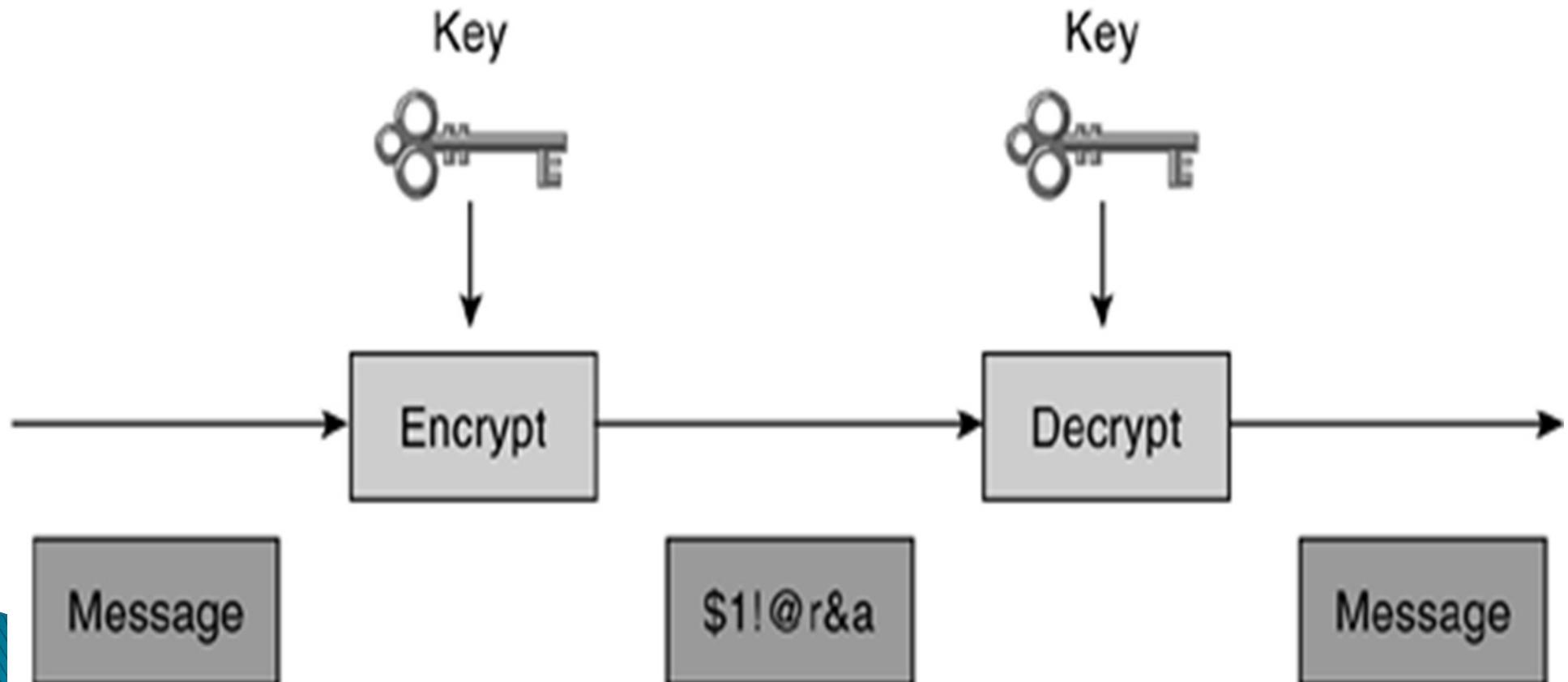


# Kriptografi

- Algoritma Kunci Simetris
- Algoritma Kunci Asimetris

# Algoritma Kunci Simetris

- Menggunakan kunci yang sama untuk mengenkripsi dan mendekripsi pesan.



# Symmetric Key Algorithms

- ▶ Ada dua teknik dalam kriptografi enkripsi simetris: stream cipher dan block cipher. Stream cipher mengenkripsi satu per satu bit pesan dalam satu waktu, sedangkan block cipher mengambil sejumlah bit dan mengenkripsi mereka sebagai satu unit.

- Algoritma kunci simetris umumnya lebih cepat untuk dijalankan daripada algoritma kunci asimetris.
- Kerugian untuk algoritma kunci simetris adalah persyaratan menggunakan kunci rahasia bersama. Kunci rahasia harus dipertukarkan antar pihak melalui saluran yang aman sebelum enkripsi dapat terjadi.

# Data Encryption Standard

- DES dikembangkan IBM pada tahun 1975, dan telah bertahan sangat baik terhadap kriptoanalisis selama bertahun-tahun.
- DES adalah algoritma enkripsi simetris dengan panjang kunci yang tetap 56 bit.
- Algoritma ini masih bagus, tapi karena panjang kunci pendek, maka rentan terhadap serangan brute force yang memiliki sumber daya yang cukup.

