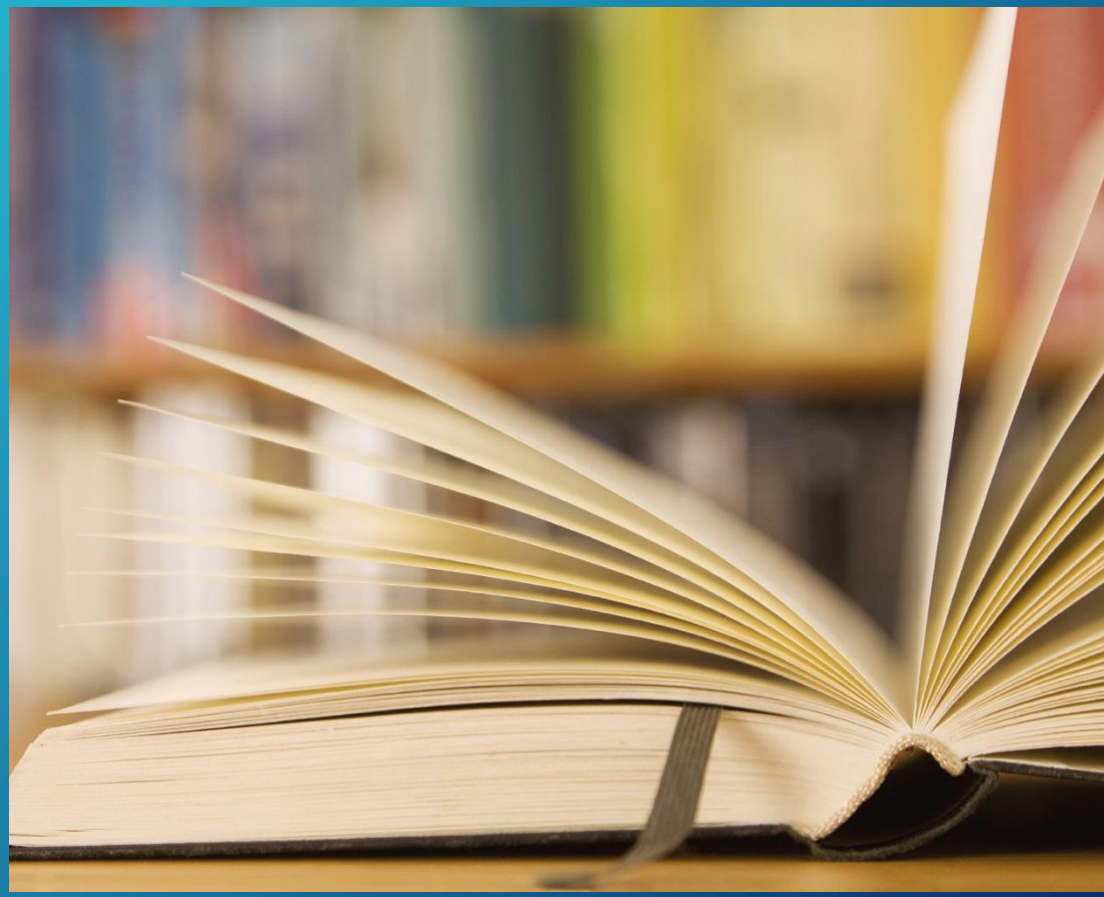
SEKOLAH TINGGI TEKNOLOGI
TERPADU NURUL FIKRI
CHARACTER BUILDING CAMPUS
STT - NF

KRIPTOGRAFI

Keamanan Sistem Informasi – Aditya Putra, ST., MT.



Agenda

- Cryptography Concepts
- Encryption Algorithm/Ciphers: DES & AES
- Hash Concepts
- Hash Algorithm/Ciphers
- Encryption Tools
- Public Key Infrastructure (PKI)
- Signed Certificate (CA) vs Self Signed Certificate
- Cryptography Attack
- Countermeasure - How to Defend

Cryptography Concepts

- Kriptografi merupakan seni untuk menjaga pesan agar aman. Proses pengamanan pesan ini sering dikenal dengan Enkripsi
- Proses kriptografi memungkinan konversi plaintext menjadi text acak yang terenkripsi.
- Algoritma enkripsi juga disebut dengan cipher, sehingga text yang telah dienkripsi disebut dengan ciphertext
- Implementasi Kriptografi telah dilakukan untuk mengamankan data penting seperti email, pesan chat, transaksi web, data personal, data korporat, aplikasi e-commerce, dll

