

# Cryptography Symmetric Asymmetric Keys

❖ *DigitalSherlock.com | SafeHack.com*

❖ *Date: 2005/05/27*

❖ *Document Name: cryptography\_symmetric\_asymmetric.pdf*

❖ *GNU Free Documentation License*

❖ *Version 1.00, 2005-05-27*

❖ *Copyright © 2005 Adonis, MSc, Eng, CISSP, Security+, CEH, GSec, MCSE, etc.*

❖ *Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation.*

## ❖ Symmetric (Private Keys)

### ➤ Characteristics

- **Both parties will be using the same key for encryption and decryption**
- **Can only provide confidentiality**
- **Are fast and can be hard to break**
- **Sender and receiver use the same key for encryption and decryption**
- **Secret key should be changed frequently**
- **Most symmetric algorithms are published**
- **Work factor is function of the key size**
- **Computational efficiency advantage 1-100 million bits/sec.**
- **Have both public and private information**
  - Public
    - ♦ Algorithm for enciphering plaintext
    - ♦ Possibly some plaintext and cipher text
    - ♦ Possibly encipherment of chosen plaintext
  - Private
    - ♦ The KEY
    - ♦ One cryptographic transformation out of many possible transformations
- **Key Size Compare**
  - 512 Bits Public <> 64 Bits Private
  - 1792 Bits Public <> 112 Bits Private
  - 2304 Bits Public <> 128 Bits Private
  - It is hard to break a key > 128 bits
- **Two main types of symmetric algorithms**
  - Block cipher
    - ♦ A block cipher divides the message into groups of bits and encrypts them
    - ♦ Block ciphers are usually implemented in software
    - ♦ The algorithm is publicly known
    - ♦ DES is a block cipher that divides a message into 64 bit blocks
- **Strengths**
  - Much faster than asymmetric systems
  - Hard to break if using a large key size
  - Provide Confidentiality
- **Weaknesses**
  - Key distribution (requires a secure mechanism to deliver key properly)
  - CANNOT be used to create digital signature
  - Scalability (each pair of users needs a unique pair of keys)
  - Limited Security it provides confidentiality but not authenticity or non-repudiation
  - Does not provide Authentication or nonrepudiation
  - Out-of-band method

- ♦ The key is transmitted through another channel than the message

## ➤ Two types of symmetric algorithms

- **Block ciphers**
  - Block ciphers sometimes work in a mode that emulates a stream cipher
- **Stream Cipher**
  - Operate on continuous streams of plain text (as 1's and 0's)
  - The message is divided into blocks of bits
  - Use Diffusion and confusion in their methods
    - ♦ Confusion - Different unknown key values are used
    - ♦ Diffusion - Putting the bits within the plaintext through many different functions so that they are dispersed throughout the algorithm
  - Uses Substitution boxes (S-boxes) In each step
    - ♦ S-box - Contains a lookup table that instructs how the bits should be permuted or moved around
    - ♦ The key that is used in the decryption process dictates what S-boxes are used and in what order.
  - Operate on small units of plaintext, bits (continuous data stream)
  - It is the key that determines what functions are applied to the plaintext and in what order.
  - Usually implemented in hardware
  - Are more suitable for software implementations, because they work with blocks of data
  - Stream cipher use a keystream generator and encrypt a message one bit at a time
  - The algorithm is publicly known
  - Statistically unpredictable
  - Much faster than any block cipher
  - Effective Stream algorithm contains
    - ♦ Long periods of no repeating patterns within keystream values
    - ♦ Statistically unpredictable
    - ♦ The keystream is not linearly related to the key
  - This method is not much used in modern cryptography
- **Block Cipher**
  - Operate on fixed size blocks of plain text
  - Breaks the plaintext into blocks and encrypts each with the same algorithm
  - The properties of a cipher should contain confusion and diffusion
  - Diffusion
    - ♦ Spread the plaintext character over many ciphertext characters. Done using permutations
    - ♦ Different unknown key values cause confusion
    - ♦ Putting the bits within the plaintext through many functions cause diffusion
    - ♦ Not accomplished through p-boxes
    - ♦ DES implements this product 16 times
  - Confusion
    - ♦ Conceals statistical connection using substitution
    - ♦ Not accomplished through s-boxes
    - ♦ Block cipher use S-boxes
    - ♦ An S-box is non-linear because it generates a 4-bit output string from 6-bit input
  - Are more suitable for software implementations, because they work with blocks of data which is usually the width of a data bus (64 bits).
  - More suitable implemented in software