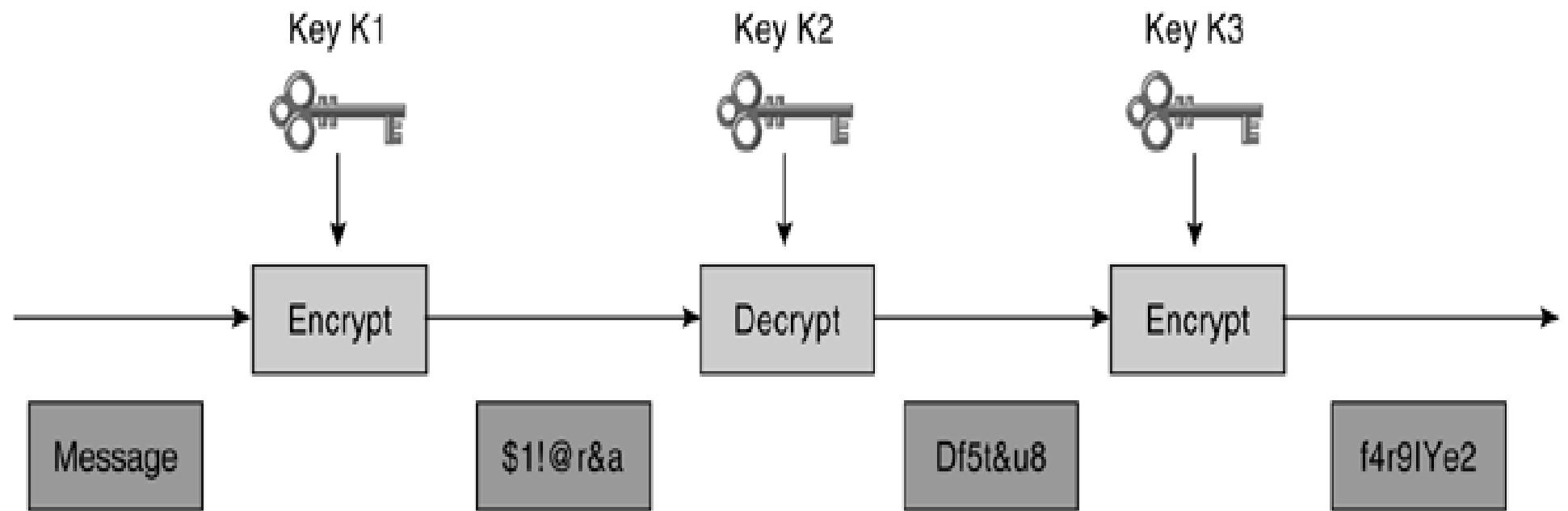
# Data Encryption Standard

- DES biasanya beroperasi dalam modus blok, di mana mengenkripsi data dalam blok 64-bit. Algoritma dan kunci yang sama digunakan untuk enkripsi dan dekripsi.
- Karena DES didasarkan pada fungsi-fungsi matematika sederhana, maka dapat dengan mudah diimplementasikan dalam hardware.

# Triple Data Encryption Standard

- Salah satu cara efektif untuk meningkatkan panjang kunci DES tanpa mengubah algoritma itu sendiri adalah dengan menggunakan algoritma yang sama dengan kunci yang berbeda beberapa kali berturut-turut.
- Menerapkan teknik DES tiga kali berturut-turut ke blok teks biasa disebut Triple DES (3DES).

# Triple Data Encryption Standard



# Triple Data Encryption Standard

- Ketika sebuah pesan yang akan dienkripsi dengan 3DES, sebuah metode yang disebut EDE (Encrypt Decrypt Encrypt) digunakan. EDE metode yang dijelaskan dalam daftar berikut:
  1. Pesan dienkripsi dengan 56-bit kunci pertama, K1.
  2. Data didekripsi dengan kunci kedua 56-bit, K2.
  3. Data dienkripsi lagi dengan kunci ketiga 56-bit, K3.

# Triple Data Encryption Standard

- Prosedur EDE menyediakan enkripsi dengan panjang kunci yang efektif 168 bit. Jika kunci K1 dan K3 adalah sama (seperti dalam beberapa implementasi), enkripsi yang kurang aman dengan panjang kunci 112 bit tercapai.

# Triple Data Encryption Standard

- Untuk mendekripsi pesan, harus menggunakan prosedur sebagai berikut, yang merupakan kebalikan dari metode Ede:
  1. Mendekripsi ciphertext dengan kunci K3
  2. Mengenkripsi data dengan kunci K2
  3. Akhirnya, mendekripsi data dengan kunci K1

# Triple Data Encryption Standard

- Mengenkripsi data tiga kali dengan tiga kunci yang berbeda tidak secara signifikan meningkatkan keamanan.
- Metode EDE harus digunakan. Mengenkripsi tiga kali berturut-turut dengan 56-bit yang berbeda sama dengan panjang kunci yang efektif 58-bit dan bukan 128-bit, seperti yang diharapkan.

# AES

- Pada tanggal 2 Oktober 2000, The US National Institute of Standards and Technology (NIST) mengumumkan pemilihan cipher Rijndael sebagai algoritma AES. Cipher ini, dikembangkan oleh Joan Daemen dan Vincent Rijmen, memiliki panjang blok dan kunci yang variabel.

# AES

- Algoritma ini menetapkan cara menggunakan kunci dengan panjang 128, 192, atau 256 bit untuk mengenkripsi blok dengan panjang 128, 192, atau 256 bit.
- Kedua blok dan panjang kunci dapat diperluas dengan mudah untuk kelipatan dari 32 bit.

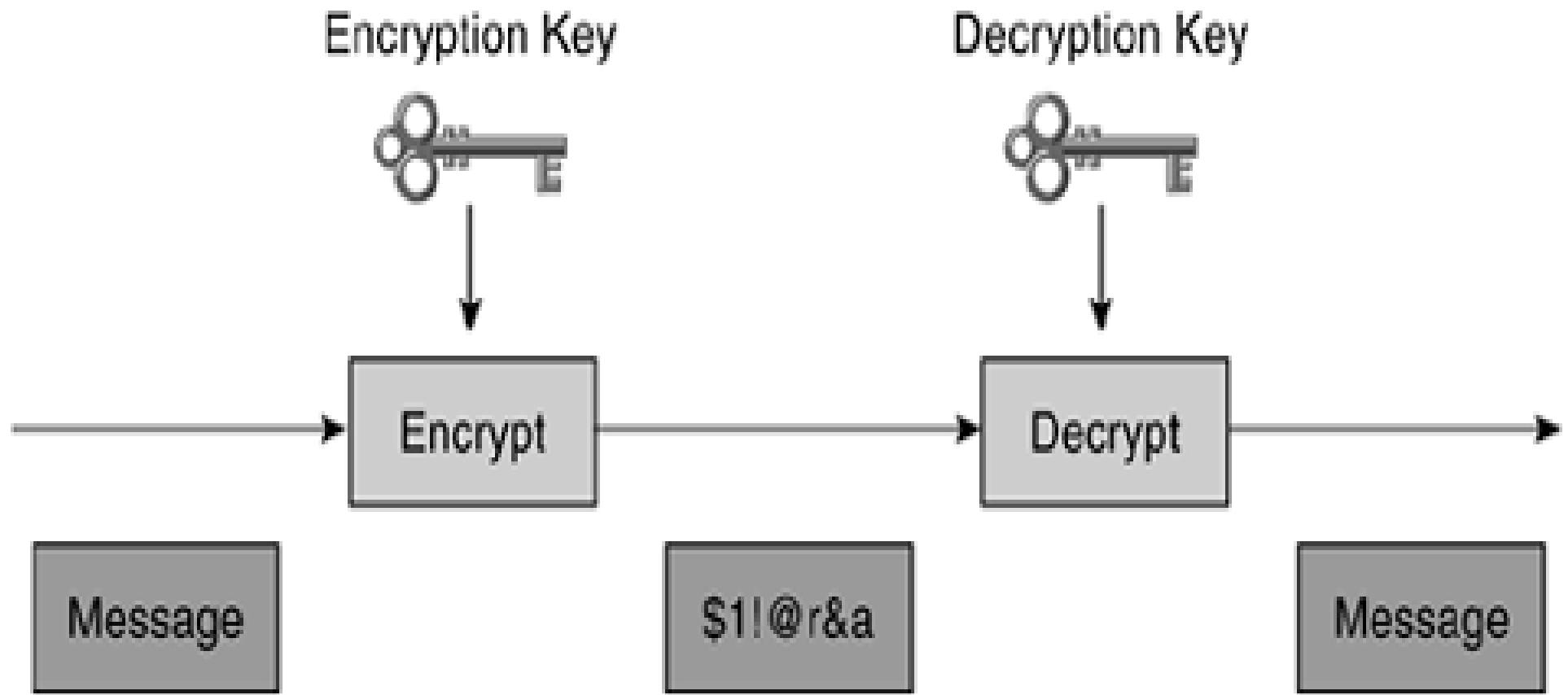
# AES

- AES dipilih untuk menggantikan DES dan 3DES karena mereka terlalu lemah (DES, dalam hal panjang kunci) atau terlalu lambat (3DES). AES lebih efisien dan lebih cepat, biasanya 5 kali dibandingkan dengan DES pada hardware yang sama. AES juga lebih cocok untuk throughput yang tinggi, terutama jika digunakan perangkat lunak enkripsi keseluruhan.

