Keamanan Sistem Komputer

Authentication, Hash Function,

Digital Signatures, Quantum Cryptography

Identification vs Authentication

- Identifikasi, menyatakan identitas suatu subjek
- Autentikasi, membuktikan kebenaran identitas yang dinyatakan

Authentication

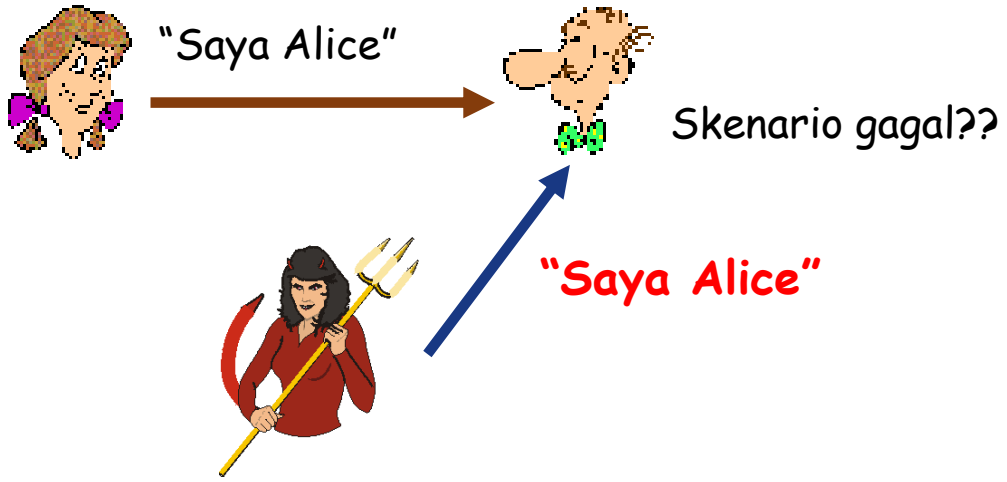
Mekanisme autentikasi dapat menggunakan:

- Sesuatu yang diketahui user (Password, PIN, secret handshake, dll)
- Sesuatu yang ada pada user (Biometrics: fingerprint, voice pattern, face, dll)
- Sesuatu yang dimiliki user (kartu identitas, kartu ATM, kunci, dll)

Pada jaringan, tiga hal tersebut tidak dapat dibuktikan secara otentik sehingga terjadi permasalahan baru

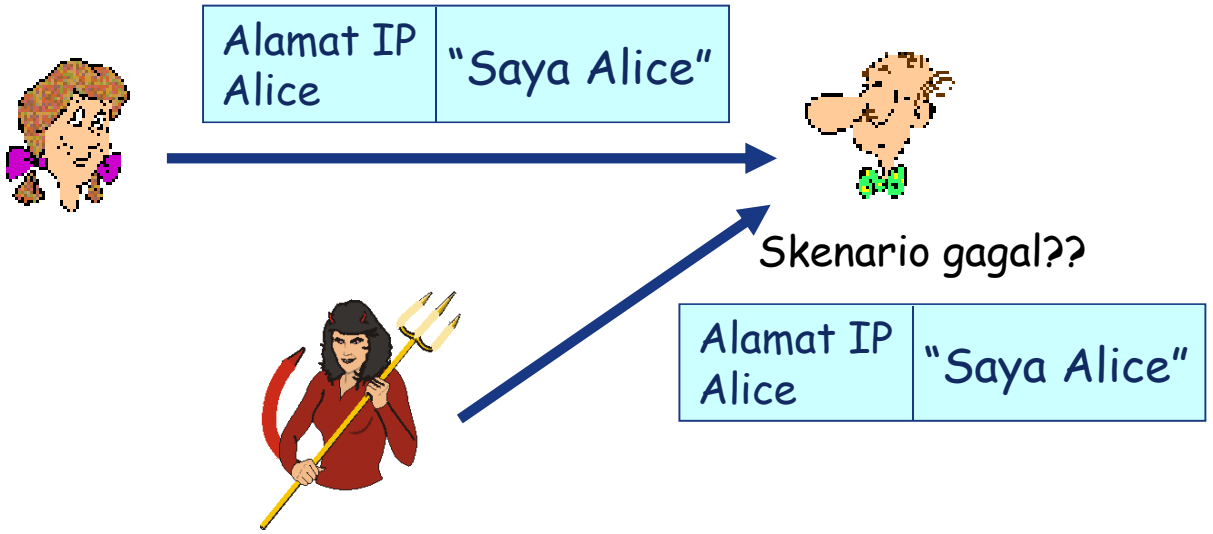
Authentication : Usaha pertama

- Tujuan : Bob ingin Alice membuktikan identitas dirinya
  - Protokol ap1.0 : Alice mengatakan “Saya Alice”



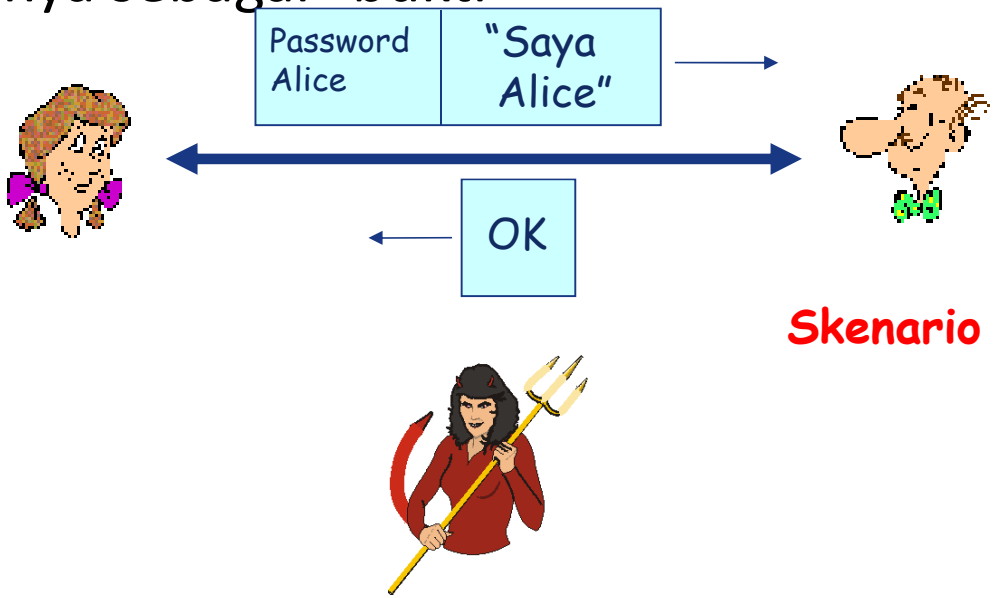
Authentication : Usaha Kedua

- Protokol ap2.0 : Alice mengatakan “Saya Alice” di paket IP mengandung alamat IPnya



