Chapter 4

Basic Concepts

In the preceding two chapters, we introduced a modal logic that we will use throughout this book to describe and reason about various aspects of access control. This access-control logic and its collection of inference rules provide a rigorous and formal basis for answering the question “Should this request be granted?”

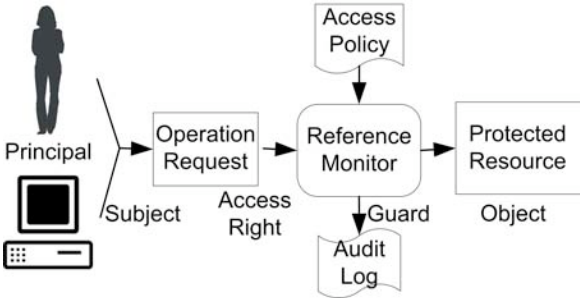
In learning any language for the first time, it’s important to start using the language to describe objects and concepts of interest to us. We begin this chapter by introducing the concept of a reference monitor, which provides a context for the remainder of this chapter. We then develop the fundamental access-control concepts of tickets, capabilities, and access-control lists. We also discuss various authentication mechanisms and how they relate to access control. Throughout this chapter, there are numerous examples to illustrate how the access-control logic is used.

4.1 Reference Monitors

Access control is about guarding objects, assets, resources, services, communications, files, databases, information, or people. In everyday life, physical locks of various forms guard against unauthorized access to all sorts of resources, ranging from house doors and high-school lockers to automobile ignitions and safety-deposit boxes. In other situations, a person or machine may require us to provide an artifact (e.g., subway token or bus pass) to gain access to particular resources. For example, we need a boarding pass to get on a bus, train, or airplane. We need to have a ticket before we can get in to see a movie. If we are invited to a formal dinner such as a wedding reception, there may be a guest list and name cards at each seat in the banquet hall.

In computer and information systems, the guards are called *reference monitors*, and they protect resources such as memory, files, databases, programs, and communication channels. Figure 4.1 provides an abstract view of how a reference monitor fits into a system. Initially, a principal—either a person, machine, or process—makes a request to the reference monitor to perform some operation on a protected object. Principals that make requests are often referred to as *subjects*, and the right to perform an operation on an object is typically called an *access right* or *privilege*. The reference monitor’s decision regarding whether or not to grant the request depends on

FIGURE 4.1 Abstract view of reference monitors



several factors: what is being asked for, potentially the identity and other attributes of the subject making the request, and finally the access policy. When a decision is made, it is recorded in an audit log where a record of all the access requests and decisions are kept.

All reference monitors must satisfy three essential properties:

1. *Completeness:* The reference monitor cannot be bypassed.

   Completeness means that there is no way to access a protected resource without coming under the scrutiny of the guard or reference monitor. Physically, completeness means that there is no “hole in the fence.” In information systems, completeness means that all paths to a state in which access is granted go through an access controller.

2. *Isolation:* The reference monitor is tamper proof.

   Isolation is primarily satisfied by physical protection mechanisms. For example, machines on which sensitive information is stored must be housed in locked and guarded machine rooms where access to the machine rooms is constantly watched.

3. *Verifiable:* The reference monitor is correctly implemented.

   The property of being verified correct means that we must have fully and precisely specified what the reference monitor is supposed to do.

Conceptually, the desired access-control behavior of a system can be easily represented by *an access-control matrix*, as introduced by Lampson (Lampson, 1971). The rows of an access-control matrix M are labeled by possible subjects (i.e., principals), while the columns are labeled by the protected objects. Each entry $M_{s,o}$ in the matrix indicates the access rights that subject s possesses with respect to the object o. As an example, Table 4.1 contains a simple access-control matrix for a system involving three subjects (with identifiers `alice`, `bob`, `carol`) and four protected objects (files numbered 1 through 4). According to this table, the principal `alice` has the rights to read $file_1$, to read or write $file_2$, and to execute $file_4$; `alice` has no

	file$_1$	file$_2$	file$_3$	file$_4$
alice	read	read, write		execute
bob		read	execute	
carol	read, write		execute	execute

Table 4.1: Example of an access-control matrix

access rights for *file*$_3$. In contrast, bob has no access rights for *file*$_1$ or *file*$_4$, but can read *file*$_2$ and execute *file*$_3$.

When assigning rights to principals, it is wise to follow the *principle of least privilege* (Saltzer and Schroeder, 1975): principals should be assigned only the rights necessary for completing their tasks and no more.

Access-control matrices provide a very simple and straightforward way to describe systems, particularly those that have relatively small numbers of subjects and objects. However, they alone are insufficient for reasoning about access control in more complex systems that involve large numbers of subjects and objects or that permit proxies and delegation. In particular, they do not address issues related to how subjects are authenticated or the trust assumptions that underlie the authentication process.

The purpose of the access-control logic is to bridge this gap, to help us to deduce and to specify what a reference monitor's behavior should be for any given policy and request. From a logical point of view, a reference monitor decides whether or not to approve an access request based on the inference rules of the access-control logic and within the context of an access policy and trust assumptions.

As we have already seen, we can use the logic to describe a subject's request to access a particular object as follows:

$$\mathit{Subject}\ \textsf{says}\ \langle \mathit{access\ right,\ object}\rangle .$$

To grant the request, the reference monitor must be able to determine that the access policy authorizes the subject to perform the requested operation. In terms of the logic, this determination amounts to using the inference rules of the logic to derive the atomic proposition

$$\langle \mathit{access\ right,\ object}\rangle$$

from assumptions about the given policy and any additional trust assumptions that are relevant to the situation.

Policy statements state who or what has jurisdiction over certain statements, as well as who or what is authorized with certain rights. These statements usually take the form

$$P\ \textsf{controls}\ \varphi,$$

where P is the authority and φ is the statement over which P has jurisdiction.

