**Strona 1**

Uwierzytelnianie serwerów – RADIUS

RADIUS to skrót od Remote Autthentication Dial-In User Service (usługa zdalnego [uwierzytelniania](https://pl.wikipedia.org/wiki/Uwierzytelnianie) użytkowników, którzy wdzwaniają się do systemu)

Server RADIUSA przechowuje UserId, hasło i centralne parametry autoryzacji (ACL)

W odróżnieniu od TACACS, RADIUS obsługuje serwery uwierzytelniające Proxy, więc informacje uwierzytelniające użytkownika są skalowalne.

RADIUS używa protokołu CHAP (Challenge Handshake Authentication Protocol) do uwierzytelnienia użytkowników.

Client/Server używa wspólnego hasła przechowywanego w pliku konfiguracyjnym do szyfrowania i deszyfrowania CHAP, ale nie pakietów danych

RADIUS używa pojedynczego pakietu UDP dla szybkości i wydajności.

--

**Kryptografia Domen – part 1**

Kryptografia domen dotyczy zasad, sposobów i metod stosowania matematycznych algorytmów i transformacji danych do przekazywania informacji, tak by była zapewniona jej integralność, poufność i uwierzytelnienie.

Należy znać podstawy kryptografii; publiczne i prywatne klucze algorytmów w warunkach ich działania i używania. Konstrukcja algorytmu, dystrybucja klucza i zarządzanie oraz metody ataku; aplikacje, stworzony i używany do podpisów cyfrowych w celu zapewnienia autentyczności elektronicznych transakcji i nie odrzucanie zainteresowanych stron oraz do organizacji i zarządzania infrastrukturą klucza publicznego (PKIs) oraz cyfrowych certyfikatów i zarządzania.

---

Terminy, Definicja, koncept i historia

* Typy szyfrów: klasyczne szyfry i nowoczesne szyfry
* Algorytmy kryptografii: kryptograficzna funkcja hashująca (skrótu), kryptografia klucza symetrycznego, kryptografia klucza asymetrycznego, kryptografia hybrydowa.
* ---
* Utylizacja kryptografii: infrastruktura klucza publiczngo (PKI), http, S\_HTTP, IPsec, SSH, SET, Single Sign-On (SSO), zabezpieczenie maili
* Typy ataków krypto: krypto analityczne i kryptograficzne
* Dyskusja o eksporcie technologii krypto
* ---
* Cele kryptografii: poufność (niezautoryzowana osoba nie ma dostępu do informacji), integralność (pewność, że wiadomość nie została zmodyfikowana)
* Kryptografia zapewnia poufność – szyfrowanie, integralność przyswajalna skrótowa wiadomość), uwierzytelnianie – do zweryfikowania tożsamości obiektu, niezaprzeczalność – nadawca nie może zaprzeczyć wysyłając
* Należy zwrócić uwagę: uwierzytelnianie nie jest częścią triady C.I.A,
* --
* kryptografia to nauka o tajnym piśmie,
* kryptologia to nauka o kryptografii i krypto analizie,
* krypto system to hardware’owa lub software’owa implementacja kryptografii.
* Algorytm to sprecyzowana reguła (lub zbiór reguł) określająca, jak rozwiązać dany problem lub wykonać określone zadanie.
* Szyfr – operacja kryptograficznego przekształcenia bitów lub znaków.

**Strona 2**

* Tekst otwarty / jawny test – dane w postaci niezakodowanej
* Szyfrogram / kryptogram – zaszyfrowane dane
* Szyfrowanie – kodowanie przy użyciu klucza
* Deszyfrowania – odkodowanie przy użyciu klucza
* Kryptoanaliza – praktyka łamania systemów lub uzyskiwanie jawnego tekstu z szyfrogramu bez użycia klucza
* Czynnik pracy – czas, wysiłek oraz niezbędne źródła do złamania krypto systemu
* ---
* Klucz – tajna wartość w formie sekwencji znaków używana do szyfrowania i deszyfrowania
* Zgrupowany klucz – instancja, gdzie 2 klucze generują taki sam szyfrogram z tego samego jawnego tesktu
* Przestrzeń kluczy – wszystkie możliwe wartości do konstruowania kluczy, im większa przestrzeń kluczy, tym lepiej
* Wektor inicjujący (IV) – w krypto jest to blok bitów używany do inicjalizowania wejścia algorytmu szyfrowania blokowej sekwencji jawnego tekstu – zwiększa bezpieczeństwo poprzez wprowadzenie dodatkowej kryptograficznej wariancji do synchronizowania kryptograficznego wyposażenia
* ---