Authentication : Usaha Ketiga

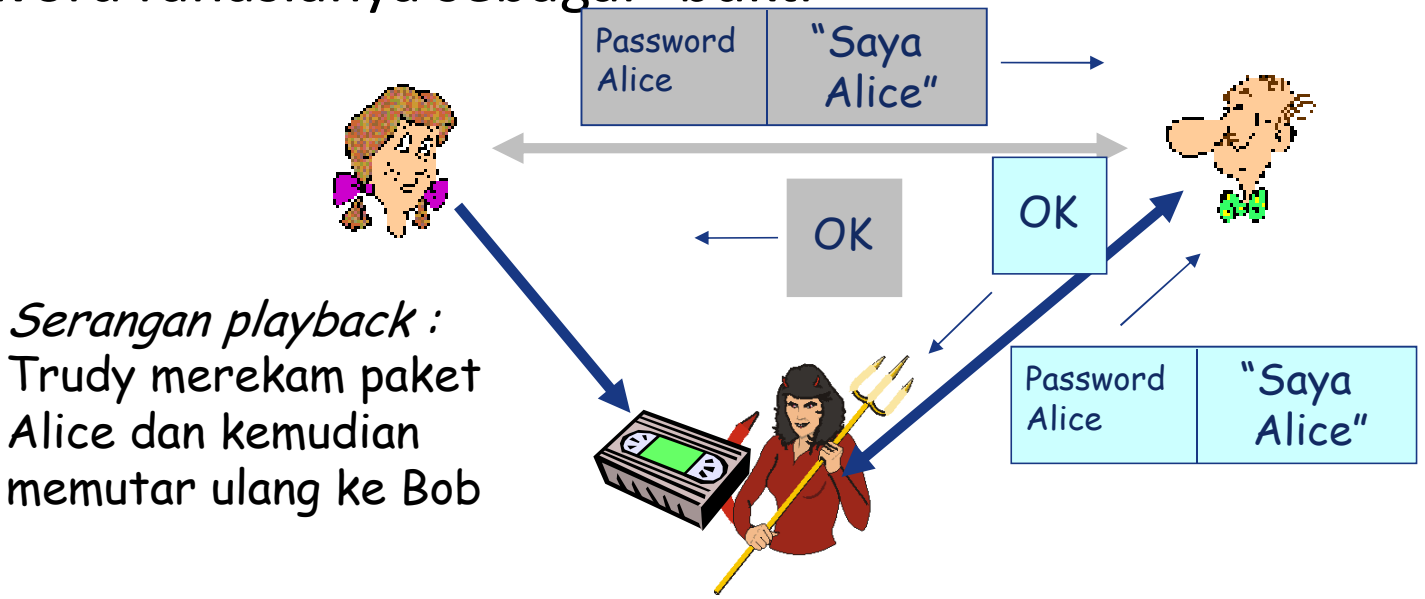
- Protokol ap3.0 : Alice menyatakan “Saya Alice” dan mengirim password rahasianya sebagai “bukti”



Skenario gagal??

Authentication : Usaha Ketiga

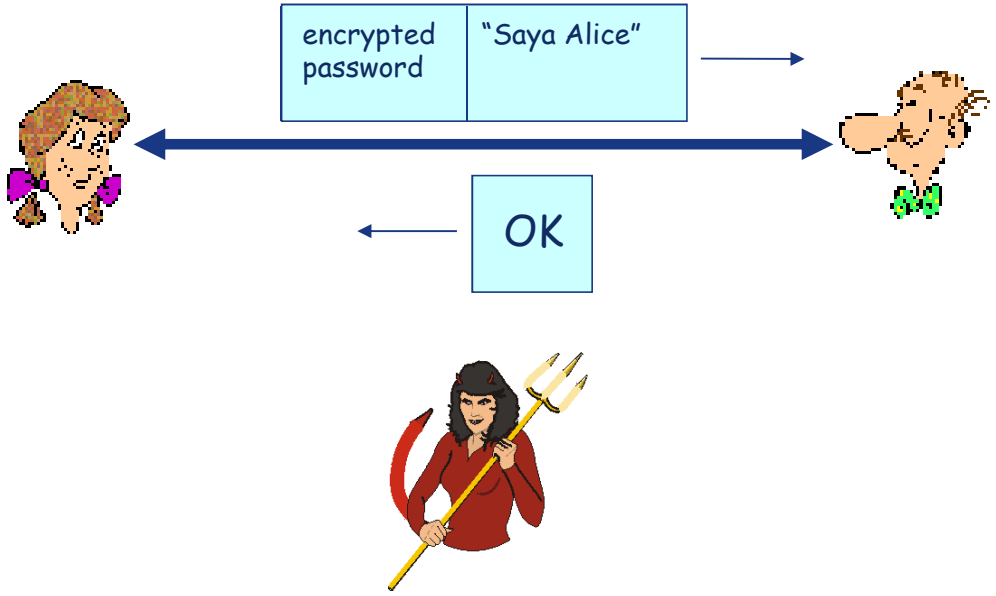
- Protokol ap3.0 : Alice menyatakan “Saya Alice” dan mengirim password rahasianya sebagai “bukti”



Serangan playback :
Trudy merekam paket Alice dan kemudian memutar ulang ke Bob

Authentication : Usaha Ketiga (Rev.)

