

Teknik-Teknik Distribusi *Confidential Keys*

Key Management Technique

Agung Nursilo | Ricky Aji Pratama
www.noersilo.web.id | noersilo@yahoo.com

A. Key layering and Cryptoperiod

Berbagai macam teknik dan protocol yang ada untuk mendistribusikan kunci rahasia harus tetap terjaga (baik itu *private key* maupun *symmetric key*). Dalam tulisan ini kami akan membahas dua dari sekian banyak teknik yang ada yaitu *key layering* dan *symmetric-key certificates*.

↓ Cryptographic objective (usage)	Algorithm type	
	public-key	symmetric-key
confidentiality†	encryption	encryption
data origin authentication‡	signature	MAC
key agreement	Diffie-Hellman	various methods
entity authentication (by challenge-response protocols)	1. signature 2. decryption 3. customized	1. MAC 2. encryption

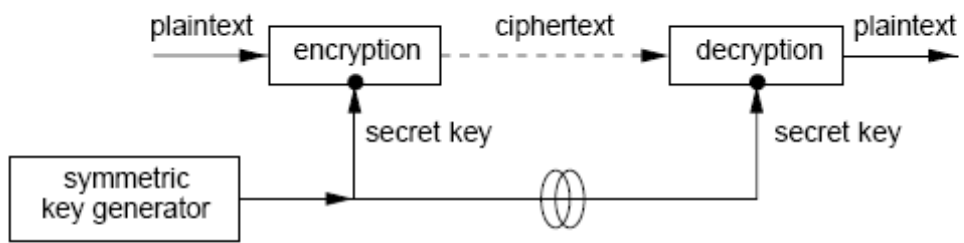
Table 13.2: *Types of algorithms commonly used to meet specified objectives.*

Pada tabel 13.2 di atas menjelaskan mengenai pengelompokkan kunci menurut kegunaannya. Pada class *confidentiality* bisa diklasifikasikan lagi menurut informasi asli yang sedang dilindungi, meliputi data user dan material kunci. Hal ini menyarankan suatu *natural key layering* sebagai berikut :

1. *Master keys*

Master keys merupakan kunci yang memiliki level hierarki tertinggi yang tidak diproteksi secara kriptografis. Kunci ini didistribusikan secara manual atau pada awalnya diinstal dan diproteksi oleh prosedur kontrol dan fisik atau pengisolasian elektronik. Dengan kata lain *master key* biasanya ditanamkan dalam mesin atau alat.

(a) Symmetric-key encryption



(b) Public-key encryption

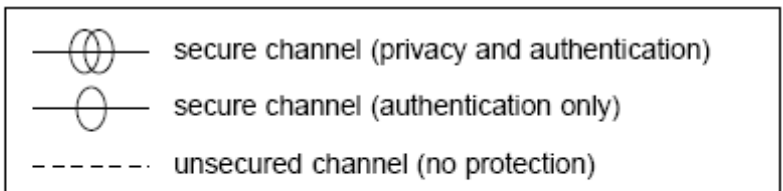
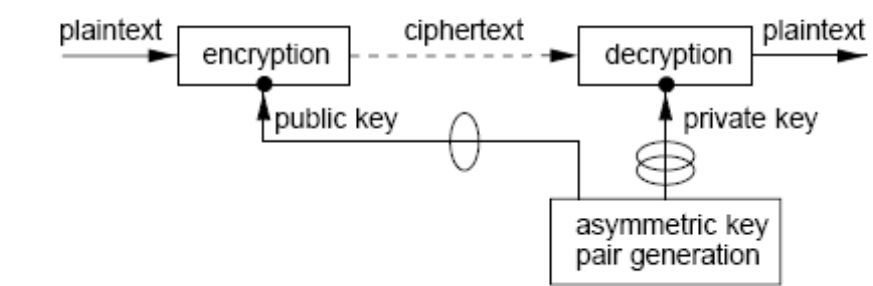


Figure 13.4: Key management: symmetric-key vs. public-key encryption.

2. Key encrypting keys

Key encrypting keys atau dapat disebut sebagai *symmetric keys* maupun enkripsi *public keys*, digunakan pada *key transport* atau penyimpanan kunci-kunci lainnya. Sehingga dapat disebut sebagai *key transport keys*, dan diamankan oleh kunci lainnya.

