大学计算机教育国外著名教材系列

影印版

Introduction to Computer Security

计算机安全导论

Michael T. [illegible]drich
Roberto Tamassia 著

清华大学出版社

大学计算机教育国外著名教材系列（影印版）

Introduction to Computer Security

计算机安全导论

Michael T. Goodrich
Roberto Tamassia
著

清华大学出版社
北京

图书在版编目(CIP)数据

计算机安全导论 = Introduction to Computer Security：英文/（美）古德里奇（Goodrich, M. T.），（美）塔玛萨（Tamassia, R.）著. --影印本. --北京：清华大学出版社，2013.1
大学计算机教育国外著名教材系列（影印版）
ISBN 978-7-302-30719-8

Ⅰ. ①计… Ⅱ. ①古… ②塔… Ⅲ. ①计算机安全－高等学校－教材－英文 Ⅳ. ①TP309

中国版本图书馆 CIP 数据核字（2012）第 278567 号

责任编辑：龙启铭
封面设计：张海清
责任印制：何　芊

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：北京四季青印刷厂
装 订 者：三河市溧源装订厂
发 行 者：全国新华书店
开　　本：185×230　　印张：36
版　　次：2013 年 1 月第 1 版　　印　　次：2013 年 1 月第 1 次印刷
印　　数：1~3000
定　　价：79.00 元

产品编号：040262-01

Contents

Chapter 1

Introduction

Contents

1.1 Fundamental Concepts

In this chapter, we introduce several fundamental concepts in computer security. Topics range from theoretical cryptographic primitives, such as digital signatures, to practical usability issues, such as social engineering. This chapter provides an informal and intuitive description of a variety of topics that will be covered in more detail in the rest of the book.

Existing computer systems may contain legacy features of earlier versions dating back to bygone eras, such as when the Internet was the sole domain of academic researchers and military labs. For instance, assumptions of trust and lack of malicious behavior among network-connected machines, which may have been justifiable in the early eighties, are surprisingly still present in the way the Internet operates today. Such assumptions have led to the growth of Internet-based crime.

An important aspect of computer security is the identification of ***vulnerabilities*** in computer systems, which can, for instance, allow a malicious user to gain access to private data and even assume full control of a machine. Vulnerabilities enable a variety of ***attacks***. Analysis of these attacks can determine the severity of damage that can be inflicted and the likelihood that the attack can be further replicated. Actions that need to be taken to defend against attacks include identifying compromised machines, removing the malicious code, and patching systems to eliminate the vulnerability.

In order to have a secure computer system, sound ***models*** are a first step. In particular, it is important to define the ***security properties*** that must be assured, anticipate the types of ***attacks*** that could be launched, and develop specific defenses. The ***design*** should also take into account usability issues. Indeed, security measures that are difficult to understand and inconvenient to follow will likely lead to failure of adoption. Next, the hardware and software ***implementation*** of a system needs to be rigorously ***tested*** to detect programming errors that introduce vulnerabilities. Once the system is deployed, procedures should be put in place to ***monitor*** the behavior of the system, detect security breaches, and react to them. Finally, security-related ***patches*** to the system must be applied as soon as they become available.

Computer security concepts often are better understood by looking at issues in a broader context. For this reason, this book also includes discussions of the security of various physical and real-world systems, including locks, ATM machines, and passenger screening at airports.

1.1.1 Confidentiality, Integrity, and Availability

Computers and networks are being misused at a growing rate. Spam, phishing, and computer viruses are becoming multibillion-dollar problems, as is identity theft, which poses a serious threat to the personal finances and credit ratings of users, and creates liabilities for corporations. Thus, there is a growing need for broader knowledge of computer security in society as well as increased expertise among information technology professionals. Society needs more security-educated computer professionals, who can successfully defend against and prevent computer attacks, as well as security-educated computer users, who can safely manage their own information and the systems they use.