Historia kryptografii:

Dla egzaminu CISSP, czytaj sekrety i kłamstwa, cyfrowe bezpieczeństwo w sieciowym świecie według Bruce Schneier lub Kodołamacz: historia kodów i szyfrów według Stephen Pincock.

* 1500 p.n.e. – tablica z Mezopotamii zawierająca wzór na zrobienie glazury dla ceramiki
* 487 p.n.e. – Grek użył urządzenia nazywanego scytale/ skytale – obsada wokół której długi, cienki pasek ze skóry został opakowany i zapisany
* 50-60 p.n.e. – Juliusz Cezar użył prostego zastąpienia alfabetu (poprzez przesuwanie liter o stałą ilość) w rządowych komunikatach.
* 1790 – Thomas Jefferson wynalazł koło szyfrowe (kolejność dysków jest kluczem
* 1854 – Charles Babbaqe ponownie odkrył koło szyfrowe
* ---
* 1919 – 1922 – wydane patentów przez Gilberta Vernama do szyfru Vernama
* 1930 – 1941 – niemiecka armia użyła maszyn szyfrujących Lorenz SZ 40 i SZ 42 bazujący na strumieniu szyfru Vernama do zaszyfrowania wiadomości dalekopisowej – strumień szyfru używał pseudolosowych bitów XOR’owanych z jawnym tekstem.
* 1933-1945 – jednostki terenowe niemieckiej armii użyły Enigmy – maszyny szyfrującej wiadomości – elektromechaniczny wirnik maszyny szyfrującej używa polialfabetycznego podstawienia
* ---
* 1943 – 44 – brytyjscy łamacze kodów wynalazł Colossus Mark 1 i Colossus Mark 2 do odszyfrowania maszyny szyfrującej Lorenza – wynaleziony przez Maxa Newmana i Tommy’iego Flowersa, używa analizy częstotliwości
* 1938-44 – brytyjscy łamacze kodów wynaleźli Bombe do rozszyfrowania Enigmy- wynalezione przez Alana Turinga, używa analizy częstotliwości
* ---
* 1976 – NSA wybrany jako IBM’owa modyfikacja szyfru Lucyfera do bycia standardem szyfrowania danych (DES)
* 1976 - Whitfield Diffie & Martin Hellman opublikowali New Directions in Cryptography
* 1978: Ronald L. Rivest, Adi Shamir & Leonard M. Adleman (RSA) opublikowali RSA Algorithm for Public Key System.
* 1984 – szyfr ROT13 wprowadzony na systemach UNIX, szyfruje jawny tekst przez zmianę liter o 13 pozycji
* 1991 - Phil Zimmermann wydał pierwszą wersję PGP (Pretty Good Privacy).
* 2000 - Algorytm Joana Daemana i Vincenta Rijman’sa Rijndaela został wybrany przez NIST jako zaawansowany standard szyfrowania danych (AES).
* ---

Algorytmy i operacje w kryptografii

* Algorytm kryptograficzny – ustalenie matematycznej funkcji i zasad, które na wejściu wezmą jawny tekst i klucz, a na wyjściu dadzą szyfrogram.
* Operacja kryptograficzna – szyfrowanie/deszyfrowanie – szyfrowanie to zamiana jawnego tekstu na szyfrogram w celu zachowania poufności danych – deszyfrowanie to zamiana szyfrogramu z powrotem na jawny tekst
* --

Siła szyfrowania

Celem krypto systemu jest narażenie go na zbyt drogie lub czasochłonne wynagrodzenie wysiłku.

