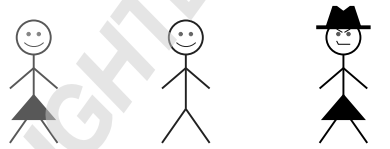
Chapter 1

Introductions

"Begin at the beginning," the King said, very gravely,
"and go on till you come to the end: then stop."
— Lewis Carroll, *Alice in Wonderland*

1.1 The Cast of Characters

Following tradition, Alice and Bob, are the good guys. Alice and Bob, who are pictured in Figure 1.1 (a) and (b), respectively, generally try to do the right thing. Occasionally, we'll require an additional good guy or two, such as Charlie or Dave. A recurring theme of this book is that stick people often make dumb mistakes, just like real people.



(a) Alice (b) Bob (c) Trudy

Figure 1.1 The main actors

Trudy, pictured in Figure 1.1 (c), is our generic bad guy who is trying to attack the system in some way. Some authors employ a team of bad guys where the name implies the particular nefarious activity. In such usage, Trudy is an "intruder," Eve is an "eavesdropper," and so on. To simplify things, we'll let Trudy be our all-purpose bad guy, although Eve might make a brief cameo appearance. Just like the bad guys in classic Hollywood Westerns, our bad guys always wear a black hat.

Alice, Bob, Trudy, and the rest of the gang need not be humans. For example, one of many possible permutations would have Alice as a laptop, Bob a server, and Trudy a human.

Information Security: Principles and Practice, Third Edition. Mark Stamp.

1.2 Alice's Online Bank

Suppose that Alice starts an online banking business, appropriately named Alice's Online Bank,[1] or AOB. What are Alice's information security concerns? If Bob is Alice's customer, what are his information security concerns? Are Bob's concerns the same as Alice's? If we look at AOB from Trudy's perspective, what security vulnerabilities might we see?

First, let's consider the traditional triumvirate of confidentiality, integrity, and availability, or CIA,[2] in the context of Alice's Bank. Then we'll point out some of the many other possible security concerns.

1.2.1 Confidentiality, Integrity, and Availability

Confidentiality deals with preventing unauthorized reading of information. AOB probably wouldn't care much about the confidentiality of the information it deals with, except for the fact that its customers certainly do. For example, Bob doesn't want Trudy to know how much money he has in his savings account. Alice's Bank would also face legal problems if it failed to protect the confidentiality of such information.

Integrity deals with preventing, or at least detecting, unauthorized "writing" (i.e., changes to data). Alice's Bank must protect the integrity of account information to prevent Trudy from, say, increasing the balance in her account or changing the balance in Bob's account. Note that confidentiality and integrity are not the same thing. For example, even if Trudy cannot read the data, she might be able to modify it, which, if undetected, would destroy its integrity. In this case, Trudy might not know what changes she had made to the data (since she can't read it), but she might not care—sometimes just causing trouble is good enough for Trudy.

Denial of service, or DoS, attacks are a relatively recent concern. Such attacks try to reduce access to information. As a result of the rise in DoS attacks, data availability has become a fundamental issue in information security. Availability is a concern for both Alice's Bank and Bob—if AOB's website is unavailable, then Alice can't make money from customer transactions and Bob can't get his business done. Bob might then take his business elsewhere. If Trudy has a grudge against Alice, or if she just wants to be malicious, she might attempt a denial of service attack on AOB.

1.2.2 Beyond CIA

Confidentiality, integrity, and availability are only the beginning of the information security story. Beginning at the beginning, consider the situation when AOB's customer Bob logs on to his computer. How does Bob's computer determine that "Bob" is really Bob and not Trudy? And when Bob logs

[1] Not to be confused with "Alice's Restaurant" [52].

[2] No, not *that* CIA.

into his account at Alice's Online Bank, how does AOB know that "Bob" is really Bob, and not Trudy? Although these two authentication problems appear to be similar on the surface, under the covers they are almost completely different.

