Chapter 15

Comparing Secret Key, Public Key, and Message Digests

Some cryptographers believe that secret key, public key, and message digest systems are so different that comparisons are irrelevant. However, we think you'll find this review useful. In this chapter we compare and contrast secret and public key encryption in terms of speed, key length, and cryptographic assurances so that you can readily understand why they are used together in the real-world systems discussed in Part IV.

Various public key cryptographic systems offer various mixes of cryptographic assurances and operation speeds. Because RSA is the most widely used approach and offers every cryptographic assurance, in this chapter we often use RSA as a proxy for public key cryptography.[1]

Although DES is all but obsolete, encrypting with DES three times (Triple DES) is the de facto standard until Rijndael, the new standard, becomes widely used. Interestingly, both DES and RSA were released at about the same time in the late 1970s. Secret key systems have been studied and used for thousands of years; public key cryptography is less than 50 years old. Nevertheless, both types can be attacked. Although public key can be attacked in ways that secret key cannot (see Chapter 22), neither is more immune to attack than the other.

Public key cryptography (e.g., RSA) can do anything that secret key cryptography (e.g., DES) can do and more. So why does anyone use secret key cryptography?

Encryption Speed

Encryption speed: Secret key is faster.

Speed is the biggest difference between the two types. DES is at least 100 times faster than RSA; a 30-second DES encryption takes RSA more than 50 minutes. In some cases, DES is as much as 1,000 times faster than RSA.

[1] This is not meant to endorse RSA or DES.

Although Triple DES (DES encryption done three times) is obviously slower than DES, all of NIST's candidates for AES are faster than (single) DES. For example, RC6, a NIST AES finalist, is about three times faster than DES in some software implementations.

In general, message digests complete a little faster than do secret key methods, but message digests don't offer confidentiality.

Public key cryptography is used only sparingly.

As you'll see in Part IV, real-world systems such as secure e-mail, Secure Socket Layer, and IPsec encrypt (and decrypt) much more with secret key cryptography. Public key cryptography is used only sparingly, mostly for authentication and exchanging (or agreeing on) a secret key.

Key Length

Secret keys are much smaller than public keys.

A few years ago, DES (with a 56-bit key) and RSA implemented with a 512-bit key were considered adequate for secure encryption. This means that an RSA key was about 10 times the size of a DES key.

As you've seen, 56-bit encryption is so insecure that NIST required all AES candidates to support at least 128-bit key length. Similarly, RSA users have also been forced to double their key size to at least 1,024 bits, and many users demand an even bigger key. This means that the minimum advisable secret key length (128) is still about one-tenth the length of the advisable RSA key (1,024).

A new wrinkle: elliptic curve cryptography

RSA keys are getting so big that many public key implementations have been forced to use a new public key technology called elliptic curve cryptography (ECC), described in Appendix A. Because ECC uses much smaller keys than RSA, Motorola, for example, uses ECC in its cellular phones.

Ease of Key Distribution

Key distribution: Public key is better.

As you saw in Chapter 7, secret key distribution is a tough problem. Each sender and receiver must share a secret key. Managing secret key networks is feasible only for a small network of users and is all but impossible for a network of 1,000 or more users. Secret key cryptography by itself is not an option for a network of millions of users, so it can't support Internet e-commerce. Public key was invented mostly to solve the problem of secret key distribution.

Public keys don't need to be secret.

The security of a public key cryptographic system relies on each person knowing his or her private key, undisclosed to anyone and not shared with anyone. A public key, by contrast, can be freely distributed to more than one person. A public key can be put on a business card, published on an Internet site, or e-mailed as plaintext.

But public keys must be protected.

Because the public key doesn't have to be a secret between Alice and Bob, public keys are easier to distribute than secret keys are. But as you'll see in Chapter 22, if BlackHat can trick Bob into thinking that BlackHat's public key is really Alice's public key, BlackHat can read all of Bob's confidential messages to Alice. Digital certificates help prevent BlackHat from perpetrating this fraud (see Chapters 16 through 19). Even with its distribution problems, for ease of key distribution, public key is better than secret key.

Cryptographic Assurances

As shown in Table 15-1, both methods offer confidentiality, user authentication, and message integrity. Public key also offers nonrepudiation; secret key, without a trusted third party, doesn't offer this assurance.

Symmetric (Secret) Key

Review: symmetric cryptography

Recall that secret key cryptography is often referred to as symmetric cryptography because the same key encrypts and decrypts (see Figure 15-1). Usually there are only two holders of a particular secret key. Both holders share their secret key equally; they are equal partners, and neither can do more or less with the secret key than the other. Secret key cryptography implements confidentiality, authentication, and integrity symmetrically for both holders of the secret key; either holder can demand (and offer) confidentiality, authentication, and message integrity from (to) the other.

Asymmetric (Public) Key

Review: asymmetric cryptography

In contrast, public key is called asymmetric cryptography. There is one and only one private key holder and usually many public key holders. The public and private key holders are not equal partners. Consequently, the cryptographic assurances—confidentiality, authentication, integrity, and nonrepudiation—are asymmetrically shared among private and public key holders (see Figure 15-2).

Table 15-1 Cryptographic assurances.

