Chapter15. System Security

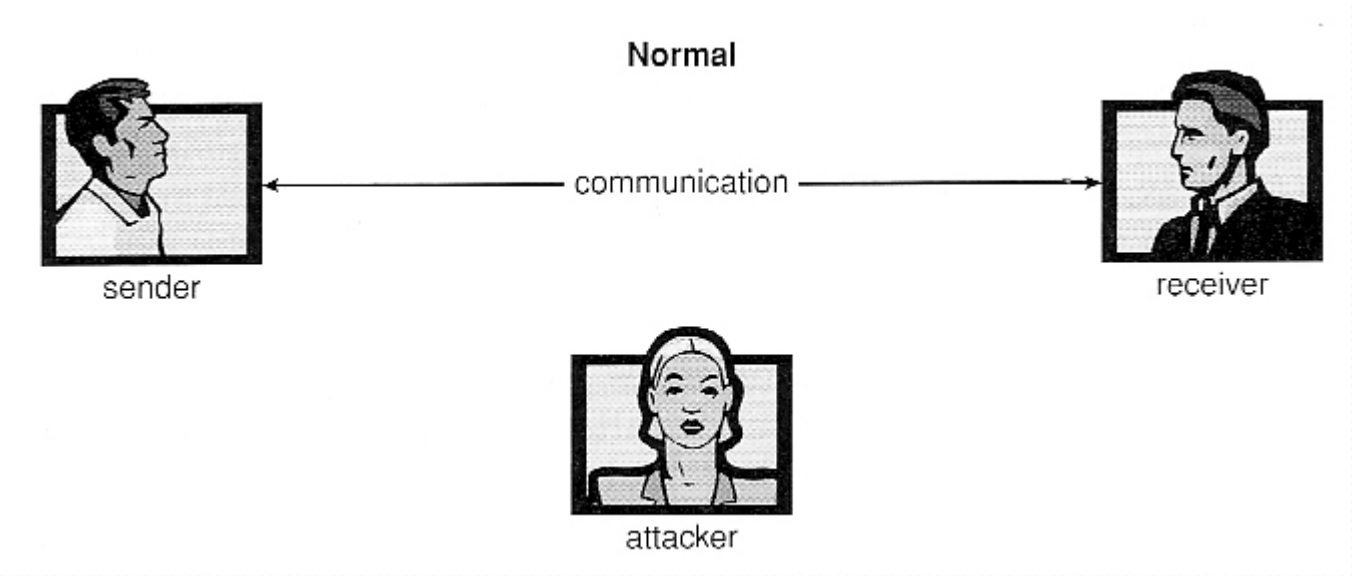
Contents

- The Security Problem
- Program Treats
- System and Network Threats
- Cryptography as a Security Tool
- User Authentication
- Implement Security Defenses
- Firewalling to Protect System and Networks