Authentication on a standalone computer often requires that Bob's password be verified. To do so securely, some clever techniques from the field of cryptography are required. On the other hand, authentication over a network is open to many kinds of attacks that are not usually relevant on a standalone computer. Potentially, the messages sent over a network can be viewed by Trudy. To make matters worse, Trudy might be able to intercept messages, alter messages, and insert messages of her own making. If so, Trudy can simply replay Bob's old messages in an effort to, say, convince AOB that she is really Bob. As a result, authentication over a network requires careful attention to protocol, that is, the composition and ordering of the exchanged messages. Cryptography also plays a critical role in security protocols.

Once Bob has been authenticated by AOB, then Alice must enforce restrictions on Bob's actions. For example, Bob can't look at Charlie's account balance or install new accounting software on the AOB system. However, Sam, the AOB system administrator, can install new software. Enforcing such restrictions falls under the broad rubric of authorization. Note that authorization places restrictions on the actions of authenticated users. Since authentication and authorization both deal with issues of access to various computing and network resources, we'll lump them together under the clever title of access control.

All of the information security mechanisms discussed so far are implemented in software. And, if you think about it, other than the hardware, is there anything that is not software in a modern computing system? Today, software systems tend to be large, complex, and rife with bugs. A software bug is not just an annoyance, it is a potential security issue, since it may cause the system to misbehave. Of course, Trudy loves misbehavior.

What software flaws are security issues, and how are they exploited? How can AOB be sure that its software is behaving correctly? How can AOB's software developers reduce (or, ideally, eliminate) security flaws in their software? We'll examine these software development-related questions (and much more) in this book.

Although bugs can (and do) give rise to security flaws, these problems are created unintentionally by well-meaning developers. On the other hand, some software is written with the intent of doing evil. Examples of such malicious software, or malware, includes the all-too-familiar computer viruses and worms that plague the Internet today. How do these nasty beasts do what they do, and what can Alice's Online Bank do to limit their damage? What can Trudy do to increase the nastiness of such pests? We'll consider these and related questions.

Of course, Bob has many software concerns, too. For example, when Bob enters his password on his computer, how does he know that his password has not been captured and sent to Trudy? If Bob conducts a transaction at `www.alicesonlinebank.com`, how does he know that the transaction he sees on his screen is the same transaction that actually goes to the bank? That is, how can Bob be confident that his software (not to mention the network) is behaving as it should, instead of as Trudy would like it to behave? We'll consider these sorts of questions as well.

1.3 About This Book

Lampson [69] believes that real-world security boils down to the following:

- Specification/policy — What is the system supposed to do?
- Implementation/mechanism — How does it do it?
- Correctness/assurance — Does it really work?

Your humble author would humbly[3] add a fourth category:

- Human nature — Can the system survive "clever" users?

The focus of this book is primarily on the implementation/mechanism front. Your self-assured author assures you that this is appropriate, nay essential, for an introductory course, since the strengths, weaknesses, and inherent limitations of the mechanisms directly affect all other aspects of security. In other words, without a reasonable understanding of the mechanisms, it is not possible to have an informed discussion of other relevant security issues.

The material in this book is divided into four major parts. The first part deals with cryptography, while the next part covers access control. Part III shifts the focus to network security, where the emphasis is on security protocols. The final major part of the book deals with the vast and relatively ill-defined topic of software. Hopefully, the previous discussion of AOB[4] has convinced you that these major themes are all relevant to real-world information security.

In the remainder of this chapter, we'll give a quick preview of each of these four major themes. The chapter concludes with a summary, followed by several not-to-be-missed homework problems.

1.3.1 Cryptography

Cryptography is a fundamental tool in information security. Cryptography has many uses, including providing confidentiality and integrity, among other vital information security functions. We'll discuss cryptography in detail, as a working knowledge of crypto basics is essential background for any informed discussion of information security.