3. Data keys

Data keys biasanya digunakan untuk menyediakan operasi-operasi kriptografi pada data user (seperti enkripsi dan otentikasi). *Data keys* secara umum merupakan *short-term symmetric keys*, selain itu juga *asymmetric signature private keys* dapat disebut sebagai *data keys* yang biasanya berupa *longer-term keys*.

Kunci pada suatu layer digunakan untuk melindungi kunci lain yang ada di layer bawahnya. Hal ini bertujuan untuk membuat *attacks* lebih sulit dan membatasi pembukaan atau recovery karena kebocoran suatu kunci.

Kebocoran sebuah *key encrypting key* (apalagi *master key*) akan berdampak pada semua kunci yang dilindunginya (di bawahnya). Langkah-langkah khusus digunakan untuk melindungi master key, termasuk dengan pembatasan akses yang digunakan, proteksi hardware, dan menyediakan akses kunci hanya dalam pengawasan bersama.

Cryptoperiod merupakan kunci yang pada periode waktu tertentu masih valid dipakai dalam kegiatan kriptografi oleh pihak yang berwenang.

Cryptoperiod digunakan untuk :

1. Membatasi jumlah informasi dr kriptanalisis;
2. Membatasi besarnya kebocoran jika kunci diketahui pihak lain;
3. Membatasi penggunaan algoritma tertentu sampai dengan perkiraan batas waktu efektif penggunaannya;
4. Membatasi waktu usaha untuk menembus mekanisme akses fisik, prosedur dan logika yang melindungi kunci dr pihak yg tidak berwenang;
5. Membatasi periode dimana informasi dpt diketahui secara tidak sengaja dari material;
6. Membatasi waktu untuk dapat dilakukannya kriptanalisis secara intensif.

Ada pun faktor-faktor yang mempengaruhi cryptoperiod adalah :

1. *Risk Factor*

Periode kunci kripto yang pendek akan memperkuat keamanan

- Kekuatan dr mekanisme kriptografis, misalnya algoritma, panjang kunci, ukuran blok dan mode operasi;
- Bentuk fisik dr mekanisme, misalnya implementasi software pada PC;
- Lingkungan dr kegiatan operasi, misalnya pengamanan akses fasilitas terbatas, kantor umum, atau akses terminal umum;

- Volume informasi atau jumlah transaksi berita;
- Umur keamanan data;
- Fungsi keamanan, misalnya enkripsi data, digital signature, produksi kunci, proteksi kunci;
- Metode *re-keying*;
- *Up-date* kunci atau proses penurunan kunci;
- Jumlah node pd jaringan yg berbagi kunci yg sama;
- Jumlah salinan kunci dan distribusi salinan tsb;
- Ancaman terhadap informasi, misalnya terhadap siapa informasi perlu diamankan.

2. *Consequence Factor*

Semakin besar dampak kebocoran, maka periode kunci harus semakin pendek

- Sensitifitas informasi : berdampak pada jangka waktu dr informasi itu sendiri;
- Pentingnya proses yang dilindungi oleh kriptografi;
- Biaya untuk recovery bila informasi atau proses berhasil diketahui.

3. Dan faktor yang lainnya

- **Komunikasi vs penyimpanan;**

Kunci untuk mengamankan kerahasiaan selama komunikasi berlangsung yang memiliki cryptoperiod yang lebih pendek dibandingkan dengan kunci untuk mengamankan data yang disimpan, karena biaya untuk mengenkripsi ulang data tersebut dengan kunci yang berbeda akan memberatkan

- **Biaya dari revokasi kunci dan penempatannya kembali.**

Pada beberapa kasus, biaya yang dibutuhkan untuk penggantian kunci akan sangat besar. Contohnya adalah mengganti kunci yang digunakan untuk mengenkripsi dan dekripsi database yang terdistribusi.