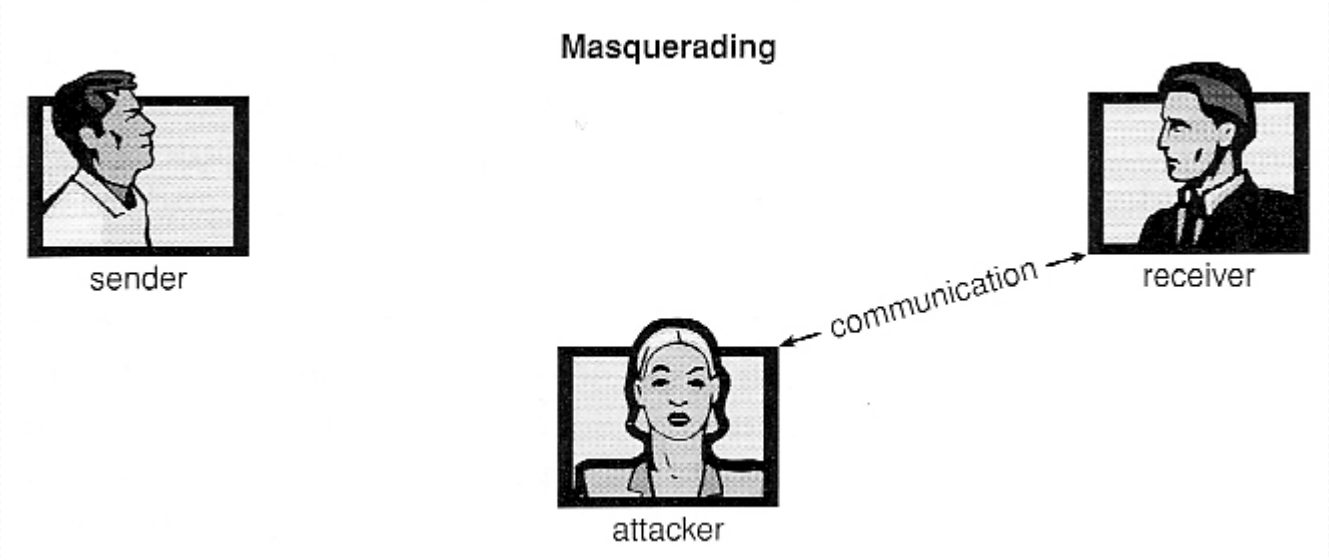
The Security Problem

- ***Security*** requires not only an adequate protection system but also ***consideration of external environment*** within which the system operates.
  - Breach of confidentiality
  - Breach of integrity
  - Breach of availability
  - Theft of service
  - Denial of services

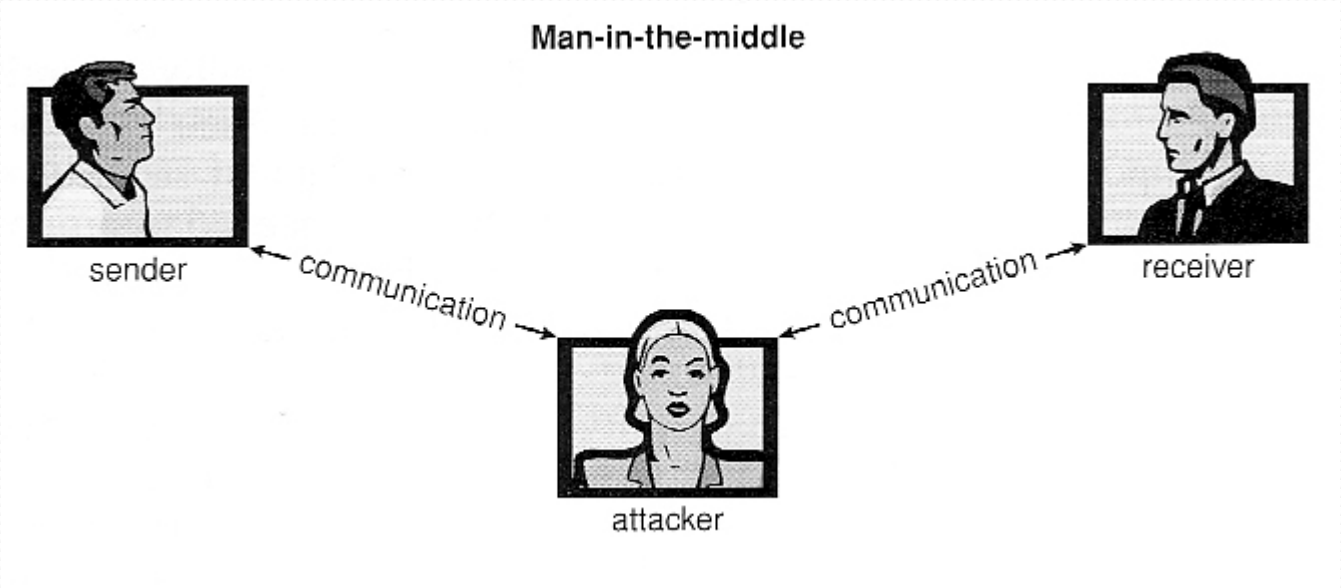
The Security Problem
Standard security attacks