Trust assumptions are usually assumptions about proxies or symbols of authority, such as the queen's seal, a principal's public key, or tickets issued by an airline.

These statements usually have the form

$$P \Rightarrow Q,$$

where P is the recognized proxy for principal Q. As we will see in the subsequent sections, these statements express explicitly the reference monitor's reliance on the unforgeability of certificates and credentials.

A *certificate* is a signed statement made by a recognized authority, such as a birth certificate (signed perhaps by a town's health department) or a driver's license (signed by a state department of motor vehicles). A *credential* is a certificate that asserts a particular principal's authorization to perform some action; for example, a driver's license asserts a given driver's authorization to drive a certain class of vehicle.

The next section develops the specifics of requests, policy statements, and trust assumptions within the context of tickets and access-control lists. Subsequent sections introduce the notions of certificates and credentials, particularly as they apply to both identity-based and group-based authentication.

4.2 Access-Control Mechanisms: Tickets and Lists

As we have already seen, reference monitors guard those objects deemed necessary to protect. There are two common protection schemes employed by reference monitors: one is based on *tickets* (or *capabilities*), while the other depends on *access-control lists*.

Tickets signify a *capability* to access a resource. For example, a ticket to a movie theater grants the holder of a theater ticket access to a seat in a particular theater at a particular time. Tickets are presumed to be unforgeable—only the appropriate controlling authorities are able to issue tickets. In ticket-oriented protection schemes, principals possess tickets for the resources or objects to which they have access. Principals gain access to protected resources or objects by presenting the appropriate ticket to the reference monitors protecting those resources.

In list-oriented protection schemes, an *access-control list* has the names of principals who are allowed to access the resource being protected. An example application of access-control lists is a restaurant where patrons are seated at dining tables only if they have a reservation. The access-control list itself must also be protected: that is, only certain principals are recognized to have the authority to adjust whose names appear on the access-control list.

In this section, we formalize the concepts of tickets and access-control lists using the access-control logic.

4.2.1 Tickets

Tickets are a common means to control access to protected objects. For example, when airline passengers prepare to board an airplane, the gate agents check each passenger's ticket or boarding pass to see if the person presenting the ticket is authorized to board the airplane. The gate agents are the reference monitors guarding access to the airplane's doors. The boarding pass signifies the capability to access a particular flight. Ideally, it is an unforgeable document issued by the airline (i.e., the controlling authority) granting permission to board a particular airplane at a particular time.

In general, ticket-oriented access control requires the following four components, which we express as statements in the logic:

1. The access policy:

   $$\mathit{authority}\ \textsf{controls}\ (\mathit{subject}\ \textsf{controls}\ \langle \mathit{access\ right,\ object}\rangle)$$

   The policy statement asserts that the controlling authority has jurisdiction over which subjects can exercise an access right on an object.

2. The ticket:

   $$\mathit{ticket}\ \textsf{says}\ (\mathit{subject}\ \textsf{controls}\ \langle \mathit{access\ right,\ object}\rangle)$$

   A ticket is a credential stating that the subject has the right to access a particular object.

3. Additional trust assumption:

   $$\mathit{ticket} \Rightarrow \mathit{authority}$$

   This assumption states that tickets are tokens of authority issued by the controlling authority. As such, it is imperative that these tokens or tickets be unforgeable.

4. The access request:

   $$\mathit{subject}\ \textsf{says}\ \langle \mathit{access\ right,\ object}\rangle$$

   The access request occurs when the subject actually presents the ticket. For example, an airline passenger presenting a ticket is requesting to sit at her assigned seat on the designated flight. Of course, a subject may request access without actually presenting a ticket, but the reference monitor should not grant access in that case. For example, a person in the air terminal who tries to board an airplane without presenting a valid ticket should not be allowed to do so.

Together, these four components provide sufficient information for the reference monitor to determine that the requested action $\langle \mathit{access\ right,\ object}\rangle$ should be permitted. If we generalize the atomic proposition $\langle \mathit{access\ right,\ object}\rangle$ to an arbitrary

FIGURE 4.2 Formal proof of the Ticket Rule

1. $\textit{subject}\ \textsf{says}\ \varphi$	Access request
2. $\textit{authority}\ \textsf{controls}\ (\textit{subject}\ \textsf{controls}\ \varphi)$	Access policy
3. $\textit{ticket} \Rightarrow \textit{authority}$	Trust assumption
4. $\textit{ticket}\ \textsf{says}\ (\textit{subject}\ \textsf{controls}\ \varphi)$	Ticket
5. $\textit{authority}\ \textsf{says}\ (\textit{subject}\ \textsf{controls}\ \varphi)$	3, 4 speaks for
6. $\textit{subject}\ \textsf{controls}\ \varphi$	2, 5 controls
7. φ	6, 1 controls

statement φ, we can derive the following useful inference rule:

$$\textit{Ticket}\quad \frac{\begin{array}{c}\textit{subject}\ \textsf{says}\ \varphi \quad \textit{authority}\ \textsf{controls}\ (\textit{subject}\ \textsf{controls}\ \varphi) \\ \textit{ticket} \Rightarrow \textit{authority} \quad \textit{ticket}\ \textsf{says}\ (\textit{subject}\ \textsf{controls}\ \varphi)\end{array}}{\varphi}.$$

The formal proof of this rule appears in Figure 4.2. The next example illustrates how the Ticket Rule applies to the airline-ticket scenario.

Example 4.1

Suppose Tina has an airplane ticket assigning her to seat 25D on Smooth Air Flight #1. When her row is called, she presents her ticket to the gate agents for flight #1, and she is granted access.