TIPE-TIPE KRIPTOGRAFI

Symmetric Encryption

- Enkripsi Simetris menggunakan key yang sama pada saat melakukan enkripsi dan dekripsi

Asymmetric Encryption

- Pada Enkripsi Asimetris, key yang digunakan pada saat enkripsi berbeda dengan key yang digunakan untuk melakukan dekripsi. Key ini dikenal dengan public dan private key

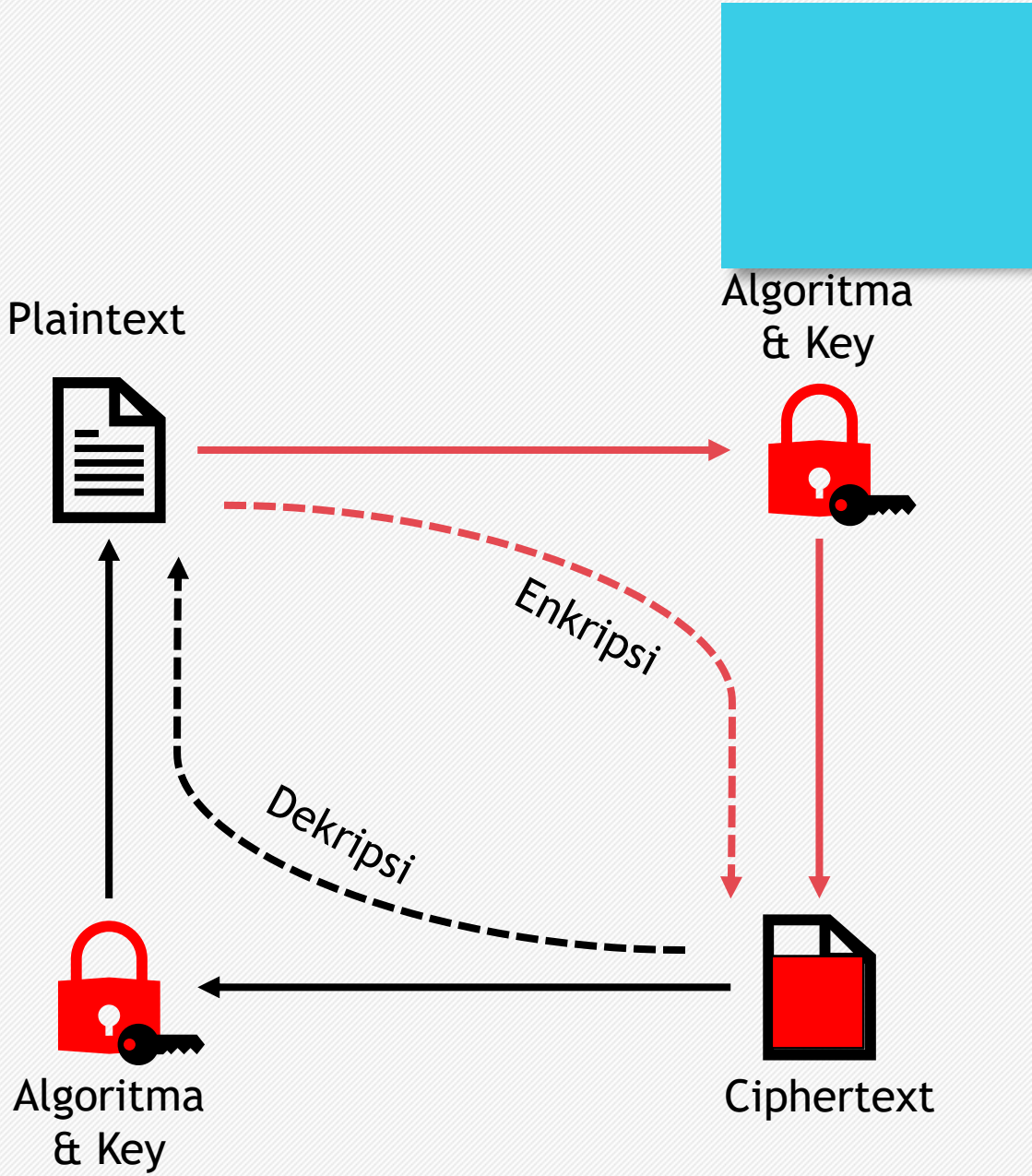
Cryptography Concepts

Enkripsi adalah proses untuk mengubah pesan atau teks asli *(plaintext)* menjadi *ciphertext* sehingga informasi yang terkandung pada teks tersebut tersembunyi.