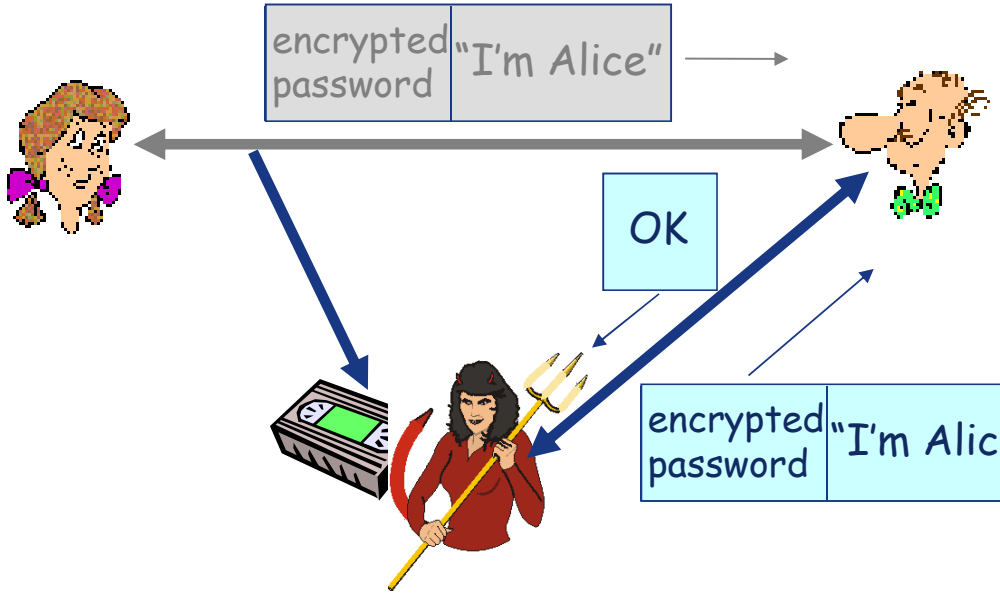
- Protokol ap3.1 : Alice menyatakan “Saya Alice” dan mengirim password rahasia terenkripsi sebagai “bukti”



Skenario gagal??

Authentication : Usaha Ketiga (Rev.)

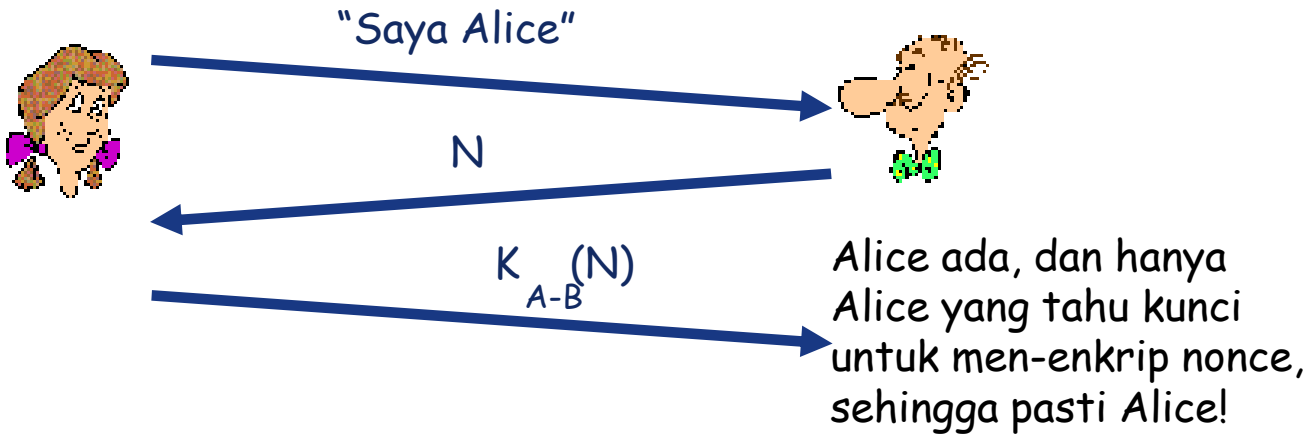
- Protokol ap3.1 : Alice menyatakan “Saya Alice” dan mengirim password rahasia terenkripsi sebagai “bukti”



Rekam!! Dan Playback!!

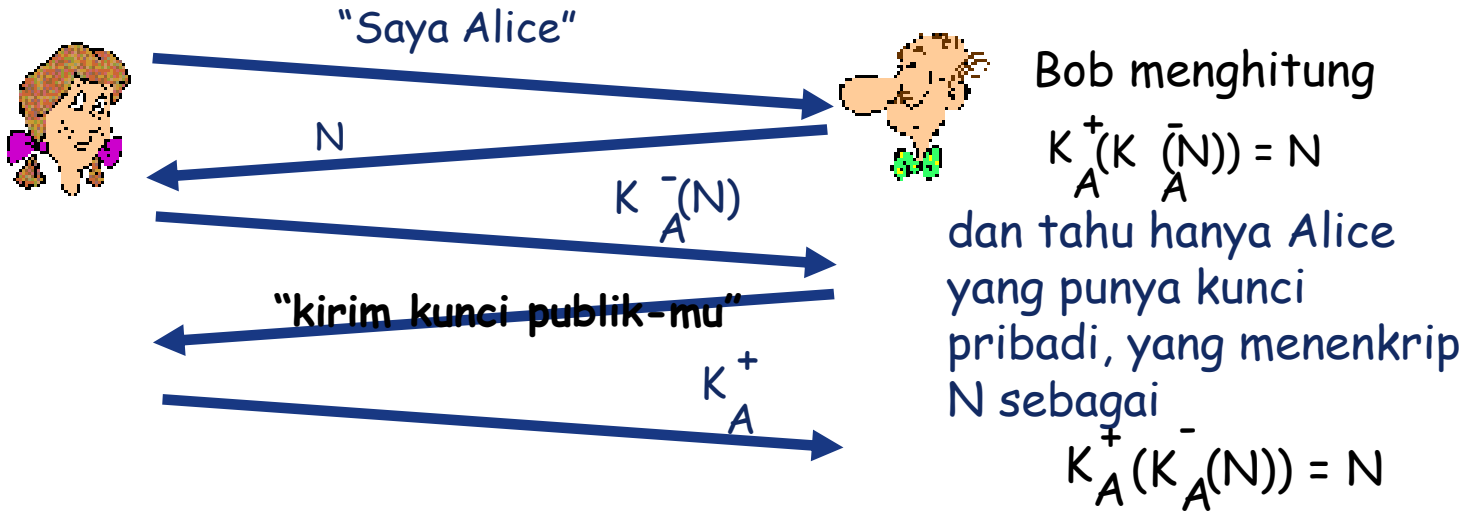
Authentication : Usaha Terakhir

- Tujuan : Menghindari serangan playback
- Nonce : nomor (timestamp) sekali pakai
- Protokol ap4.0 : Sebagai bukti Alice “ada”, Bob mengirim Alice nonce, N.
- Alice harus mengembalikan N ter-enkripsi kunci rahasia bersama