Selain lapisan kunci hirarki di atas, kunci dapat diklasifikasikan didasarkan pada waktu pertimbangan sebagai berikut

1. *Long-term keys*

Termasuk didalamnya adalah *master key*, *key-encrypting key* yang sering digunakan pada *key agreement*

2. *Short-term key*

Termasuk didalamnya adalah *key established* dengan *key transport* atau *key agreement* dan sering digunakan pada *data key* atau *session key* untuk komunikasi tunggal.

Secara umum, aplikasi komunikasi melibatkan short long key, sementara penyimpanan data aplikasi memerlukan long term key. Long term key biasanya melindungi short term key. Diffie-Hellman keys adalah pengecualian dalam beberapa kasus . Cryptoperiods membatasi penggunaan kunci untuk periode tetap, setelah itu mereka harus menggantinya.

Short-term use vs. Protection

Istilah short term pada short term key mengacu pada waktu dari penggunaan kunci oleh pihak yang berwenang, bukan perlindungan seumur hidup. Sebagai contoh, sebuah kunci enkripsi yang digunakan hanya untuk sekali pakai saja mungkin tetap diminta untuk memberikan perlindungan yang cukup untuk menahan serangan jangka panjang (sekitar 20 tahun), sedangkan jika tanda tangan yang diverifikasi segera dan tidak pernah diperiksa kembali, tanda tangan kunci mungkin perlu untuk memberikan perlindungan hanya untuk waktu yang relatif singkat waktu. Lebih parah konsekuensi dari sebuah kunci rahasia yang diungkapkan, semakin besar kemampuan musuh untuk mendapatkan akses ke sana, dan semakin besar waktu atau usaha musuh untuk menerobos kesana.

B. Key Translation Center dan Symetric-key Certificates

Pada bagian ini kita akan membahas mengenai teknik-teknik yang termasuk kedalam *key translation center*, dan termasuk didalamnya mengenai penggunaan *symmetric-key certificates*.

1. *Key Translation Center*

Key translation center (KTC) merupakan suatu server terpercaya yang mengijinkan dua pihak A dan B, dimana kedua pihak tersebut tidak berhubungan (*sharing*) material

kunci secara langsung, untuk membangun suatu komunikasi yang aman selama kunci K_{AT} dan K_{BT} dipertukarkan melalui T. A bisa saja mengirim suatu pesan rahasia M kepada B menggunakan suatu protokol seperti yang diberikan dibawah ini :

Protocol Message translation protocol menggunakan KTC

SUMMARY : A berhubungan dengan sebuah server (KTC) T dan pihak B.

RESULT : A mengirimkan pesan M (atau bisa juga *session key*) ke B.

1. *Notation.* E merupakan algoritma enkripsi simetrik. M bisa juga sebagai *session key* dari K.
2. *One-time setup.* A dan T men-*share* kunci K_{AT} . Begitu pun halnya B dan T men-*share* K_{BT} .
3. *Protocol message.*

$$A \rightarrow T : A, E_{K_{AT}}(B, M) \quad (1)$$

$$A \leftarrow T : E_{K_{BT}}(M, A) \quad (2)$$

$$A \rightarrow B : E_{K_{BT}}(M, A) \quad (3)$$

4. *Protocol action.*

- (a) A mengenkripsi M dengan menggunakan K_{AT} , dan mengirimkannya ke T.
- (b) Setelah mendekripsi pesannya, B mengenkripsi lagi pesan tersebut menggunakan K_{BT} .
- (c) T kemudian mengirimkan lagi pesan tersebut ke A untuk nantinya A akan mengirimkannya ke B.

Perlu dicatat bahwa hanya salah satu pihak saja yang perlu berkomunikasi dengan T, apakah itu A atau B. Selain protokol di atas, bisa saja A mengirimkan M langsung ke B, nantinya B akan meminta T untuk membuka pesan M tersebut dan T mengirimkannya lagi ke B. Berikut ini keamanan pada protokol *Message translation protocol* di atas :

- Pihak A, yang berhubungan dengan kunci enkripsi pesan (1), termasuk dalam pesan (2) sebagai suatu *secure indication* (kepada B) atas sumber aslinya. Kunci yang sah menawarkan metode yang lebih kuat untuk mencegah substitusi kunci.
- Sebuah perbedaan antara format pesan (1) dan (2) yang dikehendaki untuk mencegah seorang musuh dari pengiriman kembali ke A sebagai sebuah pesan (3) yang memang asli dari B.
- Dimungkinkan terjadinya *message replay*. Karena dalam protokol di atas tidak

terdapat aspek otentikasi (user).