The following analysis justifies the decision by the gate agents to let Tina board the airplane:

1. Tina $\textsf{says}$ $\langle$seat 25D, flight #1$\rangle$	Tina's request
2. Smooth Air $\textsf{controls}$ (Tina $\textsf{controls}$ $\langle$seat 25D, flight #1$\rangle$)	Access policy
3. ticket $\Rightarrow$ Smooth Air	Trust assumption
4. ticket $\textsf{says}$ (Tina $\textsf{controls}$ $\langle$seat 25D, flight #1$\rangle$)	Tina's ticket
5. $\langle$seat 25D, flight #1$\rangle$	1, 2, 3, 4 ticket rule

Line 1 represents Tina's request to board flight #1 and sit in seat 25D, which happens when Tina walks up to the gate agent and presents her ticket. Line 2 indicates that the gate agent recognizes Smooth Air's authority over which subjects can access flight #1. Line 3 states that tickets represent the authority of Smooth Air; that is, the gate agent assumes that Smooth Air will issue tickets only to people who should be passengers on flight #1. Implicit in this line is that the gate agent recognizes the ticket as being a valid ticket issued by Smooth Air. For example, if Tina were to present a piece of notebook paper that has her name and a seat assignment scribbled on it, the gate agent would not accept that paper as representing the authority of Smooth Air.

Line 4 describes what the ticket says, namely that Tina is assigned to seat 25D on flight #1. Line 5—which is obtained from the previous lines through the Ticket Rule—demonstrates that the gate agent's decision to allow Tina to board flight #1 is justified, given the previous conditions. ◊

In the preceding example, the boarding pass really is a ticket in the technical sense: the gate agent checks only the validity of the boarding pass and does not confirm Tina's identity. Most likely, anyone who handed Tina's valid boarding pass to the gate agent would be allowed on the plane, regardless of their actual identity.

✎ **Exercise 4.2.1** *Suppose a theater ticket is sold by the box office to a patron to see* Gone with the Wind *in Theater 5 at 7:30 p.m. Using the access-control logic, describe the patron's request, the access-control policy of the theater, the trust assumptions, and movie ticket. Based on your descriptions, formally justify admitting the patron to see the movie.*

4.2.2 Lists

Another widely used access-control mechanism is a *list* of principals with their access rights to protected objects. These lists are known as *access-control lists* (ACLs).

In access-control list schemes, reference monitors possess the list of authorized users and their privileges. Principals identify themselves and make their requests to access the protected resources. The reference monitor compares the request to the access-control list: if there is a match, then access is granted. Unlike ticket-oriented access control, principals do not possess a credential that says they have a right to access a protected resource.

ACLs are a very common protection mechanism. For example, if a scientist wants to visit a secure government or commercial facility, she must call ahead to ensure that the guards at the gates of the facility have her name on a visitor's list. When she arrives at the gate, she identifies herself to the guard, who checks her identity and looks to see if she is on the guest list. If her name is on the list, then she is allowed into the facility. A more benign but common example is a wedding or formal party where a greeter (acting as the reference monitor) checks off guest names on a guest list.

In general, protection schemes involving access-control lists require the following components:

1. Access policy:

$$\mathit{authority}\ \textsf{controls}\ (\mathit{subject}\ \textsf{controls}\ \langle \mathit{access\ right,\ object}\rangle)$$

   The guard or reference monitor recognizes the jurisdiction of some controlling authority to decide who is allowed what access to the protected object.

2. Access-control list:

$$\mathit{ACL}\ \textsf{says}\ \begin{cases} \mathit{subject}_1\ \textsf{controls}\ \langle \mathit{access\ right}_1, \mathit{object}_1\rangle \wedge \\ \mathit{subject}_2\ \textsf{controls}\ \langle \mathit{access\ right}_2, \mathit{object}_2\rangle \wedge \\ \cdots \wedge \\ \mathit{subject}_n\ \textsf{controls}\ \langle \mathit{access\ right}_n, \mathit{object}_n\rangle \end{cases}$$

An ACL is essentially a listing of subjects and their access rights.

3. Trust assumption:

$$\textit{ACL} \Rightarrow \textit{authority}$$

This assumption states that the ACL *always* reflects the wishes of the authority. As is the case with tickets, it is imperative that the ACL's integrity be assured.

4. Access request:

$$\textit{subject} \text{ says } \langle \textit{access right, object} \rangle$$

The request occurs when the subject presents herself to the guard (who can establish her identity, as necessary) and makes the request.

With the exception of the form of the access-control list, these components are the same as those for tickets. Authorities determine who has access, and trust assumptions are needed to interpret statements from authority. It is within this context that subjects' requests are considered.

The only difference is in the statements directly associated with the ticket and with the access-control list. A ticket associates a specific subject with a specific authorization:

$$\textit{ticket} \text{ says } (\textit{subject}_i \text{ controls } \varphi_i)$$

In contrast, an access-control list associates many subjects with many authorizations, having the following general form:

$$\textit{ACL} \text{ says } \begin{cases} \textit{subject}_1 \text{ controls } \varphi_1 \wedge \\ \dots \\ \textit{subject}_i \text{ controls } \varphi_i \wedge \\ \dots \end{cases}$$

However, it is possible to extract from an access-control list the relevant authorization, using the *Says Simplification* rules introduced in Exercise 3.2.5:

$$\frac{P \text{ says } (\varphi_1 \wedge \varphi_2)}{P \text{ says } \varphi_1} \qquad \frac{P \text{ says } (\varphi_1 \wedge \varphi_2)}{P \text{ says } \varphi_2}$$

Using a combination of these two rules, we can extract (for any i)

$$\textit{ACL} \text{ says } (\textit{subject}_i \text{ controls } \varphi_i)$$

from the access-control list description.

It is therefore possible to use the *Ticket Rule* to also reason about access-control lists. The following example is an illustration.