Elemen dari Enkripsi

- **Algoritma** adalah fungsi yang digunakan untuk melakukan enkripsi dan dekripsi
- **Key** adalah salah satu factor yang menentukan kekuatan suatu enkripsi
- **Plaintext** merupakan pesan atau teks asli
- **Ciphertext** teks output hasil enkripsi

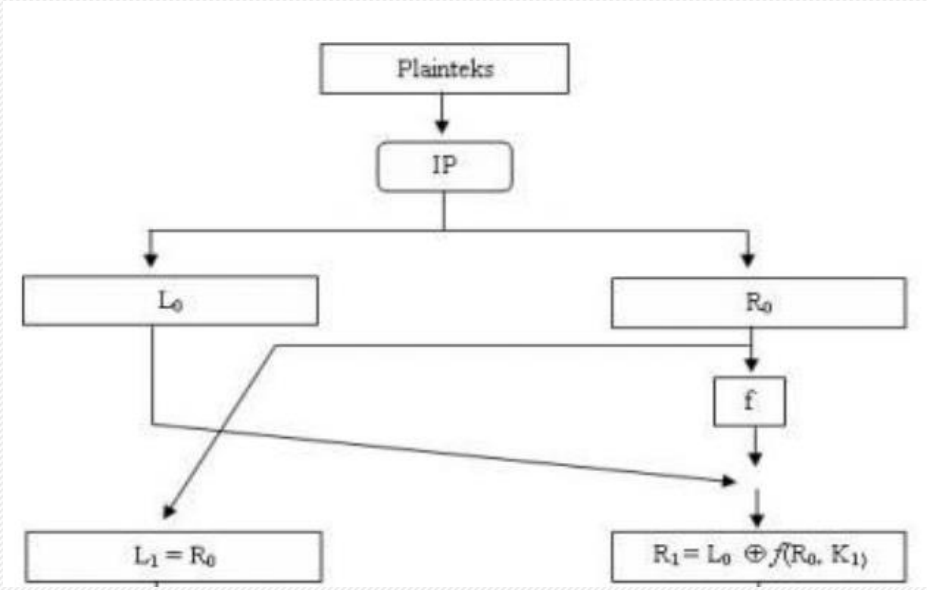
Enkripsi bersifat dua arah. Artinya Ciphertext dapat dikembalikan menjadi Plaintext dan proses ini dikenal dengan **Dekripsi**



Encryption Algorithm/Ciphers

Data Encryption Standard (DES)