Siła metod szyfrowania pochodzi z: algorytmów, tajemnicy klucza, długości klucza, wektorów inicjalizacji IV i metod szyfrowania

Czynnik pracy odnosi się do tego, jak trudno poznać algorytm lub klucz (w zależności który z nich nie jest udostępniony publicznie) używany w krypto systemie, przybliża to ile wysiłku zajęłoby atakującemu rozpoznanie metody szyfrującej.

**Strona 3**

Klasyczne szyfry: szyfr podstawieniowy, szyfr transpozycyjny, szyfr polialfabetyczny, zatajenie.

Nowoczesne szyfry: szyfr blokowy, szyfr strumieniowy, steganografia, kombinacja (może być połączeniem poprzednich szyfrów nowoczesnych)

--

Szyfr podstawieni owy wstawia w miejsce części informacji inną informację – najczęściej odbywa się poprzez podstawienie liter w alfabecie, przykładami są: szyfr Cezara i ROT13

Mamy ABCDEFGHIJKLMNOPQRSTUVWXYZ i przesuwamy wszystko o 3 to otrzymujemy DEFGHIJKLMNOPQRSTUVWXYZABC

--

Szyfr polialfabetyczny jest rodzajem szyfru podstawieniowego, wynaleziony przez Blaise de Vigenère w XIX wieku, algorytm polega na powtarzaniu sekretnego klucza wzdłuż jawnego tekstu.

--

Szyfr transpozycyjny zmienia kolejność znaków w wiadomości, klucz wyznacza pozycję na którą przeskoczy dany znak. Klucz nie jest standardowy, jest mapowaniem kolejności.

Np. (1,2,3,4,5)=(3,4,5,2,1) -> WORLD = RLDOW

--

Szyfr transpozycyjny – permutacje w tym szyfrze odbywają się w blokowych macierzach. Oznacza to, że wiadomość jest rozłożona w macierzy.

--

Zatajenie polega na ukryciu wiadomości w innej dłuższej. Dostajesz tekst i Twoim kluczem jest wzięcie co 6 słowa, jeśli wybierzesz co 6 słowo z otrzymanej widomości otrzymasz prawdziwą informację.

--

Szyfr blokowy operuje na jawnym tekście o stałej długości. Algorytm pobiera kawałek zwykłego tekstu o stałej długości i tworzy z niego szyfrogram o tej samej długości. Zazwyczaj jest to 8 bajtów, czyli 64 bity. Implementacja software’owa zazwyczaj. Generalnie wolniejszy od strumieniowego. Przykłady to DES, Triple DES, AES, IDEA.

**Strona 4**

Szyfry strumieniowe:

* Operują na stałych strumieniach tekstu jawnego
* Zazwyczaj są zaimplementowane w sprzęcie
* Kompleksowe działania z długim kluczem który się nie powtarza
* Statystycznie nieprzewidywalne
* Strumień klucza nie powinien zależeć liniowo od klucza
* Przykłady: RC4, SEAL, VEST
* ---

Steganografia jest metodą ukrywania danych na innym nośniku danych, tak, że ich nie widać.

* Microdot bardzo popularne podczas II wojny światowej
* Pliki komputerowe (obrazki, mp3 lub wideo) mają obszary danych, których nie używają, lub które są nieistotne dla nich i steganografia zastępuje ten obszar danymi, które mają zostać ukryte.
* Pliki mogą być przekazywane dalej i nikt nie wie, że coś w nich jest ukryte.
* Może być używana do wstawiania ukrytych znaków wodnych.

Notatka do egzaminu: Steganografia nie używa algorytmów lub kluczy do szyfrowania, ukrywa dane w innym obiekcie.

--

Funkcja haszująca bierze na wejściu dowolną długość, a na wyjściu zwraca wartość o stałym rozmiarze – „wartość haszowaną”

Maksymalna liczba bitów na wejściu i na wyjściu jest ustalana przez tego, kto wymyśla funkcję haszującą.

Odporność na pre-image. Dobra funkcja skrótu działa w jedną stronę. Oryginalne wejście (wiadomość lub plik) nie może być pochodną wartości haszowanej.

Odporność na kolizje. Funkcja haszująca nie powinna zwracać dwóch takich samych wyników.