## ➤ Example of Symmetric Keys

- **DES**
  - DES History

- ◆ Data Encryption Standard
- ◆ DES describe Data Encryption Algorithm (DEA )
- ◆ IBM developed Lucifer in 1974 Adopted by ANSI as DES in (1977)
- ◆ DES - Used for commerical NON-classified
- ◆ Replaced by AES (Rijndael) in November 1998 as announced by NSA
- DES Algorithm
  - ◆ Provide Encryption
  - ◆ Use block encryption (block cipher)
  - ◆ Works on 64-bit blocks of data
    - The key is 56 bits long
  - ◆ Key, begins with 64 bit key and strips 8 parity bits
  - ◆ 56 bits make up the true key and 8 bits are used for parity
  - ◆ 64 bit in -> 64 bit out
  - ◆ DES - divides a message into 64 bit blocks and employs the S-box type functions on them
  - ◆ A block 64 bits is divided in half and DES operate on them one at a time
  - ◆ The characters are put through 16 rounds of transposition and substitution functions
  - ◆ The order and type of transposition and substitution depend on the value of the key
  - ◆ When 64 bits blocks of plaintext go in, 64 bits blocks ciphertext come out
  - ◆ 56 bit key =  $2(56)$  or 70 quadrillion possible keys
  - ◆ Single-Chip LSI implentation
- DES operates in 4 modes
  - ◆ ECB - Electronic Code Book
    - Native mode of DES
    - Provides the recipe of substitutions and permutations that will be performed on the block of plaintext
    - Operate like a Code Book
    - 64 bits data blocks enter directly the device
    - ECB is applied to 64 bits of plain text and produces corresponding 64 bit blocks of ciphertext
    - 64 input vector is broken in to two block (right block and left block)
    - Each 32 bit block is copied into a 48 bit block
    - Each 48 bit block is XORed with a 48 bit encryption key
    - Exists pairs of plain text an corresponding code
    - In EBC mode, a block of plaintext and a key will always give the same ciphertext
    - Mosed Used In
      - Because it work as Block Cipher it is helpful when using encryption in databases
      - ECB is best with small amounts of data
        - Challenge response operations
        - Key management tasks
        - Encrypt PINs in ATM machines
  - ◆ CBC - Cipher Block Chaining
    - Each block of text, the key, and the value based on the pervious block is processed in the algorithm and applied to the next block of text
    - Plaintext block of 64 bits
    - First ciphertext will then be XORed with the next plaintext 64 bit block
    - CBC does not reveal the pattern like ECB
    - IN THIS MODE ERROR WILL PROPAGATE
  - ◆ CFB - Cipher Feedback Mode
    - Is a steam cipher
    - In this mode previously generated ciphertext from the last encrypted block is inputed into the algorithm to generate random values
    - These random values are processed with the current block of plaintext to create ciphertext
    - The ciphertext is used as feedback into the key generation source
    - This mode is used when encrypting individual characters is required

- IN THIS MODE ERROR WILL PROPAGATE
  - ◆ OFB- Output Feedback
    - Is a stream cipher
    - In OFB mode DES block cipher cross the line between block cipher and stream cipher
    - The ciphertext is used as feedback into the key generation source
    - Feedback is used to generate the key stream
    - Therefore the key stream varies
    - OFB use Keystream for encryption and decryption
    - Simulates one-time-pad
    - IN THIS MODE ERROR WILL NOT PROPAGATE
  - DES Security
    - ◆ Two techniques to improve encryption of plaintext: Confusion, Diffusion
    - ◆ Replaced by AES in November 1998
- **Double/Triple DES**
  - Double DES
    - ◆ The key is 112 bits long
    - ◆ Work factor about the same as single DES
    - ◆ DES - EE2 (encrypt key 1, encrypt key 2, decrypt key 1)
    - ◆ No more secure
  - Triple DES
    - ◆ No successful attack reported
    - ◆ Use 48 rounds of computation and up to three different keys
    - ◆ Much more stronger than DES or Double DES
    - ◆ DES - EE3 (encrypt key 1, encrypt key 2, decrypt key 3) - most secure
    - ◆ Provide Encryption
- **IDEA**
  - International Data Encryption Algorithm
  - Developed originally in 1990 as Proposed Encryption Standard (PES)
  - Developed at ETH Zurich in Switzerland by Xuejia Lai
  - Was renamed in 1992 to IDEA
  - Use block encryption (block cipher)
  - Works on 64-bit blocks of data
    - ◆ The key is 128 bits long
  - IDEA operates on 16 bit sub-blocks using algebraic calculations.
  - The 64-bit data block is divided into 16 smaller blocks
  - Each has 8 eight rounds of mathematical functions performed on it
  - Is used in the PGP encryption software
  - Considered to be very secure
  - No practical attacks on it have been published
  - Provide Encryption
- **Blowfish**
  - Blowfish was designed by Bruce Schneier
  - Use block encryption (block cipher)
  - Works on 64-bit blocks of data
  - Variable keys length (up to 448 bits)
  - Flexible symmetric key often used in Secure Shell SSH
  - The data blocks go through 16 rounds of cryptographic functions
  - Blowfish utilizes the idea of randomized S-boxes
  - Provide Encryption
- **RC4**