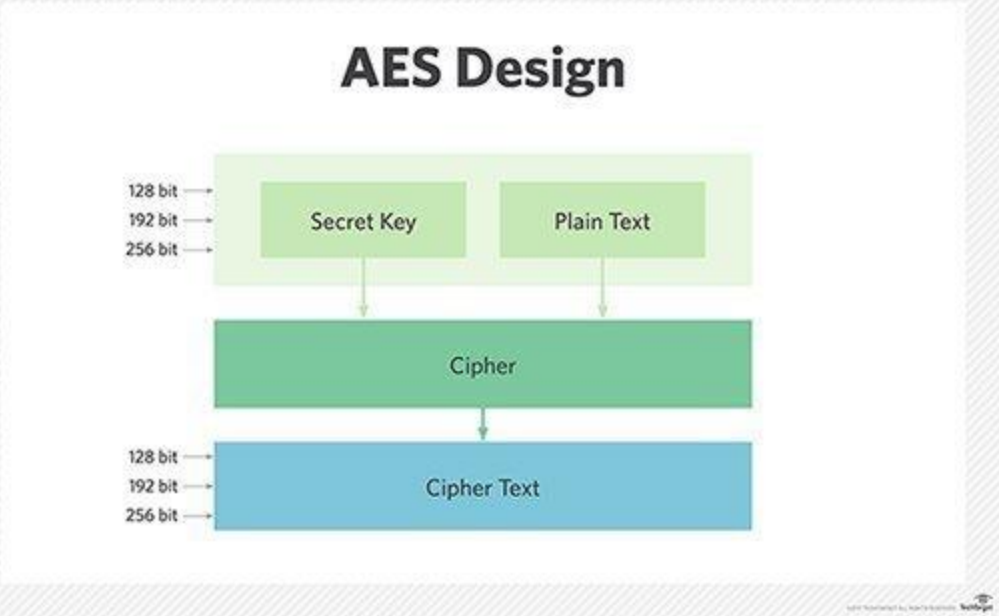
- DES didesain untuk melakukan encipher dan dechiper blok data dalam bentuk 64 bit data yang dikendalikan menggunakan key yang berjumlah 56 bit
- DES adalah algoritma yang mengubah bit plaintext menjadi bit ciphertext dengan jumlah/Panjang bit yang sama
- Pada awalnya, dengan panjang kunci 56 bit, serangan brute force memerlukan waktu 1.142 tahun hingga terpecahkan. Namun pada tahun 1999 DES berhasil dipecahkan kurang dari 1 hari.
- Atas kelemahan ini beberapa pengguna DES melakukan proses pengulangan ciphering hingga 3 kali yang disebut dengan 3DES untuk meningkatkan keamanan DES



Encryption Algorithm/Ciphers

Advanced Encryption Standard (AES)

- AES adalah algoritma key simetris menggunakan algoritma baru **Rijndael** dan menggantikan DES karena isu keamanan
- AES menggunakan kunci 128 (AES-128), 192 (AES-192), dan 256 (AES-256) dengan ukuran blok yang tetap yaitu 128 bit
- Penggunaan AES memenuhi 3 tujuan:
  - Keamanan → harus minimal memiliki tingkat keamanan yang sama dengan 3DES
  - Harga → tanpa royalty sehingga murah untuk diterapkan pada smartcard dengan memory yang kecil
  - Implementasi → efisien dan minimal secepat 3DES saat dijalankan pada mesin 8 hinggat 64 bit dan berbagai perangkat lunak