--

Kryptograficzne funkcje haszujące są wykorzystywane w celu zapewnienia integralności, uwierzytelnienia i niezaprzeczalności

Skrót wiadomości, jeśli komunikat lub plik są używane jako wejście funkcji haszującej

Uwierzytelnienie wiadomości, jeśli tajny klucz wraz z komunikatem znajdzie się na wejściu funkcji haszującej.

Podpis cyfrowy, gdy prywatny klucz jest używany jako wejście, a wyjście może być zweryfikowane przy pomocy klucza publicznego.

Kryptograficzne funkcje haszujące są także używane jako „przypadkowe kleszcze” w pseudo losowych generatorach liczb (PRNGs)

--

Rodzaje funkcji haszujących:

Nie kluczowy porządek (dla integralności): integralny kod wiadomości (MIC), modyfikacja wykrywania kodu (MDC)

Kluczowy porządek (dla uwierzytelnienia): Kod uwierzytelniający wiadomość (MAC): Tajny klucz + wiadomość, kluczowy skrót MAC lub skrócony MAC (HMAC): MAC + MDC

Podpis cyfrowy (dla niezaprzeczalności)

Popularne algorytmy skracania wiadomości: MD5, RIPE-MD, HAVAL, FIPS 186-2: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512.

Popularne algorytmy podpisów cyfrowych: ElGamal, FIPS 180-2: DSA, EC-DSA.

--

Funkcje haszujące z nie kluczowym porządkiem:

Kluczowy porządek jest istotny dla zapewnienia integralności funkcji. Wyjściowa haszowana wartość o stałej długości jest nazwana: skrótem wiadomości (MD) w świecie kryptografii lub modyfikacją wykrytego kodu (MDC) opatentowane przez IBM.

--

Funkcje haszujące z kluczowym porządkiem:

Kluczowy porządek jest używany do zapewnienia wierzytelności funkcji. Wyjściowa haszowana wartość o stałej długości jest nazwana: Uwierzytelnianie kodu wiadomości (MAC), tajny klucz (symetryczny krótkotrwały klucz) + wiadomość (integralność kodu wiadomości (MIC) i kluczowy HASH MAC (HMAC)

**Strona 5**

Podpis cyfrowy (lub cyfrowy odcisk palca) jest używany do niezaprzeczalności funkcji. Jest on hybrydową kryptografią, która używa nie-kluczowych funkcji haszujących i asymetrycznych par kluczy (publiczny i prywatny)

--

Prawdziwa losowa liczba jest trudna do znalezienia, to co generują maszyny jest pseudo losowe. Najzwyklejszy pseudo losowy generator liczb (PRNG) używa liniowo przystającej formuły: Xn+1 = aXn + b) mod m

Żeby dodać „losowości”, PRNG wyjście jest przepuszczane przez kryptograficzną funkcję haszujacą.

--

Kryptografia symetrycznych kluczy dotyczy pojedynczych tajnych (symetrycznych) kluczy: Nadawca i odbiorca muszą mieć ten sam tajny klucz, który nadawca używa do szyfrowania, a odbiorca do deszyfrowania wiadomości.

--

Obie strony muszą znać tajny klucz, wymagany jest specjalny mechanizm bezpieczeństwa do przekazywania tego klucza.

Każda para użytkowników potrzebuje unikalnego klucza: gdy liczba użytkowników rośnie, zarządzanie kluczami staje się utrudnione, grupa 100 osób potrzebuje aż 4,950 kluczy. Kryptografia symetrycznych kluczy zapewnia jedynie poufność. Należy połączyć ją z MAC do integralności i uwierzytelnienia wiadomości.

Popularne algorytmy kluczy symetrycznych: DES, 3DES, AES,

RC6, Twofish, Blowfish, etc.