- Designed by Ron Rivest in 1987
- RC4 - Use Stream Encryption (Stream cipher)
- Stream Cipher encrypt ALL data in real Time
- Variable keys length (128 bits is standard)
- It used to be a trade secret
- Provide Encryption
- **RC5**
  - Use block encryption (block cipher)
  - Variable block size 32/64/128
  - Variable key size from 0 bits to 2040 bits
  - Variable number of rounds from 0 to 255
  - Developed by Ronald Rivest in 1994
  - Cracked 56 bit
  - Provide Encryption
- **Rijndael Block Cipher**
  - October 2 2000 NIST announced the selection of the Rijndael Block Cipher
  - Developed by Joan Daemon and Vincent Rijmen
  - Use block encryption (block cipher)
  - Categorized as an iterated block cipher with variable block length
  - Key size 128, 192, 256 bits
    - ♦  $3.4 \times 10^{38}$  possible 128 bit key combinations
    - ♦  $6.2 \times 10^{57}$  possible 192 bit key combinations
    - ♦  $1.1 \times 10^{77}$  possible 256 bit key combinations
  - Resistance to all known attacks
  - Design Simplicity, work over ATM, ISDN
  - Code compactness and speed on wide variety of platforms
  - Rijndael is a substitution linear transformation cipher
  - It use triple discret invertible uniform transformations
  - Rijndael uses a variable number of rounds
    - ♦ 9 rounds if the key/block size is 128 bits
    - ♦ 11 rounds if the key/block size is 192 bits
    - ♦ 13 rounds if the key/block size is 256 bits
  - Use Three 3 layers of distinct and invertible transformations
    - ♦ The non-linear layer
    - ♦ The linear mixing layer
    - ♦ The key addition layer
- **AES Advanced Encryption Standard**
  - Was announced in January 1997 by NIST as replacement for DES
  - Is the new Federal Information Processing Standard (FIPS) Publication that will specify a cryptographic algorithm for use by U.S. Government organizations
  - Block cipher with a variable block length and key length
  - Use block encryption (block cipher) of 128 bits in size
  - Key sizes of 128, 192 and 256 bits
  - Employs a round transformation
    - ♦ Employs a round transformation that is comprised of three layers of distinct and invertible transformations
    - ♦ The non-linear layer
    - ♦ The linear mixing layer
    - ♦ The key addition layer
  - Royalty-free

- Easy to implement in hardware and software
- In August 1999, NIST selected five algorithms
  - ♦ MARS
    - Developed by IBM (the same team that developed Lucifer)
  - ♦ RC6
    - Developed by RSA
  - ♦ October 2, 2000, NIST announced that Rijndael had been selected
  - ♦ Serpent
    - Developed by Ross Anderson, Eli Biham and Lars Knudsen
  - ♦ Twofish Algorithm
    - Developed by Counterpane based on Blowfish (also by Counterpane) - Bruce Schneier
    - Twofish is a symmetric block cipher that use Transposition.
    - 128 bit blocks in 16 rounds, up to 256 bit keys
    - Twofish uses a single key of any length up to 256 bits
    - Employs pre-whitening before first round and after second round
    - Need to break whitening keys in addition to Twofish key
    - This cipher has key dependent S-boxes like Blowfish
- Is suited for
  - ♦ High speed chips with no area restrictions
  - ♦ Compact co-processor on a smart card
- **Twofish**
  - Twofish was designed by Bruce Schneier
  - Works on 128-bit blocks of data
  - Uses a single key of any length up to 256 bits
  - Much Faster than Blowfish, work on Smart cards
  - Twofish is unpatented, license-free, and freely available for use
- **Skipjack**
  - Designed by NSA, is secret and less well-trusted encryption algorithm
  - Contained in the Clipper chip and Fortezza Chip
  - It use 32 rounds on 64-bit data block
  - It uses an 80-bit key to encrypt 64-bit blocks of data
  - Skipjack is expected to be more secure than DES
  - In June of 1998 Skipjack was declassified by the NSA
- **MARS**
  - By Zunic et. al., IBM
  - Works on 128-bit blocks of data
  - Support key longer than 400 bits
  - Using a special type of a Feistel network, work on Smart cards
- **Serpent**
  - By Anderson, Biham and Knudsen
  - Works on 128-bit blocks of data
  - Support key length up to 256 bits
  - It is optimized for Intel-Based Chips