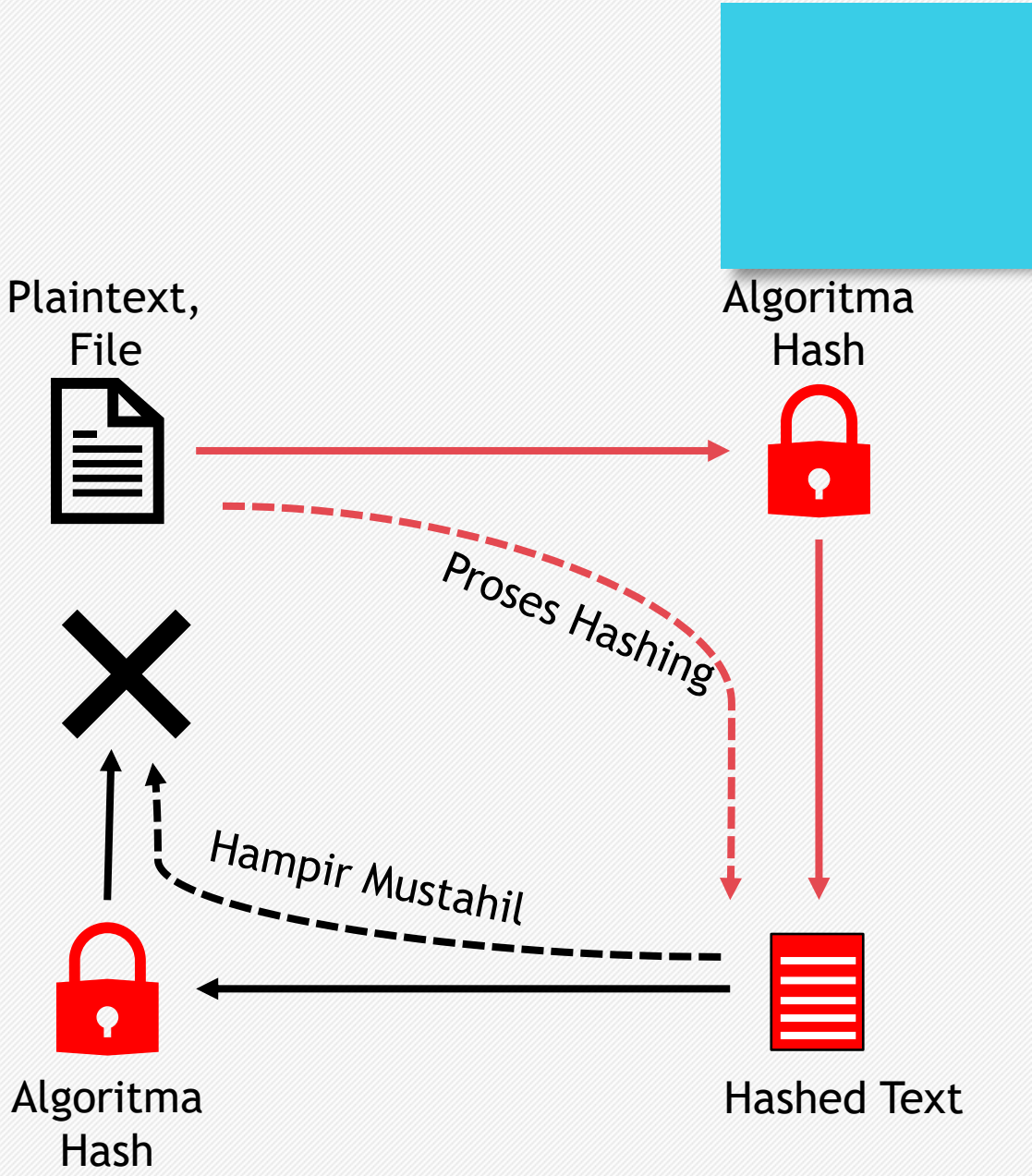


Hash (1)

Berbeda dengan enkripsi, Hash adalah fungsi yang bersifat 1 arah atau dengan kata lain, *hashed text* hampir mustahil untuk bisa dikembalikan kembali menjadi *plaintext*

Perbedaan lain antara Hash dan enkripsi adalah tujuannya. Tujuan yang ingin dicapai enkripsi adalah *CONFIDENTIALITY*, maka pada Hash yang ingin dicapai adalah *INTEGRITY*

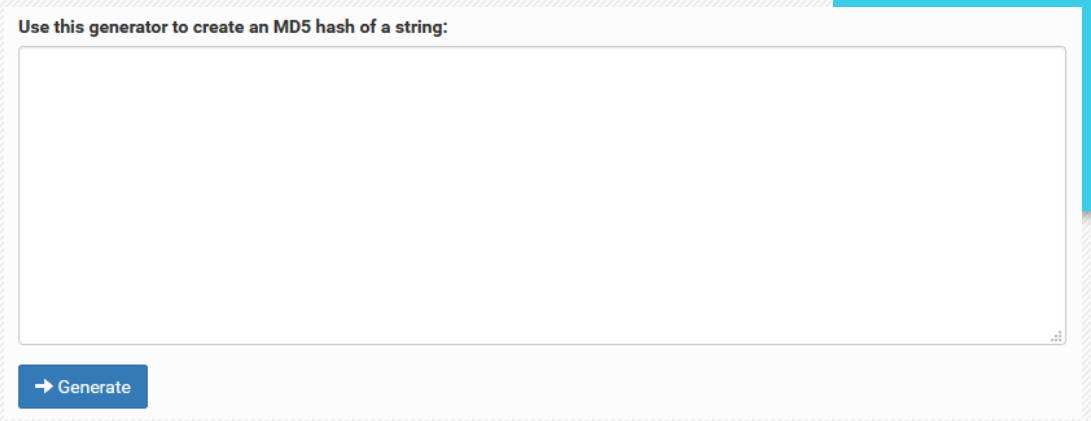
Dari awal, Enkripsi dilakukan untuk mengamankan informasi supaya jika data berhasil disadap atau dicuri maka nilai informasinya masih terjaga. Sehingga hanya pemilik sah dari data itu saja yang mampu untuk mendekripsikannya Kembali. Ini tujuan CONFIDENTIALITY-nya