Example 4.2

Suppose Erika is invited to a dinner meeting where delicate negotiations are to be held. Because of the sensitive nature of the talks, the meeting is being held in a private dining room. At the door of the dining room is a maître d' with a guest list of all the attendees. The guest list, which was authorized by the restaurant's manager, has three names: Erika, Darnell, and Gina.

To gain entry to the dining room, Erika identifies herself to the maître d' and he lets her in. For simplicity, we will assume that the maître d' recognizes Erika, Darnell, and Gina by sight, so that no further identification is needed. (We will consider more realistic and complicated situations later in this chapter.). The following analysis justifies the decision of the maître d' to let Erika into the private dining room:

1. $\text{Erika } \mathsf{says}\ \langle\text{enter, dining room}\rangle$	Erika's request
2. $\text{Manager } \mathsf{controls}\ (\text{Erika } \mathsf{controls}\ \langle\text{enter, dining room}\rangle)$	Access policy
3. $\text{ACL} \Rightarrow \text{Manager}$	Trust assumption
4. $\text{ACL } \mathsf{says} \begin{cases} \text{Erika } \mathsf{controls}\ \langle\text{enter, dining room}\rangle \wedge \\ \text{Darnell } \mathsf{controls}\ \langle\text{enter, dining room}\rangle \wedge \\ \text{Gina } \mathsf{controls}\ \langle\text{enter, dining room}\rangle \end{cases}$	Guest List
5. $\text{ACL } \mathsf{says}\ (\text{Erika } \mathsf{controls}\ \langle\text{enter, dining room}\rangle)$	4 simplify says
6. $\langle\text{enter, dining room}\rangle$	1, 2, 3, 5 ticket rule

Line 1 represents Erika's spoken request to be let into the private dining room. Line 2 reflects the restaurant's policy that the manager in charge has authority over who can enter the private dining room. Line 3 represents the maître d's belief that the guest list is authorized by the manager and accurately reflects the manager's wishes. Line 4 describes the guest list—a list of subjects who can exercise entry rights to the private dining room. Line 5 is obtained by repeated application of the *Simplify Says* inference rule to Line 4, and Line 6 is obtained by the Ticket Rule. ◊

When expressing ACLs in the logic, it is important to remember that the two formulas

$$(P\ \mathsf{controls}\ \varphi_1)\ \wedge\ (P\ \mathsf{controls}\ \varphi_2)$$

and

$$P\ \mathsf{controls}\ (\varphi_1 \wedge \varphi_2)$$

are *not* equivalent (see Exercise 3.3.7). In the former case, P is trusted on *each* of φ_1 and φ_2; P's request to perform *either* should be granted. In the latter case, however, P is trusted on the *conjunction* of φ_1 and φ_2: P must request *both* in order for either to be granted. For this reason, an ACL will often contain many separate clauses for the same principal, detailing all of the individual access rights that principal has.

Similarly, the two formulas

$$(P\ \mathsf{controls}\ \varphi)\ \wedge\ (Q\ \mathsf{controls}\ \varphi)$$

and

$$P\ \&\ Q\ \mathsf{controls}\ \varphi$$

are not equivalent. The first formula states that both P and Q are *individually* authorized for φ. In contrast, the second formula states that both P and Q must request φ for access to be granted. For instance, in Example 4.2, Erika, Darnell, and Gina are each authorized to enter the private dining room, and each may enter the room without the others. Suppose instead that the ACL were defined in the following way:

$$\text{ACL}\ \mathsf{says}\ \mathit{Erika}\ \&\ \mathit{Darnell}\ \&\ \mathit{Gina}\ \mathsf{controls}\ \langle\text{enter, dining room}\rangle.$$

This case would correspond to restaurants that seat parties only after everyone has arrived: Erika, Darnell, and Gina could enter the dining room only if all three make the request to do so.

Exercise 4.2.2 *In operating systems with discretionary access control, a user can specify who else can read, write, or execute her files. Suppose Carter creates a program* foo*, and he wants Dan to be able to execute* foo *but neither read* foo *nor write to it. Also, Carter wishes to grant read, write, and execute privileges to Jan. Assume that Dan and Jan have userids* `dan` *and* `jan`*, respectively. Formalize the above description and formally justify why Jan's request to execute* foo *should be granted.*

Exercise 4.2.3 *Prove that the formulas*

$$P\ \&\ Q\ \mathsf{controls}\ \varphi, \qquad (P\ \mathsf{controls}\ \varphi)\ \wedge\ (Q\ \mathsf{controls}\ \varphi)$$

are not equivalent. That is, find a particular formula φ *and a particular Kripke structure* $\mathcal{M} = \langle W, I, J\rangle$ *such that*

$$\mathcal{M} \not\models (P\ \&\ Q\ \mathsf{controls}\ \varphi) \equiv ((P\ \mathsf{controls}\ \varphi)\ \wedge\ (Q\ \mathsf{controls}\ \varphi)).$$

4.2.3 Logical and Pragmatic Implications

We have looked in detail at ticket-oriented and list-oriented mechanisms for controlling access to protected objects. As we have seen, tickets and lists differ in the following two ways:

1. Tickets are typically possessed—or easily accessible—by principals wishing to access a protected object. In contrast, ACLs are possessed by reference monitors.

2. Typically, when tickets are used, the reference monitor does not need to identify the principal making the request. Instead, the principal possessing the ticket needs only give it to the reference monitor to exercise the rights corresponding to the ticket. In contrast, a reference monitor using an ACL must somehow determine the identity of the principal requesting access to determine what rights, if any, the principal has.