[3] This sentence is brought to you by the Department of Redundancy Department.

[4] You did read that, right?

We'll begin our coverage of cryptography with a look at a handful of classic cipher systems. In addition to their obvious historical and entertainment value, these classic ciphers illustrate the fundamental principles that are employed in modern digital cipher systems, but in a more user-friendly format.

With this background, we'll be prepared to study modern cryptography. Symmetric key cryptography and public key cryptography are the two major branches of cryptography, and each plays a prominent role in information security. We'll spend an entire chapter on symmetric ciphers, and another chapter on public key systems. We then turn our attention to cryptographic hash functions, which are another fundamental security tool. Hash functions are used in many different contexts, some of which are surprising, or even bordering on the counterintuitive (e.g., blockchain).

Then we'll briefly consider a few special topics that are related to cryptography. For example, we'll discuss steganography, where the goal is, essentially, to hide information in plain sight.

1.3.2 Access Control

As mentioned above, access control deals with authentication and authorization. In the area of authentication, we'll consider many issues related to passwords. Passwords are the most oft-used form of authentication today, but this is primarily because passwords are cheap, and definitely not because they are the most secure option.[5]

We'll consider how to securely store passwords. Then we'll delve into the issues surrounding secure password selection and related issues. In real world systems, passwords often represent a major security vulnerability.

The alternatives to passwords include biometrics and various physical devices, such as smartcards. We'll consider some of the security benefits of these alternate forms of authentication. In particular, we'll discuss several biometric authentication techniques.

Recall that authorization deals with restrictions placed on authenticated users. The two classic methods for enforcing such restrictions are so-called access control lists[6] and capabilities. We'll look at the plusses and minuses of each of these methods.

Authorization leads naturally to a few relatively specialized topics. We'll discuss multilevel security, which leads us into the rarified air of security modeling. We also discuss covert channels and inference control, which are challenging issues to deal with in practical systems.

[5] If someone asks you why a specific weak security measure is used when better options are available, the correct answer is usually "money," or it might simply be due to an inability to overcome inertia.

[6] Access control list, or ACL, is one of many overloaded terms that arise in the field of information security.

1.3.3 Network Security

Our third major topic is network security, where our emphasis is on security protocols. First, we provide a general introduction to networking, with special attention to the security issues that arise. This includes a discussion of firewalls, for example.

Then we consider the problems that arise when authenticating over a network. Many examples are provided, each of which illustrates a particular security pitfall. For example, replay attacks are a critical issue, and hence we consider effective ways to prevent such attacks.

Cryptography is an essential ingredient in authentication protocols. We'll give examples of protocols that use symmetric cryptography, as well as examples that rely on public key cryptography. Hash functions also have an important role to play in security protocols.

Our study of simplified authentication protocols will illustrate some of the many subtleties that can arise in this field—a seemingly insignificant change can completely change the security of a protocol. We'll also highlight a variety of specific techniques that are commonly used in real-world security protocols.

Then we'll move on to study several real-world security protocols. First, we look at the so-called Secure Shell, or SSH, which is a relatively simple example. Next, we consider the Secure Sockets Layer, or SSL, which is used extensively to secure e-commerce on the Internet. The SSL protocol is elegant and efficient, and it is well designed for its specific purpose.

We also discuss IPsec, which is another Internet security protocol. Conceptually, SSL and IPsec share many similarities, but the implementations differ greatly. In contrast to SSL, IPsec is complex—it's often said to be over-engineered. Due to its complexity, some fairly significant security issues are present in IPsec. The contrast between SSL and IPsec illustrates some of the inherent challenges in designing security protocols.

Another real-world protocol that we'll consider is Kerberos, which is an authentication system based on symmetric cryptography. Kerberos follows a much different approach than either SSL or IPsec.

We'll also discuss two wireless security protocols, WEP and GSM. Both of these protocols have many security flaws, including problems with the underlying cryptography, as well as issues with the protocols themselves. These issues make both of these topics interesting case studies.