- Suatu integritas memeriksa mekanisme dalam pesan terenkripsi harus digunakan untuk mengizinkan T mendeteksi perusakan (*tampering*) atas pesan asli $A(1)$, begitu pun dengan (2) dan (3).
- Serangan *chosen-text* terhadap kunci K_{BT} pada (2) dapat dicegah dengan cara enkripsi mode CBC dan menyisipkan suatu inisial *field* yang berisi bilangan acak.

2. *Symmetric-key Certificates*

Sekarang dalam *symmetric-key certificates* $E_{KT}(K_{BT}, B)$ dienkripsi menggunakan sebuah *symmetric master key* K_T yang hanya diketahui oleh T . Suatu parameter *lifetime* L juga dapat disertakan dalam *certificate* tersebut sebagai tanda validasi. Sebuah *certificate* melayani layaknya sebuah memo dari T ke dirinya sendiri, dan diberikan kepada B sehingga B kemudian dapat mengembalikannya lagi ke T ketika perlu mengakses kunci simetrik B K_{BT} untuk mentranslasi pesan. Ketimbang T menyimpan semua *user-keys*, kini T hanya menyimpan K_T secara aman. *Symmetric-key certificates* dapat diterapkan pada protokol *Message translation protocol* di atas dengan hanya mengganti pesan pertamanya saja seperti di bawah ini :

$$A \rightarrow T : SCert_A, E_{K_{AT}}(B, M), SCert_B \quad (1)$$

Sebuah database publik dapat dibuat dengan sebuah masukan nama suatu user dan sertifikat kunci simetriknya. Untuk membuat pesan (1), A mengambil sertifikat kunci simetrik B dan punya A sendiri. T mengantarkan terjamahannya, mengambil K_{AT} dan K_{BT} dari sertifikat-sertifikatnya, namun juga sekalian memeriksa apakah A merupakan pihak yang dimaksud B .

C. Kriptoperiod untuk kunci Asimetrik

Untuk pasangan kunci, masing-masing pasangan kunci memiliki periode kunci (*cryptoperiod*). Artinya, setiap kunci yang digunakan oleh sebuah "originator" untuk menerapkan perlindungan kriptografi (misalnya, membuat tanda tangan digital) atau oleh "penerima" untuk kemudian

memproses informasi yang dilindungi (misalnya, verifikasi tanda tangan digital), tetapi tidak keduanya. Di mana kunci publik didistribusikan pada sertifikat kunci publik, periode kunci untuk setiap pasangan kunci tidak selalu sama dengan masa berlaku sertifikat.

Contoh masalah-masalah yang terkait dengan cryptoperiod publik kriptografi kunci meliputi:

1. Cryptoperiod dari *private key transport key* mungkin lebih lama daripada periode kunci dari *private key transport key*. Kunci publik digunakan untuk periode waktu yang tetap untuk mengenkripsi material kunci. Jangka waktu yang dapat diindikasikan oleh tanggal kadaluarsa pada sertifikat kunci publik. Kunci pribadi perlu dipertahankan selama membutuhkan recover (dekripsi) kunci yang dienkripsi oleh kunci publik (biasanya setelah kunci publik telah hancur).
2. Sebaliknya, periode kunci dari *private authentication key* yang digunakan untuk menandatangani informasi yang dikirim ini pada dasarnya sama dengan cryptoperiod dari kunci publik yang terkait (misalnya, otentikasi kunci publik). Yaitu, ketika kunci pribadi tidak akan digunakan untuk menandatangani pengiriman, kunci publik tidak lagi diperlukan.
3. Jika *private signature key* ini akan digunakan untuk menghasilkan tanda tangan digital sebagai bukti keaslian, cryptoperiod dari kunci private dapat secara signifikan lebih pendek daripada yang cryptoperiod dari kunci verifikasi tanda tangan publik. Dalam kasus ini, kunci private biasanya digunakan untuk periode waktu tertentu, setelah waktunya tiba, pemilik kunci akan menghancurkan kunci pribadi. Kunci publik mungkin tersedia untuk jangka waktu lebih lama untuk memverifikasi tanda tangan. Namun, faktor-faktor lain seperti kekuatan algoritma penandatanganan, nilai dari tanda tangan, dan kemungkinan pemalsuan harus dipertimbangkan.