Despite these differences, tickets and ACLs share similar logical meanings and logical contexts:

1. Logically speaking, tickets and ACLs make statements of the same form:

   $$\begin{gathered} \mathit{ticket} \text{ says } (\mathit{subject} \text{ controls } \varphi) \\ \mathit{ACL} \text{ says } (\mathit{subject} \text{ controls } \varphi) \end{gathered}$$

   In the case of tickets, the *subject* may simply be *Bearer*, in that anyone bearing or possessing the ticket has the corresponding access right. In the case of ACLs, specific principals are generally identified.

2. Both tickets and ACLs are used in the context of policies that specify which authority has jurisdiction over access decisions. These policies regarding jurisdiction are expressed in the logic as follows:

   $$\mathit{authority} \text{ controls } (\mathit{subject} \text{ controls } \varphi)$$

3. Both tickets and ACLs are the mechanisms for controlling access and thus must be protected from fraud. As a result, both must faithfully represent the desires of the authorities they represent. Statements regarding this faithful representation of the authority are expressed in the logic as follows:

   $$\begin{gathered} \mathit{ticket} \Rightarrow \mathit{authority} \\ \mathit{ACL} \Rightarrow \mathit{authority} \end{gathered}$$

From a logical standpoint, it is no accident that reference monitors depend upon the same inference rules to deduce whether an access request should be granted. Although ticket-based access may appear on the surface to be very different from list-based access, the information conveyed by tickets and lists is the same, as is the larger context of policy statements and trust assumptions.

There is an important pragmatic consideration regarding the use of the derived *Ticket Rule* in our examples. Recall that the ticket rule is as follows:

$$\mathit{Ticket\ Rule} \quad \frac{\begin{gathered}\mathit{subject} \text{ says } \varphi \quad \mathit{authority} \text{ controls } (\mathit{subject} \text{ controls } \varphi) \\ \mathit{ticket} \Rightarrow \mathit{authority} \quad \mathit{ticket} \text{ says } (\mathit{subject} \text{ controls } \varphi)\end{gathered}}{\varphi}$$

From an implementation standpoint, an engineer designing a reference monitor *will not* implement a general purpose inference engine or theorem prover based on the inference rules of the access-control logic. Instead, what he or she will build amounts to a mechanization of a derived inference rule specific to his or her situation.

In particular, many reference monitors are essentially implemented as *checklists*. They determine whether the various necessary tickets and certificates are present; if so, access is granted. Such approaches can be justified by derived inference rules where the elements on the checklist are the *premises* and the desired action on an object is the *conclusion*. Furthermore, there is usually a specific *context* within which the reference monitor operates. These contexts will sometimes provide *implicit* premises, rather than *explicit* checklist items.

For example, suppose that Margaret is an engineer building a reference monitor for a copy machine. The intention is that, if a person has a ticket—perhaps a card he inserts into the copy machine—then he can make copies. In the actual implementation, Margaret's chief concern is to develop a way to check the validity of an inserted copy card; if it is valid, then a copy operation should be performed whenever the copy button is pressed. The verified validity of the copy card amounts to a statement of the form

$$\mathit{card} \textsf{ says } (\mathit{bearer} \textsf{ controls } \mathit{copy}),$$

while each press of the copy button amounts to the following statement:

$$\mathit{bearer} \textsf{ says } \mathit{copy}.$$

The *Ticket* rule allows Margaret to justify her design formally, but only by explicitly stating the implicit assumptions about what card validity means. Specifically, the analysis requires her to acknowledge an authority that governs who can make copies and that the copy card must speak for that authority:

$$\mathit{auth} \textsf{ controls } (\mathit{bearer} \textsf{ controls } \mathit{copy}), \quad \mathit{card} \Rightarrow \mathit{auth}.$$

The value of such an analysis is that potentially implicit assumptions are made explicit. If there are future changes to the operating context (e.g., the card-making equipment is stolen or compromised), then it is easier to determine whether or not systems need to be reconfigured or even redesigned.

4.3 Authentication

Authentication is the task of identifying a subject who is making a request. More abstractly, authentication is the task of associating one principal (e.g., the process requesting access to a guarded resource or the face and person requesting access to a guarded facility) with another principal (e.g., an individual or group included on an ACL).

In this section we take a detailed look at several authentication scenarios and describe them in the access-control logic.

4.3.1 Two-Factor Authentication

Authentication is generally based on one or more kinds of information or *factors*:

- Type 1: something you *know*, such as a PIN, or password.
- Type 2: something you *have*, such as card, key, or token.
- Type 3: something you *are*, such as your fingerprint, face, or voice.

Authentication can be performed using one or more items of information drawn from each factor. *Two-factor authentication* uses information drawn from two of the preceding types. An example of two-factor authentication based on information drawn from types 1 and 2 is identifying a principal based on a personal identification number PIN (type 1) and a badge or certificate (type 2).

Many badges have the principal's name and the PIN associated with the principal in protected form on the badge. A typical security badge might have a person's name, face, and security clearance on the front of the badge, and have the name, security clearance, and PIN stored in encrypted form accessible by a magnetic card reader. Most cards have both physical and electronic security. Physical security measures include the way the card was manufactured: the person's picture is part of the plastic, there is a stamp or signature written over visible information such as security clearances and names, and so on. Electronic security can include encryption and signing of electronic information such as names, PINs, and security clearances. If a smart card is used, key information will be wiped out if physical tampering is attempted.

For an older picture ID with a magnetic stripe that contains a protected PIN, we can express the two-factor authentication as follows:

$$\text{authority } \mathsf{says}\ ((\text{face}_{subject}\ \&\ \text{PIN}_{subject}) \Rightarrow \text{subject}).$$

We recognize that the above formula is exactly the form of a certificate (i.e., a signed statement) issued by an authority: the certificate in this case associates a particular face and PIN with a particular subject or principal. The ID badge is used by the reference monitor as a certificate. Again, it is crucial that badges and other certificates be impossible—or at least very difficult—to counterfeit.