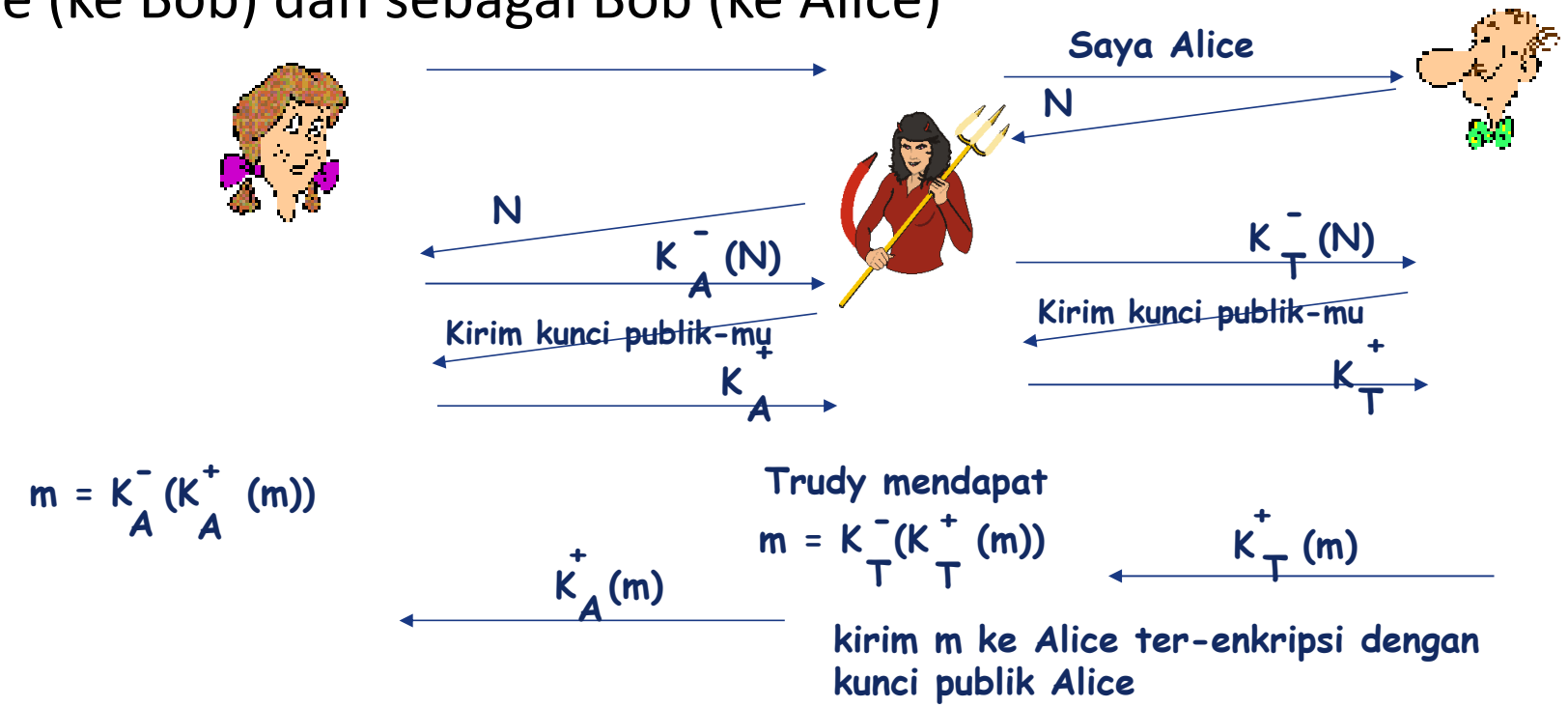
Upgrade Protokol!

- Protokol ap4.0 : Menggunakan kunci simetrik bersama
  - Bisa di autentikasi dengan teknik kunci publik?
- Protokol ap5.0 : Menggunakan Nonce, kriptografi kunci publik



Protokol ap5.0 : Lubang keamanan

- Serangan "Man (woman) in the middle" : Trudy berpura-pura sebagai Alice (ke Bob) dan sebagai Bob (ke Alice)



Protokol ap5.0 : Lubang keamanan

- Serangan "Man (woman) in the middle" : Trudy berpura-pura sebagai Alice (ke Bob) dan sebagai Bob (ke Alice)

Sukar dideteksi:

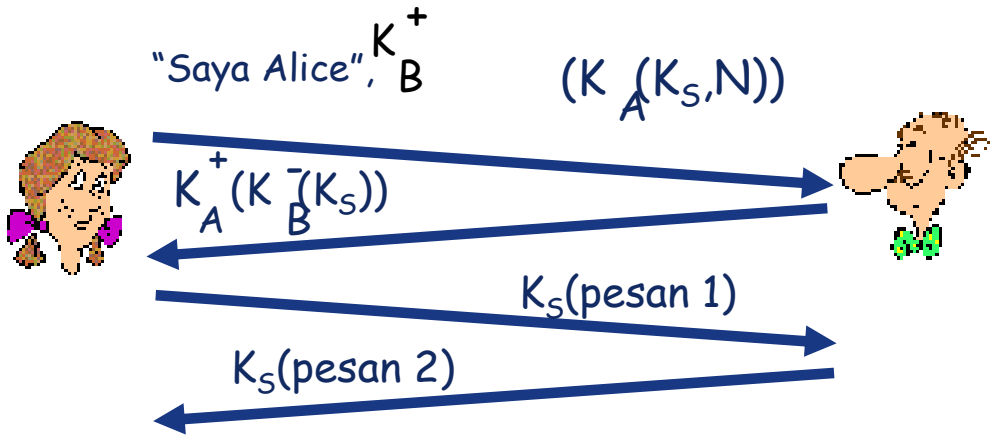
- Bob menerima apapun yang dikirim Alice dan sebaliknya.
- Masalahnya Trudy juga!
- Catatan: Masalahnya adalah distribusi kunci, jika saja Bob tahu atau dapat mendapatkan kunci publik Alice secara aman, tidak akan terjadi masalah!

Autentikasi kunci publik rahasia hybrid

- Idenya adalah menggunakan kunci publik untuk mengirimkan kunci sesi komunikasi
- Harus dipunyai autentikasi bersama untuk menjamin kunci sesi dapat dipercaya
- Dalam kasus ini diasumsikan bahwa Alice dan Bob sudah tahu kunci publik lawannya.

Autentikasi kunci publik rahasia hybrid

- Pada pesannya ke Bob, Alice menyatakan dirinya, dan mengirim pesan yang hanya Bob dapat mendekrip dengan kunci pribadinya.
  - Setelah mendekrip pesan luar, pesan dalam hanya dapat didekrip dengan kunci publik Alice, membuktikan dialah pengirimnya.
  - Mengandung kunci sesi baru K_s dan sebuah nonce untuk menjamin kebaruan.
- Bob mengirim balasan ke Alice yang hanya bisa didekrip dengan kunci pribadinya.
  - Setelah mendekrip pesan luar, pesan dalam hanya dapat didekrip dengan kunci publik Bob, membuktikan dialah pengirimnya.
  - Mengandung kunci sesi lagi. Jika cocok dengan kunci sesi yang dikirimkannya, maka keduanya sudah mengautentikasi dirinya sendiri dan bisa menggunakan kunci itu.



