny
- Kryptogram

**Moc kryptograficzna** – zdolność systemu kryptograficznego do ochrony danych przed atakami. Zależy od:

- tajności klucza
- trudności odgadnięcia klucza
- trudności *odwrócenia* algorytmu szyfrowania bez znajomości klucza
- istnienia sposobów odszyfrowania danych bez znajomości klucza
- możliwości odszyfrowania kryptogramu na podstawie znajomości części tekstu jawnego

### Algorytmy kryptograficzne

**Algorytmy z kluczem prywatnym** (z *kluczem symetrycznym*) – ten sam klucz jest używany do szyfrowania i deszyfrowania

**Algorytmy z kluczem publicznym** (z *kluczem asymetrycznym*) – do szyfrowania używa się *klucza publicznego*, a do deszyfrowania *klucza prywatnego*.

**Systemy hybrydowe** – wolniejszy system z kluczem publicznym służy do wymiany *kluczy sesyjnych*, które są później używane w czasie sesji do wymiany w oparciu o klucz prywatny.

### Algorytmy z kluczem prywatnym

**ROT13** - nie wymaga klucza i nie zapewnia bezpieczeństwa. Jest to zwykły szyfr podstawieniowy. Każda litera tekstu jest zastępowana literą położoną o 13 pozycji dalej w alfabecie (cyklicznie). Wykorzystywany do ukrywania ryzykownych dowcipów w grupach dyskusyjnych.

Przykład odszyfrowania pliku:

tr „[a-z][A-Z]” „[n-z][a-m][N-Z][A-M]” < plik

**crypt** - program szyfrujący UNIXa wzorowany na systemie *Enigmy*. Można posługiwać się hasłami o zmiennej długości. Nie jest to system bezpieczny, gdyż niektóre programy mogą automatycznie odszyfrować pliki nie znając hasła ani tekstu jawnego. Inny (bezpieczny) program o tej samej nazwie jest używany do szyfrowania haseł.

**skipjack** – utajniony algorytm opracowany przez *National Security Agency (NSA)*. Algorytm ten zastosowano w szyfrującym układzie scalonym *Clipper*. Używa kluczy 80-bitowych.

**IDEA** – *International Data Encryption Algorithm* opublikowany w roku 1990. Uznawany jest za system stosunkowo mocny. Używa kluczy 128-bitowych. Wykorzystywany jest w popularnym programie PGP do szyfrowania plików i poczty. Obłożony jest serią patentów, więc korzystanie z niego może być utrudnione.

**RC2** – szyfr blokowy, strzeżony jako tajny przez *RSA Data Security*. Ujawniony został w anonimowym liście do grup dyskusyjnych w 1996 roku. Uważany za dość mocny. Sprzedawany jest z implementacją pozwalającą na stosowanie kluczy o długości od 1 do 2048 bitów. W wersjach sprzedawanych na eksport długość klucza jest często ograniczona do 40 bitów.

**RC4** – szyfr strumieniowy, strzeżony jako tajny przez *RSA Data Security*. Ujawniony został w anonimowym liście do grup dyskusyjnych w 1994 roku. Uważany za dość mocny. Sprzedawany jest z implementacją pozwalającą na stosowanie kluczy o długości od 1 do 2048 bitów. W wersjach sprzedawanych na eksport długość klucza jest często ograniczona do 40 bitów.

**RC5** – szyfr blokowy opublikowany w 1994 roku. Pozwala użytkownikowi określać długość stosowanych kluczy, wielkość bloku danych i liczbę cykli szyfrowania.

**DES** – *Data Encryption Standard* został skonstruowany w latach siedemdziesiątych przez *National Bureau of Standards and Technology* (obecnie *National Institute of Standards and Technology*) i firmę IBM. Opatentowany. Używa kluczy 56-bitowych. Implementacje są certyfikowane przez NIST i najczęściej wymagają realizacji sprzętowej. Od momentu powstania był on kilkakrotnie poprawiany i wzmacniany. Został również przyjęty jako norma ANSI (X3.92-1981/R1987). W 1977 roku rząd federalny wydał specjalną publikację FIPS PUB (*FIPS Publication*) nr 47, w której DES został opisany.

### Algorytmy z kluczem publicznym

**RSA** – system opracowany przez późniejszych profesorów MIT (Rivest, Shamir, Adleman). Klucz może mieć dowolną długość. Jest chroniony patentem USA nr 4405892 (19 grudnia 1977). Patent przyznano 20.09.1983 i wygasa 20 września 2000 r. Ponieważ opis algorytmu opublikowano przed zgłoszeniem wniosku patentowego, metody tej można używać bezpłatnie na całym świecie poza USA. Z tego powodu jest znacznie bardziej popularny poza USA niż w Stanach.

