Ordering Information: **Wireless Internet & Mobile Business How to Program** **&** **The Complete Wireless Internet & Mobile Business Training Course**

- View the complete **Table of Contents**
- Read the **Preface**
- Download the **Code Examples**

To view all the Deitel products and services available, visit the Deitel Kiosk on InformIT at **www.informIT.com/deitel**.

To follow the Deitel publishing program, sign-up now for the *DEITEL™ BUZZ ONLINE* e-mail newsletter at **www.deitel.com/newsletter/subscribeinformIT.html** To learn more about Deitel instructor-led corporate training delivered at your location, visit **www.deitel.com/training** or contact Christi Kelsey at (978) 461-5880 or e-mail: **christi.kelsey@deitel.net**.

Note from the Authors: This article is an excerpt from Chapter 6, Section 6.4 of *Wireless Internet & Mobile Business How to Program*. This article introduces public-key cryptography and discusses its role in Internet and wireless Internet security. We discuss different types of public-key cryptography including RSA and different systems that use this type of security. This is an introductory article suitable for readers of any level.

6.4 Public-Key Cryptography

In 1976, Whitfield Diffie and Martin Hellman, researchers at Stanford University, developed *public-key cryptography* to solve the problem of exchanging keys securely. Public-key cryptography is asymmetric. It uses two inversely related keys: a *public key* and a *private key*. The private key is kept secret by its owner, while the public key is freely distributed. If the public key is used to encrypt a message, only the corresponding private key can decrypt it, and vice versa (Fig. 6.1). Each party in a transaction has both a public key and a private key. To transmit a message securely, the sender uses the receiver's public key to encrypt the message. The receiver then decrypts the message using his or her unique private key. Assuming that the private key has been kept secret, the message cannot be read by anyone other than the intended receiver. Thus the system ensures the privacy of the message. The defining property of a secure public-key algorithm is that it is "computationally infeasible" to deduce the private key from the public key. Although the two keys are mathematically related, deriving one from the other would take enormous amounts of computing power and time, enough to discourage attempts to deduce the private key. An outside party cannot participate in communication without the correct keys. The security of the entire process is based on the secrecy of the private keys. Therefore, if a third party obtains the private key used in decryption, the security of the whole system is compromised. If a system's integrity is compromised, the user can simply change the key, instead of changing the entire encryption or decryption algorithm.

Either the public key or the private key can be used to encrypt or decrypt a message. For example, if a customer uses a merchant's public key to encrypt a message, only the merchant can decrypt the message, using the merchant's private key. Thus, the merchant's identity can be authenticated, since only the merchant knows the private key. However, the merchant has no way of validating the customer's identity, since the encryption key the customer used is publicly available.

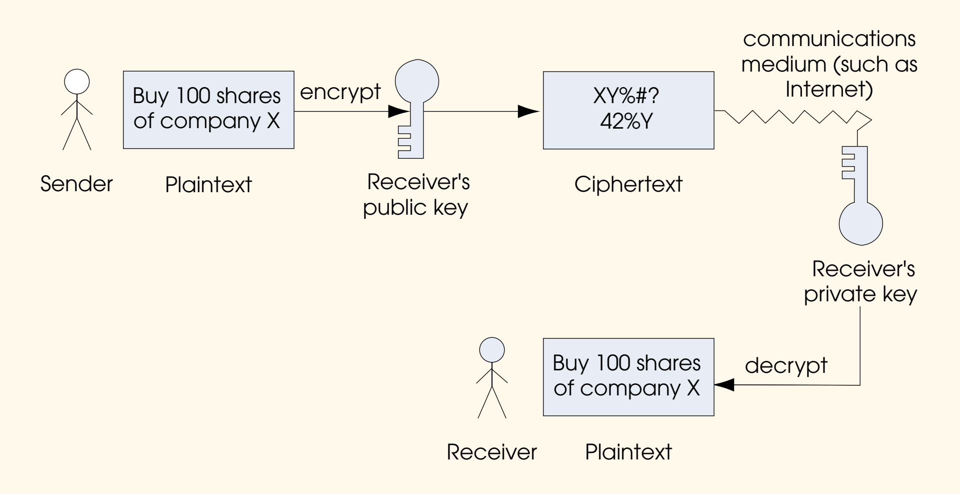

Fig. 6.1 Encrypting and decrypting a message using public-key cryptography.

If the decryption key is the sender’s public key and the encryption key is the sender’s private key, the sender of the message can be authenticated. For example, suppose a customer sends a merchant a message encrypted using the customer’s private key. The merchant decrypts the message using the customer’s public key. Since the customer encrypted the message using his or her private key, the merchant can be confident of the customer’s identity. This process authenticates the sender, but does not ensure confidentiality, as anyone could decrypt the message with the sender’s public key. This systems works as long as the merchant can be sure that the public key with which the merchant decrypted the message belongs to the customer, and not a third party posing as the customer.