# Asymmetric Key Algorithms

- Sebuah algoritma kunci asimetris menggunakan sepasang kunci kriptografi yang berbeda untuk mengenkripsi dan mendekripsi teks biasa.
- Dua kunci yang berhubungan secara matematis. Sebuah pesan dienkripsi dengan algoritma menggunakan salah satu kunci, dan didekripsi oleh algoritma yang sama dengan kunci yang lain.

# Asymmetric Key Algorithms



# Asymmetric Key Algorithms

- Algoritma asimetris dirancang sedemikian rupa sehingga kunci untuk enkripsi berbeda dengan kunci untuk dekripsi.
- Kunci untuk dekripsi tidak dapat dihitung dari kunci enkripsi (setidaknya tidak dalam jumlah waktu yang wajar) dan sebaliknya.
- Biasa panjang kunci untuk algoritma asimetris berkisar 512-2.048 bit.

# Asymmetric Key Algorithms

- Algoritma asimetris relatif lambat (hingga 1000 kali lebih lambat dari algoritma simetris).
- Desain ini didasarkan pada masalah-masalah komputasi seperti pemfaktoran angka yang sangat besar, perhitungan logaritma diskrit pada angka yang sangat besar.

# Diffie–Hellman

- Whitfield Diffie dan Martin Hellman mengembangkan algoritma Diffie–Hellman pada tahun 1976.
- Keamanan yang berasal dari sulitnya menghitung logaritma diskrit dari angka besar.
- Protokol yang memungkinkan dua pengguna untuk bertukar kunci rahasia melalui media yang tidak aman tanpa ada rahasia.

# Diffie-Hellman

- Protokol memiliki dua parameter sistem, p dan g. Mereka publik dan dapat digunakan oleh semua orang. Parameter p adalah bilangan prima, dan parameter g (biasanya disebut generator) adalah integer yang lebih kecil dari p, tetapi dengan properti sebagai berikut: Untuk setiap bilangan n antara 1 dan p, ada bilangan  $g^k \bmod p$  yang berlaku bahwa  $n = g^k \bmod p$ .

# Diffie–Hellman

The following steps describe the Diffie-Hellman exchange:

**Step 1.** Alice and Bob agree on generator  $g$  and modulus  $p$ .

**Step 2.** Alice chooses a random number  $A$  and sends Bob its public value  $A' = g^A \text{ mod } p$ .

**Step 3.** Bob chooses a random number  $B$  and sends Alice his public value  $B' = g^B \text{ mod } p$ .

**Step 4.** Alice computes  $k = (B')^A \text{ mod } p$ .

**Step 5.** Bob computes  $k' = (A')^B \text{ mod } p$ .

**Step 6.** Both  $k$  and  $k'$  are equal to  $g^{AB} \text{ mod } p$ .

# Rivest, Shamir, Adelman

- Rivest, Shamir, Adelman (RSA) adalah algoritma kunci publik ditemukan oleh Ron Rivest, Adi Shamir, dan Len Adelman dan dipatenkan pada tahun 1977.
- Hak Paten kedaluwarsa pada September 2000, dan algoritma ini sekarang berada dalam domain publik.
- Dibandingkan dengan algoritma lain, RSA adalah jauh lebih mudah untuk dipahami dan digunakan.

# Rivest, Shamir, Adelman

- Algoritma RSA sangat fleksibel dan memiliki panjang kunci variabel di mana, jika perlu, kecepatan dapat dipertukarkan untuk tingkat keamanan.
- Kunci RSA biasanya 512-2.048 bit.
- Keamanan RSA didasarkan pada sulitnya memfaktorkan bilangan yang sangat besar. Jika metode yang mudah untuk pemfaktoran ini ditemukan, maka efektivitas dari RSA akan hancur.

# Rivest, Shamir, Adelman

**Step 1.** Select two large prime numbers,  $p$  and  $q$ .

**Step 2.** Compute  $n$  using the following formula:

$$n = p \times q$$

**Step 3.** Choose a huge prime  $e$ , with the constraint that  $e$  and  $(p - 1)(q - 1)$  are relatively prime. The public key is  $(e, n)$ .

**Step 4.** Calculate the private key  $d$ :

$$e \cdot d \equiv 1 \pmod{(p - 1)(q - 1)}$$

$$d = e^{-1} \pmod{(p - 1)(q - 1)}$$

# Rivest, Shamir, Adelman

- Angka  $d$  dan  $n$  juga relatif prima. Angka  $e$  dan  $n$  adalah kunci publik. Angka  $d$  adalah kunci pribadi.
- Angka  $p$  dan  $q$  tidak lagi diperlukan. Mereka hanya digunakan untuk menghitung nilai-nilai lain dan dapat dibuang namun tidak pernah terungkap.