One of the first things we need to do in a book on computer security is to define our concepts and terms. Classically, information security has been defined in terms of the acronym ***C.I.A.***, which in this case stands for ***confidentiality***, ***integrity***, and ***availability***. (See Figure 1.1.)

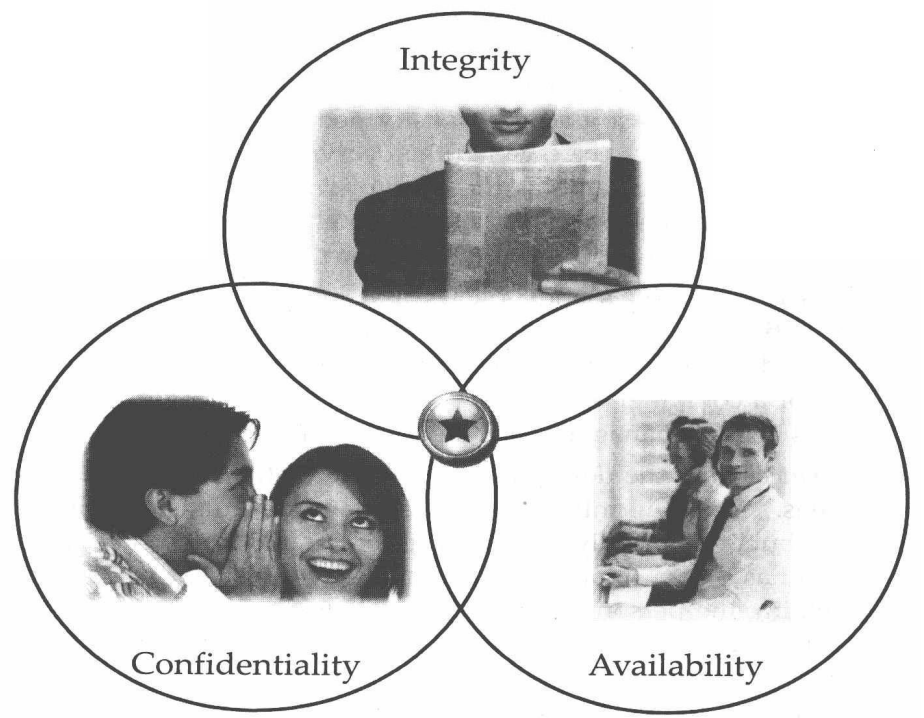

Figure 1.1: The C.I.A. concepts: confidentiality, integrity, and availability.

Confidentiality

In the context of computer security, ***confidentiality*** is the avoidance of the unauthorized disclosure of information. That is, confidentiality involves the protection of data, providing access for those who are allowed to see it while disallowing others from learning anything about its content.

Keeping information secret is often at the heart of information security, and this concept, in fact, predates computers. For example, in the first recorded use of cryptography, Julius Caesar communicated commands to his generals using a simple cipher. In his cipher, Caesar took each letter in his message and substituted D for A, E for B, and so on. This cipher can be easily broken, making it an inappropriate tool for achieving confidentiality today. But in its time, the Caesar cipher was probably fairly secure, since most of Caesar's enemies couldn't read Latin anyway.

Nowadays, achieving confidentiality is more of a challenge. Computers are everywhere, and each one is capable of performing operations that could compromise confidentiality. With all of these threats to the confidentiality of information, computer security researchers and system designers have come up with a number of tools for protecting sensitive information. These tools incorporate the following concepts:

- ***Encryption***: the transformation of information using a secret, called an encryption key, so that the transformed information can only be read using another secret, called the decryption key (which may, in some cases, be the same as the encryption key). To be secure, an encryption scheme should make it extremely difficult for someone to determine the original information without use of the decryption key.
- ***Access control***: rules and policies that limit access to confidential information to those people and/or systems with a "need to know." This need to know may be determined by identity, such as a person's name or a computer's serial number, or by a role that a person has, such as being a manager or a computer security specialist.
- ***Authentication***: the determination of the identity or role that someone has. This determination can be done in a number of different ways, but it is usually based on a combination of something the person has (like a smart card or a radio key fob storing secret keys), something the person knows (like a password), and something the person is (like a human with a fingerprint). The concept of authentication is schematically illustrated in Figure 1.2.
- ***Authorization***: the determination if a person or system is allowed access to resources, based on an access control policy. Such authorizations should prevent an attacker from tricking the system into letting him have access to protected resources.