These two methods of public-key encryption can actually be used together to authenticate both participants in a communication (Fig. 6.2). Suppose a merchant wants to send a message securely to a customer so that only the customer can read it, and suppose also that the merchant wants to provide proof to the customer that the merchant (not an unknown third party) actually sent the message. First, the merchant encrypts the message using the customer's public key. This step guarantees that only the customer can read the message. Then the merchant encrypts the result using the merchant’s private key, which proves the identity of the merchant. The customer decrypts the message in reverse order. First, the customer uses the merchant’s public key. Since only the merchant could have encrypted the message with the inversely related private key, this step authenticates the merchant. Then the customer uses the customer’s private key to decrypt the next level of encryption. This step ensures that the content of the message was kept private in the transmission, since only the customer has the key to decrypt the message. Although this system provides extremely secure transactions, the setup cost and time required prevent widespread use.

The most commonly used public-key algorithm is *RSA*, an encryption system developed in 1977 by MIT professors Ron Rivest, Adi Shamir and Leonard Adleman.[7] Today, most Fortune 1000 companies and leading e-commerce businesses use their encryption and authentication technologies. With the emergence of the Internet and the World Wide Web, their security work has become even more significant and plays a crucial role in e-commerce transactions. Their encryption products are built into hundreds of millions of copies of the most popular Internet applications, including Web browsers, commerce servers and e-mail systems. Most secure e-commerce transactions and communications on the Internet use RSA products. For more information about RSA, cryptography and security, visit **`www.rsasecurity.com`**.

Pretty Good Privacy (PGP) is a public-key encryption system used for the encryption of e-mail messages and files. PGP was designed in 1991 by Phillip Zimmermann.[8] PGP can also be used to provide digital signatures that confirm the author of an e-mail or public posting.

PGP is based on a “web of trust;” each client in a network can vouch for another client’s identity to prove ownership of a public key. The “web of trust” is used to authenticate each client. If users know the identity of a public key holder, through personal contact or another secure method, they validate the key by signing it with their own key. The Web grows as more users validate the keys of others. To learn more about PGP and to download a free copy of the software, go to the MIT Distribution Center for PGP at **`web.mit.edu/network/pgp.html`**.

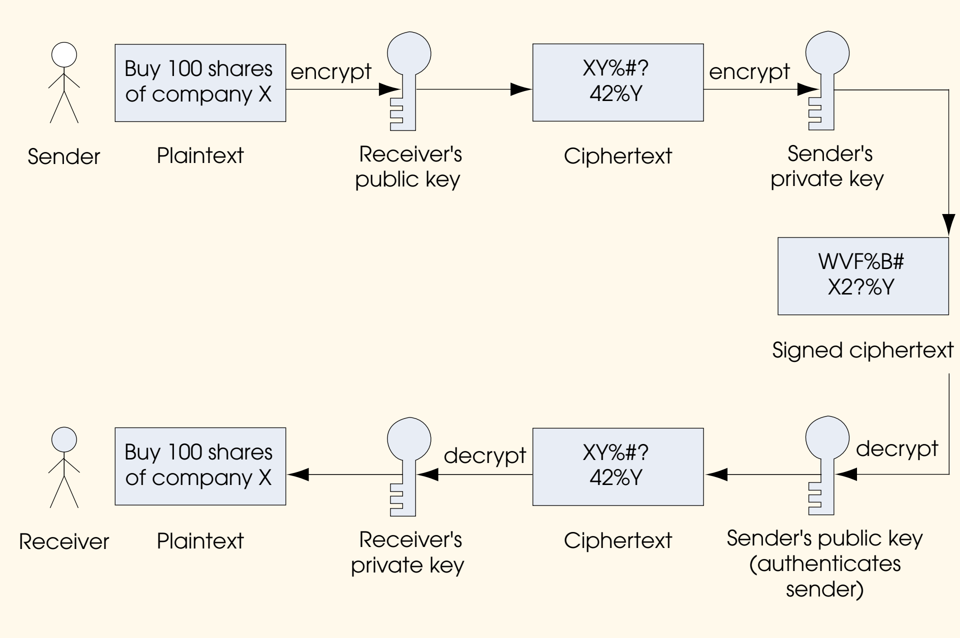

Fig. 6.2 Authentication with a public-key algorithm

WORKS CITED

7. `<www.rsasecurity.com/rsalabs/rsa_algorithm>`

8. `<www.pgpi.org/doc/overview>`