--

Są 2 szyfry, które używają algorytmów symetrycznych kluczy do dekodowania:

* Blokowe szyfry - pobierają kawałek zwykłego tekstu o stałej długości i tworzy z niego szyfrogram o tej samej długości.
* Strumieniowe szyfry – generują strumień klucza (sekwencję bitów), łączą go z tekstem jawnym bit po bicie przy użyciu operacji XOR i tworzą strumień szyfrogramu. Szyfry z kluczem jednorazowym (np. szyfr Vernama) jest rodzajem szyfru strumieniowego. Cały strumień klucza jest całkowicie losowy i używany jedynie raz.
* ---

---\

Ekskluzywny OR (XOR) jest operacją, wykonywaną na 2 bitach, jeśli 2 bity są takie same wynikiem tej operacji jest 0, w przeciwnym wypadku 1.

**Strona 6**

Szyfry blokowe używają „konfuzji” i „dyfuzji” w ich metodach deszyfrowania

Konfuzja odnosi się do tworzenia relacji pomiędzy kluczem i szyfrogramem najbardziej skomplikowanej jak to możliwe.

Dyfuzja dotyczy takiej właściwości, że redundancja w statystykach tekstu jawnego jest „rozproszona” w statystykach szyfrogramu.

--

Konfuzję można osiągnąć poprzez substytucję operacji szyfru.

S-Box zawiera elementy konfuzji szyfrów blokowych i krypto systemie symetrycznego klucza, S-box bierze jakąś liczbę bitów na wejściu (m) i przekształca je na jakąś liczbę bitów na wyjściu (n). Implementowana jako „m x n” tabela odnośników do zapewnienia elementów konfuzji szyfru blokowego.

S-box może być zaimplementowany jako:

Tabela odnośników (np. DES)

Liniowa transformacja poprzez mnożenie macierzy (np. AES)

Dynamicznie z klucza (np. Blowfish lub Twofish)

--

Dyfuzja może zostać osiągnięta poprzez transpozycję operacji szyfrowania.

Permutacja Sieci Feistela dostarcza szyfrom blokowym elementy dyfuzji w krypto systemie symetrycznych kluczy. Przykłady Feistela lub zmodyfikowanego Feistela: DES, TDES, Blowfish, Twofish, RC5, etc. Uogólniony Feistel: RC2, RC6, Skipjack, etc.

Kolumnowa transpozycja nie jest permutacją sieci Feistela, ale także zawiera elementy dyfuzji szyfrów blokowych.

--

Ogólnie wszystkie szyfry blokowe działają w 5 trybach:

2 Tryby blokowe: ECB i CBC oraz 3 Tryby strumieniowe: CFB, OFB, CTR

FIPS 81 jest określony w 4 trybach jako federalny standard, tryb licznika nie został wzięty pod uwagę. NIST wycofało FIPS 81 19 maja 2005r.

--

Elektroniczna książka kodowa (ECB):

Blok 64 bitów danych przetwarzanych pojedynczo, po jednym na raz

Deszyfrowanie rozpoczyna się od początku szyfrogramu.

Zalety: szybkość i prostota

Wady: podatne na rozpoznanie tekstu jawnego przez niepożądane osoby

--

Tryb wiązania bloków zaszyfrowanych (CBC):

64-bitowy tekst jawny jest ładowany sekwencyjnie, a następnie XORowany z 64-bitowym wektorem inicjującym, połączenie przetwarzane na szyfrogram w ramach tajnego klucza, pierwszy szyfrogram jest XORowany z następnym tekstem jawnym.

Najczęściej stosowany tryb pracy.

--

Wektor inicjujący (IV) jest blokiem bitów używanych na początku procesu szyfrowania blokowych lub strumieniowych szyfrów.

IV jest ogólnie używany w: generatorach liczb losowych (RNG) lub generatorach liczb pseudolosowych (PRNG)

IV eliminuję aktualizację kluczy

W szyfrach strumieniowych, IV jest ładowany do wewnętrznego kluczowego sekretnego stanu szyfru, po którym określona liczba rund szyfrowania zostanie wykonana przed zwolnieniem pierwszego bitu wyjścia.

W szyfrach blokowych IV jest liniowo dodawany do pierwszego bloku tekstu jawnego przed szyfrowaniem.

--

Szyfrowanie ze sprzężeniem zwrotnym (CBF):

Poprzedni blok szyfrogramu jest szyfrowany, a wyjście jest połączone z blokiem tekstu jawnego za pomocą XOR i w ten sposób powstaje bieżący blok szyfrogramu. IV jest używany jako „nasiono” dla procesu. Wzorce tekstu jawnego są ukryte poprzez operacje XOR.