The Security Problem
Standard security attacks



The Security Problem
Standard security attacks



The Security Problem
Four Security Level

- Physical
  - Even *general access to terminals in a computer room* offers some opportunities for an attacker.
- Human
  - Social Engineering – *Pishing*, *Dumpster diving*, etc
- Operating System
  - runaway processes (denial of service), memory-access violations, stack overflow violations, etc
- Network

Program Threats
Trojan Horse and others

- Trojan Horse is a ***program*** that secretly performs some ***maliciousness in addition to its visible actions***.
  - *Current directory changer*
  - The *same name* as a legitimate program
  - *Login emulator*
  - *Spyware*
- *Trap Door*
- *Logic bomb*

Program Threats
Stack and Buffer Overflow

```
#include <stdio.h>
#define BUFFER_SIZE 256

int main(int argc, char *argv[])
{
   char buffer[BUFFER_SIZE];

   if (argc < 2)
      return -1;
   else {
      strcpy(buffer,argv[1]);
      return 0;
   }
}
```

Figure 15.2 C program with buffer-overflow condition.

Program Threats
Stack and Buffer Overflow

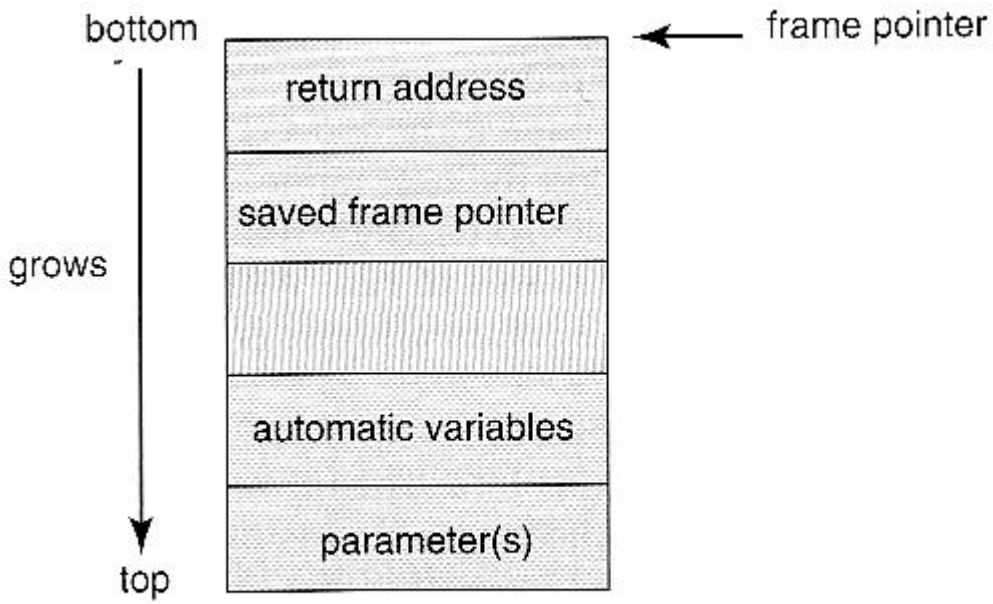


Figure 15.3 The layout for a typical stack frame.