1.3.4 Software

In the final part of the book, we'll take a look at some aspects of security that are specifically related to software. This is a huge topic, yet the two chapters in this book manage to hit on most of the fundamental issues. For starters, we'll discuss security flaws and malware, which were mentioned above. We'll also consider software reverse engineering, which illustrates how a dedicated attacker can deconstruct software, even without access to the source code.

1.4 The People Problem

Users are surprisingly capable when it comes to unintentionally inflicting damage on security systems. For example, suppose that Bob wants to purchase an item from, say, `amazon.com`. Bob can use his Web browser to securely contact Amazon using the SSL protocol (discussed in Part III), which relies on various cryptographic techniques (see Part I). Access control issues arise in such a transaction (Part II), and all of these security mechanisms are enforced in software (Part IV). So far, so good. However, we'll see that there is a practical attack on this transaction that Trudy can conduct, which will cause Bob's Web browser to issue a warning. If Bob heeds the warning, Trudy's attack will be foiled. Unfortunately, the odds are good that Bob will ignore the warning, which has the effect of negating this sophisticated security architecture. That is, the security can be broken due to user error, even if the cryptography, protocols, access control, and software all performed flawlessly.

To take just one more example, consider passwords. Users want to choose easy to remember passwords, but this also makes it easier for Trudy to guess passwords. A possible solution is to assign strong passwords to users. However, this is generally a bad idea since it is likely to result in passwords being written down and posted in prominent locations, likely making the system less secure than if users were allowed to choose their own (weaker) passwords.

As mentioned above, the primary focus of this book is on understanding security mechanisms—the nuts and bolts of security. Yet in several places throughout the book, various "people problems" arise. It would be possible to write several volumes on this topic, but the bottom line is that, from a security perspective, we would like to remove humans from the equation as much as is humanly possible.

For more information on the role that humans play in information security, a good source is Ross Anderson's book [3]. Anderson's book is filled with case studies of security failures, many—if not most—of which have at least one of their roots somewhere in the actions of the supposed good guys, Alice and Bob. While we expect Trudy to do bad things, surprisingly often the actions of Alice and Bob serve to help, rather than hinder, Trudy.

1.5 Principles and Practice

This book is not a theory book. While theory certainly has its place, in your opinionated author's opinion, many aspects of information security are not yet ripe for a meaningful theoretical treatment.[7] Of course, some topics are inherently more theoretical than others. But even relatively theoretical

[7]Consider, for example, the infamous buffer overflow attack. Historically, this one of the most serious security flaws of all time. What is the grand theory behind this particular exploit? There isn't any—it's essentially made possible by a quirk in the way that memory is laid out in modern processors.

security topics can be learned to a reasonable depth without diving too deeply into the theory. For example, cryptography can be (and often is) taught from a highly mathematical perspective. However, with rare exception, a little elementary math is all that is needed to understand cryptographic principles.

This book is certainly not an attacker's how-to guide either. Nevertheless, your practical author has consciously tried to keep the focus on real-world issues, but at a deep enough level to give the reader some understanding of—and appreciation for—the underlying concepts. The goal is to get into some depth without overwhelming the reader with excessive trivial details. Admittedly, this is a delicate balancing act and, no doubt, many will disagree that a proper balance has been struck. In your defensive author's defense, it should be noted that this book touches on a very large number of security issues related to a wide variety of fundamental principles. This breadth necessarily comes at the expense of some rigor and detail.

For those who yearn for a more theoretical treatment of the some of the topics covered here, Bishop's book [10] is the obvious choice. There are numerous fine books and articles available that focus in more detail on the various security topics discussed in this book. Your favorite search engine will quickly reveal many such sources.