D. Symmetric Key Usage Periods Cryptoperiods

Untuk kunci simetris, kunci tunggal digunakan untuk menerapkan kedua perlindungan (misalnya, mengenkripsi atau komputasi MAC) dan pengolahan informasi yang dilindungi (misalnya, decrypting yang dienkripsi informasi atau memverifikasi MAC). Periode waktu di mana

perlindungan kriptografi dapat diterapkan pada data disebut juga *originator usage period* dan periode waktu selama informasi yang dilindungi sedang diproses disebut *recipient usage period*. Sebuah kunci simetris tidak dapat digunakan untuk memberikan perlindungan setelah berakhirnya jangka waktu penggunaan originator. *Recipient usage period* dapat melampaui *originator usage period*. Hal ini memungkinkan semua informasi yang telah dilindungi oleh originator untuk diproses oleh penerima sebelum pengolahan kunci harus dinonaktifkan. Namun dalam banyak kasus, *originator* dan *recipient usage period* adalah sama. Cryptoperiod dari kunci simetris adalah periode waktu dari dimulainya *originator usage period* hingga berakhirnya *recipient usage period*.

Perhatikan bahwa dalam beberapa kasus yang telah ditentukan cryptoperiods mungkin tidak memadai untuk lingkungan keamanan data yang dilindungi. Jika lingkungan keamanan yang diperlukan melebihi cryptoperiod, maka perlindungan akan perlu diterapkan kembali menggunakan kunci baru.

Contoh penggunaan *usage period* meliputi:

- a. Ketika sebuah kunci simetris hanya digunakan untuk mengamankan komunikasi, periode waktu dari aplikasi originator dari perlindungan kepada proses penerima dapat diabaikan.
- b. Ketika sebuah kunci simetris digunakan untuk melindungi informasi yang tersimpan, *originator usage period* mungkin berakhir lebih awal dari jangka waktu penggunaan penerima (ketika memori informasi diproses)
- c. Ketika sebuah kunci simetris digunakan untuk melindungi informasi yang tersimpan, jangka waktu penggunaan penerima dapat mulai setelah awal masa penggunaan originator. Untuk Misalnya, informasi dapat dienkripsi sebelum disimpan dalam *compact disk*. Pada beberapa kemudian waktu, kunci boleh didistribusikan dalam rangka untuk mendekripsi dan memulihkan informasi.

E. Rekomendasi untuk Cryptoperiod Jenis Spesifik Key

Yang disyaratkan cryptoperiod dalam pemberian kunci mungkin dapat dipengaruhi oleh jenis kunci sebanyak oleh lingkungan pengguna dan karakteristik data yang dijelaskan di atas. Cryptoperiod pada umumnya merekomendasi untuk berbagai jenis kunci yang disarankan di bawah

ini. Faktor-faktor yang dijelaskan sebelumnya, seharusnya digunakan untuk menentukan criptoperiod sebenarnya untuk keperluan khusus pada lingkungan cryptoperiods,

F. Rekomendasi untuk Bahan Keying Lainnya

Material kunci lainnya tidak memiliki pembangun cryptoperiod yang baik. Berikut rekomendasi yang ditawarkan mengenai disposisi material kunci lainnya:

1. Domain parameter tetap berlaku sampai diubah.
2. IV dikaitkan dengan informasi yang membantu untuk melindungi, dan diperlukan sampai informasi dan perlindungannya tidak lagi diperlukan.
3. Berbagi rahasia akan hancur segera setelah mereka tidak lagi diperlukan untuk memperoleh keying Materi.
4. Ping bibit harus dihancurkan segera setelah digunakan.
5. Informasi publik lainnya tidak boleh disimpan lebih lama dari yang diperlukan untuk kriptografi pengolahan.
6. Hasil Intermediate dibinasakan segera setelah digunakan.

Source : [Handbook of Applied Cryptography](#)