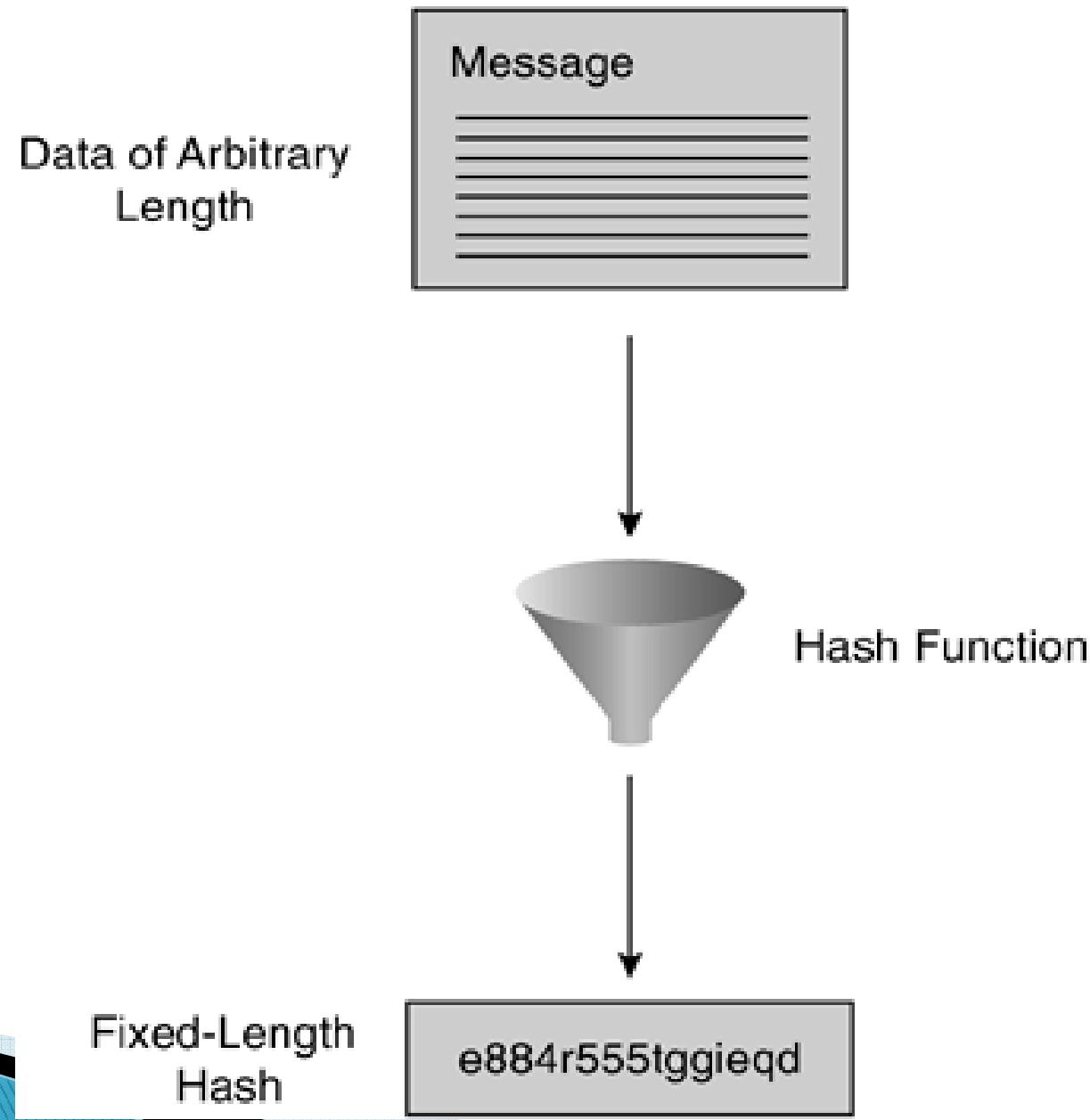
# Pretty Good Privacy

- Pretty Good Privacy (PGP) adalah paket perangkat lunak yang awalnya dikembangkan oleh Philip R. Zimmermann kriptografi yang menyediakan rutin untuk e-mail dan penyimpanan file aplikasi. Hal ini didasarkan pada protokol kriptografi yang ada, dan dapat berjalan di banyak platform. PGP pesan menyediakan enkripsi, kompresi data, dan tanda tangan digital.

# Hashing Algorithms

- Hashing adalah salah satu mekanisme yang digunakan untuk jaminan integritas data. Hashing didasarkan pada satu arah fungsi matematika, yang relatif mudah untuk menghitung, tetapi secara signifikan lebih sulit untuk dihitung balik.