**ElGamal** – system oparty na matematyce modułowej i wykładniczej. Może być używany do szyfrowania i generowania podpisów cyfrowych (jak RSA).

**DSA** – *Digital Signature Algorithm* opracowany NSA i przyjęty przez NIST jako norma FIPS (*Federal Information Processing Standard*). Klucze mogą mieć dowolną długość, ale tylko w przedziale 512-1024 zapewniają zgodność z normą FIPS. Może być wykorzystywany jako system do generowania podpisów cyfrowych jak również do szyfrowania.

**Diffie-Hellmana** – system opracowywania i wymiany wspólnego kryptograficznego klucza prywatnego przez publiczny kanał komunikacyjny (nie jest właściwie metodą szyfrowania).

Obie strony umawiają się na pewne wspólne wartości liczbowe i każda z nich buduje klucz. Klucze są wymieniane i każda ze stron może wtedy wygenerować klucz sesji, którego intruz nawet znając początkowe wartości nie może łatwo otrzymać.

Istnieje kilka wersji tego protokołu różniących się liczbą uczestników biorących udział w wymianie.

## System DES

System **Data Encryption Standard** jest w tej chwili jednym z najbardziej popularnych systemów szyfrujących. Wykonuje serię permutacji bitowych, podstawień i operacji rekombinacyjnych na blokach zawierających po 64 bity danych. W każdym przebiegu dane są permutowane z bitami 56-bitowego klucza. Proces składa się z 16 przebiegów. W każdym przebiegu wykorzystuje się inne tabele permutacji i inne bity klucza.

Dobre realizacje programowe są dostępne w wielu miejscach poprzez anonimowe FTP. Złamanie szyfru jest możliwe przez wypróbowanie wszystkich możliwych kluczy – trzeba tylko mieć odpowiedni sprzęt (inny sprzęt był dostępny w latach 70).

Bezpieczeństwo DES można poprawić stosując operację wielokrotnego szyfrowania (*superszyfrowanie*). W **podwójnym DES** w każdym z dwóch przebiegów używany jest inny klucz. Nie jest on znacznie bezpieczniejszy niż pojedynczy.

Metoda **potrójnego DES** polega na: szyfrowaniu kluczem 1; deszyfrowaniu kluczem 2; szyfrowaniu kluczem 3. Aby kryptogram zdeszyfrować należy wykonać działania odwrotne: deszyfrowanie kluczem 3; szyfrowanie kluczem 2; deszyfrowanie kluczem 1. W wielu zastosowaniach używa się tej samej wartości klucza 1 i 3 bez istotnego narażenia bezpieczeństwa. Szacuje się, że do czasu pojawienia się nowych fundamentalnych usterek, słabości algorytmu lub przemowych odkryć w kryptografii, jest to najbardziej bezpieczny algorytm jakiego człowiek mógłby potrzebować.

## System RSA

Moc RSA wynika z trudności rozłożenia dużej liczby (kilkaset cyfr) na czynniki pierwsze. Nie istnieją żadne metody efektywnego rozkładu.

## Podpisy cyfrowe

**Skrót wiadomości** (*kryptograficzna suma kontrolna - hasz*) jest specjalną liczbą będącą produktem funkcji, którą jest bardzo trudno odwrócić. **Podpis cyfrowy** to (zwykle) skrót wiadomości zaszyfrowany za pomocą osobistego klucza osoby podpisującej się - używany w celu potwierdzenia treści. W tej sytuacji proces szyfrowania nazywamy **podpisywaniem**. Podpis cyfrowy:

- wskazuje, czy dana wiadomość została zmieniona (zapewnienie spójności),
- umożliwia weryfikację osoby podpisującej się (zapewnienie autentyczności),
- zapewnienie niewypierania się podpisującego.

Każdy dokument można wyposażać w skrót wiadomości i dołączyć go do przesyłanego dokumentu. Odbiorca może na nowo wyliczyć skrót i porównać go ze skrótem odebranym. Zgodność skrótów świadczy o autentyczności dokumentu. Taką procedurę stosuje *Computer Emergency Response Team (CERT)* podczas przesyłania łat i poprawek dla programów związanych z bezpieczeństwem.

==== KRYPT 08a ====

==== KRYPT 08b ====

## Typowe algorytmy generowania skrótów:

**MD2, MD4, MD5** - funkcje te zwracają liczby 128-bitowe na podstawie tekstu o dowolnej długości. Tekst jest dzielony na fragmenty o ustalonej wielkości i na każdym z nich przeprowadza się serię operacji matematycznych. MD2 został opublikowany w RFC 1319. Nie wykazuje słabych punktów, ale jest wolny. Jego szybsza modyfikacja nosi nazwę MD4 (RFC 1186 i 1320). Ponieważ opisano możliwości potencjalnych ataków na MD4, więc opracowano MD5 (RFC 1321). Jest trochę wolniejszy niż MD4.