Vulnerability in Authentication (Password)

- Setiap password dapat ditebak
- Tingkat kekuatan suatu password ditentukan oleh berapa banyak tebakan harus dilakukan.

Vulnerability in Authentication (Password)

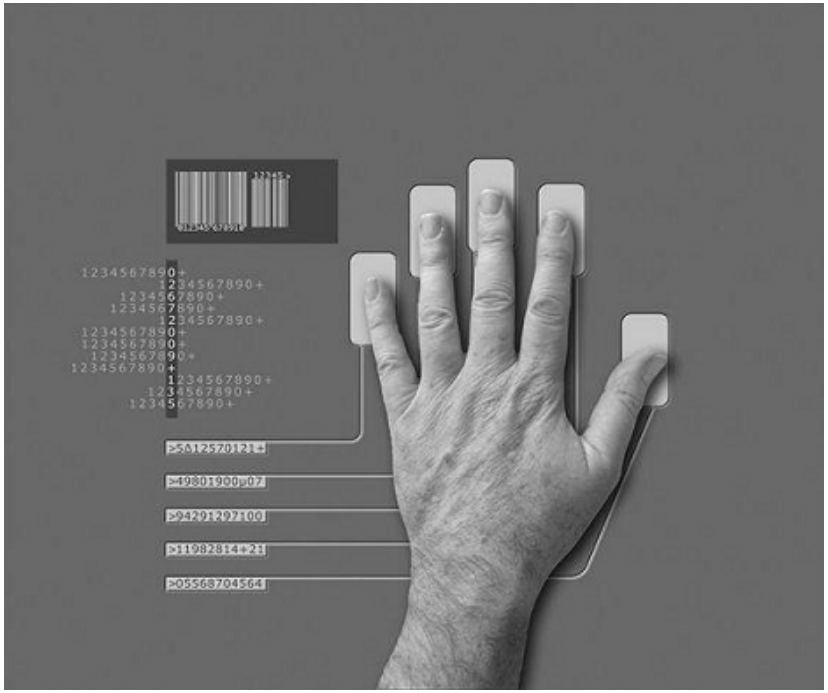
Untuk meningkatkan security password:

- Tidak hanya menggunakan karakter a-z
- Gunakan password yang panjang
- Hindari penggunaan nama atau kata yang umum
- Gunakan sesuatu yang mudah diingat
- Gunakan password yang bervariasi
- Ganti password secara berkala
- Jangan catat password
- Jangan beritahu orang lain

Vulnerability in Authentication (Biometrics)

Beberapa karakteristik fisik yang dapat dikenali adalah:

- Fingerprint
- Hand geometry (bentuk dan ukuran jari)
- Retina dan iris
- Voice, Face
- Facial features (bentuk hidung, jarak mata)
- Handwriting, signature, hand motion



Vulnerability in Authentication (Biometrics)

Permasalahan dari penggunaan biometri:

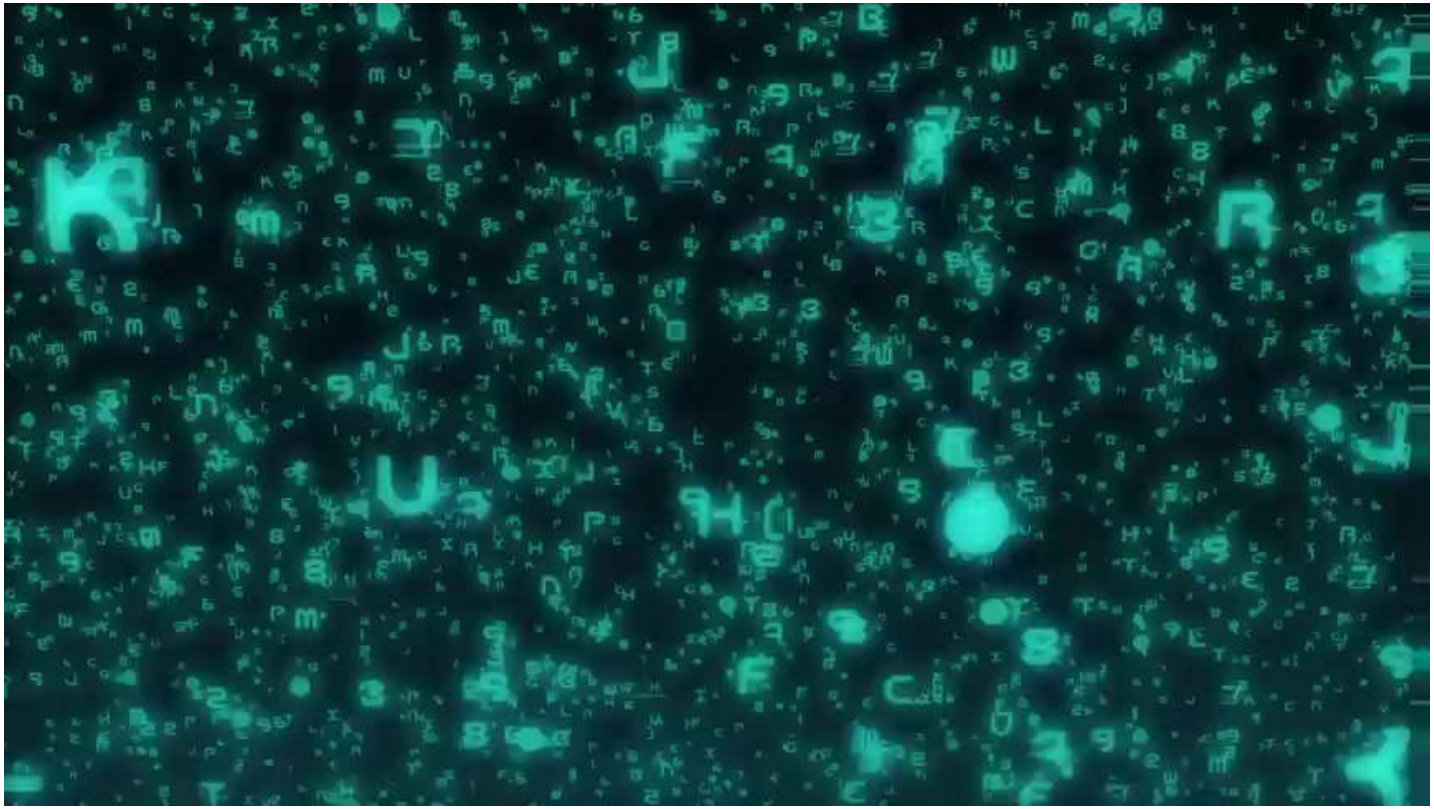
- Perangkat yang digunakan relative mahal
- Tingkat akurasi dari perangkat
- Kemungkinan terjadinya *biometric forgeries* (contoh: *gummy fingers*)

Vulnerability in Authentication (Tokens)

Salah satu serangan pada tokens, atau sesuatu yang dimiliki user, adalah *skimming* pada mesin ATM.