Program Threats
Stack and Buffer Overflow

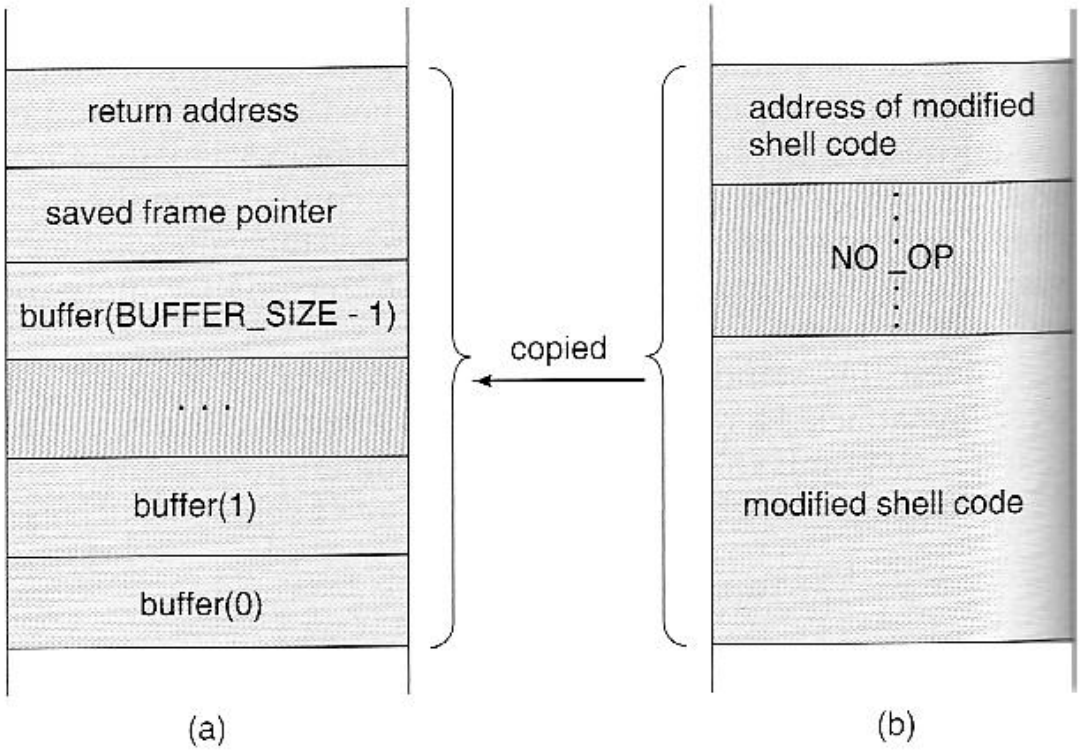


Figure 15.4 Hypothetical stack frame for Figure 15.2, (a) before and (b) after

Program Threats

Viruses

- A virus is a ***fragment of code embedded in an otherwise legitimate program***, designed to *replicate itself* and wreaking *havoc*.
  - **File**
  - **Boot**
  - **Macro**
  - **Source code**
  - **Polymorphic**
  - **Encrypted**
  - **Stealth**
  - **Tunneling**
  - **Multipartite**
  - **Armored**

System and Network Threats

- ***Worms***
  - *A process* that uses the fork / spawn process to make copies of itself in order to wreak havoc on a system.
- ***Port Scanning***
  - Attempting to connect to every known (or common or possible) network port on some remote machine, and to attempt to make contact.
- ***Denial of Services(DOS)***
  - *DOS* attacks do not attempt to actually access or damage systems, but *merely to clog them up so badly* that they cannot be used for any useful work