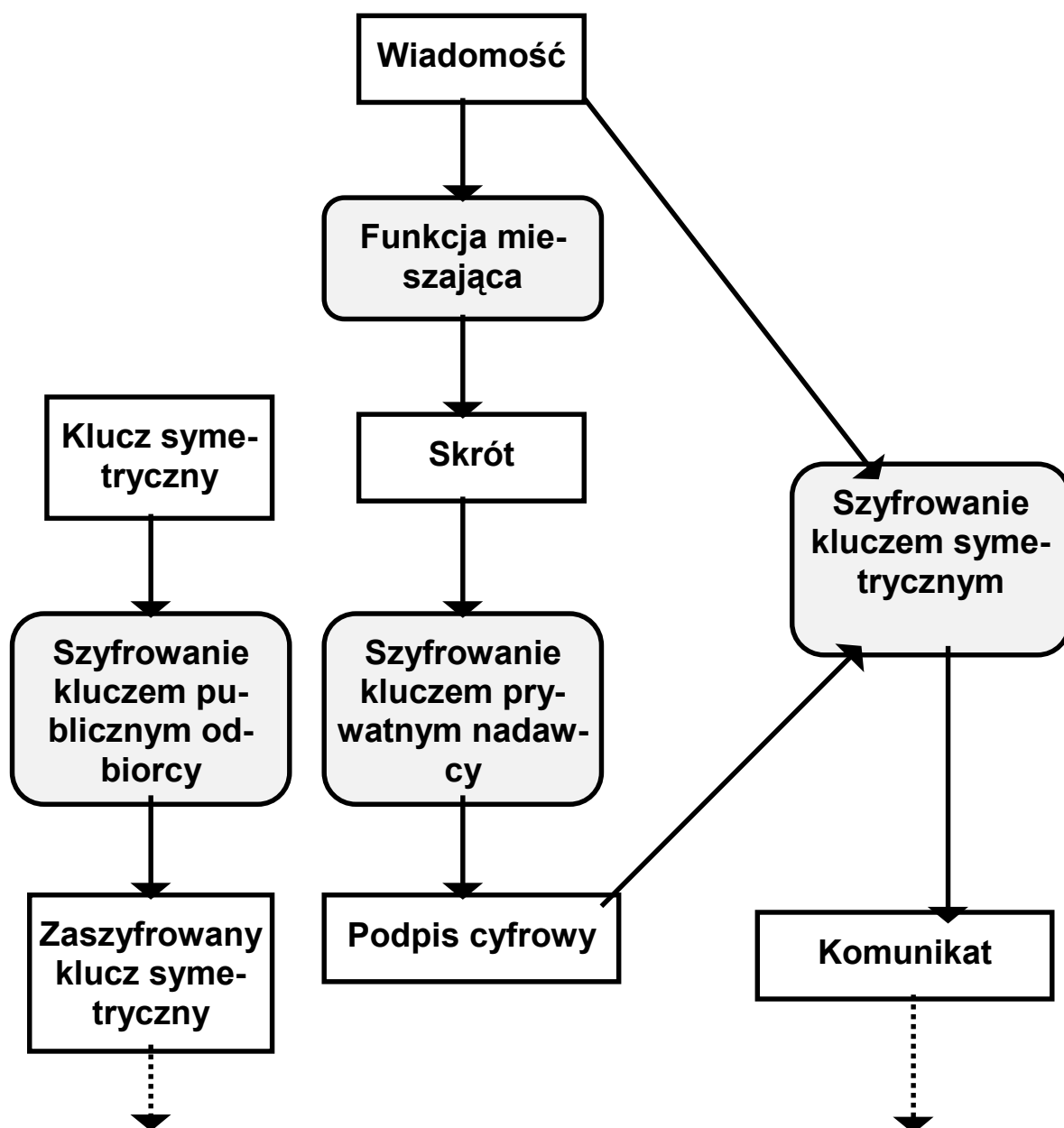
**HAVAL** - modyfikacja algorytmu MD5. Może generować dane wyjściowe w przedziale 92-256 bitów. Zmienna jest również ilość etapów przetwarzania wewnętrznego. Może działać szybciej niż MD5, ale będzie to powodowało obniżenie mocy generowanych danych.

**SNEFRU** - generuje kody o długości 128 lub 512 bitów. Zmienna jest również ilość etapów przetwarzania wewnętrznego. Algorytm ten posiada słabe punkty. Zalecane jest używanie trybu 8-etapowego. Jest jednak wtedy znacznie wolniejszy niż MD5 czy HAVAL.