# Hashing Algorithms

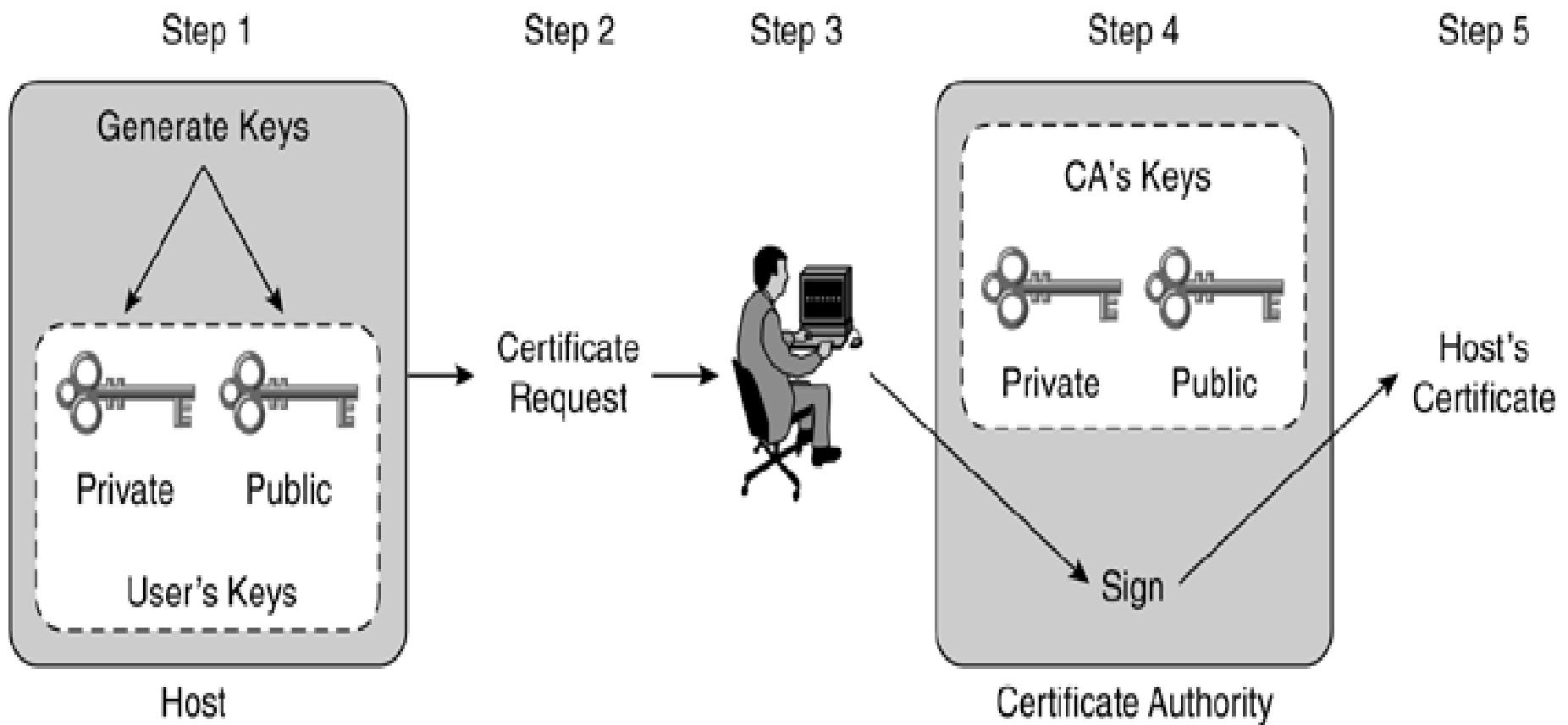


# Hashing Algorithms

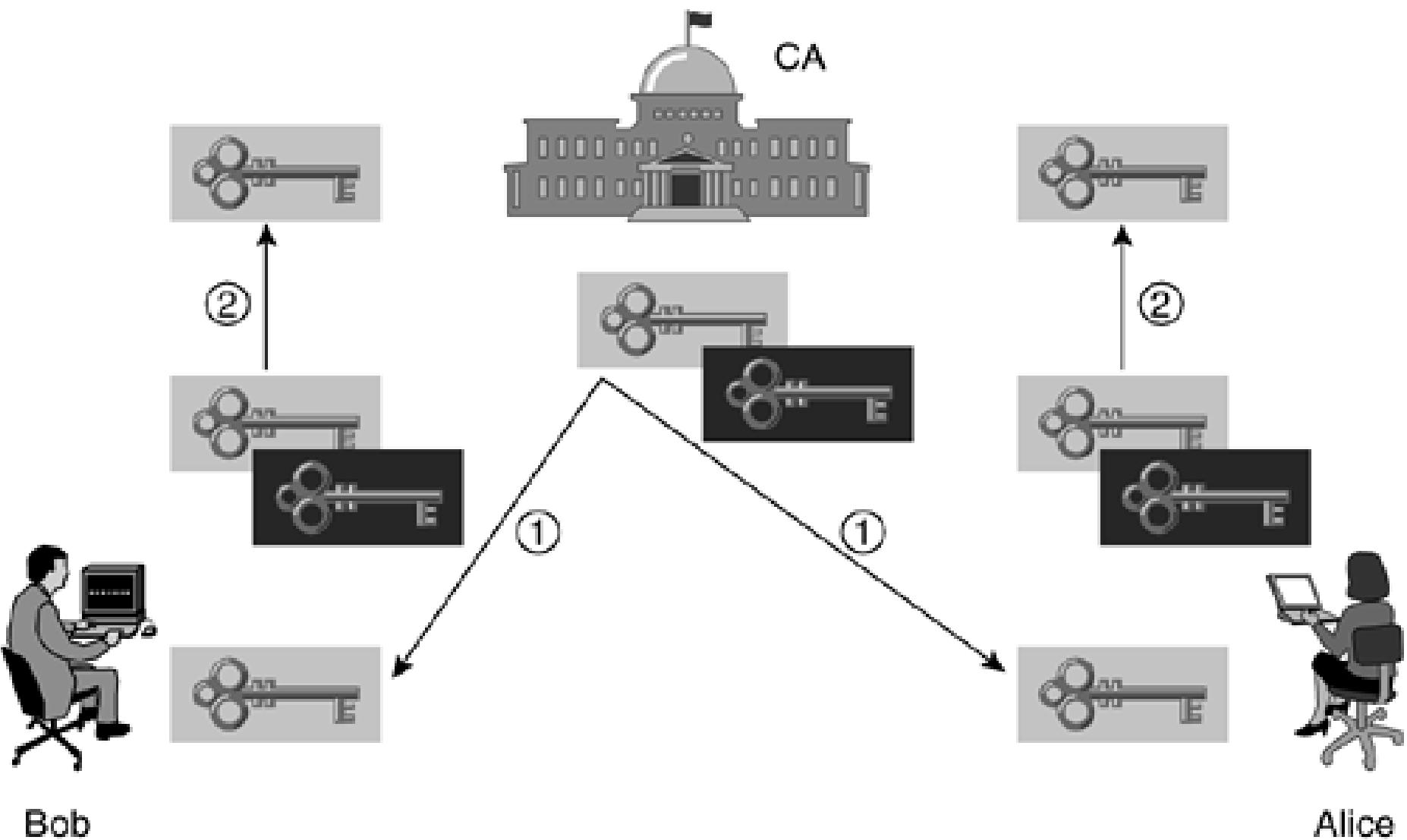
- Message Digest 5 (MD5) with 128-bit digest
- Secure Hash Algorithm 1 (SHA-1) with 160-bit digest

# Public Key Distribution

- Exchanging the public keys out-of-band or over a secure channel The exchange takes place via another channel or over a secure, already protected channel. This last approach requires the establishment of an additional secured channel between the two entities.
- Exchanging the public keys over an insecure channel In this case, the received keys have to be verified out-of-band (for example, by reading the key back over the telephone to the sending party).