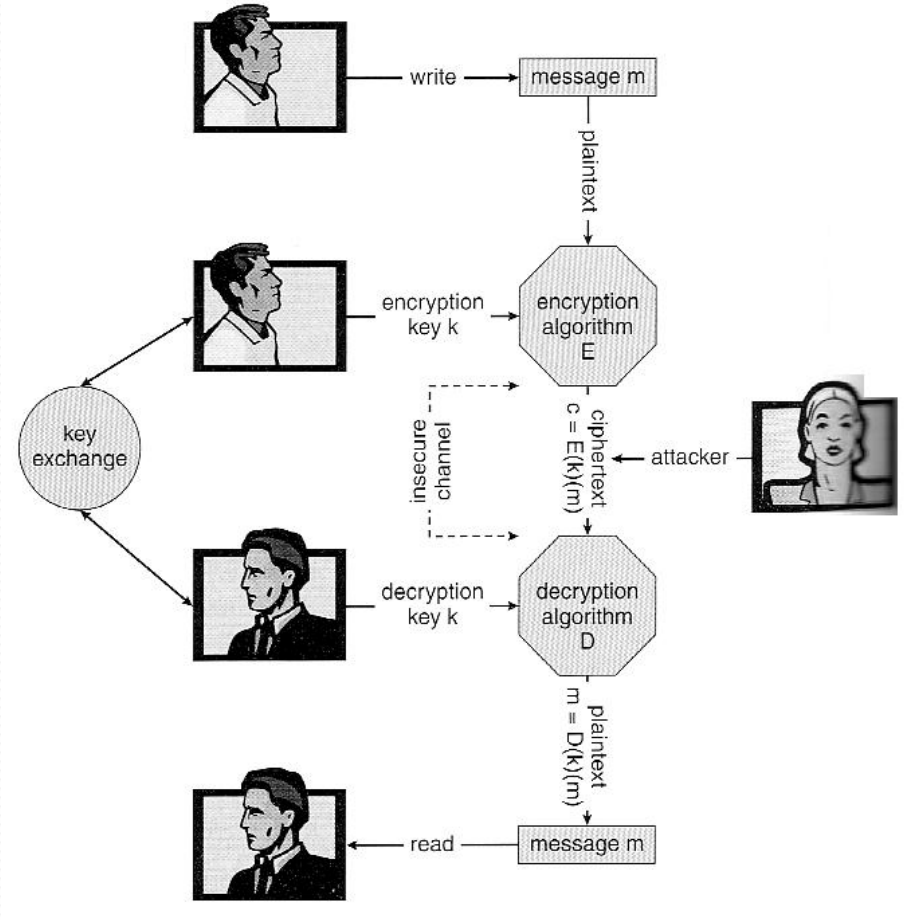
Cryptography as a Security Tool

- Within a given computer the transmittal of messages is safe, reliable and secure, because the OS knows exactly where each one is coming from and where it is going.
- On a *network*, however, things aren't so straightforward.
  - ***Trust***
  - ***Confidentiality***
- **Cryptography** can help with both of these problems, through a system of *secrets* and *keys*.

Cryptography as a Security Tool Encryption

- Components
  - K : Set of Keys
  - M : Set of messages
  - E : A Function. $K \rightarrow (M \rightarrow C)$
  - D : A Function. $K \rightarrow (C \rightarrow M)$

Cryptography as a Security Tool Encryption



Cryptography as a Security Tool Encryption–Symmetric Encryption

- With ***symmetric encryption*** the same key is used for both encryption and decryption, and ***must be safely guarded.***
  - The *Data-Encryption Standard(DES)*
  - The *Advanced Encryption Standard(AES)*
  - *RC5* can vary in key length, block size, and the number of transformations, and *runs on a wide variety of CPUs* using only basic computations.
  - *RC4* is a *stream cipher.*

Cryptography as a Security Tool Encryption–Asymmetric Encryption

- With ***asymmetric encryption***, the ***decryption key,*** K_d, is not the same as ***the encryption key,*** K_e.
- The encryption key can be made publicly available, and only the decryption key needs to be kept secret.

Cryptography as a Security Tool Encryption–Asymmetric Encryption

- RSA Algorithm
  - Based on two large prime numbers, ***p*** and ***q***, and their product ***N***.
  - $(K_e * K_d) \% [(p - 1) * (q - 1)] = = 1$
  - The encryption algorithm
    - $c = E(K_e)(m) = m^{Ke} \% N$
  - The decryption algorithm
    - $m = D(K_d)(c) = c^{Kd} \% N$

Cryptography as a Security Tool Encryption–Asymmetric Encryption

- Example
  - $p = 7$, $q = 13$, $N = 7 * 13 = 91$, $m = 42$
  - $(p - 1) * (q - 1) = 6 * 12 = 72$
  - Select $K_e < 72$ and relatively prime to 72 : *5*
  - K_d, such that $(K_e * K_d) \% 72 = = 1$: *29*
  - The public key : (5, 91)
  - The private key : (29, 91)
  - Encrypt: $c = 42^5 \% 91 = 35$
  - Decrypt: $m = 35^{29} \% 91 = 42$