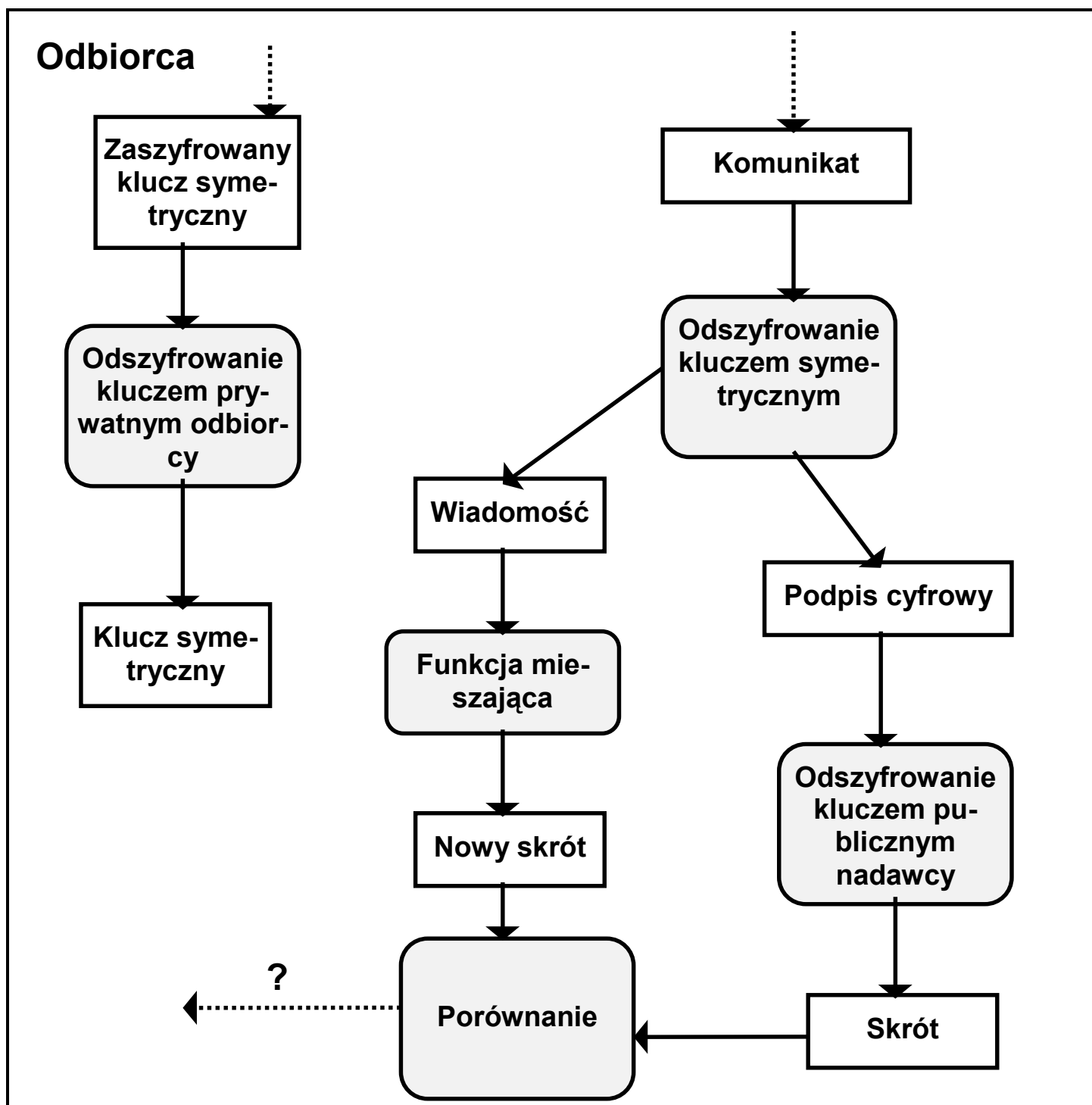
## Literatura:

1. S.Garfinkel, G.Spafford. *Practical Unix and Internet Security*. O'Reilly & Associates 1996 (*tłum.* RM 1997).
2. V.Ahuja. *Network & Internet Security*. Academic Press 1996 (*tłum.* MIKOM 1997).
3. D.Atkins. *Internet Security: Professional Reference*. New Riders Publishing 1997 (*tłum.* LT&P 1997)
4. L.Klander. *Hacker Proof*. Jamsa Press, 1997 (*tłum.* MIKOM 1998).

## Nadawca



KRYP 08a



KRYP 08b