# Trusted Third Party

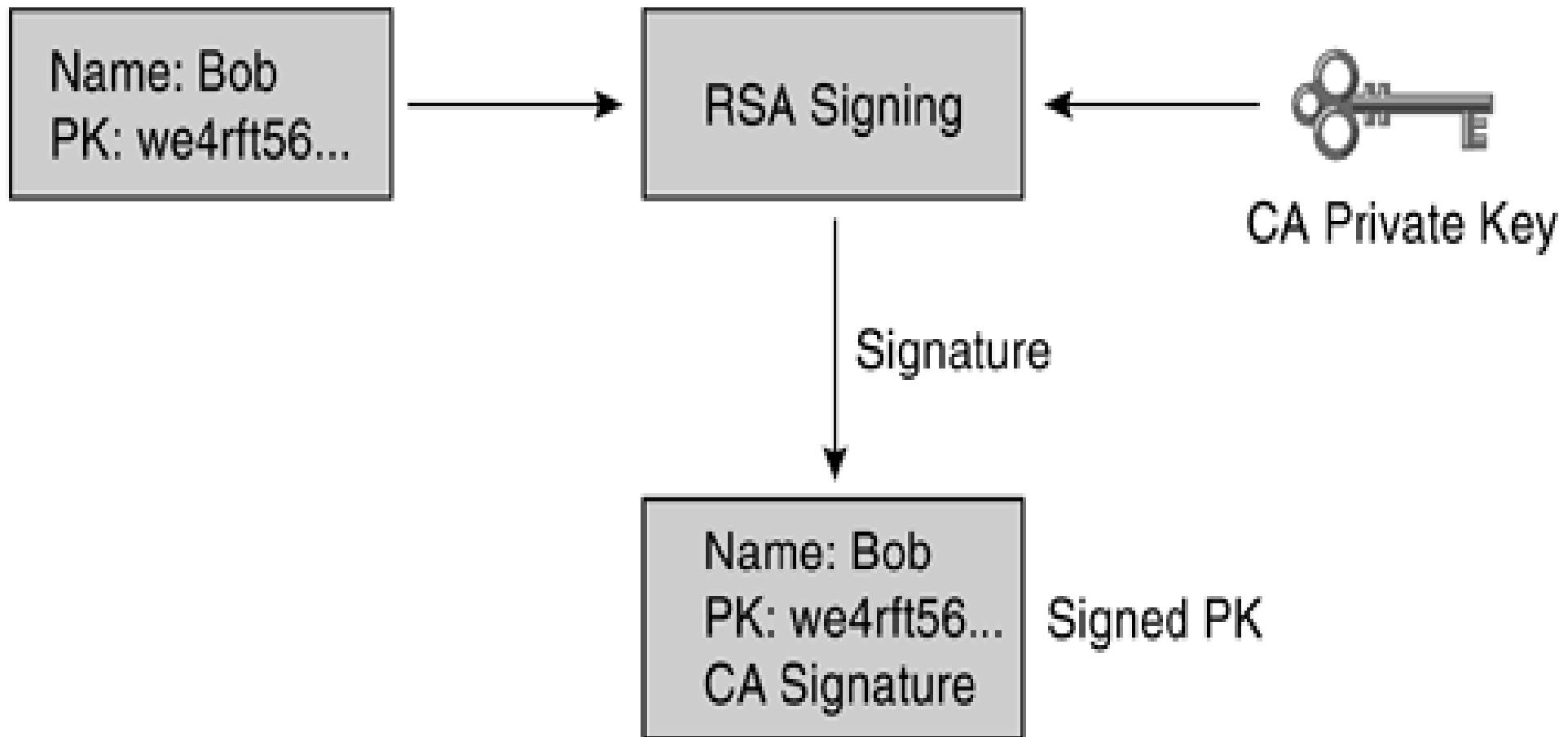


1. Alice and Bob securely exchange their public keys using one of the previously mentioned methods.
2. Alice and Bill also securely exchange their public keys.
3. Alice can now digitally sign Bill's public key using PGP and send it to Bob.
4. Bob can verify Alice's signature. He has her public key, and he can consider Bill's public key to be authentic if he trusts Alice.

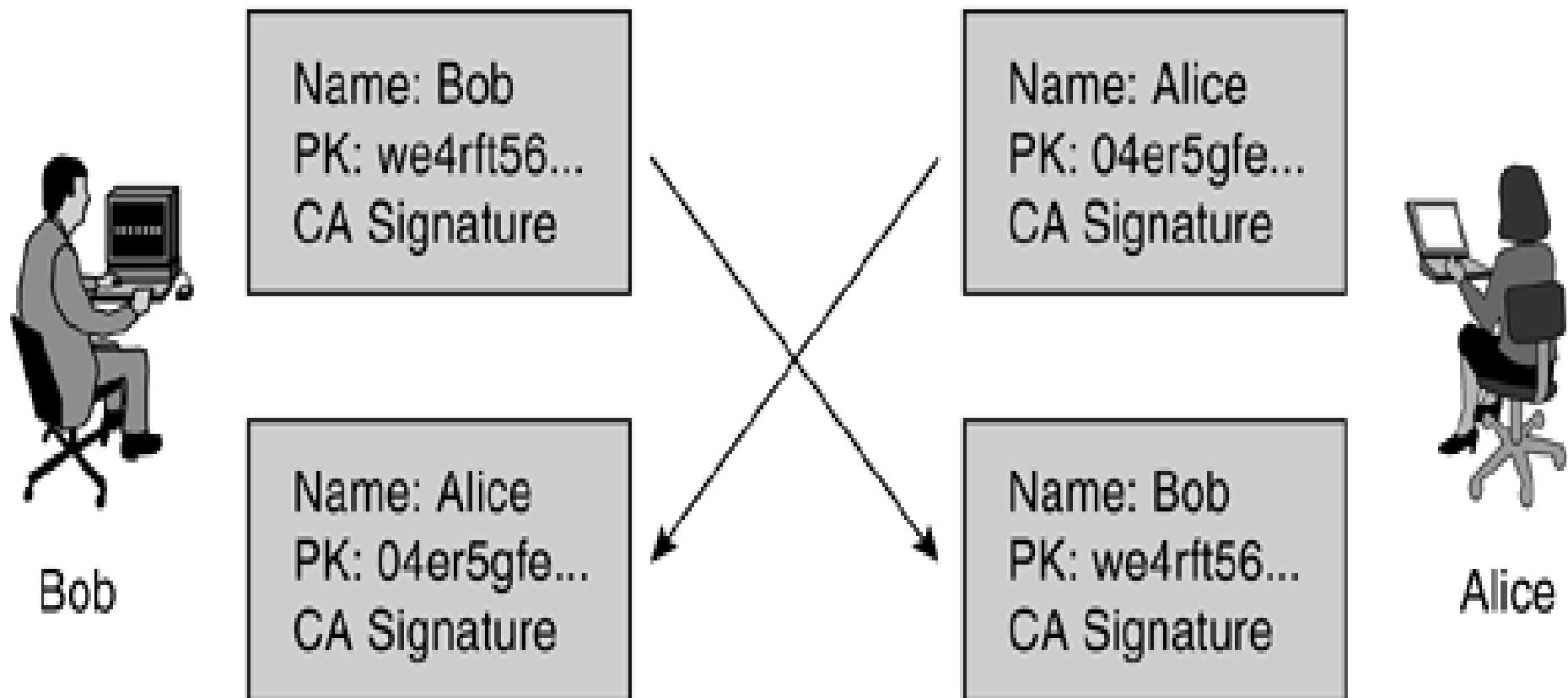
# Trusted Third Party

- Bob and Alice are users who want to communicate securely, and the certificate authority (CA) is the trusted third party.
- In the first step, Bob and Alice accept the public key from the CA.
- In the second step, Bob and Alice send their public keys to the CA.