Cryptography as a Security Tool Encryption–Authentication

- ***Authentication*** involves *verifying the identity of the entity* who transmitted a message.
  - K : Set of Keys
  - M : Set of messages
  - A : Set of authenticators
  - S : A Function. $K \rightarrow (M \rightarrow A)$
  - V : A Function. $K \rightarrow (M*A \rightarrow \{true, false\})$
- Hashing → Encryption → Decryption

Cryptography as a Security Tool Encryption–Authentication

- ***Hashing***
  - ***Hash functions, H(m)*** generate a small fixed-size block of data from m : *message digest*, or *hash value*
  - The function must be ***collision resistant on m :***
    - $H(m') \neq H(m)$
  - MD5 / SHA-1
- ***A message-authentication code(MAC)***
  - Uses symmetric encryption and decryption of the message digest.
  - Generates a cryptographic checksum.
- ***Digital-Signature algorithm***

Cryptography as a Security Tool Encryption–Authentication

- There are three *good reasons* for having separate algorithms for encryption of messages and authentication of messages.
  - Authentication algorithms typically ***require fewer calculations***, making verification a faster operation than encryption.
  - Authenticators are almost always smaller than the messages, improving *space efficiency*.
  - Sometimes we want authentication only, and not confidentiality, such as when a vendor issues a new software patch.

Cryptography as a Security Tool Encryption–Key Distribution

- Key distribution with symmetric cryptography
  - Because all keys must be kept secret, and they obviously can't be transmitted over unsecure channels.
  - ***out-of-band***, say via paper or a confidential conversation. : Key management problem.
  - Solution : Asymmetric cryptography

Cryptography as a Security Tool Encryption–Key Distribution

- ***Digital Certificate***
  - A public key digitally signed by a *trusted party*.

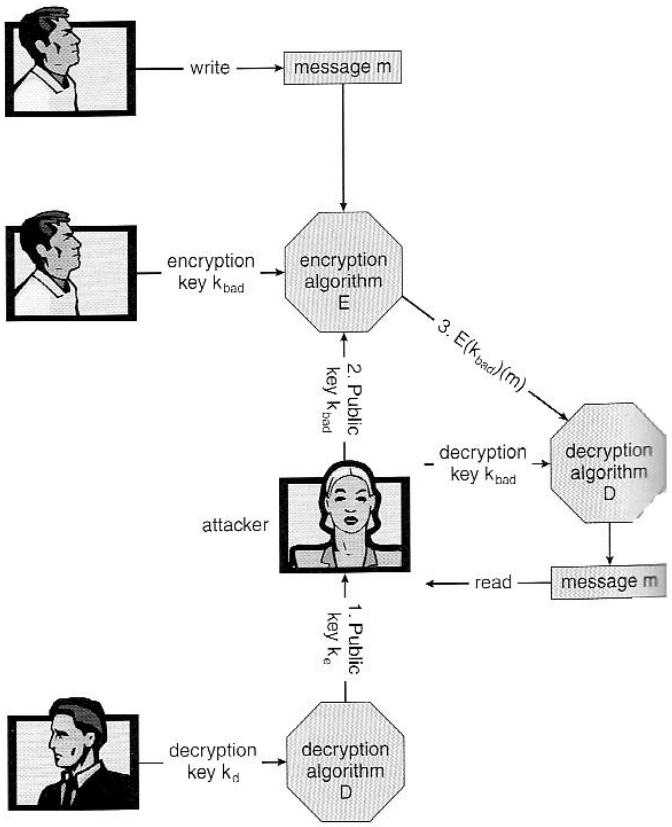


Figure 15.9 A man-in-the-middle attack on asymmetric cryptography.