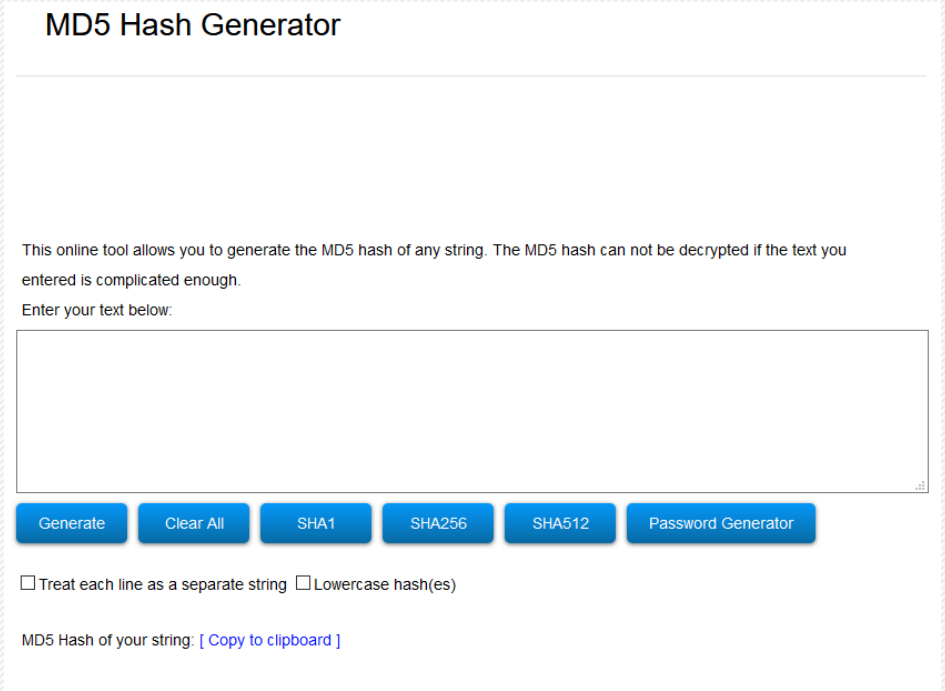


Hash Algorithm

Message digest Function (MD5)

- Algoritma MD5 melakukan proses dengan mengeluarkan output berupa 128-bit fingerprint berdasarkan inputnya
- MD5 tidak collision resistant, artinya tidak selamanya menghasilkan output yang unik, sehingga disarankan menggunakan hash algoritma terkini seperti SHA2 dan SHA3
- Namun demikian algoritma ini masih digunakan pada banyak aplikasi digital signature, file integrity checkin, dan proses storing passwords





Hash Algorithm

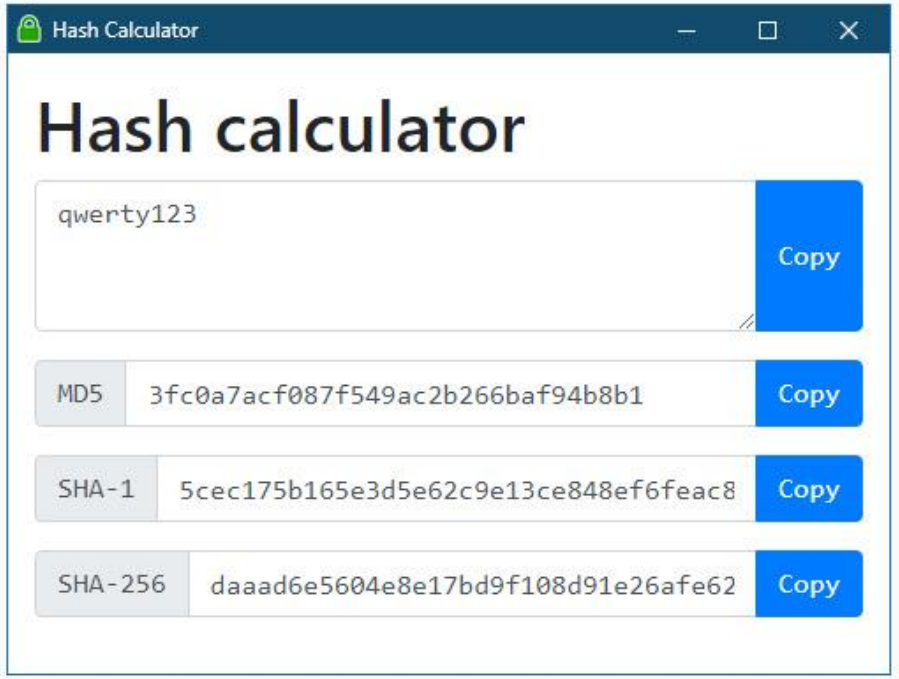
Secure Hashing Algorithm (SHA)