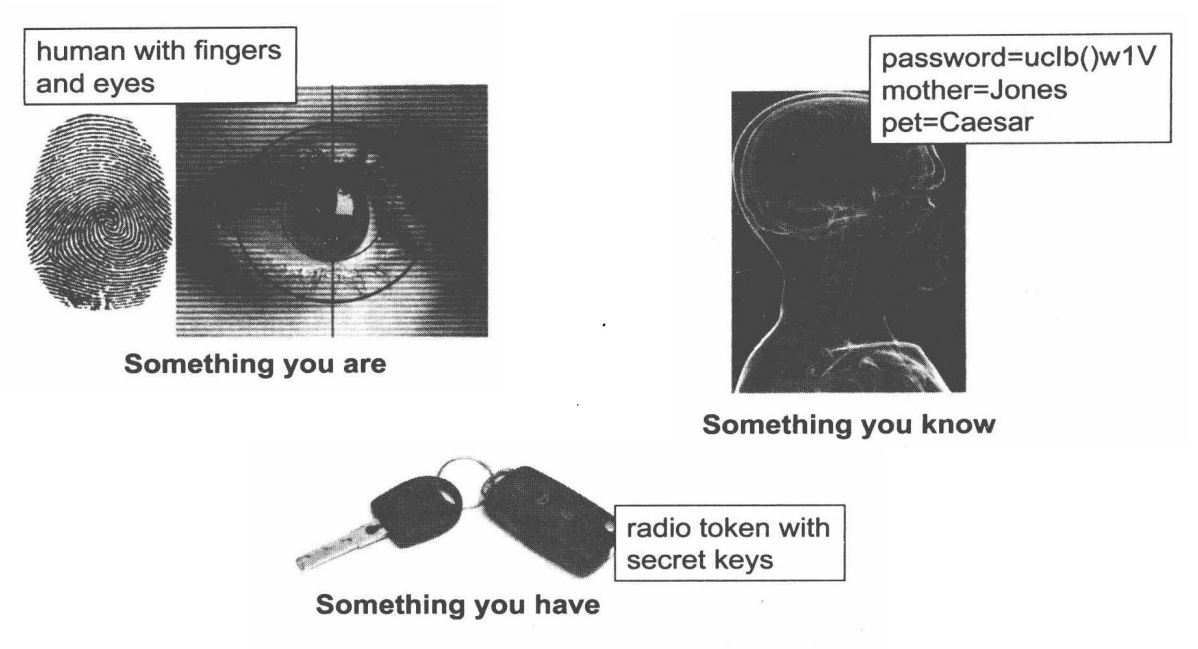

Figure 1.2: Three foundations for authentication.

- ***Physical security***: the establishment of physical barriers to limit access to protected computational resources. Such barriers include locks on cabinets and doors, the placement of computers in windowless rooms, the use of sound dampening materials, and even the construction of buildings or rooms with walls incorporating copper meshes (called ***Faraday cages***) so that electromagnetic signals cannot enter or exit the enclosure.

When we visit a web page that asks for our credit card number and our Internet browser shows a little lock icon in the corner, there is a lot that has gone on in the background to help ensure the confidentiality of our credit card number. In fact, a number of tools have probably been brought to bear here. Our browser begins the process by performing an authentication procedure to verify that the web site we are connecting to is indeed who it says it is. While this is going on, the web site might itself be checking that our browser is authentic and that we have the appropriate authorizations to access this web page according to its access control policy. Our browser then asks the web site for an encryption key to encrypt our credit card, which it then uses so that it only sends our credit card information in encrypted form. Finally, once our credit card number reaches the server that is providing this web site, the data center where

the server is located should have appropriate levels of physical security, access policies, and authorization and authentication mechanisms to keep our credit card number safe. We discuss these topics in some detail in this book.