- Mendapatkan informasi dari magnetic card pada kartu
- Merekam pin yang dimasukkan pada mesin
- Membuat dummy card dengan data yang diperoleh

Create Fake Credit Card



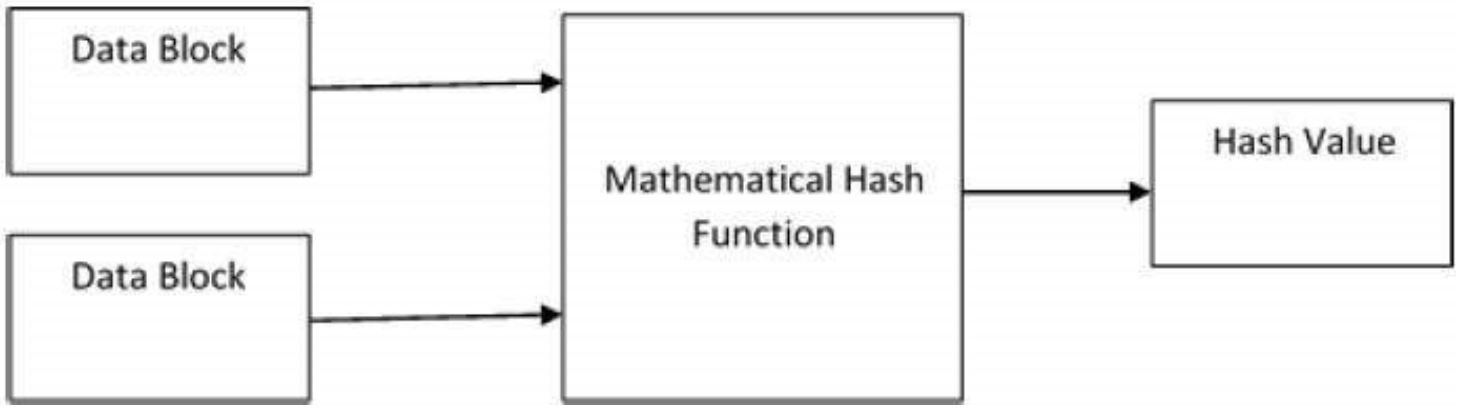
ATM Skimming

Hash Function

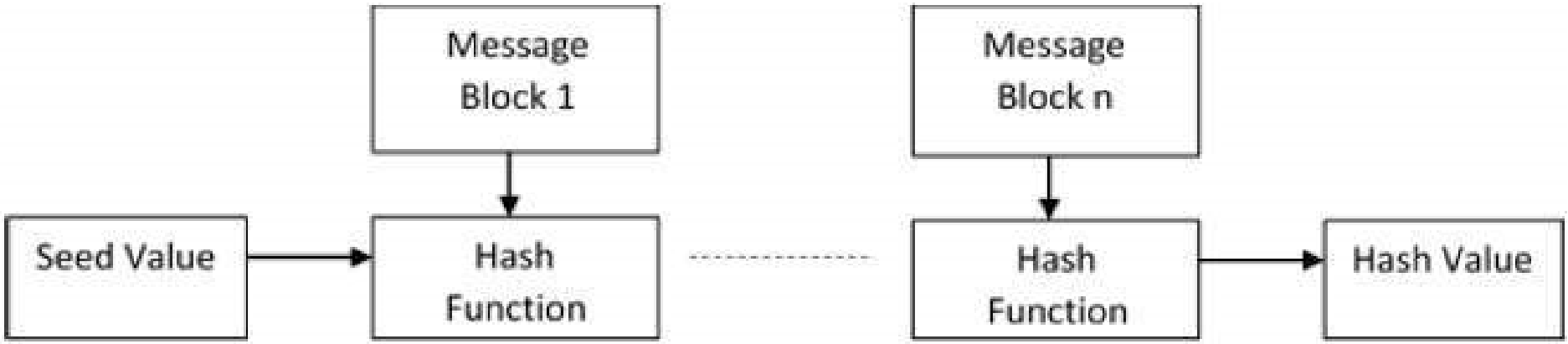
- Hash Function, merupakan fungsi satu arah
- $y = F(x)$
- Nilai yang dihasilkan oleh fungsi hash disebut *message digest* atau *hash values*.
- Fungsi hash harus memenuhi beberapa hal:
  - Sulit untuk dilakukan *reverse hash function* dari segi komputasi
  - Jika diberikan input yang berbeda, sulit untuk mendapatkan *hash value* yang sama

Hash Function

- Fungsi hash beroperasi pada dua block data untuk menghasilkan *hash code*.
- Ukuran block data bergantung pada algoritma yang digunakan.