- Algoritma ini dipublish oleh National Institute of Standards and Technology (NIST) sebagai US Federal Information Processing Standard
- SHA1

  Memproduksi 160-bit digest dari plaintext dengan Panjang maksimum (264-1) bits dan memiliki kemiripan dengan MD5
- SHA 2

  Kelompok dari 2 fungsi hash yang sama yang menggunakan ukuran blok yang berbeda yaitu SHA-256 (32-bit words) dan SHA-512 (64-bit words)
- SHA3

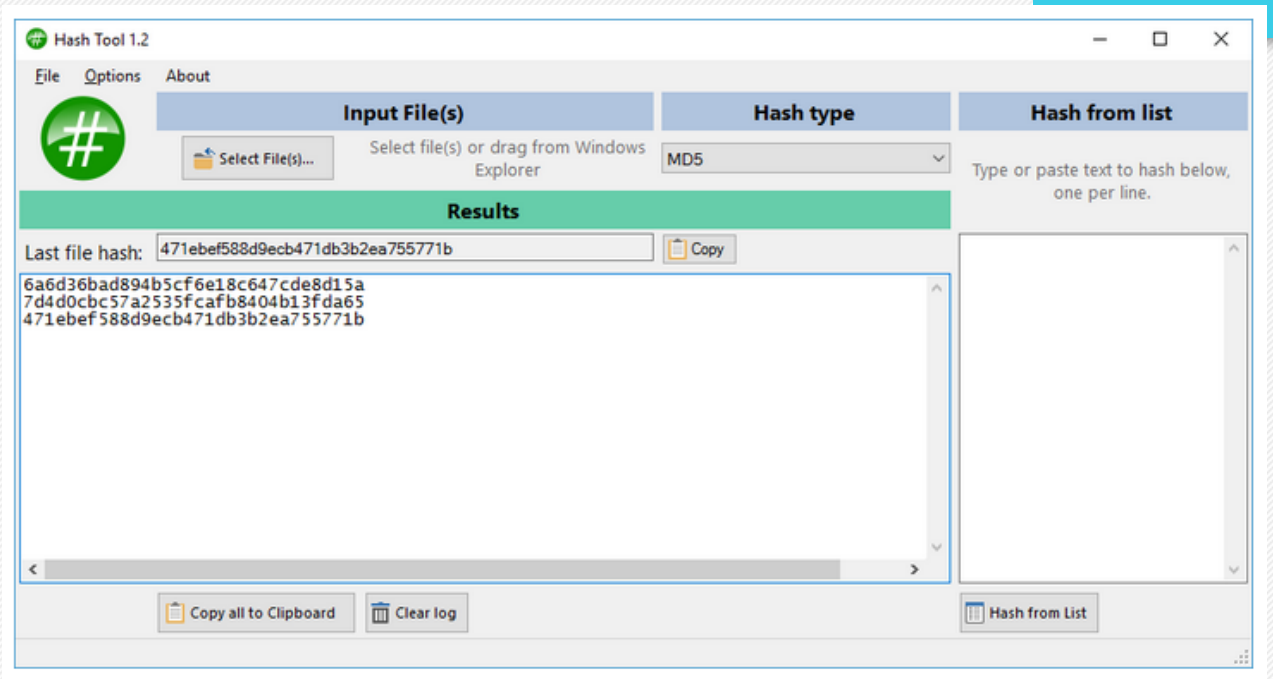
  Algoritma ini menggunakan sponge construction yang melakukan XOR terhadap message blok menjadi initial bit of the state, yang kemudian dipermutasi