Figure 4.3 contains an analysis of two-factor authentication based on certificates. Line 1 represents the subject using factor_1 and factor_2 to make a request φ. Line 2 is the access policy that states that authority_1 is in charge of who has access. Line 3 corresponds to an access-control list entry stating that *subject* has access right φ. Line 4 state that authority_2 has jurisdiction over certificates identifying principals using factor_1 and factor_2. Line 5 is the information contained in the certificate or badge possessed by the *subject*. Lines 6 through 9 are derived using the inference rules of the access-control logic.

The two-factor proof of Figure 4.3 corresponds to the following derived inference rule:

$$\textit{Two Factor Auth}\quad \frac{\begin{array}{c}(\text{factor}_1\ \&\ \text{factor}_2)\ \mathsf{says}\ \varphi \\ \text{authority}_1\ \mathsf{controls}\ (\text{subject}\ \mathsf{controls}\ \varphi) \\ \text{authority}_1\ \mathsf{says}\ (\text{subject}\ \mathsf{controls}\ \varphi) \\ \text{authority}_2\ \mathsf{controls}\ ((\text{factor}_1\ \&\ \text{factor}_2) \Rightarrow \text{subject}) \\ \text{authority}_2\ \mathsf{says}\ ((\text{factor}_1\ \&\ \text{factor}_2) \Rightarrow \text{subject})\end{array}}{\varphi}.$$

The following example demonstrates the use of this derived rule for reasoning about access based on two-factor authentication.

FIGURE 4.3 Template for two-factor authentication

1. $(\text{factor}_1 \,\&\, \text{factor}_2)$ says φ	Access request
2. authority_1 controls (subject controls φ)	Access policy
3. authority_1 says (subject controls φ)	ACL entry
4. authority_2 controls $((\text{factor}_1 \,\&\, \text{factor}_2) \Rightarrow \text{subject})$	Jurisdiction
5. authority_2 says $((\text{factor}_1 \,\&\, \text{factor}_2) \Rightarrow \text{subject})$	Certificate
6. $(\text{factor}_1 \,\&\, \text{factor}_2) \Rightarrow \text{subject}$	4, 5 Controls
7. subject says φ	6, 1 Says
8. subject controls φ	2, 3 Controls
9. φ	8, 7 Controls

Example 4.3

Omar works for a military research lab. The security office has issued Omar a badge ($badge_{Omar}$) and personal identification number (PIN_{Omar}). The doors to the lab are protected by card readers with keypads. To enter the lab, Omar must swipe his badge in the card reader and then punch in his PIN. If Omar uses his card and punches in his PIN, then the reference monitor controlling the door is able to identify him, look him up on an electronic access-control list, and determine that he is an employee in good standing and should be admitted.

This description corresponds to two-factor authentication. Omar must know his PIN (a type-1 factor) and then he must possess a badge (a type-2 factor). The combination of the two are used to identify Omar to the electronic reference monitor that guards the lab doors.

As is the case with many badges, the magnetic stripe on Omar's card contains protected information, such as a badge identifier and Omar's PIN. The card reader on the doors can identify both the badge identifier and protected PIN.

The following proof justifies letting Omar enter the lab when he presents his badge and PIN.

1. $(\text{badge}_{Omar} \,\&\, \text{PIN}_{Omar})$ says $\langle\text{enter, lab}\rangle$	Access request
2. security office controls (Omar controls $\langle\text{enter, lab}\rangle$)	Access policy
3. security office says (Omar controls $\langle\text{enter, lab}\rangle$)	ACL entry
4. security office controls $((\text{badge}_{Omar} \,\&\, \text{PIN}_{Omar}) \Rightarrow \text{Omar})$	Jurisdiction
5. security office says $((\text{badge}_{Omar} \,\&\, \text{PIN}_{Omar}) \Rightarrow \text{Omar})$	Info in badge
6. $\langle\text{enter, lab}\rangle$	Two Factor ACL

The preceding scenario is largely analogous to the use of bank cards and PINs by automated teller machines. The important difference is that withdrawals also depend on conditions related to the account balances.

4.3.2 Using Credentials from Other Authorities

So far we have described access scenarios that depend on credentials issued by the same authority responsible for the protected resource. In many situations, however, this need not be the case. For example, airlines routinely use state-issued driver's licenses with pictures and federally issued passports to identify passengers. Guards at airports use a combination of government-issued documents with pictures (such as passports and licenses) coupled with airline boarding passes.

The informal justification that allows such credentials to be used is based on trust in the process by which these credentials are issued. For example, government-issued picture IDs are accepted based on the assumption that it is hard to dupe the government into issuing a fake picture ID. That is, the government's registration process for issuing picture IDs is deemed to be sufficiently robust. Likewise, airline tickets are accepted as credentials because airlines have strong economic interests to control who can get on their airplanes: it is consequently difficult to defraud an airline and obtain a ticket without paying for it (either with money or frequent-flyer miles).

A typical goal when making access-control decisions based on other credentials is two-fold: first, determine whether the subject making the request is who they say they are, and then determine whether they have access rights to some other object that is meaningful and relevant to you.

A template for this type of situation appears in Figure 4.4. Lines 1 through 9 are the initial assumptions. Line 1 describes a subject with a face, voice, fingerprint, or userid and password, making a request φ_1. Lines 2 and 3 are the recognition of the jurisdiction of authority A_1 regarding φ_1 and the stated access policy. Note that the policy expressed in line 3 differs in form from previous examples: it states that, if the subject has access to the object corresponding to φ_2, then the subject also has access to the object corresponding to φ_1. For example, if Penny has permission to board an airplane leaving from a given airport, then Penny has permission to enter that airport. Lines 4 through 6 relate to recognizing the jurisdiction of authority A_2 on associating a factor with a subject (factor $\Rightarrow$ subject), the credential itself, and its assumed integrity. Similarly lines 7 through 9 relate to recognizing the jurisdiction of A_3 to control φ_2 and the integrity of the certificate authorizing *subject* to control φ_2.