User Authentication

- ***Passwords***
  - Password Vulnerabilities
  - Encrypted Passwords
  - One-Time Passwords
- ***Biometrics***

Implementing Security Defenses

- ***Security Policy***
- ***Vulnerability Assessment***
  - Port scanning.
  - Check for bad passwords. .
  - Unauthorized programs in system directories.
  - Incorrect permission bits set.
- ***Intrusion Detection***
  - Intrusion-Detection System(IDS)
  - Intrusion-Prevention System(IPS)
  - Signature based detection
  - Anomaly detection

Implementing Security Defenses

- Virus Protection : ***Disinfecting*** viruses
  - Pattern Matching
  - Sandbox
- ***Safe Computing***
- ***Auditing, Accounting, Logging***

Firewalling to Protect Systems and Networks

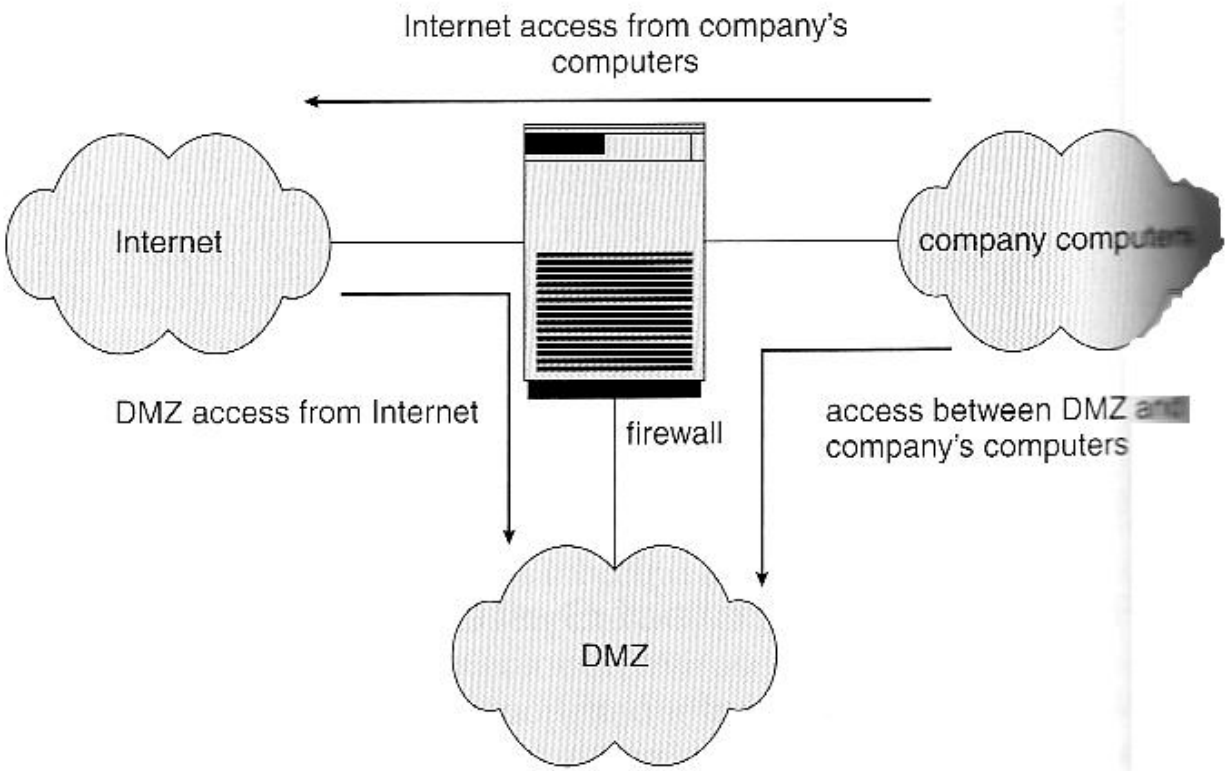


Figure 15.10 Domain separation via firewall.