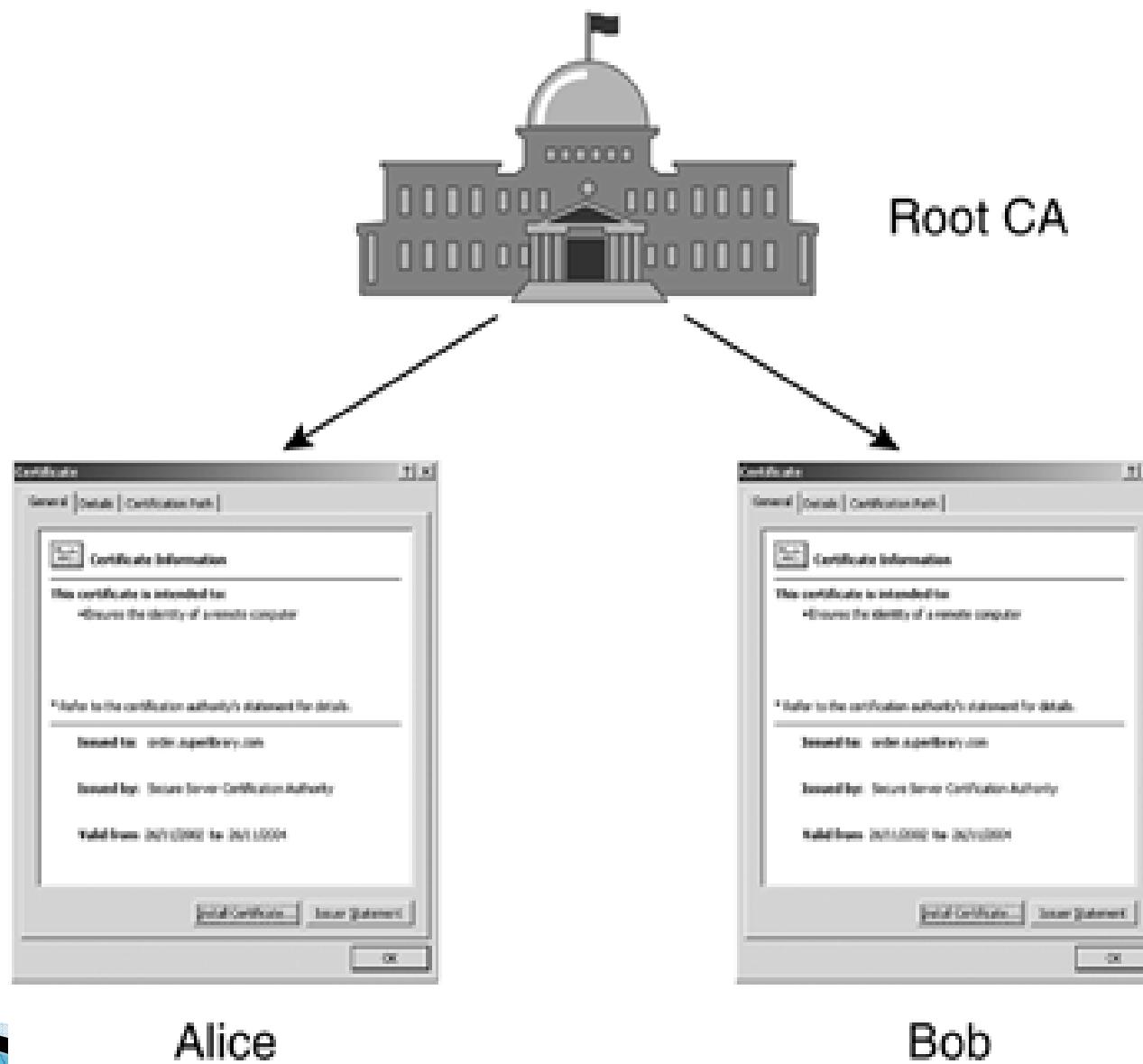
# Public Key Signing



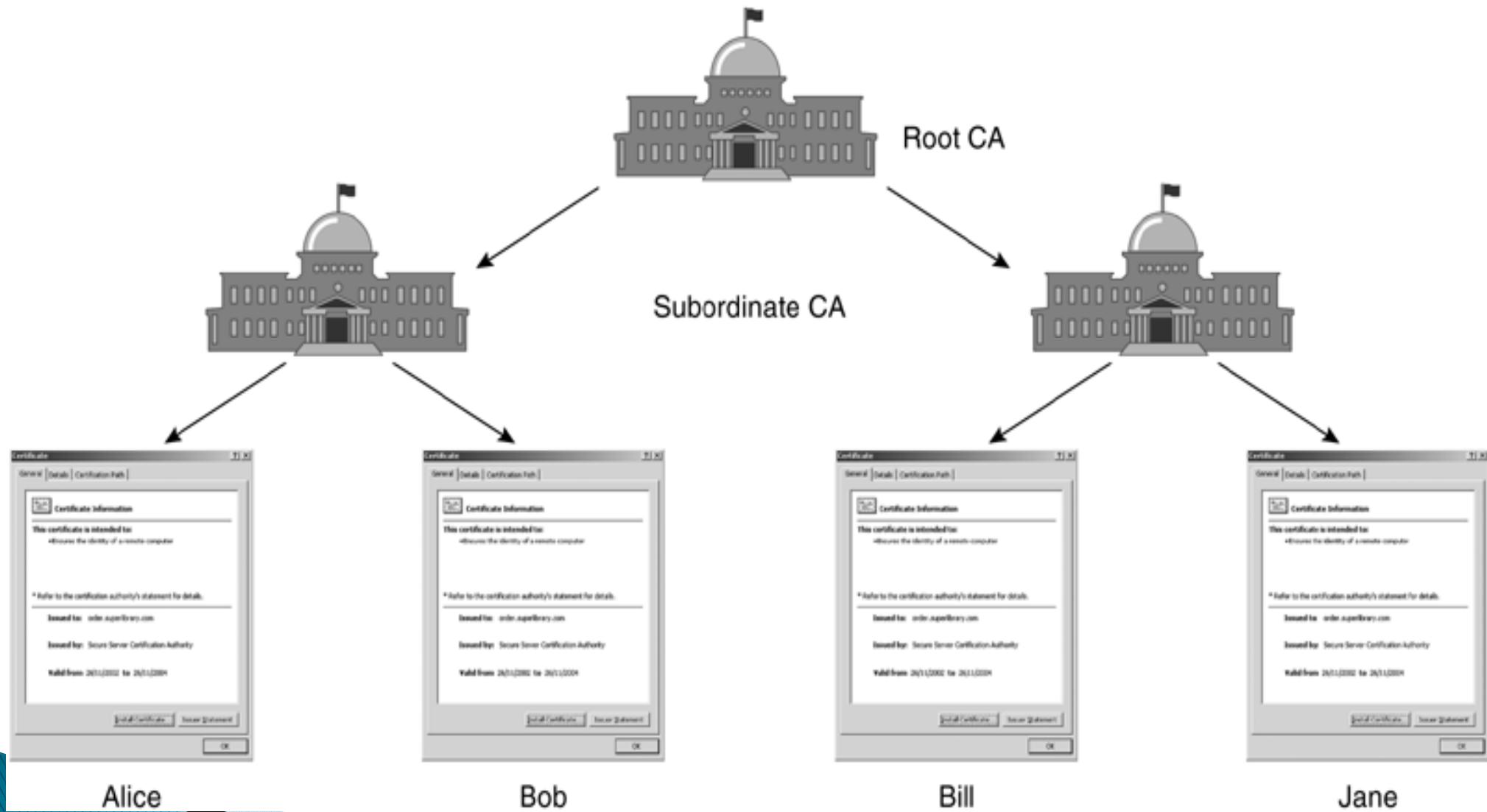
# Key Exchange



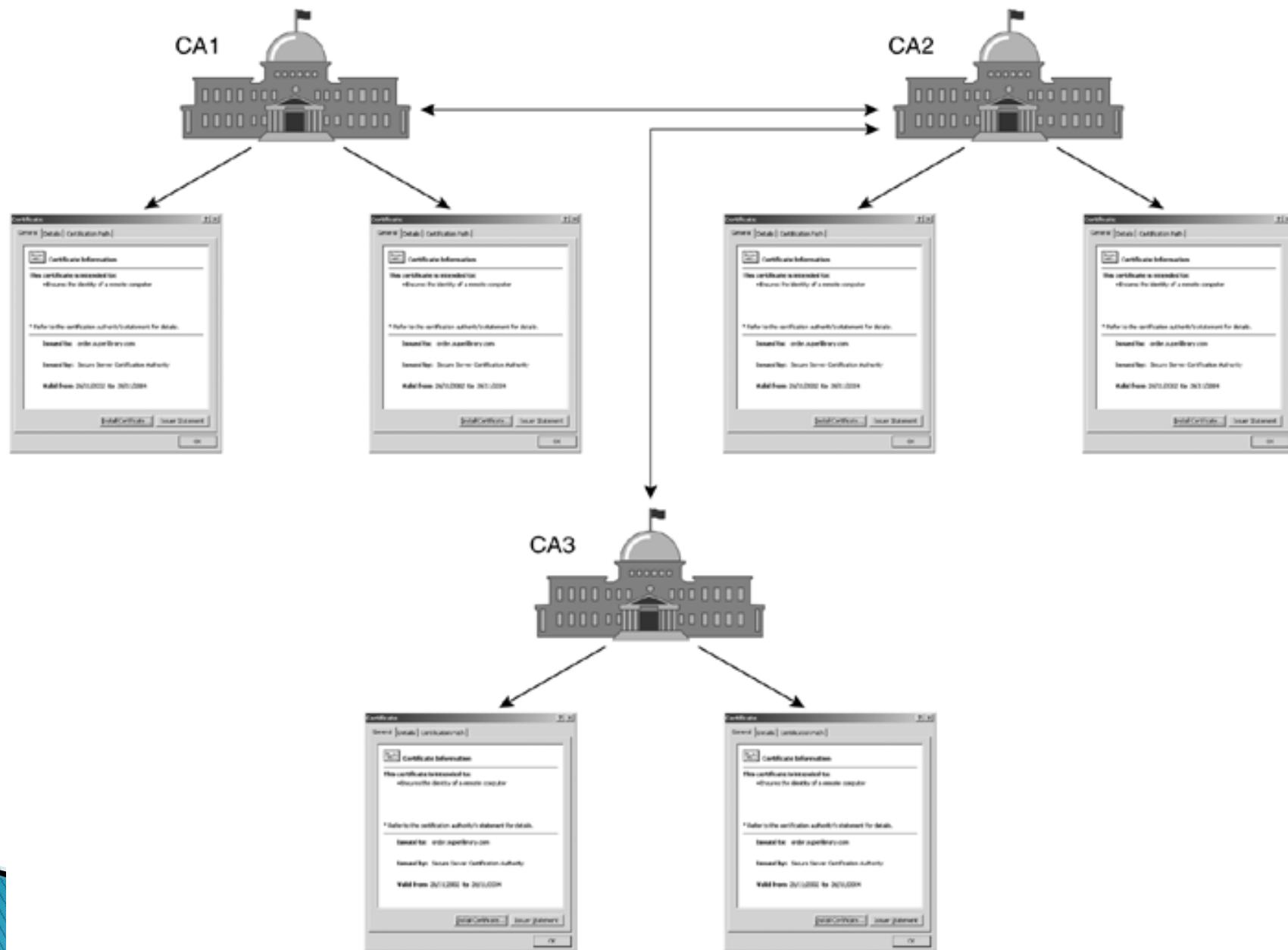
# Single Root CA



# Hierarchical CA



# Cross-Certified CA



# Enrollment Procedure

The user obtains the CA certificate with the CA's public key. This public key is used to verify the digital signature on other certificates.

The user sends identity information and the public key to the CA.

The CA authenticates the user, signs the submitted information, and returns the signed data in the form of a certificate.

# two out-of-band authentication procedures

- Verification by the user that the correct CA certificate is received
- Verification by the CA that it has received the correct enrollment information from the user

# Various enrollment protocols

- File-based requests The end user formats the enrollment request in the form of a PKCS #10 message in a file. This file is transferred to the CA, which signs the information and returns a PKCS #10 response file with the embedded certificate.
- Web-based requests This protocol runs over the HTTP protocol and is used by web browsers.
- Simple Certificate Enrollment Protocol (SCEP) This is a lightweight, HTTP-based protocol for enrollment of VPN devices.

## Certificate Viewer:"forums.comodo.com"



General Details

**This certificate has been verified for the following uses:**

SSL Server Certificate

### Issued To

Common Name (CN) forums.comodo.com  