The proof in Figure 4.4 yields the following derived inference rule:

$$\textit{ID \& Ticket}\quad \frac{\begin{array}{c}\text{factor says } \varphi_1\\ A_1 \text{ controls } ((\text{subject controls } \varphi_2) \supset \text{subject controls } \varphi_1)\\ A_1 \text{ says } ((\text{subject controls } \varphi_2) \supset \text{subject controls } \varphi_1)\\ A_2 \text{ controls } (\text{factor} \Rightarrow \text{subject})\\ \text{certificate says } (\text{factor} \Rightarrow \text{subject})\\ \text{certificate} \Rightarrow A_2\\ A_3 \text{ controls } (\text{subject controls } \varphi_2)\\ \text{ticket says } (\text{subject controls } \varphi_2)\\ \text{ticket} \Rightarrow A_3\end{array}}{\varphi_1}.$$

To be explicit, this template illustrates one particular way that credentials from multiple authorities might fit together as part of an access-control scheme. Not all

FIGURE 4.4 Template for using credentials from multiple authorities

1. factor says φ_1	Request
2. A_1 controls ((subject controls φ_2) $\supset$ subject controls φ_1)	Jurisdiction
3. A_1 says ((subject controls φ_2) $\supset$ subject controls φ_1)	Access policy
4. A_2 controls (factor $\Rightarrow$ subject)	Jurisdiction
5. certificate says (factor $\Rightarrow$ subject)	ID card
6. certificate $\Rightarrow A_2$	Trust assumption
7. A_3 controls (subject controls φ_2)	Jurisdiction
8. ticket says (subject controls φ_2)	Ticket
9. ticket $\Rightarrow A_3$	Trust assumption
10. subject controls $\varphi_2 \supset$ subject controls φ_1	2, 3 Controls
11. A_2 says (factor $\Rightarrow$ subject)	6, 5 Derived speaks for
12. factor $\Rightarrow$ subject	4, 11 Controls
13. subject says φ_1	12, 1 Derived speaks for
14. A_3 says (subject controls φ_2)	9, 8 Speaks for
15. subject controls φ_2	7, 14 Controls
16. subject controls φ_1	15, 10 Modus ponens
17. φ_1	16, 13 Controls

situations will necessarily fit into this specific template. The following example applies the *ID & Ticket* rule to a common situation that does fit this template: the task of controlling access to an airport terminal where only ticketed passengers with DMV-issued driver's licenses can enter.

Example 4.4

Penny is a ticketed passenger on flight #1. To gain entry to the airport terminal, she must show a state-issued picture ID along with her boarding pass. When she reaches airport security, she presents her driver's license. She also presents her boarding pass for flight #1. The following formalization justifies why Penny is allowed to enter the airport terminal:

1. Face says $\langle$enter, airport$\rangle$	Request
2. Security controls ((Penny controls $\langle$25D, flight #1$\rangle$) $\supset$ Penny controls $\langle$enter, airport$\rangle$)	Jurisdiction
3. Security says ((Penny controls $\langle$25D, flight #1$\rangle$) $\supset$ Penny controls $\langle$enter, airport$\rangle$)	Access policy
4. DMV controls (Face $\Rightarrow$ Penny)	Jurisdiction
5. license says (Face $\Rightarrow$ Penny)	ID card
6. license $\Rightarrow$ DMV	Trust assumption
7. Airline controls (Penny controls $\langle$25D, flight #1$\rangle$)	Jurisdiction
8. ticket says (Penny controls $\langle$25D, flight #1$\rangle$)	Ticket
9. ticket $\Rightarrow$ Airline	Trust assumption
10. $\langle$enter, airport$\rangle$	1,2,3,4,5,6,7,8,9 ID & Ticket Rule

Line 1 is the request as interpreted by airport security: a specific face is making a request to enter the airport terminal. Line 2 corresponds to the security guard at the airport recognizing the authority of airport security to make policy. Line 3 is the actual policy stating that, if the person presenting the ticket is really Penny (i.e., her face matches the face on the driver's license), then Penny can enter the airport. Lines 4 through 6 correspond to the guard recognizing the authority of the state DMV, seeing Penny's driver's license with her picture on it, and accepting the license as authentically issued by the DMV. Lines 7 through 9 have a similar interpretation with respect to the airline and Penny's airplane ticket. $\Diamond$

Exercise 4.3.1 *Meg keeps a safe-deposit box (box #1205) at her local bank, in which she stores important documents. She can access those documents by visiting the bank, getting the safe-deposit box, and then opening it with her bank-issued keycard ($card_M$) and PIN (PIN_M).*

At the bank, access to safe-deposit boxes is governed as follows:

- *As a strategy to prevent theft, safe-deposit boxes can be removed from the vault only through the cooperation of the Vault Officer and the Teller Supervisor. Specifically, the Vault Officer and the Teller Supervisor must simultaneously insert special keys (key_{VO} and key_{TS}, respectively) and enter the specific box's number.*

- *Once removed from the vault, the safe-deposit box is brought to the account holder (in this case, Meg), who may open the box by inserting her keycard and entering the correct PIN number.*

Let $\langle release, \#1205\rangle$ and $\langle open, \#1205\rangle$ be (respectively) the operations of releasing box 1205 from the vault and of opening box 1205. Use expressions in the access-control logic to answer the following questions regarding the certifications, credentials, and access-control policies needed for the bank's safe-deposit system.

a. What are the specific access requests that are placed to release box 1205 from the vault and to open box 1205?

b. What are the ACL (access-control list) entries that govern the vault's release and Meg's opening of box 1205?

c. What are the additional certificates, recognition of authority, and trust assumptions that are necessary for determining whether to grant release or open requests for box 1205?

d. Suppose that Meg's safe-deposit box is brought to her, and that her keycard is valid. Using the relevant assumptions from Parts (1), (2), and (3) above, give a formal proof that the request $\langle open, \#1205\rangle$ will be granted.