## ❖ Asymmetric (Public Key)

### ➢ Characteristics

- **Two asymmetric keys are mathematically related, public and private key**
- **Public Key uses two keys Public and Private generated by asymmetric algorithm**
- **Secret key is generated by symmetric algorithm**
- **It is a Hybrid use of two different algorithm: symmetric, asymmetric**

- **Uses 2 asymmetric keys**
  - It use one-way hash  $Y=F(X)$ 
    - ♦ Easy to compute Y if X is given
    - ♦ VERY difficult to derive X given Y
    - ♦ Trapdoor
- **Public Key cannot derive the private Key**
- **CAN be used to create digital signature**
- **Key Size**
  - 512 Bits Private <> 64 Bits Public
  - 1792 Bits Private <> 112 Bits Public
  - 2304 Bits Private <> 128 Bits Public
- **Strenghts**
  - Better key distribution than symmetric systems
  - Better scalability than symmetric systems
  - Can encrypt and provide confidentiality, authentication, and nonrepudiation
- **Weaknesses**
  - 1,000 to 10,000 times slower than symmetric keys
- **Secure message format**
  - Encrypted by the receiver's public key
- **Open message format**
  - Encrypted by the sender's private key
- **Secure and signed format**
  - Encrypted by the senders private key and
  - Then encrypted with the receivers public key

## ➤ **Example of Asymmetric Keys**

- **RSA**
  - Developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman and it is the DE FACTO standard for Digital signatures
  - RSA is currently the most important public key algorithm
  - Security come from Factoring product of 2 large prime
  - In Spetember 2000, the RSA patent expired, so it free for public use
  - As Euclid proved over two thousand years ago, there are infinitely many prime numbers.
  - The key size should be greater than 1024 bits
  - Keys of size, say, 2048 bits should give security for decades
  - Provides
    - ♦ Digital signature
    - ♦ Secure Key exchange / Distribution
    - ♦ Encryption functionality
    - ♦ Authentication (digital signatures)
    - ♦ Both encrypt and digitally sign your email
    - ♦ Included as part of the Web browsers from Microsoft and Netscape
    - ♦ PGP
    - ♦ Government system that use public key
  - Why would anyone not implement RSA?
    - ♦ If there was not enough computing power to generate enough keys for a large network
    - ♦ RSA does require a relatively powerful computer with lots of memory and disk storage to generate multiple keys.
  - Attacked with Factoring attack
- **ECC**