1.6 Problems

The problem is not that there are problems. The problem is expecting otherwise and thinking that having problems is a problem.
— Theodore I. Rubin

1. Among the fundamental challenges in information security are confidentiality, integrity, and availability, or CIA.
   a) Define each of the terms confidentiality, integrity, and availability.
   b) Give a concrete example where both confidentiality and integrity are critically important.
   c) Give a concrete example where integrity is more important than confidentiality.
   d) Give a concrete example where availability is the overriding concern.
2. From a bank's perspective, which is likely to be more important (and why), the integrity of its customer's data or the confidentiality of the data? From the perspective of the bank's customers, which is more important (and why)?
3. Some authors distinguish between secrecy, privacy, and confidentiality. In this usage, secrecy is equivalent to our use of the term confidentiality, whereas privacy is secrecy applied to personal data, and confidentiality

(in this misguided sense) is somewhat more restrictive than the terminology as used in this book, as it refers to an obligation not to divulge certain information.

a) Discuss a real-world situation where privacy is an important security issue.

b) Discuss a real-world situation where confidentiality (in this restricted sense) is a critical security issue.

4. Cryptography is sometimes said to be "brittle," in the sense that it can be very strong, but when it breaks, it's strength is shattered.[8] In contrast, some security features can "bend" without breaking completely—security may be lost as a result of such bending, but some useful level of security can remain.

a) Other than cryptography, give an example of a security mechanism that is brittle.

b) Provide an example of a security mechanism that is not brittle, that is, the security can bend without completely breaking.

5. Read Diffie and Hellman's classic paper [30].

a) Briefly summarize the paper.

b) Diffie and Hellman give a system for distributing keys over an insecure channel (see Section 3 of the paper). How does this system work?

c) Diffie and Hellman also conjecture that a "one way compiler" might be used to construct a public key cryptosystem. Do you believe this is a plausible approach? Why or why not?

6. The most famous cipher of World War II is the German Enigma. This cipher was broken by the Allies and intelligence gained from Enigma messages proved invaluable. At first, the Allies were very careful when using the information gained from broken Enigma messages—sometimes the Allies did not use information that could have given them an advantage. However, later in the war, the Allies (and, in particular, the Americans) were much less careful, as they tended to use virtually all information obtained from broken Enigma messages.

a) Briefly discuss a significant World War II event where broken Enigma messages played a major role.

b) The Allies were cautious about using information gained from broken Enigma messages for fear that the Germans would realize their cipher was compromised. Discuss two different approaches that the Germans might have taken if they had realized that the Enigma was broken.

[8] Shadoobie [116].

c) At some point, it should have become obvious to the Germans that the Enigma was broken, yet the cipher was used until the end of the war. Why did the Nazis continue to use the Enigma?

7. When you want to authenticate yourself to your computer, most likely you type in your username and password. The username is considered public knowledge, so it is the password that authenticates you. Your password is "something you know."
   a) It is also possible to authenticate based on "something you are," that is, a physical characteristic. Such a characteristic is known as a biometric. Give an example of biometric-based authentication.
   b) It is also possible to authenticate based on "something you have," that is, something in your possession. Give an example of authentication based on something you have.
   c) Two-factor authentication requires that two of the three authentication methods (something you know, something you have, something you are) be used. Give an example from everyday life where two-factor authentication is used. Which two of the three "somethings" are used?
8. CAPTCHAs [133] are often used in an attempt to restrict access to humans (as opposed to automated processes).
   a) Give a real-world example where you were required to solve a CAPTCHA to gain access to some resource. What did you have to do to solve the CAPTCHA?
   b) Discuss various technical methods that might be used to break the CAPTCHA you described in part a) of this problem.
   c) Outline a non-technical method that might be used to attack the CAPTCHA from part a).
   d) How effective is the CAPTCHA in part a)? How user-friendly is the CAPTCHA?
   e) Do you hate solving CAPTCHAs as much as your easily-annoyed author?
9. Suppose that a particular security protocol is well designed and secure. However, there is a fairly common situation where insufficient information is available to complete the security protocol. In such cases, the protocol fails and, ideally, communication between the participants, say, Alice and Bob, should not be allowed to occur. However, in the real world, protocol designers must decide how to handle cases where protocols fail and, as a practical matter, both security and convenience must be considered. Comment on the relative merits of each of the following solutions to protocol failure. Be sure to mention the relative security and user-friendliness of each.