Exercise 4.3.2 *Suppose you are the chief operating officer of a rental car company. One of your primary objectives is to make sure you rent cars only to qualified drivers.*

a. Informally describe your method for assessing whether a domestic customer is qualified to rent a car, what evidence you will use, how you will check the authenticity of the evidence, and who you trust in the process.

b. Formally describe your method using the access-control logic.

c. Most domestic rental car companies rent cars to international customers but require additional credentials. Describe your process for establishing the qualifications of international customers.

d. Formally describe your method using the access-control logic.

4.3.3 Groups

In many situations, access control is *group based*, rather than *individual based*. That is, access-control decisions may be based on whether the requester belongs to a specific group, rather than on the specific identity of the individual. As a common example, access to the teacher's lounge in a high school is typically granted to all members of the faculty.

In reality, groups are simply another sort of principal, and group-based access policies are defined in the same way as identity-based policies. For example, the following statement expresses the access policy for the high-school teacher's lounge:

$$\mathit{Faculty}\ \textsf{controls}\ \langle \mathit{enter}, \mathit{lounge}\rangle.$$

Group membership can be expressed using the speaks-for relationship, as in the following statement:

$$\mathit{Leslie} \Rightarrow \mathit{Faculty}.$$

Using the *Derived Controls* rule, it is straightforward to deduce that any request from Leslie to enter the teacher's lounge should be granted.

Many everyday documents are certificates that assert group membership of one kind or another. Alec's United States passport is a certificate issued by the U.S. State Department that asserts that Alec is a U.S. citizen:

$$\mathit{US\ State\ Department}\ \textsf{says}\ \mathit{Alec} \Rightarrow \mathit{US\ Citizen}.$$

Miranda's student identification card, issued by New State University (NSU) when she first enrolled, is a certificate stating that she is an NSU student:

$$\mathit{NSU}\ \textsf{says}\ \mathit{Miranda} \Rightarrow \mathit{NSU\ Student}.$$

Exercise 4.3.3 *Oliver has recently become a Premium member of the frequent-flyer club of Fly-By-Night Airlines. The membership package he received from Fly-By-Night included a membership card with his name and frequent-flyer number printed on it. As a Premium member, Oliver is eligible to enter the Fly-By-Night lounges at participating airports. On his next trip, Oliver decides to use this benefit: he* *heads to the Fly-By-Night lounge and presents his membership card and a photo identification (his driver's license) to the staffer who is guarding the door.*

Formally describe the access-policy of the lounge, Oliver's request to enter, and any other necessary trust assumptions required for Oliver to be granted access to the lounge. Based on these descriptions, formally justify letting Oliver into the Fly-By-Night lounge.

Exercise 4.3.4 *Sam is a citizen and resident of the United States, returning home after visiting Canada. She arrives at the US border with her passport in hand.*

a. The act of driving her car to the border agent's station can be interpreted as a request to enter the United States. Express this request in the access-control logic.

b. Sam's passport can be interpreted as a certificate (i.e., a signed statement) from the U.S. State Department that associates Sam's face with her name. It also identifies Sam as belonging to the group of U.S. citizens.

Formalize Sam's passport as an expression in the access-control logic.

c. The standard policy is that US citizens with proof of citizenship may enter the country, provided the border agent deems them not to be a threat (i.e., the border agent may exercise discretion).

Formalize this policy in the access-control logic.

d. Suppose that the border agent deems Sam not to be a threat.

Identify any other necessary assumptions (expressed as formulas in the access-control logic) about certificates, jurisdiction, et cetera to justify letting Sam enter the United States.

e. Using only those expressions given in the previous parts of the question, give a formal proof justifying Sam's entry into the United States.

4.4 Summary

Our goal is to be able to justify an answer to the question "Should this access-control request be granted?" Answering this question correctly relies on sorting through the maze of tickets, certificates, credentials, access-control lists, and various means of authentication. This analysis further depends upon the authorities that are recognized and the extent of their jurisdiction.

In this chapter, we demonstrated how to describe the mechanisms (e.g., tickets, ACLs, certificates) and the broader context (e.g., recognized authorities and jurisdictions) in our access-control logic. We also demonstrated several derived rules that support reasoning about access requests. In the process, we developed templates for

FIGURE 4.5 Learning outcomes for Chapter 4

After completing this chapter, you should be able to achieve the following learning outcomes at several levels of knowledge:

Application

- Express requests, access-control policies, credentials, and notions of jurisdiction in the access-control logic.
- Express tickets, access-control lists, authentication factors, and group membership in the access-control logic.

Synthesis

- When given a scenario that involves tickets, lists, and individual or group-based authentication, formalize the scenario, identify all necessary trust assumptions, and formally justify the granting of a request.

analyzing a wide variety of access-control scenarios. These templates demonstrate the utility of derived rules that can be reused in a variety of situations. They also illustrate how the language can be used to describe new mechanisms and concepts.

The learning outcomes associated with this chapter appear in Figure 4.5.

4.5 Further Reading

For classic papers on access control and protection, we suggest Butler Lampson's "Protection" (Lampson, 1971) and Saltzer and Schroeder's "The Protection of Information in Computer Systems" (Saltzer and Schroeder, 1975). In addition, Matt Bishop's textbook *Computer Security: Art and Science* (Bishop, 2003) provides a compendium of access-control models and devotes a chapter to authentication.