**Strona 7**

Tryb sprzężenia zwrotnego wyjściowego (OFB) jest podobny do CFB, spodziewana ilość XORowania z każdym blokiem tekstu jawnego jest generowana niezależnie przez tekst jawny i szyfrogram. IV jest używany jako „nasiono” dla procesu.

--

Strumieniowy tryb licznikowy (CTR) – Nie FIPS!

Podobnie, jak OFB spodziewana ilość XORowania z każdym blokiem tekstu jawnego jest generowana niezależnie przez tekst jawny i szyfrogram. Zaszyfrowana wartość CTR generuje strumień klucza do XORowania ze strumieniem wiadomości, podobnie jak szyfr strumieniowy.

--

Standard szyfrowania danych (DES) jest krypto systemem symetrycznego klucza, bazującym na IBMowym szyfrze Lucyfera. Jest nim FIPS 46-1 od 1977. DES jest algorytmem opartym na 64-bitowym bloku (56 bitów tajnego klucza i 8 bitów parzystych), który używa 56-bitowego klucza i 16 rund transpozycji i podstawiania do zaszyfrowania każdej z 8 liter tekstu jawnego (64 bity).

--

Animacja procesu DES:

Rozkład klucza, 56-bitowy klucz dzieli się na dwa 28-bitowe Podklucze

Dla każdej kolejnej rundy obie połówki są obracane w lewo o 1 lub 2 bity

Początkowa permutacja (IP): permutacja 64-bitowego bloku wejściowego na 64 bitową macierz IP

Rundy: Rozciąganie, miksowanie kluczy, podstawienie, permutacja.

**Strona 8**

TDES jest tymczasowym rozwiązaniem od NIST do zapewnienia lepszego rozwiązania niż DES. Używa 48 rund transpozycji i podstawień. TDES jest w przybliżeniu 256 razy silniejszy niż DES.

TDES działa w 4 różnych trybach:

DES-EEE: TDES szyfrowanie z 3 różnymi kluczami

DES-EDE: operacje szyfrowania-deszyfrowania z 3 różnymi kluczami

DES-EEE2: TDES szyfrowanie z 2 różnymi kluczami, pierwsza i trzecia operacja używa tego samego klucza

DES-EDE2: TDES operacje szyfrowania-deszyfrowania z 2 różnymi kluczami, pierwsza i trzecia operacja używa tego samego klucza

--

TDES jest używany jako zasadniczy algorytm szyfru blokowego dla poniższych trybów operacji: Elektroniczna książka kodowa, Tryb wiązania bloków zaszyfrowanych, Szyfrowanie ze sprzężeniem zwrotnym, Tryb sprzężenia zwrotnego wyjściowego. Np. TECB (TDES-EEE w ECB).

--

NIST ogłosił potrzebę wymiany TDES w 1997r.

Wystąpiło zapotrzebowanie na algorytmy wspierające klucze w rozmiarach 128, 192 i 256 bitów.

Wśród kandydatów AES znaleźli się:

MARS – opracowany przez IBM, który rozwinął Lucyfera na podstawie którego powstał DES

RC6 – opracowany przez RSA

Serpent – opracowany przez Rossa Andersona, Eli Bihama i Larsa Knudsena

TwoFish – opracowany przez Counterpane System

Rijndael – opracowany przez Vincenta Rijmena i Joana Deamona

CAST – opracowany przez Entrust Technologies

--

Algorytm Rijndael opracowany przez Vincenta Rijmena i Joana Deamona stał się AES po długotrwałych testach (FIPS 197)

AES jest symetrycznym blokowym szyfrem który może przetwarzać 128-bitowy blok danych, używa klucza o długości 128, 192 lub 256 bitów, zmienna liczba rund, każda runda składa się z 4 kroków (zastępowanie bajtów, przesunięcie wiersza, mieszanie kolumn, dodanie okrągłego klucza), został opracowany do obsługi dodatkowego rozmiaru bloku i długości klucza, jednak nie został przyjęty do standardu AES.