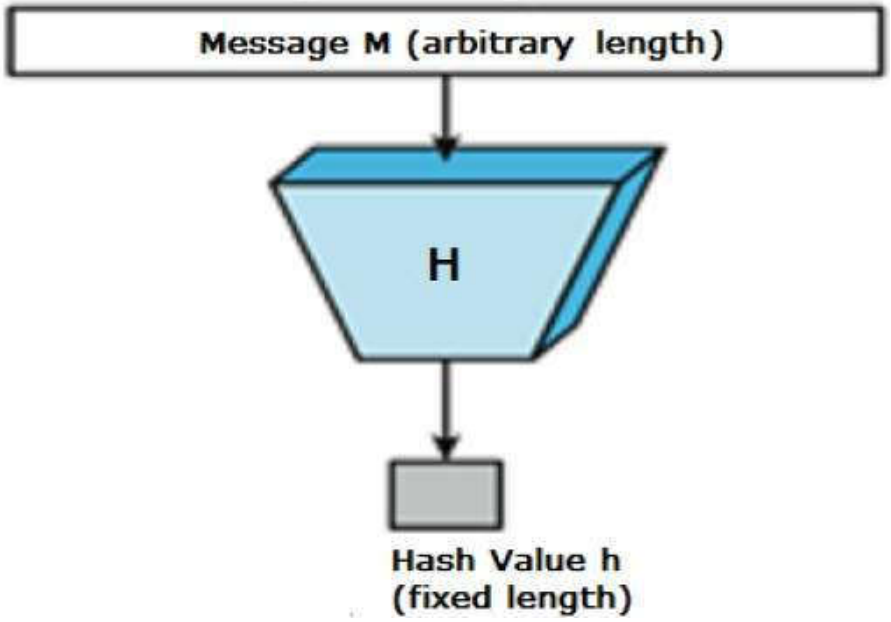


Hash Function

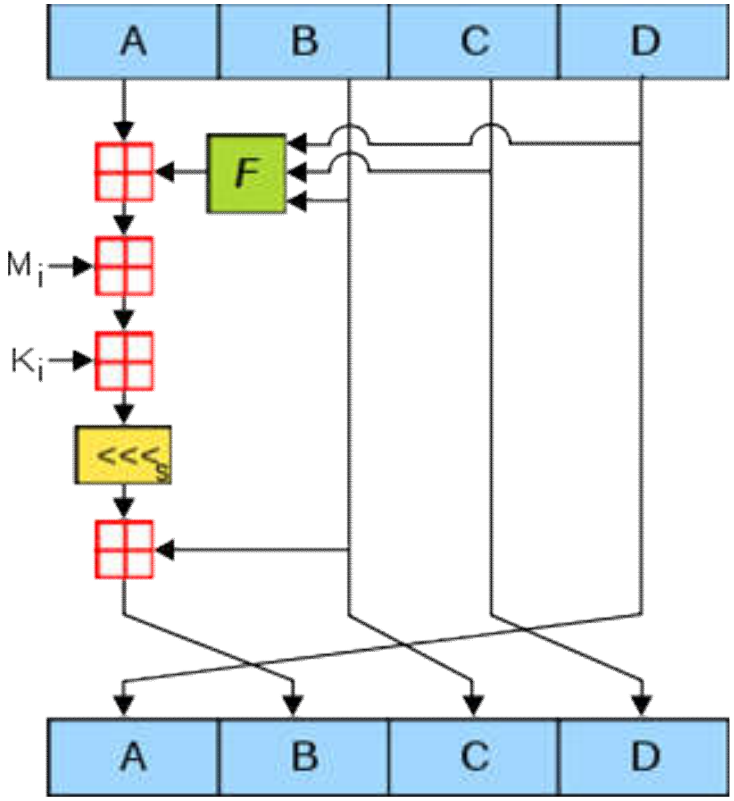


MD5

- MD5 dibuat oleh Ron Rivest, salah satu fungsi hash yang banyak digunakan.
- MD5 merupakan perbaikan MD4 setelah MD4 sukses dibobol
- MD5 menerima masukan berupa pesan dengan ukuran sembarang, dan menghasilkan *message digest* yang panjangnya 128 bit



MD5



Pembuatan MD5

1. Penambahan padding bits
2. Penambahan nilai panjang plain text
3. Inisialisasi buffer Message Digest
4. Pengolahan pesan dalam block berukuran 512 bit

Pembuatan MD5 (Langkah 1)

- Input dipecah kedalam block berukuran 512 bit
- Dilakukan padding agar panjangnya dapat dibagi oleh 512
- Proses padding:
  - Tambahkan bit ‘1’ pada bagian akhir pesan
  - Diikuti dengan bit ‘0’ sampai 64 bit kurang dari kelipatan 512
  - 448 mod 512

Pembuatan MD5 (Langkah 2)

- Sisa 64 bit terakhir (dari langkah sebelumnya) diisi dengan nilai integer yang merepresentasikan panjang data awal, dalam bit.
- Setelah langkah ini, data memiliki panjang dengan kelipatan 512 bit, atau setara dengan 16 (32-bit) words.

Pembuatan MD5 (Langkah 3)

- Menggunakan 4 buffer (A, B, C, D) masing-masing berukuran 32-bit.
- Inisialisasi keempat register (dengan nilai hexa, dimulai dari *low-order bytes*)

```
word A: 01 23 45 67
word B: 89 ab cd ef
word C: fe dc ba 98
word D: 76 54 32 10
```

Pembuatan MD5 (Langkah 4)

- Fungsi yang masing-masing menggunakan 3 (32-bit) input dan menghasilkan 1 (32-bit) output.

```
F(X,Y,Z) = XY v not(X) Z
G(X,Y,Z) = XZ v Y not(Z)
H(X,Y,Z) = X xor Y xor Z
I(X,Y,Z) = Y xor (X v not(Z))
```

MD5 hashes

- *Message digest* yang dihasilkan dituliskan dalam digit hexa.

```
MD5("The quick brown fox jumps over the lazy dog") =
9e107d9d372bb6826bd81d3542a419d6
```

```
MD5("The quick brown fox jumps over the lazy dog.") =
e4d909c290d0fb1ca068ffaddf22cbd0
```

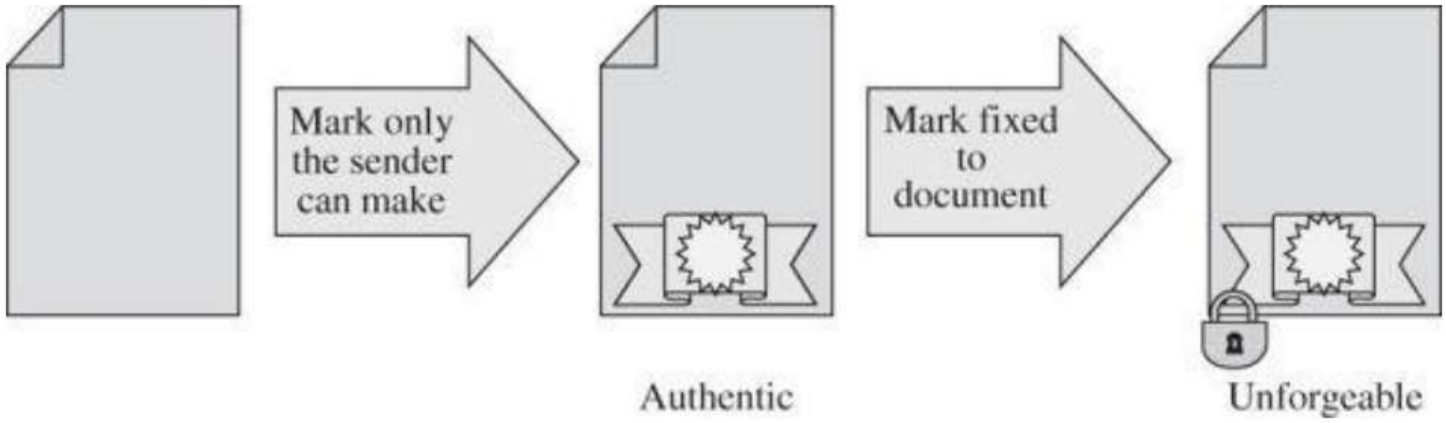
- Perubahan kecil pada pesan memberikan hasil yang berbeda, dikenal dengan *Avalanche effect*.

