

Cryptology

Abraham van der Merwe
abz@clug.org.za

12 August 2003

Overview

- Terminology
 - Cryptography
 - Crypto-analysis
- Symmetric ciphers
 - Block ciphers
 - Stream ciphers
- Assymmetric ciphers

Block ciphers

- Substitutions
 - Mono-alphabetic
 - Poli-alphabetic
- Transpositions
 - Operating on blocks of data
- Product ciphers
 - Combinations of above

Substitutions

- Mono-alphabetic
 - Caesar / ROT-13
 - Limited key space
 - Free masons “pigpen” system
 - Letter frequencies
- Poli-alphabetic
 - Vigenere
 - Kasiski-test
 - Sequences repeat
 - Discover key length
 - Substrings of mono-alphabetic characters

Transpositions

- Operate on blocks of characters
 - No more letter frequency attacks
 - Column transposition
 - Limited key space
 - Hill System
 - Open to cleartext / ciphertext attacks

Product ciphers

- Data Encryption Standard (DES)
 - Very slow
 - Small key space
 - Been around for too long
- Enhancements
 - Triple DES
 - $E(K_z, D(K_y, E(K_x, x)))$
 - $E(K_z, E(K_y, E(K_x, x)))$
 - ANSI/FIPS modes of operation

Product ciphers (continued)

- Electronic Code Book (ECB)
 - $C_n = E(P_n)$
- Cipher Block Chaining (CBC)
 - $C_0 = E(P_0 \oplus K)$
 - $C_n = E(P_n \oplus C_{n-1})$
- Propagating Cipher Block Chaining (PCBC)
 - $C_0 = E(P_0 \oplus K)$
 - $C_n = E(P_n \oplus P_{n-1} \oplus C_{n-1})$
- K-bit Cipher Feedback (CFB)
- K-bit Output Feedback (OFB)

Product Ciphers (continued)

- Advanced Encryption Standard (AES)
 - Objectives
 - Fast (encrypt streaming video in real-time)
 - Small (fit on 8-bit smart cards)
 - Easy to implement in hardware (not many gates)
 - Must be key agile (key change often, e.g. IPSEC)
 - Have to be parallelized
 - Need to work as hash function
 - Candidates
 - MARS, RC6, RIJNDAEL, SERPENT, TWOFISH
 - Winner
 - RIJNDAEL

Stream Ciphers

- The “theoretically secure” cipher
 - Every bit of data change
- Solution
 - Generate a continuous stream of random data
 - XOR with cleartext
- Problem
 - What is “random” data and do we really want it?
 - How to generate enough “random” data?

Stream Ciphers (continued)

- Pseudo-randomness
 - Golomb's requirements for stream s with period N to be pseudo-random
 - Number of ones and zero should differ by more than one
 - For stretches, at least
 - Half should have length 1
 - A quarter should have length 2
 - An eighth should have length 3
 - Etc.
 - The following auto-correlation equation holds

$$N \cdot \kappa(t) = \sum_{i=0}^{N-1} (2s_i - 1)(2s_{i+t} - 1) = \begin{cases} N, & \text{if } t=0 \\ R, & \text{if } 1 \leq t \leq N-1 \end{cases}$$

Stream Ciphers (continued)

- Random number generators

- Primitive polynomials

- Recursive formulas $D_{m-1}^{t+1} = D_{m-1}^t + D_{m-2}^t + \dots + D_1^t + D_0^t \pmod{2}$
 $D_j^{t+1} = D_{j+1}^t$

- Example $1 + x + x^3$
 $\Rightarrow D_3^{t+1} = D_3^t + D_1^t + D_0^t \pmod{2}$
 $\Rightarrow D_2^{t+1} = D_3^t$
 $\Rightarrow D_1^{t+1} = D_2^t$
 $\Rightarrow D_0^{t+1} = D_1^t$

- Linear feedback shift registers

- Prone to cleantext / ciphertext attacks
 - Use non-linear functions on output of multiple LFSR's
 - Chain LFSR's

- A5 system used to encrypt GSM

Assymmetric ciphers

- Problems
 - How to replace keys and ciphers?
 - Authenticity
 - Trust
- Solutions
 - Mathematically hard problems
 - Factorization of primes (RSA)
 - Easy to multiply primes
 - Difficult to factorize the result
 - Discrete logarithms (ElGamal)
 - Easy to calculate exponentials of primes
 - Difficult to calculate discrete logarithms

Assymmetric ciphers (continued)

- The Mathematics

- Euler's function $\phi(m) = \prod_{i=1}^n (p_i^{e_i} - p_i^{e_i-1})$

- The Algorithm

- Start with primes p, q

- If $n=pq$ then $ed \equiv 1 \pmod{\phi(n)}$

- n, e are your “public” key

- d are your “private” key

- To encrypt data $e(x) \equiv x^e \pmod{n}$

- To decrypt data $d(y) \equiv y^d \pmod{n}$

- Data integrity $e(d(x)) \equiv (x^d)^e \equiv x^{de} \pmod{n} \equiv x \pmod{n}$