- Elliptic curve cryptosystem (ECC)
- Set of points that satisfy the equation  $y^2 = x^3 + ax + b$
- ECC - It uses an algebraic system defined on points of an elliptic curve to provide public-key algorithms.
- Security come from Calculating discrete logarithms in a finite field
- Employ modular exponentiation
- Elliptic curve discrete logarithm problem
- Uses algebraic system defined on points of elliptic curve
- Proposed by Victor Miller 1985 & Neal Koblitz
- Provides
  - ♦ Digital signature
  - ♦ Secure Key exchange / Distribution
  - ♦ Encryption functionality
- Advantages of Elliptic Curves
  - ♦ Highest strength/bit of public key systems
  - ♦ Ideal for very small hardware implementations e.g wireless device and cell phone
  - ♦ Big saving over other public key systems
  - ♦ Computation, Bandwidth, Storage
  - ♦ Bandwidth reduced
  - ♦ Short signature and certificates
  - ♦ Fast encryption and signature speed
  - ♦ Hardware and software, Smart card
  - ♦ Separable Stages, Encryption and digital signatures stages separable to simplify export
  - ♦ Shorter key sizes can be used to achieve the same security of conventional public-key cryptosystems
- **Diffie-Hellman**
  - Invented in 1976 By W.Diffie M.E hellman
  - First public key algorithm
  - Diffie-Hellman - Uses a key exchange algorithm (KEA)
  - Diffie-Hellman is the Primary Alternative to RSA
  - Security come from Calculating discrete logarithms in a finite field
  - Not used for message encryption/decryption
  - Patent expired in 1997 and was released to public
  - Diffie-Hellman protocol and RSA appear to have remained two of the strongest up to now.
  - Allowing the construction of a common secret key over an insecure communication channel
  - Usually Diffie-Hellman is implemented in software
  - KEA - Key Exchange Algorithm is a variation of diffie-hellman used for key exchange
  - Discrete logarithm algorithms can be used to attack Diffie-Hellman
  - Attacks against Diffie-Hellman include also the man-in-the-middle attack
  - The Diffie-Hellmann key exchange is vulnerable to a middleperson attack.
  - Possible solutions include the use of digital signatures and other protocol variants.
  - Provides
    - ♦ Secure Key exchange / Distribution
- **EI Gamal**
  - Provides
    - ♦ Digital Signature
    - ♦ Secure Key exchange / Distribution
  - Is public key algorithm
  - Based on Calculating discrete logarithms in a finite field

- This is a straightforward extension of Diffie/Hellman's
- Generates a shared secret and uses it as a one-time pad to encrypt one block of data
- **Digital Signature Standard (DSS)**
  - NIST proposed in 1991
  - US standard developed for creating Secure message digests
  - Uses secure hash algorithm (SHA) and DSA Digital Signature Algorithm
  - Enables use of RSA digital signature algorithm
  - Key size 512-1024 bits
  - Based on modified El-Gamal
  - RSA and DSA are the best known and most widely used digital signature algorithm
  - Use the Secure Hash Algorithm . The SHA computes a fixed length message digest from a variable length input message.
  - This message digest is then processed by the DSA to either generate or verify the signature.
  - Does not provide encryption only provide digital signature
  - DSS uses public keys to produce a digital signature but it can't encrypt or decrypt data messages by itself.
- **Digital signature algorithm (DSA)**
  - Produces a digital signature in the form of a pair of large numbers.
  - Uses a private key to sign a message and a public key to verify the signature.
  - Proposed by the U.S. Government. (NIST) made available to public in 1994
  - DSA is used to Digitally sign the message
- **Merkle-Hellman Knapsack**
  - Having set of items with fixed weights
  - Determining which items can be added in order to obtain a given total weight
  - Illustrated using Super increasing weights
- **Activities Involving Elliptic Curve**
  - IEEE, P1363 (public-key crypto)
  - Covers main public key techniques
  - RSA, ECC, El Gamal, Diffie-Hellman
  - ANSI X9
  - Elliptic curve Digital Signature Algorithm
  - (ECDSA) proposed work item
  - ANSI ASC X9
  - Elliptic curve key agreement and key management proposed work item
  - ISO/IEC CD 148883 "Digital Signature with appendix"
  - Variety of digital signature mechanisms
- **Calculating discrete logarithms in a finite field**
  - El Gamal
  - Diffie-Hellman
  - Shnorrs signature Algorithm
  - Elliptic Curve
  - Nybergueppels signature algorithm
- **Factoring product of 2 large prime**
  - RSA

## ❖ Symmetric Vs. Asymmetric

### ➤ Symmetric

- **Key - One key is shared between two entities**

- **Key Exchange - Out-of-Hand**
- **Key Length - Fixed Key length**
- **Speed - Algorithm is less complex and Faster**
- **Use - Bulk encryption means encrypting files and communication path**
- **Provides**
  - Confidentiality
  - Integrity

➤ **Asymmetric**

- **Key - Two Keys, one entity has a public key another entity has Private Key**
- **Key Exchange - Symmetric key is encrypted and sent with a message**
- **Speed - Algorithm is more complex and Slower**
- **Key Length - Variable Key Length**
- **Use - Key encryption and Key distribution**
- **Provides**
  - Confidentiality
  - Integrity
  - Authentication
  - Nonrepudiation