Assurance	Prevents	Secret Key	Public Key
Confidentiality	Snooping	x	x
Authentication	Masquerading	x	x
Integrity	Message alteration without detection	x	x
Nonrepudiation	Sender's false denial		x

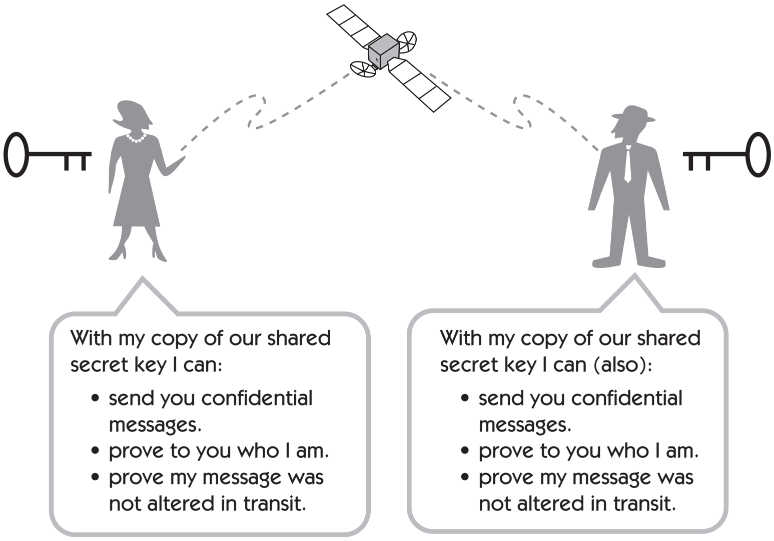
With my copy of our shared secret key I can:
• send you confidential messages.
• prove to you who I am.
• prove my message was not altered in transit.
With my copy of our shared secret key I can (also):
• send you confidential messages.
• prove to you who I am.
• prove my message was not altered in transit.

Figure 15-1 In symmetric (secret key) cryptography the holders look symmetric and have identical attributes.

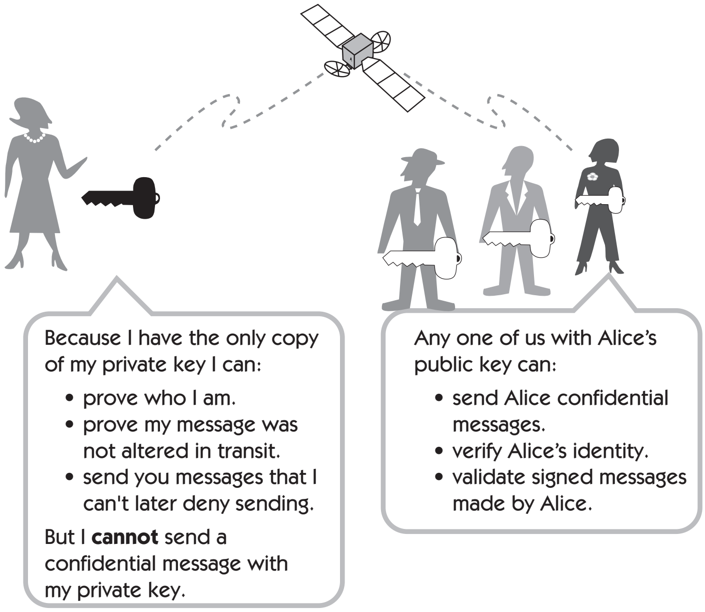
Because I have the only copy of my private key I can:
• prove who I am.
• prove my message was not altered in transit.
• send you messages that I can't later deny sending.
But I **cannot** send a confidential message with my private key.
Any one of us with Alice's public key can:
• send Alice confidential messages.
• verify Alice's identity.
• validate signed messages made by Alice.

Figure 15-2 In asymmetric (public key) cryptography the holders look different and have different attributes.

Only the public key holders (Bob) can send confidential messages to the private key holder (Alice). In contrast, if Alice encrypts a message with her private key, it's decipherable by Bob or anyone with the public key, so it's not confidential. But Alice's signed (private key encrypted) messages prove her authenticity (identity) and prove that her messages were not altered in transit (integrity). In addition, she can't later deny sending the message (nonrepudiation).

Review

Table 15-2 summarizes similarities and differences in various aspects of secret key and public key cryptography.

Table 15-2 Summary of secret and public key attributes.

Attribute	Secret Key	Public/Private Key
Years in use	Thousands	Less than 50
Current main use	Bulk data encryption	Key exchange, digital signatures
Current standard	DES, Triple DES, and Rijndael	RSA, Diffie-Hellman, DSA (Elliptic curve is a challenging newcomer)
Encryption / decryption speed	Fast	Slow
Keys	Shared secret between at least two people (usually only two)	Private: kept concealed by one person Public: widely distributed
Key exchange	Difficult and risky to transfer a secret key	Easy and less risky to deliver a public key Private key never shared
Key length	56-bit obsolete 128-bit considered safe	1,024 suggested (RSA) Some users demand 2,048 ~172 (elliptic curve)
Confidentiality, authentication, message integrity	Yes	Yes
Nonrepudiation	No Need trusted third party to act as witness	Yes Digital signatures: don't need trusted third party
Attacks	Yes	Yes