Digital Signature

- Mekanisme dimana proses autentikasi dilakukan dengan menambahkan bit pattern pada file.
- Menggunakan kriptografi asimetrik.

Digital Signature

- *Digital signature* harus memenuhi dua persyaratan berikut:
  - *Unforgeable*; Jika *S* menandatangani pesan *M* dengan *Sig(S,M)*, maka tidak ada orang lain yang dapat menghasilkan [*M,Sig(S,M)*].
  - *Authentic*; Jika *R* menerima [*M,Sig(S,M)*] yang diakui dari *S*, *R* dapat mengecek apakah *signature* tersebut benar dari *S*.



Digital Signature

- Persyaratan tambahan untuk *digital signature*:
  - Tidak dapat diubah; Setelah ditransmisikan, *M* tidak dapat diubah oleh *S*, *R*, ataupun pihak ketiga.
  - Tidak dapat digunakan kembali; Jika pesan sebelumnya dikirim kembali, akan terdeteksi oleh *R*.

Digital Signature

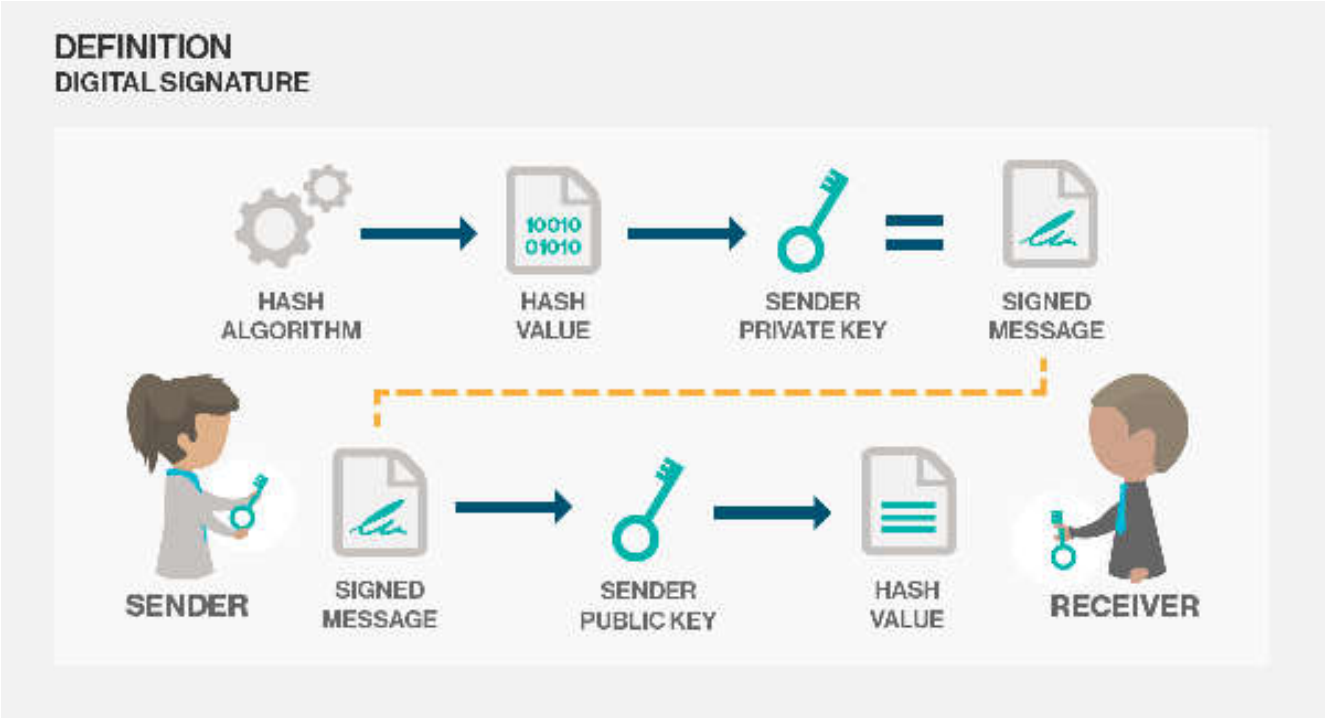
Hal yang dibutuhkan untuk membuat *digital signature*

- Kriptografi asimetrik
- Fungsi hash

Digital Certificate: dokumen elektronik yang memiliki *digital signature*.

Digital Signature Concept

1. Gunakan fungsi hash untuk menghitung *message digest* dari file.
2. Lakukan proses enkripsi pada *message digest* yang dihasilkan.



Digital Signature Verification

Proses verifikasi dilakukan dengan langkah berikut

1. Hitung *current hash value*

   Menghitung *hash value* dari pesan yang diterima

2. Hitung *original hash value*

   Melakukan proses dekripsi untuk mendapatkan nilai *hash value* sebelum dienkripsi

3. Bandingkan kedua nilai *hash value*

   Jika kedua nilai identik, proses verifikasi berhasil. Jika tidak, *invalid digital signature*

Invalid Signatures

Kemungkinan penyebab *invalid digital signatures*:

- Jika *digital signature* diubah dan dilakukan proses dekripsi, *original hash value* yang diperoleh bukan nilai dari pesan asli.
- Jika pesan diubah setelah dilakukan proses *signing*, *current hash value* yang dihitung akan berbeda dengan *original hash value*.
- Jika kunci public tidak berkorespondensi dengan kunci privat yang digunakan pada proses *signing*, *original hash value* yang diperoleh setelah proses dekripsi bukan *original hash value* yang sebenarnya.

Quantum Cryptography?

TUGAS (Tulis tangan, di kertas A4)

Buatlah sebuah tulisan yang menjawab pertanyaan dibawah ini.

- Apa itu quantum cryptography?
- Apa yang melatarbelakangi kemunculannya?
- Jika ada algoritma yang menggunakan quantum cryptography, jelaskan cara kerja algoritma tersebut.