a) When the protocol fails, a brief warning is given to Alice and Bob, but communication is allowed to continues as if the protocol had succeeded, without any intervention required from either Alice or Bob.

b) When the protocol fails, a warning is given to Alice and she decides (by clicking a checkbox) whether communication is allowed to continue or not.

c) When the protocol fails, a notification is given to Alice and Bob and the protocol terminates.

d) When the protocol fails, the protocol terminates, with no explanation given to Alice or Bob.

10. Automatic teller machines (ATMs) are an interesting case study in security. Anderson [3] claims that when ATMs were first developed, most attention was paid to high-tech attacks. However, most real-world attacks on ATMs were decidedly low tech.

    a) Examples of high-tech attacks on ATMs would include breaking the encryption or authentication protocol. If possible, find a real-world case where a high-tech attack on an ATM has actually occurred and provide the details.

    b) Shoulder surfing is an example of a low-tech attack. In a shoulder-surfing scenario, Trudy stands behind Alice in line and watches the numbers Alice presses when entering her PIN. Then Trudy bonks Alice in the head and takes her ATM card. Give another example of a low-tech attack on an ATM that has actually occurred in the real world.

11. Large and complex software systems invariably have many bugs.

    a) For honest users, such as Alice and Bob, buggy software is certainly annoying but why is it a security issue?

    b) Why does Trudy love buggy software?

12. Malware is software that is intentionally malicious, that is, malware is designed to do damage or break the security of a system. Malware comes in many familiar varieties, including viruses, worms, and Trojans.

    a) Has your computer ever been infected with malware? If so, what did the malware do and how did you get rid of the problem? If not, how have you been so lucky?

    b) In the past, most malware was designed to annoy users. Today, it is believed (with good evidence) that most malware is written for profit. How could malware possibly be profitable?

13. In the movie *Office Space*, software developers attempt to modify company software so that for each financial transaction, any leftover fraction

of a cent goes to the software developers, instead of staying where it belongs—with the company. The idea is that for any particular transaction, nobody will notice the missing fraction of a cent, but over time the developers will accumulate a large sum of money. This type of attack is sometimes known as a *salami attack*.

a) Discuss a real-world example of a salami attack.

b) In the movie, the salami attack fails. Why?

14. It has been said that "complexity is the enemy of security".

a) Give an example of commercial software to which this statement applies, that is, find an example of software that is large and complex and has had significant security problems.

b) Find a security protocol to which this statement applies.

15. Suppose that this textbook was sold online (as a PDF) by your money-grubbing author for, say, $5. Then the author would make more money off each copy sold than he currently does[9] and people who purchase the book would save a lot of money.

a) What are the security issues related to the sale of an online book?

b) How could you make the selling of an online book more secure, from the copyright holder's perspective?

c) How secure is your approach in part b)? How user-friendly is your approach in part b)? What are some possible attacks on your proposed system?

16. The PowerPoint slides at [135] describe a security class project where students successfully hacked the Boston subway system.

a) Summarize each of the various attacks. What was the crucial vulnerability that enabled each attack to succeed?

b) The students planned to give a presentation at the self-proclaimed "hacker's convention," Defcon. At the request of the Boston transit authority, a judge issued a temporary restraining order that prevented the students from talking about their work. Do you think this was justified, based on the material in the slides?

c) What are war dialing and war driving? What is war "carting"?

d) Comment on the production quality of the "melodramatic video about the warcart" (a link to the video can be found at [124]).

[9] Believe it or not.