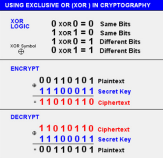
**Module 6 Cryptography & PKI**

**6.2 Crypto-Algorithms**

**Obfuscation**

* Act of making something difficult to understand
* Substitution cipher (Eg. ROT13 – rotate 13 places)
* XOR – exclusive OR

1. Example



**Symmetric Algorithms**

* DES (Data Encryption Standard)

1. Adopted by NIST in 1977
2. Block cipher using 64-bit blocks – 56-bit key + 8 bits parity
3. Short key length subject to brute-force attacks

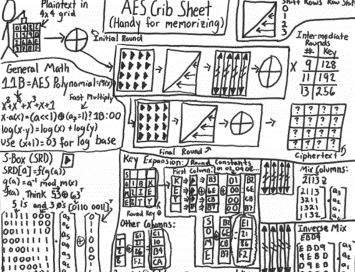
* 3DES (Triple DES)

1. DES algorithm computed 3 times
2. Using “key bundle” that comprises 3 different DES keys, each 56-bit = total bit strength of 163 bits (AKA 3TDEA)
3. Options to reuse keys

* AES (Advanced Encryption Standard)

1. Original name Rijndael
2. Free for any use public/private/commercial/non-commercial
3. Adopted by NIST in 2001
4. Block cipher with 128 bit block size
5. 3 key lengths – 128, 192 & 256 bits
6. Uses multiple encryption rounds – 10 rounds for 128-bit keys, 12 rounds for 192-bit keys & 14 rounds for 256-bit keys

**AES Encryption Explained**



**Symmetric Algorithms**

* RC4/RC5/RC6 (Rivest Cipher)

1. RC4 is stream cipher, RC5 & RC6 are block ciphers
2. Works with key sizes between 40 & 2048 bits

* Blowfish/Twofish

1. Symmetric block cipher that can use variable-length keys (from 32 to 448 bits)
2. Twofish uses 128-bit blocks

* International Data Encryption Algorithm (IDEA)

1. 128-bit key
2. Similar to DES but more secure
3. Used in Pretty Good Privacy (PGP)

* One-Time Pad (OTP)

1. Most secure crypto implementation
2. Use of key as long as plain-text message
3. Only used once & then destroyed

* Skipjack

1. NSA developed block cipher used in clipper chip
2. Uses 80-bit key to encrypt 64-bit blocks of data

**Cipher Modes**

* CTR (Counter Mode)

1. Turns block cipher into stream cipher
2. Used to generate key stream
3. Each block combines nonce/IV with sequentially assigned number to produce unique counter block that is then encrypted

* CBC (Cipher-Block Chaining)

1. Uses IV with 1st block
2. Thereafter each block of plain text obfuscated with cipher text from previous block before its encrypted
3. Introduces more diffusion & reduces effects of plain-text attacks

* GCM (Galois/Counter Mode)

1. Provides both integrity & confidentiality
2. GCM uses CTR with 128-bit blocks
3. Each 128-bit block given encrypted number. That result then obfuscated with plain text producing cipher text
4. GMAC (Galois Message Authentication Code) – authentication-only variant of GCM which can be used as incremental message authentication code

* ECB (Electronic CodeBook)

1. Divides message into blocks & encrypts each block
2. No longer recommended – same plain-text block encrypted into same cipher-text block each time

**Asymmetric Encryption**

* Uses 2 keys – 1 to encrypt other to decrypt
* Keys mathematically related
* Public/private key encryption
* Only private key kept secret
* Only private key can decrypt message

**Asymmetric Algorithms**

* Extra computational overhead
* Used primarily for

1. Secure exchange of shared keys used for symmetric encryption
2. Digital signatures

* Solves issue of key exchange with symmetric encryption
* Rivest, Shamir & Adleman (RSA)

1. Used for key exchange & digital signatures
2. Key can be any length
3. Algorithm works by multiplying 2 large prime numbers
4. Derives 2 different numbers; 1 public key other private key

* Diffie-Hellman key exchange (D-H)

1. 2 parties, w/o prior agreement can agree on secret key that is known only to them
2. Only used to generate shared key (not encryption)
3. Key can be safely/secretly shared on public network

* Diffie-Hellman Ephemeral (DHE)

1. Uses different key for every conversation
2. Supports Perfect Forward Secrecy (PFS)

* Elliptic Curve Cryptography (ECC)

1. Technique using elliptic curves to calculate simple but difficult-to-break encryption keys
2. Uses smaller key sizes to obtain same level of security (160-bit ECC = 1024-bit RSA)

* Elliptic Curve Diffie-Hellman Ephemeral (ECDHE)

1. Variant of DHE using ECC for Perfect Forward Secrecy

* El Gamal – extension to Diffie-Hellman using ephemeral key
* Pretty Good Privacy (PGP) & GNU Privacy Guard (GPG)

1. Developed by Philip R. Zimmerman in 1991
2. Used to encrypt & sign email messages

**Hashing**

* “Digital fingerprint”
* Work by taking string of any length & producing fixed-length string for output
* Changing original changes hash value
* Originator takes hash of file & provides hash to receiver which then compares it with received file hash to ensure file integrity

**Hashing Algorithms**

* Secure Hash Algorithm (SHA, SHA-1, SHA-2, SHA-3)

1. Developed by US NSA
2. SHA-1 can generate 160-bit hash from any variable-length string of data
3. SHA-2 = SHA-224, SHA-256, SHA-384 & SHA-512 (based on digest lengths)
4. SHA-3 published in 2012. Not widely used yet

* Message Digest Algorithm (MD2, MD4, MD5)

1. Most widely known hashing functions
2. Produces 16-byte hash value usually expressed as 32 digit hexadecimal number
3. Considered compromised. Rainbow tables have been published which allow people to reverse MD5 hashes made w/o good salts

* Message Authentication Code (MAC)

1. Authentication of messages using secret key
2. Used in electronic fund transfers to protect against fraud

* Hash-based Message Authentication Code (HMAC)

1. Combines cryptographic hash function & secret cryptographic key
2. Does not encrypt message, only key

* RACE Integrity Primitives Evaluation Message Digest (RIPEMD)

1. Design based on MD4
2. 160-bit version of algorithm (RIPEMD-160) performs comparably to SHA-1

**Rainbow Tables & Salts**

* Rainbow Table

1. Pre-computed table for reversing cryptographic hash functions
2. All of possible hashes computed in advance

* Salt – random data used as additional input to hashes

**Key Stretching**

* Processes used to take weak key & make it stronger (usually longer)
* Bcrypt

1. Based on Blowfish algorithm
2. Provides adaptive hash function based on key factor

* PBKDF2 (Password-Based Key Derivation Function 2)

1. Algorithm applies pseudo-random function to password, combined with salt of at least 64 bits & then repeats process at least 1000 times