For instance, in Section 2.4.2, we study a number of real demonstrated risks to physical eavesdropping. For example, researchers have shown that one can determine what someone is typing just by listening to a recording of their key strokes. Likewise, experiments show that it is possible to reconstruct the image of a computer screen either by monitoring its electromagnetic radiation or even from a video of a blank wall that the screen is shining on. Thus, physical security is an information security concept that should not be taken for granted.

Integrity

Another important aspect of information security is ***integrity***, which is the property that information has not be altered in an unauthorized way.

The importance of integrity is often demonstrated to school children in the ***Telephone game***. In this game, a group of children sit in a circle and the person who is "it" whispers a message in the ear of his or her neighbor on the right. Each child in the circle then waits to listen to the message from his or her neighbor on the left. Once a child has received the message, he or she then whispers this same message to their neighbor on the right. This message passing process continues until the message goes full circle and returns to the person who is "it." At that point, the last person to hear the message says the message out loud so that everyone can hear it. Typically, the message has been so mangled by this point that it is a great joke to all the children, and the game is repeated with a new person being "it." And, with each repeat play, the game reinforces that this whispering process rarely ever preserves data integrity. Indeed, could this be one of the reasons we often refer to rumors as being "whispered"?

There are a number of ways that data integrity can be compromised in computer systems and networks, and these compromises can be benign or malicious. For example, a benign compromise might come from a storage device being hit with a stray cosmic ray that flips a bit in an important file, or a disk drive might simply crash, completely destroying some of its files. A malicious compromise might come from a computer virus that infects our system and deliberately changes some the files of our operating system, so that our computer then works to replicate the virus and send it to other computers. Thus, it is important that computer systems provide tools to support data integrity.

The previously mentioned tools for protecting the confidentiality of information, denying access to data to users without appropriate access rights, also help prevent data from being modified in the first place. In addition, there are several tools specifically designed to support integrity, including the following:

- ***Backups***: the periodic archiving of data. This archiving is done so that data files can be restored should they ever be altered in an unauthorized or unintended way.

- ***Checksums***: the computation of a function that maps the contents of a file to a numerical value. A checksum function depends on the entire contents of a file and is designed in a way that even a small change to the input file (such as flipping a single bit) is highly likely to result in a different output value. Checksums are like trip-wires—they are used to detect when a breach to data integrity has occurred.

- ***Data correcting codes***: methods for storing data in such a way that small changes can be easily detected and automatically corrected. These codes are typically applied to small units of storage (e.g., at the byte level or memory word level), but there are also data-correcting codes that can be applied to entire files as well.

These tools for achieving data integrity all possess a common trait—they use ***redundancy***. That is, they involve the replication of some information content or functions of the data so that we can detect and sometimes even correct breaches in data integrity.

In addition, we should stress that it is not just the content of a data file that needs to be maintained with respect to integrity. We also need to protect the ***metadata*** for each data file, which are attributes of the file or information about access to the file that are not strictly a part of its content. Examples of metadata include the user who is the owner of the file, the last user who has modified the file, the last user who has read the file, the dates and times when the file was created and last modified and accessed, the name and location of the file in the file system, and the list of users or groups who can read or write the file. Thus, changing any metadata of a file should be considered a violation of its integrity.

For example, a computer intruder might not actually modify the content of any user files in a system he has infiltrated, but he may nevertheless be modifying metadata, such as access time stamps, by looking at our files (and thereby compromising their confidentiality if they are not encrypted). Indeed, if our system has integrity checks in place for this type of metadata, it may be able to detect an intrusion that would have otherwise gone unnoticed.