Encryption/Hash Tools

Hash Generator Tools

- Hash Calc (www.slavasoft.com)
- Hash Calculator (www.mcafee.com)
- Hash Tool (www.digitalvolcano.co.uk)
- OnlineMD5 (onlinemd5.com)



Encryption Tools

- AxCrypt
- Folder Lock
- CryptoExpert 8
- CertainSafe
- VeraCrypt
- winAES
- GNU Privacy Guard
- AES Crypt
- Steganos LockNote

Public Key Infrastructure (PKI)

PKI adalah kumpulan hardware, software, orang, kebijakan dan prosedur yang dibutuhkan untuk membuat, mengatur, mendistribusikan, menggunakan, menyimpan, dan mencabut digital certificate

Komponen PKI

- **Certificate Management Systems:** Membuat, mendistribusikan, menyimpan dan memverifikasi sertifikat
- **Digital Certificate:** Membangun credential seseorang yang sedang melakukan transaksi online
- **Validation Authority (VA):** Menyimpan sertifikat (dengan public key-nya)
- **Certificate Authority (CA):** Mengeluarkan dan memverifikasi digital certificate
- **End User:** Pelaku yang meminta, mengatur, dan menggunakan sertifikat
- **Registration Authority (RA):** Berperan sebagai pelaku verifikasi untuk CA

Signed Certificate (CA) vs Self Signed Certificate

SIGNED CERTIFICATE

- Pengguna menggunakan trustworthy CA dan membeli digital certificate
- Pengguna mendapatkan public key dari CA yang digunakan untuk menandatangani dokumen
- Dokumen tersebut kemudian dikirim kepada penerima
- Penerima dapat melakukan verifikasi sertifikat dengan menanyakan pada VA
- VA melakukan verifikasi sertifikat kepada penerima tanpa menunjukkan private key-nya

SELF-SIGNED CERTIFICATE

- Pengguna membuat public dan private key menggunakan tools seperti Adobe Reader, Java's Keytool, Apple's Keychain, dll
- Pengguna menggunakan public key untuk menandatangani dokumen
- Dokumen tersebut kemudian dikirim kepada penerima
- Penerima meminta kepada pengguna private key untuknya
- Pengguna memberikan private key kepada penerima

Cryptography Attack

BRUTE-FORCE ATTACK

- Serangan berupa upaya menebak key dengan mencoba satu persatu kemungkinan hingga terpecahkan
- Serangan ini membutuhkan resource yang tinggi dan waktu yang lama, namun lebih pasti akan terpecahkan
- Kesuksesannya tergantung dari Panjang key, waktu dan mekanisme keamanan yang digunakan

HASH COLLISION ATTACK

- Serangan ini dilakukan untuk menemukan input yang berbeda dari output hash yang sama
- Ini dimungkinkan jika dilakukan cryptanalysis dengan melakukan eksploitasi digital signature untuk mendecode data
- SHA-1 mengeluarkan output berupa kombinasi angka dan huruf dalam Panjang string yang sama

Cryptography Attack

RAINBOW TABLE ATTACK

- Serangan ini bertipa serangan kriptografi dimana attacker menggunakan rainbow table untuk melakukan reverse fungsi cryptographic suatu hash
- Rainbow table adalah table yang sudah dikomputasi berupa kumpulan word lists seperti kamus file dan brute force list dan nilai hash-nya
- Attacker melakukan proses hash terhadapa kemungkinan password dan membangdingkannya dengan table hash yang telah dikomputasi tersebut (rainbow table).

Countermeasure - How to Defend

- Akses cryptographic key (password) harus diberikan kepada aplikasi/pengguna secara langsung
- Install IDS dan lakukan monitoring terhadap pertukaran dan akses terhadap key
- Password harus digunakan untuk meng-enkripsi key jika disimpan dalam disk
- Key tidak boleh disimpan dalam source code
- Pada algoritma simetris, ukuran key haruslah 168 atau 256 bit, khususnya untuk transaksi yang besar
- Pada hash, gunakanlah ukuran key 168 atau 256 bit
- Gunakanlah hanya tools atau produk yang terpercaya dalam memilih algoritma kriptografi
- Batasi maksimal input username/password yang salah dalam durasi waktu tertentu