--

Jak DES, AES jest algorytmem blokowego szyfrowania dla krypto systemu symetrycznego klucza, używa zasad konfuzji i dyfuzji, używa podstawieniowo-permutacyjnej sieci (nie sieci Feistela)

Dla obu: szyfru i odwrócenia szyfru, algorytm AES używa „okrągłej funkcji”, która składa się z 4 różnych operacji na bajtach: zastępowanie bajtów (konfuzja), przesunięcie wiersza (dyfuzja), mieszanie kolumn (dyfuzja), dodanie okrągłego klucza (konfuzja).

--

Animacja 128-bitowego procesu szyfrowania AES

Rozbudowa klucza przy użyciu wykazu kluczy Rijndaela

Inicjowanie: Dodanie okrągłego klucza

Rundy (zastępowanie bajtów, przesunięcie wiersza, mieszanie kolumn, dodanie okrągłego klucza), w ostatniej rundzie nie ma mieszania kolumn

--

Międzynarodowy algorytm szyfrowania danych (IDEA)

Używa 64-bitowych bloków danych na wejściu i na wyjściu

Bazuje na 3 matematycznych funkcjach: XOR, dodanie modulo 216 (65536), mnożenie modulo 216 + 1 (65537).

Używa 8 rund transpozycji i podstawień.

--

RC5 - Opracowany przez Rona Rivesta, cechy danych uzależnione od obrotów, zmienne rozmiary bloków (32, 64 lub 128 bitów), zmienna długość klucza (0-2040 bitów), zmienna ilość rund.

RC6 – Także opracowany przez Rona Rivesta, był kandydatem do AES, bazując na RC5, używa bloku danych o długości 128 bitów, wspiera klucze 128, 192 i 256 bitowe.

Blowfish – bardzo wydajny szyfr blokowy, opracowany przez Bruce Schneier, rozmiar klucza 32-448 bitów (w krokach 8 bitów), 64-bitowy blok danych, zoptymalizowany do 32-bitowego mikroprocesora.

**Strona 9**

RC4 – najczęściej realizowane oprogramowanie szyfru strumieniowego, używa SSL i WEP, Także opracowany przez Rona Rivesta, używa algorytmu pseudo-losowego generowania, zmienny rozmiar klucza, bardzo wydajny, szybszy niż inne szyfry blokowe, szyfry strumieniowe mogą być trudne do prawidłowego zaimplementowania.

--

Kryptografia asymetrycznego klucza dotyczy 2 matematycznych zależności ale z różnymi kluczami, znanymi jako para kluczy (prywatny i publiczny)

Klucz publiczny jest pochodną klucza prywatnego, jedynie właściciel ma klucz prywatny, jednokierunkowy matematyczny link, prywatny klucz nie może zostać wydedukowany (teoretycznie) na podstawie analizy klucza publicznego.

--

Kryptografia klucza asymetrycznego jest matematycznie bardziej złożona niż kryptografia klucza symetrycznego.

Algorytmy faktoryzacji

Algorytmy logarytmów dyskretnych – skończony logarytm dyskretny i nieskończona krzywa eliptyczna logarytmu dyskretnego

Kryptografia klucza publicznego działa znacznie wolniej niż kryptografia klucza symetrycznego ( 100 razy wolniej w software i 1000-10000 razy wolniej w hardware), rozmiar klucza musi być relatywnie duży

--

Algorytmy faktoryzacji bazują na factoringu „pół-pierwszych” liczb (rzeczywistych)

RSA PKCS#1 – generowanie klucza – prywatny i publiczny klucz RSA, szyfrowanie klucza publicznego c = me mod n, gdzie c jest szyfrogramem, m jest wiadomością, e jest kluczem publicznym, n jest modułem RSA

Deszyfrowanie klucza publicznego = m = cd mod n, gdzie d jest kluczem prywatnym

Podpis cyfrowy - m = se mod n, gdzie s reprezentuje podpis

--

Skończony logarytm dyskretny

Bazuje na matematycznym dowodzie o uogólnionym problemie logarytmu dyskretnego (GDLP), gdzie obliczanie potęgowania na skończonych wartościach jest proste, ale obliczanie logarytmu dyskretnego jest trudne

Parametry mogą być bardzo duże (512, 1024, 2048 bitów), Diffie-Hellman, El Gamal, i DSA. Brutalne ataki na logarytm dyskretny są niemożliwe, ale jest narażony na wybrane ataki.