Organization (O) Comodo Group, Inc.  
Organizational Unit (OU) Comodo EV SSL  
Serial Number 7B:52:AD:14:D1:0D:B0:52:34:90:95:55:36:0C:14:7E

### Issued By

Common Name (CN) COMODO EV SGC CA  
Organization (O) COMODO CA Limited  
Organizational Unit (OU) <Not Part Of Certificate>

### Validity

Issued On 7/13/2009  
Expires On 7/14/2011

### Fingerprints

SHA1 Fingerprint DB:93:EE:2B:C9:71:59:15:B8:97:6D:AD:A8:28:72:93:EF:1B:84:3E  
MD5 Fingerprint 5A:CF:47:B2:6C:E4:42:6A:10:63:AC:9F:3D:90:48:B7

Close

# Certificate Viewer:"forums.comodo.com"



[General](#) [Details](#)

## Certificate Hierarchy

- AddTrust External CA Root
  - COMODO EV SGC CA
    - forums.comodo.com

## Certificate Fields

- forums.comodo.com
  - Certificate
    - Version
    - Serial Number
    - Certificate Signature Algorithm
  - Issuer
  - Validity
    - Not Before
    - Not After



## Field Value

CN = COMODO EV SGC CA  
O = COMODO CA Limited  
L = Salford  
ST = Greater Manchester  
C = GB

[Export...](#)

[Close](#)

# Revocation Procedure

- The private key is compromised.
- The contract is terminated.
- The private key is lost.
- A VPN router is replaced.

# A certificate can be placed on a CRL

1. The certificate is no longer valid.
2. The CA administrator is contacted and requested to revoke the certificate. The administrator may require additional authentication.
3. The CA administrator places the certificate on the CRL.
4. A new CRL is published.
5. End users check the CA for a new CRL after their old CRL has expired.