--

|  |  |  |  |
| --- | --- | --- | --- |
| Krok | Alice | Wymiana metodologii | Bob |
| 1 | Wybranie z Bobem Y i P | Yx (mod P)  Wiedza publiczna | Wybranie z Alice Y i P |
| 2 | Y=11, P=13 | 11x (mod 13) | Y=11, P=13 |
| 3 | Wybranie sekretnego numeru (2) | Wybranie sekretnego numeru | Wybranie sekretnego numeru (5) |
| 4 | 112 (mod 13)  121 (mod 13) = 4 | Obliczenie jednokierunkowej funkcji przy użyciu sekretnego numeru | 115 (mod 13)  161051 (mod 13) = 7 |
| 5 | Wysłanie „4” do Boba | Wysłanie rezultatu do drugiej osoby | Wysłanie „7” do Alice |
| 6 | 72 (mod 13)  49 (mod 13) = 10 | Wziąć klasyczne równanie i podstawić sekretny numer | 45 (mod 13)  1024 (mod 13) = 10 |
| 7 | Symetrycznym kluczem zostaje „10” | Obie osoby mają ten sam numer bez ujawniania sekretu | Symetrycznym kluczem zostaje „10” |

**Strona 10**

Wiadomość m = 3

Wybór 2 losowych liczb pierwszych p=19, q=13

n=pq, n= 247

Wybór losowego klucza e = 7

Obliczenie klucza prywatnego do deszyfrowania

d = e-1 mod (p-1)(q-1)

d = ((19-1)(13-1))/7 = 216/7 = 31 (w zaokrągleniu)

Klucz publiczny = (n,e) = (247,7)

Szyfrowanie: c = me mod n 🡪 **c = 37 mod 247** 🡪

c = 211 (szyfrogram)

Deszyfrowanie: m = cd mod n 🡪 m = 21131 mod 247 🡪

m = 3 (tekst jawny)

--

Kryptograficzna krzywa eliptyczna (ECC) używa algebraicznych systemów określonych na punktach krzywej eliptycznej do określenia kryptografii klucza publicznego.

ECC bazuje na matematycznym problemie czynników, które są parami współrzędnych położonych na krzywej eliptycznej.

Zalety: Najwyższa wytrzymałość, szybkie szyfrowanie i prędkość podpisu, małe podpisy i certyfikaty (idealne dla kart smart).

Przykłady: ECC, EC-DH, EC-DSA, EC-ElGamal

--

Kryptografia klucza asymetrycznego jest bardziej złożona i wolniejsza niż symetrycznego, ale zarządzanie kluczem jest uproszczone, tylko jedna osoba musi znać klucz prywatny, znajomość klucza publicznego nie zagraża bezpieczeństwu transmisji komunikatów.

Dystrybucja klucza jest skalowalna. Każdy przedmiot ma tylko jedną parę kluczy, liczba kluczy = 2n (1000 użytkowników, 2000 kluczy)

Para klucz prywatny i publiczny są matematycznie zależne, ale różne.

--

|  |  |  |
| --- | --- | --- |
| Atrybut | Symetryczny | Asymetryczny |
| Klucz | 1 dzielony tajny klucz | Para kluczy (prywatny i publiczny) |
| Wymiana klucza | Trudna – musi być wykonana w bezpieczny sposób | Relacja ustala oba klucze, publiczny jest powszechnie znany |
| Prędkość | Mniej złożona i szybsza | Bardziej złożona i wolniejsza |
| Długość klucza (bity) | Mniejsza (80 do 256+) | Większa (10024 do 2048+), ale ECC (160 do 512+) |
| Użycie | Całkowite szyfrowanie | Szyfrowanie kluczem, dystrybucja klucza |
| Bezpieczeństwo | Poufność, integralność | Poufność, integralność, wierzytelność, niezaprzeczalność |